TECHNICAL REFERENCE

PAYONE Platform Channel Client API

Table of Contents

Preface

This document serves as a technical reference for the PAYONE Platform and is exclusively intended for the identified recipient. If you are not the intended recipient or the recipient's representative please notice that the content is protected by copyright and that any form of acknowledgement, publication, duplication or distribution of the contents of this document is illegal.

Brand names

All brand names are property of their respective companies. "BS PAYONE" and "more than payment." are registered trademarks of BS PAYONE GmbH.

Protection of information

This document is released on the condition of strict confidentiality. Any sharing and/or disclosure towards third parties requires the express consent in writing by BS PAYONE GmbH.

Disclaimer

This document has been prepared with the utmost care. There is, however, no warranty as to its complete accuracy.

Amendments

BS PAYONE GmbH reserves the right to make product improvements and other changes without notice.

Copyright

The copyright for this document is held by BS PAYONE GmbH, Lyoner Straße 9, 60528 Frankfurt am Main, Germany.

This document is released on the condition of strict confidentiality. Any sharing and/or disclosure towards third parties require the express consent in writing by BS PAYONE GmbH.

History of changes

Changes		
Date	**Version**	**Comments**
2012-11-16	1.0	Transaktions-/Sessionstatus-Encoding (ISO-8859-1) documented.
2012-11-16	1.0	userid, accessid extended from 8 to 12 digits
2013-01-03	1.01	new document layout
2013-01-03	1.01	list of iDEAL banks updated
2013-02-27	1.02	Sessionstatus, errata "reverse_cancel" -> "cancel_reversal" New countries added to request bankaccountcheck Note to currency handling Note to IP V4 / IP V6 New payment types BillSAFE New parameters for TransactionStatus "vsettlement"
2013-04-15	1.03	New banks added to EPS bankgrouptype New banks added to iDEAL bankgrouptype Response to addresscheck: firstname and lastname may be returned New errorcode 938 Parameter "email" is mandatory for payment type BSV Table 5.4 added with possible combinations for BankAccountCheck.
2013-05-15	1.04	New errorcode 951
2013-09-11	1.05	Updated error text for error code 3200
2013-11-07	1.06	SEPA-specific extensions parameters "iban" / "bic" can be used – even without bankcode / bankaccount **Please note that API-responses may be extended by new response key/values at any time to provide additional features**
2013-11-14	1.07	note added for PMI configuration to enable additional response parameters Usage of IBAN / BIC: IBAN/BIC can be used together with bankcountry. If bankcountry is given it must match with IBAN. bankcountry is mandatory with bankcode, bankaccount PCS-merchants: mandate-identifications must be generated automatically
2013-11-19	1.08	New error code 2012 Response "bankaccountcheck": parameter "bankcountry" was missing
2013-12-04	1.09	Requests "ManageMandate" and "getfile" are only available with Server API Request "authorization" has been copied twice into API-documentation. Duplicate request has been removed.
2013-12-10	1.10	Comments to successurl, … changed: old: only if not provided in the PMI new: only if responsetype=REDIRECT or required by corresponding request additional response parameter added for direct debit requests Direct Debit Netherlands removed from chapter "1. General"
2014-03-05	1.11	Parameter "mandate_identification" was already included into hash calculation, but was missing in documentation, chapter 3.1.4

Changes		
Date	**Version**	**Comments**
		Request "managemandate" has been added to Client-API -> new chapter 3.3 (Administration)
		Request "managemandate" is now available with BBAN (Germany only) and with userid/customerid if BBAN is attached to user
		Response data have been extended by "mandate_dateofsignature"
		Mode "live" / "test" consistently written in lower case.
		BankAccountCheck will return either "INVALID" or "ERROR" in case of invalid data.
		Error messages added and text corrected
2014-03-19	1.12	new error codes 878, 905, 909, 1007, 1367, 1372, 1373
		Parameter "telephonenumber" shortened from 50 to 30 characters
2014-09-05	1.13	new error codes 940, 941, 952, 1337, 1338, 1374, 1375
		Parameter for Klarna (KLV) added
		Parameter "shipping_addressaddition" has been removed as it is not used for any payment type
		Parameter "state" (regions) extended to these countries: US, CA, CN, JP, MX, BR, AR, ID, TH, IN
		Parameter "ecommercemode" was missing for request "authorization"
		Parameter "ecommercemode" was already used in hash-calculation, but this was missing in the documentation
		Documentation for TxStatus has been corrected: "cancellation" -> "cancelation"
		IMPORTANT NOTE: **Starting with 2014-12-29 the range of IP-addresses of the PAYONE Platform will be extended!** • **old: 213.178.72.196, 213.178.72.197, 217.70.200.0/24** • **new: 213.178.72.196, 213.178.72.197, 217.70.200.0/24, 185.60.20.0/24** **Please ensure that your systems will be able to accept outgoing and incoming connections to and from all these IP-addresses.**
2015-01-12	1.14	New error code 1012 added
		Parameter for Klarna installment (KLS) have been added
		Transaction Status PAYONE -> merchant: • new parameter "transaction_status" has been introduced (if your systems are not prepared to process new parameters, please contact our PAYONE Technical Support)
		Parameter "api_version" added -> see standard parameter
		only available with api_version >= 3.9: • New response "pending" added for "preauthorization" / "authorization"
		Request "createaccess" corrected: • "invoice_deliverydate" removed • "invoice_deliveryenddate" removed • it[n], id[n], pr[n], no[n], de[n], va[n], sd[n], ed[n], invoiceid removed • id_trail[n], no_trail[n], pr_trail[n], de_trail[n], va_trail[n], ti_trail[n] added

Changes		
Date	**Version**	**Comments**
		• id_recurring[n], no_recurring[n], pr_recurring[n], de_recurring[n], va_recurring[n], ti_recurring[n] added
2015-02-25	1.15	Transaction Status PAYONE -> merchant: • description added for parameter "notify_version" (in use since January 2015) • txaction "failed": description corrected (mistake by copy&paste). -> this txaction is not in use yet. Parameter "api_version" added to hash calculation
2015-04-08	1.16	• PCS (PAYONE Collection Service) has been migrated to PPS (PAYONE Payment Service) • Chapter 1, "please note" chapter supplemented by reference to costs • Hash parameter "api_version" was listed at the end of hashed parameters (to indicate new value). Now it has been put into alphabetical order.
2015-06-03	1.17	• Additional support of hash algorithm "sha2-384" (HMAC-SHA-2, 384) PMI-setting to enforce usage of "sha2-384" (Hash-value of SessionStatus and TransactionStatus still remains with MD5 by now and is subject to change). • Transaction Status PAYONE -> merchant , txaction "failed" is used with "Barzahlen" and expired refund. • Character set for parameter "id" (product number) has been clearly defined • The hash values (key -> chapter, key -> 4.1, 4.2) are currently given as MD5. This currently still remains with MD5 and is subject to change in future to SHA2-384.
2015-06-17	1.18	• Description of "PAYONE hosted-iFrame mode" in chapter 3.1.5. -> PCI DSS V 3.0 SAQ A compliant implementation.
2015-06-29	1.19	• Chapter 3.1.5.3 calculation of hash corrected / explained.
2015-07-01	1.20	• Chapter 3.1.5.6, additional tips added • Chapter 3.1.5, known issue added with old version of framework "prototype"
2015-07-23	1.21	• Chapter 3.1.5, typo corrected o old: new payone.ClientApi.Request(…) o new: new Payone.ClientApi.Request(…)
2015-08-06	1.22	• Parameter "reference" for GPY must be min. 4, max. 20 characters

Changes		
Date	**Version**	**Comments**

2015-12-07	1.23	• Online-Payment P24 added
		• Old IP-addresses for Session-Status and Transaction-Status removed.
		• CreateAccess: Added that "access_expiretime" and "period_unit_trail", "period_length_trail" can not be used. "access_expiretime" should not be used any more.
		• AddressCheck Person: added that firstname/lastname are mandatory
		• BankAccountCheck:
		old: Errorcode 888 (IBAN invalid) was returned with response "ERROR"
		new: Errorcode 888 (IBAN invalid) is returned with response "INVALID"
		• New error codes 972 and 973 have been added
		• Parameter "bankcode" and "clearing_bankcode" have changed from N..8 to AN..11
		• Parameter "bankaccount" and "clearing_bankaccount" have changed from AN..14 to AN..26
		• Added support IP V4/V6 for customers IP-address
		• ManageMandate:
		   o currently ManageMandate always reponds with "APPROVED" or "ERROR".
		   o in future (Q3 2016) ManageMandate will respond with "ERROR" for erroneous requests and may respond with "INVALID" for requests with invalid data
		   -> responses "APPROVED", "ERROR" and "INVALID" may be used.

2016-01-28	1.24	• itemtype (Parameter „it") now does support „voucher" for PPE (PayPal Express) • Additional test data for "consumerscore" and "address check person". Please refer to separate document "PAYONE_Platform_Testdata_EN.pdf"
2016-02-15	**1.25**	• **New SSL-certificates for PAYONE -> https://www.payone.de/en/platform-integration/platform/important-technical-information/ssl-certificates/**
2016-02-23	1.26	• Naming changed from "truncated cardpan" to "masked cardpan". Parameter name "truncated cardpan" does remain unchanged. In fact "masked cardpan" means display of first six and last four digits while "truncated cardpan" means showing only last four digits. • Actually the pseudo card pan is 13 to 16 digits long. **In future the pseudo card pan will be 19 digits long.** This is already specified by format "N..19" and in future the full range of 19 digits will be used.
2016-03-18	1.27	• **Check of correct encoding will be enforced.** I.e.: If request-encoding is specified with "UTF-8" (Parameter "encoding") and non-matching characters are detected (e.g. "ß" instead of "U+00DF" or "Ö" instead of "U+00D6") the request will be rejected with errorcode=2013.
2016-05-09	1.28	• iDEAL: Bunq-Bank added (Parameter "bankgrouptype") • Contract/createaccess: Limit (3 years) for trail and recurring period documentated. • **Chapter 3.1.3: Special remarks to usage of not-used parameters.**

Changes		
Date	**Version**	**Comments**

2016-07-26	1.29	• Additional features for "PAYONE hosted iFrame mode"
		    o Config-block for element "cardcvc2" has been extended by array "length" to specify the exact required CVC-length per cardtype
		    o response data have been extended by selected "cardtype" and used "cardexpiredate"
		    o Attribute "placeholder" can now be set for input fields.
		• Responses for "customermessage" and "errormessage" can now have a maximum length of 1024 characters (previous length was limited to 255 characters). This is required due to special payment methods.
		• American Express Safekey (3-D Secure) is planned for 2016-10-01
		• China Union Pay / CUP (as credit card) is planned for 2016-10-01
		• IBAN / BIC / bankcode / bankaccount can be omitted for SOFORTBanking (OBT/PNT) on API-level as they are returned from SOFORT and saved by PAYONE (for later upcoming SEPA-credits in case of refund).
		• only available with api_version >= 3.10:
		Response for "customermessage" can be more specific in case of error by containing detailed error messages from external payment gateways (e.g. Ratepay, …)

2016-07-27	1.30	• Example in chapter "Changing display text for error messages" corrected ("=" instead of ":")
2016-07-28	1.31	• Additional response data "cardtype" and "cardexpiredate" for creditcardcheck have been added at the JSON-sample, but were missing in the creditcardcheck-responsedata.

Changes		
Date	**Version**	**Comments**

2016-11-10	1.32	• **Parameter "id_trail" and "id_recurring" have been shortened from AN..100 to AN..32. This is necessary as these values will be used for later usage with parameter "id" in authorization-requests. And there the "id" is limited to AN..32.**
		
		• For request "addresscheck" and "consumerscore" new values "UKN" and "PUG", "PNZ", "PNP" have been added for "personstatus" in response data.
		
		• Codes for "personstatus" have been moved to chapter "Codes"
		
		• For request "consumerscore" key/value for "divergence" will not be returned in response data – required by german data protection law.
		
		• Format for request-parameter "bankcode" changed from "AN..11" to "N8" (fixed 8) for direct debit requests as only DE is supported with bankcode/bankaccount (BBAN). For international direct debit IBAN is required.
		
		• Format for request-parameter "bankaccount" changed from "AN..26" to "N..10" for direct debit requests as only DE is supported with bankcode/bankaccount (BBAN). For international direct debit IBAN is required.
		
		• Request-parameters "bankbranchcode" and "bankcheckdigit" removed. They were only valid for SEPA-countries and they have moved to IBAN.
		
		• Mistake in documentation in chapter "Responses -> TransactionStatus":
		   o "clearing_amount" was described as N..7,2 in largest unit (e.g. Euro)
		   o but is returned as N..10 in smallest unit (e.g. Cent).
		
		• Sample Code for AJAX mode (chapter 3.1.6) had a mistake:
		   o wrong: callback_function_name
		   o correct: callbackFunctionName

2016-12-02	1.33	• New error codes for Blacklist (731 to 734) • Clarification for usage of Client-API (see 3.1.6, "important note")

2017-02-15	1.34	• Added comments to URLs (successurl, backurl, errorurl) in section "special remark", to email-encoding and whitespaces in values.
		
		• New risk check available from Arvato Infoscore:
		   o consumerscoretype IF: Informa-Consumer-Score
		
		• New risk checks available from Boniversum:
		   o addresschecktype BB -> postal address check
		   o addresschecktype PB -> person address check
		   o consumerscoretype CE -> postal address check, person address check, check for special addresses, VERITA Score Premium Ident S; this check will return new "score=U" (for unknown)
		   o new response-parameter "addressstatus" added

Changes		
Date	**Version**	**Comments**
2017-04-03	1.35	- New feature for "PAYONE, hosted iFrame": automatic cardtype detection - By adding a new configuration-block "config.autoCardtypeDetection" a simple cardtype detection can be enabled. - For cardtype a new value "#" can be initially set to enforce the user to select a cardtype. - Added some PHP-samples for automatic cardtype detection - Added new bankgrouptypes for EPS payments - Contact details changed PAYONE.DE -> PAYONE.COM
2017-05-04	1.36	- Removed depricated bankgrouptypes for EPS payments - Parameter "firstname" also mandatory for payment type PDT - Please use only capital letters for country-codes and state-codes - Please use only small letters for language-codes - Added errorcodes 350 and 351 for PAYONE secure purchase on invoice
2017-07-25	1.37	- Added errorcode 981 for Amazon Payments - Tipps added to Hosted iFrames -> Table of config-attributes (3.1.5.2) - Correction: iDEAL "ABN_AMRO_BANK" is **not** deprecated - Additional test PANs to simulate credit card failures -> refer to separate test data document. - Parameter "email" is mandatory for payment type P24
2017-08-29	1.38	- Ratepay Subtypes RPD, RPP, RPS, RPV added - Design changed to BS PAYONE
2017-11-15	1.39	- TransactionStatus extended with "txaction=failed" for all types of payment - TransactionStatus extended with "errorcode" in case of "txaction=failed" - Contracts may have a duration (trail & recurring) of maximum 5 years - Address data are required for bank transfers with IBAN-countries CH, SM, MC, PM, JE, GG due to new regulation on the transfer of funds (Geldtransferverordnung) valid from 2017-11-19 - New parameter "businessrelation" introduced with values "b2b", "b2c". This will be required from 2018-01-01 to indicate business to business transactions especially for "secure invoice" (POV) and "secure direct debit" (POD) - New parameter "customer_is_present" introduced with values "yes", "no" to indicate whether the customer is online now and can enter some data.

1 General

This technical reference includes detailed descriptions and examples for the communication with the PAYONE Platform.

PAYONE Platform supports the following methods of payment:

Direct debit:	Germany, Austria
	(ELV: electronic direct debit system)
Credit card:	Visa, MasterCard, American Express, JCB, Diners Club, Discover,…
Debit card:	Maestro International, Maestro UK, Carte Bleue
Online transfer:	Sofortbanking, giropay, eps (electronic payment standards), PostFinance E-Finance, PostFinance Card, iDEAL, Przelewy24
Transfer:	Prepayment (worldwide), open invoice (worldwide), cash on delivery (worldwide)
e-wallets:	PayPal
Financing:	BillSAFE Invoice (deprecated), Klarna Invoice

PAYONE Platform includes the following optional modules:

Accounting:	Detection of incoming payments and overdue accounts that result from return debit notes, chargebacks and invoices which have not been settled by the specified date.
Contract:	Administration of subscriptions and recurring payments
Invoicing:	Generating invoices and credit memos
Collect:	Automatic recovery of overdue accounts via dunning processes and encashment
Protect:	Check of accuracy and evaluation of the submitted customer data
Reporting:	Specific export options for all transaction details
Billing:	Aggregated billing of individual purchases and subscriptions

The administration of subscriptions (Contract), the creation of invoices (Invoicing) and the dunning processes (Collect) are, depending on the settings, automatically carried out in the background. You can, however, use API to control these procedures.

The communication is based on HTTPS-POST requests (key/value pairs) between the merchant's systems and PAYONE Platform.

The PAYONE Platform and its connected systems are designed for IP addresses Version 4.

This technical reference may include functions that are not activated for your merchant account due to contractual terms. If you have any questions or problems please do not hesitate to contact our service team.

Please note:

Mode "test" / "live":

Basically all API-requests can be used in mode "test" and "live" in the same way. But please note that the processes may differ slightly different in mode "test" and "live". So in mode "test" a lot of downstreamed processes are simulated by the PAYONE Platform and are not forwarded to other service providers.

Please also note that you should not use any live data in mode "test". The PAYONE Platform offers a set of test data to simulate several test cases in payment processing. Please request the document "PAYONE Platform Test procedures and test data".

Attention: Requests in mode "live" are always processed and forwarded to service providers – even if test data are used. By this additional cost may come up (e.g. by post delivery of documents or by chargebacks).

Costs:

Please note that depending on the transaction type used costs may occur in addition to transaction fees. See our List of Prices and Services for details.

1.1 Accounts

The PAYONE Platform includes merchant accounts and what is known as sub accounts. For the settlement of your goods you need at least one sub account to which your payments will be allocated.

Each merchant account can include any number of sub accounts. This combination of merchant and sub accounts offers a multitude of flexible options to the merchant.

The merchant can, for example, allocate marketing campaigns to different sub accounts in order to receive exact statistics concerning all transactions, accesses, revenues, subscriptions and purchases generated through the corresponding marketing campaign. The merchant can thus easily measure and analyse the success of his marketing campaigns with just one merchant account.

This combination of merchant and sub accounts can also be used for multilevel marketing platforms (partner programs) or resellers.

1.2　Payment portals

In order to carry out payment processes via the PAYONE Platform, you must first create a payment portal. All settings regarding payment processes and debtor management are anchored in the payment portals. All payment processes are conducted via the different payment portals.

The PAYONE Platform has two different versions of payment portals: "Access" and "Shop".

The fundamental difference between the two payment portal versions is the following: In the "Access" version you need to set up orders/contract templates and the PAYONE Platform can handle the access management for you. You can define how long or how often your customers have access to your products and services after a successful payment process. In the same manner, subscriptions are supported by the PAYONE Platform payment portals of the version "Access".

"Access" payment portals are thus specifically useful for accounting digital products or services which will grant your customer access for a specific period of time determined by you or if the PAYONE Platform is to manage a subscription.

In payment portals of the version "Shop" a one-time settlement occurs. In this case it is not necessary to set up offers because the products and services that are to be settled are dynamically submitted to the PAYONE Platform. It is therefore possible to settle actual as well as digital products and services.

Versions:

"Access" version:　　　　Time-based settlement

(settlement of digital products and services for a specific period of time, such as memberships or subscriptions)

"Shop" version:　　　　Product /event-based settlement

(one-time settlement of actual or digital products and services)

1.3 Debtor accounts

With each initialisation of a payment process the PAYONE Platform sets up a debtor account and opens up a payment process in this account.

Each payment process includes a unique PAYONE payment process ID (txid). An individual balance is kept for each payment process. A payment process usually includes an invoice and, where applicable, several credit memos. All payments or return debit notes are automatically allocated to the corresponding payment process. Once a payment request is settled, the balance is reduced by the corresponding amount. In the case of return debit notes or chargebacks the balance is increased by the amount of the return debit notes.

During each booking the master data/payment data for the customer is saved. Each customer (debtor) is assigned a PAYONE debtor ID (userid) by the PAYONE Platform. If you enter the PAYONE debtor ID assigned by the PAYONE Platform (userid) for follow-up bookings for the same debtor, the booking will automatically be assigned to the same debtor.

The second option is to use your own customer ID (customerid). If you use the same customer ID (customerid) for two different bookings, the bookings will also be allocated to the same internal debtor by the PAYONE Platform.

Advantage:

All payment processes by the same debtor are managed automatically internally. Among other things, this makes it possible to synchronise the booking, dunning and encashment processes by combining several open requests for one debtor within one process. In addition, the payment processes or the master data/ payment data for one debtor can easily be administered.

By storing customer data in the PAYONE Platform it is moreover possible to initiate follow-up bookings for a customer without needing to submit the customer data. It is therefore not necessary for the merchant to store e.g. credit card information.

Attention:

When follow-up bookings for the same customer (debtor) (same userid or customerid) are carried out, the debtor's master data is updated / overwritten with the current values.

1.4 Invoicing

The PAYONE Platform can automatically generate invoices and credit memos for you and send these, e.g. as PDF documents, to your customer via email or post.

With the "Access" version the description provided in the offer you have generated is automatically used as the invoice item.

With the "Shop" version you have the possibility to supply the PAYONE Platform with your complete shopping cart including article number, quantity, description, price and VAT. These positions are automatically used as invoice items.

You can create the invoices according to your specifications.

Once the invoice is activated and the invoice/credit memo has been successfully carried out, an invoice/credit memo in your design is automatically created by the PAYONE Platform and sent to the customer as a PDF document via email or post. Afterwards you can download the invoices sent at any time in the PMI (PAYONE Merchant Interface).

For configuration of the PAYONE Platform invoicing module please contact the PAYONE merchant service.

1.5 Dunning processes and encashment

At your request the PAYONE Platform will carry out commercial dunning processes as well as the transfer to encashment. Within the dunning process, the customer will receive up to three reminders (e.g. via email, post) with requests for payment. If the dunning process is without success, the case can be transferred to an external encashment agency. All reminders that have been sent can be viewed via PMI (PAYONE Merchant Interface).

If an invoice is not settled by the specified date or in the case of return debit notes and chargebacks (credit card) the case is automatically transferred to the PAYONE Platform's internal dunning.

In the reminders, the customer receives an overview of all outstanding requests and of any additional fees that may have resulted. The email includes all data relevant for payment and the customer is therefore able to settle all outstanding requests by credit transfer straight away. The incoming payments are automatically assigned to the outstanding request by the PAYONE debtor management system. If the dunning procedure is without success, the case is transferred to an encashment agency.

The merchant is supplied with all outstanding requests via the TransactionStatus (see chapter 4.2). In the same manner the TransactionStatus transmits the settlement of every outstanding request. The customer is optionally provided with a confirmation mail acknowledging the settlement of the outstanding request.

For configuration of the PAYONE Platform Collect module please contact the PAYONE Merchant Service.

1.6 Administration of subscriptions

With the help of the Contract module the PAYONE Platform manages subscriptions and recurring payments. Terms, prices and dependencies can be defined freely within the PMI (PAYONE Merchant Interface), which means that complex order models can be displayed as well.

In order for subscriptions to be managed automatically via the PAYONE Platform, you must first provide the key details of the subscription in the PMI (PAYONE Merchant Interface). For this purpose, create a payment portal of the type "Access" and corresponding offers (templates) for the different subscriptions (see chapter 2.1). Here, you can define terms, prices, etc. for the subscription.

To initialise a subscription use the corresponding order ID (template) and a "createaccess" request. If the first booking is successful, a subscription will be created for the customer using the template.

All bookings created by the administration of subscriptions via the TransactionStatus (see chapter 4.2) are submitted to the merchant. If Invoicing is active, the customer will automatically receive an invoice with each booking.

For configuration of the PAYONE Platform Contract module please contact the PAYONE merchant service.

2 Payment portals

2.1 Payment portal version "Access"

2.1.1 Creating a payment portal of the version "Access"

In order to invoice your products and services with access management or administration of subscriptions via PAYONE, you must first create at least one payment portal of the version "Access".

Options

Portal name: General description of the payment portal.

URL: URL on which your products or services you wish to settle via this payment portal are to be found.

Sender email: Email-address that is used for emails to end customers

2.1.2 Adding an offer to a payment portal

Options initial term

Name: Will be used as article description for the creation of invoices.

Login time: Defines the duration for the initial contract term. (Days/months/years)
Do not exceed a duration of 60 months / 5 years.

Price: Price for the initial contract term.

Options follow-up term / repeat term

Automatic renewal: Activate this check box to activate repeat terms. If you activate this option, the subscription will be renewed until it is actively terminated.

Name: Will be used as article description for the creation of invoices.

Repeat cycle: Defines the duration of the repeat cycle. (Days/months/years).
Do not exceed a duration of 60 months / 5 years.

Price: Price of the repeat cycle

Currency: The currency applies for both the initial term and the repeat term.

VAT tax rate: The VAT tax rate applies for both the initial term and the repeat term.

2.1.3 Extended options of the payment portal

Options

SessionStatus URL:	Status changes for an open session (access) are transmitted to this URL via HTTP request. (see chapter 4.1)
	If you are using the SessionControl script, enter the corresponding URL here.
TransactionStatus URL:	Status changes for a payment process are transmitted to this URL via HTTP request. (see chapter 4.2)
Key:	This key is transmitted to the API/ SessionStatus/ TransactionStatus URL as an MD5 or SHA2-384 hash value with every call.
	Attention: The hash value has to be given in lower case.
Purpose:	Purpose used for this offer

2.2   Payment portal version "Shop"

2.2.1   Creating a payment portal of the version "Shop"

In order for the PAYONE System to be able to handle the payments for one or several products at a time, you must first create a payment portal of the version "Shop". Please notice the differences to the payment portal of the version "Access" (see 2.1).

Options

Shopname:	General description of the shop portal.
URL:	URL on which to find your products you wish to sell here.
TransactionStatus URL:	Status changes for a payment process are transmitted to this URL via HTTP request. (see chapter 4.2)
Key:	This key is transmitted to the API/ SessionStatus/ TransactionStatus URL as an MD5 or SHA2-384 hash value with every call.
	Attention: The hash value has to be given in lower case.

3   Interface definitions

3.1   General

3.1.1   Data transfer

The data transfer is processed directly via an HTTPS-POST or HTTPS-GET request by the browser of the customer.

The data is returned in JSON data format or as parameter in an HTTP redirect.

The requests must be sent to the following URL:

Client API URL: https://secure.pay1.de/client-api/

3.1.2 Standard parameter

With each request the following parameters must always be submitted.

Standard parameter			
Parameter	**Required**	**Format**	**Comment**
mid	+	N..6	Merchant account ID
portalid	+	N..7	Payment portal ID
api_version[1]	+	Default	3.8 Actual API-version (Default if not present) 3.9 New API-version from 2015-01-05 3.10 New API-version from 2016-06-01
mode	+	Default	test: Test mode live: Live mode
request	+	Default	Queries: - creditcardcheck - bankaccountcheck …
responsetype	+	Default	- JSON - REDIRECT
hash	+	AN..32 AN..96	MD5 hash code (see chapter 3.1.4) SHA2-384 hash code (see chapter 3.1.4)
successurl	o	AN..255	URL "payment successful" (only if responsetype=REDIRECT or required by corresponding request)
errorurl	o	AN..255	URL "faulty payment" (only if responsetype=REDIRECT or required by corresponding request)
encoding	-	Default	ISO 8859-1 (default) UTF-8

---

[1] New parameter "api_version" should be added to actual implementations as it will be mandatory in future.

3.1.3   Special remark

- All information described as "Unixtimestamp" refers to coordinated universal time (UTC) and is hence not subject to changing from daylight saving time to standard time.

- Only use key-value-pairs which are filled with meaningful data. All parameters that are not required for a request must not be used. Do not use dummy-values (like "-" or "x") and do not use empty values. E.g.:

  - request "updateuser" does not require a parameter "clearingtype" nor "currency".
    -> Do not send e.g. "clearingtype=" or "clearingtype=-", …

  - request "getinvoice" does not require a parameter "amount" nor "language".
    -> Do not send e.g. "amount=" or "amount=0", …

  - request "preauthorization" with "clearingtype=cc" (creditcard) does not require bankdata.
    -> Do not send e.g. "bankcountry=", "bankcountry=x", "iban=" or "iban=x", …

  - Do not use dummy values like "birthdate=00000000" or "birthdate=19700101"
    -> then do not send parameter "birthdate" at all.

- Use correct encoding: You may specify two different encodings (ISO 8859-1 or UTF-8). Please set the encoding you really use and don't mix them up. This may lead to denied requests or to misinterpreted data.

- Please use only capital letters for country-codes and state-codes.

- Please use only small letters for language-codes.

- URLs (like successurl, errorurl, backurl) should not contain special characters (e.g. "+") as they can be mis-interpretated sometimes.
  e.g.: "+" (plus) is converted to " " (space) with service SOFORT-Überweisung (SB/PNT)

- PAYONE API does not modify any given data. Please ensure that e.g. whitespace characters (leading, trailing of in between (for IBAN, BIC, PAN/PPAN)) are removed before sending data to PAYONE.

- URLs and E-Mail-addresses with non-latin-characters have to be translated to ASCII using Punycode before passing to PAYONE API as PAYONE does not modify any given data.

3.1.4 Calculation of the HASH value

The hash value protects the request parameters from changes by the customer, e.g. price changes.

The hash value is calculated from the request parameters and a secret key using the hash function hash("md5" , $data) or hash_hmac("sha2-384", $data, $key). Any parameter values to be protected are concatenated in alphabetical order. Finally the key will be attached to the string and the hash value is calculated.

Parameters are sorted by their name. It does not matter in which order they are used in the request URL. See below table for the parameters to be protected.

You can assign the key to be used in the PMI (PAYONE Merchant Interface). Do not pass this key to third parties in any case. You can also choose the desired hash algorithm in the PMI:

PMI, configuration, payment portals, hash calculation	
PMI-Selection	**Meaning**
md5 (actual method)	PAYONE platform expects hash value as md5 (md5 is compromised and should be replaced by sha2-384) for compatibility reason this is currently default.
sha2-384 (recommended method)	PAYONE platform expects hash value as sha2-384 (if you want to use sha2-384 hash values this option has to be activated)
md5_or_sha2-384 (during migration)	PAYONE platform accepts both hash calculations (md5 and sha2-384) (this option should only be use for migration from md5 to sha2-384)

The following parameter values must be included in the HASH value:

Parameter im HASH		
access_aboperiod	document_date	period_length_trail
access_aboprice	due_time	period_unit_recurring
access_canceltime	eci	period_unit_trail
access_expiretime	ecommercemode	portalid
access_period	encoding	pr[x]
access_price	errorurl	pr_recurring[x]
access_starttime	exiturl	pr_trail[x]
access_vat	getusertoken	productid
accesscode	id[x]	reference
accessname	id_recurring[x]	request

addresschecktype	id_trail[x]	responsetype
aid	invoice_deliverydate	settleaccount
amount	invoice_deliveryenddate	settleperiod
amount_recurring	invoice_deliverymode	settletime
amount_trail	invoiceappendix	storecarddata
api_version	invoiceid	successurl
backurl	it[x]	ti[x]
booking_date	mandate_identification	ti_recurring[x]
businessrelation	mid	ti_trail[x]
checktype	mode	userid
clearingtype	narrative_text	va[x]
consumerscoretype	no[x]	va_recurring[x]
currency	no_recurring[x]	va_trail[x]
customerid	no_trail[x]	vaccountname
de[x][2]	param	vreference
de_recurring[x]	period_length_recurring	
de_trail[x]		

---

[2] [x] always means that all elements of the given array have to be part of the hash.

Attention:

PAYONE Platform expects the calculated hash value converted to lower case; e.g. 87dbc7c369b85b7a699adff1a2b27bab ("md5") or d0ff16426ec9071cfe12c4440738875f…caf28c5ac9abb52b5d ("sha2-384").

An example in PHP:

```
    $request="authorization";    // Request
    $portalid=2000001;           // Portal ID
    $mid=10001;                   // Merchant Account ID
    $aid=10002;                   // Sub Account ID
    $key="secret";                // Key (configurable in the payment portal)
    $responsetype="REDIRECT";     // Response type

    $id[1]="123-345"              // Your item no.
    $pr[1]=5900;                  // Price in cent
    $no[1]=1;                     // Amount
    $de[1]="Puma Outdoor";        // Item description
    $va[1]=19;                    // Percentage of value added tax (option)
    $amount=round($pr[1]*$no[1]);// Total
    $currency="EUR";              // Currency
    $reference="73464354";        // Merchant Reference no.
    $customerid="123456";         // Merchant customer no. (option)

    // usage of md5-hash
    // select "md5" in PMI-portal-settings
    // $hash=md5($aid . $amount . $currency . $customerid . $de[1] .
    //         $id[1] . $mid . $no[1] . $portalid . $pr[1] . $reference .
    //         $request . $responsetype . $va[1] .
    //         $key);  // Parameters in sorted order + key

    // usage of sha2-384-hash
    // select "sha2-384" in PMI-portal-settings
    $hash=hash_hmac("sha384", $aid .
            $amount .
            $currency .
            $customerid .
            $de[1] .
            $id[1] .
            $mid .
            $no[1] .
            $portalid .
            $pr[1] .
            $reference .
            $request .
            $responsetype .
            $va[1], // Parameters in sorted order
            $key);  // $key is an individual parameter !
```

Attention:

The PAYONE Platform expects the hash value in small letters.

Please pay attention that for "sha2-384" the portal key is not part of parameters for hash-function, but a separate parameter passed to hash-function "hash_hmac".

3.1.5 PAYONE hosted-iFrame mode (credit card data, PCI DSS)

With the PAYONE hosted-iFrame mode you have the flexibility with design of integration and an easy PCI DSS certification with level SAQ A. The basic requirements to be eligible with SAQ A[3] are:

- Your company accepts only card-not-present (e-commerce or mail/telephone-order) transactions;
- All payment acceptance and processing are entirely outsourced to PCI DSS validated third-party service providers;
- Your company has no direct control of the manner in which cardholder data is captured, processed, transmitted, or stored;
- Your company does not electronically store, process, or transmit any cardholder data on your systems or premises, but relies entirely on a third party(s) to handle all these functions;
- Your company has confirmed that all third party(s) handling acceptance, storage, processing, and/or transmission of cardholder data are PCI DSS compliant; and
- Your company retains only paper reports or receipts with cardholder data, and these documents are not received electronically.

and

- The entirety of all payment pages delivered to the consumer's browser originates directly from a third-party PCI DSS validated service provider(s).

With PAYONE hosted-iFrame mode you will be compliant with all these requirements as the input fields for sensitive credit card data are designed as iframes and are provided by the PCI DSS certified PAYONE platform.

Include the following JavaScript in your HTML page containing the form:

JavaScript URL: https://secure.pay1.de/client-api/js/v1/payone_hosted_min.js

Remark

This JavaScript includes support of classic Client-API features described in chapter "AJAX mode" and "Redirect mode" (see following chapters) to support bank based payments and the PAYONE hosted iFrame mode.

To use the features of the classic Client-API with the new JavaScript you simply have to change "new PayoneRequest(…)" to "new Payone.ClientApi.Request(…)".

Known issues

There is a known issue with an old version (1.6 or before) of framework "prototype" – please ensure to use the actual version (at least 1.7).

---

[3] Please refer to PCI DSS Security Standards listed in SAQ A V3 on https://de.pcisecuritystandards.org

3.1.5.1 Brief description

The merchant has to implement:

- A selector for cardtype (see example: <select id="cardtype">)
  Configure your credit card acceptance here with the desired credit card types.
  The selected cardtype has to be passed to the public method "setCardType".
  You may also specify in config.cardtype available cardtypes for selection in a PAYONE iFrame.

- Place a label for each credit card input field and a div-container for the input iframes (see example: <div id="cardpanInput"></div>)

- Optional: place a div-container to display error messages returned from the creditcardcheck (see example: <div id="error"></div>)

- Define an object to configure the design of the iFrames (see example: config =)
  - Optional: Attribute "cardtype" can be configured with allowed cardtypes that are displayed in a PAYONE iFrame.
    -> Then the cardtype selector in your shop has to be removed.
  - The attributes "cardpan", "cardcvc2", "cardexpiremonth", "cardexpireyear" are fixed.
  - The value for attribute "selector" refers to the name of your div-container where the PAYONE iFrame should be rendered.
  - You may specify for each field the type of input (text, password, tel [4] and select (month/year only) using the attribute "type" and a CSS-style using the attribute "style".
  - The value for attribute "error" refers to the name of your div-container where error-messages should be displayed.
  - Pass your currently selected language to the config-object

- **NEW FEATURE:** Decide whether you would like to use the feature of automatic cardtype detection. To use this feature simply add a block "autoCardtypeDetection" to the config-section. If enabled the user simply has to enter the PAN and the cardtype is detected automatically (for most major brands). By using a callback-function you may even get the result of the automatic detection, e.g. in order to change your creditcard brand logos.

---

[4] type "tel" can be used to display a simple, numeric phone keyboard on mobile devices.

Once all fields are set you will have to start the public function "iframes.creditCardCheck" which will initiate a creditcardcheck-request. This request will check the entered creditcard details and return a response-object which contains either:

- response.status = "VALID"
- response.pseudocardpan containing the unique pseudocardnumber (Pseudo-PAN)
- response.truncatedcardpan containing the masked creditcard number (masked PAN)
- response.cardtype containing the selected cardtype
- response. cardexpiredate containing the entered expiredate (YYMM)

or

- response.status = "INVALID"
- response.errorcode -> please refer to the section "error messages"
- response.errormessage in the specified language

3.1.5.2 Table of config-attributes

Tipps / known issues:

- Some browsers do have problems if only "width" or "height" is specified. So if you want to define height or width for an input field, provide always both.

3.1.5.2.1 Attributes and values in object "config.fields"

cardtype	cardpan	cardcvc2	cardexpiremonth	cardexpireyear	attribute	value
x					cardtypes	Define possible cardtypes for selection in PAYONE iFrame, e.g. ["V", "M", "A"]. New cardtype "#" has been implemented to enforce user selection and display "Please select" initially. If used, it should be the first element of the array.
x	x	x	x	x	selector	Name of your div-container, e.g. "cardpan"
	x	x	x	x	size	Size of input field in characters, e.g. "20"
	x	x	x	x	maxlength	Maximum length of accepted input, e.g. "20"
		x			length	Array of exact length of accepted input per CC-type e.g.: length: { "A": 4, "V": 3, "M": 3, "J": 0 }
	x	x	x	x	type	Define type of input field: "text" input is visible "tel" input is visible, simple keyboard is displayed on mobile devices "password" input is masked "select" display selection/drop-down with possible values (only valid for month and year)
x	x	x	x	x	style	CSS style properties, e.g. "font-size: 1em; border: 1px solid #000; background: white; color: red; width: 145px; height: 70px; font-family: 'Courier'; font-style: italic; font-weight: bold; text-align: center; letter-spacing: 2px;" *Remark:* • *You may use any CSS style property except "url".* • *If "url" is used the style will be ignored as PCI DSS does not allow external ressources.* • *These characters must not be used: "\", "/", "?", "<" and ">"*

x	x	x	x	x	iframe	Size, e.g.: iframe: {
						   height: "25px",
						   width: "250px"

					}

3.1.5.2.2 Attributes and values in object "config.defaultStyle"

attribute	value
input	CSS style properties for input fields, e.g. "font-size: 1em; border: 1px solid #000; width: 175px;"
select	CSS style properties for select fields, e.g. "font-size: 1em; border: 1px solid #000;"

iframe	Size in pixel, e.g.: iframe: {
	   height: "25px",
	   width: "250px"
	}

3.1.5.2.3 Attributes and values in object "config.autoCardtypeDetection"

attribute	value
supportedCardtypes	Define possible cardtypes that should be accepted by automatic detection, e.g. ["V", "M", "A"]
callback	Define a function that is used as a callback-function if result of automatic cardtype detection changes. Detection starts after first 6 digits of the PAN – this is the BIN (Bank Identification Number) of a credit card and the official id for cardtype and origin detection. This callback-function can be used to change highlighting of logos for selected / detected cardtypes. Possible result codes in callback-function: <table><tr><th>Resultcode</th><th>Meaning</th></tr><tr><td>A</td><td>Detected for PAN starting "34", "37"</td></tr><tr><td>D</td><td>Detected for PAN starting "36", "38", "39"</td></tr><tr><td>M</td><td>Detected for PAN starting "51", "52", …, "55"</td></tr><tr><td>O</td><td>Detected for PAN starting "50", "56", "57", "58"</td></tr><tr><td>V</td><td>Detected for PAN starting "40", "41", …, "49"</td></tr><tr><td>?</td><td>Cardtype not automatically detected</td></tr><tr><td>-</td><td>Cardtype automatically detected, but not allowed because not in array supportedCardtypes</td></tr></table>Further cardtypes may be added in future.
deactivate	Can be used to preconfigure the config-block and still deactivate automatic cardtype-detection. Possible values: true || false

3.1.5.3 Table of request-attributes

These attributes and values are allowed in object "request":

attribute	value
request	fixed value: 'creditcardcheck'
responsetype	fixed value: 'JSON'
storecarddata	fixed value: 'yes'
mode	mode for transactions, either 'live' or 'test'
encoding	your encoding, either 'ISO-8859-1' or 'UTF-8'
mid	your Merchant ID
aid	your Account ID
portalid	your Portal ID
hash	MD5 or sha2_384 hash over request values (alphabetical order) plus portal key in your PMI portal configuration. e.g.: - aid: '10002', // your AID - encoding: 'UTF-8', // desired encoding - mid: '10001', // your MID - mode: 'live', // desired mode - portalid: '2000002', // your PortalId - request: 'creditcardcheck', // fixed value - responsetype: 'JSON', // fixed value - storecarddata: 'yes', // fixed value - PMI Portal key: '123456' MD5('10002UTF-810001live2000002creditcardcheckJSONyes123456') =5c4014cebeb361d9e186fd42c810b9b1 or hash_hmac("sha384", "10002UTF-810001live2000002creditcardcheckJSONyes", "123456") =1cf456bf692453613ebb992a3fb859cc347ddc7e94e2ca764efbe8b0089de6964ab1266df0831e59de89dc5291070fe7

3.1.5.4 Simple example: Automatic cardtype detection – with VISA/Mastercard and some debug

This is a very simple example to show how to configure PAYONE hosted-iFrame mode:

- without cardtype-selection cardtype selection
- with automatic cardtype detection enabled
- with logos enabled to show the user the detected cardtype

Simply change:

- config.fields.*.style to your CSS layout
- request.*-fields to your mid, aid, portalid, mode, encoding and hash value
- set array "supportedCardtypes" to your cardtypes you'd like to process
- callback.function to enable / disable your logos if you want to

and finally

- initiate a preauthorization / authorization via received "pseudocardpan".

```
<!DOCTYPE html>
<html>
    <head lang="en">
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <link href="static/static/client_api.css" rel="stylesheet" type="text/css" media="screen" />
    <link rel="shortcut icon" type="image/x-icon" href="static/static/favicon.ico" />
    <meta charset="UTF-8">
    <title>Client API with the new hosted iFrames example</title>
</head>
<body>
<h1>PAYONE Client API with new hosted iFrames example</h1>
<form name="paymentform" action="#" method="post">
    <fieldset>

        <input type="hidden" name="pseudocardpan" id="pseudocardpan">
        <input type="hidden" name="truncatedcardpan" id="truncatedcardpan">
        <input type="hidden" name="cardtypeResponse" id="cardtypeResponse">
        <input type="hidden" name="cardexpiredateResponse" id="cardexpiredateResponse">

        <img style="border: #FFF solid 5px; width: 100px;" src="static/visa.png" alt="" id="visa" /> <img
src="static/mastercard.png" alt="" id="mastercard"  style="border: #FFF solid 5px; width: 100px;" /><br />

        <label for="cardpanInput">Cardpan:</label>
        <span class="inputIframe" id="cardpan"></span> <!-- Container for PAYONE-Script -> PAYONE will place
an iFrame in here -->

        <label for="cvcInput">CVC:</label>
        <span class="inputIframe" id="cardcvc2"></span>

        <label for="expireInput">Expire date (mm/yyyy):</label>
        <span class="inputIframe" id="expireInput">
            <span id="cardexpiremonth"></span>
            <span id="cardexpireyear"></span>
        </span>

        <div id="error"></div>
        <br />
        <input id="submit" type="button" value="Run credit card check" />
        <input id="submitWithOutCompleteCheck" type="button" value="Run CCC (w/o complete check)" />
    </fieldset>
</form>
<h2>The JSON received from the server</h2>
<div id="jsonResponse"><pre id="jsonResponsePre">Nothing received yet.</pre></div>
<h2>Autodetection callback result</h2>
<div id="autodetectionResponse"><pre id="autodetectionResponsePre">Nothing received yet.</pre></div>

<script>
    var request,
        supportedCardtypes = ["V", "M"],
```

```
            config = {
                fields: {
                    cardpan: {
                        selector: "cardpan",
                        style: "font-size: 14px; border: 1px solid #000;",
                        type: "input"
                    },
                    cardcvc2: {
                        selector: "cardcvc2",
                        type: "password",  // Could be "text" as well.
                        style: "font-size: 14px; border: 1px solid #000;",
                        size: "4",
                        maxlength: "4",
                        length: { "V": 3, "M": 3 } // enforce 3 digit CVC für VISA and Mastercard
                    },
                    cardexpiremonth: {
                        selector: "cardexpiremonth",
                        type: "text", // select (default), text, tel
                        size: "2",
                        maxlength: "2",
                        iframe: {
                            width: "40px"
                        },
                        style: "font-size: 14px; width: 30px; border: solid 1px #000; height: 22px;"
                    },
                    cardexpireyear: {
                        selector: "cardexpireyear",
                        type: "text", // select (default), text, tel
                        iframe: {
                            width: "60px"
                        },
                        style: "font-size: 14px; width: 50px; border: solid 1px #000; height: 22px;"
                    }
                },
                defaultStyle: {
                    input: "font-size: 1em; border: 1px solid #000; width: 175px;",
                    select: "font-size: 1em; border: 1px solid #000;",
                    iframe: {
                        height: "22px",
                        width: "180px"
                    }
                },

                autoCardtypeDetection: {
                    supportedCardtypes: supportedCardtypes,
                    callback: function(detectedCardtype) {
                        // For the output container below.
                        document.getElementById('autodetectionResponsePre').innerHTML = detectedCardtype;

                        if (detectedCardtype === 'V') {
                            document.getElementById('visa').style.borderColor = '#00F';
                            document.getElementById('mastercard').style.borderColor = '#FFF';
                        } else if (detectedCardtype === 'M') {
                            document.getElementById('visa').style.borderColor = '#FFF';
                            document.getElementById('mastercard').style.borderColor = '#00F';
                        } else {
                            document.getElementById('visa').style.borderColor = '#FFF';
                            document.getElementById('mastercard').style.borderColor = '#FFF';
                        }
                    } //,
                    // deactivate: true // To turn off automatic card type detection.
                },

                language: Payone.ClientApi.Language.de, //, // Language to display error-messages (default:
Payone.ClientApi.Language.en)
                error: "error" // area to display error-messages (optional)
            };

    request = {
        request: 'creditcardcheck',             // fixed value
        responsetype: 'JSON',                    // fixed value
        mode: 'live',                            // desired mode
        mid: '10001',                            // your MID
        aid: '10002',                            // your AID
        portalid: '2000002',                     // your PortalId
        encoding: 'UTF-8',                       // desired encoding
        storecarddata: 'yes',                    // fixed value
        // hash calculated over your request-parameter-values (alphabetical request-order) plus PMI portal
key
        // hash:
'1cf456bf692453613ebb992a3fb859cc347ddc7e94e2ca764efbe8b0089de6964ab1266df0831e59de89dc5291070fe7'
        hash: '5c4014cebeb361d9e186fd42c810b9b1'      // see Chapter 3.1.5.3
    };
```

```
    var iframes = new Payone.ClientApi.HostedIFrames(config, request);

    // Function called by submitting PAY-button
    function pay() {

        if (iframes.isComplete()) {
            // Perform "CreditCardCheck" to create and get a PseudoCardPan; then call your function
"payCallback"
            iframes.creditCardCheck('payCallback');
        } else {
            alert("Not complete. Nothing done.");
        }
    }

    document.getElementById("submit").onclick = function () {
        pay();
    };

    document.getElementById("submitWithOutCompleteCheck").onclick = function () {
        iframes.creditCardCheck('payCallback');
    };

    function payCallback(response) {

        console.debug(response);
        if (response.status === "VALID") {
            document.getElementById("pseudocardpan").value = response.pseudocardpan;
            document.getElementById("truncatedcardpan").value = response.truncatedcardpan;
            document.getElementById("cardtypeResponse").value = response.cardtype;
            document.getElementById("cardexpiredateResponse").value = response.cardexpiredate;
        }

        if (typeof response === 'object') {

            var responseAsString = 'time: ' + new Date().getTime() + "\n";
            for (var key in response) {

                if (response.hasOwnProperty(key)) {
                    responseAsString += key + ': ' + response[key] + "\n";
                }
            }

            document.getElementById('jsonResponsePre').innerHTML = responseAsString;
        }
    }
</script>
</body>
</html>
```

3.1.5.5 Simple example: Automatic cardtype detection – with some debug

This example (not nice, but useful) will show how to configure PAYONE hosted-iFrame mode with cardtype selection in a PAYONE iFrame and automatic cardtype detection enabled. The field for cardtype selection can be used to override the automatic detection.

Simply change:

- set array "supportedCardtypes" to your cardtypes you'd like to process
- config.fields.*.style to your CSS layout
- request.*-fields to your mid, aid, portalid, mode, encoding and hash value

and finally

- initiate a preauthorization / authorization via received "pseudocardpan".

```
<!DOCTYPE html>
<html>
    <head lang="en">
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <link href="www.your-url.com/static/client_api.css" rel="stylesheet" type="text/css" media="screen" />
    <link rel="shortcut icon" type="image/x-icon" href="www.your-url.com/static/favicon.ico" />
    <meta charset="UTF-8">
    <title>Client API with the new hosted iFrames</title>
</head>
<body>
<h1>PAYONE Client API with new hosted iFrames</h1>
<h2>The hosted iFrames beside two regular input fields</h2>
<form name="paymentform" action="#" method="post">
    <fieldset

        <input type="hidden" name="pseudocardpan" id="pseudocardpan">
        <input type="hidden" name="truncatedcardpan" id="truncatedcardpan">
        <input type="hidden" name="cardtypeResponse" id="cardtypeResponse">
        <input type="hidden" name="cardexpiredateResponse" id="cardexpiredateResponse">

        <label for="cardtype">Card type:</label>
        <span class="inputIframe" id="cardtype"></span>

        <label for="cardpanInput">Cardpan:</label>
        <span class="inputIframe" id="cardpan"></span> <!-- Container for PAYONE-Script -> PAYONE will place
an iFrame in here -->

        <label for="cvcInput">CVC:</label>
        <span class="inputIframe" id="cardcvc2"></span>

        <label for="expireInput">Expire date (mm/yyyy):</label>
        <span class="inputIframe" id="expireInput">
            <span id="cardexpiremonth"></span>
            <span id="cardexpireyear"></span>
        </span>

        <label for="firstname">Firstname:</label>
        <input id="firstname" type="text" name="firstname" value="" placeholder="Firstname">

        <label for="lastname">Lastname:</label>
        <input id="lastname" type="text" name="lastname" value="" placeholder="Lastname">

        <div id="error"></div>
        <br />
        <input id="submit" type="button" value="Run credit card check" />
    </fieldset>
</form>
<h2>The JSON received from the server</h2>
<div id="jsonResponse"><pre id="jsonResponsePre">Nothing received yet.</pre></div>
<h2>Autodetection callback result</h2>
<div id="autodetectionResponse"><pre id="autodetectionResponsePre">Nothing received yet.</pre></div>
<h2>Set card type manually</h2>
<div class="maxWidth">
    <p class="infoText">
        It is possible to set the card types manually. It is not required to use the hosted iFrame with the
card type
```

```
          drop down select. Therefore you can use the following two buttons to change the card type either to
V or to J.
          This can be used to test detailed configuration options with the cardcvc2-iFrame.
    </p>
</div>
<div class="maxWidth">
    <p class="infoText">
        <input type="text" id="manCardType" value="" maxlength="1" style="width: 50px; display: inline;" />
        <button id="setCardTypeManually">Set card type manually to the value on the left</button>
    </p>
</div>
<div id="setToVDiv"><button id="setToV">Set Credit Card to V manually</button></div>
<div id="setToJDiv"><button id="setToJ">Set Credit Card to J manually</button></div>
<div class="maxWidth">
    <p class="infoText">
        This button uses the setCardType() with an empty string. This should not trigger a runtime error on
the JavaScript-
        console of your browser.
    </p>
</div>
<div class="maxWidth">
    <p class="infoText">
        This button sends valid JSON to change the credit card to 'M' as the setCardType() method would do.
However
        we do not send the origin parameter within the JSON so it should be ignored. Attention: The card
type selector
        would not be changed, even if the request is valid.
    </p>
</div>
<div class="maxWidth">
    <p class="infoText">
        In some exceptional cases the iFrames that we create during runtime can receive messages by
postMessage() which
        are not meant for them. This button sends some crap string that does not represent a JSON string.
This should not
        deal any damage to the JavaScript runtime within payone_hosted*.js or payone_iframe*.js.
    </p>
</div>
<script>

    var request,
        supportedCardtypes = ["#", "V", "M", "J", "U", "P"],
        config;

    /*

    // Some overwriting and placeholder enabling examples.

    // Example for enabling placeholder attribute in <input type="text" /> fields. Defaults to empty, so
placeholder will not be applied.
    Payone.ClientApi.Language.de.placeholders.cardpan = 'Kartennummer';
    Payone.ClientApi.Language.de.placeholders.cvc = 'CVC';
    Payone.ClientApi.Language.de.placeholders.expireMonth = '(MM)';
    Payone.ClientApi.Language.de.placeholders.expireYear = '(JJJJ)';

    // Example for overwriting error messages in english.
    Payone.ClientApi.Language.en.invalidCardpan: "your error text";
    Payone.ClientApi.Language.en.invalidCvc: "your error text";
    Payone.ClientApi.Language.en.invalidPanForCardtype: "your error text";
    Payone.ClientApi.Language.en.invalidCardtype: "your error text";
    Payone.ClientApi.Language.en.invalidExpireDate: "your error text";
    Payone.ClientApi.Language.en.invalidIssueNumber: "your error text";
    Payone.ClientApi.Language.en.transactionRejected: "your error text";

    // Example for overwriting default month i18n in english (instead of 1, 2, 3 etc.)
    Payone.ClientApi.Language.en.months.month1 = "January";
    Payone.ClientApi.Language.en.months.month2 = "February";
    Payone.ClientApi.Language.en.months.month3 = "March";

    */

    config = {
        fields: {
            cardpan: {
                selector: "cardpan",
                style: "font-size: 14px; border: 1px solid #000;",
                type: "input"
            },
            cardcvc2: {
                selector: "cardcvc2",
                type: "password",  // Could be "text" as well.
                style: "font-size: 14px; border: 1px solid #000;",
                size: "4",
```

```
                // This is a configuration example for something like:
                // * Amex: Exactly 4 digits.
                // * Visa: Exactly 3 digits.
                // * JCB: No CVC, input disabled.
                // * Mastercard: Not configured, up to maxlength digits or none.
                // Notice that our validation code should be stable with case insensitive configurations
made by
                // the merchant. This is why we have a lower case "a" and an upper case "V" and "J" for this
demo
                // configuration.
                maxlength: "4",
                length: { "a": 4, "V": 3, "J": false }
            },
            cardexpiremonth: {
                selector: "cardexpiremonth",
                type: "text", // select (default), text, tel (for alternative keyboards on mobile devices)
                size: "2",
                maxlength: "2",
                iframe: {
                    width: "40px"
                },
                style: "font-size: 14px; width: 30px; border: solid 1px #000; height: 22px;"
            },
            cardexpireyear: {
                selector: "cardexpireyear",
                type: "text", // select (default), text, tel (for alternative keyboards on mobile devices)
                iframe: {
                    width: "60px"
                },
                style: "font-size: 14px; width: 50px; border: solid 1px #000; height: 22px;"
            },
            cardtype: {
                selector: "cardtype",
                cardtypes: supportedCardtypes
            }
        },
        defaultStyle: {
            input: "font-size: 1em; border: 1px solid #000; width: 175px;",
            select: "font-size: 1em; border: 1px solid #000;",
            iframe: {
                height: "22px",
                width: "180px"
            }
        },

        autoCardtypeDetection: {
            supportedCardtypes: supportedCardtypes,
            callback: function(detectedCardtype) {
                // Individual callback code starts here:
                document.getElementById('autodetectionResponsePre').innerHTML = detectedCardtype;
                // Individual callback code ends.
            } //,
            // deactivate: true // To turn off automatic card type detection. Must be explicitly set to
boolean true.
        },

        language: Payone.ClientApi.Language.de, //, // Language to display error-messages (default:
Payone.ClientApi.Language.en)
        error: "error" // area to display error-messages (optional)
    };

    request = {
        request: 'creditcardcheck',            // fixed value
        responsetype: 'JSON',                  // fixed value
        mode: 'live',                          // desired mode
        mid: '10001',                          // your MID
        aid: '10002',                          // your AID
        portalid: '2000002',                   // your PortalId
        encoding: 'UTF-8',                     // desired encoding
        storecarddata: 'yes',                  // fixed value
        // hash calculated over your request-parameter-values (alphabetical request-order) plus PMI portal
key
        // hash:
'1cf456bf692453613ebb992a3fb859cc347ddc7e94e2ca764efbe8b0089de6964ab1266df0831e59de89dc5291070fe7'
        hash: '5c4014cebeb361d9e186fd42c810b9b1'      // see Chapter 3.1.5.3
    };
    var iframes = new Payone.ClientApi.HostedIFrames(config, request);

    document.getElementById('setCardTypeManually').onclick = function() {
        iframes.setCardType(document.getElementById('manCardType').value);
    };

    document.getElementById('setToV').onclick = function() {
```

```
        iframes.setCardType('V');
    };

    document.getElementById('setToJ').onclick = function() {
        iframes.setCardType('J');
    };

    // Function called by submitting PAY-button
    function pay() {

        if (iframes.isComplete()) {
            // Perform "CreditCardCheck" to create and get a PseudoCardPan; then call your function
"payCallback"
            iframes.creditCardCheck('payCallback');
        } else {
            alert("Not complete. Nothing done.");
        }
    }

    document.getElementById("submit").onclick = function () {
        pay();
    };

    function payCallback(response) {

        console.debug(response);
        if (response.status === "VALID") {
            document.getElementById("pseudocardpan").value = response.pseudocardpan;
            document.getElementById("truncatedcardpan").value = response.truncatedcardpan;
            document.getElementById("cardtypeResponse").value = response.cardtype;
            document.getElementById("cardexpiredateResponse").value = response.cardexpiredate;
        }

        if (typeof response === 'object') {

            var responseAsString = 'time: ' + new Date().getTime() + "\n";
            for (var key in response) {

                if (response.hasOwnProperty(key)) {
                    responseAsString += key + ': ' + response[key] + "\n";
                }
            }

            document.getElementById('jsonResponsePre').innerHTML = responseAsString;
        }
    }
</script>
</body>
</html>
```

3.1.5.6 Simple example: cardtype selected in PAYONE iFrame

This example will show how to configure PAYONE hosted-iFrame mode with cardtype selection in a PAYONE iFrame.

Simply change:

- config.fields.cardtypes to your cardtypes you'd like to process
- config.fields.*.style to your CSS layout
- request.*-fields to your mid, aid, portalid, mode, encoding and hash value

and finally

- initiate a preauthorization / authorization via received "pseudocardpan".

```
<html>
<head lang="en">
    <meta charset="UTF-8">
    <title>Simple example for PAYONE hosted-iFrame mode</title>

    <style type="text/css" media="screen, projection">
        * {
            margin: 0;
            padding: 0;
        }
        body {
            background: #FFF;
            color: #000;
            font: 0.9em "Helvetica";
        }
        fieldset {
            padding: 1em;
            border: 1px solid #000;
            width: 275px;
            margin: 10px;
        }
        label {
            margin-right: 10px;
            float: left;
            width: 80px;
            padding-top: 0.3em;
            text-align: right;
        }
        input, select{
            font-size: 1em;
            border: 1px solid #000;
            padding: 0.1em;
        }
        select {
            margin-right: 10px;
        }

        input, .inputIframe, select {
            display: block;
            margin-bottom: 10px;
        }

        input {
            width: 175px;
        }

        #paymentsubmit {
            float: right;
            width: auto;
            padding: 5px;
            margin-bottom: 0px;
            margin-right: 10px;
        }
        #errorOutput {

            text-align: center;
            color: #ff0000;
            display: block;
        }
```

```
        </style>
</head>

<body>
<script type="text/javascript" src="https://secure.pay1.de/client-api/js/v1/payone_hosted_min.js"></script>
<form name="paymentform" action="" method="post">
    <fieldset>
        <input type="hidden" name="pseudocardpan" id="pseudocardpan">
        <input type="hidden" name="truncatedcardpan" id="truncatedcardpan">

        <!-- place your input fields -->
        <label for="cardtypeInput">Card type:</label>
        <span id="cardtype" class="inputIframe"></span>

        <label for="cardpanInput">Cardpan:</label>
        <span id="cardpan" class="inputIframe"></span>

        <label for="cvcInput">CVC:</label>
        <span id="cardcvc2" class="inputIframe"></span>

        <label for="expireInput">Expire Date:</label>
        <span id="expireInput" class="inputIframe">
            <span id="cardexpiremonth"></span>
            <span id="cardexpireyear"></span>
        </span>

        <label for="firstnameInput">Firstname:</label>
        <input id="firstname" type="text" name="firstname" value="">

        <label for="lastnameInput">Lastname:</label>
        <input id="lastname" type="text" name="lastname" value="">

        <div id="errorOutput"></div>

        <input id="paymentsubmit" type="button" value="Submit" onclick="check();">
    </fieldset>
</form>

<div id="paymentform"></div>
<script>
    var request, config;

    config = {
        fields: {
            cardtype: {
                selector: "cardtype",               // put name of your div-container here
                cardtypes: ["V", "M", "A"]          // define possible cardtypes in PAYONE iFrame
            },
            cardpan: {
                selector: "cardpan",                // put name of your div-container here
                type: "text",                       // text (default), password, tel
                style: "font-size: 1em; border: 1px solid #000;"
            },
            cardcvc2: {
                selector: "cardcvc2",               // put name of your div-container here
                type: "password",                   // select(default), text, password, tel
                style: "font-size: 1em; border: 1px solid #000;",
                size: "4",
                maxlength: "4",                     // set max. length for CVC input; empty values possible
                length: { "A": 4, "V": 3, "M": 3, "J": 0 } // set required CVC length per cardtype
                                                    // if set exact length required; 0=CVC input disabled
            },
            cardexpiremonth: {
                selector: "cardexpiremonth",        // put name of your div-container here
                type: "select",                     // select(default), text, password, tel
                size: "2",
                maxlength: "2",
                iframe: {
                    width: "50px"
                }
            },
            cardexpireyear: {
                selector: "cardexpireyear",         // put name of your div-container here
                type: "select",                     // select(default), text, password, tel
                iframe: {
                    width: "80px"
                }
            }
        },
        defaultStyle: {
            input: "font-size: 1em; border: 1px solid #000; width: 175px;",
            select: "font-size: 1em; border: 1px solid #000;",
            iframe: {
```

```
                height: "33px",
                width: "180px"
            }
        },
        error: "errorOutput",                        // area to display error-messages (optional)
        language: Payone.ClientApi.Language.de      // Language to display error-messages
                                                     // (default: Payone.ClientApi.Language.en)
    };

    request = {
        request: 'creditcardcheck',                 // fixed value
        responsetype: 'JSON',                       // fixed value
        mode: 'live',                               // desired mode
        mid: '10001',                               // your MID
        aid: '10002',                               // your AID
        portalid: '2000002',                        // your PortalId
        encoding: 'UTF-8',                          // desired encoding
        storecarddata: 'yes',                       // fixed value
        // hash calculated over your request-parameter-values (alphabetical request-order) plus PMI portal
key
        // hash:
'1cf456bf692453613ebb992a3fb859cc347ddc7e94e2ca764efbe8b0089de6964ab1266df0831e59de89dc5291070fe7'
        hash: '5c4014cebeb361d9e186fd42c810b9b1'      // see Chapter 3.1.5.3
    };
    var iframes = new Payone.ClientApi.HostedIFrames(config, request);

    function check() {                              // Function called by submitting PAY-button
        if (iframes.isComplete()) {
            iframes.creditCardCheck('checkCallback');// Perform "CreditCardCheck" to create and get a
                                                     // PseudoCardPan; then call your function
"checkCallback"
        } else {
            console.debug("not complete");
        }
    }

    function checkCallback(response) {
        console.debug(response);
        if (response.status === "VALID") {
            document.getElementById("pseudocardpan").value = response.pseudocardpan;
            document.getElementById("truncatedcardpan").value = response.truncatedcardpan;
            document.paymentform.submit();
        }
    }

</script>
</body>
</html>
```

3.1.5.7 Simple example: cardtype set by shop-system

This example will show how to configure PAYONE hosted-iFrame mode with cardtype selection in your shop / website

Simply change:

- config.fields.*.style to your CSS layout
- request.*-fields to your mid, aid, portalid, mode, encoding and hash value

and set

- your selected credit card type via public method "setCardType"

and finally

- initiate a preauthorization / authorization via received "pseudocardpan".

```
<html>
<head lang="en">
    <meta charset="UTF-8">
    <title>Simple example for PAYONE hosted-iFrame mode</title>
    <style type="text/css" media="screen, projection">
        * {
            margin: 0;
            padding: 0;
        }
        body {
            background: #FFF;
            color: #000;
            font: 0.9em "Helvetica";
        }
        fieldset {
            padding: 1em;
            border: 1px solid #000;
            width: 275px;
            margin: 10px;
        }
        label {
            margin-right: 10px;
            float: left;
            width: 80px;
            padding-top: 0.3em;
            text-align: right;
        }
        input, select{
            font-size: 1em;
            border: 1px solid #000;
            padding: 0.1em;
        }
        select {
            margin-right: 10px;
        }

        input, .inputIframe, select {
            display: block;
            margin-bottom: 10px;
        }

        input {
            width: 175px;
        }

        #paymentsubmit {
            float: right;
            width: auto;
            padding: 5px;
            margin-bottom: 0px;
            margin-right: 10px;
        }
        #errorOutput {
            text-align: center;
```

```
                color: #ff0000;
                display: block;
            }
        </style>
</head>
<body>
<script type="text/javascript" src="https://secure.pay1.de/client-api/js/v1/payone_hosted_min.js"></script>
<form name="paymentform" action="" method="post">
    <fieldset>
        <input type="hidden" name="pseudocardpan" id="pseudocardpan">
        <input type="hidden" name="truncatedcardpan" id="truncatedcardpan">

        <!-- configure your cardtype-selection here -->
        <label for="cardtypeInput">Card type</label>
        <select id="cardtype">
            <option value="V">VISA</option>
            <option value="M">Mastercard</option>
            <option value="A">Amex</option>
        </select>

        <label for="cardpanInput">Cardpan:</label>
        <span class="inputIframe" id="cardpan"></span>

        <label for="cvcInput">CVC:</label>
        <span id="cardcvc2" class="inputIframe"></span>

        <label for="expireInput">Expire Date:</label>
        <span id="expireInput" class="inputIframe">
            <span id="cardexpiremonth"></span>
            <span id="cardexpireyear"></span>
        </span>

        <label for="firstname">Firstname:</label>
        <input id="firstname" type="text" name="firstname" value="">

        <label for="lastname">Lastname:</label>
        <input id="lastname" type="text" name="lastname" value="">

        <div id="errorOutput"></div>

        <input id="paymentsubmit" type="button" value="Submit" onclick="check();">
    </fieldset>
</form>

<div id="paymentform"></div>
<script>
    var request, config;

    config = {
        fields: {
            cardpan: {
                selector: "cardpan",                // put name of your div-container here
                type: "text",                       // text (default), password, tel
                style: "font-size: 1em; border: 1px solid #000;"
            },
            cardcvc2: {
                selector: "cardcvc2",               // put name of your div-container here
                type: "password",                   // select(default), text, password, tel
                style: "font-size: 1em; border: 1px solid #000;",
                size: "4",
                maxlength: "4",                     // set max. length for CVC input; empty values possible
                length: { "A": 4, "V": 3, "M": 3, "J": 0 } // set required CVC length per cardtype
                                                    // if set exact length required; 0=CVC input disabled
            },
            cardexpiremonth: {
                selector: "cardexpiremonth",        // put name of your div-container here
                type: "select",                     // select(default), text, password, tel
                size: "2",
                maxlength: "2",
                iframe: {
                    width: "50px"
                }
            },
            cardexpireyear: {
                selector: "cardexpireyear",         // put name of your div-container here
                type: "select",                     // select(default), text, password, tel
                iframe: {
                    width: "80px"
                }
            }
        },
        defaultStyle: {
            input: "font-size: 1em; border: 1px solid #000; width: 175px;",
```

```
            select: "font-size: 1em; border: 1px solid #000;",
            iframe: {
                height: "33px",
                width: "180px"
            }
        },
        error: "errorOutput",                      // area to display error-messages (optional)
        language: Payone.ClientApi.Language.de     // Language to display error-messages
                                                   // (default: Payone.ClientApi.Language.en)
    };

    request = {
        request: 'creditcardcheck',                // fixed value
        responsetype: 'JSON',                      // fixed value
        mode: 'live',                              // desired mode
        mid: '10001',                              // your MID
        aid: '10002',                              // your AID
        portalid: '2000002',                       // your PortalId
        encoding: 'UTF-8',                         // desired encoding
        storecarddata: 'yes',                      // fixed value
        // hash calculated over your request-parameter-values (alphabetical request-order) plus PMI portal
key
        // hash:
'1cf456bf692453613ebb992a3fb859cc347ddc7e94e2ca764efbe8b0089de6964ab1266df0831e59de89dc5291070fe7'
        hash: '5c4014cebeb361d9e186fd42c810b9b1'     // see Chapter 3.1.5.3
    };
    var iframes = new Payone.ClientApi.HostedIFrames(config, request);

    document.getElementById('cardtype').onchange = function () {
        iframes.setCardType(this.value);           // on change: set new type of credit card to process
    };

    function check() {                             // Function called by submitting PAY-button
        if (iframes.isComplete()) {
            iframes.creditCardCheck('checkCallback');// Perform "CreditCardCheck" to create and get a
                                                   // PseudoCardPan; then call your function
"checkCallback"
        } else {
            console.debug("not complete");
        }
    }

    function checkCallback(response) {
        console.debug(response);
        if (response.status === "VALID") {
            document.getElementById("pseudocardpan").value = response.pseudocardpan;
            document.getElementById("truncatedcardpan").value = response.truncatedcardpan;
            document.paymentform.submit();
        }
    }

</script>
</body>
</html>
```

3.1.5.8 Additional tips

3.1.5.8.1 Changing display text for month drop down

To change the display text in month selection drop down just add this code block just before initiating the iFrames via "new Payone.ClientApi.HostedIFrames (config, request)" in your shop:

```
Payone.ClientApi.Language.de.months.month1 = "Januar";
Payone.ClientApi.Language.de.months.month2 = "Februar";
Payone.ClientApi.Language.de.months.month3 = "März";
…
Payone.ClientApi.Language.en.months.month1 = "January";
Payone.ClientApi.Language.en.months.month2 = "February";
Payone.ClientApi.Language.en.months.month3 = "March";
…
```

Please pay attention to replace "en" by your desired language. Further languages can be added.

3.1.5.8.2 Changing display text for error messages

To change the display text for error messages add this code block just before initiating the iFrames via "new Payone.ClientApi.HostedIFrames (config, request)" in your shop:

```
Payone.ClientApi.Language.en.invalidCardpan = "your error text";
Payone.ClientApi.Language.en.invalidCvc = "your error text";
Payone.ClientApi.Language.en.invalidPanForCardtype = "your error text";
Payone.ClientApi.Language.en.invalidCardtype = "your error text";
Payone.ClientApi.Language.en.invalidExpireDate = "your error text";
Payone.ClientApi.Language.en.invalidIssueNumber = "your error text";
Payone.ClientApi.Language.en.transactionRejected = "your error text";
…
other languages if desired
```

Please pay attention to replace "en" by your desired language. Further languages can be added.

3.1.5.8.3 Specifying placeholder text for input fields

A placeholder text can be specified per input field (type "text", "password" and "tel"). Therefore add this block just before initiating the iFrames via "new Payone.ClientApi.HostedIFrames (config, request)" in your shop:

```
Payone.ClientApi.Language.de.placeholders.cardpan = 'Kartennummer';
Payone.ClientApi.Language.de.placeholders.cvc = 'CVC';
…
Payone.ClientApi.Language.en.placeholders.cardpan = 'Cardnumber';
Payone.ClientApi.Language.en.placeholders.cvc = 'CVC';
…
```

Please pay attention to replace "en" by your desired language. Further languages can be added.

3.1.6   AJAX mode

In AJAX mode, values entered in a form can be checked and processed directly without reloading the web page. In order to enable this, two JavaScripts are required: one JavaScript inside the form and one JavaScript hosted by PAYONE. The second JavaScript is required because JavaScript does not allow cross-domain access. Checking and processing the data via AJAX should take place before the form is sent to the shop.

Include the following JavaScript in your HTML page containing the form:

AJAX JavaScript URL: https://secure.pay1.de/client-api/js/ajax.js

Remark

Payment data should not come into touch with your system. This is very important with credit card data. Certification according to the PCI standard is not necessary only if the card data does not come into touch with your systems. Sensible payment data should be removed from the form after checking and processing via the client API (see below) in order to not have contact with your systems. Any other data can be queried in preceding steps.

Important note

This mode is also known as "direct post". The input fields are placed on the merchant payment page and not provided by PAYONE. Therefore the merchant needs to comply with PCI DSS SAQ A-EP certification **if the full creditcard number (PAN) is processed** (using the pseudocardnumber with "direct post" (e.g. to ask the customer for CVC) is fine).

To be SAQ A compliant PAYONE recommends implementation of the PAYONE hosted-iFrame-solution when processing the full original creditcard number (PAN).

JavaScript/AJAX code example

```javascript
var data = {
        request : 'creditcardcheck',
        responsetype : 'REDIRECT',            // JSON or REDIRECT available
        mode : 'live',
        mid : '10000',
        aid : '10001',
        portalid : '2000000',
        encoding : 'UTF-8',
        storecarddata : 'yes',
        hash : '19062005567ca72601cc9d031f9a94b1',
        cardholder : document.Testform.cardholder.value,
        cardpan : document.Testform.cardpan.value,
        cardtype : document.Testform.cardtype.value,
        cardexpiremonth : document.Testform.cardexpiremonth.value,
        cardexpireyear : document.Testform.cardexpireyear.value,
        cardcvc2 : document.Testform.cardcvc2.value,
        language : 'en'
}

var options = {
        return_type : 'object',
        callbackFunctionName: 'processPayoneResponse'
}

function processPayoneResponse(response) {
        if (response.get('status') == 'VALID') {
                document.Testform.cardpan.value='';
                document.Testform.cardcvc2.value='';

        document.Testform.pseudocardpan.value=response.get('pseudocardpan')
                document.Testform.submit();
        }
        else {
                alert(response.get('customermessage'));
        }
}

var request = new PayoneRequest(data, options);
request.checkAndStore();
```

JSON response example

```json
{
   "status" : "ERROR",
   "errorcode" : "1078",
   "errormessage" : "Parameter {cardpan} incorrect or missing",
   "customermessage" : "Invalid cardpan. Please check the information on the card.",
}
```

3.1.7 Redirect mode

Form inputs are sent directly to the client API when working in redirect mode. The client API processes the request and forwards the user to a specific page of the online shop, depending on the result. The client API itself does not produce any output.

Redirect URLs

Success URL   The customer is forwarded to this URL if the query was successful. (e.g. if the payment was successful or if the address check was positive)

Error URL   The customer is forwarded to this URL if the query was not successful. (e.g. if the payment was denied or if the address check was negative)

Response data are attached to the URL as GET parameters and can be processed by you.

The end customer is directly forwarded to the "redirecturl" if provided in the PAYONE Platform response. After that, the end customer is send back to your success URL or error URL.

Remark

Payment data should not come into touch with your system. This is very important with credit card data. Certification according to the PCI standard is not necessary only if the card data does not come into touch with your systems. Sensible payment data should be sent directly from the form to the client API and should not be forwarded trough your systems (see below). Any other data can be queried in preceding steps.

Important note

This mode is also known as "direct post". The input fields are placed on the merchant payment page and not provided by PAYONE. Therefore the merchant needs to comply with PCI DSS SAQ A-EP certification. To be SAQ A compliant PAYONE recommends implementation of the PAYONE hosted-iFrame-solution.

HTML example

```
<form action="https://secure.pay1.de/client-api/" method="POST">
<input type="hidden" name="portalid" value="2000000">
<input type="hidden" name="aid" value="10000">
<input type="hidden" name="mode" value="test">
<input type="hidden" name="request" value="bankaccountcheck">
<input type="hidden"
  name="successurl" value="http://www.yoursite.com/success.php">
<input type="hidden"
  name="errorurl" value="http://www.yoursite.com/error.php">
<input type="hidden" name="hash" value="70eaec2a33fa1b4674c0b1ge5e982966">
<input type="hidden" name="responsetype" value="REDIRECT">
<input type="hidden" name="language" value="en">
<input type="hidden" name="bankcountry" value="DE">
<table>
<tr>
  <td>Account number</td>
  <td><input type="text" name="bankaccount"></td>
</tr>
<tr>
  <td>Sort code</td>
  <td><input type="text" name="bankcode"></td>
</tr>
</table>
<input type="submit" value="Buy now!">
</form>
```

Redirect example

```
http://www.yoursite.com/error.php?status=ERROR&errocode=1083&errormessage=
Parameter+%7Bbankaccount%7D+incorrect+or+missing&customermessage=
Invalid+account+number.+Please+check+the+information+on+the+card.
```

3.2   Payment

3.2.1   Initiating payment reservation (preauthorization)

With the "preauthorization" request the transmitted amount is reserved. According to the type of payment, specific functions are carried out.

This option is <u>only</u> available in the "Shop" option.

Credit card	Credit card data is verified and stored.
	The amount is reserved on the customer's card
Debit payment	Account data is verified and stored.
Online Bank Transfer	Account data is verified
	Online transfer session is initialised
	Amount will be charged to the account directly after payment is completed
e-wallet	E-wallet session is initialised
	Amount is reserved after the payment has been completed
Prepayment	Data is stored
	Payment process is initialised
Invoice	Data is stored
	Payment process is initialised
Cash on delivery	Data is stored
	Payment process is initialised
Financing	Payment type BillSAFE requires item data, billing and delivery address need to be identical.

PAYONE Business

No receivable is entered upon this request.

Request "preauthorization"			
Parameter	**Required**	**Format**	**Comment**
aid	+	N..6	Sub account ID
clearingtype	+	Default	elv — Debit payment cc — Credit card rec — Invoice vor — Prepayment cod — Cash on delivery sb — Online Bank Transfer wlt — e-wallet fnc — Financing
reference	+	AN..20	Merchant reference number for the payment process. (Permitted symbols: 0-9, a-z, A-Z, .,-,_,/) special limits / restrictions: GPY min./max: AN 4..16
amount	+	N..7	Total amount (in smallest currency unit! e.g. cent)
currency	+	Default	Currency (ISO 4217)
param	-	AN1..255	Individual parameter
narrative_text	-	AN1..81	Dynamic text element on account statements (3 lines with 27 characters each) and credit card statements.
customer_is_present	-	Default	Indicates whether customer is present or not. Value yes — Customer is "present" and can enter his/her data in the shop. no — Customer is not present and can not enter any data.
Parameter (PAYONE Secure Invoice)			
clearingsubtype	+	Default	Clearingsubtype POV — PAYONE secure purchase on invoice
Parameter (personal data)			
customerid	-	AN1..20	Merchant's customer ID (Permitted symbols: 0-9, a-z, A-Z, .,-,_,/)
userid	-	N6..12	Debtor ID (PAYONE) userid must already exist and mode (test/live) must must match. If userid is used then personal data are loaded automatically.
businessrelation	o	Default	Will be available from 2017-12-01 or later. Will be required for payment type POV from 2018-01-01. Values b2c — Indicates business to consumer

Request "preauthorization"			
Parameter	**Required**	**Format**	**Comment**
			b2b indicates business to business
salutation	-	AN1..10	Title (e.g. "Mr.", "Mrs.", "company")
title	-	AN1..20	Title (e.g. "Dr", "Prof.")
firstname	o	AN1..50	First name (optional if company is used) Mandatory for payment type KLV, KLS, PDT
lastname	+	AN2..50	Surname
company	-	AN2..50	Company
street	-	AN1..50	Street number and name (required: at least one character) Mandatory for payment type KLV, KLS mandatory if direct debit (ELV) used with IBAN-countries CH, SM, MC, PM, JE, GG
addressaddition	-	AN1..50	Address line 2 (e.g. "7th floor", "c/o Maier") Mandatory for payment type KLV, KLS NL
zip	-	AN2..10	Postcode Mandatory for payment type KLV, KLS mandatory if direct debit (ELV) used with IBAN-countries CH, SM, MC, PM, JE, GG
city	-	AN2..50	City Mandatory for payment type KLV, KLS mandatory if direct debit (ELV) used with IBAN-countries CH, SM, MC, PM, JE, GG
country	+	Default	Country (ISO 3166)
state	-	Default	State (ISO 3166-2 subdivisions) (only if country=US, CA, CN, JP, MX, BR, AR, ID, TH, IN)
email	-	AN..50	Email address Mandatory for payment type BSV, KLV, KLS, P24
telephonenumber	-	AN..30	Telephone number
birthday	-	N8	Date of birth (YYYYMMDD) Mandatory for payment type KLV, , KLS
language	-	Default	Language indicator (ISO 639) Mandatory for payment type KLV, KLS KLV, KLS support sv, nb (norwegian), fi, da, de, nl
vatid	-	AN..50	VAT identification number
gender	-	Default	f=female, m=male Mandatory for payment type KLV, KLS DE, NL, AT
personalid	-	AN..32	Person specific numbers or characters Mandatory for payment type KLV, KLS SE, FI, DK, NO Format/allowed characters: A-Z, a-z, 0-9, +-./()
ip	-	AN..39	Customer's IP-V4-address (123.123.123.123) or IP-V6-address

Request "preauthorization"			
Parameter	**Required**	**Format**	**Comment**
			Mandatory for payment type KLV, KLS
Parameter (delivery data)			
shipping_firstname	-	AN..50	First name
shipping_lastname	-	AN..50	Surname
shipping_company	-	AN..50	Company
shipping_street	-	AN1..50	Street number and name (required: at least one character)
shipping_zip	-	AN2..10	Postcode
shipping_city	-	AN2..50	City
shipping_state	-	Default	State (ISO 3166-2 subdivisions) (only if country=US, CA, CN, JP, MX, BR, AR, ID, TH, IN) Mandatory for PPE if shipping_country listed above
shipping_country	-	Default	Country (ISO 3166)
Parameter (debit payment)			
iban	o	AN..35	International Bank Account Number Only capital letters, no spaces If both (BBAN and IBAN) are submitted, IBAN is splitted into BBAN and processed.
bic	o	AN..11	Bank Identifier Code Only capital letters, no spaces
bankcountry	o	Default	Account type/ country For use with BBAN: DE Mandatory with bankcode, bankaccount Optional with iban/bic as bankcountry will be determined by iban
bankaccount	o	N..10	Account number (BBAN) DE only: IBAN/BIC can be calculated by PAYONE platform automatically given by bankcode/bankaccount
bankcode	o	N8	Sort code (BBAN) (only in DE) DE only: IBAN/BIC can be calculated by PAYONE platform automatically given by bankcode/bankaccount (BBAN)
bankaccountholder	o	AN..35	Account holder
mandate_identification	o	AN..35	Can be used to enforce a merchant specific mandate identification. The mandate_identification must be unique. Allowed characters: A-Z a-z 0-9 + - . ( ) If the mandate_identification is not set PAYONE will create a unique mandate identification (pattern: PO-nnnnnnnnnn). PPS: This parameter must not be used!

Request "preauthorization"			
Parameter	**Required**	**Format**	**Comment**
Parameter (online transfer)			
onlinebanktransfertype	+	Default	PNT Sofortbanking (DE, AT, CH, NL) GPY giropay (DE) EPS eps – online transfer (AT) PFF PostFinance E-Finance (CH) PFC PostFinance Card (CH) IDL iDEAL (NL) P24 Przelewy24 (PL)
bankcountry	+	Default	Account type/ country (DE, AT, CH, NL, PL)
bankaccount[5]	o	N..10	Account number (giropay & Sofortbanking only) DE only: IBAN/BIC can be calculated by PAYONE
bankcode[5]	o	N8	Sort code (giropay & Sofortbanking only) DE only: IBAN/BIC can be calculated by PAYONE
bankgrouptype	o	Default	Bank Group (see chapter 5) (eps & iDEAL only)
iban[5]	o	AN..35	International Bank Account Number Only capital letters, no spaces If both (BBAN and IBAN) are submitted, IBAN is splitted into BBAN and processed.
bic[5]	o	AN..11	Bank Identifier Code Only capital letters, no spaces
successurl	o	AN..255	URL "payment successful" (only if responsetype=REDIRECT or required by corresponding request)
errorurl	o	AN..255	URL "faulty payment" (only if responsetype=REDIRECT or required by corresponding request)
backurl	o	AN..255	URL "Back" or "Cancel" (only if responsetype=REDIRECT or required by corresponding request)

---

[5] Can be omitted for OBT/PNT (Sofortbanking) from 2016-10-01 on as bank data are received from SOFORT and stored for later credits (processed as SEPA-credit).

Request "preauthorization"			
Parameter	Required	Format	Comment
Parameter (e-wallet)			
wallettype	+	Default	Wallet provider PPE: PayPal Express
successurl	o	AN..255	URL "payment successful" (only if responsetype=REDIRECT or required by corresponding request)
errorurl	o	AN..255	URL "payment faulty" (only if responsetype=REDIRECT or required by corresponding request)
backurl	o	AN..255	URL "Back" or "Cancel" (only if responsetype=REDIRECT or required by corresponding request)
it[n]	o	Default	For PPE: Item type goods Goods shipment Shipping charges handling Handling fee
id[n]	o	AN..32	Product number, order number, etc. Permitted symbols: 0-9 a-z A-Z ()[]{} +-_#/:
pr[n]	o	N..8	Unit price (in smallest currency unit! e.g. cent)
no[n]	o	N..3	Quantity
de[n]	o	AN..255	Description (on invoice) For PPE maximum 127 characters are allowed.
va[n]	o	N..4	VAT rate (% or bp) value < 100 = percent value > 99 = basis points (e.g. 1900 = 19%)
Parameter (cash on delivery)			
shippingprovider	+	Default	Shipping company DHL: DHL, Germany BRT: Bartolini, Italy
Parameter (credit card)			
cardpan	+	N..19	Card number
cardtype	+	Default	Card type V Visa M MasterCard A American Express D Diners J JCB O Maestro International C Discover

Request "preauthorization"			
Parameter	**Required**	**Format**	**Comment**
			B Carte Bleue
			P China Union Pay
cardexpiredate	+	N4	Expiry date YYMM
cardcvc2	o	N..4	Credit verification number (CVC)
cardissuenumber	-	N..2	Card issue number (only Maestro UK cards)
cardholder	-	AN..50	Card holder
ecommercemode	-	Default	Credit card transaction type:
			internet eCommerce Transaction (SSL secured)
			3dsecure 3-D Secure transaction (can be enabled alternatively in the risk settings)
			moto Mail or telephone order transaction
successurl	o	AN..255	URL "payment successful" (only if 3-D secure has been activated)
errorurl	o	AN..255	URL "faulty payment" (only if 3-D secure has been activated)
Parameter (credit card with pseudo card number)			
pseudocardpan	+	N..19	Pseudo card number (This card number can be submitted alternatively to the rest of the card data)
Parameter (Financing)			
financingtype	+	Default	Financing type BSV BillSAFE Invoice KLV Klarna Invoice KLS Klarna Installment
it[n]	+	Default	For BSV / KLV / KLS: Item type goods Goods shipment Shipping charges handling Handling fee voucher Voucher / discount Note: Item type "handling" only available after assignment by BillSAFE.
id[n]	+	AN..32	Product number, order number, etc. Permitted symbols: 0-9 a-z A-Z ()[]{} +-_#/:
pr[n]	+	N..8	Unit price (in smallest currency unit! e.g. cent)
no[n]	+	N..3	Quantity
de[n]	+	AN..255	Description (on invoice) Note:

Request "preauthorization"			
Parameter	**Required**	**Format**	**Comment**
			For BSV maximum 100 characters are allowed.
va[n]	-	N..4	VAT rate (% or bp)
			value < 100 = percent
			value > 99 = basis points (e.g. 1900 = 19%)
			Mandatory for payment type KLV, KLS
Key			
N..x	Numeric value (x characters maximum)		
AN..x	Alphanumeric value (x characters maximum)		

Response "preauthorization"			
Parameter	**Required**	**Format**	**Comment**
status	+	Default	APPROVED / REDIRECT / ERROR / PENDING
Parameter (APPROVED)			
txid	+	N..12	Payment process ID (PAYONE)
userid	+	N..12	Debtor ID (PAYONE)
Parameter (PENDING)[6] (currently only supported for KLV/KLS)			
txid	+	N..12	Payment process ID (PAYONE)
userid	+	N..12	Debtor ID (PAYONE)
Parameter (REDIRECT) (3-D Secure/online transfer/e-wallet)			
txid	+	N..12	Payment process ID (PAYONE)
userid	+	N..12	Debtor ID (PAYONE)
redirecturl	+	AN..255	Redirect URL
Parameter (ERROR)			
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message for the merchant
customermessage	-	AN..1024	Error message for the end customer
			(Language selection is based on the end customer's language, „language")

---

[6] New response value – only available with "api_version=3.9". Indicates that a payment process is pending at external payment processor. If "api_version=3.8" or not present then "APPROVED" is returned for compatibility.

			With "api_version>=3.10" you may get more detailed error messages from the external payment processor (e.g. Ratepay)
Parameter (credit card – if AVS has been ordered[7])			
protect_result_avs	-	A1	AVS return value, see chapter 5.3
Parameter (Prepayment/cash on delivery)			
clearing_bankaccountholder	+	AN..35	Recipient bank account holder
clearing_bankcountry	+	Default	Recipient account type/ country (e.g. DE, AT, etc.)
clearing_bankaccount	+	AN..26	Recipient account number
clearing_bankcode	o	AN..11	Recipient sort code
clearing_bankiban	-	AN..35	Recipient IBAN
clearing_bankbic	-	AN..11	Recipient BIC
clearing_bankcity	-	AN..50	Recipient city or bank
clearing_bankname	-	AN..50	Recipient bank name
Parameter (direct debit[8])			
mandate_identification	-	AN..35	used mandate_identification for debit payment
mandate_dateofsignature	-	N..10	date when mandate has been created (format YYYYMMDD)
creditor_identifier	-	AN..35	used CID for debit payment
creditor_name	-	AN..35	as recorded in PAYONE Platform
creditor_street	-	AN..35	as recorded in PAYONE Platform
creditor_zip	-	AN..10	as recorded in PAYONE Platform
creditor_city	-	AN..50	as recorded in PAYONE Platform
creditor_country	-	Default	Country (ISO 3166)
creditor_email	-	AN..50	as recorded in PAYONE Platform

---

[7] AVS (Address Verification System) is currently only supported for American Express

[8] **Parameter "mandate_identification" will not be provided currently to prevent declined requests by merchant systems that do not expect these key/values by now. Parameter has to be enabled in PMI, Configuration, Payment portals, General: "Additional response-data" for mode test and live.**

3.2.2 Initiating payment process (authorization)

With the "authorization" request the payment process is initiated. According to the type of payment, specific functions are carried out.

This request is <u>only</u> available in the "Shop" version.

Credit card	Card data is verified
	Card is charged immediately
Debit payment	Account data is verified
	Debit payment is carried out
Online transfer	Account data is verified where applicable
	Online transfer session is initialised
	Amount will be charged to the account directly after payment is completed
e-wallet	E-wallet session is initialised
	Amount will be charged to the account directly after payment is completed
Prepayment	<u>Not</u> supported by this request!
Invoice	Payment process is initialised

PAYONE Business

The request is booked to the transaction account immediately after the successful payment. The time of payment therefore depends on the type of payment. The only exception is the invoice - in this case the request is booked immediately independent of the payment.

The TransactionStatus informs you about the status of the requests. The TransactionStatus communicates, among other things, incoming payments when using the payment type invoice/prepayment and return debit notes or chargebacks if applicable (see chapter 4.2).

Module invoicing

For invoicing via PAYONE Platform you must submit all items that shall appear on the invoice. To do so, submit the parameters id[n], pr[n], no[n], de[n], va[n]. Start with item n=1. For all following articles use n=2, n=3, and so on. Accordingly, id[2],pr[2],no[2],de[2],va[2] are the parameters for item 2, and so on.

Attention: If a parameter is missing, the whole item will not be considered!

Request "authorization"			
Parameter	Required	Format	Comment
aid	+	N..6	Sub account ID
clearingtype	+	Default	elv — Debit payment cc — Credit card rec — Invoice cod — Cash on delivery sb — Online Bank Transfer wlt — e-wallet fnc — Financing
reference	+	AN..20	Merchant reference number for the payment process (Permitted symbols: 0-9, a-z, A-Z, .,-,_,/) special limits / restrictions: GPY min./max: AN 4..16
amount	+	N..7	Total amount in cent (in smallest currency unit! e.g. cent)
currency	+	Default	Currency (ISO 4217)
param	-	AN1..255	Individual parameter
narrative_text	-	AN1..81	Dynamic text element on account statements (3 lines with 27 characters each) and credit card statements.
customer_is_present	-	Default	Indicates whether customer is present or not. Value yes — Customer is "present" and can enter his/her data in the shop. no — Customer is not present and can not enter any data.
Parameter (PAYONE Secure Invoice)			
clearingsubtype	+	Default	Clearingsubtype POV — PAYONE secure purchase on invoice
Parameter (PAYONE Business)			
document_date	-	N8	Document date (YYYYMMDD)
booking_date	-	N8	booking date (YYYYMMDD)
due_time	-	N..11	Due date (Unixtimestamp) meaning by clearingtype: rec — due time of the invoice; needed to start dunning run elv — debit will be initiated on this date. Please pay attention that 5 or 2 days are added to the due_time – depending on FIRST or RECURRENT debit.

Request "authorization"			
Parameter	**Required**	**Format**	**Comment**
Parameter (BillSAFE, module Invoicing)			
invoiceid	-	AN..20	Merchant's invoice number
invoice_deliverymode	-	Default	M Post P PDF (via email)
invoice_deliverydate	-	N8	Delivery date (YYYYMMDD)
invoice_deliveryenddate	-	N8	Delivery period end date (YYYYMMDD)
invoiceappendix	-	AN..255	Dynamic text on the invoice

it[n]	+	Default	For BSV / KLV / KLS: Item type
			  goods Goods
			  shipment Shipping charges
			  handling Handling fee
			  voucher Voucher / discount
			Note:
			Item type "handling" only available after assignment by BillSAFE.

id[n]	+	AN..32	Product number, order number, etc. Permitted symbols: 0-9 a-z A-Z ()[]{} +-_#/:
pr[n]	+	N..8	Unit price (in smallest currency unit! e.g. cent)
no[n]	+	N..3	Quantity
de[n]	+	AN..255	Description (on invoice) For BSV maximum 100 characters are processed. For PPE maximum 127 characters are processed.
va[n]	-	N..4	VAT rate (% or bp) Mandatory for payment type KLV, KLS
sd[n]	-	N8	Delivery date (YYYYMMDD)
ed[n]	-	N.8	Delivery period end date (YYYYMMDD)
Parameter (personal data)			
customerid	-	AN1..20	Merchant's customer ID (Permitted symbols: 0-9, a-z, A-Z, .,-,_,/)
userid	-	N6..12	Debtor ID (PAYONE) userid must already exist and mode (test/live) must must match. If userid is used then personal data are loaded automatically.

businessrelation	o	Default	Will be available from 2017-12-01 or later. Will be required for payment type POV from 2018-01-01.
			Values
			  b2c Indicates business to consumer
			  b2b indicates business to business

Request "authorization"			
Parameter	Required	Format	Comment
salutation	-	AN1..10	Title (e.g. "Mr", "Mrs", "company")
title	-	AN1..20	Title (e.g. "Dr", "Prof.")
firstname	o	AN1..50	First name (optional if company is used) Mandatory for payment type KLV, KLS, PDT
lastname	+	AN2..50	Surname
company	-	AN2..50	Company
street	-	AN1..50	Street number and name (required: at least one character) Mandatory for payment type KLV, KLS mandatory if direct debit (ELV) used with IBAN-countries CH, SM, MC, PM, JE, GG
addressaddition	-	AN1..50	Address line 2 (e.g. "7th floor", "c/o Maier") Mandatory for payment type KLV, KLS in NL
zip	-	AN2..10	Postcode Mandatory for payment type KLV, KLS mandatory if direct debit (ELV) used with IBAN-countries CH, SM, MC, PM, JE, GG
city	-	AN2..50	City Mandatory for payment type KLV, KLS mandatory if direct debit (ELV) used with IBAN-countries CH, SM, MC, PM, JE, GG
state	-	Default	State (ISO 3166-2 subdivisions) (only if country=US, CA, CN, JP, MX, BR, AR, ID, TH, IN)
country	+	Default	Country (ISO 3166)
email	-	AN..50	Email address Mandatory for payment type BSV, KLV, P24
telephonenumber	-	AN..30	Telephone number Mandatory for payment type KLV, KLS
birthday	-	N8	Date of birth (YYYYMMDD) Mandatory for payment type KLV, KLS
language	-	Default	Language indicator (ISO 639) Mandatory for payment type KLV, KLS KLV, KLS support sv, nb (norwegian), fi, da, de, nl
vatid	-	AN..50	VAT identification number
gender	-	Default	f=female, m=male Mandatory for payment type KLV, KLS DE, NL, AT
personalid	-	AN..32	Person specific numbers or characters Mandatory for payment type KLV, KLS SE, FI, DK, NO Format/allowed characters: A-Z, a-z, 0-9, +-./()
ip	-	AN..39	Customer's IP-V4-address (123.123.123.123) or IP-V6-address

Request "authorization"			
Parameter	Required	Format	Comment
			Mandatory for payment type KLV, KLS
Parameter (delivery data)			
shipping_firstname	-	AN..50	First name
shipping_lastname	-	AN..50	Surname
shipping_company	-	AN..50	Company
shipping_street	-	AN1..50	Street number and name (required: at least one character)
shipping_zip	-	AN2..10	Postcode
shipping_city	-	AN2..50	City
shipping_state	-	Default	State (ISO 3166-2 subdivisions) (only if country=US, CA, CN, JP, MX, BR, AR, ID, TH, IN) Mandatory for PPE if shipping_country listed above
shipping_country	-	Default	Country (ISO 3166)
Parameter (debit payment)			
iban	o	AN..35	International Bank Account Number Only capital letters, no spaces If both (BBAN and IBAN) are submitted, IBAN is splitted into BBAN and processed.
bic	o	AN..11	Bank Identifier Code Only capital letters, no spaces
bankcountry	o	Default	Account type/ country For use with BBAN: DE Mandatory with bankcode, bankaccount Optional with iban/bic as bankcountry will be determined by iban
bankaccount	o	N..10	Account number (BBAN) DE only: IBAN/BIC can be calculated by PAYONE platform automatically given by bankcode/bankaccount
bankcode	o	N8	Sort code (BBAN) (only in DE) DE only: IBAN/BIC can be calculated by PAYONE platform automatically given by bankcode/bankaccount (BBAN)
bankaccountholder	-	AN..35	Account holder
mandate_identification	o	AN..35	Can be used to enforce a merchant specific mandate identification. The mandate_identification has to be unique. Allowed characters: A-Z a-z 0-9 + - . ( ) If the mandate_identification is not set PAYONE will create a unique mandate identification. PPS: This parameter must not be used!

Request "authorization"			
Parameter	**Required**	**Format**	**Comment**
Parameter (online transfer)			
onlinebanktransfertype	+	Default	PNT Sofortbanking (DE, AT, CH, NL) GPY giropay (DE) EPS eps – online transfer (AT) PFF PostFinance E-Finance (CH) PFC PostFinance Card (CH) IDL iDEAL (NL) P24 Przelewy24 (PL)
bankcountry	+	Default	Account type/ country (DE, AT, CH, NL, PL)
bankaccount[9]	o	N..10	Account number (giropay & Sofortbanking only) DE only: IBAN/BIC can be calculated by PAYONE
bankcode[10]	o	N8	Sort code (giropay & Sofortbanking only) DE only: IBAN/BIC can be calculated by PAYONE
bankgrouptype	o	Default	Bank Group (see chapter 5) (eps & iDEAL only)
iban[10]	o	AN..35	International Bank Account Number Only capital letters, no spaces If both (BBAN and IBAN) are submitted, IBAN is splitted into BBAN and processed.
bic[10]	o	AN..11	Bank Identifier Code Only capital letters, no spaces
successurl	o	AN..255	URL "payment successful" (only if responsetype=REDIRECT or required by corresponding request)
errorurl	o	AN..255	URL "faulty payment" (only if responsetype=REDIRECT or required by corresponding request)
backurl	o	AN..255	URL "Back" or "Cancel" (only if responsetype=REDIRECT or required by corresponding request)
Parameter (e-wallet)			
wallettype	+	Default	Wallet provider PPE: PayPal Express

---

[9] Can be omitted for OBT/PNT (Sofortbanking) from 2016-10-01 on as bank data are received from SOFORT and stored for later credits (processed as SEPA-credit).

Request "authorization"			
Parameter	**Required**	**Format**	**Comment**
successurl	o	AN..255	URL "payment successful" (only if responsetype=REDIRECT or required by corresponding request)
errorurl	o	AN..255	URL "faulty payment" (only if responsetype=REDIRECT or required by corresponding request)
backurl	o	AN..255	URL "Back" or "Cancel" (only if responsetype=REDIRECT or required by corresponding request)
Parameter (cash on delivery)			
shippingprovider	+	Default	Shipping company DHL: DHL, Germany BRT: Bartolini, Italy
Parameter (credit card)			
cardpan	+	N..19	Card number
cardtype	+	Default	Card type V Visa M MasterCard A American Express D Diners J JCB O Maestro International C Discover B Carte Bleue P China Union Pay
cardexpiredate	+	N4	Expiry date YYMM
cardcvc2	o	N..4	Credit verification number (CVC)
cardissuenumber	-	N..2	Card issue number (only Maestro UK cards)
cardholder	-	AN..50	Card holder
ecommercemode	-	Default	Credit card transaction type: internet eCommerce Transaction (SSL secured) 3dsecure 3-D Secure transaction (can be enabled alternatively in the risk settings) moto Mail or telephone order transaction
successurl	o	AN..255	URL "payment successful" (only if responsetype=REDIRECT or required by corresponding request)
errorurl	o	AN..255	URL "faulty payment" (only if responsetype=REDIRECT or required by corresponding request)

Request "authorization"			
Parameter	**Required**	**Format**	**Comment**
Parameter (credit card with pseudo card number)			
pseudocardpan	+	N..19	Pseudo card number (This card number can be submitted alternatively to the rest of the card data)
Key			
N..x	Numeric value (x characters maximum)		
AN..x	Alphanumeric value (x characters maximum)		

Response "authorization"			
Parameter	**Required**	**Format**	**Comment**
status	+	Default	APPROVED / REDIRECT / ERROR / PENDING
Parameter (APPROVED)			
txid	+	N..12	Payment process ID (PAYONE)
userid	+	N..12	Debtor ID (PAYONE)
Parameter (PENDING[10])			
txid	+	N..12	Payment process ID (PAYONE)
userid	+	N..12	Debtor ID (PAYONE)
Parameter (REDIRECT) (3-D Secure/online transfer/e-wallet)			
txid	+	N..12	Payment process ID (PAYONE)
userid	+	N..12	Debtor ID (PAYONE)
redirecturl	+	AN..255	Redirect URL
Parameter (ERROR)			
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message
customermessage	-	AN..1024	Error message for the end customer (Language selection is based on the end customer's language, „language") With "api_version>=3.10" you may get more detailed error messages from the external payment processor (e.g. Ratepay)
Parameter (credit card – if AVS has been ordered[11])			
protect_result_avs	-	A1	AVS return value, see chapter 5.3
Parameter (prepayment/invoice/cash on delivery)			
clearing_bankaccountholder	+	AN..35	Recipient bank account holder
clearing_bankcountry	+	Default	Recipient account type/ country (e.g. DE, AT, etc.)
clearing_bankaccount	+	AN..26	Recipient account number
clearing_bankcode	o	AN..11	Recipient sort code
clearing_bankiban	-	AN..35	Recipient IBAN
clearing_bankbic	-	AN..11	Recipient BIC
clearing_bankcity	-	AN..50	Recipient city or bank
clearing_bankname	-	AN..50	Recipient bank name

[10] New response value – only available with "api_version=3.9". Indicates that a payment process is pending at external payment processor. If "api_version=3.8" or not present then "APPROVED" is returned for compatibility.

[11] AVS (Address Verification System) is currently only supported for American Express

clearing_instructionnote	o	AN..200	For BSV: Note to payment handling For KLV, KLS: A URL pointing to a PDF of the invoice (If invoice by post or by e-mail isn't activated). The URL is valid for 30 days.
Parameter (direct debit[12])			
mandate_identification	-	AN..35	used mandate_identification for debit payment
mandate_dateofsignature	-	N..10	date when mandate has been created (format YYYYMMDD)
creditor_identifier	-	AN..35	used CID for debit payment
creditor_name	-	AN..35	as recorded in PAYONE Platform
creditor_street	-	AN..35	as recorded in PAYONE Platform
creditor_zip	-	AN..10	as recorded in PAYONE Platform
creditor_city	-	AN..50	as recorded in PAYONE Platform
creditor_country	-	Default	Country (ISO 3166)
creditor_email	-	AN..50	as recorded in PAYONE Platform
clearing_date	-	N..8	only provided if due_time is not specified
clearing_amount	-	N..10	only provided if due_time is not specified

---

[12] **Parameters will not be provided currently to prevent declined requests by merchant systems that do not expect these key/values by now. Parameters have to be enabled in PMI, Configuration, Payment portals, General: "Additional response-data" for mode test and live. Settings will be available by end of November.**

3.2.3 Creating a contract (createaccess)

For processing a purchase transaction in connection with session management or with initiating a subscription

This request is only available with the version "Access".

Important notes:

- For the time-based settlement of goods or subscriptions you must first create offers in your merchant area.
- You have the option to submit values that overwrite the settings for the offer used (e.g. price, term, etc.). These optional submitted values will be used for all following periods of this contract. The original offer is not changed.
- If the time at which access is first granted is in the future or the initial term is free of charge, a reservation, which will later be cancelled automatically, is carried out. The amount for the reservation is one payment unit (e.g. 1 EUR). This procedure is simply used to verify the customer's payment data and to check the customer data using risk checks, provided that these are active. The first "real" booking will be effected at the start of the first period that requires a charge.
- The periods (defined by "period_unit_trail", "period_length_trail", "period_unit_recurring", "period_length_ recurring") are limited to 36 months / 1095 days maximum.

Request "createaccess"			
Parameter	**Required**	**Format**	**Comment**
aid	+	N..6	Sub account ID
clearingtype	+	Default	elv Debit payment cc Credit card rec Invoice
reference	+	AN..20	Merchant reference number for the payment process. (Permitted symbols: 0-9, a-z, A-Z, .,-,_,/) special limits / restrictions: GPY min/max: AN 4..16
productid	+	N..7	ID for the offer
param	-	AN1..255	Individual parameter (This parameter is passed on to all follow-up bookings of the same contract)
accessname	-	AN..32	Customer's user name
accesscode	-	AN..32	Customer's password
access_starttime	-	N..11	Access starting date as Unixtimestamp
access_expiretime	-	N..11	Expiry date of the initial term as Unixtimestamp (e.g. date of first renewal) -> should not be used. Please use "period_unit_trail", "period_length_trail" instead. -> do not use with "period_unit_trail", "period_length_trail"
access_canceltime	-	N..11	Date of cancellation as Unixtimestamp in the future (after this date no further renewals will be carried out)
amount_trail	o	N..6	Total amount for initial term Must equal the sum (quantity x price) of all items for the initial term. (in smallest currency unit! e.g. cent) Required when item is submitted. Amount can be "0" (e.g. for test period).
period_unit_trail	-	Default	Time unit for initial term, possible values: Y Value "length" is in years M Value "length" is in months D Value "length" is in days Do not use with "access_expiretime". Do not exceed 5 years / 60 months.
period_length_trail	o	N..4	Duration of the initial term. Can only be used in combination with period_unit_trail. Required when period_unit_trail is submitted. Do not use with "access_expiretime"

Request "createaccess"			
Parameter	**Required**	**Format**	**Comment**
amount_recurring	o	N..6	Total amount of all items of one period during the subsequent term (in smallest currency unit! e.g. cent)
			Must equal the sum (quantity x price) of all items during the subsequent term.
			Required when item is submitted.
			Amount must not be "0".
period_unit_recurring	-	Default	Time unit for subsequent term, possible values:
			Y Value "length" is in years
			M Value "length" is in months
			D Value "length" is in days
			N No subsequent term given
			Do not exceed 5 years / 60 months.
period_length_recurring	o	N..3	Duration of the subsequent term. Can only be used in combination with period_unit_recurring.
			Required when period_length_recurring is submitted.
Parameter (Billing)			
vaccountname	+	AN1..20	ID/name of the merchant's billing account (allowed characters: 0-9, a-z, A-Z, .,-,_,/)
settle_period_length	+	N1..3	Length of the accounting period
settle_period_unit	+	Default	Time unit of the accounting period, possible values:
			Y Value "length" is in years
			M Value "length" is in months
			D Value "length" is in days
settletime	-	N10	Date for the next invoicing Unixtimestamp in the future.
payout_open_balance	-	Default	Credits will be booked on the most recent payment process.
			yes Amount will be credited immediately (default)
			no Credited amount will be offset with the subsequent transaction
Parameter (Invoicing)			
invoice_deliverymode	-	Default	M Post
			P PDF (via email)
invoiceappendix	-	AN..255	Dynamic text on the invoice
id_trail[n]	+	AN..32	Item number (initial term)
			Permitted symbols: 0-9 a-z A-Z ()[]{} +-_#/:

Request "createaccess"			
Parameter	**Required**	**Format**	**Comment**
no_trail[n]	+	N..5	Quantity (initial term)
pr_trail[n]	+	N..7	Unit price of the item in smallest currency unit (initial term)
de_trail[n]	+	AN..255	Description (initial term)
va_trail[n]	-	N..4	VAT rate (% or bp) (initial term) value <= 99 = per cent value > 99 = basis points (e.g. 1900 = 19%)
ti_trail[n]	-	AN..100	Title (initial term)
id_recurring[n]	+	AN..32	Item number (subsequent term) Permitted symbols: 0-9 a-z A-Z ()[]{} +-_#/:
no_recurring[n]	+	N..5	Quantity (subsequent term)
pr_recurring[n]	+	N..7	Unit price of the item in smallest currency unit (subsequent term)
de_recurring[n]	+	AN..255	Description (subsequent term)
va_recurring[n]	-	N..4	VAT rate (% or bp) (subsequent term) value <= 99 = per cent value > 99 = basis points (e.g. 1900 = 19%)
ti_recurring[n]	-	AN..100	Title (subsequent term)
Parameter (personal data)			
customerid	-	AN1..20	Merchant's customer ID (Permitted symbols: 0-9, a-z, A-Z, .,-,_,/)
userid	-	N6..12	Debtor ID (PAYONE) userid must already exist and mode (test/live) must must match. If userid is used then personal data are loaded automatically.
salutation	-	AN1..10	Title (e.g. "Mr", "Mrs", "company")
title	-	AN1..20	Title (e.g. "Dr", "Prof.")
firstname	o	AN1..50	First name (optional if company is used)
lastname	+	AN2..50	Surname
company	-	AN2..50	Company
street	-	AN1..50	Street number and name (required: at least one character) mandatory if direct debit (ELV) used with IBAN-countries CH, SM, MC, PM, JE, GG
addressaddition	-	AN1..50	Address line 2 (e.g. "7th floor", "c/o Maier")
zip	-	AN2..10	Postcode mandatory if direct debit (ELV) used with IBAN-countries CH, SM, MC, PM, JE, GG
city	-	AN2..50	City

Request "createaccess"			
Parameter	**Required**	**Format**	**Comment**
			mandatory if direct debit (ELV) used with IBAN-countries CH, SM, MC, PM, JE, GG
country	+	Default	Country (ISO 3166)
email	-	AN..50	Email address Mandatory for payment type BSV
telephonenumber	-	AN..30	Telephone number
birthday	-	N8	Date of birth (YYYYMMDD)
language	-	Default	Language indicator (ISO 639)
vatid	-	AN..50	VAT ident. number
ip	-	AN..39	Customer's IP-V4-address (123.123.123.123) or IP-V6-address
Parameter (debit payment)			
iban	o	AN..35	International Bank Account Number Only capital letters, no spaces If both (BBAN and IBAN) are submitted, IBAN is splitted into BBAN and processed. BBAN parameters are ignored.
bic	o	AN..11	Bank Identifier Code Only capital letters, no spaces
bankcountry	o	Default	Account type/ country For use with BBAN: DE Mandatory with bankcode, bankaccount Optional with iban/bic as bankcountry will be determined by iban
bankaccount	o	N..10	Account number (BBAN) DE only: IBAN/BIC can be calculated by PAYONE platform automatically given by bankcode/bankaccount
bankcode	o	N8	Sort code (BBAN) (only in DE) DE only: IBAN/BIC can be calculated by PAYONE platform automatically given by bankcode/bankaccount (BBAN)
bankaccountholder	-	AN..35	Account holder
mandate_identification	o	AN..35	Can be used to enforce a merchant specific mandate identification. The mandate_identification has to be unique. Allowed characters: A-Z a-z 0-9 + - . ( ) If the mandate_identification is not set PAYONE will create an unique mandate identification. PPS: This parameter must not be used!
Parameter (credit card)			
cardpan	+	N..19	Card number
cardtype	+	Default	Card type

Request "createaccess"			
Parameter	Required	Format	Comment
			V Visa
			M MasterCard
			A American Express
			D Diners
			J JCB
			O Maestro International
			C Discover
			B Carte Bleue
			P China Union Pay
cardexpiredate	+	N4	Expiry date YYMM
cardcvc2	o	N..4	Credit verification number (CVC)
cardissuenumber	-	N..2	Card issue number (only Maestro UK cards)
cardholder	-	AN..50	Card holder
pseudocardpan	o	N..19	Pseudo card number (This card number can be submitted alternatively to the rest of the card data)
Key			
N..x	Numeric value (x characters maximum)		
AN..x	Alphanumeric value (x characters maximum)		

* 3-D Secure protection is only available for the first payment. Access for this may not be in the future.

Response "createaccess"			
Parameter	Required	Format	Comment
status	+	Default	APPROVED / ERROR
Parameter (APPROVED)			
txid	+	N..12	Payment process ID (PAYONE)
userid	+	N..12	Debtor ID (PAYONE)
accessid	+	N..12	Access ID (PAYONE)
Parameter (ERROR)			
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message
customermessage	-	AN..1024	Error message for the end customer (Language selection is based on the end customer's language, „language")

3.2.4 Initiating payment process on a virtual account (vauthorization)

For booking a request on a virtual account.

PAYONE Platform can automatically initiate a settlement to settle the balance of the account.

This option is available in the "Shop" and "Access" option.

Request "vauthorization"			
Parameter	Required	Format	Comment
aid	+	N..6	Sub account ID
clearingtype	+	Default	elv Debit payment cc Credit card rec Invoice
vreference	+	AN..20	Merchant's transaction reference number (Permitted symbols: 0-9, a-z, A-Z, .,-,_,/)
vaccountname	+	AN1..20	ID/name of the merchant's billing account (Permitted symbols: 0-9, a-z, A-Z, .,-,_,/)
amount	+	N..7	Total amount (in smallest currency unit! e.g. cent)
currency	+	Default	Currency (ISO 4217)
settle_period_length	+	N1..3	Length of the accounting period
settle_period_unit	+	Default	Time unit of the accounting period, possible values: Y: Value in years M: Value in months D: Value in days
settletime	-	N10	Date of the next automatic invoice (Unixtimestamp)
payout_open_balance	-	Default	Credits will be booked on the most recent payment process. yes: Amount will be credited immediately no: Credited amount will be offset with the subsequent transaction default: Value given with request for "createaccess"
param	-	AN1..255	Individual parameter
Parameter (Invoicing)			
invoice_deliverymode	-	Default	M Post P PDF (via email) N no delivery
id[n]	+	AN..32	Product number, order number, etc. Permitted symbols: 0-9 a-z A-Z ()[]{} +-_#/:
pr[n]	+	N..8	Unit price (in smallest currency unit! e.g. cent)
no[n]	+	N..3	Quantity
de[n]	+	AN..255	Description (on invoice)
va[n]	-	N..4	VAT rate (% or bp)
Parameter (personal data)			
customerid	-	AN1..20	Merchant's customer ID (Permitted symbols: 0-9, a-z, A-Z, .,-,_,/)

Request "vauthorization"			
Parameter	**Required**	**Format**	**Comment**
userid	-	N6..12	Debtor ID (PAYONE) userid must already exist and mode (test/live) must must match. If userid is used then personal data are loaded automatically.
salutation	-	AN1..10	Title (e.g. "Mr", "Mrs", "company")
title	-	AN1..20	Title (e.g. "Dr", "Prof.")
firstname	o	AN1..50	First name (optional if company is used)
lastname	+	AN2..50	Surname
company	-	AN2..50	Company
street	-	AN1..50	Street number and name (required: at least one character) mandatory if direct debit (ELV) used with IBAN-countries CH, SM, MC, PM, JE, GG
addressaddition	-	AN1..50	Address line 2 (e.g. "7th floor", "c/o Maier")
zip	-	AN2..10	Postcode mandatory if direct debit (ELV) used with IBAN-countries CH, SM, MC, PM, JE, GG
city	-	AN2..50	City mandatory if direct debit (ELV) used with IBAN-countries CH, SM, MC, PM, JE, GG
country	+	Default	Country (ISO 3166)
email	-	AN..50	Email address Mandatory for payment type BSV
telephonenumber	-	AN..30	Telephone number
birthday	-	N8	Date of birth (YYYYMMDD)
language	-	Default	Language indicator (ISO 639)
vatid	-	AN..50	VAT ident. number
ip	-	AN..39	Customer's IP-V4-address (123.123.123.123) or IP-V6-address
Parameter (debit payment)			
iban	o	AN..35	International Bank Account Number If both (BBAN and IBAN) are submitted, IBAN is splitted into BBAN and processed. BBAN parameters are ignored.
bic	o	AN..11	Bank Identifier Code Only capital letters, no spaces
bankcountry	o	Default	Account type/ country For use with BBAN: DE Mandatory with bankcode, bankaccount Optional with iban/bic as bankcountry will be determined by iban
bankaccount	o	N..10	Account number (BBAN)

Request "vauthorization"			
Parameter	Required	Format	Comment
			DE only: IBAN/BIC can be calculated by PAYONE platform automatically given by bankcode/bankaccount
bankcode	o	N8	Sort code (BBAN) (only in DE) DE only: IBAN/BIC can be calculated by PAYONE platform automatically given by bankcode/bankaccount (BBAN)
bankaccountholder	-	AN..35	Account holder
mandate_identification	o	AN..35	Can be used to enforce a merchant specific mandate identification. The mandate_identification has to be unique. Allowed characters: A-Z a-z 0-9 + - . ( ) Only capital letters, no spaces If the mandate_identification is not set PAYONE will create an unique mandate identification. PPS: This parameter must not be used!
Parameter (credit card)			
cardpan	+	N..19	Card number
cardtype	+	Default	Card type V Visa M MasterCard A American Express D Diners J JCB O Maestro International C Discover B Carte Bleue P China Union Pay
cardexpiredate	+	N4	Expiry date YYMM
cardcvc2	o	N..4	Credit verification number (CVC)
cardissuenumber	-	N..2	Card issue number (only Maestro UK cards)
cardholder	-	AN..50	Card holder
Key			
N..x	Numeric value (x characters maximum)		
AN..x	Alphanumeric value (x characters maximum)		

Response "vauthorization"			
Parameter	**Required**	**Format**	**Comment**
status	+	Default	APPROVED / ERROR
Parameter (APPROVED)			
vxid	+	N..12	Reference number on the virtual account
vaid	+	N..8	Billing account ID (PAYONE)
userid	+	N..12	Debtor ID (PAYONE)
Parameter (ERROR)			
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message
customermessage	-	AN..1024	Error message for the end customer (Language selection is based on the end customer's language, „language")

3.3   Administration

3.3.1   Create a mandate (managemandate)

Create a pending mandate – used for SEPA Direct Debit. Mandate will be activated and persisted by a payment-request (preauthorization / authorization) or by customer management (updateuser) – this request has to follow a "managemandate"-request within 2 hours.

Request "managemandate"			
Parameter	**Required**	**Format**	**Comment**
aid	+	N..6	Sub account ID
clearingtype	+	Default	elv Debit payment
mandate_identification	-	AN..35	Can be used to enforce a merchant specific mandate identification. The mandate_identification must be unique. Allowed characters: A-Z a-z 0-9 + - . ( )
			If the mandate_identification is not set PAYONE will create an unique mandate identification (pattern: PO-nnnnnnnnnn).
			PPS: This parameter must not be used!
currency	+	Default	Currency (ISO 4217)
Parameter (personal data)			
customerid	o	AN1..20	Merchant's customer ID
			(Permitted symbols: 0-9, a-z, A-Z, .,-,_,/)
			either
			• existing customerid
			• or existing userid
			• or lastname/city/bankcountry/IBAN/BIC
			must be given
userid	o	N6..12	Debtor ID (PAYONE)
			userid must already exist and mode (test/live) must must match.
			If userid is used then personal data are loaded automatically.
lastname	+	AN..50	Surname
firstname	-	AN1..50	First name (optional if company is used)
company	-	AN2..50	Company
street	-	AN1..50	Street number and name (required: at least one character)
zip	-	AN2..10	Postcode
city	o	AN2..50	City
			only optional if already recorded at existing customer referenced by userid/customerid
country	+	Default	Country (ISO 3166)
email	o	AN..50	Email address
language	-	Default	Language indicator (ISO 639), default="de"

Request "managemandate"			
Parameter	Required	Format	Comment
Parameter (bank account data – Germany only) optional if already recorded at existing customer referenced by userid/customerid			
bankcountry	+	Default	Account type/ country For use with BBAN: DE Mandatory with bankcode, bankaccount Optional with iban/bic as bankcountry will be determined by iban
bankaccount	+	N..10	Account number (BBAN) (only for DE) DE only: IBAN/BIC can be calculated by PAYONE platform automatically given by bankcode/ bankaccount
bankcode	+	N8	Sort code (BBAN) (only for DE) DE only: IBAN/BIC can be calculated by PAYONE platform automatically given by bankcode/ bankaccount (BBAN)
Parameter (bank account data – Germany plus all other countries) optional if already recorded at existing customer referenced by userid/customerid			
bankcountry	o	Default	Account type/ country (ISO 3166) if not specified it will be determined by IBAN
iban	+	AN..35	International Bank Account Number Only capital letters and digits, no spaces only optional if already recorded at existing customer referenced by userid/customerid
bic	+	AN..11	Bank Identifier Code Only capital letters and digits, no spaces

Response "managemandate"			
Parameter	Required	Format	Comment
status	+	Default	APPROVED / ERROR
Parameter (APPROVED)			
mandate_identification	+	AN..35	Mandate identification (either taken from request or newly created)
mandate_status	+	Default	pending: for a new created mandates active: for existing / active mandates
mandate_text	+	TEXT	Content of mandate text HTML-formatted, URL-encoded
creditor_identifier	+	AN..35	Creditor identifier as recorded at PAYONE
iban	+	AN..35	IBAN
bic	o	AN..11	BIC, if given in request

Parameter (ERROR) -> actual response; will be always used for errournous requests **Parameter (INVALID)** -> may be used in future Q3/2016 for invalid data in request			
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message for the merchant
customermessage	-	AN..1024	Error message for the end customer (Language selection is based on the end customer's language, „language")

3.4   Verification / Scoring

3.4.1 Verifying credit cards (creditcardcheck)

The CreditCardCheck checks credit cards for plausibility in real-time.

Additionally, this request offers the possibility to store credit card data. In this case, the response submits what is known as a pseudo card number.

Request "creditcardcheck"			
Parameter	**Required**	**Format**	**Comment**
aid	+	N..6	Sub account ID
cardpan	+	N..19	Card number
cardtype	+	Default	Card type V Visa M MasterCard A American Express D Diners J JCB O Maestro International C Discover B Carte Bleue P China Union Pay
cardexpiredate	+	N4	Expiry date YYMM
cardcvc2	o	N..4	Credit verification number (CVC)
cardissuenumber	-	N..2	Card issue number (only Maestro UK cards)
storecarddata	-	Default	no: Card data is not stored yes: Card data is stored, a pseudo card number is returned
language	-	Default	Language indicator (ISO 639)
Key			
N..x	Numeric value (x characters maximum)		
AN..x	Alphanumeric value (x characters maximum)		

Response "creditcardcheck"			
Parameter	**Required**	**Format**	**Comment**
status	+	Default	VALID = card number valid INVALID = card number invalid ERROR = request faulty
Parameter (VALID)			
pseudocardpan	o	N..19	Pseudo card number (is returned if storecarddata=yes)
truncatedcardpan	o	AN..19	Masked cardpan, i.e. 411111xxxxxx1111 (is returned if storecarddata=yes)
cardtype	o	Default	Cardtype selected by consumer
cardexpiredate	o	N..4	Card expiredate selected by consumer in YYMM
Parameter (INVALID)			
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message
customermessage	-	AN..1024	Error message for the end customer (Language selection is based on the end customer's language, „language")
Parameter (ERROR)			
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message
customermessage	-	AN..1024	Error message for the end customer (Language selection is based on the end customer's language, „language")

3.4.2 Account verification (bankaccountcheck)

The BankAccountCheck checks the plausibility of German account details according to the current check digit algorithms and carries out a POS blacklist inquiry if required.

Request "bankaccountcheck"			
Parameter	**Required**	**Format**	**Comment**
aid	+	N..6	Sub account ID
checktype	-	Default	0 = regular check (default)
			1 = check against POS blacklist (only DE)
language	-	Default	Language indicator (ISO 639)
Check of international bank account			
bankcountry	o	Default	Account type/ country
			Optional with iban/bic as bankcountry will be determined by iban
iban	+	AN..35	IBAN[13, 14]
			Only capital letters and digits, no spaces
			If both (BBAN and IBAN) are submitted, IBAN is splitted into BBAN and processed. BBAN parameters are ignored.
bic	+	AN..11	BIC
			Only capital letters and digits, no spaces
Check of national bank account			
bankcountry	+	Default	"DE" only
bankaccount	+	N..10	Account number [15]
bankcode	+	N8	Sort code
Key			
N..x			Numeric value (x characters maximum)
AN..x			Alphanumeric value (x characters maximum)

---

[13] IBAN / BIC will be validated. It's not checked whether IBAN and BBAN (Bankcode/Bankaccount) refer to the same account.

[14] For Germany IBAN, BIC will be converted into bankcode, bankaccount automatically

[15] For Germany bankcode, bankaccount will be converted into IBAN, BIC automatically

Response "bankaccountcheck"			
Parameter	**Required**	**Format**	**Comment**
status	+	Default	VALID = account details valid
			INVALID / ERROR = account details invalid
			BLOCKED = account details found on blacklist (checktype=1 only)
			ERROR = request faulty
Parameter (VALID)			
bankcountry	o	Default	Account type/ country (ISO-3166)
			Please find available checks in chapter 5.4
bankcode	o	AN..11	Sort code in standardised country format (if standardisation is supported)
bankaccount	o	AN..26	Sort Account number in standardised country format (if standardisation is supported)
bankbranchcode	o	N..5	Branch code
bankcheckdigit	o	AN..2	Check digit
iban	o	AN..35	IBAN[16]
bic	o	AN..11	BIC
Parameter (INVALID)			
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message
customermessage	-	AN..1024	Error message for the end customer
			(Language selection is based on the end customer's language, „language")
Parameter (ERROR)			
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message
customermessage	-	AN..1024	Error message for the end customer
			(Language selection is based on the end customer's language, „language")

[16] **Parameters will not be provided currently to prevent declined requests by merchant systems that do not expect these key/values by now. Parameters have to be enabled in PMI, Configuration, Payment portals, General: "Additional response-data" for mode test and live.**

3.4.3 Address verification (addresscheck)

Verification and correction of address details.

The verification of the address is carried out to minimize the risk of non-payment and to correct the entered address information. Deviations in the address details are corrected automatically.

For best results please provide "streetname" and "streetnumber" in separate fields. If not possible you can use "street" only.

For "addresscheck person" first-/last name is obviously mandatory. Gender and birthdate are optional but recommended for best results.

Request "addresscheck"			
Parameter	**Required**	**Format**	**Comment**
aid	+	N..6	Sub account ID
addresschecktype	+	Default	PAYONE-Services BA Addresscheck Basic PE Addresscheck Person Boniversum-Services BB Addresscheck Basic PB Addresscheck Person
firstname	o	AN1..50	First name (required for "Addresscheck Person")
lastname	o	AN2..50	Surname (required for "Addresscheck Person")
company	-	AN2..50	Company
street	o	AN1..50	Street number and name (required if "streetname" and "streetnumber" not submitted) (required: at least one character)
streetname	o	AN..50	Street name (required if "street" not submitted)
streetnumber	o	AN..5	Street number (required if "street" not submitted)
zip	+	AN2..10	Postcode
city	+	AN2..50	City
state	o	Default	State (ISO 3166-2 subdivisions) (only if country=US, CA, CN, JP, MX, BR, AR, ID, TH, IN)
country	+	AN..3	Country (ISO 3166) Check "BB", "PB": only "DE" supported
birthday	-	N8	Date of birth (YYYYMMDD)
telephonenumber	-	AN..30	Telephone number (0049123456789)
gender	-	Default	f=female, m=male
language	-	Default	Language indicator (ISO 639)
Key			
N..x		Numeric value (x characters maximum)	
AN..x		Alphanumeric value (x characters maximum)	

Response "addresscheck"			
Parameter	**Require d**	**Format**	**Comment**
status	+	Default	VALID = address valid
			INVALID = address invalid
			ERROR = error
Parameter (VALID)			
secstatus	+	N2	10 = address correct
			20 = address could be corrected
personstatus	+	Default	Person address check result; details see code list ("Parameters for address check person returns")
firstname	-	AN..50	Return of firstname (corrected if applicablce)
lastname	-	AN..50	Return of lastname (corrected if applicable)
street	+	AN..50	Return of street + street no. (corrected if applicable)
streetname	-	AN..50	Return of street name (corrected if applicable)
streetnumber	-	AN..5	Return of street name (corrected if applicable)
zip	+	AN..10	Return of postal code (corrected if applicable)
city	+	AN..50	Return of city (corrected if applicable)
Parameter (INVALID)			
secstatus	+	N2	30 = address could not be corrected
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message
customermessage	-	AN..1024	Error message for the end customer
			(Language selection is based on the end customer's language, „language")
Parameter (ERROR)			
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message
customermessage	-	AN..1024	Error message for the end customer
			(Language selection is based on the end customer's language, „language")

3.4.4 Verification of creditworthiness (consumerscore)

Checking customer data for any existing adverse criteria.

The customer check is carried out to determine the risk of non-payment of demands against customers. With the help of consumer information (adverse criteria) a score value is determined which you can use to assess the creditworthiness of the customers, get a realistic picture of the risk of non-payment and offer the corresponding types of payment.

For best results please provide "streetname" and "streetnumber" in separate fields. If not possible you can use "street" only.

For "consumerscore" first-/last name is obviously mandatory. Gender and birthdate are optional but recommended for best results.

Request "consumerscore"			
Parameter	**Required**	**Format**	**Comment**
aid	+	N..6	Sub account ID
addresschecktype	+	Default	**PAYONE-Services** BA Addresscheck Basic PE Addresscheck Person NO Do not carry out address check **Boniversum-Services** – must be combined with consumerscoretype "CE" PB Addresscheck Person
consumerscoretype	+	Default	**Arvato Infoscore Services** – must be combined with addresschecktype "NO", "BA", "PE" IH Infoscore (hard criteria) IA Infoscore (all criteria) IB Infoscore (all criteria + bonus score) IF Infoscore (all criteria + bonus score) + Informa-Consumer-Score **Boniversum Services** – must be combined with addresschecktype "PB" CE Boniversum VERITA Score including special addresses (status-codes AS1..AS8)

Request "consumerscore"			
Parameter	**Required**	**Format**	**Comment**
firstname	+	AN1..50	First name
lastname	+	AN2..50	Surname
company	-	AN2..50	Company
street	+	AN1..50	Street number and name (required if "streetname" and "streetnumber" not submitted) (required: at least one character)
streetname	o	AN..50	Street name (required if "street" not submitted)
streetnumber	o	AN..5	Street number (required if "street" not submitted)
zip	+	AN2..10	Postcode
city	+	AN2..50	City
country	+	Default	Country (ISO 3166) , actually only "DE" supported
birthday	-	N8	Date of birth (YYYYMMDD)
telephonenumber	-	AN..30	Telephone number (0049123456789)
gender	-	Default	f=female, m=male
language	-	Default	Language indicator (ISO 639)
Key			
N..x	Numeric value (x characters maximum)		
AN..x	Alphanumeric value (x characters maximum)		

Response "consumerscore"			
Parameter	**Required**	**Format**	**Comment**
status	+	Default	VALID address valid INVALID address invalid ERROR error
Parameter (VALID)			
secstatus	o	N2	Only provided for addresschecktype BA/PE 10 = address correct 20 = address could be corrected
score	+	Default	G "green", high degree of creditworthiness Y "yellow", average degree of creditworthiness R "red", low degree of creditworthiness U Unknown
scorevalue	-	N..4	**Arvato Infoscore**: 100-990 The **higher** the value, the **higher** the degree of creditworthiness **Boniversum VERITA Score**: 0-6000 The **higher** the value, the **lower** the degree of creditworthiness
secscore	-	AN..9	Negative criteria; details see code list ("Parameters for Consumerscore returns") only provided for consumerscoretype "IA", "IB", "IH" and "IF"
personstatus	+	Default	Person address check result; details see code list ("Parameters for address check person returns")
addressstatus	-	Default	Person address check result; details see code list ("Consumerscore Boniversum VERITA score - addressstatus") Only available with "CE" and "PB" and only if special addresses indicated.
firstname	+	AN..50	Return of first name (corrected if applicable)
lastname	+	AN..50	Return of surname (corrected if applicable)
street	+	AN..50	Return of street + street no. (corrected if applicable)
streetname	-	AN..50	Return of street name (corrected if applicable)
streetnumber	-	AN..5	Return of street number (corrected if applicable)
zip	+	AN..10	Return of postal code. (corrected if applicable)
city	+	AN..50	Return of city. (corrected if applicable)
Parameter (INVALID)			
secstatus	+	N2	30 = address could not be corrected
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message
customermessage	-	AN..1024	Error message for the end customer

Response "consumerscore"			
			(Language selection is based on the end customer's language, „language")
Parameter (ERROR)			
errorcode	+	N..6	Error number
errormessage	+	AN..1024	Error message
customermessage	-	AN..1024	Error message for the end customer
			(Language selection is based on the end customer's language, „language")

4 Responses

4.1 Parameter for SessionStatus query

According to the configuration of your payment portal you will receive access status changes for accesses to your premium sector. You will only receive these status messages with payment portals of the "Access" version. You can use them to protect your premium sector or to receive information about a subscription. The data is submitted to the URL specified in the merchant area. The data transfer is based on simple HTTP-POST request (key/value pairs).

The SessionStatus is sent from the following IP addresses: 185.60.20.0/24 (i.e. 185.60.20.1 to 185.60.20.254). Please configure your firewall to allow incoming packets from these IP addresses.

HTTP request from PAYONE to the merchant's server			
Parameter	**Required**	**Format**	**Comment**
key	+	AN..32	Key can be selected freely (see options payment portal) as MD5 value (The key hash values is currently given as MD5. This currently still remains with MD5 and is subject to change in future to SHA2-384.)
clearingstyp	+	Default	Type of payment used for this access. elv Debit payment cc Credit card vor Prepayment rec Invoice sb Online bank transfer
serverip	-	Default	payone.de -> this parameter may be removed in future.
accessid[x]	+	N3..12	Access ID (PAYONE)
action[x]	+	Default	Event, which refers to one customer each. "add" , "remove", "abocancel", "renew "cancel_reversal", "lock", "unlock" (see below)
portalid[x]	+	N..7	Payment portal ID
productid[x]	+	N..7	ID for the offer
expiretime[x]	+	N..12	Unix timestamp at which access expires
userid[x]	+	N..12	Debtor ID (PAYONE)
customerid[x]	-	AN1..20	Merchant's customer ID
accessname[x]	-	AN..32	Customer's user name
accesscode[x]	-	AN..32	Customer's password
ip[x]	-	AN..15	Customer IP
param[x]	-	AN..15	Individual parameter
Key	**x**	**x**	**x**
N..x	x	x	Numeric value (x characters maximum)
AN..x	x	x	Alphanumeric value (x characters maximum)

[x]	x	x	In this manner changes for several customers can be submitted simultaneously in one request.
			[x] = position number, e.g. [0],[1],...)

Important note: Please note that new parameters may be added at any time without previous notice. Therefore, you should use the parameter name for the evaluation and not the sequence, which may be subject to change at any time!

Expected reply to the request:

As a reply to the request, the string "**SSOK**" is expected. Each request is repeated in a 1-hour cycle until it is answered with "SSOK". This procedure ensures that all requests will be processed by your system. Simply issue the "SSOK" in a script via the "print" command. Make sure that this character string is the first that is printed from this script, e.g. `print ("SSOK");`

Important:

- Status messages from PAYONE to merchant's server are always ISO-8859-1 encoded.

- No other characters may be issued from this character string, e.g. no HTML code.

- Do not return an error without gathering information about this error.

- Make sure the request is always answered with an SSOK within 48 hours.

- If a specific request shall not be processed, issue an SSOK anyway to prevent the request from interfering with the processing of other requests.

- Without the return of an SSOK you will not receive any further status reports for that subscription.

- The request must be answered with "SSOK" within 10 seconds. This time-out cannot be extended. The answer does only confirm receipt of the SessionStatus, the evaluation can and should follow asynchronously to receiving the answer.

Sequence of events

After the start of the initial term an "add" request is deployed to your system. Different pieces of information about this customer are submitted (see above). After the access has expired, you will receive a "remove" request.

Events (action)

With each access status change you receive a request. Via the "action" variable you receive information about the status of the access.

add

An access portal has been opened.

remove

Access has expired and will not be renewed.

renew

Access was renewed/reduced (e.g. renewal of a subscription).

abocancel

The customer has cancelled the subscription for this access portal.

lock

Access has been blocked.

unlock

Access has been unblocked.

cancel_reversal

The termination of the subscription has been revoked.

4.2   Parameter for the TransactionStatus query

According to the configuration of your payment portal you will receive the data and the status for each payment process via the URL you have submitted. The data transfer is based on simple HTTP-POST request (key/value pairs).

The TransactionStatus is sent from the following IP addresses: 185.60.20.0/24 (i.e. 185.60.20.1 to 185.60.20.254). Please configure your firewall to allow incoming packets from these IP addresses.

HTTP request from PAYONE to the merchant's server			
Parameter	**Req uire d**	**For mat**	**Comment**
key	+	AN.. 32	Payment portal key as MD5 value (The key hash values is currently given as MD5. This currently still remains with MD5 and is subject to change in future to SHA2-384.)
txaction	+	Defa ult	"appointed", "capture", "paid", "underpaid", "cancelation", "refund", "debit", "reminder", "vauthorization", "vsettlement", "transfer", "invoice", "failed" (See explanation below)
transaction_stat us	-	Defa ult	"completed", "pending" new parameter in use from January 2015
notify_version	-	Defa ult	7.3 without "notify_version" and without "transaction_status" 7.4 with "notify_version" and with "transaction_status" (completed/pending) 7.5 with txaction "failed" (actually supported for Barzahlen with expired refund; other payment types will follow)
mode	+	Defa ult	test Test mode live Live mode
portalid	+	N..7	Payment portal ID
aid	+	N..6	Sub account ID
clearingtype	+	Defa ult	elv Debit payment cc Credit card vor Prepayment

HTTP request from PAYONE to the merchant's server			
Parameter	**Req uire d**	**For mat**	**Comment**
			rec Invoice cod Cash on delivery sb Online bank transfer wlt e-Wallet fnc Financing
txtime	+	N..1 1	Initiating payment process (Unix timestamp)
currency	+	Defa ult	Currency (ISO 4217)
userid	+	N..1 2	Debtor ID (PAYONE)
customerid	-	AN1. .20	Merchant's customer ID
param	-	AN.. 255	Individual parameter that was, where applicable, submitted while payment was initiated
Parameter (personal data)			
Invoice-Address			
firstname	o	AN.. 50	First name (optional if company is used)
lastname	+	AN.. 50	Surname
company	-	AN.. 50	Company
street	-	AN.. 50	Street number and name
zip	-	AN.. 10	Postcode
city	-	AN.. 50	City
country	+	Defa ult	Country (ISO 3166)
Delivery-Address			
shipping_firstna me	-	AN.. 50	First name (optional if company is used)
shipping_lastna me	-	AN.. 50	Surname
shipping_compa ny	-	AN.. 50	Company
shipping_street	-	AN.. 50	Street number and name

HTTP request from PAYONE to the merchant's server						
Parameter	**Req uire d**	**For mat**	**Comment**			
shipping_zip	-	AN.. 10	Postcode			
shipping_city	-	AN.. 50	City			
shipping_countr y	-	Defa ult	Country (ISO 3166)			
email	-	AN.. 50	Email address			
Parameter for the status message of a payment process						
txid	+	N..1 2	Payment process ID (PAYONE)			
reference	+	AN.. 20	Merchant reference number for the payment process			
sequencenumb er	+	N..2	Sequence number at the time of the event for this payment process (0..n)			
price	+	N..7, 2	Payment request (in largest currency unit! e.g. Euro)			
receivable	-	N..7, 2	Total payment request (in largest currency unit! e.g. Euro); not set for encashment reminder status information without paid amount			
balance	-	N..7, 2	Balance of transaction account (in largest currency unit! e.g. Euro) ; not set for encashment reminder status information without paid amount Negative amount: positive balance Positive amount: payment request			
failedcause	-	Defa ult	Reason for return debit note or incorrect collection (see chapter 5)			
errorcode				o	N..4	Errorcode in case of txaction="failed"

HTTP request from PAYONE to the merchant's server						
Parameter	**Req uire d**	**For mat**	**Comment**			
Additional parameter Contract for the status message of a payment process						
productid	+	N..7	ID for the offer			
accessid	+	N3.. 12	Access ID			
expiretime				-	N..1 2	Unix Timestamp an dem der Zugang abläuft
Additional parameter for payment type debit payment						
bankcountry	-	Defa ult	Account type/ country			
bankaccount	-	AN.. 26	Account number (masked)			
bankcode	-	AN.. 11	Sort code			
bankaccountholder				-	AN.. 35	Account holder
Additional parameter for payment type debit payment (only for authorization with appointed and only if "due_time" is not specified)						
iban	o	AN.. 35	International Bank Account Number (masked)			
bic	o	AN.. 11	Bank Identifier Code			
mandate_identi fication	o	AN.. 35	Used mandate_identification			
creditor_identifi er	o	AN.. 35	Merchant's creditor identifier			
clearing_date	o	N8	clearing date (format YYYYMMDD)			

HTTP request from PAYONE to the merchant's server						
Parameter	**Req uire d**	**For mat**	**Comment**			
clearing_amount				o	N..1 0	Payment request (in smallest currency unit! e.g. cent)
Additional parameter for payment type credit card						
cardpan	+	N..1 9	Card number			
cardtype	+	Defa ult	Card type V Visa M MasterCard A American Express D Diners J JCB O Maestro International C Discover B Carte Bleue P China Union Pay			
cardexpiredate	+	N4	Expiry date YYMM			
cardholder				-	AN.. 35	Name of cardholder
Additional parameter for payment type BillSAFE / Klarna						
clearing_bankac countholder	-	AN.. 35	Recipient bank account holder			
clearing_bankco untry	-	Defa ult	Recipient account type/ country (e.g. DE, AT, etc.)			
clearing_bankac count	-	AN.. 26	Recipient bank account			
clearing_bankco de	-	AN.. 11	Recipient sort code			
clearing_bankib an	-	AN.. 35	Recipient IBAN			
clearing_bankbi c	-	AN.. 11	Recipient BIC			
clearing_bankcit y	-	AN.. 50	Recipient city or bank			

HTTP request from PAYONE to the merchant's server						
Parameter	**Req uire d**	**For mat**	**Comment**			
clearing_bankna me	-	AN.. 50	Recipient bank name			
clearing_legalno te	-	AN.. 500	Note to claim assignment			
clearing_duedat e	-	N8	Due date of payment (format YYYYMMDD)			
clearing_referen ce	-	AN.. 50	Reference			
clearing_instructionnote				-	AN.. 200	Note to payment handling
Additional parameter Collect (txaction=remin der) for the status message of a payment process						
reminderlevel	+	Defa ult	Customer's reminder status 1…4: Dunning level 1-4 5: Encashment A: Dunning procedure ended S: Dunning procedure begins M: Dunning proposal list I: Encashment proposal list 0: Dunning procedure completed			
encashment_st atuscode	-	AN.. 20	Internal status code of the encashment agency, if provided by the encashment agency.			
encashment_statuslongtext				-	AN.. 255	Free text: if the encashment agency has reported a long text (detailed information) on the status
Parameter Invoicing (txaction=invoic e)						
txid	+	N..1 2	Payment process ID (PAYONE)			

HTTP request from PAYONE to the merchant's server			
Parameter	**Req uire d**	**For mat**	**Comment**
reference	+	AN.. 20	Merchant reference number for the payment process
sequencenumb er	+	N..2	Sequence number at the time of the event for this payment process (0..n)
invoiceid	+	AN.. 20	Merchant's invoice number
invoice_grossa mount	+	N..7, 2	Gross invoice amount
invoice_date	+	N8	Invoice date (format YYYYMMDD)
invoice_delivery date	-	N8	Delivery date (format YYYYMMDD)
invoice_deliveryenddate	-	N8	Delivery period end date (format YYYYMMDD)
Parameter Billing (txaction=vauth orization/vsettl ement)			
vaid	+	N..8	Billing account ID (module billing)
balance	+	N..7, 2	Balance of billing account (in largest currency unit! e.g. Euro) Negative amount: positive balance Positive amount: payment request
vreference	+	AN.. 20	Merchant's transaction reference number (This is the reference for the corresponding payment process for a vsettlement)
vxid	+	N..1 2	Billing account entry ID
Parameter Billing (txaction=vsettl ement)			
txid	o	N..1 2	Corresponding payment process ID

HTTP request from PAYONE to the merchant's server						
Parameter	**Req uire d**	**For mat**	**Comment**			
sequencenumb er	o	N..2	Sequence number of settled payment process ID			
settled_vxid[n]				o	N..1 2	Array of settled vxid's starting with n=0. Array will not be sent if more than 500 vxid's are settled.
Key						
N..x		Numeric value (x characters maximum)				
AN..x		Alphanumeric value (x characters maximum)				

Important note: Please note that new parameters may be added at any time without previous notice. Therefore, you should use the parameter name for the evaluation and not the sequence, which may be subject to change at any time!

Expected reply to the request:

As a reply to the request, the string "**TSOK**" is expected. Each request is repeated in a 1 to 6 hour cycle until it is answered with "TSOK". This procedure ensures that all requests will be processed by your system. Simply issue the "TSOK" in a script via the "print" command. Make sure that this character string is the first that is printed from this script, e.g. `print ("TSOK");`

Important:

- Status messages from PAYONE to merchant's server are always ISO-8859-1 encoded.

- No other characters may be issued from this character string, e.g. no HTML code.

- Do not return an error without gathering information about this error.

- Make sure the request is always answered with a TSOK within 48 hours.

- If a specific request shall not be processed, issue a TSOK anyway to prevent the request from interfering with the processing of other requests.

- Without the return of an TSOK you will not receive any further status reports for that payment process.

- The request must be answered with "TSOK" within 10 seconds. This time-out cannot be extended. The answer does only confirm receipt of the TransactionStatus, the evaluation can and should follow asynchronously to receiving the answer.

4.2.1 List of events (txaction)

With each payment process status change you receive a request. The last event is submitted to you via the parameter "txaction". The status of the request is provided via the balance of the payment process (parameter "balance") and the amount of the request (parameter "receivable").

appointed

Via "appointed" you are informed about the successful initiation of the payment process. This request is affected immediately after the first successful booking.

Important note:

- The new parameter "transaction_status" indicates whether the event "appointed" is pending or completed.
  -> see list of status (transaction_status)

capture

Via "capture" you are informed about the booking of a request or the collection of your reserved amount. The amount of the request (receivable) is increased in this case. If no settlement of balances occurs, the balance changes as well.

paid

Via "paid" you are informed that the booking has been processed by the credit institution or that the customer has paid the invoice in full. The balance for the request in this case is smaller than or equal to zero.

underpaid

Via "underpaid" you are informed about an underpayment. The balance for the request in this case is greater than zero.

cancelation

Via "cancelation" you are informed that a payment process has resulted in a return debit note. In the case of electronic direct debit processes (ELV) insufficient funds in the account may also be the cause. The balance for the request in this case is greater than zero.

refund

Via "refund" you are informed if an amount has been refunded. The amount of the request (receivable) is decreased in this case.

debit

Via "debit" you are informed about the booking of a request/credit for a request. The amount of the request (receivable) changes in this case. If no settlement of balances occurs, the balance changes as well.

transfer

Via "transfer" you are informed if an amount has been transferred. The amount of the open balance (balance) changes in this case.

reminder (Attention: This request must be activated by PAYONE.)

Via "reminder" you are informed about the current status of the dunning procedure.

vauthorization (Attention: This request must be activated by PAYONE.)

Via "vauthorization" you are informed about a booking affected into a billing account (module billing).

vsettlement (Attention: This request must be activated by PAYONE.)

Via "vsettlement" you are informed about a settlement effected on a particular billing account (module billing).

invoice (Attention: This request must be activated by PAYONE.)

Via "invoice" you are informed that an invoice or a credit voucher has been created.

failed (for future use – not implemented yet.)

Via "failed" you are informed that the booking has finally failed. No further actions are possible.

4.2.2 List of status (transaction_status)

Via "pending" you are informed that the payment transaction is (still) pending at the external payment processor. The following transaction status may be "pending" (again), "completed" (external payment processor completed the actual transaction successfully).

With each payment process status change you receive a request. The last event is submitted to you via the parameter "txaction".

The parameter "transaction_status" is currently introduced with event-txaction "appointed" only. Other event-txaction with parameter "transaction_status" may follow (e.g. "paid", "debit", …).

Please note:

- **The parameter "transaction_status" is optional and not available for all payment transactions ("txaction") and all payment types (as not all payments and processors do support "pending" / "completed").**
- **It may happen that you will receive two times the same txaction (e.g. "appointed"). First with "pending" and then with "completed".**

pending

The event indicated by "txaction" is pending and may change later. i.e. an event "appointed/pending" (txaction/transaction_status) indicates that the payment is pending and in process at the 2nd payment processor.

Another event may follow to inform change of status by txaction e.g. "appointed/completed", "failed/completed".

Also another "appointed/pending" may follow to indicate that transaction is still pending.

completed

Indicates that the event itself has reached final status.

However a new "txaction" (e.g. "paid", "cancelation", …) may follow to inform of change of status.

The new "txaction" can then be "paid/pending", "paid/completed", … or "failed/completed".

4.2.3 Explanation of price, balance, receivable

Field	Description
price	Value of the initial claim
balance	The outstanding balance of this transaction: • negative: Customer has a claim against merchant, e.g. merchant received money without effort • positive: Merchant has a claim against the customer
receivable	Account balance for the transaction. • With a "preauthorization" the value "receivable" is not set as the merchant did not provide the service yet (e.g. delivering goods). • With type of payment "cash In advance" the value "receivable" is not set as the merchant will only provide its service when money has arrived.

4.2.4 Sample: authorization, CC

Merchant's request	Seq-No	Time	TX-Action/transaction_state[17]	Seq-No	price	balance	receiv-able	Comment
			HTTP request from PAYONE to the merchant's server					
Request authorization CC amount=15061	0	T=0	appointed /completed	0	150.61	150.61	150.61	
	0	+4 min	paid	0	150.61	0	150.61	

---

[17] Internal reference: 148021809

4.2.5 Sample: preauthorization/capture, CC

Merchant's request			HTTP request from PAYONE to the merchant's server					Comment
	Seq-No	Time	TX-Action/transaction_state [18]	Seq-No	price	balance	receiv-able	
Request preauthorization CC amount=2950	0	T=0	appointed /complet ed	0	29.50	0.00	0.00	
Request capture	1	+2 hours	paid	1	29.50	0	29.50	

[18] Internal reference:

4.2.6 Sample: authorization, ELV with cancelation

Merchant has configured:

- Due time ELV = 7 days
- Fee 1. reminder = 0,00 Euro after 7 days
- Fee 2. reminder = 1,00 Euro after 7 days
- Fee 3. reminder = 2,40 Euro after 7 days
- Encashment transfer = 5,00 Euro after 7 days
- TxStatus without reminder-information

Merchant's request			HTTP request from PAYONE to the merchant's server					Comment
	Seq-No	Time	TX-Action/transaction_state [19]	Seq-No	price	balance	receiv-able	
Request authorization ELV amount=4612	0	T=0	appointed /completed	0	46.12	46.12	46.12	Merchant initiates payment via SEPA direct debit
		+15 min	paid	0	46.12	0	46.12	PAYONE platform has processed direct debit
		+7 days	cancelation	0	46.12	54.72	54.72	PAYONE platform has detected a return debit note initiated by end customer and added bank charges of 8,60 EUR and 0 Euro dunning fee
		+14 days (7+7)	debit	1	46.12	55.72	55.72	PAYONE platform processed dunning note and added 1,00 Euro dunning fee
		+21 days	debit	2	46.12	57.72	57.72	PAYONE platform processed dunning note and added 2,00 Euro dunning fee
		+28 days	debit	3	46.12	62.72	62.72	PAYONE platform processed dunning note and added 5,00 Euro dunning fee

---

[19] Internal reference: 146039593

4.2.7 Sample: preauthorization/capture, REC with credit note

Merchant has configured:

- Due time Invoice = 14 days
- Fee 1. reminder = 0,00 Euro after 3 days
- Fee 2. reminder = 2,00 Euro after 10 days
- Fee 3. reminder = 4,00 Euro after 10 days
- TxStatus without reminder-information

Merchant's request			HTTP request from PAYONE to the merchant's server					Comment
	Seq-No	Time	TX-Action/transaction_state [20]	Seq-No	price	balance	receiv-able	
Request preauthorization REC amount=11500	0	T=0	appointed /completed	0	115.00	0.00	0.00	Merchant initiates payment via payment type invoice
Request capture	1	+1 day	capture	1	115.00	115	115	Merchant has delivered ordered items
		+27 days (14+3+10)	debit	2	115.00	117	117	PAYONE platform generates reminder document and added 2 Euro dunning fee
		+10 days	debit	3	115.00	121	121	PAYONE platform generates reminder document and added 4 Euro dunning fee
PMI: credit note by 15,00 Euro	4	+13 days	debit	4	115.00	106	106	PAYONE platform processed credit note initiated via PMI

---

[20] Internal reference: 136229273

4.2.8 Sample: preauthorization, WLT (with "pending")

Merchant's request			HTTP request from PAYONE to the merchant's server					Comment
	Seq-No	Time	TX-Action/tran saction_stat e[21]	Seq-No	price	balance	receiv-able	
Request authorization WLT amount=1561	0	T=0	appointed /pending	0	15.61	0.00	0.00	
	0	+6 seconds	appointed /complet ed	0	15.61	0	0.00	
	0	+6 min	paid	0	15.61	-15.61	0	

---

[21] Internal reference: 126353063

5 Codes

5.1 Reasons for return debit notes

Parametername: failedcause

Code	Signification
soc	Insufficient funds
cka	Account expired
uan	Account no. / name not identical, incorrect or savings account
ndd	No direct debit
rcl	Recall[22]
obj	Objection[23]
ret	Return[24]
nelv	Debit cannot be collected
cb	Credit card chargeback
ncc	Credit card cannot be collected

[22] Recall: The recipient of the payment recalls the direct debit order.

[23] Objection: The payer objects to the direct debit.

[24] Return: The executing bank returns the order (due to insufficient funds, account closure, decease of the account holder, ...).

5.2   Bank groups

Parametername: bankgrouptype

Status	Code	Signification
	Banks for EPS (Austria)	
existing	ARZ_OAB	Apothekerbank
existing	ARZ_BAF	Ärztebank
existing	BA_AUS	Bank Austria
existing	ARZ_BCS	Bankhaus Carl Spängler & Co.AG
existing	EPS_SCHEL	Bankhaus Schelhammer & Schattera AG
existing	BAWAG_PSK	BAWAG P.S.K. AG
existing	BAWAG_ESY	Easybank AG
existing	SPARDAT_EBS	Erste Bank und Sparkassen
existing	ARZ_HAA	Hypo Alpe-Adria-Bank International AG
existing	ARZ_VLH	Hypo Landesbank Vorarlberg
existing	HRAC_OOS	HYPO Oberösterreich,Salzburg,Steiermark
existing	ARZ_HTB	Hypo Tirol Bank AG
existing	ARZ_IMB	Immo-Bank
existing	EPS_OBAG	Oberbank AG
existing	RAC_RAC	Raiffeisen Bankengruppe Österreich
existing	EPS_SCHOELLER	Schoellerbank AG
existing	ARZ_OVB	Volksbank Gruppe
existing	EPS_VRBB	VR-Bank Braunau
new 04/2017	EPS_AAB	Austrian Anadi Bank AG
new 04/2017	EPS_BKS	BKS Bank AG
new 04/2017	EPS_BKB	Brüll Kallmus Bank AG
new 04/2017	EPS_VLB	BTV VIER LÄNDER BANK
new 04/2017	EPS_CBGG	Capital Bank Grawe Gruppe AG
new 04/2017	EPS_DB	Dolomitenbank
new 04/2017	EPS_NOEGB	HYPO NOE Gruppe Bank AG
new 04/2017	EPS_NOELB	HYPO NOE Landesbank AG
new 04/2017	EPS_HBL	HYPO-BANK BURGENLAND Aktiengesellschaft
new 04/2017	EPS_MFB	Marchfelder Bank
new 04/2017	EPS_SPDBW	Sparda Bank Wien
new 04/2017	EPS_SPDBA	SPARDA-BANK AUSTRIA
new 04/2017	EPS_VKB	Volkskreditbank AG
	Banks for iDEAL (The Netherlands)	
existing	ABN_AMRO_BANK	ABN Amro
existing	BUNQ_BANK	Bunq

existing	RABOBANK	Rabobank
existing	ASN_BANK	ASN Bank
existing	SNS_BANK	SNS Bank
existing	TRIODOS_BANK	Triodos Bank
existing	SNS_REGIO_BANK	~~SNS~~ Regio Bank
existing	ING_BANK	ING Bank
existing	KNAB_BANK	Knab ~~Bank~~
existing	VAN_LANSCHOT_BANKIERS	van Lanschot ~~Bank~~
deprecated	~~FRIESLAND_BANK~~	~~Friesland Bank~~

5.3   Parameters for AVS returns

Parametername: protect_result_avs

Under "street number" any numeric parts of the street name and street number, as well as the postal code of the invoice address are validated, e.g. "Münchhausenstraße 22, 3rd Floor", "D-12345" -> "223" and "12345" are validated.

Code	Signification
A	Street number is OK, postal code is not OK
F	Street number and postal code are OK
N	Neither street number nor postal code are OK
U	Request is not supported
Z	Street number is not OK, but postal code is OK

5.4   BankAccountCheck (Available countries)

Continious updates of check digit algorithms of Deutsche Bundesbank incl. check of Bankcode and of Österreichischen Nationalbank:

	bank-country	bank-account	bank-code	bank-branch-code	bank-check-digit
Germany (DE)	x	x	x	-	-
Austria (AT)	x	x	x	-	-
Swiss (CH)	x	x	x	-	-
Key: x =mandatory, o = optional, - = not used					

In these European countries the bank account will be syntax-checked and also validated with a check digit algorithm:

	bank-country	bank-account	bank-code	bank-branch-code	bank-check-digit
Niederlande (NL)	x	x	-	-	-
Schweden (SE)	x	x	o	-	-
Belgien (BE)	x	x	o	-	o
Frankreich (FR)	x	x	x	x	x
Norwegen (NO)	x	x	o	-	-
Polen (PL)	x	x	o	-	o
Key: x =mandatory, o = optional, - = not used					

In these countries the bank account will be syntax-checked:

	bank-country	bank-account	bank-code	bank-branch-code	bank-check-digit
Denmark (DK)	x	x	o	-	-
Finland (FI)	x	x	o	-	-
Italy (IT)	x	x	o	o	o
Spain (ES)	x	x	o	o	o
United Kingdom (GB)	x	x	x	-	-
Key: x =mandatory, o = optional, - = not used					

5.5 Parameters for address check person returns

5.5.1 AddressCheck PAYONE Services - personstatus

Parametername: personstatus

Code	Signification
NONE	no verification of personal data carried out
PAB	first name & surname unknown
PHB	surname known
PKI	ambiguity in name and address
PNP	address cannot be checked, e.g. fake name used
PNZ	cannot be delivered (any longer)
PPB	the person can be confirmed for this address.
PPF	postal address details incorrect
PPV	person deceased
PUG	postal address details correct but building unknown
PUZ	person has moved, address not corrected
UKN	unknown return values are mapped to UKN

5.5.2 AddressCheck Boniversum Services - personstatus

Parametername: personstatus

mapped to Code	Boniversum internal status	Signification	Signification (DE)
PPF	00	postal address details incorrect	Nicht identifiziert.
PPB	02	the person can be confirmed for this address.	Person identifiziert.
PHB	03	surname known	Haushalt identifiziert.
PAB	05	first name & surname unknown	Gebäude identifiziert.
PPV	06	person deceased	Person verstorben.

5.6 Parameters for Consumerscore returns

5.6.1 Consumerscore arvato Infoscore - secscore

Parametername: secscore

I. "Weak" negative attributes (Data from collection - pre court):		Weiche Negativmerkmale
IA	Initiation of collection process	Inkasso-Mahnverfahren eingeleitet
AM	Continuation of extrajudicial collection procedures after partial payments or instalments.	Fortlauf des außergerichtlichen Inkasso-Mahnverfahrens nach Teil- bzw. bei Ratenzahlung
IE	Dismissal of extrajudicial collection proceedings du to futility	Einstellung des außergerichtlichen Inkasso-Mahnverfahrens wegen Aussichtslosigkeit

II. "Middle" negative attributes (Data from collection - court):		Mittlere Negativmerkmale
MB	Payment ordered by the court	Antrag auf Mahnbescheid
VB	Enforcement order from the court	Antrag auf Vollstreckungsbescheid
TR	Payment installment plan resulting from court decision	Ratenzahler nach Forderungs-Titulierung
ZWA	Levy of execution to liquidate personal assets (non real-estate)	Zwangsvollstreckung in das bewegliche Vermögen
ZWI	Levy of execution to liquidate personal assets (real-estate)	Zwangsvollstreckung in das unbewegliche Vermögen
FRP	Futile collections attempt	fruchtlose Pfändung
LP	Garnishment of wages	Lohn- oder sonstige Forderungspfändung aufgrund eines gerichtlichen Pfändungs- und Überweisungsbeschlusses
UF	Non collectable legally binding payment claim	uneinbringliche, titulierte Forderung
UBV	Relocation leaving no forwarding address	unbekannt verzogen (unter Hinterlassung von Verbindlichkeiten aus Geld- oder Warenkrediten)
SU		Suchauftrag zu - unter Hinterlassung von Verbindlichkeiten - unbekannt verzogenen Personen (Adresse über Einwohneramt nicht ermittelbar)

III. "hard" negative attributes (Data from public lists of debtors / / Bundesanzeiger):		Harte Negativmerkmale
HB	Warrant to provide Affidavit	Haftbefehl zur Erzwingung der eidesstattlichen Versicherung
HV	Execution of warrant to provide Affidavit	Vollstreckung des Haftbefehls zur Erzwingung der EV
EV	Affidavit or oath of disclosure	eidesstattliche Versicherung ("Offenbarungseid")
EEV	Amendment to Affidavit	Ergänzung der eidesstattlichen Versicherung
WEV	Repeated affidavit or oath of disclosure	wiederholte eidesstattliche Versicherung (§ 903 ZPO)

SVV		Verweigerung der Vermögensauskunft gem. §882c Abs. 1 Nr. 1 ZPO
SAV		erkennbare Aussichtslosigkeit der Vollstreckung gem. §882c Abs. 1 Nr. 2 ZPO
SNZ		Nichtzahler gem. §882c Abs. 1 Nr. 3 ZPO
IVE	Extrajudicial settlement attempt in the course of insolvency proceedings	außergerichtlicher Einigungsversuch im Rahmen des Insolvenzverfahrens
III. "hard" negative attributes (Data from public lists of debtors // Bundesanzeiger):		**Harte Negativmerkmale**
ISP	Initiation of debt settlement plan / initiations of insolvency proceedings	Schuldenbereinigungsplan-Verfahren eingeleitet / Antrag auf Eröffnung des Insolvenzverfahrens
IVS	Court ordered preliminary safeguard provisions in the course of insolvency	Anordnung von vorläufigen Sicherungsmaßnahmen im Insolvenzverfahren
IVA	Court ordered revocation of preliminary safeguard provisions in the course of insolvency	Aufhebung von vorläufigen Sicherungsmaßnahmen im Insolvenzverfahren
IBE	Institution of insolvency proceedings / court order	Eröffnung des Insolvenzverfahrens / Beschluss
IBA	Dismissal of insolvency proceedings du to insufficient assets	Abweisung bzw. Einstellung des Insolvenzverfahrens mangels Masse (§ 26 Abs. 2 InsO)
IWP	Reversal of insolvency proceedings / start of period of good conduct	Aufhebung des Insolvenzverfahrens
IRB	Discharge grant of outstanding debts	Erteilung der Restschuldbefreiung
IRV	Discharge denial of outstanding debts	Versagung der Restschuldbefreiung
KON	Dismissal of bankruptcy petitions du to insufficient assets (Prior to 1999)	Abweisung des Antrags auf Eröffnung des Konkurses mangels Masse (§ 107 KO)
KER	Institution of bankruptcy proceedings (Prior to 1999)	Eröffnung des Konkursverfahrens (§ 108 KO)
KEM	Suspension of bankruptcy proceedings due to insufficient assets (Prior to 1999)	Einstellung des Konkursverfahrens mangels Masse (§§ 202, 204 KO)
KAS	Revocation of bankruptcy proceedings (Prior to 1999)	Aufhebung des Konkursverfahrens (nach Schlusstermin) (§ 163 KO)
VGE	Institution of reorganizations proceedings	Eröffnung des Vergleichsverfahrens (§ 11 ff VerglO)
VGA	Institution of reorganizations proceedings	Abweisung des Antrages auf Eröffnung des Vergleichsverfahrens
VEM	Suspension of reorganizations proceedings	Einstellung des Vergleichsverfahrens nach Rücknahme des Vergleichsvorschlages (§ 99 ff VerglO)
VAS	Revocation of reorganizations proceedings	VAS Aufhebung des Vergleichsverfahrens (§ 90 ff VerglO)
GVA	Dismissal of petition to initiate comprehensive execution for insufficient assets	GVA Abweisung des Antrages auf Eröffnung der Gesamtvollstreckung mangels Masse (§ 4 Abs. 2 GesO)
GVE	Commercial Insolvency Arrangement	Eröffnung bzw. Anordnung der Gesamtvollstreckung (§ 5 GesO)
GEM	Commercial Insolvency	Einstellung des Gesamtvollstreckungsverfahrens
GAS	Aborted Commercial Insolvency	Aufhebung des Gesamtvollstreckungsverfahrens

Other		Sonstige Negativmerkmale
+++	Deceased	verstorben
AE		Adressermittlung (Person unter der Adresse nicht zustellbar)
HI		Hinweise zur Person oder Firma
HA		Risikohinweise zur Adresse (JVA, Obdachlosenunterkunft, o. ä.); TKZ 5

5.6.2 Decision matrix Arvato Infoscore Boni-Score / Informa-Score

Parametername: scorevalue

Hinweise zur Interpretation von Scorewerten
für Bezieher von Boniscore und/oder Scores der Informa-Score-Familie

In der nachstehenden Grafik finden Sie die gültigen Scorewerte des Boni-Score (d.h. Für Personen mit Negativmerkmalen) und für die Informa-Consumer-Scores (für Personen ohne Negativmerkmale) mit den dazugehörigen Ausfallquoten.
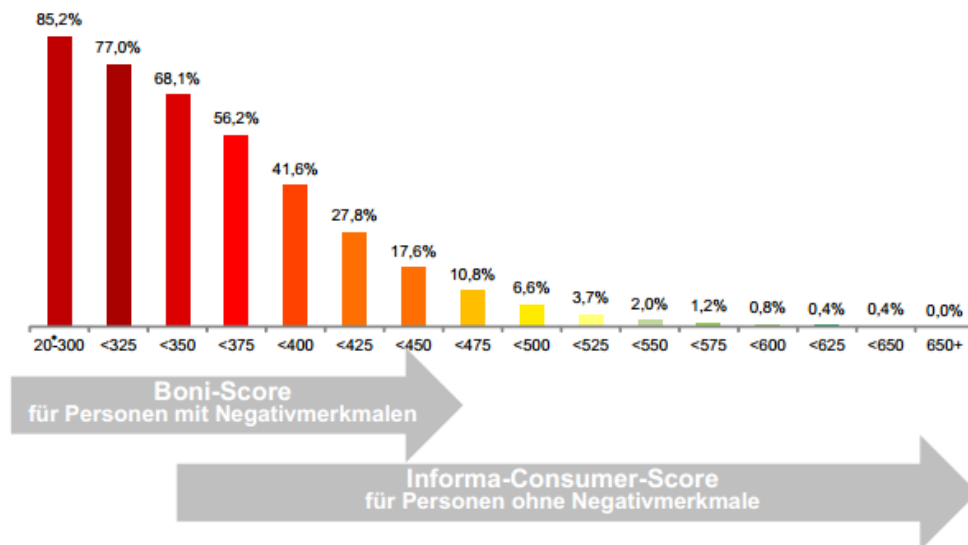
Hinweise

- Die beiden Scores sind bezüglich ihrer Ausfallquoten adjustiert.
- Die dargestellte Schlechtenquote ist definiert als Wahrscheinlichkeit für das Auftreten einer Zahlungsstörung innerhalb eines Jahres. **Sie kann und soll nur als Anhaltspunkt dienen, da sie je nach Branche, Kundenklientel etc. variiert.**

In Ausnahmefällen kann kein gültiger Scorewert gemäß der obigen Tabelle berechnet werden. In diesem Fall ist der Ausgabewert wie folgt zu interpretieren:

Ausgabewert	Bedeutung	Hinweis
0	Kein Score berechenbar	
1**	Postalisch falsch	Die Ausgabewerte 0-6 stellen **keine Aussage** zum Zahlungsausfallrisiko dar, sondern besagen, dass - aus unterschiedlichen Gründen - kein Score berechnet werden konnte.
2**	Hinweisadresse	
3**	Person verstorben	
4**	Teilnehmeridentcode falsch	
5**	Kein berechtigtes Interesse	
6**	Business-Sperre	

[25]

5.6.3 Decision matrix Boniversum VERITA score

Decision matrix page 1

Entscheidungsmatrix VERITA Premium Ident S (Mittleres Risiko)								Zahlungsart	Kreditlimit Unsichere ZA
Adressvalidierung		**Identifizierung**		**Sonderadressen**				**Zahlungsart**	**Kreditlimit Unsichere ZA**
Return-Code	Erläuterung	Return-Code	Erläuterung	Adressflag	Bedeutung			Empfehlung	Empfehlung
Seite 11 von 21 der XML-Dokumentation, § 3.3		Seite 13 von 21 der XML-Dokumentation, § 3.5.1		Seite 17 von 21 der XML-Dokumentation, § 3.7					
03	Adresse NICHT in Ordnung							Vorkasse / Aussteuern bzw. manuelle Prüfung	
01 oder 02	Adresse in Ordnung	06	Person verstorben					Vorkasse	
01 oder 02	Adresse in Ordnung	00 bis 05	Person nicht verstorben	S1	Sonderadresse höchstes Risiko			Sicher	
01 oder 02	Adresse in Ordnung	00 bis 05	Person nicht verstorben	S2	Sonderadresse sehr stark erhöhtes Risiko			Sicher	
01 oder 02	Adresse in Ordnung	00 bis 05	Person nicht verstorben	S3	Sonderadresse stark erhöhtes Risiko			Sicher	
01 oder 02	Adresse in Ordnung	00 bis 05	Person nicht verstorben	S4	Sonderadresse deutlich erhöhtes Risiko			Sicher	
01 oder 02	Adresse in Ordnung	00 bis 05	Person nicht verstorben	S5	Sonderadresse spürbar erhöhtes Risiko			Alle	25%
01 oder 02	Adresse in Ordnung	00 bis 05	Person nicht verstorben	S6	Sonderadresse mäßig erhöhtes Risiko			Alle	25%
01 oder 02	Adresse in Ordnung	00 bis 05	Person nicht verstorben	S7	Sonderadresse leicht erhöhtes Risiko			Alle	50%
01 oder 02	Adresse in Ordnung	00 bis 05	Person nicht verstorben	S8	Sonderadresse schwach erhöhtes Risiko			Alle	50%
01 oder 02	Adresse in Ordnung	00 bis 05	Person nicht verstorben	Z	keine Sonderadresse			siehe nächste Seite	

[26]

Decision matrix page 2

Entscheidungsmatrix **VERITA** Premium Ident S (Mittleres Risiko)					
Identifizierung		**VERITA Score**		**Zahlungsart**	**Kreditlimit Unsichere ZA**
Return-Code	Erläuterung	Scoreklasse	Scorebandbreite	Empfehlung	Empfehlung
Seite 13 von 21 der XML-Dokumentation, § 3.5.1		Seite 15 von 21 der XML-Dokumentation, § 3.6.3			
02	Person identifiziert	1 bis 4	Scorewert von 1 bis 29	Alle	100%
02	Person identifiziert	5 bis 6	Scorewert von 30 bis 56		75%
02	Person identifiziert	7 bis 8	Scorewert von 57 bis 123		50%
02	Person identifiziert	9	Scorewert von 124 bis 1000		25%
02	Person identifiziert	Keine Scorerückgabe bzw. Scorewert = 0			50%
02	Person identifiziert	-1 bis -5	Creditreform-Inkasso oder Negative Pooldaten	Sicher	
02	Person identifiziert	-6	Härtestes Merkmal Gericht	Vorkasse	
03	Haushalt identifiziert	1 bis 2	Scorewert von 1 bis 13	Alle	75%
03	Haushalt identifiziert	3 bis 4	Scorewert von 14 bis 29		50%
03	Haushalt identifiziert	5 bis 6	Scorewert von 30 bis 56		25%
03	Haushalt identifiziert	7 bis 9	Scorewert von 57 bis 1000	Sicher	
03	Haushalt identifiziert	Keine Scorerückgabe bzw. Scorewert = 0			
05	Gebäude identifiziert	1 bis 2	Scorewert von 1 bis 13	Alle	75%
05	Gebäude identifiziert	3 bis 4	Scorewert von 14 bis 29		50%
05	Gebäude identifiziert	5	Scorewert von 30 bis 40		25%
05	Gebäude identifiziert	6 bis 9	Scorewert von 41 bis 1000	Sicher	
05	Gebäude identifiziert	Keine Scorerückgabe bzw. Scorewert = 0			
00	Nicht identifiziert	1 bis 3	Scorewert von 1 bis 20	Alle	50%
00	Nicht identifiziert	4 bis 5	Scorewert von 21 bis 40		25%
00	Nicht identifiziert	6 bis 9	Scorewert von 41 bis 1000	Sicher	
00	Nicht identifiziert	Keine Scorerückgabe bzw. Scorewert = 0			

5.6.4 Consumerscore Boniversum VERITA score - scorevalue

Parametername: scorevalue

VERITA Score	Klasse	Scorewerte	Schlechtrate [*]
Wenn KEINE personenbezogenen Negativmerkmale vorliegen	1	1 bis 8	0,7
	2	9 bis 13	1,2
	3	14 bis 20	1,7
	4	21 bis 29	2,4
	5	30 bis 40	3,6
	6	41 bis 56	4,9
	7	57 bis 80	6,9
	8	81 bis 123	10,5
	9	124 bis 1000	17,8
	0	Es sind zu wenige Daten für eine Berechnung vorhanden.	

	Klasse	Filterwerte	Beschreibung
Wenn personenbezogene Negativmerkmale vorliegen	-1	5000	geschlossene Inkassofälle (älter als 2 Jahre), negative Kontoschließungen (älter als 2 Jahre)
	-2	5200	geschlossene titulierte Inkassofälle (älter als 2 Jahre), Ausbuchungen (älter als 2 Jahre), gerichtliches Mahnverfahren (älter als 2 Jahre), erteilte Restschuldbefreiung (1 Jahr oder jünger)
	-3	5400	geschlossene Inkassofälle (2 Jahre alt oder jünger), aktueller Zahlungsrückstand 30/60/90 Tage
	-4	5600	offene kaufmännische Inkassofälle (Mahnsachen), geschlossene titulierte Inkassofälle (2 Jahre alt oder jünger), negative Kontoschließungen (2 Jahre alt oder jünger), Kontofälligstellung (2 Jahre alt oder jünger)
	-5	5800	offene titulierte Inkassofälle (Überwachungssachen), Ausbuchung (2 Jahre alt oder jünger), gerichtliches Mahnverfahren (2 Jahre alt oder jünger)
	-6	6000	Harte Gerichtsmerkmale (Eidesstattliche Versicherung, Nichtabgabe der Vermögensauskunft, Gläubigerbefriedigung ausgeschlossen oder nach einem Monat nicht nachgewiesen, Haftanordnung, Insolvenz)

(*) Statistische Wahrscheinlichkeit, dass die bewertete Person in den folgenden 12 Monaten Zahlungsschwierigkeiten aufweist. [27]

---

[27] ©2016 Boniversum: Available in German language only and only an indication. Your business settings may vary.

VERITA Score	Beschreibung	Ampel-wert	Ampel-beschreibung
1 bis 56	Annehmen. Es liegen keine personenbezogenen Negativmerkmale vor.	03	GRÜN
57 bis 1000	Vorsicht!	02	GELB
5000 bis 6000	Es liegen Negativdaten zu der Person vor.	01	ROT
0	Es sind zu wenig Daten für eine Berechnung vorhanden.	00	Nicht angezeigt

5.6.5 Consumerscore Boniversum VERITA score - addressstatus

Parametername: addressstatus

mapped to Code	Boniversum internal status	Signification	Signification (DE)
Special addresses -> recommended to check addresses manually			
AS1	S1	Prisons, institutions for psychosocial care / addiction, etc., homeless shelters, sheltered facilities for criminals, other homes without recreation and holiday homes, youth hostels, educational centers, youth centers and open house houses	Justizvollzugsanstalten, Stationäre Einrichtungen zur psychosozialen Betreuung/ Suchtbekämpfung u.Ä., Obdachlosenheime, betreute Übergangseinrichtungen für Straftäter, Sonstige Heime (ohne Erholungs- und Ferienheime), Jugendwohnheime, Erziehungsheime, Jugendzentren und Häuser der offenen Tür
AS2	S2	Hospitals, retirement and rehabilitation clinics, nursing homes, nursing homes, disabled workshops, retirement homes; Old people's homes, disabled people's homes, old people's homes	Krankenhäuser, Vorsorge- und Rehabilitationskliniken, Pflegeheime, Altenpflegeheime, Behindertenwerkstätten, Altenheime; Alten- und Behindertenwohnheime, Altenwohnheime
AS3	S3	Games, betting and lotteries, playing halls & operation of game machines, retailing of motor fuels (petrol stations), package shops, packing stations	Spiel-, Wett- und Lotteriewesen, Spielhallen & Betrieb von Spielautomaten, Einzelhandel mit Motorenkraftstoffen (Tankstellen), Paketshops, Packstationen
AS4	S4	Gastronomy, cafés, serving drinks	Gastronomie, Cafés, Ausschank von Getränken
AS5	S5	Other social work (without housing), placement and transfer of workers	Sonstiges Sozialwesen (ohne Heime), Vermittlung und Überlassung von Arbeitskräften
AS6	S6	Operation of sports facilities, fitness centers	Betrieb von Sportanlagen, Fitnesszentren
AS7	S7	Accommodation, holiday houses and holiday apartments, campsites, amusement and theme parks	Beherbergung, Ferienhäuser und Ferienwohnungen, Campingplätze, Vergnügungs- und Themenparks
AS8	S8	Education, boarding school	Erziehung und Unterricht, Internate

5.7   Error messages

No.	Signification	Activity
1	Card issuer temporarily not available	Please try again later.
2	Authorization declined	Decline purchase
4	Card not approved	Decline purchase
5	Authorization declined	Decline purchase
7	Required CVC code not specified or not valid	Specify/change CVC
12	Transaction invalid	Decline purchase
13	Limit exceeded	Decline purchase
14	Invalid card	Decline purchase
21	Activity not processed	Time for Capture after PreAuthorization has been exceeded (among other things)
30	Format Error in request message (e.g. CVC missing).	
31	Invalid card type	Change card type
33	Expiry date invalid, incorrect or in the past	Correct expiry date
34	Manipulation suspected	Decline purchase
43	Card stolen	Decline purchase
51	Limit exceeded or account balance insufficient	Decline purchase
56	Card unknown	Decline purchase
61	Card blocked	Decline purchase
62	Card cancelled	Decline purchase
65	Card has been used too often	Decline purchase
80	Amount no longer available	Reservation period has elapsed. Please repeat preauthorization
91	Card issuer temporarily not available	Temporary problem
101	The authorisation has expired.	Restart entire process.
102	Authorisation denied.	
103	Transaction could not be completed successfully.	Customer is to select an alternative method of payment.
104	Customer could not pay.	Transaction rejected.
105	The customer has to be redirected to Wallet Provider in order to select a new payment method.	Redirect customer to Wallet Provider
106	The order has expired.	Close order
107	Risk assessment has denied this transaction.	
108	The transaction has already been completed.	
109	Transaction (credit) denied	The refund was rejected.
110	Problem with mandate of customer at external service provider.	See external Message and contact external service provider or customer.
111	The customer has not yet completed the wallet transaction at the external payment provider.	
201	The transaction was denied by iDEAL.	
202	The transaction was denied by PostFinance.	
250	SEPA mandate does not exist	Please add SEPA mandate
251	SEPA mandate not confirmed	Please confirm SEPA mandate

No.	Signification	Activity
252	SEPA mandate parameter not valid or missing	Incorrect SEPA mandate parameter
253	SEPA configuration error	Contact PAYONE please.
254	Request not supported for this SEPA configuration	
255	SEPA mandate place of confirmation missing	Please add place of SEPA mandate confirmation
301	Amount for this payment method too low	Change amount
303	Capture amount does not match financing amount	Change amount
304	Parameter Token is expired	
305	Customer has not yet completed the transaction	
306	Transaction already has customer payments	
307	Financing Provider does not secure this transaction	Change payment type
308	Parameter {settleaccount{ does not fit to payment type financing	Change value for Parameter {settleaccount}
350	Payment guarantee has expired	
351	Payment guarantee has been rejected	
701	Payment denied after BIN check	Decline purchase
702	Payment was denied because of the BIN country	Decline purchase
703	Payment denied after IP check	Decline purchase
704	Payment was denied because of the IP country	Decline purchase
710	Payment denied after POS check	Decline purchase
721	Payment denied after Velocity IP check	Decline purchase
722	Payment denied after Velocity card number check	Decline purchase
723	Payment denied after Velocity account number check	Decline purchase
724	Payment denied after Velocity e-mail check	Decline purchase
731	Payment rejected by the Blacklist-IP-Check	
732	Payment rejected by the Blacklist-Cardpan-Check	
733	Payment rejected by the Blacklist-Bankaccount-Check	
734	Payment rejected by the Blacklist-Email-Check	
872	Pseudo card PAN not found	
876	AVS mismatch: The address provided does not match billing address of card holder.	Decline purchase
877	Invalid card number (Luhn check for this PAN incorrect)	Correct card number
878	Invalid card number (syntax check for this PAN incorrect)	Correct card number
879	CVC2 code incorrect length or incorrect syntax	Correct CVC2
880	Card type does not correspond with card number	Select different card type
881	Bank details cannot be used for online banking.	
882	Bank type not supported	Select different bank type
883	The account number is invalid for this sort code	Decline purchase
884	The sort code is invalid	Decline purchase
885	Bank is not supported by giropay	
886	Online transfer type not supported	Select different online transfer type
887	Invalid BIC	
888	Invalid IBAN	
889	Country of the BIC not supported.	
890	Payment declined due to current risk setting	Decline purchase
891	Debtor limit exceeded, payment refused	Decline purchase

No.	Signification	Activity
892	Country of the account not supported.	
894	account data incomplete: account holder name missing or invalid	Correct the account holder name
895	account data incomplete: street missing or invalid	Correct the street name
896	account data incomplete: zip missing or invalid	Correct the postal code
897	account data incomplete: city missing or invalid	Correct the city name
900	Internal error	Contact PAYONE please.
901	Configuration error	Contact PAYONE please.
902	Unknown error with external service provider.	Contact PAYONE please.
903	No connection to external service provider possible (timeout, breakdown)	Contact PAYONE please.
904	Temporary error	Please try again later.
905	Merchant authentication / authorization at service provider failed	Correct authentication data at service provider, please.
906	Refund amount too high	Amount exceeds the original amount or remaining sum too low
907	Sequence no incorrect	Select different sequence number
908	Capture amount too high	Amount exceeds the original amount or remaining sum too low
909	Database connection failed	Contact PAYONE please.
910	Currency does not correspond with the referenced payment process	Change currency
911	Reference number already exists	Change reference number
912	Reference number does not correspond with the referenced payment process	Change reference number
913	Referenced payment process (txid) not found	Enter different txid
915	Refund not possible	The connector settings do not allow a refund
916	Amount error	Amount incorrect sign or zero
917	Refund limit exceeded	Please contact PAYONE
918	Upper or lower limit amount exceeded	Please contact PAYONE
919	Limit revenue exceeded	Please contact PAYONE
920	MaxPayoutAmount exceeded	Reduce amount
921	Account balance insufficient	
922	Unknown account	
923	Payment type not available for this currency or card type	Set correct currency, type of payment, type of card, type of account or portalid
924	Test mode not available	Please contact PAYONE
925	Live mode not available	Please contact PAYONE
926	Incorrect mode for the referenced payment process	Change mode (live/test)
927	No refund connector.	Please contact PAYONE
928	Article with itemtype prepaid not allowed in this request	
929	If the request contains articles with itemtype prepaid, the amount must be 0	
930	Connection does not support value of {recurrence}	change {recurrence}
931	Request does not support type of payment	Activity not possible for this type of payment

No.	Signification	Activity
932	Collection not possible for this type of payment	Activity not possible for this type of payment
933	Multiple partial capture not supported for this method of payment	Change method of payment
934	Multiple partial capture not enabled	Please contact PAYONE
935	Amount for multiple partial capture cannot be 0	
936	Authorisation without incoming payment not possible for this type of payment	
937	settleaccount does not fit to capturemode	Change settleaccount or capturemode
938	Voucher is not supported by this payment type	Remove voucher from aticle list
939	The amount of a prepaid article must be negative.	
940	Currency or cardtype not available at external service provider.	Contact external service provider please.
941	Feature referenced Transaction is not enabled at service provider.	Contact service provider and request missing feature, please.
944	Amount too small	
945	Amount too high	
950	Desired status change not possible for this payment process	Please try again later.
951	The maximum number of actions for this txid has been reached	
952	The Transaction was already completed.	
960	Target payment process (txid) not found	Transfer not possible
961	No incoming transfer found for this payment process	
962	Transferable amount not available	Amount too big or too small.
970	Operation was cancelled by the user.	
971	The user has not finished his action in given time.	
972	Transaction has expired	
973	Minimum age not reached	
980	The transaction timed out at third party.	Please try again in asynchronous mode.
981	Payment declined by third party.	Payment method not available for this transaction. Customer needs to choose other payment method.
982	Payment rejected by third party.	Please decline the payment transaction.
990	Maintenance work	Please try again later.
991	Maintenance work at external service provider.	Please try again later.
1000	Parameter faulty or missing	
1001	Parameter {clearingtype} faulty or missing	
1002	Parameter {param} faulty	
1003	Parameter {reference} faulty or missing	
1004	Parameter {processing_time} faulty	
1005	Parameter {currency} faulty or missing	
1006	Parameter {due_time} faulty	
1007	Parameter {add_paydata} faulty or missing	
1008	Parameter {txid} faulty or missing	
1009	Parameter {sequencenumber} faulty or missing	
1010	Parameter {mode} faulty or missing	
1011	Parameter {workorderid} incorrect or missing	
1012	Parameter {api_version} incorrect or missing	
1013	Parameter {clearingsubtype} incorrect	

No.	Signification	Activity
1016	Parameter {amount} faulty or missing	
1018	Parameter {max_payout_amount} faulty	
1024	Parameter {booking_date} faulty	Check format
1025	Parameter {document_date} faulty	Check format
1031	Parameter {settleaccount} faulty	
1032	Parameter {capturemode} faulty	
1035	Parameter {transactiontype} faulty or missing	
1036	Parameter {customer_is_present} faulty or missing	
1037	Parameter {recurrence} faulty or missing	
1041	Parameter {successurl} faulty or missing	
1042	Parameter {backurl} faulty or missing	
1043	Parameter {errorurl} faulty or missing	
1055	Parameter {financingtype} faulty or missing	
1065	Parameter {onlinebanktransfertype} faulty or missing	
1071	Parameter {xid} faulty or missing	
1072	Parameter {storecarddata} faulty or missing	
1073	Parameter {pseudocardpan} faulty or missing	
1075	Parameter {cardissuenumber} faulty or missing	
1076	Parameter {cardtype} faulty or missing	
1077	Parameter {cardexpiredate} faulty or missing	
1078	Parameter {cardpan} faulty or missing	
1079	Parameter {cardcvc2} faulty or missing	
1080	Parameter {bankaccountholder} faulty or missing	
1081	Parameter {bankbranchcode} faulty or missing	
1082	Parameter {bankcountry} faulty or missing	
1083	Parameter {bankaccount} faulty or missing	
1084	Parameter {bankcode} faulty or missing	
1085	Parameter {bankgrouptype} faulty or missing	
1086	Parameter {bankcheckdigit} faulty or missing	
1087	Parameter {BIC} faulty or missing	
1088	Parameter {IBAN} faulty or missing	
1089	Parameter {elvtype} faulty or missing.	
1090	Parameter {shippingprovider} faulty or missing	
1091	Incomplete or missing bank-data. Define {bankaccount} or {iban}.	
1095	Parameter {wallettype} faulty or missing	
1096	Parameter {cashtype} incorrect or missing	
1100	Parameter {mandate_identification} faulty or missing	
1101	Parameter {mandate_status} faulty or missing	
1102	Parameter {mandate_dateofsignature} faulty or missing	
1103	Parameter {mandate_sequencetype} faulty or missing	
1104	Parameter {mandate_language} faulty or missing	
1110	Parameter {bankaccountholder_firstname} faulty or missing	
1111	Parameter {bankaccountholder_lastname} faulty or missing	
1112	Parameter {bankaccountholder_street} faulty or missing	

No.	Signification	Activity
1113	Parameter {bankaccountholder_zip} faulty or missing	
1114	Parameter {bankaccountholder_city} faulty or missing	
1115	Parameter {bankaccountholder_country} faulty or missing	
1116	Parameter {bankaccountholder_email} faulty or missing	
1117	Parameter {bankaccountholder_company} faulty or missing	
1201	Parameter {encoding} faulty or missing	
1202	Parameter {request} faulty or missing	
1203	Parameter {mid} faulty or missing	
1204	Parameter {aid} faulty or missing	
1205	Parameter {portalid} faulty or missing	
1206	Parameter {key} faulty	
1208	Parameter {responsetype} faulty	
1300	Parameter {customerid} faulty	
1301	Parameter {firstname} faulty or missing	
1302	Parameter {lastname} faulty or missing	
1303	Parameter {company} faulty or missing	
1310	Parameter {userid} faulty	
1311	Parameter {street} faulty or missing	
1312	Parameter {streetnumber} faulty or missing	
1313	Parameter {streetname} faulty or missing	
1314	Parameter {addressaddition} faulty or missing	
1321	Parameter {zip} faulty or missing	
1331	Parameter {city} faulty or missing	
1337	Billing address faulty or missing	
1338	Parameter combination {city} and {state} faulty or missing	
1339	Parameter {state} faulty or missing	State unknown or incorrect state
1340	Parameter {country} faulty or missing	Does not correspond with ISO 3166
1341	Parameter {language} faulty or missing	Does not correspond with ISO 639
1343	Parameter {birthday} faulty or missing	
1344	Parameter {telephonenumber} faulty or missing	
1345	Parameter {personalid} faulty or missing	
1346	Parameter {gender} faulty or missing	
1350	Parameter {email} faulty or missing	
1367	Parameter {shipping_street} faulty or missing	
1368	Parameter {shipping_company} faulty or missing	
1369	Parameter {shipping_state} faulty or missing	Shipping state unkown or wrong country
1370	Parameter {shipping_country} faulty or missing	Shipping country unkown or wrong
1371	Parameter {shipping_addressaddition} faulty or missing	
1372	Parameter {shipping_zip} faulty or missing	
1373	Parameter {shipping_city} faulty or missing	
1374	Parameter combination {shipping_city} and {shipping_state} faulty or missing	
1375	Shipping address faulty or missing	
1380	Parameter {ip} faulty or missing	

No.	Signification	Activity
1381	Parameter {use_customerdata} faulty or missing	
1390	Parameter {delete_carddata} faulty or missing	
1391	Parameter {delete_bankaccountdata} faulty or missing	
1421	Parameter {exiturl} faulty or missing	
1432	Parameter {checktype} faulty or missing	
1433	Parameter {scoretype} faulty or missing	
1434	Parameter {consumerscoretype} faulty or missing	
1435	Parameter {addresschecktype} faulty or missing	
1471	Parameter {phone_prefix} incorrect or missing	
1472	Parameter {phone_number} incorrect or missing	
1473	Parameter {pin} incorrect or missing	
1501	Parameter {reminderlevel} faulty or missing	
1502	Parameter {remindertime} faulty or missing	
1503	Parameter {sendreminder} faulty or missing	
1504	Parameter {failedcause} faulty or missing	
1610	Article list faulty or incomplete	
1611	Parameter {id[n]} faulty or missing	
1612	Parameter {pr[n]} faulty or missing	
1613	Parameter {no[n]} faulty or missing	
1614	Parameter {va[n]} faulty or missing	
1615	Parameter {sd[]} faulty or missing	
1616	Parameter {ed[]} faulty or missing	
1617	Parameter {it[]} faulty or missing	
1618	Gross-amount of an item trespass the maximum	
1631	Parameter {invoice_title} faulty or missing	
1632	Parameter {invoice_deliverymode} faulty or missing	
1633	Parameter {invoice_deliverydate} faulty or missing	
1634	Parameter {invoice_deliveryenddate} faulty or missing	
1650	Parameter {file_reference} faulty or missing	
1651	Parameter {file_type} faulty or missing	
1652	Parameter {file_format} faulty or missing	
1700	Parameter {action} faulty or missing	
1705	Parameter {accessid} faulty or missing	
1706	Parameter {access_price} faulty or missing	
1707	Parameter {access_aboprice} faulty or missing	
1708	Parameter {access_expiretime} faulty or missing	
1709	Parameter {access_period} faulty	
1710	Parameter {access_aboperiod} faulty	
1711	Parameter {access_starttime} faulty or missing	
1712	Parameter {access_canceltime} faulty or missing	
1713	Parameter {access_vat} faulty or missing	
1721	Parameter {productid} faulty or missing	
1723	Parameter {id_trail} incorrect or missing.	
1724	Parameter {no_trail} (quantity) faulty or missing	

No.	Signification	Activity
1725	Parameter {pr_trail} (price) faulty or missing	
1726	Parameter {de_trail} (description) faulty or missing	
1727	Parameter {va_trail} (VAT rate) faulty or missing	
1728	Parameter {period_unit_trail} faulty or missing	
1729	Parameter {period_length_trail} faulty or missing	
1730	Parameter {id_recurring} incorrect or missing.	
1731	Parameter {no_recurring} (quantity) faulty or missing	
1732	Parameter {pr_recurring} (price) faulty or missing	
1733	Parameter {de_recurring} (description) faulty or missing	
1734	Parameter {va_recurring} (VAT rate) faulty or missing	
1735	Parameter {period_unit_recurring} faulty or missing	
1736	Parameter {period_length_recurring} faulty or missing	
1737	No item data found for initial term.	
1738	No item data found for subsequent term.	
1739	Parameter {amount_trail} faulty or missing	
1740	Parameter {amount_recurring} faulty or missing	
1753	The indicated total price of all articles from the initial term did not correspond with the amounts of the individual items.	
1754	The indicated total price of all articles from the subsequent term did not correspond with the amounts of the individual items.	
1755	Incomplete item data for the initial term.	
1756	Incomplete item data for the subsequent term.	
1757	Incomplete time definition for the initial term.	
1758	Incomplete time definition for the subsequent term.	
1759	For more than one item position in the initial term the price cannot be transmitted without items.	
1760	For more than one item position in the subsequent term the price cannot be transmitted without items.	
1761	Parameter {ti_trail} faulty or missing	
1762	Parameter {ti_recurring} faulty or missing	
1810	Parameter {vaccountname} faulty or missing	
1811	Parameter {vreference} faulty or missing	
1813	Parameter {settleperiod} faulty or missing	
1814	Parameter {settletime} faulty or missing	
1815	Parameter {settle_period_unit} faulty or missing	
1816	Parameter {settle_period_length} faulty or missing	
1817	Parameter {payout_open_balance} faulty or missing	
1999	Parameter {ecommercemode} faulty or missing	
2001	Channel not enabled for this payment portal	Contact PAYONE please.
2002	Request not supported by activated channel	Contact PAYONE please.
2003	MerchantID not found or no rights	Change MerchantID
2004	SubAccountID not found or no rights	Change SubAccountID
2005	PortalID not found or no rights	Change PortalID
2006	Key incorrect	Change key

No.	Signification	Activity
2007	Hash incorrect	Change hash
2008	Invalid request - invalid parameter	
2011	Parameters from different API versions cannot be used simultaneously.	
2012	Parameters conflict in request or do not match	Check parameters and repeat request
2013	API-request contains invalid characters	Check Encoding and do not use control characters.
3000	Function not available	
3001	Parameter getusertoken invalid. Valid values are yes/no	
3002	A user-id or a customer-id must be set.	
3003	Usertoken not found.	The usertoken was not found. It is either not present or expired.
3004	The usertoken is not allowed within the current request.	The usertoken is not allowed within the current request.
3005	Usertoken and Userid are not to be transmitted simultaniously.	The use of usertoken and userid is mutually exclusive.
3006	The usertoken has a wrong format.	The usertoken transmitted is faulty. It should have a length of 24chars and should consist of alphanumerical characters including the .
3007	The transmitted customerid is not allowed.	You are already using the transmitted customerid.
3008	The desired customer is already present.	The customer to create is already present, you probably used a token generated by yourself.
3009	The customer does not exist.	Tht customer identified by the token does not exist.
3010	The modes do not match.	The modes of request and dataset have to match. They need to be either test or live.
3200	Customer not found or insufficient privileges	Change customer number
3301	Bank data cannot be deleted, order prevailing	
3302	Bank data cannot be deleted, VirtualAccount not balanced	
3303	Card data cannot be deleted, order prevailing	
3304	Card data cannot be deleted, VirtualAccount not balanced	
4001	Address check not enabled	
4002	Type of address check not enabled	
4010	Country not supported	Cancel or continue without address check
4011	Country not supported	Cancel or continue without consumer score check
4041	Address not found	
4042	Address not unique	
4051	City not found	Change city
4052	City ambiguous	Address correct, however it could not be corrected.
4053	Street not found	Change street, post code or city
4054	Street ambiguous	Address correct, however it could not be corrected.
4055	Wrong post office box number	Change post office box number
4056	Street number error	Change street number

No.	Signification	Activity
4057	Overseas address	No reference could be found for the submitted address.
4058	Street invalid	Change street
4101	Consumer score check not enabled	
4102	Scoretype incorrect	
4110	Credit reform, company credit information - no hits	Change query
4112	Credit reform, company credit information - too many hits	Change query
4200	Merchant does not participate in the 3D Secure procedure	Continue payment process without 3-D Secure
4201	Request is not supported by payment portal type	Change payment portal
4208	Card does not participate in 3D Secure	Continue payment process
4209	PaRes Check not necessary	Cancel payment process
4210	3D Secure error	Continue payment process without 3-D Secure
4211	PaRasCheck error (status=A).	Continue with 3DS (ECI06)
4212	PaRasCheck error (status=U).	Continue without 3DS (ECI07)
4213	PaRasCheck error (authentication failed).	Cancel
4214	3D Secure transaction requires XID.	Cancel activity
4215	Indicated ECI mode not supported by 3D Secure	Cancel activity
4216	XID not found.	Cancel activity
4217	3D Secure authentication not completed yet	Booking declined
4218	3D Secure authentication failed	Booking declined
4219	Successful 3D Secure authentication required	Booking declined
4302	Check type incorrect	Change check type
4743	Callback: Country not supported	Cancel or continue without callback
4744	Callback: Prefix disabled	Only use local or mobile numbers
4745	Callback: Number incorrect	Area code or base number incorrect
5050	Dunning level cannot be set	Dunning status does not allow setting a new dunning level
5052	Error on repeated sending of a reminder.	Please try again later
5054	Dunning date could not be set	
6032	Invoice not found	
6500	File not found	Check reference, format, type
6501	File not unique	Check reference, format, type
6502	Service temporarily unavailable	Please try again later.
7005	Access ID not found or no rights	
7008	Contract expiry date in the past or before the start date	
7011	Contract start date in the past	
7012	Term of contract has already started	
7021	ProductID not found or no rights	
7030	Operation not allowed under current access status.	
7040	The total amount of the items in the initial term equals 0.	
7041	The total amount of the items in the subsequent term equals 0.	
7042	The property {InvoiceAppendix} does not have the correct data type.	
7043	The IP address does not have the correct data type.	

No.	Signification	Activity
8010	VirtualAccountName already used for another customer	Select different name or select correct customer.
8011	Reference number (virtual account) already exists	Change reference number
8020	Currency does not correspond with the virtual account	Change currency
8030	Portal ID does not correspond with the virtual account.	Change portal ID
8040	Mode does not correspond with the virtual account.	Change mode

5.8 ISO Currencies (ISO 4127)

The PAYONE Platform and its shop modules are optimized for currencies with two decimals. Currencies without or with three decimals can be generally processed with certain limitations – please contact the PAYONE Merchant Service for further information.

E.g.:

Currency	Alpha	Num	Decimals
Euro	EUR	978	2
Australian Dollar	AUD	036	2
Swiss Franc	CHF	756	2
Danish Krone	DKK	208	2
Pound Sterling	GBP	826	2
Norwegian Krone	NOK	578	2
New Zealand Dollar	NZD	554	2
Swedish Krona	SEK	752	2
US Dollar	USD	840	2

5.9 ISO Countries (ISO 3166)

ISO	Country		ISO	Country
AD	Andorra		LA	Lao People's Democratic Republic
AE	United Arab Emirates		LB	Lebanon
AF	Afghanistan		LC	Saint Lucia
AG	Antigua And Barbuda		LI	Liechtenstein
AI	Anguilla		LK	Sri Lanka
AL	Albania		LR	Liberia
AM	Armenia		LS	Lesotho
AN	Netherlands Antilles		LT	Lithuania
AO	Angola		LU	Luxembourg
AQ	Antarctica		LV	Latvia
AR	Argentina		LY	Libyan Arab Jamahiriya
AS	American Samoa		MA	Morocco
AT	Austria		MC	Monaco
AU	Australia		MD	Moldova, Republic Of
AW	Aruba		MG	Madagascar
AZ	Azerbaijan		MH	Marshall Islands
BA	Bosnia And Herzegowina (Herzegowina)		MK	Republic Of Macedonia (The Former Yugoslav)
BB	Barbados		ML	Mali
BD	Bangladesh		MM	Myanmar
BE	Belgium		MN	Mongolia
BF	Burkina Faso		MO	Macau
BG	Bulgaria		MP	Northern Mariana Islands
BH	Bahrain		MQ	Martinique
BI	Burundi		MR	Mauritania
BJ	Benin		MS	Montserrat
BM	Bermuda		MT	Malta
BN	Brunei Darussalam		MU	Mauritius
BO	Bolivia		MV	Maldives
BR	Brazil		MW	Malawi
BS	Bahamas		MX	Mexico
BT	Bhutan		MY	Malaysia
BV	Bouvet Island		MZ	Mozambique
BW	Botswana		NA	Namibia
BY	Belarus		NC	New Caledonia
BZ	Belize		NE	Niger
CA	Canada		NF	Norfolk Island
CC	Cocos (Keeling) Islands		NG	Nigeria

ISO	Country		ISO	Country
CF	Central African Republic		NI	Nicaragua
CG	Congo		NL	Netherlands
CH	Switzerland		NO	Norway
CI	Cote d'Ivoire		NP	Nepal
CK	Cook Islands		NR	Nauru
CL	Chile		NU	Niue
CM	Cameroon		NZ	New Zealand
CN	China		OM	Oman
CO	Colombia		PA	Panama
CR	Costa Rica		PE	Peru
CU	Cuba		PF	French Polynesia
CV	Cape Verde		PG	Papua New Guinea
CX	Christmas Island		PH	Philippines
CY	Cyprus		PK	Pakistan
CZ	Czech Republic		PL	Poland
DE	Germany		PM	St. Pierre And Miquelon
DJ	Djibouti		PN	Pitcairn
DK	Denmark		PR	Puerto Rico
DM	Dominica		PT	Portugal
DO	Dominican Republic		PW	Palau
DZ	Algeria		PY	Paraguay
EC	Ecuador		QA	Qatar
EE	Estonia		RE	Reunion
EG	Egypt		RO	Romania
EH	Western Sahara		RU	Russian Federation
ER	Eritrea		RW	Rwanda
ES	Spain		SA	Saudi Arabia
ET	Ethiopia		SB	Solomon Islands
EU	Europe (SSGFI only)		SC	Seychelles
FI	Finland		SD	Sudan
FJ	Fiji		SE	Sweden
FK	Falkland Islands (Malvinas)		SG	Singapore
FM	Micronesia, Federated States Of		SH	St. Helena
FO	Faroe Islands		SI	Slovenia
FR	France		SJ	Svalbard And Jan Mayen Islands
FX	France, Metropolitan		SK	Slovakia (Slovak Republic)
GA	Gabon		SL	Sierra Leone
GB	United Kingdom (UK)		SM	San Marino
GD	Grenada		SN	Senegal
GE	Georgia		SO	Somalia

ISO	Country		ISO	Country
GF	French Guiana		SR	Suriname
GH	Ghana		ST	Sao Tome And Principe
GI	Gibraltar		SV	El Salvador
GL	Greenland		SY	Syrian Arab Republic
GM	Gambia		SZ	Swaziland
GN	Guinea		TC	Turks And Caicos Islands
GP	Guadeloupe		TD	Chad
GQ	Equatorial Guinea		TF	French Southern Territories
GR	Greece		TG	Togo
GS	South Georgia And The South Sandwich Islands		TH	Thailand
GT	Guatemala		TJ	Tajikistan
GU	Guam		TK	Tokelau
GW	Guinea-Bissau		TM	Turkmenistan
GY	Guyana		TN	Tunisia
HK	Hong Kong		TO	Tonga
HM	Heard And Mc Donald Islands		TP	East Timor
HN	Honduras		TR	Turkey
HR	Croatia (local name: Hrvatska)		TT	Trinidad And Tobago
HT	Haiti		TV	Tuvalu
HU	Hungary		TW	Taiwan, Province Of China
ID	Indonesia		TZ	Tanzania, United Republic Of
IE	Ireland		UA	Ukraine
II	International (SSGFI only)		UG	Uganda
IL	Israel		UM	United States Minor Outlying Islands
IN	India		US	United States
IO	British Indian Ocean Territory		UY	Uruguay
IQ	Iraq		UZ	Uzbekistan
IR	Iran (Islamic Republic Of)		VA	Vatican City State (Holy See)
IS	Iceland		VC	Saint Vincent And The Grenadines
IT	Italy		VE	Venezuela
JM	Jamaica		VG	Virgin Islands (British)
JO	Jordan		VI	Virgin Islands (U.S.)
JP	Japan		VN	Vietnam
KE	Kenya		VU	Vanuatu
KG	Kyrgyzstan		WF	Wallis And Futuna Islands
KH	Cambodia		WS	Samoa
KI	Kiribati		YE	Yemen
KM	Comoros		YT	Mayotte
KN	Saint Kitts And Nevis		YU	Yugoslavia

ISO	Country		ISO	Country
KP	Korea, Democratic People's Republic Of		ZA	South Africa
KR	Korea, Republic Of		ZM	Zambia
KW	Kuwait		ZR	Zaire
KY	Cayman Islands		ZW	Zimbabwe
KZ	Kazakhstan			

5.10 ISO Subdivisions (ISO 3166-2)

These values are used for parameter „state" and „shipping_state" and represent principal subdivisions of a country.

Here a list of a few codes – for complete reference refer to http://en.wikipedia.org/wiki/ISO_3166-2

5.10.1   US-Subdivisions

ISO	Country		ISO	Country
AK	Alaska		MT	Montana
AL	Alabama		NC	North Carolina
AR	Arkansas		ND	North Dakota
AS	American Samoa		NE	Nebraska
AZ	Arizona		NH	New Hampshire
CA	California		NJ	New Jersey
CO	Colorado		NM	New Mexico
CT	Connecticut		NV	Nevada
DC	District of Columbia		NY	New York
DE	Delaware		OH	Ohio
FL	Florida		OK	Oklahoma
GA	Georgia		OR	Oregon
GU	Guam		PA	Pennsylvania
HI	Hawaii		PR	Puerto Rico
IA	Iowa		RI	Rhode Island
ID	Idaho		SC	South Carolina
IL	Illinois		SD	South Dakota
IN	Indiana		TN	Tennessee
KS	Kansas		TX	Texas
KY	Kentucky		UM	United States Minor
LA	Louisiana		UT	Utah
MA	Massachusetts		VA	Virginia
MD	Maryland		VI	Virgin Islands, U.S.
ME	Maine		VT	Vermont
MI	Michigan		WA	Washington
MN	Minnesota		WI	Wisconsin
MO	Missouri		WV	West Virginia
MP	Northern Mariana Islands		WY	Wyoming
MS	Mississippi			

5.10.2   CA-Subdivisions

ISO	Country		ISO	Country
AB	Alberta		PE	Prince Edward Island
BC	British Columbia		QC	Quebec
MB	Manitoba		SK	Saskatchewan
NB	New Brunswick		NT	Northwest Territories
NL	Newfoundland and Labrador		NU	Nunavut
NS	Nova Scotia		YT	Yukon
ON	Ontario			

5.10.3   MX-Subdivisions

ISO	Country		ISO	Country
BCN	Baja California		MOR	Morelos
DIF	Distrito Federal		NAY	Nayarit
AGU	Aguascalientes		NLE	Nuevo León
BCS	Baja California Sur		OAX	Oaxaca
CAM	Campeche		PUE	Puebla
COA	Coahuila		QUE	Querétaro
COL	Colima		ROO	Quintana Roo
CHP	Chiapas		SLP	San Luis Potosí
CHH	Chihuahua		SIN	Sinaloa
DUR	Durango		SON	Sonora
GUA	Guanajuato		TAB	Tabasco
GRO	Guerrero		TAM	Tamaulipas
HID	Hidalgo		TLA	Tlaxcala
JAL	Jalisco		VER	Veracruz
MEX	México		YUC	Yucatán
MIC	Michoacán		ZAC	Zacatecas

5.10.4   AR-Subdivisions

Code	Country		Code	Country
C	Ciudad Autónoma de Buenos Aires		M	Mendoza
B	Buenos Aires		N	Misiones
K	Catamarca		Q	Neuquén
H	Chaco		R	Río Negro
U	Chubut		A	Salta
X	Córdoba		J	San Juan
W	Corrientes		D	San Luis
E	Entre Ríos		Z	Santa Cruz
P	Formosa		S	Santa Fe
Y	Jujuy		G	Santiago del Estero
L	La Pampa		V	Tierra del Fuego
F	La Rioja		T	Tucumán

5.10.5   BR-Subdivisions

Code	Country		Code ISO	Country
DF	Distrito Federal		PB	Paraíba
AC	Acre		PR	Paraná
AL	Alagoas		PE	Pernambuco
AP	Amapá		PI	Piauí
AM	Amazonas		RJ	Rio de Janeiro
BA	Bahia		RN	Rio Grande do Norte
CE	Ceará		RS	Rio Grande do Sul
ES	Espírito Santo		RO	Rondônia
GO	Goiás		RR	Roraima
MA	Maranhão		SC	Santa Catarina
MT	Mato Grosso		SP	São Paulo
MS	Mato Grosso do Sul		SE	Sergipe
MG	Minas Gerais		TO	Tocantins
PA	Pará			

5.10.6   CN-Subdivisions

Code	Country		Code	Country
11	Beijing		43	Hunan
12	Tianjin		44	Guangdong
13	Hebei		45	Guangxi
14	Shanxi		46	Hainan
15	Nei Mongol (mn), (Inner Mongolia)		50	Chongqing
21	Liaoning		51	Sichuan
22	Jilin		52	Guizhou
23	Heilongjiang		53	Yunnan
31	Shanghai		54	Xizang (Tibet)
32	Jiangsu		61	Shaanxi
33	Zhejiang		62	Gansu
34	Anhui		63	Qinghai
35	Fujian		64	Ningxia
36	Jiangxi		65	Xinjiang
37	Shandong		71	Taiwan
41	Henan		91	Hong Kong (en) Xianggang (zh)
42	Hubei		92	Macao (en) Aomen (zh)

5.10.7   ID-Subdivisions

Code	Country		Code	Country
AC	Aceh, Aceh		MA	Maluku, Maluku
BA	Bali, Bali		MU	Maluku Utara, North Maluku
BB	Bangka Belitung, Bangka–Belitung Islands		NB	Nusa Tenggara Barat, West Nusa Tenggara
BT	Banten, Banten		NT	Nusa Tenggara Timur, East Nusa Tenggara
BE	Bengkulu, Bengkulu		PA	Papua, Papua
GO	Gorontalo, Gorontalo		PB	Papua Barat, West Papua
JA	Jambi, Jambi		RI	Riau, Riau
JB	Jawa Barat, West Java		SR	Sulawesi Barat, West Sulawesi
JT	Jawa Tengah, Central Java		SN	Sulawesi Selatan, South Sulawesi
JI	Jawa Timur, East Java		ST	Sulawesi Tengah, Central Sulawesi
KB	Kalimantan Baratm, West Kalimantan		SG	Sulawesi Tenggara, Southeast Sulawesi
KS	Kalimantan Selatan, South Kalimantan		SA	Sulawesi Utara, North Sulawesi
KT	Kalimantan Tengah, Central Kalimantan		SB	Sumatera Barat, West Sumatra
KI	Kalimantan Timur, East Kalimantan		SS	Sumatera Selatan, South Sumatra
KU	Kalimantan Utara, North Kalimantan		SU	Sumatera Utara, North Sumatra

Code	Country		Code	Country
KR	Kepulauan Riau, Riau Islands		JK	Jakarta Raya, Jakarta
LA	Lampung, Lampung		YO	Yogyakarta, Yogyakarta

5.10.8 IN-Subdivisions

Code	Country		Code	Country
AP	Andhra Pradesh		NL	Nagaland
AR	Arunachal Pradesh		OR	Odisha
AS	Assam		PB	Punjab
BR	Bihar		RJ	Rajasthan
CT	Chhattisgarh		SK	Sikkim
GA	Goa		TN	Tamil Nadu
GJ	Gujarat		TG	Telangana
HR	Haryana		TR	Tripura
HP	Himachal Pradesh		UT	Uttarakhand
JK	Jammu and Kashmir		UP	Uttar Pradesh
JH	Jharkhand		WB	West Bengal
KA	Karnataka		AN	Andaman and Nicobar Islands
KL	Kerala		CH	Chandigarh
MP	Madhya Pradesh		DN	Dadra and Nagar Haveli
MH	Maharashtra		DD	Daman and Diu
MN	Manipur		DL	Delhi
ML	Meghalaya		LD	Lakshadweep
MZ	Mizoram		PY	Puducherry

5.10.9 JP-Subdivisions

Code	Country		Code	Country
23	Aiti (Aichi)		20	Nagano
05	Akita		42	Nagasaki
02	Aomori		29	Nara
38	Ehime		15	Niigata
21	Gihu (Gifu)		44	Ôita (Oita)
10	Gunma		33	Okayama
34	Hirosima (Hiroshima)		47	Okinawa
01	Hokkaidô (Hokkaido)		27	Ôsaka (Osaka)
18	Hukui (Fukui)		41	Saga
40	Hukuoka (Fukuoka)		11	Saitama
07	Hukusima (Fukushima)		25	Siga (Shiga)
28	Hyôgo (Hyogo)		32	Simane (Shimane)
08	Ibaraki		22	Sizuoka (Shizuoka)
17	Isikawa (Ishikawa)		12	Tiba (Chiba)
03	Iwate		09	Totigi (Tochigi)

Code	Country		Code	Country
37	Kagawa		36	Tokusima (Tokushima)
46	Kagosima (Kagoshima)		13	Tôkyô (Tokyo)
14	Kanagawa		31	Tottori
39	Kôti (Kochi)		16	Toyama
43	Kumamoto		30	Wakayama
26	Kyôto (Kyoto)		06	Yamagata
24	Mie		35	Yamaguti (Yamaguchi)
04	Miyagi		19	Yamanasi (Yamanashi)
45	Miyazaki			

5.10.10 TH-Subdivisions

Code	Country		Code	Country
10	Krung Thep Maha Nakhon [Bangkok]		82	Phangnga
S	Phatthaya		93	Phatthalung
37	Amnat Charoen		56	Phayao
15	Ang Thong		67	Phetchabun
38	Bueng Kan		76	Phetchaburi
31	Buri Ram		66	Phichit
24	Chachoengsao		65	Phitsanulok
18	Chai Nat		54	Phrae
36	Chaiyaphum		14	Phra Nakhon Si Ayutthaya
22	Chanthaburi		83	Phuket
50	Chiang Mai		25	Prachin Buri
57	Chiang Rai		77	Prachuap Khiri Khan
20	Chon Buri		85	Ranong
86	Chumphon		70	Ratchaburi
46	Kalasin		21	Rayong
62	Kamphaeng Phet		45	Roi Et
71	Kanchanaburi		27	Sa Kaeo
40	Khon Kaen		47	Sakon Nakhon
81	Krabi		11	Samut Prakan
52	Lampang		74	Samut Sakhon
51	Lamphun		75	Samut Songkhram
42	Loei		19	Saraburi
16	Lop Buri		91	Satun
58	Mae Hong Son		17	Sing Buri
44	Maha Sarakham		33	Si Sa Ket
49	Mukdahan		90	Songkhla
26	Nakhon Nayok		64	Sukhothai
73	Nakhon Pathom		72	Suphan Buri

Code	Country		Code	Country
48	Nakhon Phanom		84	Surat Thani
30	Nakhon Ratchasima		32	Surin
60	Nakhon Sawan		63	Tak
80	Nakhon Si Thammarat		92	Trang
55	Nan		23	Trat
96	Narathiwat		34	Ubon Ratchathani
39	Nong Bua Lam Phu		41	Udon Thani
43	Nong Khai		61	Uthai Thani
12	Nonthaburi		53	Uttaradit
13	Pathum Thani		95	Yala
94	Pattani		35	Yasothon

5.11 ISO Languages (ISO 639-1)

List of common ISO 639-1 Codes

ISO	Country		ISO	Country
de	German		lb	Luxembourgish
en	English		lt	Lithuanian
bg	Bulgarian		nl	Dutch
cs	Czech		no	Norwegian
es	Spanish; Castilian		pl	Polish
fi	Finnish		pt	Portuguese
fr	French		ro	Romanian
hr	Croatian		sl	Slovenian
hu	Hungarian		tr	Turkish
ja	Japanese			

6   Glossar

Name	Description
BBAN	Basic Bank Account Number Describes the bank account specified by "classic" values: countrycode bankaccount bankcode (where applicable) bankbranchcode (where applicable) bankcheckdigit (where applicable)
BIN	Bank Identification Number Describes the issuing bank of a credit card and is represented by the first six digits of a credit card number (PAN).
CVV	Card Verification Code (aka CVC, "cardcvc2") The card verification code is a 3 or 4 digit code and printed on the rear side of the credit card. It must not be stored at any time.
IBAN	International Bank Account Number IBAN and BIC specify an international bank account. BIC is planned to be optional within Germany from 2014-02-01 and optional within international bank transfer (in Europe) from 2016-02-01.
masked cardpan	The complete creditcard number must not be logged or displayed at any time. The masked creditcard number (masked cardpan) is shown as 6X4 and may be stored or shown to a user, e.g.: 1234 56xx xxxx 1234 or 1234 56xx xxx1 234
MOTO	Mail Or Telephone Order Describes the order process where an operator processes a payment with details for a customer. The customer provides the payment details (credit card details, bank account details) via email or phone.
PAN	Primary account number Is the complete credit card number (from 13 to 19 digits) and must not be processed or stored without PCI DSS certification (neither in database nor in log files). PAN may only be recorded either encrypted (only with PCI DSS certification) or masked/truncated (max: 6x4).
Payment Process ID	Unique payment process Id assigned to each payment process by PAYONE Platform.
PPS	PAYONE Payment Services All payments are processed via PAYONE accounts. Single contract with PAYONE, centralized payout services.
truncated cardpan	The complete creditcard number must not be logged or displayed at any time. The truncated creditcard number (truncated cardpan) is shown as X4 and may be stored or shown to a user, e.g.: xxxx xxxx xxxx 1234 or xxxx xxxx xxx1 234

7   Contact

Do you have questions or do you need further information?

Then contact us. The BS PAYONE Technical Support is always happy to provide assistance.

BS PAYONE GmbH

Office Kiel

Fraunhoferstr. 2-4

24118 Kiel

Phone: +49 431 25 968 - 500

Fax: +49 431 25 968 - 1500

www.bspayone.com

tech.support@bspayone.com