



Developer's Guide

Prepared for:

DataCash Merchants

Date:

January 2012

Document Control					
Reference	API001	Date	19/01/2012	Version	2.37
File Name	DataCash_Developers_Guide_2.37				
REVISION HISTORY					
Revision	Date	Comments / Reason			
2.0 (Release)	26/11/2009	Developers Guide from DPG and APG combined			
2.1	27/11/2009	Added Fuel Card Service			
2.2	08/12/2009	Added PayPal Reference Transactions Added Bankcard Query Transactions			
2.3	11/01/2010	InstaDebit and ELV details added (FR)			
2.4	28/01/2010	Standardisation of formatting and layout (JF)			
2.5	09/02/2010	Updated HPS section with details for dynamic capture fields and expired session handling (NB) Updated PayPal Reference Transactions (NB)			
2.6	19/02/2010	Updated HpsTxn response table and related XML (CTI) Updated paReq example – textarea (JF)			
2.7	03/03/2010	Grammar and format editing (JF) Update to DataCash MPI And MPI Only (JF) Update to Pre Registered Card Service card retention from 12 to 13 months (JF)			
2.8	21/04/2010	Change HPS to HCC (JF) Updated HCC Section with details for dynamic data placeholders (RWM)			
2.9	22/04/2010	Updates to ExperCash section (SJ) Added ExperCash BackCharge information (SJ)			
2.10	25/05/2010	New section added for Full HPS (RWM) Updates to links to Appendices (AM) Added GiroPay sample (AM)			
2.11	07/06/2010	New section added for Payout transactions (RJB)			
2.12	02/08/2010	Update to Direct Debit Notifications re Drawdown Notifications (JF) Update to <iframe /> mechanism example (JF)			
2.13	16/09/2010	Updates to Single Card Payments and Card Holder Present sections for authorize_referral_request transactions (SPD) Update to details of required URLs for HPS HCC and 3DS (JF)			
2.14	21/10/2010	Added information of the new eCheck integration into APG (HA) Added ClickandBuy and Envoy OneClick information for Payout into APG (RB)			
2.15	27/10/2010	Addition of new section (2.8.7) describing the Vtid Configuration request and response (CI)			
2.16	16/11/2010	Added information on the ExperCash IFrame. See section 0 (RJB) Added information on the Merchant Custom Fields. See section 0 (RJB) Added information on the Risk checking for Retail Merchants. See section 0 (RJB)			

2.17	21/12/2010	Inclusion of additional Sales Channels (refer to section 2.4.7.1.2) (JF) Max characters of prod_description field changed from 50 to 100 characters (refer to section 2.4.7.1.7) (JF) Added additional supported fields for T3M (DEJ) Added information on FraudOnlyTxn (DEJ) Added information on OneClick responses – refer to section 0 (RJB)
2.18	19/01/2011	Update to contacts for HPS (JF) Update to FraudOnlyTxn (DEJ)
2.19	04/02/2011	Added SaleAmountDetails to the APG Purchase Method. Section 0 (RJB) Added RSG sections 2.4.10 & 2.4.11 (DEJ)
2.20	15/03/2011	Added American Express ISO8583 AdditionalData Section (DEJ) Added UseSaleCurrency element to section 3.3.3.1.7 (HA)
2.21	08/04/2011	Added information of the new Amazon integration into APG, adding LineItem information under purchaseDetails (SJ) Added version section (JW) Added version 2 specific section to MPI Only section (JW) Updated T3M documentation to include airline examples (DEJ)

		Removed capitalisation on the vTID.configuration documentation (SA) PayPal:- re-ordering of elements in alphabetical order with required field first and optional after. (SA) PayPal: PayPalTxn.reference. Is required for G (use from S)(SA) PayPal.AirlineItineraryData is optional for D&C, PayPal.Items optional for D (SA) PayPal.ShippingAddress optional for D (SA) PayPal: Updated "Values for paymentstatus" table in "pending"line to explain that this only applies PayPal.payment_type = "Any" (SA) PayPal:update the "timestamp" fields in the XML response examples to be more recent ones (SA) PayPal:updated PayPal Auth Capture, PayPal.payment_action must be "order" for S (SA) PayPal: updated PayPal Auth Capture, PayPal.payment_action may be "order" for D, or "sale" if wishing to convert the existing txn (SA) PayPal: Example URL for re-direct - added the URL for the sandbox (www.sandbox.paypal) (SA) PayPal: Added an xml for paypal elements (SA) PayPal: AirlineItineraryData and FlightDetails. Updated the PayPal Do Reference service. (SA) PayPal: Amended the title on the second do_capture transaction example. (SA) PayPal: Updated the PayPal do reference for: TxnDetails.amount; PayPalTxn.reference; PayPalTxn.payment_action. (SA) PayPal: Added Extra XML examples (SA) PayPal: Updated the title for Example XML Requests for update_billing_agreement (SA) PayPal: Updated the title for Example XML Request for do_reference_transaction and added an extra example (SA) Removed reference to device_category=1 in the request.Transaction.TxnDetails.ThreeDSecureBrowser (SA) Added missing information to "Schema Elements for Query Response" about the "HpsTxn" element (SA) Added version 2 specific response changes (MPM)
2.22	25/05/2011	

2.23	03/06/2011	Removed DPG-Realtime Fraud Screening (MS) Removed 3rdman references from text and replaced with Datacash Fraud Service(DFS) (MS) Updated the merchant_url description for 3Dsecure (SA)
2.24	08/06/2011	Updated the example XML under credit and debit card service section (SA) Updated the Secure element for 3d transaction to include American Express (LL)
2.25	06/07/2011	Added section for PayPal Query Transactions (RWM)
2.26	13/07/2011	Added Tokenization elements (WB)
2.27	19/07/2011	Updated broken links
2.28	11/08/2011	Update to Tokenization XML and element name as well as the reference to the appendices.
2.29	15/08/2011	Updated the Tokenization spelling to the US spelling: tokenization
2.30	02/09/2011	Updated to include risk: https responses
2.31	22/09/2011	Updates to APG Section Offline Bank Transfer, iDeal and WebMoney has been added to Purchases (RJB) Removed Not Applicable items from the Required Payment Lists (RJB) Added EntroPay and Ukash to the Payout section of APG. (RJB) Added extra nodes in the Offline Bank Transfer response XML. Updated to include Switch to Issuer (DMcC)
2.32	24/10/2011	Update to Merchant Narrative section to include txn_refund (JMW)
2.33	08/11/2011	Added documentation for PreCall transaction (DEJ) Added documentation for MiscCustomerInfo Block (DEJ) Added 2 additional fields for Chase and version 2 transactions and example xml (MPM)

2.34	21/11/2011	<p>Update to include new Alternative Payments Section (EDI-DPG). (DEJ)</p> <p>Update to the HPS TXN Elements, Attributtes and Example XML(OHB)</p> <p>Added details of the Local Direct Debit functionality (NXB)</p>
2.35	23/12/2011	<p>Updated the values for Paypal: "localecode"</p> <p>Updated the link to "website" in the Elements of Cv2Avs</p>
2.36	09/01/2012	<p>Removed <i>type</i> attribute from the examples where the <i>Secure</i> element can be found.</p> <p>Also removed the 'Attributes of service' table for Secure element. (PXR)</p> <p>Added description of Custom Data element (RWM)</p>
2.37	19/01/2012	<p>Edited "Elements of HistoricTxn" authcode and reference (SA)</p> <p>Edited Example XML threedsecure_authorize_referral_request, changed element from "auth_code" to "authcode" (SA)</p> <p>Updated hyperlinks (AS)</p>
2.38	25/01/2012	<p>Restructured HCC section, providing additional examples and cross-referencing to other services (SA)</p> <p>Restructured HPS section, providing additional examples and cross-referencing to other services (SA)</p> <p>Moved "Example XML Request for an additional fulfill when configured for SplitShipment" to appropriate section</p> <p>Added additional cross-references and FAQ link to Bankcard Query Service (SA)</p>

TABLE OF CONTENTS

PAGE

1.	Introduction to the DataCash Developers Guide	1
1.1	Audience	1
1.2	Scope	1
1.3	Support.....	1
1.4	How to use this document	1
1.4.1	Which Gateway?.....	1
1.4.2	The Layout Structure Used In This Document.....	2
2.	DPG - DataCash Payment Gateway	3
2.1	Introduction to DPG	3
2.1.1	XML Requests and Responses	3
2.2	Single Card Payments	6
2.2.1	Credit and Debit Card Service	6
2.2.3	Additional Data (American Express ISO8583).....	21
2.2.4	Misc Customer Info.....	30
2.2.5	Line Item Detail	31
2.2.6	Airlines Transaction Records	39
2.2.7	Merchant Narrative	54
2.2.8	Fuel Card Service	56
2.2.9	Hosted Card Capture.....	65
2.2.10	Hosted Payment Service.....	84
2.3	Repeat Card Payments	96
2.3.1	Payment Tokenization	96
2.3.2	Capture Method Recurring Transactions.....	100
2.3.3	Historic Recurring Transactions	105
2.3.4	Fire and Forget Recurring Transactions.....	113
2.4	Fraud Prevention	123
2.4.1	AVSCV2.....	123
2.4.2	URU®	140
2.4.3	ReD Fraud Screening	175
2.4.4	3-D Secure, with DataCash MPI.....	198
2.4.5	MPI Only	212
2.4.6	3-D Secure, with 3 rd party MPI	224
2.4.7	Batched Fraud Screening.....	230
2.4.8	Fraud Only Transactions	254
2.4.9	Risk Services Gateway (RSG)	258
2.4.10	Pre & Post Authorisation Fraud Screening	261
2.4.11	HTTPS Responses	286
2.5	BACS Service	288
2.5.1	Direct Debit	288
2.5.2	Direct Credit – Direct Debit Refunds.....	316
2.5.3	Direct Credit – Standard.....	321
2.5.4	Direct Credit – Card Collection Accounts	327
2.6	Batch Input	332
2.6.1	Available DPG Services.....	333
2.6.2	Overview of Common Elements and Attributes	334
2.6.3	Example Batch Requests	348
2.7	Card Holder Present.....	365
2.7.1	Chip and Pin	365
2.7.2	Preallocated References.....	379
2.8	Other DPG Services	383
2.8.1	PayPal.....	383

2.8.2	Variable End of Day	452
2.8.3	RBS Gift Card.....	455
2.8.4	PrePay Technology (PPT)	464
2.8.5	Fexco Dynamic Currency Conversion.....	476
2.8.6	Bankcard Query Transactions.....	478
2.8.7	vTID Configuration Transactions.....	484
2.8.8	PreCall Transaction	486
2.9	Card Tokenization.....	491
2.9.1	Introduction	491
2.9.2	Token generation	491
2.10	Switch to Issuer Transactions.....	494
2.10.1	Versioning	494
2.10.2	Standard transaction - response elements	494
2.10.3	Query transaction – response elements	495
2.10.4	Transactions including CSC data	495
2.10.5	Transactions including AVS data	497
2.11	Local Direct Debit	500
2.11.1	Service Overview.....	500
2.11.2	LocalDDTxn	500
2.11.3	TxnDetails	502
2.11.4	HistoricTxn	503
2.12	Alternative Payment Methods.....	506
2.12.1	Common Request Elements	506
2.12.2	iDeal Transactions	507
2.12.3	Sofort Transactions.....	509
2.12.4	WebMoney Transactions	512
2.12.5	MisterCash Transactions	514
2.12.6	La Caixa Transactions	517
2.12.7	Nordea Transactions	519
2.12.8	Common Response Elements	520
2.12.9	Offline Bank Transfer (OBT) Response Elements	521
2.12.10	Redirect Response Elements.....	521
2.12.11	Compatibility With HPS	523
3.	APG – Alternative Payment Gateway	526
3.1	Introduction To APG.....	526
3.1.1	Glossary	526
3.2	APG Overview	526
3.2.1	APG SOAP Web Service	526
3.2.2	APG Redirect Solution	527
3.2.3	Security	527
3.2.4	Transaction Flow	528
3.3	APG SOAP Web Service	534
3.3.1	Shared XML Request and Response elements.....	534
3.3.2	Purchase	540
3.3.3	Payout	654
3.3.4	Transaction Querying	682
3.3.5	RetrieveSettlementValue	687
3.3.6	GetInformation	692
3.3.7	Modification	709
3.4	APG Event Notifications	716

1. INTRODUCTION TO THE DATACASH DEVELOPERS GUIDE

1.1 AUDIENCE

This document is intended to be used by developers, technical personnel and business analysts to facilitate a successful integration by a merchant to the gateways provided by DataCash.

This is reflected in the layout of the document, which covers the submission of data in XML form to DataCash.

1.2 SCOPE

This document describes the interfaces, messages, data formats, and responses used for integration into DataCash. This is a technical API document and as such does not cover the business reasons or business process for integration.

This document is supplemented with an Appendices document.

Both the Developers Guide and the Appendices are available for download from the DataCash Developers Area: <https://testserver.datacash.com/software/download.cgi>.

This document describes some of the services that DataCash currently provides at a high level. The other services' descriptions can be found on the DataCash website, and a link to the relevant webpage will be provided in this document.

1.3 SUPPORT

For any assistance or information pertaining to existing or new DataCash services, please [contact Support](#) via the DataCash website.

1.4 HOW TO USE THIS DOCUMENT

1.4.1 Which Gateway?

This document is split into two main parts, each catering for specific processing needs.

- The first part deals with integrating into the DPG (DataCash Payment Gateway) system. If you would like to process credit card transactions via DataCash, then this section applies to you. Please see Section 2 DPG - DataCash Payment Gateway.
 - Note that this system also includes functionality to process PayPal and UK Direct Debit (BACs) transactions.
- The second part deals with integrating into the APG (Alternative Payment Gateway) system. If you would like to process non-card purchase or payout method transactions via DataCash (electronic wallets, prepaid products, and bank transfers), then this section applies to you. Please see section 3 functionality.
- The APG section .

Both of these systems provide Risk Screening Services.

1.4.2 *The Layout Structure Used In This Document*

Despite the length of this document, there is a simple structure in place to make it easy to use.

Firstly, as described above, the document is split into two sections: one for card processing and the other for non-card payment method processing.

Secondly, within each of the two main sections the Services that are provided by each gateway are listed.

Thirdly, for each Service there are four sub sections:

1. Request XML complex structures
2. Request examples
3. Response XML complex structures
4. Response examples

Fourthly, for each XML complex structure, information is provided about:

- Description and position
- Elements
- Attributes
- Examples

And lastly, full examples of the Request and Response complex structures are provided, showing a number of variations depending on the scenario.

Cross-references are used throughout this document, for XML complex structures that are shared between various Services.

(To navigate forward, click on the link)

[Back to Contents](#)

2. DPG - DATA CASH PAYMENT GATEWAY

2.1 INTRODUCTION TO DPG

This section of the document is the DataCash Developer's Guide for the DPG. It is aimed at developers and technical personnel who will be integrating applications for card, PayPal and UK Direct Debit processing.

As each DPG Service requires particular information to be submitted and this information tends to be grouped within similar areas (parent elements) of the XML schema, the names of the parent elements will first be introduced.

Each parent will then be placed into its context in the schema and its child elements discussed. This includes any restrictions on the format, length and transaction type of each element.

If the DPG Service contains additional aspects (for example notifications for Standard Direct Debit and Standard Recurring Transactions), this will also be covered.

2.1.1 *XML Requests and Responses*

There are certain features of the Request and Response that are applicable to all DPG Services; these elements are covered in this section. Others are only used for a particular Service or group of Services; these elements will be covered within the documentation for that Service.

2.1.1.1 Request

The Request has an optional attribute "version" which can be sent to the DPG.

The version of the request when supplied indicates version specific functionality which may vary depending on the type of request.

The attribute corresponds to an unsigned integer.

It is important to note that when a version attribute is specified with a value greater than 1, the Response Element will be returned with a version attribute that directly corresponds to the value passed in with the Request.

The vast majority of requests will not have separate versions. When the version attribute is not supplied this value will default to "1". Please note that where version is not explicitly mentioned when describing specific types of requests in this document, it can be assumed that version 1 is the only supported version of the request. It can also be assumed that supplying a greater version number will not adversely affect current functionality for these services.

Where version is explicitly documented when describing specific types of request, special attention should be given to the effects of changing the version attribute of the Request.

Example Request Using Optional Version Attribute

```
<Request version="2">
...
</Request>
```

In the Request, only the Authentication element is used for all DPG Services.

2.1.1.2 Authentication

Element Name: Authentication
Position: Request

Element Name	Description	Values / Limitations
client	Specifies the account the transaction will be processed on (vTID)	As advised by DataCash
password	The password for the account.	As advised by DataCash ¹

- When your account is setup, DataCash will advise you of the password. This password has a maximum lifetime of twelve months and you are also responsible for changing it each time a person who knows it leaves your organisation.

Further information about generating a new password is available in the Support Centre: http://datacash.custhelp.com/cgi-bin/datacash.cfg/php/enduser/std_adp.php?p_faqid=567
(keyword search "DPG password")

Example XML for Authentication complex elements

```
<Authentication>
  <client>99000001</client>
  <password>mypasswd</password>
</Authentication>
```

2.1.1.3 Response

There are several elements to the Response that are returned regardless of the Service used. These are:

Element Name	Description
Status	Numeric return code indicating the result of the transaction
Reason	A text field expanding on the status of the transaction
datacash_reference	The DataCash reference of the transaction.
Time	The Unix Timestamp at which the transaction reached the DataCash server
Mode	Indicates the current status of your account – one of: live, test or accreditation

There are also the following additional elements that may be returned depending upon the result of the transaction:

Element Name	Description
Information	If an error is generated, additional information is often returned to allow the source of the error to be identified
merchant_reference	Your reference number

[Back to Contents](#)

2.2 SINGLE CARD PAYMENTS

2.2.1 Credit and Debit Card Service

A technical introduction to this Service is available on the website:

<http://www.datacash.com/gettingproducts.php?id=Bank-Card-Processing->

2.2.1.1 Schema Elements for Request

In this section the required fields for each transaction type will be presented, along with example XML for those fields. The XML is presented in italics for those fields that are not required for all transaction types. If the transaction type is specified in the XML, this is **highlighted** in the cases where other transaction types can be used in its place.

As not all transaction types use the same fields, each field will be labelled with the following key:

- O - Optional
- R - Required, field must be presented
- X - Excluded, presenting this field will cause the transaction to fail
- M - Mandatory if Available, if the information is available, it should be presented

Please refer to the [website](#) for definitions of the transaction types and examples of when you may wish to implement them.

The following keys will be used for each transaction type:

- A - auth
- P - pre
- R - refund
- E - erp
- C - cancel
- F - fulfill
- TR - txn_refund
- ARR - authorize_referral_request

Initial Transactions with Card details

The `auth`, `pre`, `refund` and `erp` transaction types all require the same information about the transaction to be provided. This data is passed in these distinct places in the schema:

- Request
 - Authentication – section 2.1.1.2
 - Transaction
 - CardTxn – the type and authorisation code, section 2.2.1.1.2
 - Card – this contains all the information about the Card, section 2.2.1.1.1
 - TxnDetails – contains details of the transaction, section 2.2.1.1.3

Historic Transactions

The cancel, fulfill, txn_refund and authorize_referral_request transaction types all require similar information about the transaction to be provided. This data is passed in these distinct places in the schema:

- Request
 - Authentication – section 2.1.1.2
 - Transaction
 - HistoricTxn – information relating to the original transaction section 2.2.1.1.4
 - Card – this contains all the information about the Card, section 2.2.1.1.1
 - TxnDetails – contains details of the transaction, section 2.2.1.1.3

2.2.1.1.1 Card

Element Name: Card

Position: Request.Transaction.CardTxn

Elements of Card						
Element Name	Description	Values / Limitations	A	P	R	E
pan	card number	must be a valid card number between 13 and 19 digits in length	R	R	R	R
expirydate	expiry date for the card	must be passed in mm/yy format	R	R	R	R
startdate	start date for the card	must be passed in mm/yy format	M	M	M	M
issuemumber	issue number of the card	must be one or two digits long	M	M	M	M

The CardInfo files can be used to determine whether a startdate or an issuemumber is required for each card.

Example XML for Card complex elements

```

<Card>
  <pan>444433*****1</pan>
  <expirydate>04/06</expirydate>
</Card>

<Card>
  <pan>675998*****1</pan>
  <expirydate>04/06</expirydate>
  <startdate>01/99</startdate>
  <issuenumber>01</issuenumber>
</Card>

```

2.2.1.1.2 CardTxn

For transactions using card details only. In addition to the basic Card details, the transaction type and authorisation code (if applicable) are sent in the CardTxn element:

Element Name:	CardTxn
Position:	Request.Transaction

Elements of CardTxn						
Element Name	Description	Values / Limitations	A	P	R	E
Card	see section 2.2.1.1.1					
authcode	authorisation code received from the bank	If presented, must be value received from Banks Authorisation centre	O	O	O	O
method	the transaction type	auth pre refund erp	R	R	R	R

Example XML for CardTxn complex elements

```

<CardTxn>
  <Card>...</Card>
  <authcode>123123</authcode>
  <method>pre</method>
</CardTxn>

```

2.2.1.1.3 TxnDetails

For all transactions:

Element Name:	TxnDetails
Position:	Request.Transaction

Elements of TxnDetails										
Element Name	Description	Values/ Limitations	A	P	R	E	C	F	TR	ARR
merchantreference	A unique reference number for each transaction	Min 6, max 30 alphanumeric characters . Must be unique	R	R	R	R	-	*	-	-
amount	The value of the transaction		R	R	R	R	-	O	O	-
custom_data	Custom data to be stored against the transaction, for view/search in Reporting system	Max 30 alphanumeric characters.	O	O	O	O	O	O	O	O

Attributes for Elements of TxnDetails										
Attribute Name	Attribute of Element	Values / Limitations	A	P	R	E	C	F	TR	ARR
currency	amount	Currency is passed in the 3 character ISO 4217 Alphabetic format (Default is GBP) e.g. GBP, USD, EUR	O	O	O	O	O	O	X	-

Note: For fulfill transactions, the merchantreference field is not required. However, if the Merchant is subscribed to the SplitShipment Service and has already fulfilled part of the transaction, the merchantreference field must be presented for any subsequent fulfill transactions submitted on that transaction.

Example XML for TxnDetails complex elements:
Card Transaction

```
<TxnDetails>
  <merchantreference>myreference123459988</merchantreference>
  <amount currency="EUR">146.99</amount>
  <custom_data>1AZ23HHH</custom_data>
</TxnDetails>
```

Historic Transaction

```
<TxnDetails>
  <amount>146.99</amount>
</TxnDetails>
```

Additional fulfill when subscribed to SplitShipment

```
<TxnDetails>
  <amount>146.99</amount>
  <merchantreference>myreference123459988_1</merchantreference>
</TxnDetails>
```

2.2.1.1.4 HistoricTxn

This element is the direct equivalent of the Card element (for transactions using card details).

Element Name:	HistoricTxn
Position:	Request.Transaction

Elements of HistoricTxn

Element Name	Description	Values / Limitations	C	F	TR	ARR
reference	DataCash unique reference of the original transaction	must be a valid transaction	R	R	R	R
authcode	DataCash authorisation code of the original transaction. For fulfill and authorize_referral_request, if the original transaction was referred, then provide the value received from Banks Authorisation centre	-	n/a	R	-	R
method	The transaction type	fulfill txn_refund cancel authorize_referral_request	R	R	R	R

Example XML Request for HistoricTxn complex elements

```
<HistoricTxn>
  <reference>4100200039275407</reference>
  <authcode>641413</authcode>
  <method>fulfill</method>
</HistoricTxn>
```

2.2.1.1.4.1 Acquirer Specific Notes

For the Chase PaymentTech acquirer, an optional `reversal="true"` attribute can be supplied within the `method` element when a cancel is being performed. This has the effect of attempting an online transaction reversal with the acquirer, in addition to excluding the transaction from the settlement file.

2.2.1.1.5 AdditionalInfo

Element Name	AdditionalInfo
Position	Request.Transaction.TxnDetails
Children	Yes
Attributes	No

Elements of AdditionalInfo

Element Name	Description	Values/Limitations	Required
addendumdata			O
risk_bypass		Y/N	O
CustomerInfo	See below		O

2.2.1.1.5.1 CustomerInfo

Element Name	CustomerInfo
Position	Request.Transaction.TxnDetails.AdditionalInfo
Children	Yes
Attributes	No

Elements of CustomerInfo

Element Name	Description	Values/Limitations	Required
customer_ip_address	IP Address	Must be a valid IP Address	O

2.2.1.2 XML Examples Requests

2.2.1.2.1 Transactions with Card Details

Example XML Request for refund

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <CardTxn>
      <Card>
        <pan>633300*****1</pan>
        <expirydate>04/06</expirydate>
        <startdate>01/04</startdate>
      </Card>
      <method>refund</method>
    </CardTxn>
    <TxnDetails>
      <merchantreference>1000001</merchantreference>
      <amount currency="GBP">95.99</amount>
      <custom_data>1AZ23HHH</custom_data>
    </TxnDetails>
  </Transaction>
</Request>
```

If the `method` is changed, the same XML could also be used for `auth`, `pre` and `erp` transactions.

Example XML Request for pre with an authcode

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <CardTxn>
      <Card>
        <pan>444433*****1</pan>
        <expirydate>12/08</expirydate>
        <startdate>03/05</startdate>
      </Card>
      <authcode>439673</authcode>
      <method>pre</method>
    </CardTxn>
    <TxnDetails>
      <merchantreference>1000023</merchantreference>
      <amount currency="USD">1800.00</amount>
    </TxnDetails>
  </Transaction>
</Request>
```

2.2.1.2.2 Historic Transactions**Example XML Request for cancel**

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <reference>4900200000000001</reference>
      <method>cancel</method>
    </HistoricTxn>
  </Transaction>
</Request>
```

Example XML Request for fulfill

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount>25.00</amount>
    </TxnDetails>
    <HistoricTxn>
      <reference>4900200000000001</reference>
      <authcode>A6</authcode>
      <method>fulfill</method>
    </HistoricTxn>
  </Transaction>
</Request>
```

Example XML Request for an additional fulfill when configured for SplitShipment

```
<Request>
  <Authentication>...</Authentication>
  <Transaction>
    <HistoricTxn>
      <authcode>534872</authcode>
      <reference>4400201011754019</reference>
      <method>fulfill</method>
    </HistoricTxn>
    <TxnDetails>
      <amount>30.99</amount>
      <merchantreference>fulfill_2</merchantreference>
    </TxnDetails>
  </Transaction>
</Request>
```

Example XML Request for a txn_refund

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount>10.00</amount>
    </TxnDetails>
    <HistoricTxn>
      <reference>3100000088888888</reference>
      <method>txn_refund</method>
    </HistoricTxn>
  </Transaction>
</Request>
```

Example XML Request for an authorize_referral_request

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <reference>3100000088888888</reference>
      <authcode>A6</authcode>
      <method>authorize_referral_request</method>
    </HistoricTxn>
  </Transaction>
</Request>
```

2.2.1.3 Schema Elements for Response

In addition to the elements covered in this section, Responses for this Service will also contain the general Response elements, as described in section 2.1.1.3.

2.2.1.3.1 CardTxn

Element Name:	CardTxn
Position:	Response

Element Name	Description	Values / Limitations
issuer	The Card Issuing Bank, if known	
card_scheme	The Card Scheme, if known. Current values which may be returned are listed	<ul style="list-style-type: none"> • American Express • Arval Fuel Card • ATM • Debit Mastercard • Diners Club • Discover • Duet • EnRoute • GE Capital • JCB • Key Fuels Card • Laser • Maestro • Mastercard • Platima • Solo • PPT • RBS Gift Card • ReD Fuel Card • Red Liquid Fuel Card • UK Fuel Card • VISA • VISA Debit • VISA Electron • Visa Purchasing
country	The Country of Issue, if known	
authcode	Authorisation code for successful transactions. Additional information from the bank if declined or referred	

Example XML Response for CardTxn complex elements

```

<CardTxn>
  <authcode>01HD</authcode>
  <card_scheme>Visa</card_scheme>
  <country>France</country>
  <issuer>Credit Lyonnais</issuer>
</CardTxn>

```

2.2.1.4 XML Example Responses

There are various results that can be returned for transactions. These can be clustered into two groups:

- Bank Responses – the transaction is submitted to the bank
- Error codes – an error occurred which prevented the transaction from being sent to the bank

If the transaction is submitted to the bank, the bank can either accept, decline or refer the transaction. Please refer to the [website](#) for additional information on why transactions can be declined or referred.

Status	Meaning
1	The bank has authorised the transaction
7	The bank has declined or referred the transaction
All others	All other responses are error codes

As there are many different return codes, this document only contains examples for status 1 and 7. A complete list of Response Codes for this service is available on the website. The Support Centre also contains extensive examples for most error codes, with illustrations of how they would appear in both Reporting and an XML Response, plus suggestions to prevent them occurring.

- [Support Centre](#)
- [Website](#)

2.2.1.4.1 Initial Transactions

Example XML Response for an authorised initial transaction

```
<Response>
  <CardTxn>
    <authcode>060642</authcode>
    <card_scheme>Mastercard</card_scheme>
    <country>United Kingdom</country>
    <issuer>HSBC</issuer>
  </CardTxn>
  <datacash_reference>300000088888888</datacash_reference>
  <merchantreference>1000001</merchantreference>
  <mode>LIVE</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1071567305</time>
</Response>
```

Example XML Response for a REFERRED transaction

```
<Response>
  <CardTxn>
    <authcode>CALL AUTH CENTRE</authcode>
    <card_scheme>Mastercard</card_scheme>
    <country>United Kingdom</country>
    <issuer>BARCLAYS BANK PLC</issuer>
  </CardTxn>
  <datacash_reference>31000008888123</datacash_reference>
  <merchantreference>1000113</merchantreference>
  <mode>LIVE</mode>
  <reason>REFERRED</reason>
  <status>7</status>
  <time>1134692433</time>
</Response>
```

Example XML Response for a DECLINED transaction

```
<Response>
  <CardTxn>
    <authcode>DECLINED</authcode>
    <card_scheme>Mastercard</card_scheme>
    <country>United Kingdom</country>
  </CardTxn>
  <datacash_reference>4400200045583767</datacash_reference>
  <merchantreference>AA004630</merchantreference>
  <mode>TEST</mode>
  <reason>DECLINED</reason>
  <status>7</status>
  <time>1169223906</time>
</Response>
```

2.2.1.4.2 Historic Transactions

Example XML Response for a successful cancel

```
<Response>
  <datacash_reference>4900200000000001</datacash_reference>
  <merchantreference>4900200000000001</merchantreference>
  <mode>TEST</mode>
  <reason>CANCELLED OK</reason>
  <status>1</status>
  <time>1151567456</time>
</Response>
```

Example XML Response for a successful fulfil

```
<Response>
  <datacash_reference>3900200000000001</datacash_reference>
  <merchantreference>3900200000000001</merchantreference>
  <mode>LIVE</mode>
  <reason>FULFILLED OK</reason>
  <status>1</status>
  <time>1071567356</time>
</Response>
```

Example XML for a successful txn_refund

```
<Response>
  <datacash_reference>400000088889999</datacash_reference>
  <HistoricTxn>
    <authcode>896876</authcode>
  </HistoricTxn>
  <merchantreference>41000008888888</merchantreference>
  <mode>LIVE</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1071567375</time>
</Response>
```

Example XML Response for a successful authorize_referral_request

```
<Response>
  <datacash_reference>3900200000000001</datacash_reference>
  <merchantreference>3900200000000001</merchantreference>
  <mode>LIVE</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1071567356</time>
</Response>
```

2.2.1.4.2.1 Acquirer Specific Notes

As mentioned in the Acquirer Specific Notes (section 2.2.1.4.1) for the XML request, the following is returned as a response to a Chase PaymentTech cancellation where the method attribute `reversal="true"` was supplied:

Example XML Response for a successfully reversed cancellation

```
<Response>
  <datacash_reference>4100900012345675</datacash_reference>
  <merchantreference>4100900012345675</merchantreference>
  <mode>TEST</mode>
  <reason reversal="success">CANCELLED OK</reason>
  <status>1</status>
  <time>1071567390</time>
</Response>
```

This additional attribute indicates the results of the online reversal. This could be 'success', 'failed' or 'not attempted'.

2.2.3 Additional Data (American Express ISO8583)

The following complex elements are specific to merchants subscribed to the American Express ISO8583 service.

Element Name:	AdditionalData
Position:	Request.TxnDetails

Element Name	Description
ITD	<p>This element should be provided for Ecommerce transactions (See section 2.2.3.1).</p> <p>This block cannot be used in conjunction with the IAC element (Ecommerce Airline Transactions). In the case where your transaction involves the sale of airline tickets, just send the IAC and not the ITD.</p>
IAC	<p>This element should be provided for Ecommerce Airline transactions. It should be populated in addition to the standard Request.Transaction.TxnDetails.AirlinesDetails elements. See section 2.2.3.2</p>
AAV	<p>This element can be provided in addition to the standard Cv2Avs element. It allows additional Cv2Avs data to be sent. See section 2.2.3.3</p>

2.2.3.1 ITD

This block should only be sent for standard Ecommerce transactions that **do not** involve the sale of Airline Tickets. If the transaction involves the sale of Airline tickets, send the IAC block instead.

Element Name: ITD
 Position: Request.TxnDetails.AdditionalData

Element Name	Description	Values / Limitations
customer_email	The customer's E-Mail address. Example: customer@example.com	Valid E-Mail format, Max 60 Bytes
customer_hostname	The customer's hostname.	Valid FQDN format, Max 60 Bytes
http_browser_type	The customers browser type.	Max 60 Bytes
ship_to_country	The country code for the shipping destination.	Max 3 Bytes
shipping_method	The shipping method used to deliver the product or service associated with the transaction.	<ul style="list-style-type: none"> • 01 - Same day • 02 - Overnight / next day • 03 - Priority 2-3 days • 04 - Ground, 4 or more days • 05 - Electronic delivery • 06 - ZZ - Reserved
MerchantProductSKU	See section 2.2.3.1.1	Up to 5 sub-elements can be supplied in this block.
customer_ip	The customer's IP address.	Valid IP Address Format, Max 15 Bytes
customer_ani	ANI (Automatic Number Identification) specified phone number that the customer used to place the order. Example: 6025551212	Numeric, Max 10 Bytes
customer_ii_digits	Telephone company-provided ANI information identifier (II) digits associated with customer_ani field above. II digits indicate the call type. For example, cellular (61-63), payphone (27), toll free (24, 25) etc.	Numeric, Max 2 Bytes

2.2.3.1.1 Merchant Product SKU

Element Name:	MerchantProductSku
Position:	Request.TxnDetails.AdditionalData.ITD

Element Name	Description	Values / Limitations
sku	Unique SKU (Stock Keeping Unit) inventory reference number of product associated with this authorization request. For multiple items, you should provide the most expensive item first. Up to 5 elements can be supplied.	Alphanumeric & Special Characters, Max 15 Bytes.

Example XML for a MerchantProductSku Element

```
...
<ITD>
...
<MerchantProductSku>
<sku>AB1</sku>
<sku>EH5</sku>
</MerchantProductSku>
...
</ITD>
...
```

2.2.3.2 IAC

This block should be sent for Ecommerce Airline transactions where the sale of airline tickets is involved. If there are no airline tickets involved, send the ITD block instead.

Element Name: IAC
Position: Request.TxnDetails.AdditionalData

Element Name	Description	Values / Limitations
departure_date	The flight departure date.	YYYYMMDD, Max 8 Bytes
PassengerName	This element contains the passengers name and other details. See section 2.2.3.2.1	
origin	The 3 character IATA airport code for the departure airport.	Max 5 Bytes
dest	The 3 character IATA airport code for the destination airport.	Max 5 Bytes
RoutingCities	Up to 10 city sub-elements can be supplied in this block. See section 2.2.3.2.2	
AirlineCarriers	Up to 9 airline sub-elements can be supplied in this block. See section 2.2.3.2.3	
farebasis	Primary & secondary discount codes indicate class of service and fare level associated with ticket. Truncate at 24 bytes, if necessary.	Max 24 Bytes
number_of_passengers	The number of passengers travelling.	Numeric, 3 Bytes
customer_ip	The customer's IP address.	Valid IP Address Format, Max 15 Bytes
customer_email	The customer's E-Mail address.	Valid E-Mail format, Max 60 Bytes

2.2.3.2.1 Passenger Name

Element Name: PassengerName
Position: Request.TxnDetails.AdditionalData.IAC

Element Name	Description	Values / Limitations
Surname	The customer's surname.	Max 40 Bytes
Firstname	The customer's firstname	Max 40 Bytes
middle_initial	The customer's middle initial(s)	Max 40 Bytes
title	The customers honorific.	Max 40 Bytes

Example XML for a PassengerName Element

```
...
<IAC>
  ...
  <PassengerName>
    <surname>Bloggs</surname>
    <firstname>Fred</firstname>
    <middle_initial>J</middle_initial>
    <title>Mr</title>
  </PassengerName>
  ...
</IAC>
...
```

2.2.3.2.2 Routing Cities

Element Name: RoutingCities
Position: Request.TxnDetails.AdditionalData.IAC

Element Name	Description	Values / Limitations
city	Up to 10 city sub-elements can be supplied in this block. The city elements should contain the Routing Airport or City Codes for each leg on the ticket (including the origin and dest).	Max 5 Bytes per sub-element

Example XML for a RoutingCities Element

```
...
<IAC>
  ...
  <RoutingCities>
    <city>SIG</city>
    <city>SYD</city>
  </RoutingCities>
  ...
</IAC>
...
```

2.2.3.2.3 Airline Carriers

Element Name: AirlineCarriers
Position: Request.TxnDetails.AdditionalData.IAC

Element Name	Description	Values / Limitations
airline	Up to 9 airline sub-elements can be supplied in this block. The Airline Carrier Code for each leg on the ticket should be provided (including origin and dest). Note that each leg must have an entry, even if multiple or all legs are on the same airline.	2 Character ICAO Airline Designator, Max 5 Bytes

Example XML for a AirlineCarriers Element

```
...
<IAC>
  ...
  <AirlineCarriers>
    <airline>AB</airline>
    <airline>CD</airline>
    <airline>EF</airline>
  </AirlineCarriers>
  ...
</IAC>
...
```

2.2.3.3 AAV

This should be provided in addition to the standard Cv2Avs block.

Element Name	Description	Values / Limitations
cm_billing_postal_code	The Card Member billing postal code.	Max 9 Bytes
cm_billing_street_address	The Card Member billing street address.	Max 20 Bytes
cm_first_name	The Card Member first name.	Max 15 Bytes
cm_last_name	The Card Member surname.	Max 30 Bytes
cm_phone_number	The Card Member telephone number	Max 10 Bytes
ship_to_postal_code	The shipping postal code.	Max 9 Bytes
ship_to_address	The shipping address.	Max 50 Bytes
ship_to_first_name	The shipping first name.	Max 15 Bytes
ship_to_last_name	The shipping last name.	Max 30 Bytes
ship_to_phone_number	The shipping phone number.	Max 10 Bytes
ship_to_country_code	The shipping destination country code.	Max 3 Bytes

2.2.3.4 Example XML

Please note that the following snippet is an example. In reality the ITD and IAC elements would never be submitted in the same request. The ITD block should be sent for standard Ecommerce transactions, whereas the IAC block should be sent instead of the ITD for Airline Ecommerce transactions.

Partial XML Request showing the AdditionalData Elements.

```

<Request>
  ...
  <TxnDetails>
    ...
    <AdditionalData>
      <ITD>
        <customer_email>example@example.com</customer_email>
        <customer_hostname>example.com</customer_hostname>
        <http_browser_type>Mozilla /4,0</http_browser_type>
        <ship_to_country>840</ship_to_country>
        <shipping_method />
        <MerchantProductSku>
          <sku>AB1</sku>
          <sku>EH5</sku>
        </MerchantProductSku>
        <customer_ip>127.0.0.1</customer_ip>
        <customer_ani>441315388400</customer_ani>
        <customer_iid_digits>27</customer_iid_digits>
      </ITD>
      <IAC>
        <departure_date>20110231</departure_date>
        <PassengerName>
          <surname>Bloggs</surname>
          <firstname>Fred</firstname>
          <middle_initial>J</middle_initial>
          <title>Mr</title>
        </PassengerName>
        <origin>EDI</origin>
        <dest>AMS</dest>
        <RoutingCities>
          <city>SIG</city>
          <city>SYD</city>
        </RoutingCities>
        <AirlineCarriers>
          <airline>AB</airline>
          <airline>CD</airline>
          <airline>EF</airline>
        </AirlineCarriers>
        <farebasis>ABC12345</farebasis>
        <number_of_passengers>2</number_of_passengers>
        <customer_ip>192.168.1.1</customer_ip>
        <customer_email>customer@example.com</customer_email>
      </IAC>
      <AAV>
        <cm_billing_postal_code>EH6 7EZ</cm_billing_postal_code>
        <cm_billing_street_address>14, Links
        Place</cm_billing_street_address>
        <cm_first_name>Joe</cm_first_name>
    
```

```

<cm_last_name>Bloggs</cm_last_name>
<cm_phone_number>441315388400</cm_phone_number>
    <ship_to_postal_code>7446</ship_to_postal_code>
    <ship_to_address>No 1 Waterview</ship_to_address>
    <ship_to_first_name>Fred</ship_to_first_name>
    <ship_to_last_name>Bloggs</ship_to_last_name>
    <ship_to_phone_number>27215284500</ship_to_phone_number>
    <ship_to_country_code>710</ship_to_country_code>
</AAV>
</AdditionalData>
...
</TxnDetails>
...
</Request>
```

2.2.4 *Misc Customer Info*

Misc Customer Info can be provided with the majority of transaction types. To use this service it must be configured on your account by DataCash support.

The following complex element can be sent in the request XML and will be captured and stored alongside the transaction by DataCash.

Element Name: MiscCustomerInfo
Position: Request.TxnDetails

Element Name	Description	Values / Limitations
firstname	The customer's first name.	Max 50 Characters.
lastname	The customer's last name / surname.	Max 50 Characters.
email	The customer's E-Mail address. Example: customer@example.com	Valid E-Mail format. Max 255 Characters.
language	The country code for the shipping destination.	Max 20 Characters.
country	The customers country.	Max 50 Characters.

2.2.5 *Line Item Detail*

A technical introduction to this Service is available on the website:

<http://www.datacash.com/gettingproducts.php?id=Corporate-Purchasing->

This service is utilised by sending a normal Credit and Debit Card Service Request with additional information. This section of documentation assumes the Credit and Debit Card Service has already been integrated and the reader is familiar with it. The Credit and Debit Card Service is described in section 2.2.1.

Please bear in mind that the `amount` field must be reconciled with the line items. The `amount` field should contain the gross amount, after tax and (where appropriate) shipping and discounts. In order to compensate for rounding errors, a tolerance of one minor currency unit per line item element is allowed.

When using the Line Item Detail Service, the transaction methods `auth`, `pre`, `refund`, `erp` and `txn_refund` can all be submitted. However, if you are using the `pre` and `erp` methods, the full amount must be fulfilled – partial fulfills cannot be used. The `txn_refund` method can also be used, providing the full value of the original transaction is refunded in one step. If a partial fulfill, or `txn_refund` is performed, the amount would not reconcile with the line items.

2.2.5.1 Schema Elements for Request

In this section, the fields that can be presented with Line Item Detail transactions will be presented along with example XML for those fields.

To process Visa or American Express Corporate Purchasing Cards, one can make use of the `LineItemDetail` schema element:

- Request
 - Authentication – section 2.1.1.2
 - Transaction
 - TxnDetails
 - LineItemDetail – information relating to the whole order, section 2.2.5.1.5
 - Shipping – details about the delivery, section 2.2.5.1.2
 - AddressDetail – details about the delivery address, section 2.2.5.1.1
 - Items – allows detail of each item to be passed, section 2.2.5.1.3
 - Item – detailed information about a single item, section 2.2.5.1.4

As the fields which can be provided depend upon the Acquiring Bank you are using, each field will be labelled with the following key:

- O - Optional
- R - Required, field must be presented
- X - Excluded, presenting this field will cause the transaction to fail
- M - Mandatory if available; if the information is available, it should be presented

2.2.5.1.1 AddressDetail

The fields within this element can only be presented for holders of American Express cards when the cardholder is billed in a different country from the supplier (a cross border transaction). If you are submitting such a transaction, it is recommended that these fields are provided.

Submitting this information for a non-American Express transaction, or an American Express non-cross border transaction will cause the transaction to be rejected.

Element Name	AddressDetail
Position	Request.Transaction.TxnDetails.LineItemDetail.Shipping
Children	Yes
Attributes	Yes

Elements of AddressDetail					
Attribute Name	Description	Values/Limitations	Amex	Amex, cross border	NWS
side	Designates whether the source or destination address	"source" or "destination"	X	R	X

Elements of AddressDetail					
Element Name	Description	Values/Limitations	Amex	Amex, cross border	NWS
address	A single line of the address. Between one and four such lines may be submitted, distinguished by the line attribute	Alphanumeric, up to 40 characters. line attribute should be a single digit between 1 and 4	X	R	X
postcode	The postal code for the address	Alphanumeric, up to 10 characters	X	R	X
country	ISO numeric code for the address	Numeric, 3-digit ISO code. For example, UK=826, France=250 etc.	X	R	X

XML Example element for AddressDetail

```

<AddressDetail side="source">
  <address line="1">My Company</address>
  <address line="2">My Street</address>
  <address line="3">Mytown</address>
  <postcode>MP1 1AB</postcode>
  <country>826</country>
</AddressDetail>

<AddressDetail side="destination">
  <address line="1">Ma Maison</address>
  <address line="2">Ma Rue</address>
  <postcode>75123</postcode>
  <country>250</country>
</AddressDetail>

```

2.2.5.1.2 Shipping

Optional shipping information may be submitted for a CPC transaction. In the case of American Express, whilst this information may be submitted, it is not used by their systems at present (with the exception of the AddressDetail information discussed above). In particular, shipping should *not* be used in invoice reconciliation for American Express transactions.

Element Name	Shipping
Position	Request.Transaction.TxnDetails.LineItemDetail
Children	Yes
Attributes	No

Attributes of Shipping					
Element Name	Description	Values / Limitations	Amex	NWS	
AddressDetail	See section 2.2.5.1.1				
destinationcountrycode	Country code of the destination	ISO Alphabetic 3 characters	O	O	
destinationpostalcode	Postcode to send to	Alphanumeric up to 10 characters	O	O	
shipfrompostalcode	Postcode of the source	Alphanumeric up to 10 characters	O	O	
shippingamount	Net (pre-tax) cost of shipping	Numeric	O	O	
shippingVATrate	Percentage rate at which shipping is taxed	Numeric	O	O	

XML Example elements for Shipping

```

<Shipping>
  <destinationcountrycode>GBR</destinationcountrycode>
  <destinationpostalcode>EH1 1AB</destinationpostalcode>
  <shipfrompostalcode>SW19</shipfrompostalcode>
  <shippingamount>5.95</shippingamount>
  <shippingVATrate>17.5</shippingVATrate>
</Shipping>

```

2.2.5.1.3 Items

Element Name	Items
Position	Request.Transaction.TxnDetails.LineItemDetail
Children	Yes
Attributes	No

Elements of Items		
Element Name	Description	Values/Limitations
Item	An individual line item, as described below	At least one item element must be submitted per transaction

Example XML is shown in section 2.2.5.1.4.

2.2.5.1.4 Item

Details of the individual line items are submitted here.

Element Name	Item
Position	Request.Transaction.TxnDetails.LineItemDetail.Items
Children	Yes
Attributes	No

Attributes of Item			
Element Name	Description	Required	Values/Limitations
commoditycode	The commodity code for this purchase	N	Four-digit commodity code. See Appendix 1 for a list.
description	Description of the item	Y	Alphanumeric up to 26 characters (Barclays, Natwest), or 40 characters (Amex).
discountamount	The item-level discount for this item	N	Numeric. Not supported by American Express
product_code	Merchant's product code	N	Alphanumeric up to 12 characters
quantity	The number of units	Y	Numeric
totalamount	The total Net (pre-VAT) cost of the items	Y	Numeric. Must reconcile with quantity, unitprice and (where appropriate) discountamount
unitcost	The amount for a single unit	Y	Numeric
unitmeasure	The units in which the items are counted	Y	Alphanumeric up to 12 characters
vatrate	The item-level VAT rate for this item	Y	Numeric

XML Example element for Items and Item

```

<Items>
  <Item>
    <commoditycode>4800</commoditycode>
    <description>Ring Binder</description>
    <unitmeasure>Box</unitmeasure>
    <unitprice>9.99</unitprice>
    <vatrate>17.5</vatrate>
    <quantity>1</quantity>
    <totalamount>9.99</totalamount>
  </Item>
  <Item>
    <commoditycode>4800</commoditycode>
    <description>Photocopier paper</description>
    <unitmeasure>Ream</unitmeasure>
    <unitcost>5.00</unitcost>
    <vatrate>17.5</vatrate>
    <quantity>100</quantity>
    <totalamount>500</totalamount>
  </Item>
</Items>

```

2.2.5.1.5 LineItemDetail

This is where information relating to the order as a whole is submitted. Since each acquiring bank has different requirements, this information is listed with the following key:

Element Name	LineItemDetail
Position	Request.Transaction.TxnDetails
Children	Y
Attributes	N

Elements of LineItemDetail				
Element Name	Description	Values/Limitations	NWS	Amex
customercode	Customer-supplied reference field	Alphanumeric, up to 16 characters	R	O
customernumber	Customer-supplied reference field	Alphanumeric, up to 17 characters	N	O
customerref1	Customer-supplied reference field	Alphanumeric, up to 20 characters	N	O
customerref2	Customer-supplied reference field	Alphanumeric, up to 20 characters	N	O
customerVATnumber	The customer's VAT number	Alphanumeric, up to 13 characters	O	O ¹

discountamount	Transaction-level discount amount	Numeric	O	O ²
merchantVATnumber	The merchant's VAT number	Alphanumeric, up to 13 characters	O	R
originalinvoicenumber	The invoice number of the original transaction (for refunds)	Alphanumeric, up to 12 characters	O	O
originalorderdate	The date on which the order was placed	YYMMDD	O	O
supplierorderreference	The supplier's reference	Alphanumeric, up to 12 characters	O	N
transactionVAT	The total VAT amount for the transaction	Numeric. Must reconcile with the individual amounts and VAT rates of the items	R	R
transactionVATstatus	Should always be '1'	'1'	R	R

Notes:

- ¹ For American Express transactions, the customer VAT number is required for cross border transactions, and any transaction made by a Belgian merchant.
- ² Whilst the discountamount field may be submitted to American Express, it is currently ignored by their system. In particular, it must *not* be used in invoice reconciliation.

2.2.5.2 XML Example Request

XML Example Transaction for an order with two items

```

<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <CardTxn>
      <method>auth</method>
      <Card>
        <pan>5700*****0001</pan>
        <expirydate>10/05</expirydate>
      </Card>
    </CardTxn>
    <TxnDetails>
      <merchantreference>12345601</merchantreference>
      <amount>600.28</amount>
      <LineItemDetail>
        <customercode>CustCode123</customercode>
        <transactionVAT>90.29</transactionVAT>
        <transactionVATstatus>1</transactionVATstatus>
        <merchantVATnumber>7E6G415</merchantVATnumber>
        <Shipping>
          <shippingamount>5.99</shippingamount>
          <shippingVATrate>17.5</shippingVATrate>
        </Shipping>
        <Items>
          <Item>
            <commoditycode>4800</commoditycode>
            <description>Ring Binder</description>
            <unitmeasure>Box</unitmeasure>
            <unitprice>9.99</unitprice>
            <vatrate>17.5</vatrate>
            <quantity>1</quantity>
            <totalamount>9.99</totalamount>
          </Item>
          <Item>
            <commoditycode>4800</commoditycode>
            <description>Photocopier paper</description>
            <unitmeasure>Ream</unitmeasure>
            <unitprice>5</unitprice>
            <vatrate>17.5</vatrate>
            <quantity>100</quantity>
            <totalamount>500</totalamount>
          </Item>
        </Items>
      </LineItemDetail>
    </TxnDetails>
  </Transaction>
</Request>

```

2.2.6 Airlines Transaction Records

This service is utilised by sending additional information in an otherwise normal Credit and Debit Card transaction. This section assumes that facility for processing Credit and Debit Card transactions has already been integrated and the reader is familiar with it. Details about the Credit and Debit Card Service are available in section 2.2.1

The DPG will also accept Airline data in a `fulfill` transaction when using the two-stage transaction model. The location of the data in the request XML remains the same. At present this only applies to Airline data.

This service is currently available for merchants using The Royal Bank of Scotland Group (inc Natwest Streamline, Natwest IMS, Ulster Bank, Clydesdale Bank, Yorkshire Bank) , Barclaycard Business and Omnipay.

2.2.6.1 Schema Elements for Request

In this section, the fields associated with Airlines transactions will be presented along with example XML for those fields.

To submit flight itinerary information, one should use the `AirlinesDetails` container element:

- Request
 - Authentication – section 2.1.1.1.1
 - Transaction
 - TxnDetails
 - AirlinesDetails – information relating to the whole order, section 2.2.6.1.1
 - Passenger – details about an individual passenger, section 2.2.6.4
 - FlightLeg – allows detail of each leg of the journey, section 2.2.6.4.1

In the following tables, each field will be labelled with the following key:

- *O* - Optional
- *R* - Required, field must be presented

2.2.6.1.1 AirlinesDetails

The `AirlinesDetails` elements contains information relating to the whole order.

Element Name	AirlinesDetails
Position	Request.Transaction.TxnDetails
Children	Yes
Attributes	No

2.2.6.3

Elements of AirlinesDetails			
Element Name	Description	Values/Limitations	Required
agencynname	The POS agency name	Alphanumeric, up to 26 characters (for submission to DPG). For BMS merchants: the first 25 characters will be used at settlement	R
agencycode	The POS IATA number	Numeric up to 8 characters	R
airlineplaninvoice	The airline plan invoice	Alpha numeric, up to 6 characters	O
airlineplannumber	The airline plan number	Alphanumeric, up to 2 characters	O
departurepoint	The departure point	Three-character origination code	R
Passenger	See Section 2.2.6.4		
FlightLeg	See Section 2.2.6.4.1		

XML Example elements for AirlinesDetails

```
<AirlinesDetails>
    <agencynname>MyAirlinesAgency</agencynname>
    <agencycode>24681357</agencycode>
    <departurepoint>EDI</departurepoint>
    <Passenger>...</Passenger>
    <Passenger>...</Passenger>
    <FlightLeg number="1">...</FlightLeg>
    <FlightLeg number="2">...</FlightLeg>
    <FlightLeg number="3">...</FlightLeg>
</AirlinesDetails>

<AirlinesDetails>
    <agencynname>MyAirlinesAgency</agencynname>
    <agencycode>12345612</agencycode>
    <airplaninvoice>1234AN</airplaninvoice>
    <airplannumber>12</airplannumber>
    <departurepoint>LON</departurepoint>
    <Passenger>...</Passenger>
    <FlightLeg number="1">...</FlightLeg>
</AirlinesDetails>
```

2.2.6.4 Passenger

There must be at least one passenger per transaction. Where there is more than one person travelling, supplying multiple Passenger elements allows the entire transaction to be authorised at once, whilst maintaining the individual invoice information for each passenger. The sum of ticket prices for each passenger should match the amount field that forms part of the normal Credit and Debit Card information. This element may repeat as often as necessary.

Element Name	Passenger
Position	Request.Transaction.TxnDetails.AirlinesDetails
Children	Yes
Attributes	No

Elements of Passenger			
Element Name	Description	Values/Limitations	Required
passengernname	The passenger's name	Alphanumeric, up to 25 characters	R
passengerref	A unique reference for the passenger	Alphanumeric, up to 17 characters for submission to DPG. For BMS merchants: the first 12 characters will be used at settlement	R
ticketnumber	Ticket number for the passenger. This may optionally be different for each passenger	14 characters. This usually consists of a 3-character prefix, 10 characters of ticket number and a single check digit at the end. The prefix characters and check digit may optionally be replaced by '0'	R
ticketprice	The ticket price for the individual passenger.		R

XML Example element for Passenger

```
<Passenger>
  <passengername>Joe Bloggs</passengername>
  <passengerref>12345678</passengerref>
  <ticketnumber>00024680135790</ticketnumber>
  <ticketprice>199.99</ticketprice>
</Passenger>
```

2.2.6.4.1 FlightLeg

Each transaction may contain details of up to four flight legs. These are each specified in their own FlightLeg container element. For a multi-flight leg journey: the point of departure for the first leg will be set to the value supplied in AirlinesDetails.departurepoint; the point of departure for subsequent legs will be taken to be the destination from the previous leg.

Element Name	FlightLeg
Position	Request.Transaction.TxnDetailsAirlinesDetails
Children	Yes
Attributes	Yes

Attributes of FlightLeg			
Attribute Name	Description	Values/Limitations	Required
number	The stage of the journey	1, 2, 3 or 4	R

Elements of FlightLeg			
Element Name	Description	Values/Limitations	Required
carriercode	Carrier code for this leg	Two character carrier code	R
class	Service class of this leg	Single alphanumeric character	R
departtax	Depature tax	Numeric, up to 12 digits including decimal point	O
departuredate	Date of departure for this leg	Date, as YYYYMMDD	R
destination	The destination for this leg	Three-character destination code	R
farebasiscode	The fare basis code	Alphanumeric, up to 6 charcters	O
flight_number	Flight number for this leg	Alphanumeric, up to 4 characters	O
stopovercode	The stop-over code	Alphanumeric, 1 character	O

XML Example elements for FlightLeg

```

<FlightLeg number="1">
  <carriercode>BA</carriercode>
  <class>1</class>
  <departuredate>20040730</departuredate>
  <destination>LHA</destination>
  <flight_number>1234</flight_number>
</FlightLeg>

<FlightLeg number="1">
  <carriercode>BA</carriercode>
  <class>1</class>
  <departtax>12.99</departtax>
  <departuredate>20040730</departuredate>
  <destination>LHA</destination>
  <farebasiscode>1223AB</farebasiscode>
  <flight_number>1234</flight_number>
</FlightLeg>

```

2.2.6.4.2 ISOAirline

The ISOAirline elements contains information relating to the whole order.

Element Name	ISOAirline
Position	Request.Transaction.TxnDetails
Children	Yes
Attributes	No

Elements of AirlinesDetails			
Element Name	Description	Values/Limitations	Required
issuingcarrier	2- or 3-character IATA assigned airline designator.	Alphanumeric	R
customercode	Cardholder supplied code	Alphanumeric, up to 27 characters	R
issuedate	Date of issue	YYMMDD	O
totalfees	Total fees		R
travelagencynname	Name of agency	Alphanumeric	R
ticketnumber	Ticket number	Alphanumeric, up to 15 characters	R

travelagencycode	IATA-assigned Agency code (7 digits plus check digit).		O
totaltaxes	Total taxes		R
totalfare	Total fare		R
pnr		Alphanumeric, up to 6 characters	R
ISOAirlinePassenger	See Section 2.2.6.4.3		
ISOAirlineLeg	See Section 2.2.6.4.1		

XML Example elements for ISOAirline

```

<ISOAirline>
    <issuingcarrier>12</issuingcarrier>
    <customercode>ABCD1234</customercode>
    <issuedate>110101</issuedate>
    <totalfees>1000.00</totalfees>
    <travelagencynname>MyAgency</travelagencynname>
    <ticketnumber>1234ABCD</ticketnumber>
    <travelagencycode>12345678</travelagencycode>
    <totalfare>800.00</totalfare>
    <totaltaxes>200.00</totaltaxes>
    <pnr>123456</pnr>
    <ISOAirlinePassenger>...</ISOAirlinePassenger>
    <ISOAirlineLeg>...</ISOAirlineLeg>
</ISOAirline>

```

2.2.6.4.3 ISOAirlinePassenger

Only details of one passenger may be provided in a transaction with ISO Airline data.

Element Name	ISOAirlinePassenger
Position	Request.Transaction.TxnDetails.ISOAIRLINE
Children	Yes

Attributes	No
------------	----

Elements of Passenger			
Element Name	Description	Values/Limitations	Required
passengername	The passenger's name	Alphanumeric, up to 25 characters	R

XML Example element for ISOAirlinePassenger

```
<ISOAirlinePassenger>
  <passengername>Joe Bloggs</passengername>
</ISOAirlinePassenger>
```

2.2.6.4.4 ISOAirlineLeg

Each transaction may contain details of up to four flight legs. These are each specified in their own `ISOAirlineLeg` container element. For a multi-flight leg journey, the point of departure for the first leg will be set to the value supplied in `ISOAirline.departurepoint` and the point of departure for subsequent legs will be taken to be the destination from the previous leg.

Element Name	<code>ISOAirlineLeg</code>
Position	<code>Request.Transaction.TxnDetails.ISOAirline</code>
Children	Yes
Attributes	Yes

Attributes of FlightLeg			
Attribute Name	Description	Values/Limitations	Required
number	The stage of the journey	1, 2, 3 or 4	R

Elements of FlightLeg			
Element Name	Description	Values/Limitations	Required
conjunctionticket		Alphanumeric, up to 15 characters	R
exchangeticket		Alphanumeric, up to 15 characters	R
couponnumber		Alphanumeric, single characters	O
fare			R
taxes			R
fees			R
endorsement	Endorsement/Restriction	Alphanumeric, up to 20 characters	R
departuredate		YYMMDD	O
departuretime		HHMM	O
originairport	3-character IATA location identifier.		R
arrivaldate		YYMMDD	O
arrivaltime		HHMM	O
destinationairpor	3-character IATA location identifier.		R

carriercode	2- or 3-character IATA assigned airline designator.		R
stopovercode		Single alphanumeric character	O
farebasiscode		Alphanumeric, up to 15 characters	R
flightnumber		Alphanumeric, up to 5 characters	R

XML Example elements for ISOAirlineLeg

```

<ISOAirlineLeg number="1">
  <conjunctionticket>1234ABCD</conjunctionticket>
  <exchangeticket>2345BCDE</exchangeticket>
  <couponnumber>1</couponnumber>
  <fare>400.00</fare>
  <taxes>100.00</taxes>
  <fees>1.00</fees>
  <endorsement>A1BC23</endorsement>
  <departuredate>110101</departuredate>
  <departuretime>0900</departuretime>
  <originairport>EDI</originairport>
  <arrivaldate>110101</arrivaldate>
  <arrivaltime>1200</arrivaltime>
  <destinationairport>JFK</destinationairport>
  <carriercode>BA</carriercode>
  <serviceclass>1A</serviceclass>
  <farebasiscode>BA</farebasiscode>
  <flightnumber>BA</flightnumber>
</ISOAirlineLeg>

```

2.2.6.5 XML Example Requests

XML Example Transaction for a ticket purchase for multi-leg, single-passenger flight

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <CardTxn>
      <method>auth</method>
      <Card>
        <pan>5700*****0001</pan>
        <expirydate>10/05</expirydate>
      </Card>
    </CardTxn>
    <TxnDetails>
      <merchantreference>2345679</merchantreference>
      <amount>1002.00</amount>
      <ISOAirline>
        <issuingcarrier>12</issuingcarrier>
        <customercode>ABCD1234</customercode>
        <issuedate>110101</issuedate>
        <totalfees>2.00</totalfees>
        <travelagencynname>MyAgency</travelagencynname>
        <ticketnumber>1234ABCD</ticketnumber>
        <travelagencycode>12345678</travelagencycode>
        <totalfare>800.00</totalfare>
        <totaltaxes>200.00</totaltaxes>
        <pnr>123456</pnr>
        <ISOAirlinePassenger>
          <passengername>Joe Bloggs</passengername>
        </ISOAirlinePassenger>
        <ISOAirlineLeg number="1">
          <conjunctionticket>1234ABCD</conjunctionticket>
          <exchangeticket>2345BCDE</exchangeticket>
          <couponnuber>1</couponnuber>
          <fare>400.00</fare>
          <taxes>100.00</taxes>
          <fees>1.00</fees>
          <endorsement>A1BC23</endorsement>
          <departuredate>110101</departuredate>
          <departuretime>0900</departuretime>
        </ISOAirlineLeg>
      </ISOAirline>
    </TxnDetails>
  </Transaction>
</Request>
```

```
<originairport>EDI</originairport>
<arrivaldate>110101</arrivaldate>
<arrivaltime>1200</arrivaltime>
<destinationairport>JFK</destinationairport>
<carriercode>BA</carriercode>
<serviceclass>1A</serviceclass>
<farebasiscode>BA</farebasiscode>
<flightnumber>BA</flightnumber>
</ISOAirlineLeg>
<ISOAirlineLeg number="1">
<conjunctionticket>3456ABCD</conjunctionticket>
<exchangeticket>4567BCDE</exchangeticket>
<couponnuber>1</couponnumber>
<fare>400.00</fare>
<taxes>100.00</taxes>
<fees>1.00</fees>
<endorsement>A1BC23</endorsement>
<departuredate>110114</departuredate>
<departuretime>1000</departuretime>
<originairport>JFK</originairport>
<arrivaldate>110101</arrivaldate>
<arrivaltime>1230</arrivaltime>
<destinationairport>EDI</destinationairport>
<carriercode>BA</carriercode>
<serviceclass>1A</serviceclass>
<farebasiscode>BA</farebasiscode>
<flightnumber>BA</flightnumber>
</ISOAirlineLeg>
</ISOAirline>
</TxnDetails>
</Transaction>
</Request>
```

XML Example Transaction for a ticket purchase for multi-leg, multi-passenger flight

```

<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <CardTxn>
      <method>auth</method>
      <Card>
        <pan>5700*****0001</pan>
        <expirydate>10/05</expirydate>
      </Card>
    </CardTxn>
    <TxnDetails>
      <merchantreference>2345679</merchantreference>
      <amount>150.00</amount>
      <AirlinesDetails>
        <agencyname>MyAirlinesAgency</agencyname>
        <agencycode>24681357</agencycode>
        <departurepoint>EDI</departurepoint>
        <Passenger>
          <passengername>Joe Bloggs</passengername>
          <passengerref>12345602</passengerref>
          <ticketnumber>ABC09876543210</ticketnumber>
          <ticketprice>100.00</ticketprice>
        </Passenger>
        <Passenger>
          <passengername>Jane Bloggs</passengername>
          <passengerref>12345603</passengerref>
          <ticketnumber>ABC09876543210</ticketnumber>
          <ticketprice>50.00</ticketprice>
        </Passenger>
        <FlightLeg number="1">
          <carriercode>BA</carriercode>
          <destination>LHA</destination>
          <departuredate>20040730</departuredate>
          <class>1</class>
        </FlightLeg>
        <FlightLeg number="2">
          <carriercode>AB</carriercode>
          <destination>JFK</destination>
          <departuredate>20040731</departuredate>
          <class>1</class>
        </FlightLeg>
      </AirlinesDetails>
    </TxnDetails>
  </Transaction>
</Request>

```

2.2.6.6 XML Example Requests

XML Example Transaction supplying data in a “fulfill” transaction

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <reference>4100200039275407</reference>
      <method>fulfill</method>
    </HistoricTxn>
    <TxnDetails>
      <AirlinesDetails>
        <agencyname>MyAirlinesAgency</agencyname>
        <agencycode>24681357</agencycode>
        <departurepoint>EDI</departurepoint>
        <Passenger>
          <passengername>Joe Bloggs</passengername>
          <passengerref>12345601</passengerref>
          <ticketnumber>ABC09876543210</ticketnumber>
          <ticketprice>100.00</ticketprice>
        </Passenger>
        <FlightLeg number="1">
          <carriercode>BA</carriercode>
          <destination>LHA</destination>
          <departureday>20040730</departureday>
          <class>1</class>
        </FlightLeg>
      </AirlinesDetails>
    </TxnDetails>
  </Transaction>
</Request>
```

2.2.7 *Merchant Narrative*

The Merchant Narrative Service allows merchants to specify the narrative presented on the cardholders statement on a per transaction basis, rather than basing the narrative on static data held by the acquiring bank.

2.2.7.1 Schema Elements for Request

Element Name: MerchantNarrative
Position:
Request.Transaction.CardTxn Request.Transaction.HistoricTxn

Elements of MerchantNarrative			
Element Name	Description	Values/Limitations	Required
narrative1	Merchant Narrative element 1	Maximum 26 characters. The following characters are permitted: Alpha numerics, !"%&'()*+,.-./:;<=>? Spaces permitted however leading and trailing whitespace will be trimmed	M
narrative2	Merchant Narrative element 2	Maximum 26 characters. The following characters are permitted: Alpha numerics, !"%&'()*+,.-./:;<=>? Spaces permitted however leading and trailing whitespace will be trimmed	M

Either narrative1, narrative2, or both can be present in the parent MerchantNarrative element, however an empty MerchantNarrative element will not be permitted.

The MerchantNarrative element can only be supplied for the following Bank Card transaction methods:

- auth
- refund
- txn_refund
- pre

- erp

The Merchant Narrative service is not compatible with the Airlines Transaction Record service.

2.2.7.2 XML Example Request

Example Merchant Narrative Request

```
<Request>

    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>

    <Transaction>
        <TxnDetails>
            <merchantreference>12345601</merchantreference>
            <amount currency="GBP">1000.00</amount>
        </TxnDetails>

        <CardTxn>
            <method>auth</method>
            <Card>
                <pan>XXXXXXXXXXXXXXXXXX</pan>
                <expirydate>XX/XX</expirydate>
            </Card>
            <MerchantNarrative>
                <narrative1>line1</narrative1>
                <narrative2>line2</narrative2>
            </MerchantNarrative>
        </CardTxn>

    </Transaction>
</Request>
```

2.2.8 Fuel Card Service

The Fuel Card service allows merchants to trade with various fuel card acquirers.

Typically this will be via cardholder present terminals, and at the time of writing will involve keyed and swiped transactions only (The majority of fuel cards don't have chips so parked transactions are not supported at the time of writing).

Refunds onto fuel cards are currently unsupported.

2.2.8.1 Schema Elements for Request

Legend

- FV = fuel_validate
- FOAS = fuel_offline_auth_settle
- FOA = fuel_online_auth
- FS = fuel_settle

2.2.8.1.1 FuelCardTxn

For transactions using card details only. In addition to the basic Card details, the transaction type and authorisation code (if applicable) are sent in the `CardTxn` element:

Element Name: FuelCardTxn
 Position: Request.Transaction

Elements of FuelCardTxn						
Element Name	Description	Values/Limitations	FV	FOAS	FOA	FS
method	The transaction type	fuel_validate fuel_offline_auth_settle fuel_online_auth fuel_settle	R	R	R	R
Card	Used for keyed transactions only. See 2.6.1.1 in Card Holder Present section		M	M	M	-
card_details	Data read from the magnetic stripe of the card, or equivalent from the ICC	Must be Base64 encoded. Only used for swiped transactions	M	M	M	-
authcode	authorisation code received from the bank	If presented, must be value received from Banks Authorisation centre	-	M	O	R

receipt_no	The EFT transaction number	A one to six digit number. The values taken in FOAS / FS transactions will be used.	O	R	O	R
odometer_reading	Distance travelled by vehicle	The odometer reading from the vehicle used in the fuel transaction. Optional	O	O	O	-
registration_number	Registration of vehicle	The registration from the vehicle involved in the fuel transaction. Optional.	O	O	O	-
FuelProducts	Complex XML element containing details about the products ordered. See FuelProducts section 2.2.8.1.3.		O	O	O	-
ICC	See 2.6.1.1.4 in Card Holder Present section		M	M	M	-
Terminal	See 2.6.1.1.3 in Card Holder Present section		M	M	M	-

2.2.8.1.2 FuelProducts

This complex element can contain one or more FuelProduct elements.

Element Name: FuelProducts
 Position: Request.Transaction.FuelCardTxn

Elements of FuelProducts						
Element Name	Description	Values/Limitations	FV	FOAS	FOA	FS
FuelProduct	Complex XML element containing details about each product ordered. See section 2.2.8.1.3.		O	O	O	-

2.2.8.1.3 FuelProduct

This complex element contains information about a specific fuel product in a transaction.

Element Name: FuelProduct
Position: Request.Transaction.FuelCardTxn.FuelProducts

Elements of FuelProducts						
Element Name	Description	Values/Limitations	FV	FOAS	FOA	FS
prod_code	Sale code of the product ordered	2 alphanumeric characters	O	O	O	-
prod_quantity	Quantity of product purchased. (e.g. litres dispensed)	Decimal number	O	O	O	-
prod_value	Monetary value of product purchased	Decimal number. Not validated against overall transaction amount.	O	O	O	-

2.2.8.2 XML Request Examples

Example XML for a fuel_validate transaction (keyed)

```

<Request>
    <Authentication>
        <password>#####</password>
        <client>88000000</client>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">145.74</amount>
            <merchantreference>74927489274</merchantreference>
            <capturemethod>keyed</capturemethod>
        </TxnDetails>
        <FuelCardTxn>
            <method>fuel_validate</method>
            <Terminal id="88547856">
                <terminal_capabilities ic_reader="true"
magnetic_stripe_reader="true" manual_card_entry="true" />
                <features_capabilities pin_pad_available="true" />
            </Terminal>
            <ICC>
                <term_type>22</term_type>
            </ICC>
            <Card>
                <pan>7019000000#####</pan>
                <expirydate>12/15</expirydate>
            </Card>
            <receipt_no>4383</receipt_no>
            <odometer_reading>10000</odometer_reading>
            <registration_number>H702MOF</registration_number>
            <FuelProducts>
                <FuelProduct>
                    <prod_code>01</prod_code>
                    <prod_quantity>50.22</prod_quantity>
                    <prod_value>99.99</prod_value>
                </FuelProduct>
                <FuelProduct>
                    <prod_code>05</prod_code>
                    <prod_quantity>23.45</prod_quantity>
                    <prod_value>45.75</prod_value>
                </FuelProduct>
            </FuelProducts>
        </FuelCardTxn>
    </Transaction>
</Request>

```

Example XML for a fuel_offline_auth_settle transaction (keyed)

```
<Request>
  <Authentication>
    <password>#####</password>
    <client>88000004</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">200.54</amount>
      <merchantreference>94927787</merchantreference>
      <capturemethod>keyed</capturemethod>
    </TxnDetails>
    <FuelCardTxn>
      <method>fuel_offline_auth_settle</method>
      <Terminal id="78547856">
        <terminal_capabilities ic_reader="true"
magnetic_stripe_reader="true" manual_card_entry="true" />
        <features_capabilities pin_pad_available="true" />
      </Terminal>
      <ICC>
        <term_type>22</term_type>
      </ICC>
      <Card>
        <pan>7019000000#####</pan>
        <expirydate>12/15</expirydate>
      </Card>
      <receipt_no>7813</receipt_no>
      <odometer_reading>29057</odometer_reading>
      <registration_number>H702MOF</registration_number>
      <FuelProducts>
        <FuelProduct>
          <prod_code>01</prod_code>
          <prod_quantity>100.01</prod_quantity>
          <prod_value>200.54</prod_value>
        </FuelProduct>
      </FuelProducts>
    </FuelCardTxn>
  </Transaction>
</Request>
```

Example XML for a fuel_online_auth transaction (swiped)

```

<Request>
  <Authentication>
    <password>#####</password>
    <client>88000009</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">50.00</amount>
      <merchantreference>1049277874</merchantreference>
      <capturemethod>swiped</capturemethod>
    </TxnDetails>
    <FuelCardTxn>
      <method>fuel_online_auth</method>
      <Terminal id="78547856">
        <terminal_capabilities ic_reader="true"
magnetic_stripe_reader="true" manual_card_entry="true" />
        <features_capabilities pin_pad_available="true" />
      </Terminal>
      <ICC>
        <term_type>22</term_type>
      </ICC>
      <card_details
type="track2_data">#####</card_details>
      <receipt_no>45671</receipt_no>
      <odometer_reading>125781</odometer_reading>
      <registration_number>H702MOF</registration_number>
    </FuelCardTxn>
  </Transaction>
</Request>

```

Example XML for a fuel_settle transaction

```

<Request>
  <Authentication>
    <password>#####</password>
    <client>88000000</client>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>fuel_settle</method>
      <receipt_no>123456</receipt_no>
      <reference>4800900012345672</reference>
      <authcode>324837</authcode>
    </HistoricTxn>
  </Transaction>
</Request>

```

2.2.8.3 Schema Elements for Response

2.2.8.3.1 Response

Note: See main response section 2.1.1.3 for other elements of the response from DataCash.

Element Name: Response
Position:

Element Name	Description
acquirer_message	Textual message from the acquirer indicating the transaction outcome
acquirer_response_code	Code returned by acquirer
referral_telephone_number	Base64 encoded referral telephone number. Decode this to get the telephone number to call in the case of an acquirer decline / referral.

2.2.8.3.2 FuelCardTxn

Element Name: FuelCardTxn
Position: Response

Element Name	Description
issuer	The Card Issuing Bank, if known
card_scheme	The Card Scheme, if known. Current values which may be returned are listed
country	The Country of Issue, if known
authcode	Authorisation code for successful transactions. Additional information from the bank if declined or referred

2.2.8.4 XML Response Examples

2.2.8.4.1 Successful transactions

Example XML to a successful fuel_online_auth transaction

```
<?xml version="1.0" encoding="UTF-8" ?>
<Response>
  <FuelCardTxn>
    <authcode>100000</authcode>
    <card_scheme>Arval Fuel Card</card_scheme>
    <country>gbr</country>
    <issuer>Arval</issuer>
  </FuelCardTxn>
  <acquirer_message>AUTH CODE:100000</acquirer_message>
  <acquirer_response_code>00</acquirer_response_code>
  <datacash_reference>4900900012345676</datacash_reference>
  <merchantreference>419110205</merchantreference>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1256921756</time>
</Response>
```

Example XML to a successful fuel_settle transaction

```
<?xml version="1.0" encoding="UTF-8" ?>
<Response>
  <datacash_reference>4900900012345676</datacash_reference>
  <merchantreference>419110205</merchantreference>
  <reason>SETTLED OK</reason>
  <status>1</status>
  <time>1253911456</time>
</Response>
```

2.2.8.4.2 Unsuccessful transactions

Example XML to an unsuccessful fuel_online_auth transaction

```
<?xml version="1.0" encoding="UTF-8" ?>
<Response>
  <FuelCardTxn>
    <authcode>DECLINED</authcode>
    <card_scheme>Arval Fuel Card</card_scheme>
    <country>gbr</country>
    <issuer>Arval</issuer>
  </FuelCardTxn>
  <acquirer_message>DECLINED</acquirer_message>
  <acquirer_response_code>05</acquirer_response_code>
  <datacash_reference>4000900012345671</datacash_reference>
  <merchantreference>419110200</merchantreference>
  <reason>DECLINED</reason>
  <referral_telephone_number>MTIzNA==</referral_telephone_number>
  <status>7</status>
  <time>1256921756</time>
</Response>
```

2.2.8.4.3 Invalid transactions

Example XML to an invalid fuel_settle transaction

```
<?xml version="1.0" encoding="UTF-8" ?>
<Response>
  <datacash_reference>4400900012345674</datacash_reference>
  <information>fuel_settle or fuel_offline_auth_settle must contain the receipt_no element</information>
  <reason>Fuels: Missing receipt_no element</reason>
  <status>764</status>
  <time>1256923751</time>
</Response>
```

2.2.9 Hosted Card Capture

The Hosted Card Capture (HCC) Service enables DataCash to capture sensitive cardholder data on behalf of merchants and process transactions with the captured data. As merchants do not capture and store the cardholder data directly, it is easier for you to comply with the Payment Card Industry security standards.

This service may also be used in conjunction to with either the Card Tokenization Service (section 2.8) or the Payment Tokenization Service (section 2.2.11) to enable the Card Holder to enter only their CVV/CV2 details on the capture page, preventing the need to re-enter their card details.

A technical introduction to this Service is available on the website:
<http://www.datacash.com/gettingproducts.php?id=Hosted-Card-Capture>

2.2.9.1 Schema Elements for Request

In this section, the required fields for each transaction type are presented along with example XML for those fields.

The following keys will be used for each transaction type:

- S – setup
- Q – query
- P – a payment transaction, one of: pre, auth, refund, erp

Initial setup transaction

The `setup` transaction provides basic details to initiate the Hosted Card Capture session. This data is passed in these distinct places in the schema:

- Request
 - Authentication – section 2.1.1.2
 - Transaction
 - TxnDetails – contains payment information that will be displayed to the customer, section 2.2.1.3.
 - HpsTxn – contains all the details specific to the HCC session setup request, section 2.2.9.1.3
 - Card – this optional section contains the token for cvv only Card Tokens, section 2.2.9.1.4
 - DynamicData – this optional section contains dynamic data to display on the capture page, section 2.2.9.1.5

Query transaction

After a customer has been returned to your website following the data capture process, you may optionally send a `query` transaction. This will allow you to determine whether the data capture was successful, without sending an authorisation. It will also allow you to obtain information about the card details that were supplied.

This data is passed in these distinct places in the schema:

- Request
 - Authentication – section 2.1.1.2

- o Transaction
 - HistoricTxn – details of the original setup transaction - section 2.2.1.1.4

Payment Using Captured Details

The auth, pre, refund and erp transaction types all require the same information about the transaction to be provided. This data is passed in these distinct places in the schema:

- Request
 - o Authentication – section 2.1.1.2
 - o Transaction
 - TxnDetails – see section 2.2.1.1.3.
 - CardTxn – see section 2.2.9.1.2.
 - Card – contains the address information for AVS checking and policy, section 2.2.9.1.4
 - o ExtendedPolicy – contains details of the AVSCV2 policy, section 2.3.1.1.3
 - ThreeDSecure – optional section containing all the details required to initiate the 3-D Secure check, section 2.3.4.1.1

2.2.9.1.1 TxnDetails

This element is presented for setup and payment transactions only.

Element Name:	TxnDetails
Position:	Request.Transaction

Elements of TxnDetails						
Element Name	Description	Values / Limitations	S	Q	P	
merchantreference	A unique reference number for each transaction	Minimum 6, maximum 30 alphanumeric characters . Must be unique	R	-	O	
amount	The value of the transaction.		R	-	R	

Attributes for Elements of TxnDetails		
Attribute Name	Attribute of Element	Values / Limitations
currency	amount	Currency is passed in the 3 character ISO 4217 Alphabetic format (Default is GBP) e.g. GBP, USD, EUR

Note: if the merchantreference field is not provided for a payment transaction, the DPG will automatically use the same merchantreference as the setup.

Example XML for TxnDetail complex elements

```
<TxnDetail>
    <merchantreference>27fh450osdfhjk3hs</merchantreference>
    <amount currency="AUD">99.99<amount>
</TxnDetail>
```

2.2.9.1.2 CardTxn

For payment transactions only. In addition to the referencing the HCC setup, the transaction type and authorisation code (if applicable) are sent in the CardTxn element:

Element Name: CardTxn
Position: Request.Transaction

Elements of CardTxn		
Element Name	Description	Values / Limitations
authcode	authorisation code received from the bank	Optional element. If presented, must be value received from Banks Authorisation centre
card_details	Specifies the DataCash reference of a successful HCC session setup.	Required. Must be a 16-digit DataCash reference.
method	the transaction type	auth pre refund erp
Card	See section 2.2.9.1.4	

Attributes for Elements of CardTxn		
Attribute Name	Attribute of Element	Values / Limitations
type	card_details	Specifies the nature of the supplied DataCash reference. For an HCC transaction, it must have the value from_hps.

Example XML for CardTxn complex elements

```
<CardTxn>
    <Card>...</Card>
    <card_details type="from_hps">4100203022036495</card_details>
    <method>pre</method>
</CardTxn>
```

2.2.9.1.3 HpsTxn

This element must be present for all setup transactions. It should not be presented for other transaction types.

Element Name: HpsTxn

Position: Request.Transaction

Elements of TxnDetails			
Element Name	Description	Values / Limitations	Required
method	The action that is being requested	Must have the value "setup"	R
page_set_id	An ID number that signifies the "page set" that should be used when presenting the data capture screen to the customer.	An integer value of a page set that has already been configured.	R
return_url	A URL from the customer's website. If this value is supplied, then when the data capture process has been completed, the customer will be redirected to this URL instead of the default URL configured within HCC.	Max size 2083. Must be fully qualified .	O
expiry_url	A URL from the customer's website. If this value is specified, the customer will be redirected to this URL instead of the default expiry URL configured within HCC on attempting to visit an expired session.	Max size 2083. Must be fully qualified .	O
cvv_only	If this element is given the value 1, it signifies that this will be a CVV Only card capture. This means that either a pre-registered card or tokenized card number are being used. The merchant must be subscribed to the HCC CVV Only service for this type of card capture.	A value of 1 means true. A value of 0 means false.	O

Card	The Card element is only required if using the AVSCV2 Service (section 2.3.1), or if a tokenized card number is being presented (section 2.8) for a CVV only transaction.	See section 2.2.9.1.4	
card_details	In a CVV Only card capture, the datacash_reference of the Payment Token (section 2.2.11)	Referenced transaction must have taken place within the previous 13 months, and have a status=1. Mutually exclusive with Card.pan element. cvv_only must be 1.	O

Attributes of elements of HpsTxn				
Attribute Name	Attribute of Element	Description	Values / Limitations	Required
type	card_details	Specifies that a Payment Token is being presented for use with the cvv only transaction	hps_prereg	R

Example HpsTxn element for a session setup to capture the cvv number, where the card has previously been captured and tokenized using the Payment Tokenization Service

```
<HpsTxn>
    <method>setup</method>
    <page_set_id>1</page_set_id>
    <cvv_only>1</cvv_only>
    <card_details type="hps_prereg">2185999900000001</card_details>
    <return_url>https://shop.example.com/process</return_url>
    <expiry_url>https://shop.example.com/expired</expiry_url>
</HpsTxn>
```

Example HpsTxn element for a session setup with Dynamic data

```
<HpsTxn>
  <method>setup</method>
  <page_set_id>99</page_set_id>
  <return_url>https://shop.example.com/process</return_url>
  <expiry_url>https://shop.example.com/expired</expiry_url>
  <DynamicData>...</DynamicData>
</HpsTxn>
```

2.2.9.1.4 Card

The `Card` element is only required for `setup` transactions if a tokenized card number is being presented (section 2.8) for a CVV only transaction.

The `Card` element may also be presented for payment transactions if using the AVSCV2 Service (section 2.3.1).

Element Name:	Card
Position:	Request.Transaction.HpsTxn

Elements of CardTxn				
Element Name	Description	Values / Limitations	S	P
Cv2Avs	Contains the AVSCV2 policy and address details, if using the AVSCV2 Service – section 2.3.1		-	O
pan	A tokenized card number created using the Card Tokenization service (section 2.8).	Mutually exclusive with <code>card_details</code> element. <code>cvv_only</code> must be 1.	R	-

Attributes of elements of Card				
Attribute Name	Attribute of Element	Description	Values / Limitations	Required
type	pan	The Card Token as captured using the Card Tokenization service (section 2.8)	token	R

Example Card element for setup

```
<Card>
<pan type="token"> 3C12DE55292A13227E14D92C6E8D8F4843D21654 </method>
</Card>
```

Example Card element for a payment

```
<Card>
<Cv2Avs>...</Cv2Avs>
</Card>
```

2.2.9.1.5 DynamicData

This optional element contains values that can be used to replace placeholder values in the card capture page.

Element Name:	DynamicData
Position:	Request.Transaction.HpsTxn

Elements of TxnDetails			
Element Name	Description	Values / Limitations	Required
dyn_data_1	The value intended to replace the placeholder __DYNAMIC_DATA_FIELD_1__ in the card capture page.	Max length: 2048 characters. The value must be enclosed in CDATA tags if it contains HTML.	O
dyn_data_2	The value intended to replace the placeholder __DYNAMIC_DATA_FIELD_2__ in the card capture page.	Max length: 2048 characters. The value must be enclosed in CDATA tags if it contains HTML.	O
dyn_data_3	The value intended to replace the placeholder __DYNAMIC_DATA_FIELD_3__ in the card capture page.	Max length: 2048 characters. The value must be enclosed in CDATA tags if it contains HTML.	O
dyn_data_4	The value intended to replace the placeholder __DYNAMIC_DATA_FIELD_4__ in the card capture page.	Max length: 2048 characters. The value must be enclosed in CDATA tags if it contains HTML.	O
dyn_data_5	The value intended to replace the placeholder __DYNAMIC_DATA_FIELD_5__ in the card capture page.	Max length: 2048 characters. The value must be enclosed in CDATA tags if it contains HTML.	O

dyn_data_6	The value intended to replace the placeholder __DYNAMIC_DATA_FIELD_6__ in the card capture page.	Max length: 2048 characters. The value must be enclosed in CDATA tags if it contains HTML.	O
dyn_data_7	The value intended to replace the placeholder __DYNAMIC_DATA_FIELD_7__ in the card capture page.	Max length: 2048 characters. The value must be enclosed in CDATA tags if it contains HTML.	O
dyn_data_8	The value intended to replace the placeholder __DYNAMIC_DATA_FIELD_8__ in the card capture page.	Max length: 2048 characters. The value must be enclosed in CDATA tags if it contains HTML.	O
dyn_data_9	The value intended to replace the placeholder __DYNAMIC_DATA_FIELD_9__ in the card capture page.	Max length: 2048 characters. The value must be enclosed in CDATA tags if it contains HTML.	O

Up to nine values are permitted. If a placeholder value is empty or not present, then the placeholder will be removed from the card capture page that the customer sees.

In order for HTML to be used in placeholder values, the content of the dyn_data element must be enclosed in CDATA tags.

Example DynaminData element containing of HTML enclosed by CDATA tags

```
<DynamicData>
    <dyn_data_1><! [CDATA[some <b>dynamic</b> data] ]></dyn_data_1>
    <dyn_data_2><! [CDATA[more dynamic data] ]></dyn_data_2>
</DynamicData>
```

2.2.9.1.1 HistoricTxn

This element is used when performing query transactions

Element Name:	HistoricTxn
Position:	Request.Transaction

Elements of HistoricTxn			
Element Name	Description	Values / Limitations	
method	The transaction type	query	R
reference	The datacash_reference of the HCC session setup request		R

Example XML Request Fragment for HistoricTxn complex elements

```
<HistoricTxn>
    <reference>3900900100010001</reference>
    <method>query</method>
</HistoricTxn>
```

2.2.9.2 Example XML Requests

2.2.9.2.1 Session setup requests

Example XML request to set up a session, including optional XML elements

```
<Request>
    <Authentication>
        <password>*****</password>
        <client>88000000</client>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <merchantreference>mref-123456</merchantreference>
            <amount currency="GBP">100.00</amount>
        </TxnDetails>
        <HpsTxn>
            <method>setup</method>
            <page_set_id>1</page_set_id>
            <return_url>https://shop.example.com/process</return_url>
            <expiry_url>https://shop.example.com/expired</expiry_url>
        </HpsTxn>
    </Transaction>
</Request>
```

Example XML request to set up a session to only capture the cvv number, where the card has previously been captured and tokenized using the Payment Tokenization Service

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>88000000</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>mref-123456</merchantreference>
      <amount currency="GBP">100.00</amount>
    </TxnDetails>
    <HpsTxn>
      <method>setup</method>
      <page_set_id>1</page_set_id>
      <cvv_only>1</cvv_only>
      <card_details type="hps_prereg">2185999900000001</card_details>
      <return_url>https://shop.example.com/process</return_url>
      <expiry_url>https://shop.example.com/expired</expiry_url>
    </HpsTxn>
  </Transaction>
</Request>
```

Example XML request to set up a session to only capture the cvv number, where the card has previously been captured and tokenized using the Card Tokenization Service

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>88000000</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>mref-123456</merchantreference>
      <amount currency="GBP">100.00</amount>
    </TxnDetails>
    <HpsTxn>
      <method>setup</method>
      <page_set_id>1</page_set_id>
      <cvv_only>1</cvv_only>
      <Card>
        <pan type="token">
          8466A4B11D4F34227E50FF51A6B9CA946B4871FB</pan>
      </Card>
      <return_url>https://shop.example.com/process</return_url>
      <expiry_url>https://shop.example.com/expired</expiry_url>
    </HpsTxn>
  </Transaction>
</Request>
```

Example XML request to set up a session including DynamicData

```

<Request>
  <Authentication>
    <password>*****</password>
    <client>88000000</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>mref-123456</merchantreference>
      <amount currency="GBP">100.00</amount>
    </TxnDetails>
    <HpsTxn>
      <method>setup</method>
      <page_set_id>1</page_set_id>
      <return_url>https://shop.example.com/process</return_url>
      <expiry_url>https://shop.example.com/expired</expiry_url>
      <DynamicData>
        <dyn_data_1>
          <![CDATA[some <b>dynamic</b> data]]>
        </dyn_data_1>
        <dyn_data_2>
          <![CDATA[more dynamic data]]>
        </dyn_data_2>
      </DynamicData>
    </HpsTxn>
  </Transaction>
</Request>

```

2.2.9.2.2 Query

Example XML Session Setup Query

```

<Request>
  <Authentication>
    <password>*****</password>
    <client>88000000</client>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>query</method>
      <reference>3900900100010001</reference>
    </HistoricTxn>
  </Transaction>
</Request>

```

2.2.9.2.3 Payments

Example XML Card Transaction Using Captured Data

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>88000000</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>mref123501</merchantreference>
      <amount>1000.00</amount>
      <capturemethod>ecomm</capturemethod>
    </TxnDetails>
    <CardTxn>
      <method>auth</method>
      <card_details type="from_hps">4000900012345671</card_details>
    </CardTxn>
  </Transaction>
</Request>
```

Example XML Card Transaction Using Captured Data with Standard CV2AVS policy

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>88000000</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>mref123501</merchantreference>
      <amount>1000.00</amount>
      <capturemethod>ecomm</capturemethod>
    </TxnDetails>
    <CardTxn>
      <method>auth</method>
      <card_details type="from_hps">4000900012345671</card_details>
      <Card>
        <Cv2Avs>
          <street_address1>Flat 7</street_address1>
          <street_address2>89 Jumble Street</street_address2>
          <street_address3>Mytown</street_address3>
          <policy>2</policy>
          <postcode>AV12FR</postcode>
        </Cv2Avs>
      </Card>
    </CardTxn>
  </Transaction>
</Request>
```

Example XML Card Transaction Using Captured Data with 3-D Secure and Standard AVSCV2 policy

```

<Request>
  <Authentication>
    <password>*****</password>
    <client>88000000</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>mref123501</merchantreference>
      <amount>1000.00</amount>
      <capturemethod>ecomm</capturemethod>
      <ThreeDSecure>
        <verify>yes</verify>
        <merchant_url>https://www.mywebsite.com</merchant_url>
        <purchase_desc>CDs and DVDs</purchase_desc>
        <purchase_datetime>20120201 23:59:59</purchase_datetime>
        <Browser>
          <device_category>0</device_category>
          <accept_headers>*/*</accept_headers>
          <user_agent>IE/6.0</user_agent>
        </Browser>
      </ThreeDSecure>
    </TxnDetails>
    <CardTxn>
      <method>auth</method>
      <card_details type="from_hps">4000900012345671</card_details>
      <Card>
        <Cv2Avs>
          <street_address1>Flat 7</street_address1>
          <street_address2>89 Jumble Street</street_address2>
          <street_address3>Mytown</street_address3>
          <postcode>AV12FR</postcode>
          <policy>6</policy>
        </Cv2Avs>
      </Card>
    </CardTxn>
  </Transaction>
</Request>

```

2.2.9.3 Schema Elements for XML Response

The standard XML response elements are always returned as described within section 2.1.1.3 above.

Additional elements will also be present, depening upon the transaction type/method and outcome.

A successful setup is indicated by a status code of 1 being returned. Any another code indicates a rejected request or an error.

For query transactions, if the status returned has the value "1", then this indicates that the query has been successful. It does not indicate that the queried transaction

was successful. A full list of the possible values for the `information` element are available in this FAQ:

https://datacash.custhelp.com/cgi-bin/datacash.cfg/php/enduser/std_adp.php?p_faqid=1037

2.2.9.3.1 HpsTxn

This element will only be present for `setup` and `query` responses.

Element Name:	HpsTxn
Position:	Response

Elements of HpsTxn			
Element Name	Description	Values	Limitations
capture_status	The current status of the HCC transaction at the time of querying, e.g. "pending", "populated"	pending populated purged_used purged_unused	Only present for <code>query</code>
card_scheme	The Card Scheme, if known.	Possible values are listed in section 2.2.1.3.1	Only present for <code>query</code> if data is populated and scheme is known
country	The three-character ISO country code, indicating the card's country of issue, if known. E.g. "gbr"		Only present for <code>query</code> if data is populated and country is known
cv2_captured	Determines if a CV2 value has been captured during the process of the HCC transaction.	Yes No	Only present for <code>query</code> if data is populated
expirydate	The expiry date of the date in MMYY format		Only present for <code>query</code> if data is populated
hps_url	The HPS URL. This must be added by the merchant.		Only present for successful <code>setup</code>
issuer	The issuer of the card, if known e.g. "Halifax Plc"		Only present for <code>query</code> if data is populated
pan	An obscured version of the card number that shows only the first six and last four digits. The rest of the digits are replaced with asterisks.		Only present for <code>query</code> if data is populated

session_id	A unique value that corresponds to the session that has been set up within the HCC.		Only present for successful setup
------------	---	--	-----------------------------------

The hps_url and session_id should be used by the merchant to re-direct the [cardholder to the hosted page](#).

Example HpsTxn element for a successful session setup

```
<HpsTxn>
  <hps_url>https://dpg.example.com/hps/</hps_url>
  <session_id>1172</session_id>
</HpsTxn>
```

Example HpsTxn elements for query responses

```
<HpsTxn>
  <card_scheme>VISA</card_scheme>
  <country>gbr</country>
  <expirydate>1214</expirydate>
  <issuer>Halifax Plc</issuer>
  <pan>444433*****1111</pan>
  <capture_status>populated</capture_status>
  <cv2_captured>No</cv2_captured>
</HpsTxn>

<HpsTxn>
  <capture_status>pending</capture_status>
</HpsTxn>
```

2.2.9.3.2 DynamicCapture

This element only applies for Merchants who have customised their capture page with Dynamic Capture fields. Additional information about Dynamic Capture fields is available in the FAQ:

https://datacash.custhelp.com/cgi-bin/datacash.cfg/php/enduser/std_adp.php?p_faqid=1036

Note also that this element may only be present within the query response and will only be returned if data was captured.

Element Name:	DynamicCapture
Position:	Response.HpsTxn

Elements of DynamicCapture			
Element Name	Description	Values / Limitations	
capf1	Dynamically captured field 1, if entered.	32 chars (max)	URL encoded (except spaces)
capf2	Dynamically captured field 2, if entered.	32 chars (max)	URL encoded (except spaces)
capf3	Dynamically captured field 3, if entered.	32 chars (max)	URL encoded (except spaces)
capf4	Dynamically captured field 4, if entered.	32 chars (max)	URL encoded (except spaces)
capf5	Dynamically captured field 5, if entered.	32 chars (max)	URL encoded (except spaces)
capf6	Dynamically captured field 6, if entered.	32 chars (max)	URL encoded (except spaces)
capf7	Dynamically captured field 7, if entered.	32 chars (max)	URL encoded (except spaces)
capf8	Dynamically captured field 8, if entered.	32 chars (max)	URL encoded (except spaces)
capf9	Dynamically captured field 9, if entered.	32 chars (max)	URL encoded (except spaces)

Example DynamicCapture elements for query responses

```
<DynamicCapture>
  <capf1>John+Doe</capf1>
  <capf2>on</capf2>
</DynamicCapture>
```

2.2.9.4 Example XML Responses

2.2.9.4.1 Session Setup

The following example XML message would be returned in response to a successful session setup request.

Example XML response to a successful session setup

```
<Response>
  <HpsTxn>
    <hps_url>https://dpg.example.com/hps/</hps_url>
    <session_id>1172</session_id>
  </HpsTxn>
  <datacash_reference>4000900012345671</datacash_reference>
  <merchantreference>mref-123456</merchantreference>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1258553216</time>
</Response>
```

The following XML demonstrates a response to an unsuccessful setup request.

Example XML response to an unsuccessful session setup

```
<Response>
  <datacash_reference>4000900012345671</datacash_reference>
  <information>new_session is not a valid HPS method</information>
  <merchantreference>mref-123456</merchantreference>
  <mode>TEST</mode>
  <reason>Invalid transaction type</reason>
  <status>15</status>
  <time>1256046381</time>
</Response>
```

2.2.9.4.2 Query Response

Example XML Query Response For A Successful Data Capture

```
<Response>
  <HpsTxn>
    <card_scheme>VISA</card_scheme>
    <country>gbr</country>
    <expirydate>1214</expirydate>
    <issuer>Halifax Plc</issuer>
    <pan>444433*****1111</pan>
    <capture_status>populated</capture_status>
    <cv2_captured>No</cv2_captured>
  </HpsTxn>
  <datacash_reference>4000900012345671</datacash_reference>
  <information>You have queried an HPS transaction, which
has stored data that has not yet been used</information>
  <merchantreference>mref123401</merchantreference>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1256046399</time>
</Response>
```

Example XML Query Response For A Pending Data Capture

```
<Response>
  <HpsTxn>
    <capture_status>pending</capture_status>
  </HpsTxn>
  <datacash_reference>4000900012345671</datacash_reference>
  <information>You have queried an HPS transaction, whose
data capture is pending</information>
  <merchantreference>mref123401</merchantreference>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1256046399</time>
</Response>
```

Example XML Query Response For A Successful Data Capture

```
<Response>
  <HpsTxn>
    <card_scheme>VISA</card_scheme>
    <country>gbr</country>
    <expirydate>1214</expirydate>
    <issuer>Halifax Plc</issuer>
    <pan>444433*****1111</pan>
    <DynamicCapture>
      <capf1>A. N. Other</capf1>
      <capf3>%3Cid%3Ecat%3C%2Fid%3E</capf3>
      <capf6>cat</capf6>
      <capf9>87</capf9>
    </DynamicCapture>
  </HpsTxn>
  <datacash_reference>4000900012345671</datacash_reference>
  <information>You have queried an HPS transaction, which
has stored data that has not yet been used</information>
  <merchantreference>mref123401</merchantreference>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1256046399</time>
</Response>
```

Example XML Query Response For A Pending Data Capture

```
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <information>You have queried an HPS transaction, whose
data capture is pending</information>
    <merchantreference>mref123401</merchantreference>
    <mode>TEST</mode>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <time>1256046399</time>
</Response>
```

2.2.9.4.3 Payment Response

There are no changes to the XML response where data captured with HCC has been used. See section 2.2.1.4 for example XML responses.

2.2.10 Hosted Payment Service

The Hosted Payment Service (HPS) enables DataCash to capture sensitive cardholder data and authorise payments on behalf of merchants. Unlike the HCC Service, the merchant does not need to send subsequent requests in order to authorise the payment. The session setup contains template values that the HPS system uses to process the payment choice that the customer makes on the HPS card capture page. This can include the data required to allow the customer to pay by card or by PayPal.

Further information on HPS is available on the website:

<http://www.datacash.com/gettingproducts.php?id=Fully-Hosted-Payments-Pages-Overview>

2.2.10.1 Schema Elements for Request

In this section, the required fields for all transaction types are presented along with example XML for those fields.

HPS setup_full

The setup_full transaction initiates the setup of the HPS session. The main sections of setup messages are shown below.

- Request
 - Authentication – section 2.1.1.2
 - Transaction
 - TxnDetails – see also section 2.2.10.1.1. This contains payment information that will be displayed to the customer.
 - ThreeDSecure – see also section 2.3.4.1.1. This contains contains all the details required to initiate the 3-D Secure check.
 - TheThirdMan – see also section 2.4.7.1.1 This contains additional information about the customer to be used for Batched Fraud Screening.
 - CardTxn – see also section 2.2.1.1.2. This contains the type of card transaction and other information that will be used if the customer attempts to pay by card.
 - Card – see also section 2.2.1.1.1. In HPS Session Setup, this element is only used as a container for the Cv2Avs element.
 - Cv2Avs – see also section 2.3.3.1.8. This contains cv2 and address details.
 - PayPalTxn – see also section 2.7.1.1.2. This optional element contains all of the details specific to the PayPal transaction, if the customer chooses to pay by PayPal.
 - HpsTxn – this section contains all the details specific to the HPS session setup request, section 2.2.10.1.5

Query of HPS setup_full

At any time following a successful session setup_full, a query can be sent to determine the current state of the transaction. This can be used to determine which stage of the payment process the customer has reached, and whether a payment attempt has been made successfully:

- Request
 - Authentication – section 2.1.1.2
 - Transaction

- HistoricTxn – section 2.2.10.1.6

Query of Payment attempts

If a query has been submitted on the setup_full, one or more payment attempts may be present in the query response. To obtain detailed information about any of those payment attempts, the Merchant may submit additional query transactions against one or more of those payment attempts (as required). The XML Request elements and XML Requests are the same for both Bankcard and PayPal attempts and are discussed discussed within section 2.7.6.

Fulfillment of Payment attempts

Successful Payment attempts can be fulfilled (as required) as outlined in sections:

- Bankcard payments – use the `fulfill` transaction – section 2.2.1
- PayPal Auth Capture payments – use the `do_capture` transaction – section 2.7.1

Cancellation of Payment

Successful Payment attempts can be cancelled prior to settlement (were appropriate) as outlined in sections:

- Bankcard payments – use the `cancel` transaction – section 2.2.1
- PayPal Auth Capture payments – use the `do_void` transaction – section 2.7.1

Refunding Successful Payments

Successful Payment attempts can be cancelled prior to settlement (were appropriate) as outlined in sections:

- Bankcard payments – use the `txn_refund` transaction – section 2.2.1
- PayPal payments – use the `txn_refund` transaction – section 2.7.1

2.2.10.1.1 TxnDetails

All of the following elements and attributes are mandatory in an HPS session setup request.

Element Name:	TxnDetails
Position:	Request.Transaction

Elements of TxnDetails		
Element Name	Description	Values / Limitations
ThreeDSecure	as specified in section 2.3.4.1.1	
The3rdMan	as specified in section 2.3.7.1.1	
merchantreference	A unique reference number for each transaction	Minimum 6, maximum 30 alphanumeric characters . Must be unique .

amount	The value of the transaction	
--------	------------------------------	--

Attributes for Elements of TxnDetails		
Attribute Name	Attribute of Element	Values / Limitations
currency	amount	Currency is passed in the 3 character ISO 4217 Alphabetic format (Default is GBP) e.g. GBP, USD, EUR

2.2.10.1.2 CardTxn

Element Name:	CardTxn
Position:	Request.Transaction

Elements of CardTxn			
Element Name	Description	Values / Limitations	Required
Card	See section 2.2.10.1.3		
method	The transaction type	May only have the values "auth" or "pre"	R

This element is mandatory.

2.2.10.1.3 Card

Element Name:	Card
Position:	Request.Transaction.CardTxn

In a Full HPS Session Setup this element can only contain the Cv2Avs element. No other elements are permitted.

Elements of Cv2Avs			
Element Name	Description	Values / Limitations	Required
Cv2Avs	as specified in section 2.3.3.1.8 . There are no additional restrictions		O

2.2.10.1.4 PayPalTxn

Element Name:	PayPalTxn
---------------	-----------

Position:	Request.Transaction
-----------	---------------------

The PayPalTxn element in an HPS Session Setup Request can contain the elements that are specified in section 2.7.1.1.2 that are indicated as permitted in `set_express_checkout`, `get_express_checkout_details`, `do_express_checkout_payment`, `do_authorization` or `do_void` with the exceptions of `return_url` and `cancel_url`.

The following additional restrictions apply:

Elements of TxnDetails			
Element Name	Description	Values / Limitations	Required
method	The transaction type	When PayPalTxn is included in an HPS Session Setup, this must have the value <code>set_express_checkout</code> .	R
payment_action	Specifies the type of PayPal service to use (Express Checkout or Auth and Capture).	When PayPalTxn is included in an HPS Session Setup, this must have the value <code>order</code> .	R

The PayPalTxn element should be present if the payment page presented to the customer will have an option to pay by PayPal.

2.2.10.1.5 HpsTxn

Element Name: HpsTxn
Position: Request.Transaction

Elements of TxnDetails			
Element Name	Description	Values / Limitations	Required
method	The action that is being requested	Must have the value <code>setup_full</code>	R
page_set_id	An ID number that signifies the "page set" that should be used when presenting the data capture screen to the customer.	An integer value of a page set that has already been configured.	R
return_url	A URL from the customer's website. If this value is supplied, then when the data capture process has been completed, the customer will be redirected to this URL instead of the default URL configured within HPS.	Max size 2083. Must be <u>fully qualified</u> .	O

expiry_url	A URL from the customer's website. If this value is specified, the customer will be redirected to this URL instead of the default expiry URL configured within HPS on attempting to visit an expired session.	Max size 2083. Must be fully qualified .	O
------------	---	---	---

2.2.10.1.6 HistoricTxn

This element should only be presented for query transactions.

Element Name: HistoricTxn
 Position: Request.Transaction

Elements of HistoricTxn			
Element Name	Description	Values / Limitations	
method	The transaction type	query	R
reference	The datacash_reference of the HPS session setup request		R

Example XML Request Fragment for HistoricTxn complex elements

```
<HistoricTxn>
  <reference>3900900100010001</reference>
  <method>query</method>
</HistoricTxn>
```

2.2.10.2 Example XML Requests

2.2.10.2.1 Example Session Setup XML Requests

Example XML request to set up a session, including optional XML elements

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>88000000</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>mref-123456</merchantreference>
      <amount currency="GBP">100.00</amount>
      <capturemethod>ecomm</capturemethod>
      <ThreeDSecure>
        <verify>yes</verify>
        <merchant_url>https://www.example-merchant.co.uk</merchant_url>
        <purchase_datetime>20120121 09:31:40</purchase_datetime>
    </TxnDetails>
  </Transaction>
</Request>
```

```

<purchase_desc>goods</purchase_desc>
<Browser>
  <device_category>0</device_category>
  <accept_headers>.*</accept_headers>
  <user_agent>Mozilla</user_agent>
</Browser>
</ThreeDSecure>
<The3rdMan>
  ...
</The3rdMan>
</TxnDetails>
<CardTxn>
  <method>pre</method>
  <Card>
    <Cv2Avs>
      <postcode>...</postcode>
      <street_address1>...</street_address1>
      <street_address2>...</street_address2>
      <street_address3>...</street_address3>
      <street_address4>...</street_address4>
      <policy>3</policy>
    </Cv2Avs>
  </Card>
</CardTxn>
<PayPalTxn>
  <method>set_express_checkout</method>
  <email>foo@example.com</email>
  <max_amount>1000.00</max_amount>
  <description>Twenty NeverFail(tm) Widgets</description>
  <custom>UPS_ID=12345678</custom>
  <invnum>abc123</invnum>
  <req_confirmed_shipping>0</req_confirmed_shipping>
  <no_shipping>0</no_shipping>
  <override_address>0</override_address>
  <localecode>GB</localecode>
  <header_style img="https://www.example.com/header.png"
bordercolor="black" bgcolor="red" />
  <page_style stylename="My Style" bgcolor="white" />
  <ShippingAddress>
    <name>Test User</name>
    <street_address1>1 Main Terrace</street_address1>
    <street_address2>Highgate</street_address2>
    <city>Wolverhampton</city>
    <region>West Midlands</region>
    <country_code>GB</country_code>
    <postcode>E12 4LQ</postcode>
    <telephone_number>0121 231 3122</telephone_number>
  </ShippingAddress>
</PayPalTxn>
<HpsTxn>
  <method>setup_full</method>
  <page_set_id>1</page_set_id>
  <return_url>https://shop.example.com/process</return_url>
</HpsTxn>
</Transaction>
</Request>

```

2.2.10.2.2 Example Query Request

Example XML Session Setup Query

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>88000000</client>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>query</method>
      <reference>3900900100010001</reference>
    </HistoricTxn>
  </Transaction>
</Request>
```

2.2.10.3 Schema Elements for Response

The standard XML response elements are always returned. See 2.1.1.3 above.

Note that for query transactions, if the `status` returned has the value "1", then this indicates that the query has been successful. It does not indicate that the queried transaction was successful.

A successful setup is indicated by a status code of 1 being returned. Any another code indicates a rejected request or an error.

When querying an HPS session setup request, the contents of the "information" element in the response indicate the status of the queried transaction. The values of these are shown in the FAQ:

https://datacash.custhelp.com/cgi-bin/datacash.cfg/php/enduser/std_adp.php?p_faqid=1074

2.2.9.4.1 HpsTxn

This element may appear for both `setup_full` and `query` Responses. For query responses, it will only appear if the customer has made at least one payment attempt.

Element Name:	HpsTxn
Position:	Response

Elements of HpsTxn				
Element Name	Description	Limitations	S	Q
AuthAttempts	See section 2.2.9.4.2		-	O
datacash_reference	The datacash_reference of the successful payment attempt	Will only be present if one of the payment attempts was successful	-	O

<code>hps_url</code>	The HPS URL.		0	-
<code>session_id</code>	A unique value that corresponds to the session that has been set up within the HPS.		0	-

The `hps_url` and `session_id` should be used by the merchant to re-direct the [cardholder to the hosted page](#).

Example HpsTxn Element for a setup_full transactions

```
<HpsTxn>
  <hps_url>https://testserver.datacash.com/hps/</hps_url>
  <session_id>f2d1530f-bdb3-41f3-97b0-104dc2a48a32</session_id>
</HpsTxn>
```

Example HpsTxn Element for a query transactions

With successful payment:

```
<HpsTxn>
  <AuthAttempts>...</AuthAttempts>
  <datacash_reference>4700201011629830</datacash_reference>
</HpsTxn>
```

without successful attempts:

```
<HpsTxn>
  <AuthAttempts>...</AuthAttempts>
</HpsTxn>
```

2.2.10.3.1 AuthAttempts

This element contains a list of the payment attempts made during the session. This element will only appear if the customer has made at least one payment attempt.

Element Name:	AuthAttempts
Position:	Response.HpsTxn

Example AuthAttempts Element with a query with two payment attempts:

```
<AuthAttempts>
  <Attempt>...</Attempt>
  <Attempt>...</Attempt>
</AuthAttempts>
```

2.2.10.3.2 Attempt

This element provides details of a payment attempt made during the session.

Element Name: Attempt
Position: Response.HpsTxn.AuthAttempts

Elements of Attempt		
Element Name	Description	Values / Limitations
datacash_reference	The datacash reference of the auth attempt for the payment	
dc_response	The status code for the result of the attempt	
reason	The reason for the auth attempt	

Example AuthAttempt Element for a successful payment attempt:

```
<AuthAttempt>
  <datacash_reference>4700201011629830</datacash_reference>
  <dc_response>1</dc_response>
  <reason>ACCEPTED</reason>
</AuthAttempt>
```

Example AuthAttempt Element for an unsuccessful payment attempt:

```
<AuthAttempt>
  <datacash_reference>4700201011629830</datacash_reference>
  <dc_response>179</dc_response>
  <reason>3DS Payer failed Verification</reason>
</AuthAttempt>
```

2.2.10.3.3 PayPalTxn

This element will be present if submitting a query against a payment attempt which was made via PayPal. It provides details of a payment attempt made during the session.

Element Name: PayPalTxn

Position:	Response
-----------	----------

Elements of HpsTxn		
Element Name	Description	Values / Limitations
last_activity	Indicates that last step of the PayPal express checkout process that was performed as part of the payment attempt.	set_express_checkout get_express_checkout_details do_express_checkout_payment do_authorization do_void do_capture

Example PayPalTxn XML element when querying a PayPal Attempt

```
<PayPalTxn>
  <last_activity>get_express_checkout_details</last_activity>
</PayPalTxn>
```

2.2.10.4 Example XML Responses

2.2.10.4.1 Example Session Setup XML Responses

Example XML response to a successful session setup

```
<Response>
  <HpsTxn>
    <hps_url>https://dpg.example.com/hps/</hps_url>
    <session_id>1172</session_id>
  </HpsTxn>
  <datacash_reference>4000900012345671</datacash_reference>
  <merchantreference>mref-123456</merchantreference>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1258553216</time>
</Response>
```

Example XML response to an unsuccessful session setup

```
<Response>
  <datacash_reference>4000900012345671</datacash_reference>
  <information>new_session is not a valid HPS method</information>
  <merchantreference>mref-123456</merchantreference>
  <mode>TEST</mode>
  <reason>Invalid transaction type</reason>
  <status>15</status>
  <time>1256046381</time>
</Response>
```

2.2.10.4.2 XML Example Query Responses

Example XML Query Response on a setup_full where a successful payment attempt has been made after 2 unsuccessful attempts:

```
<Response>
  <HpsTxn>
    <session_id>1172</session_id>
    <datacash_reference>4000900012345674</datacash_reference>
    <AuthAttempts>
      <Attempt>
        <datacash_reference>4000900012345672</datacash_reference>
        <reason>Declined</reason>
        <status>7</status>
      </Attempt>
      <Attempt>
        <datacash_reference>4000900012345673</datacash_reference>
        <reason>Declined</reason>
        <status>7</status>
      </Attempt>
      <Attempt>
        <datacash_reference>4000900012345674</datacash_reference>
        <reason>ACCEPTED</reason>
        <status>1</status>
      </Attempt>
    </AuthAttempts>
  </HpsTxn>
  <datacash_reference>4000900012345671</datacash_reference>
  <merchantreference>mref-123456</merchantreference>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>_UNIX_TIME_</time>
</Response>
```

Example XML Query Response on a setup_full where all payment attempts have failed:

```

<Response>
  <HpsTxn>
    <session_id>1172</session_id>
    <AuthAttempts>
      <Attempt>
        <datacash_reference>4000900012345672</datacash_reference>
        <reason>Declined</reason>
        <status>7</status>
      </Attempt>
      <Attempt>
        <datacash_reference>4000900012345673</datacash_reference>
        <reason>Declined</reason>
        <status>7</status>
      </Attempt>
      <Attempt>
        <datacash_reference>4000900012345674</datacash_reference>
        <reason>Declined</reason>
        <status>7</status>
      </Attempt>
    </AuthAttempts>
  </HpsTxn>
  <datacash_reference>4000900012345671</datacash_reference>
  <information>You have queried an HPS transaction, where the all retry attempts were exhausted.</information>
  <merchantreference>mref-123456</merchantreference>
  <mode>TEST</mode>
  <reason>HPS: The maximum number of retry transaction was breached</reason>
  <status>822</status>
  <time>_UNIX_TIME_</time>
</Response>

```

2.2.10.4.3 XML Example PayPal Attempt Query Response

Example XML PayPal Attempt Query Response

```

<Response>
  <PayPalTxn>
    <last_activity>get_express_checkout_details</last_activity>
  </PayPalTxn>
  <datacash_reference>4000900012345673</datacash_reference>
  <reason>HPS: Paypal - Waiting for customer to confirm payment.</reason>
  <status>846</status>
  <time>_UNIX_TIME_</time>
</Response>

```

[Back to Contents](#)

REPEAT CARD PAYMENTS

2.2.11 *Payment Tokenization*

A technical introduction to this Service is available on the website:

<http://www.datacash.com/gettingproducts.php?id=Payment-Tokenization-Pre-Registered-Card->

2.2.11.1 Schema Elements for Request

In this section the required fields for each transaction type will be presented, along with example XML for those fields. If a transaction type is specified in the XML, this is **highlighted** to indicate that other transaction types can be used in its place.

This document assumes that the Credit and Debit Card Service has already been integrated and the reader is familiar with it.

The data for the Payment Tokenization Service is passed in these distinct places in the schema:

2.2.11.1.1 Request

- Authentication - section 2.1.1.2
- Transaction
 - CardTxn – the **type** and **datacash_reference** of the original transaction, section 2.6.1.1.2
 - Secure – to be used if the 3-D Secure check using a 3rd Party MPI is required, section 2.3.6.1.2
 - Card
 - Cv2Avs – to be used if the CV2AVS check is required, section 2.3.3.1.8
 - TxnDetails – section 2.2.1.1.3
 - ThreeDSecure – to be used if the 3-D Secure check using the DataCash MPI is required, section 2.3.4.1.1

2.2.11.1.2 CardTxn

The details of the initial Credit and Debit Card transaction, plus the method to be used for the Payment Tokenization transaction are submitted in the `CardTxn` element. If the CV2AVS check is being used, this information is also presented within this element:

Element Name:	CardTxn
Position(s)	Request.Transaction

Elements of CardTxn		
Element Name	Description	Values / Limitations
Card	To be used if the CV2AVS check is required.	Please refer to section 2.6.1.1.5
method	The transaction type	auth pre refund erp
card_details	The <code>datacash_reference</code> of the initial transaction	Initial transaction must have taken place within the previous 13 months, and have a <code>status=1</code>

Attributes for Elements of CardTxn		
Attribute Name	Attribute of element	Value
type	card_details	preregistered

Example XML for CardTxn elements

```
<CardTxn>
  <method>pre</method>
  <card_details type="preregistered">2185999900000001</card_details>
</CardTxn>
```

Example XML for CardTxn elements

```
<CardTxn>
  <method>auth</method>
  <card_details type="preregistered">2185999900000008</card_details>
  <Card>...</Card>
</CardTxn>
```

2.2.11.2 XML Examples Requests

Example XML Request for pre

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>123402</merchantreference>
      <amount currency="USD">249.99</amount>
    </TxnDetails>
    <CardTxn>
      <method>pre</method>
      <card_details type="preregistered">2185999900000001
        </card_details>
    </CardTxn>
  </Transaction>
</Request>
```

If the `method` is changed, the same XML could also be used for `auth`, `refund` and `erp` transactions.

It is also possible to use a Payment Token in conjunction with the HCC service – see section 2.2.9

For examples with the CV2AVS check, please refer to section 2.3.3.1.8. Examples using 3-D Secure are available in section 2.3.4 (DataCash MPI) and section 2.3.6 (3rd Party MPI).

2.2.11.3 Schema Elements for Response

The Payment Tokenization Service will return same structure of Response as the Credit and Debit Card Service. Please refer to section 2.2.1.3 of this document.

2.2.11.4 XML Examples Responses

The Payment Tokenization Service will return same structure of Response as the Credit and Debit Card Service. Examples of successful, declined and referred responses are given in the Credit and Debit Card Service section 2.2.1.4 of this document.

The initial transaction is older than 13 months, does not exist or was unsuccessful

```
<Response>
  <datacash_reference>2185999900005678</datacash_reference>
  <merchantreference>123402</merchantreference>
  <reason>Prereg: Invalid reference</reason>
  <status>250</status>
  <time>1071567305</time>
</Response>
```

The account is not currently configured for the Payment Tokenization Service

```
<Response>
  <datacash_reference>2185999900005679</datacash_reference>
  <information>This vTID is not configured to process
    pre-registered card transactions.</information>
  <merchantreference>123403</merchantreference>
  <reason>Prereg: Merchant Not Subscribed</reason>
  <status>251</status>
  <time>1074692433</time>
</Response>
```

2.2.12 Capture Method Recurring Transactions

A technical introduction to this Service is available on the website:
<http://www.datacash.com/gettingproducts.php?id=Capture-Method-Recurring-Transactions>

This service is utilised by sending a normal Credit and Debit Card Service Request with additional information. This section of the documentation assumes the reader is familiar with it. The Credit and Debit Card Service is described in section 2.2.1

2.2.12.1 Schema Elements for Request

In this section the required fields for each capture method type will be presented, along with example XML for those fields. If a capture method is specified in the XML, this is **highlighted** to indicate that other capture method types can be used in its place.

The data for the Capture Method Recurring Transactions Service is passed in three distinct places in the schema:

2.2.12.1.1 Request

- Authentication - section 2.1.1.2
- Transaction
 - CardTxn – the type and authorisation code, see section 2.2.1.1.2
 - Card – this contains all the information about the Card, section 2.2.1.1.1
 - TxnDetails – contains details of the transaction, see section 2.2.12.1.2

When using this Service, a single extra element is provided within the TxnDetails parent for transactions.

2.2.12.1.2 TxnDetails

Element Name: TxnDetails
Position(s) Request.Transaction

Elements of TxnDetails		
Element Name	Description	Values / Limitations
merchantreference	Please refer to section 0	
amount		
capturemethod	Specifies the environment of transaction	ecomm cnp cont_auth ivr

Please note that refund transaction types (including `erp` and `txn_refund`) cannot be processed in a `cont_auth` environment. These should instead be processed using your

normal environment flag.

Example XML Elements

```
<TxnDetails>
    <merchantreference>123401</merchantreference>
    <amount currency="GBP">189.00</amount>
    <capturemethod>cont_auth</capturemethod>
</TxnDetails>

<TxnDetails>
    <merchantreference>123402</merchantreference>
    <amount currency="GBP">52.82</amount>
    <capturemethod>cnp</capturemethod>
</TxnDetails>
```

2.2.12.1.2.1 NatWest Merchants

If you are processing transactions through NatWest Streamline, the following additional information should also be presented:

Attributes of child elements in TxnDetails			
Attribute Name	Element Name	Description	Values / Limitations
Streamline_CA_environment	capturemethod	Specifies the environment of original transaction	ecomm cnp

Example XML Elements for NatWest merchants

```
<TxnDetails>
    <merchantreference>123403</merchantreference>
    <amount currency="GBP">249.99</amount>
    <capturemethod>ecomm</capturemethod>
</TxnDetails>

<TxnDetails>
    <merchantreference>123403</merchantreference>
    <amount currency="USD">249.99</amount>
    <capturemethod Streamline_CA_environment="ecomm">
        cont_auth</capturemethod>
</TxnDetails>
```

2.2.12.2 XML Example Request

2.2.12.2.1 Ecomm and CNP Transactions

Example XML Request for an initial ecomm transaction

```

<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>123404</merchantreference>
      <amount currency="GBP">100.00</amount>
      <capturemethod>ecomm</capturemethod>
    </TxnDetails>
    <CardTxn>
      <method>pre</method>
      <Card>
        <expirydate>02/06</expirydate>
        <issuemumber>01</issuemumber>
        <startdate>0199</startdate>
        <pan>444433*****1</pan>
      </Card>
    </CardTxn>
  </Transaction>
</Request>

```

If the `capturemethod` above is changed, the same XML could also be used for `cnp` transactions.

When a 'refund of an existing transaction' (`txn_refund`) is being performed, the DPG will automatically use the same `capturemethod` as the existing transaction. However, if the existing transaction is a `cont_auth` transaction, the `capturemethod` should be explicitly set to the normal (i.e. `ecomm` or `cnp`) value.

Example XML Request for a `txn_refund`

```

<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>123404</merchantreference>
      <amount currency="GBP">100.00</amount>
      <capturemethod>ecomm</capturemethod>
    </TxnDetails>
    <HistoricTxn>
      <method>txn_refund</method>
      <reference>490020000000001</reference>
    </HistoricTxn>
  </Transaction>
</Request>

```

2.2.12.2.2 Cont_Auth Transactions

2.2.12.2.2.1 Non-NatWest Merchants

To send a `cont_auth` transaction, non-NatWest merchants can use the same XML Request as shown in section 2.2.12.2.1, merely by changing the value of the `capturemethod`.

2.2.12.2.2.2 NatWest Merchants

Example XML Request for a NatWest recurring transaction

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>123404</merchantreference>
      <amount currency="GBP">100.00</amount>
      <capturemethod
        Streamline_CA_environment="cnp">cont_auth</capturemethod>
    </TxnDetails>
    <CardTxn>
      <method>pre</method>
      <Card>
        <expirydate>02/06</expirydate>
        <issuenumber>01</issuenumber>
        <startdate>0199</startdate>
        <pan>444433*****1</pan>
      </Card>
    </CardTxn>
  </Transaction>
</Request>
```

2.2.13 *Historic Recurring Transactions*

A technical introduction to this Service is available on the website:
<http://www.datacash.com/gettingproducts.php?id=Historic-Recurring-Cards>

This section of the documentation assumes the reader is familiar with the Credit and Debit Card Service, which is discussed in section 2.2.1.

2.2.13.1 Schema Elements for Request

In this section the required fields for each transaction type will be presented, along with example XML for those fields.

For Recurring Account Setups

To authorise the first payment and setup an account for a card, information needs to be collected and set in the following places within the schema:

2.2.13.1.1 Request

- Authentication - section 2.1.1.2
- Transaction
 - TxnDetails – contains details of the transaction, see section 2.2.12.1.2
 - CardTxn
 - Card – this element contains the details about the card, for setups only. These details are the same as for one-off payments, which are covered in section 2.2.1.1.1
 - ContAuthTxn – section 2.2.13.1.2

For the Repeat Payments

Once the account has been setup, repeat payments can be taken from it.

- Request
- Authentication – section 2.1.1.2
- Transaction
 - TxnDetails – contains details of the transaction, section 0
 - HistoricTxn – section 2.2.13.1.3

For Cancellations of Payments and Accounts

Cancellations of accounts and payments are performed using the same data as for normal Bank Card cancellations:

- Request
- Authentication – section 2.1.1.2
- Transaction
 - HistoricTxn – section 2.2.13.1.3

2.2.13.1.2 ContAuthTxn

This element should be presented with setup and repeat payment Requests. It has one attribute and no children.

Element Name:	ContAuthTxn
Position	Request.Transaction

Attributes of ContAuthTxn			
Attributes	Attribute of Element	Description	Values / Limitations
type	ContAuthTxn	Indicates whether the transaction is a setup or a repeat payment	setup historic

Example XML ContAuthTxn elements

```
<ContAuthTxn type="setup"></ContAuthTxn>
<ContAuthTxn type="setup" />
<ContAuthTxn type="historic" />
```

2.2.13.1.3 HistoricTxn

This element is required for the repeat payments and cancellations of payments/accounts. It should not be presented with the setup transaction.

Element Name:	HistoricTxn
Position	Request.Transaction

Elements of HistoricTxn		
Element Name	Description	Values / Limitations
method	Indicates the transaction type	pre auth cancel
reference	For Payments: the datacash_reference of the account from which the payment is to be taken	Must be a valid account
	For Cancellations: the datacash_reference of the account or payment to be cancelled	Must be a valid account or payment

Example XML HistoricTxn elements

```

<HistoricTxn>
    <method>auth</method>
    <reference>4700200040912890</reference>
</HistoricTxn>

<HistoricTxn>
    <method>cancel</method>
    <reference>4700200040919424</reference>
</HistoricTxn>

```

2.2.13.2 Example XML Requests

Example XML Request for account setup. The initial transaction will be processed as ecomm

```

<Request>
    <Authentication>
        <client>99000001</client>
        <password>mypasswd</password>
    </Authentication>
    <Transaction>
        <ContAuthTxn type="setup"/>
        <TxnDetails>
            <merchantreference>385036349305556</merchantreference>
            <amount currency="GBP">1001.00</amount>
            <capturemethod>ecomm</capturemethod>
        </TxnDetails>
        <CardTxn>
            <Card>
                <expirydate>01/06</expirydate>
                <issuemumber>1</issuemumber>
                <startdate>0199</startdate>
                <pan>444433*****1</pan>
            </Card>
            <method>auth</method>
        </CardTxn>
    </Transaction>
</Request>

```

Example XML Request for account setup. The initial transaction will be processed as cnp

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>mypasswd</password>
  </Authentication>
  <Transaction>
    <CardTxn>
      <Card>
        <expirydate>01/06</expirydate>
        <pan>444433*****1</pan>
      </Card>
      <method>pre</method>
    </CardTxn>
    <ContAuthTxn type="setup"/>
    <TxnDetails>
      <merchantreference>385036</merchantreference>
      <amount currency="AUD">56.52</amount>
      <capturemethod>cnp</capturemethod>
    </TxnDetails>
  </Transaction>
</Request>
```

Example XML Request for a repeat payment

```
<Request>
  <Transaction>
    <ContAuthTxn type="historic"/>
    <TxnDetails>
      <merchantreference>3851231</merchantreference>
      <capturemethod>cont_auth</capturemethod>
      <amount currency="GBP">18.50</amount>
    </TxnDetails>
    <HistoricTxn>
      <reference>4500200040925092</reference>
      <method>auth</method>
    </HistoricTxn>
  </Transaction>
  <Authentication>
    <client>99000001</client>
    <password>mypasswd</password>
  </Authentication>
</Request>
```

Example XML Request to cancel a payment or account

```

<Request>
  <Authentication>
    <password>mypasswd</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>cancel</method>
      <reference>4800200040644359</reference>
    </HistoricTxn>
  </Transaction>
</Request>

```

2.2.13.3 Schema Elements for Response

In addition to the elements covered in this section, responses for this service will contain the general response elements outlined in section 2.1.1.3.

The CardTxn element will also be present, as described in section 2.2.1.3.1.

2.2.13.3.1 ContAuthTxn

Element Name: ContAuthTxn	
Position	Response

Elements of ContAuthTxn	
Element Name	Description
ca_reference	The reference number of the account
account_status	Provides additional information about the account status

Example ContAuthTxn element

```

<ContAuthTxn>
  <account_status>Account setup. CNP transaction
  sent successfully</account_status>
  <ca_reference>4500200040151462</ca_reference>
</ContAuthTxn>

```

2.2.13.4 Example Responses

2.2.13.4.1 Successful Responses

Example XML Response for a successful setup

```

<Response>
  <CardTxn>
    <authcode>100001</authcode>
    <card_scheme>Mastercard</card_scheme>
    <country>United Kingdom</country>
  </CardTxn>
  <ContAuthTxn>
    <account_status>Account setup. ECOMM transaction sent
      successfully</account_status>
    <ca_reference>4700200040641889</ca_reference>
  </ContAuthTxn>
  <datacash_reference>4700200040641890</datacash_reference>
  <merchantreference>setupreference12345680</merchantreference>
  <mode>...</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>...</time>
</Response>

```

Example XML Response for a successful repeat payment

```

<Response>
  <CardTxn>
    <authcode>779099</authcode>
    <card_scheme>VISA</card_scheme>
    <country>United Kingdom</country>
  </CardTxn>
  <ContAuthTxn>
    <account_status>Using account ref 4500200040925092. CONT_AUTH
      transaction complete</account_status>
  </ContAuthTxn>
  <datacash_reference>4000200040925117</datacash_reference>
  <merchantreference>385036359953704</merchantreference>
  <mode>...</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>...</time>
</Response>

```

Example XML Response for a successfully cancelled account

```
<Response>
    <datacash_reference>4400200040943162</datacash_reference>
    <information>Continuous authority account cancelled</information>
    <merchantreference>4500200040925092</merchantreference>
    <mode>...</mode>
    <reason>CANCELLED</reason>
    <status>1</status>
</Response>
```

2.2.13.4.2 Failed or Rejected Responses

Example XML Response for a rejected setup. The initial payment was declined, so the account could not be set up.

```
<Response>
    <CardTxn>
        <authcode>DECLINED</authcode>
        <card_scheme>VISA Delta</card_scheme>
        <country>United Kingdom</country>
    </CardTxn>
    <ContAuthTxn>
        <account_status>Account not setup. CNP transaction was not
            successfully authed</account_status>
    </ContAuthTxn>
    <datacash_reference>4500200040641890</datacash_reference>
    <merchantreference>...</merchantreference>
    <mode>...</mode>
    <reason>DECLINED</reason>
    <status>7</status>
    <time>...</time>
</Response>
```

Example XML Response where a recurring payment Request was declined

```
<Response>
    <CardTxn>
        <authcode>NOT AUTHORISED</authcode>
        <card_scheme>American Express</card_scheme>
    </CardTxn>
    <ContAuthTxn>
        <account_status>Using account ref 3900200040457927.
            CONT_AUTH transaction complete</account_status>
    </ContAuthTxn>
    <datacash_reference>3800200040465541</datacash_reference>
    <merchantreference>...</merchantreference>
    <mode>LIVE</mode>
    <reason>DECLINED</reason>
    <status>7</status>
    <time>...</time>
</Response>
```

Example XML Response where the account could not be found or has already been cancelled

```
<Response>
  <datacash_reference>4500200040373397</datacash_reference>
  <information>Failed to find live CA account with
    reference=47002000405791371</information>
  <merchantreference>historicreference12345690</merchantreference>
  <reason>Failed to find corresponding CA account</reason>
  <status>192</status>
  <time>...</time>
</Response>
```

2.2.14 *Fire and Forget Recurring Transactions*

A technical introduction to this Service is available on the website:

<http://www.datacash.com/gettingproducts.php?id=Bank-Card-Fire-Forget->

This section of the documentation assumes the reader is familiar with the Credit and Debit Card Service, which is discussed in section 2.2.1.

2.2.14.1 Schema Elements for Request

In this section the required fields for each transaction type will be presented, along with example XML for those fields. As not all fields are mandatory, the following key will be used:

- R - Required
- O - Optional

For Recurring Account Setups

The information required to set up an account is passed in several places in the schema:

2.2.14.1.1 Request

- Authentication - section 2.1.1.2
- Transaction
 - CardTxn – the type and authorisation code, see section 2.2.1.1.2
 - Card – this contains all the information about the Card. These details are the same as for one-off payments, which are covered in section 2.2.1.1.1
 - ContAuthTxn – contains all the information about the payments which are to be taken from the account, section 2.2.13.1.2
 - FirstPayment – enables an initial payment to be specified, section 0
 - LastPayment – enables a final payment to be specified, section 0

For Cancellations of Payments and Accounts

Account and Payment Cancellations are performed using the same data as for normal Bank Card cancellations:

2.2.14.1.2 Request

- Authentication - section 2.1.1.2
- Transaction
 - HistoricTxn – – details the transaction method cancel, and the reference number of the payment/account to cancel, section 2.2.1.1.4

2.2.14.1.3 ContAuthTxn

This element must be presented when setting up an account.

Element Name:	ContAuthTxn
Position	Request.Transaction

Elements of ContAuthTxn			
Element Name	Description	Values / Limitations	
method	Indicates the transaction type	pre auth	R
start_date	The date on which the first regular payment will be taken. Subsequent payments will be based from this date. Must be at least the day after the transaction is sent.	dd/mm/yyyy. Must be at least one day ahead of either the current date or the FirstPayment date	R
period	The frequency of payments	weekly monthly quarterly annual	R
amount	The value of each regular payment		R
number_of_payments	Used to restrict the number of regular payments to be collected. Note: this is not the total number of payments if first and/or last payments have been set.	Must be an integer	O
FirstPayment	Details of any initial payment to be made. This is in addition to the regular payments. See section 2.2.14.1.4		O
LastPayment	Details of any final payment to be made. This is in addition to the regular payments. See section 2.2.14.1.5		O

Attributes of ContAuthTxn			
Attributes	Attribute of Element	Description	Values / Limitations
currency	amount	Currency in the ISO 4217 Alphabetic format. E.g. GBP, USD, AUD	Will default to GBP if not specified

Example ContAuthTxn element. Monthly payments will be taken until the account is cancelled

```
<ContAuthTxn>
  <method>auth</method>
  <start_date>31/03/2005</start_date>
  <period>monthly</period>
  <amount>25.00</amount>
</ContAuthTxn>
```

Example ContAuthTxn element with first and last payments

```
<ContAuthTxn>
  <method>auth</method>
  <start_date>10/02/2003</start_date>
  <period>weekly</period>
  <number_of_payments>5</number_of_payments>
  <amount>10.00</amount>
  <FirstPayment>...</FirstPayment>
  <LastPayment>...</LastPayment>
</ContAuthTxn>
```

2.2.14.1.4 FirstPayment

This optional element enables an initial payment to be taken which varies in both value and timing from the regular transactions.

If this element is presented for a transaction, both its children must be populated.

Element Name:	FirstPayment
Position	Request.Transaction.ContAuthTxn

Elements of FirstPayment		
Element Name	Description	Values / Limitations
date	The date the first payment is to be taken	dd/mm/yyyy Must be at least one day ahead of the current date
amount	The value of the first payment. This may vary from the regular payments. Currency attribute may be specified if required	

Example FirstPayment element

```
<FirstPayment>
  <amount>30.00</amount>
  <date>29/06/2005</date>
</FirstPayment>
```

2.2.14.1.5 LastPayment

This optional element enables a final payment to be taken which varies in both value and timing from the regular transactions. The element may only be presented if the `number_of_payments` element has been specified (section 0). If this element is presented for a transaction, both its children must be populated.

Element Name:	LastPayment
Position	Request.Transaction.ContAuthTxn

Elements of LastPayment		
Element Name	Description	Values / Limitations
date	The date the last payment is to be taken	dd/mm/yyyy Must be at least one day after the last regular payment
amount	The value of the last payment. This may vary from the regular payments. Currency attribute may be specified if required	

Example LastPayment element

```
<LastPayment>
  <amount>20.00</amount>
  <date>12/01/2016</date>
</LastPayment>
```

2.2.14.2 Example XML Requests

Example Request for an initial charge of £7.50, followed by regular weekly payments of £5 until a cancellation Request is received

```
<Request>
  <Authentication>
    <client>99106700</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <CardTxn>
      <Card>
        <expirydate>01/06</expirydate>
        <pan>444433*****1</pan>
      </Card>
    </CardTxn>
    <ContAuthTxn>
      <method>auth</method>
      <start_date>15/07/2005</start_date>
      <period>weekly</period>
      <amount>5.00</amount>
      <FirstPayment>
        <amount>7.50</amount>
        <date>09/07/2005</date>
      </FirstPayment>
    </ContAuthTxn>
    <TxnDetails>
      <merchantreference>383914410763889</merchantreference>
      <capturemethod>ecomm</capturemethod>
    </TxnDetails>
  </Transaction>
</Request>
```

Example Request for nine quarterly payments of £69.99

```
<Request>
  <Authentication>
    <client>99106700</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <CardTxn>
      <Card>
        <expirydate>06/13</expirydate>
        <pan>3434*****3</pan>
      </Card>
    </CardTxn>
    <ContAuthTxn>
      <method>auth</method>
      <start_date>31/07/2005</start_date>
      <period>quarterly</period>
      <amount currency="GBP">69.99</amount>
      <number_of_payments>9</number_of_payments>
    </ContAuthTxn>
    <TxnDetails>
      <merchantreference>annual0000432432a</merchantreference>
      <capturemethod>ecomm</capturemethod>
    </TxnDetails>
  </Transaction>
</Request>
```

Example Request for an initial charge of \$30, twelve regular payments of \$10 on the first of each month and a final payment of \$20

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>mypasswd</password>
  </Authentication>
  <Transaction>
    <CardTxn>
      <Card>
        <expirydate>01/06</expirydate>
        <pan>444433*****1</pan>
      </Card>
    </CardTxn>
    <ContAuthTxn>
      <method>auth</method>
      <start_date>01/10/2005</start_date>
      <period>monthly</period>
      <amount currency="USD">10.00</amount>
      <number of payments>12</number of payments>
      <FirstPayment>
        <amount currency="USD">30.00</amount>
        <date>01/09/2005</date>
      </FirstPayment>
      <LastPayment>
        <amount currency="USD">20.00</amount>
        <date>31/10/2006</date>
      </LastPayment>
    </ContAuthTxn>
    <TxnDetails>
      <merchantreference>382684637615741</merchantreference>
      <capturemethod>cnp</capturemethod>
    </TxnDetails>
  </Transaction>
</Request>
```

2.2.14.3 Schema Elements for Response

In addition to the elements covered in this section, responses for this service will contain the general response elements outlined in section 2.1.1.3.

2.2.14.3.1 ContAuthTxn

Element Name:	ContAuthTxn
Position	Response

Elements of ContAuthTxn	
Element Name	Description
account_status	Returns information about the status of the account

Example XML element ContAuthTxn

```
<ContAuthTxn>
    <account_status>Account setup. First transaction due to be
        batched on 30/06/2005</account_status>
</ContAuthTxn>
```

2.2.14.4 Example XML Responses

Example XML Response for an account which has been successfully set up

```
<Response>
    <ContAuthTxn>
        <account_status>Account setup. First transaction due to be
            batched on 09/10/2006</account_status>
    </ContAuthTxn>
    <datacash_reference>4200200040373186</datacash_reference>
    <information>Continuous Authority account set up</information>
    <merchantreference>382684637615741</merchantreference>
    <mode>...</mode>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <time>...</time>
</Response>
```

Example XML Response for a card type which is not supported by the service

```
<Response>
  <datacash_reference>4200200040609090</datacash_reference>
  <information>Continuous authority is not supported on
    this card</information>
  <merchantreference>383914408564815</merchantreference>
  <mode>...</mode>
  <reason>CA Not Supported</reason>
  <status>92</status>
  <time>...</time>
</Response>
```

Example XML Response for an invalid start date

```
<Response>
  <datacash_reference>4600200040609088</datacash_reference>
  <information>The date of the first payment must be from
    tomorrow onwards</information>
  <merchantreference>12345612</merchantreference>
  <mode>...</mode>
  <reason>Invalid Date</reason>
  <status>94</status>
  <time>...</time>
</Response>
```

2.2.14.5 Notifications

Whenever a payment is taken from an account, an email will be generated containing the results of the payment(s). The email will contain up to five attachments. Each attachment will take the form of a CSV file and be named one of:

- declined.csv – for [declined](#) payments
- error.csv – payments which generated an error message
- expiring.csv – payments on cards which will expire before the next payment date
- referred.csv – for [referred](#) payments
- authorized.csv – for successful payments

These notifications can also be downloaded from Reporting.

Format for CSV files		
	Name	Description
1	Account Reference	The DataCash reference of the account from which the payment was requested.
2	Merchant Reference	Your reference for the account. As supplied when you setup the account.
3	DataCash Reference	The DataCash reference of the payment that failed. This is not supplied for referred.csv

Example CSV file

```
Account Reference,Merchant Reference,DataCash Reference
37707655,acc000045452,4300200040945208
37699077,SMITH0001245432,4100200040945209
37699081,Green458934,4900200040945210
37701359,qwerty12345,4700200040945211
37709097,rtruier49rsw,4500200040945212
37709891,egr3t434,4300200040945213
```

[Back to Contents](#)

2.3 FRAUD PREVENTION

2.3.1 AVSCV2

A technical introduction to this Service is available on the website:

<http://www.datacash.com/gettingproducts.php?id=Address-and-Security-Code-Verification>

This section covers the elements to be populated with the cv2 and address details. It also covers the sending of standard and extended policy information within the transaction.

This service is utilised by sending a normal Credit and Debit Card Service Request with additional information. This section of documentation assumes the Credit and Debit Card Service has already been integrated and the reader is familiar with it. The Credit and Debit Card Service is described in section 2.2.1.

It may also be used in conjunction with the Pre-Registered Card Service, which is described in section 2.2.11.

2.3.1.1 Schema Elements for Request

In this section the fields that can be submitted when using the AVSCV2 service will be presented, along with example XML for those fields.

Each field will be labelled with the following key:

- O - Optional
- R - Required, field must be presented
- C - Checked, the optional field will be checked if presented
- M - Mandatory if available, if the information is available, it should be presented

The following additional schema elements can be presented when performing an AVSCV2 check:

2.3.1.1.1 Request

- Transaction
 - CardTxn
 - Card
- Cv2Avs – see section 2.3.1.1.2
 - ExtendedPolicy – see section 2.3.1.1.3

2.3.1.1.2 Cv2Avs

This parent element should be populated with the information that you require to be checked. When using the either Standard or Extended Policy, this information is also passed here – only one of these elements should be presented. If you wish to use the default policy, neither element should be presented.

Please remember that if you are only presenting a particular subset of the AVSCV2 data to be checked, you must choose a policy that does not require the all fields to be checked – this would otherwise result in all transactions being rejected.

None of the child elements of Cv2Avs are mandatory. If the cv2 element is presented, the length of its contents will be checked to ensure the number is the correct length for the card type. American Express cards have four digits and all other card types have three. An empty cv2 number will cause the rejection of the transaction.

Element Name:	Cv2Avs
Position	Request.Transaction.CardTxn.Card.Cv2Avs

Elements of Cv2Avs			
Element Name	Description	Values / Limitations	Required
street_address1			
street_address2			
street_address3			
street_address4	The statement address (excluding postcode) of the customer.	Though only the numeric data can be checked, the full details can be submitted if required. Any non-numeric data will be stripped out and the numeric data will be concatenated prior to checking.	C
city	The statement city	Must be alphanumeric.	C
state_province	The statement state/province	Must be alphanumeric.	C
country	The statement country	Use the numeric country codes	C
postcode	The statement postcode	A maximum of 9 alphanumeric characters.	C
cv2	The cv2 number from the card	Must be 4 digits for Amex, or 3 digits for all other card types.	C
policy	The Standard Policy required for the transaction.	1, 2, 3, 5, 6, 7. Please refer to the website for definitions	O ¹
ExtendedPolicy	For use with Extended Policy only. See section 2.3.1.1.3		

¹ Either of the policy or ExtendedPolicy elements can be presented. To use the default policy configured on the account, neither element should be presented. If a transaction contains both standard and extended policy information, it will be rejected.

Example Cv2Avs elements for default, standard and extended policy transactions respectively. In each example, the numeric data for the address and postcode is identical and so would return the same result from the bank.

```

<Cv2Avs>
    <street_address1>1 High Street</street_address1>
    <street_address2>This Town</street_address2>
    <street_address3>Somewhere</street_address3>
    <street_address4>United Kingdom</street_address4>
    <postcode>S01 2CD</postcode>
    <cv2>123</cv2>
</Cv2Avs>

<Cv2Avs>
    <street_address1>1 High Street</street_address1>
    <city>This Town</city>
    <state_province>Somewhere</state_province>
    <country>826</country>
    <postcode>S01 2CD</postcode>
    <cv2>123</cv2>
</Cv2Avs>

<Cv2Avs>
    <street_address1>1 High Street,
        This Town, Somewhere,
        United Kingdom</street_address1>
    <postcode>S01 2CD</postcode>
    <cv2>123</cv2>
    <policy>3</policy>
</Cv2Avs>

<Cv2Avs>
    <street_address1>1</street_address1>
    <postcode>012</postcode>
    <cv2>123</cv2>
    <ExtendedPolicy>...</ExtendedPolicy>
</Cv2Avs>
```

2.3.1.1.3 ExtendedPolicy

When using the extended policy, each element and attribute must be presented.

Element Name:	ExtendedPolicy
Position	Request.Transaction.CardTxn.Card.Cv2Avs.ExtendedPolicy

Elements of ExtendedPolicy			
Element Name	Description	Values / Limitations	Required
cv2_policy			R
postcode_policy	Each of these elements has five attributes – see below.		R
address_policy			R

Attributes for the child elements of ExtendedPolicy				
Attribute Name	Element Name	Description	Values / Limitations	Required
notprovided				R
notchecked	postcode_policy	Specifies whether to accept or reject each of the five possible bank responses for the three individual elements	accept reject	R
matched	cv2_policy			R
notmatched	address_policy			R
partialmatch				R

Example Extended policy element

```
<ExtendedPolicy>
  <cv2_policy notprovided="reject"
    notchecked="accept"
    matched="accept"
    notmatched="reject"
    partialmatch="reject"/>
  <postcode_policy notprovided="reject"
    notchecked="accept"
    matched="accept"
    notmatched="reject"
    partialmatch="accept"/>
  <address_policy notprovided="reject"
    notchecked="accept"
    matched="accept"
    notmatched="reject"
    partialmatch="accept"/>
</ExtendedPolicy>
```

2.3.1.2 XML Example Requests

2.3.1.2.1 Default Policy

Example Default Policy Request

```
<Request>
  <Authentication>
    <client>99000000</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>123456</merchantreference>
      <amount currency="EUR">10.00</amount>
    </TxnDetails>
    <CardTxn>
      <Card>
        <pan>4444*****1111</pan>
        <expirydate>03/04</expirydate>
        <Cv2Avs>
          <street_address1>Flat 7</street_address1>
          <street_address2>89 Jumble
          Street</street_address2>
          <street_address3>Mytown</street_address3>
          <postcode>AV12FR</postcode>
          <cv2>123</cv2>
        </Cv2Avs>
      </Card>
      <method>auth</method>
    </CardTxn>
  </Transaction>
</Request>
```

Example Default Policy with Payment Tokenization Cards

```
<Request>
  <Authentication>
    <client>99000000</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>123456</merchantreference>
      <amount currency="EUR">10.00</amount>
    </TxnDetails>
    <CardTxn>
      <card_details type="preregistered">
        2185999900000001</card_details>
      <Card>
        <Cv2Avs>
          <street_address1>Flat 7</street_address1>
          <street_address2>89 Jumble
          Street</street_address2>
          <street_address3>Mytown</street_address3>
          <postcode>AV12FR</postcode>
          <cv2>123</cv2>
        </Cv2Avs>
      </Card>
      <method>auth</method>
    </CardTxn>
  </Transaction>
</Request>
```

See also section 2.2.9.2.3 for examples using HCC and default policy

2.3.1.2.2 Extended Policy

Example Extended Policy Request

```

<Request>
  <Authentication>
    <client>99000000</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>123456</merchantreference>
      <amount currency="EUR">10.00</amount>
    </TxnDetails>
    <CardTxn>
      <Card>
        <pan>4444*****1111</pan>
        <expirydate>03/04</expirydate>
        <Cv2Avs>
          <street_address1>Flat 7</street_address1>
          <street_address2>89 Jumble
                           Street</street_address2>
          <street_address3>Mytown</street_address3>
          <postcode>AV12FR</postcode>
          <cv2>123</cv2>
          <ExtendedPolicy>
            <cv2_policy notprovided="reject"
                        notchecked="accept"
                        matched="accept"
                        notmatched="reject"
                        partialmatch="reject"/>
            <postcode_policy notprovided="reject"
                            notchecked="accept"
                            matched="accept"
                            notmatched="reject"
                            partialmatch="accept"/>
            <address_policy notprovided="reject"
                            notchecked="accept"
                            matched="accept"
                            notmatched="reject"
                            partialmatch="accept"/>
          </ExtendedPolicy>
        </Cv2Avs>
      </Card>
      <method>auth</method>
    </CardTxn>
  </Transaction>
</Request>

```

Example Extended Policy Request when using a Payment Token in place of the Card details

```

<Request>
  <Authentication>
    <client>99000000</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>123456</merchantreference>
      <amount currency="EUR">10.00</amount>
    </TxnDetails>
    <CardTxn>
      <card_details type="preregistered">
        4800200052555666</card_details>
      <Card>
        <Cv2Avs>
          <street_address1>Flat 7</street_address1>
          <street_address2>89 Jumble
                      Street</street_address2>
          <street_address3>Mytown</street_address3>
          <postcode>AV12FR</postcode>
          <cv2>123</cv2>
          <ExtendedPolicy>
            <cv2_policy notprovided="reject"
                        notchecked="accept"
                        matched="accept"
                        notmatched="reject"
                        partialmatch="reject"/>
            <postcode_policy notprovided="reject"
                            notchecked="accept"
                            matched="accept"
                            notmatched="reject"
                            partialmatch="accept"/>
            <address_policy notprovided="reject"
                           notchecked="accept"
                           matched="accept"
                           notmatched="reject"
                           partialmatch="accept"/>
          </ExtendedPolicy>
        </Cv2Avs>
      </Card>
      <method>auth</method>
    </CardTxn>
  </Transaction>
</Request>

```

See also section 2.2.9.2.3 for examples using HCC and extended policy

2.3.1.3 Schema Elements for Response

AVSCV2 results are returned in the Cv2Avs element for successfully authorised or referred transactions. If a transaction is DECLINED, the AVSCV2 results will not be available either in the XML response or on the Reporting pages.

2.3.1.3.1 Cv2Avs

Element Name:	Cv2Avs
Position	Response.CardTxn

Most child elements and attributes of Cv2Avs are dependent upon whether the default/standard or extended policy checks where used.

Attributes of child elements in Cv2Avs			
Attribute Name	Element Name	Description	Values / Limitations
reversal	cv2avs_status	The result of the reversal request for transactions failing to meet the chosen policy	0 - failed 1 - successful

2.3.1.3.1.1 Standard and Default Policies

Elements of Cv2Avs		
Element Name	Description	Values / Limitations
cv2avs_status	The overall result of the transaction	NO DATA MATCHES ADDRESS MATCH ONLY SECURITY CODE MATCH ONLY ALL MATCH DATA NOT CHECKED
Policy	The policy the transaction was checked against.	

Example Response elements for a transaction checked using either the default or standard policy

```

<Cv2Avs>
  <cv2avs_status reversal="1">SECURITY CODE MATCH
    ONLY</cv2avs_status>
  <policy>3</policy>
</Cv2Avs>

<Cv2Avs>
  <cv2avs_status reversal="0">ADDRESS MATCH ONLY</cv2avs_status>
  <policy>2</policy>
</Cv2Avs>

<Cv2Avs>
  <cv2avs_status>ALL MATCH</cv2avs_status>
  <policy>6</policy>
</Cv2Avs>

```

2.3.1.3.1.2 For Extended Polices

Elements of Cv2Avs		
Element Name	Description	Values / Limitations
cv2avs_status	The overall result of the transaction	ACCEPTED REJECTED
address_result		One of: notprovided notchecked matched notmatched partialmatch
cv2_result		
postcode_result		
address_policy	The policy the transaction was checked against. These have the same format as in the Request – see section 2.3.1.1.3	
cv2_policy		
postcode_policy		

Attributes of child elements within Cv2Avs			
Attribute Name	Element Name	Description	Values / Limitations
numeric	address_result cv2_result postcode_result	The numerical result returned by the bank for each element.	0 – not provided 1 – not checked 2 – matched 4 - not matched 8 – partial match

Example Response Elements for extended policy transactions

```

<Cv2Avs>
  <address_policy notprovided="reject"
    notchecked="accept"
    matched="accept"
    notmatched="reject"
    partialmatch="accept"/>
  <address_result numeric='0'>notprovided</address_result>
  <cv2_policy notprovided="reject"
    notchecked="accept"
    matched="accept"
    notmatched="reject"
    partialmatch="reject"/>
  <cv2_result numeric='2'>matched</cv2_result>
  <cv2avs_status reversal='1'>REJECTED</cv2avs_status>
  <postcode_policy notprovided="reject"
    notchecked="accept"
    matched="accept"
    notmatched="reject"
    partialmatch="accept"/>
  <postcode_result numeric='4'>notmatched</postcode_result>
</Cv2Avs>

<Cv2Avs>
  <address_policy notprovided="reject"
    notchecked="accept"
    matched="accept"
    notmatched="reject"
    partialmatch="accept"/>
  <address_result numeric='2'>matched</address_result>
  <cv2_policy notprovided="reject"
    notchecked="accept"
    matched="accept"
    notmatched="reject"
    partialmatch="reject"/>
  <cv2_result numeric='1'>not checked</cv2_result>
  <cv2avs_status>ACCEPTED</cv2avs_status>
  <postcode_policy notprovided="reject"
    notchecked="accept"
    matched="accept"
    notmatched="reject"
    partialmatch="accept"/>
  <postcode_result numeric='2'>matched</postcode_result>
</Cv2Avs>

```

2.3.1.4 XML Example Responses

2.3.1.4.1 Successful AVSCV2 Responses

An example response for transactions successfully passing the standard or default policy AVSCV2 check

```
<Response>
  <CardTxn>
    <authcode>3956</authcode>
    <card_scheme>VISA</card_scheme>
    <Cv2Avs>
      <cv2avs_status>SECURITY CODE MATCH ONLY</cv2avs_status>
      <policy>2</policy>
    </Cv2Avs>
  </CardTxn>
  <datacash_reference>4000000098765888</datacash_reference>
  <merchantref>...</merchantref>
  <mode>...</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>...2</time>
</Response>
```

An example response for transactions successfully passing the extended policy AVSCV2 check

```
<Response>
  <CardTxn>
    <authcode>4382</authcode>
    <card_scheme>VISA</card_scheme>
    <Cv2Avs>
      <address_policy notprovided="reject"
                      notchecked="accept" matched="accept"
                      notmatched="reject" partialmatch="accept"/>
      <address_result numeric='2'>matched</address_result>
      <cv2_policy notprovided="reject"
                  notchecked="accept" matched="accept"
                  notmatched="reject" partialmatch="reject"/>
      <cv2_result numeric='1'>not checked</cv2_result>
      <cv2avs_status>ACCEPTED</cv2avs_status>
      <postcode_policy notprovided="reject"
                      notchecked="accept" matched="accept"
                      notmatched="reject" partialmatch="accept"/>
      <postcode_result numeric='2'>matched</postcode_result>
    </Cv2Avs>
  </CardTxn>
  <datacash_reference>4000000098765433</datacash_reference>
  <merchantref>...</merchantref>
  <mode>...</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>...</time>
</Response>
```

2.3.1.4.2 AVSCV2 Declined Responses

An example response for transactions failing to match the standard or default policy chosen

```
<Response>
  <CardTxn>
    <authcode>3956</authcode>
    <card_scheme>VISA</card_scheme>
    <Cv2Avs>
      <cv2avs_status reversal="1">ADDRESS MATCH ONLY
      </cv2avs_status>
      <policy>3</policy>
    </Cv2Avs>
  </CardTxn>
  <datacash_reference>4000000098765889</datacash_reference>
  <merchantref>123998</merchantref>
  <mode>LIVE</mode>
  <reason>CV2AVS DECLINED</reason>
  <status>7</status>
  <time>1234567812</time>
</Response>
```

An example response for transactions failing to match the extended policy chosen

```
<Response>
  <CardTxn>
    <authcode>2893</authcode>
    <card_scheme>VISA</card_scheme>
    <country>United Kingdom</country>
    <Cv2Avs>
      <address_policy notprovided="reject" notchecked="accept"
                      matched="accept" notmatched="reject"
                      partialmatch="accept"/>
      <address_result numeric="1">notchecked</address_result>
      <cv2_policy notprovided="reject" notchecked="accept"
                  matched="accept" notmatched="reject"
                  partialmatch="reject"/>
      <cv2_result numeric="2">matched</cv2_result>
      <cv2avs_status reversal="1">REJECTED</cv2avs_status>
      <postcode_policy notprovided="reject"
                       notchecked="accept" matched="accept" notmatched="reject"
                       partialmatch="accept"/>
      <postcode_result
        numeric="4">notmatched</postcode_result>
    </Cv2Avs>
  </CardTxn>
  <datacash_reference>4000000098765432</datacash_reference>
  <merchantref>123456</merchantref>
  <mode>LIVE</mode>
  <reason>CV2AVS DECLINED</reason>
  <status>7</status>
  <time>1234567890</time>
</Response>
```

2.3.1.4.3 Referred Transactions

If a transaction is referred by the bank, AVSCV2 information will still be available. If the transaction failed AVSCV2 checking, then any attempt to submit an `authorize_referral_request` or `fulfill` transaction with a new authorisation code will be declined.e

An example response for referred transactions successfully passing the standard or default policy AVSCV2 check

```
<Response>
  <CardTxn>
    <authcode>CALL AUTH CENTRE</authcode>
    <card_scheme>VISA</card_scheme>
    <Cv2Avs>
      <cv2avs_status>SECURITY CODE MATCH ONLY</cv2avs_status>
      <policy>2</policy>
    </Cv2Avs>
  </CardTxn>
  <datacash_reference>4000000098765888</datacash_reference>
  <merchantref>...</merchantref>
  <mode>...</mode>
  <reason>REFERRED</reason>
  <status>7</status>
  <time>...2</time>
</Response>
```

Example XML Response for a successful authorize_referral_request

```
<Response>
  <CardTxn>
    <Cv2Avs>
      <cv2avs_status>SECURITY CODE MATCH ONLY</cv2avs_status>
      <policy>2</policy>
    </Cv2Avs>
  </CardTxn>
  <datacash_reference>3900200000000001</datacash_reference>
  <merchantreference>...</merchantreference>
```

```

<mode>...</mode>
<reason>ACCEPTED</reason>
<status>1</status>
<time>1071567356</time>

</Response>

```

An example response for referred transactions failing to match the standard or default policy chosen

```

<Response>
  <CardTxn>
    <authcode>CALL AUTH CENTRE</authcode>
    <card_scheme>VISA</card_scheme>
    <Cv2Avs>
      <cv2avs_status>ADDRESS MATCH ONLY
      </cv2avs_status>
      <policy>3</policy>
    </Cv2Avs>
  </CardTxn>
  <datacash_reference>4000000098765889</datacash_reference>
  <merchantref>...</merchantref>
  <mode>...</mode>
  <reason>REFERRED</reason>
  <status>7</status>
  <time>1234567812</time>
</Response>

```

Example XML Response for a declined authorize_referral_request

```

<Response>
  <CardTxn>
    <Cv2Avs>
      <cv2avs_status>ADDRESS MATCH ONLY
      </cv2avs_status>
      <policy>3</policy>
    </Cv2Avs>
  </CardTxn>
  <datacash_reference>4000000098765889</datacash_reference>
  <merchantreference>...</merchantreference>

```

```
<mode>...</mode>
<reason>CV2AVS DECLINED</reason>
<status>7</status>
<time>1071567356</time>

</Response>
```

2.3.1.4.4 Declined Transactions

If a transaction is declined by the bank, no AVSCV2 information will be available. The transaction Response will therefore be the same as a declined transaction with no AVSCV2 checking. Please refer to section 2.2.1.4 for examples.

2.3.2 URU®

A technical introduction to this Service is available on the website:

<http://www.datacash.com/gettingproducts.php?id=Age-Identity-Verification-Overview>

This Service allows an individual's identity and personal information to be authenticated against a number of databases, and the results of previous authentications to be retrieved.

Some data elements used in this service use the Universal Unique ID format (UUID). This format is xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx and unless otherwise indicated is hexadecimal.

2.3.2.1 Schema Elements for Request

In this section the required fields for URU® requests will be presented, along with example XML for those fields.

As the two transaction types do not use the same fields, each field will be labelled with the following key:

- O – Optional
- R – Required
- X – Excluded, presenting this field for the transaction type will cause the transaction to fail.

It should be noted that these indicate the status of the field within the DPG only. If you are integrating this service, we strongly advise that you discuss with URU® the fields that must be used in order to gain optimum results from the service.

Authentications

The information required to process an identity authentication - `authenticate` - is passed in several distinct places in the schema:

2.3.2.1.1 Request

- Authentication - **section 2.1.1.2**
- Transaction
 - TxnDetails - contains the merchantreference, **section 2.3.2.1.2**
 - URUTxn - contains the data relating to the individual being authenticated, along with the method `authenticate`, **section 2.3.2.1.3**
 - Basic - basic details about the customer, **section 2.3.2.1.4**
 - UKData - customer details for UK customers, **section 2.3.2.1.5**
 - Address1, Address2, Address3 & Address4 - details for up to four addresses, **section 2.3.2.1.6**
 - Passport - details for UK passports, **section 2.3.2.1.7**
 - Electric - details of the electricity bill, **section 2.3.2.1.8**
 - Telephone - details about the telephone number, **section 2.3.2.1.9**
 - Driver - details of the driving licence, **section 2.3.2.1.10**
 - Birth - details about the place of birth of the customer and mother's maiden name, **section 2.3.2.1.11**
 - USData - customer details for US customers. Includes social security number, driving licence and telephone number. **Section 2.3.2.1.5**
 - USAAddress1, USAAddress2, USAAddress3 & USAAddress4 - details for up to four addresses, **section 2.3.2.1.13**
 - Employment - employment history, **section 2.3.2.1.14**
 - CreditDebitCard - credit and debit card details, **section 2.3.2.1.15**
 - InternationalPassport - details of the machine-readable international passport, **section 2.3.2.1.16**
 - Authenticate - details of the URU account to be used for the transaction, **section 2.3.2.1.17**

Log Requests

The log of a previous authentication may be retrieved using this method, `get_log_by_authentication_id`

- Request
 - Authentication - **section 2.1.1.2**
 - Transaction
 - TxnDetails - contains the merchantreference, **section 2.3.2.1.2**.
 - URUTxn - contains the query reference for the authentication being retrieved, along with the method `get_log_by_authentication_id`, **section 2.3.2.1.3**

2.3.2.1.2 TxnDetails

Element Name: TxnDetails
Position(s) Request.Transaction

Elements of TxnDetails				
Element Name	Description	Values / Limitations	Authenticate	Log Request
merchantreference	A unique reference number for each transaction	Between six and thirty alphanumeric characters	R	R

Example XML for TxnDetails elements

```
<TxnDetails>
    <merchantreference>123402ABCDE12</merchantreference>
</TxnDetails>
```

2.3.2.1.3 URUTxn

This is the base element for submission of URU specific information. As well as holding information about an individual for authentication, it contains the transaction type and, in the case of log requests, the original authentication ID.

In the case of authentication transactions, only the `method` element is mandatory. However, at least one of the other user data sub-elements must be present.

Element Name: URUTxn
Position(s) Request.Transaction

Elements of URUTxn				
Element Name	Description	Values / Limitations	Authenticate	Log Request
method	Identifies the transaction type as an authentication, or a log request	authenticate get_log_by_authentication_id	R	R
Basic	Basic information about an individual's name and date of birth	See section 2.3.2.1.4	O	X
Employment	Information about an individual's employment	See section 2.3.2.1.14	O	X
CreditDebitCard	Information about an individual's Payment Card	See section 2.3.2.1.15	O	X
InternationalPassport	Information about an individual's international machine-readable passport	See section 2.3.2.1.16	O	X
guid	The query reference of the original transaction	UUID	X	R
UKData	Information for UK residents	See section 2.3.2.1.5	O	X
USData	Information for US Residents	See section 2.3.2.1.12	O	X
Authenticate	Information about the URU account which is to be used for the transaction	See section 2.3.2.1.17	O	X

Example XML Element URUTxn for an authentication

```
<URUTxn>
  <method>authenticate</method>
  <Basic>...</Basic>
  <UKData>...</UKData>
  <Employment>...</Employment>
  <CreditDebitCard>...</CreditDebitCard>
  <InternationalPassport>...</InternationalPassport>
</URUTxn>
```

Example XML Element URUTxn for a log request

```
<URUTxn>
  <method>get_log_by_authentication_id</method>
  <guid>01234567-89ab-dcef-0123-456789abcdef</guid>
</URUTxn>
```

2.3.2.1.4 Basic

This element holds information about an individual's name and date of birth. Unless otherwise specified, all string fields accept up to 256 characters. All sub-fields are optional.

Element Name: Basic
Position(s) Request.Transaction.URUTxn

Elements of Basic		
Element Name	Description	Values / Limitations
dob_day	The day of the month of the individual's birthday	int 1-31
dob_month	The month of the year of the individual's birthday	int 1-12
dob_year	The year of the individual's birth	int 0-9999
forename	The individual's forename	string
gender	The individual's gender	Male Female Unknown Unspecified
middle_initial	The individual's middle initial(s)	string
surname	The individual's surname	string
title	The title of the individual (e.g. Mr, Miss)	string

Example XML Element Basic

```
<Basic>
  <dob_day>14</dob_day>
  <dob_month>12</dob_month>
  <dob_year>1972</dob_year>
  <forename>Joe</forename>
  <gender>Male</gender>
  <middle_initial>F</middle_initial>
  <surname>Bloggs</surname>
  <title>Mr</title>
</Basic>
```

2.3.2.1.5 UKData

This contains a number of complex elements – Address1 through to Address4, Electric, Telephone, Passport, Driver and Birth. All elements are complex (i.e. contain sub elements) and optional.

Element Name:	UKData
Position(s)	Request.Transaction.URUTxn

Elements of UKData		
Element Name	Description	Values / Limitations
Address1	See section 2.3.2.1.6	
Address2		
Address3		
Address4		
Passport	See section 2.3.2.1.7	
Electric	See section 2.3.2.1.8	
Telephone	See section 2.3.2.1.9	
Driver	See section 2.3.2.1.10	
Birth	See section 2.3.2.1.11	

Example XML Element UKData

```
<UKData>
  <Address1>...</Address1>
  <Address2>...</Address2>
  <Passport>...</Passport>
  <Electric>...</Electric>
  <Telephone exdirectory="no">
    ...
  </Telephone>
  <Driver>...</Driver>
</UKData>
```

Example XML Element UKData

```
<UKData>
  <Passport>...</Passport>
  <Telephone exdirectory="yes">...</Telephone>
  <Driver>...</Driver>
</UKData>
```

2.3.2.1.6 Address#n

Up to four addresses may be specified for addresses in the UK, labelled Address1, Address2, Address3 and Address4. Each takes the same format. Unless otherwise specified, all string fields accept up to 256 characters. All sub-fields are optional.

Element Name: Address#n
Position(s) Request.Transaction.URUTxn.UKData

Elements of Address#n		
Element Name	Description	Values / Limitations
postcode	The postcode of the address	string
building_name	The building name of the address	string
building_no	The building number of the address	string
sub_building	The sub-building name/number of the address	string
organisation	The organisation name of the address	string
street	The street name of the address	string
sub_street	The sub-street name of the address	string
town	The town name of the address	string
district	The district name of the address	string
first_year_of_residence	The first year the individual was at this address	int 0-9999
last_year_of_residence	The last year the individual was at this address	int 0-9999

Example XML Element Address1

```
<Address1>
  <postcode>AB1 2CD</postcode>
  <building_name>Duncodin</building_name>
  <building_no>1</building_no>
  <sub_building>3F2</sub_building>
  <organisation>Company Ltd</organisation>
  <street>Main Street</street>
  <sub_street>East End</sub_street>
  <town>Littleton</town>
  <district>Central</district>
  <first_year_of_residence>2003</first_year_of_residence>
  <last_year_of_residence>2004</last_year_of_residence>
</Address1>
```

Example XML Element Address2

```
<Address2>
  <postcode>AB1 2CD</postcode>
  <building_no>8</building_no>
  <street>High Street</street>
  <town>My Town</town>
</Address2>
```

2.3.2.1.7 Passport

This block holds information about an individual's UK passport. International passports should use the InternationalPassport element, section 2.3.2.1.16. The passport number is split into six parts, and submitted as number1 to number6. All sub-fields are optional.

Element Name: Passport
Position(s) Request.Transaction.URUTxn

Elements of Passport		
Element Name	Description	Values / Limitations
number1	The first 10 digits of the passport number (9 digit number plus checksum)	string, 10 characters
number2	The next 3 digits of the passport number (Issuing State code)	string, 3 characters
number3	The next 7 digits of the passport number, (date of birth plus checksum)	string, 7 characters

number4	The next 1 character of the passport number (gender)	string, 1 character
number5	The next 7 characters of the passport number (passport expiry date plus checksum)	string, 7 characters
number6	The final 2 digits of the passport number (checksum digits)	string, 2 characters
expiry_day	The day of the month of the passport expiry	int, 1-31
expiry_month	The month of the year of the passport expiry	int, 1-12
expiry_year	The year of the passport expiry	int, 0-9999

Example XML Passport elements

```

<Passport>
  <number1>1234567890</number1>
  <number2>GBR</number2>
  <number3>6201014</number3>
  <number4>M</number4>
  <number5>1231237</number5>
  <number6>12</number6>
  <expiry_day>12</expiry_day>
  <expiry_month>5</expiry_month>
  <expiry_year>2010</expiry_year>
</Passport>

<Passport>
  <number3>7506083</number3>
  <number4>F</number4>
  <number6>12</number6>
  <expiry_day>23</expiry_day>
  <expiry_month>9</expiry_month>
  <expiry_year>2008</expiry_year>
</Passport>

```

2.3.2.1.8 Electric

This block holds information about an individual's electricity bill. The bill number is split into four parts and submitted as `number1` to `number4`. Unless otherwise specified, strings accept a maximum of 256 characters. All sub-fields are optional.

Element Name:	Electric
Position(s)	Request.Transaction.URUTxn

Elements of Electric		
Element Name	Description	Values / Limitations
number1	The first 2 digits of the bill number	string, 2 characters
number2	The next 4 digits of the bill number	string, 4 characters
number3	The next 4 digits of the bill number	string, 4 characters
number4	The final 3 digits of the bill number	string, 3 character
mail_sort	The mail sort code of the bill address	string
postcode	The postcode the bill is sent to	string

Example XML Electric elements

```

<Electric>
  <number1>12</number1>
  <number2>1234</number2>
  <number3>1234</number3>
  <number4>123</number4>
  <mail_sort>AB123</mail_sort>
  <postcode>AB12 3CD</postcode>
</Electric>

<Electric>
  <number1>56</number1>
  <number2>8318</number2>
  <number3>1234</number3>
</Electric>

```

2.3.2.1.9 Telephone

This block holds information about an individual's telephone number. An optional attribute will be accepted that indicates whether or not the number is ex-directory. Unless otherwise specified, strings accept a maximum of 256 characters. All sub-fields are optional.

Element Name:	Telephone
Position(s)	Request.Transaction.URUTxn

Attributes of Telephone		
Attribute Name	Description	Values / Limitations
exdirectory	Whether or not the telephone is ex-directory	yes no

Elements of Telephone		
Element Name	Description	Values / Limitations
Number	The telephone number, including STD code	String. May be specified with or without space between STD code and number
active_month	The month the telephone number became active	int, 1-12
active_year	The year the telephone number became active	int, 0-9999

Example XML Telephone elements

```

<Telephone ex-directory="yes">
    <number>0123 4567890</number>
    <active_month>12</active_month>
    <active_year>2000</active_year>
</Telephone>

<Telephone ex-directory="no">
    <number>0123 4567892</number>
</Telephone>

```

2.3.2.1.10 Driver

This block holds information about an individual's driving licence. The licence number is split into four parts and submitted as `number1` to `number4`. Unless otherwise specified, strings accept a maximum of 256 characters. All sub-fields are optional.

Element Name: Driver
Position(s) Request.Transaction.URUTxn

Elements of Driver		
Element Name	Description	Values / Limitations
number1	The first 5 digits of the licence number	string, 5 characters
number2	The next 6 digits of the licence number	string, 6 characters
number3	The next 3 digits of the licence number	string, 3 characters
number4	The last 2 digits of the licence number	string, 2 characters
mail_sort	The mail sort code of the licence address	string
postcode	The postcode of the licence address	string

Example XML Driver elements

```

<Driver>
  <number1>HENDE</number1>
  <number2>123456</number2>
  <number3>123</number3>
  <number4>12</number4>
  <mail_sort>AB123</mail_sort>
  <postcode>AB1 2CD</postcode>
</Driver>

<Driver>
  <number2>589231</number2>
  <number4>48</number4>
</Driver>

```

2.3.2.1.11 Birth

This block holds information about an individual's birth details. Both elements are optional.

Element Name: Birth
Position(s) Request.Transaction.URUTxn.UKData

Elements of Birth		
Element Name	Description	Values / Limitations
mothers_maiden_name	Mother's maiden name	String, max 256 characters
country_of_birth	Country of birth	ENGLANDWALES OTHER UNSPECIFIED

Example XML Birth elements

```
<Birth>
  <mothers_maiden_name>Smith</mothers_maiden_name>
  <country_of_birth>UNSPECIFIED</country_of_birth>
</Birth>
```

2.3.2.1.12 USData

This contains various details for customers based in the USA.

Element Name: USData
Position(s) Request.Transaction.URUTxn

Elements of USData		
Element Name	Description	Values / Limitations
USAddress1	See 2.3.2.1.6	
USAddress2		
USAddress3		
USAddress4		
social_security_number	Social security number	string
telephone_number	Telephone number	string
driving_licence	The driver's licence number	string

Example XML USData elements

```

<USData>
  <USAddress1>...</USAddress1>
  <USAddress2>...</USAddress2>
  <USAddress3>...</USAddress3>
  <USAddress4>...</USAddress4>
  <social_security_number>string</social_security_number>
  <telephone_number>string</telephone_number>
  <driving_licence>string</driving_licence>
</USData>

```

2.3.2.1.13 USAddress#n

Up to four addresses may be specified for addresses in the USA, labelled USAddress1, USAddress2, USAddress3 and USAddress4. Each takes the same format. Unless otherwise specified, all string fields accept up to 256 characters. All sub-fields are optional.

Element Name: USAddress#n
Position(s) Request.Transaction.URUTxn.USData

Elements of Address#n		
Element Name	Description	Values / Limitations
building1	Address line 1	string
building2	Address line 2	string
building3	Address line 3	string
building4	Address line 4	string
zip_code	Postal code of the building	string
street	The street name of the building	string
city_town	Town or City	string
state_district	The state or district	string

Example XML USAddress#n elements

```

<USAddress1>
  <building1>Apt 312</building1>
  <building2>Block B</building2>
  <building3>Manhattan Heights</building3>
  <building4>44532</building4>
  <zip_code>CHF6666</zip_code>
  <street>Fake Street</street>
  <city_town>New York</city_town>
  <state_district>New York</state_district>
</USAddress1>
```

2.3.2.1.14 Employment

This block holds information about an individual's employment status and residence.

Element Name:	Employment
Position(s)	Request.Transaction.URUTxn

Elements of Employment		
Element Name	Description	Values / Limitations
residence_type	The residence status of the individual	HomeOwnerOutright HomeOwnerMortgage Tenant LivingWithRelatives
employment_status	The employment status of the individual	FTPPerm PTPerm SelfEmployed Retired Homemaker Unemployed Student ArmedForces
current_time	The length of time the individual has been in their current employment	Years0to2 Years2to5 Years5Plus

Example XML Employment elements

```

<Employment>
  <residence_type>HomeOwnerMortgage</residence_type>
  <employment_status>FTPPerm</employment_status>
  <current_time>Years0to2</current_time>
</Employment>

<Employment>
  <residence_type>Tenant</residence_type>
  <employment_status>Student</employment_status>
</Employment>

```

2.3.2.1.15 CreditDebitCard

This block holds information about an individual's payment card. If this block is present, it must take an attribute specifying the card type. Although the block is optional as a whole, if it is present then many of its sub-fields become mandatory as indicated below.

Element Name:	CreditDebitCard
Position(s)	Request.Transaction.URUTxn

Attributes of CreditDebitCard			
Attribute Name	Description	Values / Limitations	Mandatory
cardtype	The card type	VISA MASTERCARD DELTA AMEX JCB MAESTRO DINERS SOLO ELECTRON	R

Elements of CreditDebitCard			
Element Name	Description	Values / Limitations	Mandatory
card_number	The card number	string, up to 19 characters	R
card_expiry_date	The card expiry date	MMYY	R
card_issue_number	The card issue number	int, 1-9	O
card_verification_code	The CV2 number	int, 3 digits for non-Amex, 4 for Amex	R

Example XML Element CreditDebitCard

```
<CreditDebitCard cardtype="SOLO">
  <number>6759000000000000</number>
  <card_expiry_date>1209</card_expiry_date>
  <card_issue_number>3</card_issue_number>
  <card_verification_code>123</card_verification_code>
</CreditDebitCard>
```

2.3.2.1.16 InternationalPassport

This block holds information about an individual's international machine-readable passport. This element can be used for UK machine-readable passports if required, instead of the `Passport` element - section 2.3.2.1.7.

The passport number is split into nine parts and entered as `number1` to `number9`. All sub-fields are optional.

Element Name: InternationalPassport
Position(s) Request.Transaction.URUTxn

Elements of InternationalPassport		
Element Name	Description	Values / Limitations
number1	The first 9 digits of the passport number	string, 9 characters
number2	The next 1 digit of the passport number, a checksum digit for number1	string, 1 character
number3	The next 3 digits of the passport number, Issuing State code	string, 3 characters
number4	The next 7 digits of the passport number, date of birth plus checksum	string, 7 characters
number5	The next 1 digit of the passport number, gender	string, 1 character
number6	The next 7 digits of the passport number, passport expiry date and checksum	string, 7 characters
number7	The next 14 digits of the passport number	string, 14 characters
number8	The next 1 digit of the passport number	string, 1 character
number9	The last 1 digit of the passport number, checksum of entire passport number	string, 1 character
expiry_day	The day of the month of the passport expiry	int, 1-31
expiry_month	The month of the year of the passport expiry	int, 1-12
expiry_year	The year the passport expires	int, 0-9999
country_of_origin	The country of origin of the passport	string, a maximum of 256 characters

Example XML InternationalPassport elements

```

<InternationalPassport>
    <number1>123456789</number1>
    <number2>1</number2>
    <number3>123</number3>
    <number4>1234567</number4>
    <number5>1</number5>
    <number6>1234567</number6>
    <number7>12345678901234</number7>
    <number8>1</number8>
    <number9>1</number9>
    <expiry_day>21</expiry_day>
    <expiry_month>12</expiry_month>
    <expiry_year>2010</expiry_year>
    <country_of_origin>UK</country_of_origin>
</InternationalPassport>

<InternationalPassport>
    <number7>12345678901234</number7>
    <country_of_origin>Germany</country_of_origin>
</InternationalPassport>

```

2.3.2.1.17 Authenticate

This block holds information about the URU account that is to be used for this transaction. This allows the specification of a URU account other than the default account that may be set up. The block itself is optional, but if it is presented all sub fields are required.

Element Name: Authenticate
Position(s) Request.Transaction.URUTxn

Elements of Authenticate		
Element Name	Description	Values / Limitations
profile_id	The profile identifier – the UUID of the profile	String
account_name	The email address that acts as the account user name	String
password	The URU login password	String

Example XML Authenticate element

```
<Authenticate>
  <profile_id>332c87be-aac8-42e3-9039-6b93ce1a04b8
  </profile_id>
  <account_name>no.one@nowhere.com</account_name>
  <password>password</password>
</Authenticate>
```

2.3.2.2 XML Example Requests

Example XML Request for an authentication, with full details

```
<Request>
  <Authentication>
    <password>password</password>
    <client>99000000</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>12345678</merchantreference>
    </TxnDetails>
    <URUTxn>
      <method>authenticate</method>
      <Basic>
        <forename>Jane</forename>
        <middle_initial>R</middle_initial>
        <surname>Smith</surname>
        <dob_day>18</dob_day>
        <dob_month>12</dob_month>
        <dob_year>1972</dob_year>
        <gender>Female</gender>
        <title>Ms</title>
      </Basic>
      <UKData>
        <Address1>
          <postcode>AB1 2CD</postcode>
          <building_name>Duncodin</building_name>
          <building_no>1</building_no>
          <sub_building>3F2</sub_building>
          <organisation>Company Ltd</organisation>
          <street>Main Street</street>
          <sub_street>East End</sub_street>
          <town>Littleton</town>
          <district>Central</district>
          <first_year_of_residence>2003 </first_year_of_residence>
        </Address1>
        <Address2>
          <postcode>EF3 4GH</postcode>
          <building_no>1</building_no>
          <street>North Street</street>
          <town>Bighton</town>
          <first_year_of_residence>2002 </first_year_of_residence>
          <last_year_of_residence>2003</last_year_of_residence>
        </Address2>
      </UKData>
    </URUTxn>
  </Transaction>
</Request>
```

```

<Address3>
  <postcode>IJ5 6KL</postcode>
  <building_no>1</building_no>
  <street>South Street</street>
  <town>Upper Bigton</town>
  <first_year_of_residence>2001 </first_year_of_residence>
  <last_year_of_residence>2002</last_year_of_residence>
</Address3>
<Address4>
  <postcode>MN7 8OP</postcode>
  <building_no>1</building_no>
  <street>Market Street</street>
  <town>Lower Littleton</town>
  <first_year_of_residence>2000 </first_year_of_residence>
  <last_year_of_residence>2001</last_year_of_residence>
</Address4>
<Passport>
  <number1>1234567890</number1>
  <number2>USA</number2>
  <number3>1234567</number3>
  <number4>M</number4>
  <number5>1234567</number5>
  <number6>12</number6>
  <expiry_day>31</expiry_day>
  <expiry_month>12</expiry_month>
  <expiry_year>2010</expiry_year>
</Passport>
<Electric>
  <number1>12</number1>
  <number2>1234</number2>
  <number3>1234</number3>
  <number4>123</number4>
  <mail_sort>12345</mail_sort>
  <postcode>AB1 2CD</postcode>
</Electric>
<Telephone exdirectory="no">
  <number>0123 456 7890</number>
  <active_month>12</active_month>
  <active_year>2000</active_year>
</Telephone>
<Driver>
  <number1>12345</number1>
  <number2>123456</number2>
  <number3>123</number3>
  <number4>1234</number4>
  <mail_sort>6789</mail_sort>
  <postcode>AB1 2CD</postcode>
</Driver>
<Birth>
  <mothers_maiden_name>Smith</mothers_maiden_name>
  <country_of_birth>ENGLANDWALES</country_of_birth>
</Birth>
</UKData>
<Employment>
  <residence_type>Tenant</residence_type>
  <employment_status>FTPerm</employment_status>
  <current_time>Years0to2</current_time>

```

```
</Employment>
<CreditDebitCard cardtype="VISA">
    <card_number>5521480000000003</card_number>
    <card_expiry_date>1009</card_expiry_date>
    <card_issue_number>1</card_issue_number>
    <card_verification_code>321</card_verification_code>
</CreditDebitCard>
<InternationalPassport>
    <number1>123456789</number1>
    <number2>1</number2>
    <number3>USA</number3>
    <number4>1234567</number4>
    <number5>M</number5>
    <number6>1234567</number6>
    <number7>12345678901234</number7>
    <number8>1</number8>
    <number9>1</number9>
    <expiry_day>21</expiry_day>
    <expiry_month>2</expiry_month>
    <expiry_year>2009</expiry_year>
    <country_of_origin>Scotland</country_of_origin>
</InternationalPassport>
</URUTxn>
</Transaction>
</Request>
```

Example XML Request for an authentication, with partial details

```
<Request>
  <Authentication>
    <password>password</password>
    <client>99000000</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>123456789</merchantreference>
    </TxnDetails>
    <URUTxn>
      <method>authenticate</method>
      <Basic>
        <title>Dr</title>
        <forename>Ian</forename>
        <surname>Smith</surname>
        <dob_day>29</dob_day>
        <dob_month>11</dob_month>
        <dob_year>1968</dob_year>
      </Basic>
      <UKData>
        <Address1>
          <postcode>AB1 2CD</postcode>
          <building_no>46</building_no>
          <street>Middle Road</street>
          <town>Leeds</town>
          <first_year_of_residence>1992
            </first_year_of_residence>
        </Address1>
        <Telephone exdirectory="no">
          <number>0123 456 7890</number>
          <active_month>12</active_month>
          <active_year>1992</active_year>
        </Telephone>
      </UKData>
    </URUTxn>
  </Transaction>
</Request>
```

Example XML Request for a log request

```
<Request>
  <Authentication>
    <client>99000000</client>
    <password>password</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>1357924680</merchantreference>
    </TxnDetails>
    <URUTxn>
      <method>get_log_by_authentication_id</method>
      <guid>12345678-9abc-def0-1234-56789abcdef0</guid>
    </URUTxn>
  </Transaction>
</Request>
```

2.3.2.3 Schema Elements for Response

In addition to the elements covered in this section, responses for this service will contain the general response elements, covered in section 2.1.1.3

2.3.2.3.1 URUTxn

This element holds all URU specific data. Elements may be relevant only to authentications, to log requests, or to both.

Element Name:	URUTxn
Position(s)	Response

Elements of URUTxn				
Element Name	Description	Values / Limitations	Authentication	Log Request
authentication_id	The query reference of the authentication being retrieved	UUID	N	Y
authentication_count	The number of authentications submitted on the URU account	int	Y	Y
customer_ref	The merchantreference element submitted with the authentication	string	Y	Y
profile_id	The profile id submitted with the authentication	UUID	Y	Y
profile_version	The major version number of the profile used for the authentication	int	Y	Y
profile_revision	The minor version number of the profile used for the authentication	int	Y	Y
profile_name	The name of the profile used for the authentication	string	N	Y

query_ref	An authentication reference for future log requests	UUID	Y	N
state	The state the profile was in at time of authentication	PS_TST PS_PRE PS_EFF PS_RET	N	Y
Results	The results of the authentication	See section 2.3.2.3.2	Y	Y
UserData	The original user data supplied with the authentication		N	Y*
score	The score for this request based on values set by the URU account holder	int	Y	Y
cc_basic_score	Credit card basic score – based on values set by the URU account holder	int	Y	Y
timestamp	Value indicating the date/time of the authentication	string	Y	Y
SanctionMatches	Match information returned from the Sanctions Items check (if processed)	See Section 2.3.2.3.4	Y	Y
band_text	Values returned when scoring and banding are included in the URU profile	string	Y	Y

* The URU service definition provides the facility to return the original user data. However, it is not clear whether this functionality has been implemented at the URU side. If the DPG receives this data in the URU response, it will be returned in the DataCash response.

Example XML Element URUTxn for an authentication Response

```
<URUTxn>
  <authentication_count>2</authentication_count>
  <authentication_id>332c87be-aac8-42e3-9039-6b93cela04b8
    </authentication_id>
  <cc_basic_score>5</cc_basic_score>
  <customer_ref>123401</customer_ref>
  <profile_id>332c87be-aac8-42e3-9039-6b93cela04b8</profile_id>
  <profile_revision>0</profile_revision>
  <profile_version>1</profile_version>
  <Results>
    ...
  </Results>
  <score>2</score>
  <timestamp>_SOAP_DATE_</timestamp>
</URUTxn>
```

Example XML Element URUTxn for a log request Response

```
<URUTxn>
  <authentication_id>0bcb541d-8b7a-46a0-86d2-1e6a521fef5a
    </authentication_id>
  <customer_ref>1234501</customer_ref>
  <profile_id>332c87be-aac8-42e3-9039-6b93cela04b8
    </profile_id>
  <profile_name>Default</profile_name>
  <profile_revision>0</profile_revision>
  <profile_version>1</profile_version>
  <Results>
    ...
  </Results>
  <state>PS_EFF</state>
</URUTxn>
```

2.3.2.3.2 Results

This element holds a collection of result blocks. In an authentication response, these results will each be contained in URUResult2 blocks, whilst in the response from a log request they will be in URULogResult blocks.

Element Name: Results
Position(s) Response.URUTxn

Elements of Results				
Element Name	Description	Values / Limitations	Authentication	Log Request
URUResult2	Information about a single aspect of one of the checks performed by the URU server during an authentication	See section 2.3.2.3.3	Y	N
URULogResult	Information about a single aspect of one of the checks performed by the URU server during an authentication	See section 2.3.2.3.3	N	Y

Example XML Element Results for a authentication transaction Response

```

<Results>
  <URULogResult2>
    ...
  </URULogResult2>
  <URULogResult2>
    ...
  </URULogResult2>
  <URULogResult2>
    ...
  </URULogResult2>
  <URULogResult2>
    ...
  </URULogResult2>
</Results>

```

Example XML Element Results for a log request Response

```

<Results>
  <URULogResult>
    ...
  </URULogResult>
  <URULogResult>
    ...
  </URULogResult>
</Results>

```

2.3.2.3.3 URUResult2, URULogResult

Both the `URUResult2` element, found in the response from an authentication, and the `URULogResult` element, found in the response from a log request, have the same basic format, and so are dealt with together.

Element
Name: URUResult2

Position(s)

Response.URUTxn.Results

Element
Name: URULogResult

Position(s)

Response.URUTxn.Results

Elements of URUResult2, URULogResult

Element Name	Description	Values / Limitations
uru_id	A number representing the check to which this part-result belongs	int
code	A numeric code describing one aspect of the check referred to by the <code>uru_id</code>	int
text	A textual description of the code within the context of the check referred to by the <code>uru_id</code>	string

Example XML Element URUResult2 for a log request

```
<URUResult2>
  <code>1001</code>
  <text>Address #1 is valid</text>
  <uru_id>1</uru_id>
</URUResult2>
```

Example XML Element URULogResult for a log request

```
<URULogResult>
  <code>1001</code>
  <text>Address #1 is valid</text>
  <uru_id>1</uru_id>
</URULogResult>
```

2.3.2.3.4 SanctionMatches

This element holds a collection of SanctionMatch blocks.

Element Name:	SanctionMatches
Position(s)	Response.URUTxn

Elements of SanctionMatches

Element Name	Description	Values / Limitations
SanctionMatch	Information about each of the sanction match checks performed	See section 2.3.2.3.4

Example XML Element SanctionMatches

```
<SanctionMatches>
  <SanctionMatch>...</SanctionMatch>
  <SanctionMatch>...</SanctionMatch>
</SanctionMatches>
```

2.3.2.3.5 SanctionMatch

Element containing information regarding each of the sanction match checks performed. Will only appear where the URU account is configured for perform these checks.

Element Name: SanctionMatch
Position(s) Response.URUTxn.SanctionMatches

Elements of SanctionMatch		
Element Name	Description	Values / Limitations
id	A value representing the check to which this part-result belongs	string
url	The URL of the URU transaction	string
score	A numeric score describing the check referred to by the id	int

Example XML Elements SanctionMatch

```

<SanctionMatch>
  <id>Second result</id>
  <score>4</score>
  <url>https://localhost:8180/axis</url>
</SanctionMatch>

<SanctionMatch>
  <id>First result</id>
  <score>2</score>
  <url>https://localhost:8180/axis</url>
</SanctionMatch>

```

2.3.2.4 XML Example Responses

Example XML Response for an authentication

```

<Response>
  <datacash_reference>4000900012345671</datacash_reference>
  <merchantreference>sanction_match</merchantreference>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1134662619</time>
  <URUTxn>
    <authentication_count>2</authentication_count>
    <authentication_id>5C64C6F0-6D84-11DA-AC9A-9A8DBF8F5DE4
  </URUTxn>
</Response>

```

```

</authentication_id>
<cc_basic_score>5</cc_basic_score>
<customer_ref>sanction_match</customer_ref>
<profile_id>332c87be-aac8-42e3-9039-6b93ce1a04b8</profile_id>
<profile_revision>0</profile_revision>
<profile_version>1</profile_version>
<Results>
    <URUResult2>
        <code>260</code>
        <text>Part 6 was not sufficiently supplied by user</text>
        <uru_id>3</uru_id>
    </URUResult2>
    <URUResult2>
        <code>1001</code>
        <text>Address #1 is valid</text>
        <uru_id>4</uru_id>
    </URUResult2>
    <URUResult2>
        <code>110</code>
        <text>Telephone number not specified by user</text>
        <uru_id>6</uru_id>
    </URUResult2>
    <URUResult2>
        <code>101</code>
        <text>Drivers license number was not specified in full</text>
        <uru_id>7</uru_id>
    </URUResult2>
</Results>
<SanctionMatches>
    <SanctionMatch>
        <id>Second result</id>
        <score>4</score>
        <url>https://localhost:8180/axis</url>
    </SanctionMatch>
    <SanctionMatch>
        <id>First result</id>
        <score>2</score>
        <url>https://localhost:8180/axis</url>
    </SanctionMatch>
</SanctionMatches>
<score>2</score>
<timestamp>2005-12-15T16:03:41.568Z</timestamp>
</URUTxn>
</Response>

```

Example XML Response for a log response

```

<Response>
    <datacash_reference>4600900012345673</datacash_reference>
    <merchantreference>123403</merchantreference>
    <mode>TEST</mode>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <time>1134662157</time>
    <URUTxn>
        <authentication_id>26238d77-02fe-4177-8752-029a0d384e2a

```

```
</authentication_id>
<customer_ref>customerRef_datacash1@uru.com</customer_ref>
<profile_id>332c87be-aac8-42e3-9039-6b93ce1a04b8</profile_id>
<profile_name>Default</profile_name>
<profile_revision>0</profile_revision>
<profile_version>1</profile_version>
<Results>
    <URULogResult>
        <code>110</code>
        <text>Telephone number not specified by user</text>
        <uru_id>6</uru_id>
    </URULogResult>
    <URULogResult>
        <code>101</code>
        <text>Drivers license number was not specified in
            full</text>
        <uru_id>7</uru_id>
    </URULogResult>
    <URULogResult>
        <code>260</code>
        <text>Part 6 was not sufficiently supplied by user</text>
        <uru_id>3</uru_id>
    </URULogResult>
    <URULogResult>
        <code>1001</code>
        <text>Address #1 is valid</text>
        <uru_id>4</uru_id>
    </URULogResult>
</Results>
<state>PS_EFF</state>
<UserData>
    <Basic>
        <dob_day>10</dob_day>
        <dob_month>12</dob_month>
        <dob_year>1980</dob_year>
        <forename>Joe</forename>
        <gender>Male</gender>
        <surname>Bloggs</surname>
    </Basic>
    <CreditDebitCard>
        <card_expiry_date>0109</card_expiry_date>
        <card_issue_number>1</card_issue_number>
        <card_number>633300000000005</card_number>
        <card_verification_code>123</card_verification_code>
        <cardtype>SOLO</cardtype>
    </CreditDebitCard>
    <Employment>
        <current_time>Years0to2</current_time>
        <employment_status>ArmedForces</employment_status>
        <residence_type>HomeOwnerMortgage</residence_type>
    </Employment>
    <InternationalPassport>
        <number1>123456789</number1>
        <number2>1</number2>
        <number3>123</number3>
        <number4>1234567</number4>
        <number5>1</number5>
```

```
<number6>1234567</number6>
<number7>12345678901234</number7>
<number8>1</number8>
<number9>1</number9>
</InternationalPassport>
<UKData>
<Address1>
  <building_no>10</building_no>
  <postcode>AB1 2CD</postcode>
  <street>Easy Street</street>
  <town>Bington</town>
</Address1>
<Driver>
  <number1>12345</number1>
  <number2>123456</number2>
  <number3>123</number3>
  <number4>12</number4>
</Driver>
<Electric>
  <mail_sort>1234</mail_sort>
  <number1>12</number1>
  <number2>1234</number2>
  <number3>1234</number3>
  <number4>123</number4>
  <postcode>AB1 2CD</postcode>
</Electric>
<Passport>
  <expiry_day>1</expiry_day>
  <expiry_month>1</expiry_month>
  <expiry_year>2010</expiry_year>
  <number1>1234567890</number1>
  <number2>123</number2>
  <number3>1234567</number3>
  <number4>1</number4>
  <number5>1234567</number5>
  <number6>12</number6>
</Passport>
<Telephone>
  <active_month>5</active_month>
  <active_year>2002</active_year>
  <exdirectory>false</exdirectory>
</Telephone>
</UKData>
</UserData>
</URUTxn>
</Response>
```

2.3.3 ReD Fraud Screening

A technical introduction to this Service is available on the website:

<http://www.datacash.com/gettingproducts.php?id=Fraud-Screening>

This section of the documentation assumes the reader is familiar with the Credit and Debit Card Service, which is discussed in section 2.2.1.

2.3.3.1 Schema Elements for Request

In this section the required fields for each transaction type will be presented, along with example XML for those fields. The following key will be used:

- R – Required
- MS – Market Sector merchants should consider this field as mandatory
- O – Optional for Market Sector merchants
- I – Ignored if presented

If you are using a bespoke model – instead of the Market Sector model – please ensure that you are providing the correct data to enable each rule to be triggered.

Authorisation Requests

For authorisation requests, the *additional* information for fraud screening is passed in several different places in the schema. Most of these are located within the Request.Transaction.TxnDetails element:

2.3.3.1.1 TxnDetails

- Order – see section 2.3.3.1.11
 - Customer – see section 2.3.3.1.6
 - Address – the customer's address, section 2.3.3.1.3
 - Company for corporate orders, section 2.3.3.1.5
 - Address – the company's address, section 2.3.3.1.5
 - CustomerHistory – see section 2.3.3.1.7
 - BillingAddress – the card billing address, if different from the customer's address, section 2.3.3.1.4
- Shipping – see section 2.3.3.1.14
- OrderDetails – see section 2.3.3.1.11
 - LineItem – see section 3.3.2.1.6
 - Shipping – see section 2.3.3.1.14
- Recipient – see section 2.3.3.1.13
 - Address – see section 2.3.3.1.3
 - LineItem – see section 3.3.2.1.6
 - Shipping – see section 2.3.3.1.14

If you are also using the AVSCV2 Service, an additional element may be submitted in the Cv2Avs element, please see section 2.3.1.1.2.

Over-riding Challenges

To over-ride a Fraud Challenged Response, information is passed in these places in the schema:

2.3.3.1.2 Request

- Authentication – see section 2.1.1.2
- Transaction
 - HistoricTxn – see section 2.5.2.3.2.1

2.3.3.1.3 Address

The Address element is found in several different contexts, as the customer, customer's company and the recipients may all have different addresses. For orders being shipped to more than one address, each Address can be specified using multiple Recipient.Address elements. The CustomerAddress element will only be fraud screened if there are no Recipient elements present.

Element Name:	Address
Position(s)	Request.Transaction.TxnDetails.Order.Customer Request.Transaction.TxnDetails.Order.Customer.Company Request.Transaction.TxnDetails.Order.Recipient

Elements of Address			
Element Name	Description	Values / Limitations	
apartment	Apartment, Suite or Flat Number	Maximum 6 characters	O
streetaddress	street address	Maximum 30 characters	R
city	City or town	Maximum 30 characters	R
postcode	Postcode	Maximum 30 characters	R
region	US State or Canadian Province code	For addresses in USA and Canada only, must be excluded for other countries	O
country	Country	Use the numeric country codes	R
moreaddress	Additional space for address if required	Maximum 30 characters	O

Example XML element for Address

```
<Address>
  <city>New York</city>
  <country>840</country>
  <postcode>12345</postcode>
  <region>NY</region>
  <streetaddress>123 Some Street </streetaddress>
</Address>
```

Example XML element for Address

```
<Address>
  <city>Dover</city>
  <country>826</country>
  <moreaddress>Kent</moreaddress>
  <postcode>XX12 3AA</postcode>
  <streetaddress>56 Kings Road</streetaddress>
</Address>
```

Example XML element for Address

```
<Address>
  <apartment>3/1</apartment>
  <city>Edinburgh</city>
  <country>826</country>
  <moreaddress>Great Michael House</moreaddress>
  <postcode>EH6 7EZ</postcode>
  <streetaddress>14 Links Place</streetaddress>
</Address>
```

2.3.3.1.4 BillingAddress

This element must be provided if the customer's address is not the address to which the card is registered – for example if the customer has moved house & has not yet informed their bank.

If you are performing the AVS check, the card address needs to be entered in both the Cv2AVS (section 2.3.3.1.8) and BillingAddress elements.

Element Name:	BillingAddress
Position(s)	Request.Transaction.TxnDetails.Order

The child elements within this parent are the same as for Address (section 2.3.3.1.3).

Example XML elements for BillingAddress

```
<BillingAddress>
    <apartment>6</apartment>
    <city>London</city>
    <country>826</country>
    <moreaddress>Newtown</moreaddress>
    <postcode>W1 2ZZ</postcode>
    <streetaddress>Main Street</streetaddress>
</BillingAddress>
```

2.3.3.1.5 Company

This element should be provided if the order is a company purchase.

Element Name: Company

Position(s) Request.Transaction.TxnDetails.Order.Customer

Elements of Company			
Element Name	Description	Values / Limitations	
Address	See section 2.3.3.1.3		O
fax	The company fax number	1-12 numerics only	O
name	The company name	Maximum 30 characters	O
purchase_description	A description of corporate purchase	Maximum 160 characters	O
telephone	The company phone number	1-12 numerics only	O

Example XML element for Company

```
<Company>
    <Address>
        ...
    </Address>
    <fax>111122223001</fax>
    <purchase_description>Hardware</purchase_description>
    <name>My Company Ltd</name>
    <telephone>111122223000</telephone>
</Company>
```

2.3.3.1.6 Customer

This element contains various details about the customer.

Element Name: Customer
Position(s) Request.Transaction.TxnDetails.Order

Elements of Customer			
Element Name	Description	Values / Limitations	
Address	see 0		
age	Age	minimum value: 0 maximum value: 150 This value will be ignored if presented for Recipient	O
alt_telephone	Home telephone number	A maximum of twelve numerics International numbers: include Country code, exclude access code	MS
Company	See section 2.3.3.1.5		O
CustomerHistory	See section 2.3.3.1.7		O
date_of_birth	date of birth	CCYY-MM-DD	O
email	email address	60 characters max	MS
fax	Fax number	as for alt_telephone	O
forename	First name	30 characters max	MS
ip_address	The IP address of the computer placing the order	Minimum 7 characters (e.g. 1.2.3.4) Maximum of 17 (e.g. 255.255.255.255)	MS
max_age	Maximum age	minimum value: 0 maximum value: 150 This value will be ignored if presented for Recipient	O
middlename	Middle name or initial	30 characters max	O
min_age	Minimum age	minimum value: 0 maximum value: 150 This value will be ignored if presented for Recipient	O

national_id	Nationally recognised individual identification number	12 characters maximum	O
pctime	time on the Customer's PC	HH:MM:SS	O
salutation	Title	5 characters max	O
surname	Surname / family name	30 characters max	MS
telephone	Work / day phone number of Customer, or main number for Company / Recipient	A maximum of twelve numerics International numbers: include Country code, exclude the International dialling/access code	O

Example XML element for Customer

```
<Customer>
  <age>43</age>
  <email>me@hotmail.com</email>
  <Address>...</Address>
  <surname>Brown</surname>
  <forename>Mark</forename>
  <salutation>Dr</salutation>
  <ip_address>1.2.3.255</ip_address>
</Customer>
```

Example XML element for Customer

```
<Customer>
  <date_of_birth>1965-01-31</date_of_birth>
  <Address>...</Address>
  <surname>Patel</surname>
  <forename>Sanjay</forename>
  <CustomerHistory>...</CustomerHistory>
</Customer>
```

Example XML element for Customer

```
<Customer>
  <min_age>18</min_age>
  <max_age>25</max_age>
  <forename>Mary</forename>
  <surname>O'Connor</surname>
  <telephone>...</telephone>
  <alt_telephone>...</alt_telephone>
</Customer>
```

2.3.3.1.7 CustomerHistory

This element enables information about the customer history to be passed. Most of the elements within this parent will only be applicable for e-Commerce transaction – for example `cookies` and `website`, though MoTo merchants may find some elements of use (e.g. `previous_cust`). Flagging MoTo and e-Commerce is covered in section 2.3.3.1.15.

Element Name:	CustomerHistory
Position(s)	Request.Transaction.TxnDetails.Order.Customer

Elements of CustomerHistory			
Element Name	Description	Values / Limitations	
account	username, userid etc.	max 64	O
cookies	cookies enabled on customers PC	XML Boolean	O
forgot_password	did customer require the password to be reset	XML Boolean	O
formfill	The degree to which the online purchase form filled automatically	E - All fields filled automatically M - All fields filled automatically but some fields modified manually N - None, all manually filled	O
loyalty	Has customer registered for a loyalty program	XML Boolean	O
password_access	Did customer enter a password to access the account?	XML Boolean	O
previous_cust	Has customer previously bought products	XML Boolean	O
promos	Is customer registered for promotions	XML Boolean	O
website	URL of website the purchase was made on. Particularly useful for merchants with more than one website	max 60	O

Example XML element for CustomerHistory

```
<CustomerHistory>
  <previous_cust>false</previous_cust>
</CustomerHistory>
```

Example XML element for CustomerHistory

```
<CustomerHistory>
  <account>johnsmith99</account>
  <promos>false</promos>
  <loyalty>false</loyalty>
  <previous_cust>true</previous_cust>
  <password_access>true</password_access>
  <forgot_password>false</forgot_password>
  <formfill>E</formfill>
  <website>http://www.computershop.com</website>
  <cookies>true</cookies>
</CustomerHistory>
```

Example XML element for CustomerHistory

```
<CustomerHistory>
  <account>userXXX</account>
  <previous_cust>false</previous_cust>
  <formfill>N</formfill>
  <website>http://www.XXX.com</website>
</CustomerHistory>
```

2.3.3.1.8 Cv2Avs

If you are using both the ReD and AVSCV2 services, the `cv2_present` element must be considered to be mandatory. The other child elements of Cv2Avs are described in section 2.3.1.1.2.

Element Name:	Cv2Avs
Position(s)	Request.Transaction.CardTxn.Card

Elements of Cv2Avs		
Element Name	Description	Values / Limitations
cv2_present	Gauges whether the customer had the opportunity to enter the cv2 number	0 - requested from the customer, but not provided 1 - entered by customer 2 - not legible on the card 8 - cv2 details were not requested on the website 9 - card had no cv2 number

Example XML elements for Cv2Avs

```

<Cv2Avs>
  <cv2>123</cv2>
  <cv2_present>1</cv2_present>
</Cv2Avs>

<Cv2Avs>
  <street_address1>1 High Street</street_address1>
  <street_address2>This Town</street_address2>
  <street_address3>Somewhere</street_address3>
  <street_address4>United Kingdom</street_address4>
  <postcode>S01 2CD</postcode>
  <cv2_present>0</cv2_present>
</Cv2Avs>

<Cv2Avs>
  <street_address1>54 Kings Road</street_address1>
  <street_address2>This Village</street_address2>
  <street_address3>Derbyshire</street_address3>
  <postcode>DE1 1AA</postcode>
  <cv2_present>8</cv2_present>
  <ExtendedPolicy>...</ExtendedPolicy>
</Cv2Avs>

```

2.3.3.1.9 HistoricTxn

This element is required to over-ride Fraud Challenged Responses. It should not be presented for authorisations.

Element Name:	HistoricTxn
Position(s)	Request.Transaction

Elements of HistoricTxn			
Element Name	Description	Values / Limitations	
method	The transaction method	accept_fraud	R
reference	The datacash_reference of the challenged transaction	Must be a successfully authorised challenge received within the previous seven days	R

Example XML element for HistoricTxn

```

<HistoricTxn>
  <method>accept_fraud</method>
  <reference>4900200040157301</reference>
</HistoricTxn>

```

2.3.3.1.10 LineItem

This element enables information about each item within the order to be presented. Multiple `LineItem` elements can be presented. Each distinct product within the order should be submitted within its own `LineItem` element.

Element Name:	LineItem
Position(s)	Request.Transaction.TxnDetails.Order.OrderDetails Request.Transaction.TxnDetails.Order.Recipient

Elements of LineItem			
Element Name	Description	Values / Limitations	
description	a description of the product	160 max	O
gift	For gifts, any message attached to item. May instead be specified for the order as a whole in the <code>Order</code> element (section 2.3.3.1.11)	160 max	O
Shipping	See section 2.3.3.1.14		O

Attributes of LineItem			
Attribute Name	Attribute of	Description	Values / Limitations
card	gift	gift occasion	See Appendix 5 for values
category	LineItem	category of offer	S - Standard P - Promotion
manpartno	LineItem	manufacturers part number	30 characters max, inc spaces
manufacturer	LineItem	manufacturers name	50 characters max, inc spaces
product_code	LineItem	your product code	18 characters max, inc spaces
quantity	LineItem	number of items ordered	12 digits max
sku	LineItem	product ID or SKU	18 characters max, inc spaces
type	LineItem	Product type	P - physical goods D - digital goods, e.g. software C - digital content, e.g. images S - shareware M - a mixture of the above
unit_price	LineItem	Cost per item	If a decimal point is included, it must be followed by two digits

upc	LineItem	product UPC	12 characters max, inc spaces	0
wrapped	gift	Is the item/order wrapped?	XML Boolean	0

Example XML for LineItem with one item

```
<LineItem category="S" quantity="1" type="P" unit_price="499.99">
    <gift card="G" wrapped="false">Congratulations!</gift>
</LineItem>
```

Example XML for LineItem with eight copies of one product

```
<LineItem category="S" quantity="8" type="P" unit_price="5.58"/>
```

Example XML for an order with two products

```
<LineItem category="S" manpartno="part number 1J800"
manufacturer="Dell" product_code="DELL-1J800" quantity="1" type="P"
unit_price="900">
    <description>Desktop</description>
    <Shipping>
        ...
    </Shipping>
</LineItem>
<LineItem category="P" manpartno="CD0001" manufacturer="Sony"
product_code="SONY-CD0001" quantity="20" sku="sku 2" unit_price="0.50"
upc="upc 2">
    <description>CD boxes</description>
    <Shipping>
        ...
    </Shipping>
</LineItem>
```

2.3.3.1.11 Order

The majority of the extra information is contained within this element.

Elements of Order			
Element Name	Description	Values / Limitations	
Customer	See section 2.3.3.1.6	MS	
BillingAddress	See section 2.3.3.1.4	MS	
Recipient	See section 2.3.3.1.13	O	
OrderDetails	See section 2.3.3.1.12	O	
Shipping	See section 2.3.3.1.14	O	
returns_allowed	Indicates if you enable the customer to return goods to you	XML Boolean	O
gift	For gifts, any message attached to the entire order. May also be specified on a per item basis in the LineItem element (section 3.3.2.1.6)	160 max	O

Attributes of Order			
Attribute Name	Attribute of	Description	Values / Limitations
wrapped	gift	Is the item/order wrapped?	XML Boolean O
card	gift	Gift occasion	See Appendix 5 O

Example XML elements for Order

```

<Order>
  <Customer>...</Customer>
  <OrderDetails>...</OrderDetails>
  <returns_allowed>true</returns_allowed>
  <gift card="B" wrapped="true"/>
</Order>

<Order>
  <Customer>...</Customer>
  <Recipient>...</Recipient>
  <BillingAddress/>
</Order>

```

2.3.3.1.12 OrderDetails

Element Name:	OrderDetails
Position(s)	Request.Transaction.TxnDetails.Order

Elements of OrderDetails		
Element Name	Description	Values / Limitations
handling	Shipping and handling charges	O
subtotal	Order subtotal, without tax, shipping and handling	If a decimal point is included it must be followed by two digits
tax	Order tax	O
LineItem	See section 3.3.2.1.6	O

Example XML element for OrderDetails

```
<OrderDetails>
  <handling>5.00</handling>
  <subtotal>26.99</subtotal>
  <tax>5.60</tax>
  <LineItem>...</LineItem>
  <LineItem>...</LineItem>
</OrderDetails>
```

2.3.3.1.13 Recipient

This element enables information about the recipients of the order to be submitted. For example, if the customer is placing the order on behalf of a third party, details of that person can be passed for screening. Orders that are being shipped to several people can be presented by using one Recipient element for each person.

Element Name:	Recipient
Position(s)	Transaction.TxnDetails.Order.Recipient

Elements in the PartyType box are common to both Customer and Recipient and are described in section 2.3.3.1.6

Elements of Recipient		
Element Name	Description	Values / Limitations
Address	See 2.3.3.1.3	R
age		I
alt_telephone		O
email	See 2.3.3.1.6	O
fax		O

forename		O
max_age		I
middlename		O
min_age		I
salutation		O
surname		O
telephone		O
LineItem	See 3.3.2.1.6	O

Example XML element for Recipient

```
<Recipient>
  <Address>...</Address>
  <forename>Julia</forename>
  <tracking_number>123</tracking_number>
  <instructions>A</instructions>
  <comments>FAO Mrs Jane White</comments>
</Recipient>
```

2.3.3.1.14 Shipping

This element enables information about the method of shipment to be entered. It may be presented in one of three places in the schema.

Element Name: Shipping

Position(s)	Request.Transaction.TxnDetails.Order.OrderDetails.LineItem
	Request.Transaction.TxnDetails.Order.Recipient.LineItem
	Request.Transaction.TxnDetails.Order

If all items within the order are being shipped within the same parcel, the Order.Shipping element should be used. If the items are being shipped to the customer as they become available, the OrderDetails.LineItem.Shipping elements should be used for each item within the order. For orders which are being shipped to several different recipients, the Recipient.LineItem.Shipping element should be used for each item within the order.

Elements of Shipping		
Element Name	Description	Values / Limitations
carrier	Shipment carrier for order/item	F - FedEx P - USPS U - UPS L - Purolator G - Greyhound D - DHL O - Other
comments	Additional text for shipment of order/item	160 max
instructions	When the shipment is to be made	C - when order is complete A - as items become available D - on a specified date
method	The delivery method	N - Next Day/Oversight T - Two Day Service W - Three Day Service C - Lowest Cost D - Carrier designated by Customer I - International M - Military P - Store delivery service O - Other
tracking_number	shipment tracking number for item/order	max 19

Example XML element for Shipping

```

<Shipping>
  <method>N</method>
  <carrier>O</carrier>
  <tracking_number>123</tracking_number>
  <instructions>A</instructions>
  <comments>FAO Mrs Jane White</comments>
</Shipping>

```

2.3.3.1.15 TxnDetails

Element Name:	TxnDetails
Position(s)	Request.Transaction

Elements of TxnDetails			
Element Name	Description	Values / Limitations	
transactionsource	The environment in which the transaction was made. Particularly important if a single ReD account is used to screen transactions from several sources	ecommerce mail-order/telephone-order	O
Order	0		R

Example XML elements for TxnDetails

```

<TxnDetails>
    <transactionsource>ecommerce</transactionsource>
    <Order>...</Order>
</TxnDetails>

<TxnDetails>
    <transactionsource>mail-order/telephone-order</transactionsource>
    <Order>...</Order>
</TxnDetails>

```

2.3.3.2 Example XML Request

Example XML Request where the order is being shipped to two different recipients

```

<Request>
  <Authentication>...</Authentication>
  <Transaction>
    <TxnDetails>
      <Order>
        <Customer>
          <salutation>Mrs</salutation>
          <forename>J</forename>
          <surname>Brown</surname>
          <Address>
            <apartment>1</apartment>
            <streetaddress>10 Front Street</streetaddress>
            <city>Edinburgh</city>
            <region>Lothian</region>
            <country>826</country>
            <postcode>EH9 9ZZ</postcode>
          </Address>
          <email>j.brown@compuserve.com</email>
          <alt_telephone>44 131 999 8888</alt_telephone>
        </Customer>
        <Recipient>
          <forename>Jenny</forename>
          <surname>Black</surname>
          <Address>
            <streetaddress>26 The High Street</streetaddress>
            <city>Cardiff</city>
            <country>826</country>
          </Address>
          <LineItem manufacturer="CO Q" quantity="1"
                    unit_price="20.00">
            <description>Present</description>
            <gift card="R" wrapped="true">Happy Christmas</gift>
          </LineItem>
        </Recipient>
        <Recipient>
          <forename>Laura</forename>
          <surname>Green</surname>
          <Address>
            <streetaddress>99 Royal York Crescent</streetaddress>
            <city>London</city>
            <country>826</country>
          </Address>
          <LineItem manufacturer="ABC" quantity="2"
                    unit_price="10.00">
            <description>Present</description>
            <gift card="R" wrapped="true">Happy Christmas</gift>
          </LineItem>
        </Recipient>
      </Order>
    </TxnDetails>
  </Transaction>
</Request>

```

```

        </Order>
    </TxnDetails>
    <CardTxn>...</CardTxn>
</Transaction>
</Request>

```

Example XML Request where ReD and AVSCV2 checks are being used

```

<Request>
    <Authentication>...</Authentication>
    <Transaction>
        <TxnDetails>
            <Order>
                <Customer>
                    <salutation>Ms</salutation>
                    <forename>Sandra</forename>
                    <middlename>K</middlename>
                    <surname>Littlejohn</surname>
                    <Address>
                        <streetaddress>37 Queen Street</streetaddress>
                        <city>Oxford</city>
                        <country>826</country>
                        <postcode>OX20 8FP</postcode>
                    </Address>
                    <email>slittlejohn@ox.co.uk</email>
                    <alt_telephone>0870 1234 4566</alt_telephone>
                    <ip_address>48.12.3.4</ip_address>
                </Customer>
                <OrderDetails>...</OrderDetails>
            </Order>
            ...
        </TxnDetails>
        <CardTxn>
            <Card>
                <Cv2Avs>
                    <cv2_present>1</cv2_present>
                    <cv2>453</cv2>
                    <street_address1>37 Queen Street</street_address1>
                    <street_address2>Oxford</street_address2>
                    <postcode>OX20 8FP</postcode>
                    <ExtendedPolicy>...</ExtendedPolicy>
                </Cv2Avs>
                <pan>4444*****11</pan>
                <expirydate>12/06</expirydate>
            </Card>
        </CardTxn>
    </Transaction>
</Request>

```

Example XML Request where the billing address is not the same as the customer's address

```
<Request>
  <Authentication>...</Authentication>
  <Transaction>
    <TxnDetails>
      <Order>
        <Customer>
          <salutation>Mr</salutation>
          <forename>Joe</forename>
          <middlename>K</middlename>
          <surname>Brown</surname>
          <Address>
            <apartment>1</apartment>
            <streetaddress>10 Front Street</streetaddress>
            <city>Edinburgh</city>
            <region>Lothian</region>
            <country>826</country>
            <postcode>EH9 9ZZ</postcode>
          </Address>
          <email>joe.brown@compuserve.com</email>
          <alt_telephone>44 131 999 8888</alt_telephone>
          <ip_address>231.0.0.1</ip_address>
        </Customer>
        <BillingAddress>
          <apartment>6</apartment>
          <streetaddress>Main Street</streetaddress>
          <moreaddress>Newtown</moreaddress>
          <city>London</city>
          <country>826</country>
          <postcode>W1 2ZZ</postcode>
        </BillingAddress>
      </Order>
    ...
    </TxnDetails>
    <CardTxn/>
  </Transaction>
</Request>
```

Example XML Request element to override a fraud challenged transaction

```

<Request>
  <Authentication>
    <client>21859999</client>
    <password>thepassword</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>accept_fraud</method>
      <reference>9910670336394837</reference>
    </HistoricTxn>
  </Transaction>
</Request>

```

2.3.3.3 Schema Elements for Response

When a transaction is screened using this service, the general Response elements (section 2.1.1.3) and CardTxn elements (section 2.2.1.3.1) elements will be present. An additional element will also be present for authorisation Responses.

2.3.3.3.1 EbitGuard

This element contains the results of the fraud screening.

Element Name:	EbitGuard
Position(s)	Response.CardTxn

Elements of EbitGuard			
Element Name	Description	Values	
orderid	The ReD order ID. This uniquely identifies the order		
fraud_status	The overall result of the transaction	Accept Deny Challenge	

Attributes for Elements of EbitGuard			
Attribute Name	Attribute of element	Description	Values
score	fraud_status	The ReD return code	4 digit numeric

Example XML elements for EbitGuard

```

<EbitGuard>
  <fraud_status score="0150">Accept</fraud_status>
  <orderid>000000000000000000000000000000001060764550</orderid>
</EbitGuard>

<EbitGuard>
  <fraud_status score="0800">Deny</fraud_status>
  <orderid>00000100000ADC000000001118324936</orderid>
</EbitGuard>

```

2.3.3.4 Example Responses

Example XML Response XML element for fraud declined transaction

```

<Response>
  <CardTxn>
    <card_scheme>VISA Delta</card_scheme>
    <country>United States</country>
    <EbitGuard>
      <fraud_status score="0200">Deny</fraud_status>
      <orderid>000000000000000000000000000000001060764550</orderid>
    </EbitGuard>
  </CardTxn>
  <datacash_reference>4700200040952107</datacash_reference>
  <merchantreference>...</merchantreference>
  <mode>...</mode>
  <reason>FRAUD DECLINED 0200</reason>
  <status>7</status>
  <time>...</time>
</Response>

```

Example XML Response XML element for fraud challenged transaction, authorised by the bank

```

<Response>
  <CardTxn>
    <authcode>278052</authcode>
    <EbitGuard>
      <fraud_status score="0600">Challenge</fraud_status>
      <orderid>000000000000000000000000000000001060764524</orderid>
    </EbitGuard>
    ...
  </CardTxn>
  <datacash_reference>4600200040952400</datacash_reference>
  <merchantreference>0000007</merchantreference>
  <mode>...</mode>
  <reason>FRAUD CHALLENGED 0600</reason>
  <status>7</status>
  <time>...</time>
</Response>

```

Example XML Response XML element for fraud challenged transaction, declined by the bank

```

<Response>
  <CardTxn>
    <authcode>NOT AUTHORISED</authcode>
    <EbitGuard>
      <fraud_status score="0600">Challenge</fraud_status>
      <orderid>00000000000000000000000001060764524</orderid>
    </EbitGuard>
    ...
  </CardTxn>
  <datacash_reference>460020004050007</datacash_reference>
  <merchantreference>00000007</merchantreference>
  <mode>...</mode>
  <reason>DECLINED</reason>
  <status>7</status>
  <time>...</time>
</Response>

```

Example XML Response XML element for an accepted transaction

```

<Response>
  <CardTxn>
    <authcode>743065</authcode>
    <card_scheme>VISA</card_scheme>
    <country>United Kingdom</country>
    <EbitGuard>
      <fraud_status score="0100">Accept</fraud_status>
      <orderid>00000000000000000000000001060694599</orderid>
    </EbitGuard>
  </CardTxn>
  <datacash_reference>4600200040500130</datacash_reference>
  <merchantreference>00000006</merchantreference>
  <mode>LIVE</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>...</time>
</Response>

```

Example XML. A successful accept_fraud transaction

```

<Response>
  <datacash_reference>4600200040952400</datacash_reference>
  <merchantreference>4600200040952400</merchantreference>
  <mode>...</mode>
  <reason>ACCEPT FRAUD CHALLENGE OK</reason>
  <status>1</status>
  <time>...</time>
</Response>

```

Example XML. A failed accept_fraud

```
<Response>
    <datacash_reference>4300200040500382</datacash_reference>
    <merchantreference>4600200040500007</merchantreference>
    <mode>...</mode>
    <reason>accept_fraud attempted on CV2AVS DECLINED
        transaction</reason>
    <status>19</status>
    <time>...</time>
</Response>
```

2.3.4 3-D Secure, with DataCash MPI

This service enables you to perform a 3-D Secure check using the [DataCash MPI](http://www.datacash.com/gettingproducts.php?id=3D-Secure-MPI-Service) and then authorize the payment via the DPG. A technical introduction to this Service is available on the website: <http://www.datacash.com/gettingproducts.php?id=3D-Secure-MPI-Service>

The DataCash MPI service is utilised by sending a normal Credit and Debit Card Service Request with several pieces of extra information. This section of the documentation assumes the reader is familiar with the Credit and Debit Card Service, as described in section 2.2.1.

It is also possible to use this service in conjunction with the Pre-Registered Card Service, as outlined in section 2.2.11.

The following key will be used in this section of the document:

- R – Required
- M –Mandatory if 3D-Secure check is to be performed
- O – Optional

Cardholder Verification Check

The `auth` and `pre` transaction types require the same information about the transaction to be provided.

This data is passed using the following schema elements:

- Request
 - Authentication – section 2.1.1.2
 - Transaction
 - CardTxn - details about the card, section 2.2.1.1.2
 - TxnDetails - contains details of the transaction, section 2.2.1.1.3
 - ThreeDSecure – contains all the details required to initiate the 3-D Secure check, section 2.3.4.1.1
 - Browser – details of the browser the cardholder is using, section 2.3.4.1.2

Authorization and Referred Authorization

The `threeDSecure_authorization_request` and `threeDSecure_authorize_referral_request` transaction types require information to be provided in the `HistoricTxn` element:

- Request
 - Authentication – section 2.1.1.2
 - Transaction
 - HistoricTxn – section 2.3.4.1.2

2.3.4.1 Schema Elements for Request

2.3.4.1.1 ThreeDSecure

This element contains all of the extra details which are required for the 3-D Secure check to be initiated and should be provided for auth and pre Requests.

Element Name: ThreeDSecure
Position: Request.Transaction.TxnDetails

Elements of ThreeDSecure			
Element Name	Description	Values / Limitations	Required
verify	Indicates whether the transaction should be checked for enrolment in the 3-D Secure system	yes no	R
merchant_url	The URL of the website on which the payment is being made. Will be displayed to cardholder by ACS	1-2048 characters. Must be <u>fully qualified</u> (i.e. include https://)	M
purchase_desc	A short description of what has been purchased. This is shown to the card holder by the ACS.	1-125 characters	M
purchase_datetime	The date and time of the transaction	YYYYMMDD HH:MM:SS format	M
Browser	See section 2.3.4.1.2		M

Example XML for ThreeDSecure complex elements

```

<ThreeDSecure>
  <verify>yes</verify>
  <merchant_url>https://www.widgets.com</merchant_url>
  <purchase_desc>DVD</purchase_desc>
  <purchase_datetime>20060731 21:59:42</purchase_datetime>
  <Browser>...</Browser>
</ThreeDSecure>

<ThreeDSecure>
  <verify>no</verify>
</ThreeDSecure>

```

2.3.4.1.2 Browser

The `Browser` element contains all the information about the browser the cardholder is using to access your website, and must be provided for `auth` and `pre` requests where the 3-D Secure check is to be performed.

Element Name:	Browser
Position:	Request.Transaction.TxnDetails.ThreeDSecure

Elements of Browser			
Element Name	Description	Values / Limitations	Required
device_category	Indicates the type of device used to the transaction	0 – for PC	R
accept_headers	The MIME types of the headers accepted by this device		R
user_agent	The User Agent associated with the device		R

Example XML for Browser complex elements

```

<Browser>
    <device_category>0</device_category>
    <accept_headers>*/*</accept_headers>
    <user_agent>IE/6.0</user_agent>
</Browser>

<Browser>
    <device_category>0</device_category>
    <accept_headers>txt/xml, application/xml, application/xhtml+xml,
        text/html;q=0.9, text/plain;q=0.8,video/x-mng, image/png,
        image/jpeg, image/gif;q=0.2, text/css, */*;q=0.1
    </accept_headers>
    <user_agent>Mozilla/5.0 (Macintosh; U; PPC Mac OS X; en)
        AppleWebKit/417.9 (KHTML, like Gecko) Safari/417.9.2
    </user_agent>
</Browser>

```

2.3.4.1.3 HistoricTxn

The **HistoricTxn** element is used for `threeedsecure_authorization_request`, `threeedsecure_authorize_referral_request` and `fulfill` transactions. It enables these transactions to be tied back to the original transaction.

Element Name:	HistoricTxn
Position:	Request.Transaction

Elements of HistoricTxn					
Element Name	Description	Values / Limitations	3DS ref¹	3DS aut²	3DS ful³
reference	The datacash_reference number for the transaction	16 digits	R	R	R
authcode	The authorisation code provided by your Bank's Authorisation Centre	-	R	n/a	R
Method	The transaction type	See bullets below	R	R	R
pares_message	The Payer Authentication Response (PARes) returned by the ACS	As returned by the ACS	O	O	n/a

¹ `threeedsecure_authorize_referral_request`

² `threeedsecure_authorization_request`

³ `fulfill`

Attributes of elements within HistoricTxn				
Attribute Name	Attribute of Element	Description	Values / Limitations	Required
tx_status_u	method	Accepts the transaction if the ACS is unable to complete the validation of the card holder	accept	O

Example XML for HistoricTxn complex elements

```

<HistoricTxn>
    <reference>4400200042810513</reference>
    <method>threedsecure_authorization_request</method>
    <pares_message>uyt45t89cnwu3rhc98a4hterjklth4o8c
        tsrzjzth4</pares_message>
</HistoricTxn>

<HistoricTxn>
    <reference>4400200042810513</reference>
    <method tx_status_u="accept">
        threedsecure_authorization_request</method>
    <pares_message>4o7sbvohyv8s4075mytudriotu0359ucmi
        udttoyuu</pares_message>
</HistoricTxn>

```

Example XML for HistoricTxn complex elements for transactions which cannot be 3-D Secure checked (e.g. card scheme not supported, non-enrolled cards)

```

<HistoricTxn>
    <reference>4400200042810513</reference>
    <method>threedsecure_authorization_request</method>
</HistoricTxn>

```

Example XML for HistoricTxn complex elements for referred authorisation

```

<HistoricTxn>
    <reference>4400200042810513</reference>
    <authcode>12AB52</authcode>
    <method>threedsecure_authorize_referral_request</method>
    <pares_message>w5670w5689kg74u76490wf
        uvaero5ielktekvtvre</pares_message>
</HistoricTxn>

```

Example XML for HistoricTxn complex elements for a fulfill

```

<HistoricTxn>
    <reference>4400200042810513</reference>
    <authcode>12AB52</authcode>
    <method>fulfill</method>
</HistoricTxn>

```

2.3.4.2 XML Example Requests

2.3.4.2.1 Cardholder Verification Requests

Example XML for pre transactions which by-pass the 3-D Secure check

```

<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>387545924AdkWdd</merchantreference>
      <amount currency="GBP">10.04</amount>
      <ThreeDSecure>
        <verify>no</verify>
      </ThreeDSecure>
    </TxnDetails>
    <CardTxn>
      <Card>
        <pan>4444*****</pan>
        <expirydate>06/12</expirydate>
      </Card>
      <method>pre</method>
    </CardTxn>
  </Transaction>
</Request>

<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>4564234523</merchantreference>
      <amount currency="AUD">59.45</amount>
      <capturemethod>ecomm</capturemethod>
      <ThreeDSecure>
        <verify>no</verify>
        <merchant_url>https://www.mywebsite.com/12</merchant_url>
        <purchase_desc>CDs and DVDs</purchase_desc>
        <purchase_datetime>20060531 13:06:28</purchase_datetime>
        <Browser>
          <device_category>0</device_category>
          <accept_headers>*/*</accept_headers>
          <user_agent>IE/6.0</user_agent>
        </Browser>
      </ThreeDSecure>
    </TxnDetails>
    <CardTxn>
      <Card>
        <pan>4444*****</pan>
        <expirydate>06/12</expirydate>
      </Card>
    </CardTxn>
  </Transaction>
</Request>

```

```
<method>pre</method>
</CardTxn>
</Transaction>
</Request>
```

Example XML for an auth request, requesting verification

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>387545924537037</merchantreference>
      <amount currency="GBP">10.04</amount>
      <capturemethod>ecomm</capturemethod>
      <ThreeDSecure>
        <verify>yes</verify>
        <merchant_url>https://www.mywebsite.com</merchant_url>
        <purchase_desc>CDs and DVDs</purchase_desc>
        <purchase_datetime>20060201 23:59:59</purchase_datetime>
        <Browser>
          <device_category>0</device_category>
          <accept_headers>*/*</accept_headers>
          <user_agent>IE/6.0</user_agent>
        </Browser>
      </ThreeDSecure>
    </TxnDetails>
    <CardTxn>
      <Card>
        <pan>4444*****</pan>
        <expirydate>06/12</expirydate>
      </Card>
      <method>auth</method>
    </CardTxn>
  </Transaction>
</Request>
```

Example XML for a Pre-Registered Card request, requesting verification

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>387545924537037</merchantreference>
      <amount currency="GBP">10.04</amount>
      <capturemethod>ecomm</capturemethod>
      <ThreeDSecure>
        <verify>yes</verify>
        <merchant_url>https://www.mywebsite.com</merchant_url>
        <purchase_desc>CDs and DVDs</purchase_desc>
        <purchase_datetime>20060201 23:59:59</purchase_datetime>
        <Browser>
          <device_category>0</device_category>
          <accept_headers>*/*</accept_headers>
          <user_agent>IE/6.0</user_agent>
        </Browser>
      </ThreeDSecure>
    </TxnDetails>
    <CardTxn>
      <card_details type="preregistered">
        4100200043070870</card_details>
      <method>auth</method>
    </CardTxn>
  </Transaction>
</Request>
```

2.3.4.2.2 Historic Transactions

Example XML for a threedsecure_authorization_request, with PAREs

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <reference>4900200042810407</reference>
      <method tx_status_u="accept">
        threedsecure_authorization_request</method>
      <pares_message>w48gj6s5ty45ty8cviozdtrjgd
      </pares_message>
    </HistoricTxn>
  </Transaction>
</Request>
```

Example XML for a threedsecure_authorization_request without a PAREs. E.g. for a non-enrolled card, or a card scheme which is not supported by 3-D Secure

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <reference>4100200043070078</reference>
      <method>threedsecure_authorization_request</method>
    </HistoricTxn>
  </Transaction>
</Request>
```

Example XML for a threedsecure_authorize_referral_request

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <reference>4100200043070078</reference>
      <authcode>ZY987A</authcode>
      <method>threedsecure_authorize_referral_request</method>
    </HistoricTxn>
  </Transaction>
</Request>
```

Please refer to section 2.2.1.2.2 for an example fulfill request.

2.3.4.3 Schema Elements for Response

2.3.4.3.1 ThreeDSecure

This element will be returned in response to a card enrolment check, if the card is enrolled. If the card is not enrolled or is not supported, this element will not be present.

Element Name:	ThreeDSecure
Position:	Response.CardTxn

Elements of ThreeDSecure	
Element Name	Description
pareq_message	The PReq
acs_url	The URL of the ACS, to which the cardholder needs to be redirected. In the test environment this will be the URL of the testserver ACS. In the live environment this will be the URL of the Issuing Bank's ACS.

Example XML for ThreeDSecure complex elements

```
<ThreeDSecure>
  <acs_url>https://secure.barclaycard.co.uk/barclays/tdsecure/pa.jsp?
    partner=barclaycard.visa&VAA=B</acs_url>
  <pareq_message>eJxdUltugzAQ/M8pUA+AHyEQKscSLR/NB1HU5AKWsypIBRIb
    StrT14Y4JkGAdnYWdjRjdiwVQH4A2SvgyBxBWgtviCoTpuxwgjuiZ4tU
    xeLGnoffYJl6k26AeUrtqGkxCH1CEHHV2AkqVo0tcwLSEvb9sdj5MkSg1D
    N+j5GtQ2521KcJxgzNCEPd+IGvhBKFEyNNaekm3fd0qXr2nMkAOe7tU3H4
    YhPIlOSKHLULY1Q7br5KJnvWzf24aeb71WJ17k2TB/dvk2Ko4ZKf7khiE7
    4efNOuAU4xhTTAK6ff215mzo7M98qa1gTnCII2PLhDx9tkIyN2NH5p2ZB7
    1S0Ehngk7ntgHzjYnrXs/Ugpa8A90Zgba8W/PsBHv/eEhVdiYmowwv7Wu6
    xnxH4mF/ZWIh1LI3cF/i/mnWuWNmQxkPJF8w9HhY/w</pareq_message>
</ThreeDSecure>
```

2.3.4.4 XML Example Responses

2.3.4.4.1 Cardholder Verification Response

Example XML Response for a card which is not enrolled

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <CardTxn>
    <card_scheme>Visa</card_scheme>
    <country>United Kingdom</country>
    <issuer>Prudential Banking PLC</issuer>
  </CardTxn>
  <datacash_reference>4300200042810617</datacash_reference>
  <merchantreference>387546093287037</merchantreference>
  <mode>...</mode>
  <reason>3DS Card not Enrolled</reason>
  <status>162</status>
  <time>...</time>
</Response>
```

Example XML Response for a card which is enrolled

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <CardTxn>
    <card_scheme>Mastercard</card_scheme>
    <country>United Kingdom</country>
    <ThreeDSecure>
      <acs_url>https://www.clicksafe.lloydstsb.com/Lloyds/
        tdsecure/pa.jsp?partner=mc&VAA=B</acs_url>
      <pareq_message>eJxdUltugzAQ/M8pUA+AHyEQKscSLR/NB1H
        U5AKWsypIBRIbStrT14Y4JkGAdnYWdjRjdiwVQH4A2Svgiy
        BgBWgtviCoTpuxswgjuiZ4tUxeLGnoffYJl6k26AeUrtqGk
        xCH1CEHHV2AkqVoOtcwLSEvb9sdj5MkSglDN+j5GtQ2521K
        cJxgzNCEPd+IGvhBKFEyNNaekm3fd0qXr2nMkAOe7tU3H4Y
        hPIlOSKHLULY1Q7br5KJnvWzf24aeb71WJ17k2TB/dvk2Ko
        4ZKf7khiE74efNOuAU4xhTTAK6ff215mZo7M98qa1gTnCII
        2PLhDx9tkIyN2NH5p2ZB71S0EhngkN+AK7ntgHzjYnrXs/U
        gpa8A90Zgba8W/PsBHv/eEhVdiYmowwv7Wu6xnxF4mF/ZWI
        h1LI3cF/i/mnWuWNmQxkPJF8w9HhY/wFTsbsh
      </pareq_message>
    </ThreeDSecure>
  </CardTxn>
  <datacash_reference>4300200042810537</datacash_reference>
  <merchantreference>387546049537037</merchantreference>
  <mode>...</mode>
  <reason>3DS Payer Verification Required</reason>
  <status>150</status>
  <time>...</time>
</Response>
```

2.3.4.4.2 Cardholder Authorisation Responses

Example XML Response when the PARes returned is invalid

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>3000100226001681</datacash_reference>
    <information>[2:Error] Expected PARes node: got
        &apos;Error&apos;</information>
    <merchantreference>3200100226001675</merchantreference>
    <mode>...</mode>
    <reason>3DS invalid pares</reason>
    <status>176</status>
    <time>...</time>
</Response>
```

Example XML Response were a referral response is received

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <CardTxn>
        <card_scheme>Visa</card_scheme>
        <country>United Kingdom</country>
        <issuer>Prudential Banking PLC</issuer>
    </CardTxn>
    <datacash_reference>3000100226001681</datacash_reference>
    <merchantreference>13542DSDWK21</merchantreference>
    <mode>...</mode>
    <reason>3DS call auth centre</reason>
    <status>161</status>
    <time>...</time>
</Response>
```

2.3.4.4.3 Authorisation of Referred Cards

Example XML Response were a referral response to a three3dsecure_authorize_referral_request

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4900200043070079</datacash_reference>
    <merchantreference>13542DSDWK21</merchantreference>
    <mode>...</mode>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <time>...</time>
</Response>
```

2.3.4.5 Using inline "Redirect Page"

Due to the increasing use of software known as "Popup Killers", Visa and MasterCard are recommending that merchants be aware that a "Popup" window may not be the best way to direct cardholders to their issuer for authentication. Indeed, the 3-D Secure specifications have explicitly banned the use of "Popup" windows as of October 2004. The recommended way to allow Cardholders to authenticate themselves is via an in-line Redirect Page generated either via an `<iframe />` type system or as a separate "framed" page.

Example using the `<iframe />` mechanism

```

<html>
  <head>
    <title>Please Authenticate</title>
  </head>
  <body OnLoad="OnLoadEvent(); " >
    <form name="downloadForm"
      action="https://mybank.com/vbyv/verify" method="POST">
      <textarea name="PaReq"
style="display:none">thePaReqWithBreaks</textarea>
      <input type="hidden" name="TermUrl" value="https://www.
MyWidgets.Com/next.cgi">
      <input type="hidden" name="MD" value="200304012012a">
    </form>
    <script language="Javascript" >
      <!--
        function OnLoadEvent() {
          document.downloadForm.target = "ACSframe";
          document.downloadForm.submit();
        }
      //-->
    </script>
    <!-- MERCHANT TO FILL IN THEIR OWN BRANDING HERE -->
    <iframe src="blank.htm" name="ACSframe" width="390"
      height="450" frameborder="0"> </iframe>
    <!-- MERCHANT TO FILL IN THEIR OWN BRANDING HERE -->
  </body>
</html>

```

In this example, no Popup window is created. The ACS dialogue takes place within a floating frame on your page. After the cardholder has authenticated themselves they will be returned to your site by their issuers ACS posting the results to the script defined in the TermURL variable.

2.3.4.6 ACS Simulator & Magic Card Functionality

Please refer to the Developer's Guide Appendices (Appendix 2 & Appendix 3) for detailed information on the DataCash ACS Simulator and Magic Card Functionality. It is intended that these are used during merchant integration of 3DS With DataCash MPI as well as MPI Only integration, see section 2.3.5.

2.3.5 *MPI Only*

2.3.5.1 Summary of Service

Using this service, it is possible for merchants to perform 3-D Secure authentication of a transaction using the DataCash MPI, and authorize the transaction with a 3rd party Payment Gateway.

In order to use the decoupled DataCash MPI for 3-D Secure transactions a variety of different transaction types are required. Initially an enrolment check transaction must be performed which will contain all details required to initiate the 3-D Secure authentication process as well as transaction details relating to the MPI-Only transaction. The response to this message will indicate whether the cardholder is enrolled.

If the cardholder is enrolled this can be followed by a validation authentication transaction containing the PARes message returned from the ACS and a historic reference. A successful response to this transaction will provide the merchant with enough 3-D Secure information to allow the merchant to authorize the transaction with a 3rd party Payment Gateway.

Alternatively, a merchant can optionally authorize the transaction using the DPG.

2.3.5.2 Schema elements for Request

This element contains all of the extra details which are required for the 3-D Secure check to be initiated and should be provided for auth and pre Requests.

Element Name:	MpiTxn
Position:	Request.Transaction

Elements of MpiTxn			
Element Name	Description	Values / Limitations	Required
method	The method of the transaction	mpi	R
Card	The card details of the transaction to be authenticated. This element contains sub elements. See 2.1.1.1 for more information	See 2.1.1.1 for more information	M
card_details	If subscribed to the preregistered card service, this element can be used instead of the Card block above. See 2.2.11 for more information about the preregistered card service.	XML Attribute type="preregistered" should be supplied in the opening tag. The value contained should be the datacash_reference of the card transaction.	M

2.3.5.3 End to end walkthrough: Cardholder enrolled

The following scenario uses the DataCash MPI to successfully authorize a 3-D Secure transaction where the cardholder is enrolled.

2.3.5.3.1 Enrolment Check

The first stage of 3-D Secure authentication is to perform an enrolment check. The enrolment check transaction is similar to the existing Cardholder Verification Check message type as used to perform normal (i.e. coupled with Card Authorization) 3-D Secure transactions with the DataCash MPI. The main difference between these two transaction types being that the enrolment check transaction holds card details within the MpiTxn block.

It should be noted that neither Cv2Avs data, nor the Verify element will be allowed in the enrolment check transaction. CV2/AVS data can be provided in a subsequent authorization transaction if required.

Example Enrolment Check Request

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>387545924537037</merchantreference>
      <amount currency="GBP">10.04</amount>
      <ThreeDSecure>
        <merchant_url>http://www.mywebsite.com</merchant_url>
        <purchase_desc>CDs and DVDs</purchase_desc>
        <purchase_datetime>20060201 3:59:59</purchase_datetime>
        <Browser>
          <device_category>0</device_category>
          <accept_headers>*/*</accept_headers>
          <user_agent>IE/6.0</user_agent>
        </Browser>
      </ThreeDSecure>
    </TxnDetails>
    <MpITxn>
      <method>mpi</method>
      <Card>
        <pan>4444*****</pan>
        <expirydate>06/12</expirydate>
        <startdate>06/12</startdate>
        <issuemumber>06/12</issuemumber>
      </Card>
    </MpITxn>
  </Transaction>
</Request>
```

The following is an example of the xml message response for an enrolled card. The response contains the encoded PReq message along with an ACS URL link.

Example Enrolment Check Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <status>150</status>
    <reason>3DS Payer Verification Required</reason>
    <merchantreference>387546049537037</merchantreference>
    <datacash_reference>4300200042810537</datacash_reference>
    <time>...</time>
    <mode>...</mode>
    <CardTxn>
        <ThreeDSecure>
            <pareq_message>eJxdUltugzAQ/M8pUA+AHyEQKscSLR/NB1H
                U5AKWsyPIBRIbStrT14Y4JkGAdnYWdjRjdiwVQH4A2Svgiy
                BgBWgtviCoTpuxswgjuiZ4tUxeLGnoFFYJl6k26AeUrtqGk
                xCH1CEHHV2AkqVoOtcwLSEvb9sdj5MkSglDN+j5GtQ2521K
                cJxgzNCEPd+IGvhBKFEyNNaekm3fd0qXr2nMkAOe7tU3H4Y
                hPI1OSKHLULY1Q7br5KJnvWzf24aeb71WJ17k2TB/dvk2Ko
                4ZKf7khiE74efNOuAU4xhTTAK6fF215mZo7M98qa1gTnCII
                2PLhDx9tkIyN2NH5p2ZB71S0EhngkN+AK7ntgHzjYnrXs/U
                gpa8A90Zgba8W/PsBHv/eEhVdiYmowwv7Wu6xnxF4mF/ZWI
                h1LI3cF/i/mnWuWNmQxkPJF8w9HhY/wFTsbsh
            </pareq_message>
            <acs_url>https://www.clicksafe.lloydstsb.com/Lloyds/
                tdsecure/pa.jsp?partner=mc&VAA=B</acs_url>
        </ThreeDSecure>
    </CardTxn>
</Response>
```

2.3.5.3.2 Validate Authentication

In order to authenticate an enrolled card, the merchant is required to submit a 'threedssecure_validate_authentication' transaction. This request will use the datacash_reference from the enrolment check as the historic reference along with the resulting PARES message received from the ACS.

An example threedssecure_validate_authentication transaction is shown below:

Example Validate Authentication Request

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>99000001 </client>
        <password>*****</password>
    </Authentication>
    <Transaction>
        <HistoricTxn>
            <reference>4300200042810537</reference>
            <method>threedssecure_validate_authentication</method>
            <pares_message>
                uyt45t89cnwu3rhc98a4hterjklth4o8ctsrjzth4</pares_message>
            </HistoricTxn>
        </Transaction>
    </Request>
```

The response to a threedssecure_validate_authentication where the cardholder was successfully authenticated is shown below. This response contains the necessary fields should the merchant wish to authorise the transaction using a 3rd Party Payment Gateway.

Example Validate Authentication Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <CardTxn>
        <ThreeDSecure type="visa">
            <cardholder_registered>yes</cardholder_registered>
            <eci>05</eci>

            <security_code>Q0FWVkBv1ZDQVZWQ0FWVkBv1Y=</security_code>
            <xid>MDAwMDAwMDAwMDAwMTIzNDU2NzE=</xid>
        </ThreeDSecure>
    </CardTxn>
    <datacash_reference>3200900012345671</datacash_reference>
    <mode>LIVE</mode>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <time>_UNIX_TIME_</time>
</Response>
```

Validate Authentication Requests support multiple versions. Where the request version is not supplied or is equal to 1 then the above functionality will be in effect. Version 2 functionality differs from version 1 in the formatting of the Response. When supplied version greater than 1 then an additional elements are returned: 'mid' and 'acquirer' are added to the response element. All MPI transactions return the ThreeDSecure block in the response. 'cavvAlgorithm' is added to the ThreeDSecure element of the response and the 'security_code' element is replaced with the 'aav' element.

Example Validate Authentication Response from version 2 request

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="2">
  <CardTxn>
    <ThreeDSecure type="visa">
      <cardholder_registered>yes</cardholder_registered>
      <eci>05</eci>
      <cavvAlgorithm>1</cavvAlgorithm>
      <aav>Q0FWVkBVB1ZDQVZWQ0FWVkBVB1Y=</aav>
      <xid>MDAwMDAwMDAwMDAwMTIzNDU2NzE=</xid>
    </ThreeDSecure>
  </CardTxn>
  <datacash_reference>3200900012345671</datacash_reference>
  <mode>LIVE</mode>
  <mid>3658684685</mid>
  <acquirer>Rbs</acquirer>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>_UNIX_TIME_</time>
</Response>
```

If you use the version 2 for a Chase or Omnipay acquirer, 2 additional fields will be returned by the Payment Gateway:

- 'extended_status' : containing the extended status code obtained from the acquirer.
- 'extended_response_message': containing the extended description or reason obtained from the acquirer.

Example of transaction using Chase and version 2:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response version="2">
    <CardTxn>
        <ThreeDSecure type="visa">
            <cardholder_registered>yes</cardholder_registered>
            <eci>05</eci>
            <cavvAlgorithm>1</cavvAlgorithm>
            <aav>Q0FWVkBkNBV1ZDQVZWQ0FWVkBkNBV1Y=</aav>
            <xid>MDAwMDAwMDAwMDAwMTIzNDU2NzE=</xid>
        </ThreeDSecure>
    </CardTxn>
    <datacash_reference>3200900012345671</datacash_reference>
    <mode>LIVE</mode>
    <mid>3658684685</mid>
    <acquirer>Chase Paymentech</acquirer>
    <extended_status>100</extended_status>
    <extended_response_message>Approved</extended_response_message>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <time>_UNIX_TIME_</time>
</Response>
```

2.3.5.3.3 Authorization via the DPG

The subsequent authorization of an MPI Only transaction by the DPG is entirely optional. Enough information will have been returned to the merchant by the DPG in response to the validation authentication transaction to permit authorization using 3-D Secure with an alternative Payment Service Provider.

An example authorization request with the DPG is illustrated below. This authorization is similar to that used for pre-registered card transactions. The method types permitted for a subsequent authorization attempts are `auth` or `pre` and the type of the transaction must state `from_mpi`.

CV2/AVS details can be optionally provided with the authorization request as part of CardTxn details using the new Card element.

Additionally a transaction amount can also optionally be supplied. If a new amount is supplied, then the transaction currency must also be provided and must match the currency presented in the original enrolment check. Where a new amount is specified, that amount will be used instead of the amount supplied in the original enrolment check.

If the new amount is specified in an authorisation is greater than that specified in the original authentication transaction, it is at the discretion and risk of the merchant. DataCash make no claims with regard to liability shift.

Example Authorisation Request with the DPG

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>99000001</client>
        <password>*****</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">1001.02</amount>
            <merchantreference>387545924AdkWdd</merchantreference>
        </TxnDetails>
        <CardTxn>
            <method>auth</method>
            <card_details
type="from_mpi">3200900012345671</card_details>
            <Card>
                <Cv2Avs>
                    <street_address1>1 High Street</street_address1>
                    <street_address2>This Town Town</street_address2>
                    <street_address3>Somewhere</street_address3>
                    <street_address4>United Kingdom</street_address4>
                    <postcode>S01 2CD</postcode>
                    <cv2>123</cv2>
                </Cv2Avs>
            </Card>
        </CardTxn>
    </Transaction>
</Request>
```

The response to a subsequent authorization of an MPI Only transaction will be as per standard bankcard responses.

Example Authorisation Response from the DPG

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <CardTxn>
        <authcode>100000</authcode>
        <card_scheme>VISA</card_scheme>
        <country>United Kingdom</country>
    </CardTxn>
    <datacash_reference>3000900012345672</datacash_reference>
    <merchantreference>442100009</merchantreference>
    <mode>LIVE</mode>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <time>_UNIX_TIME_</time>
</Response>
```

2.3.5.4 End to end walkthrough: Cardholder not enrolled

As can be seen in the example xml response message below, where the cardholder is not enrolled the ECI value will be returned to the merchant. If the merchant wishes to continue with an authorization, they can do so using an alternative PSP or via the DPG.

2.3.5.4.1 Enrolment Check

An example response whereby the cardholder is not enrolled is shown below. This response will contain an ECI value indicating that the cardholder authentication could not be performed.

Example Enrolment Check Response – Cardholder not enrolled

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <CardTxn>
    <ThreeDSecure type="visa">
      <cardholder_registered>no</cardholder_registered>
      <eci>06</eci>
    </ThreeDSecure>
  </CardTxn>
  <datacash_reference>4300200042810537</datacash_reference>
  <merchantreference>387546049537037</merchantreference>
  <mode>LIVE</mode>
  <status>162</status>
  <reason>3DS Card not Enrolled</reason>
  <time>__UNIX_TIME__</time>
</Response>
```

2.3.5.4.2 Validate Authentication

For an MPI Only transaction with a cardholder that is not enrolled, a ThreeDSecure validate authentication transaction will not be permitted. Such transactions will be rejected by the DPG.

The following example response message would be returned in response to a threesecure_validate_authentication where the enrolment check determined that the cardholder is not enrolled.

Example Validate Authentication Response – Cardholder not enrolled

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>3200900012345671</datacash_reference>
    <mode>LIVE</mode>
    <reason>Authentication can only be performed for enrolled
cards</reason>
    <status>600</status>
    <time>_UNIX_TIME_</time>
</Response>
```

2.3.5.4.3 Authorization via the DPG

Authorization of an MPI Only transaction where the cardholder was not enrolled can optionally be performed via the DPG. The request and response messages for a successful authorization are as shown in 2.1.1.3, the section for the Authorization of an enrolled card.

2.3.5.5 Preregistered Card support

Preregistered cards can be used in the `MpiTxn` segment of the Enrolment check instead of the `Card` block.

The following scenario uses the MPI to authenticate a 3-D Secure transaction using preregistered card details.

The enrolment check specifies which earlier (successful) transaction to use to retrieve the card details as highlighted below:

Example Enrolment Check using Preregistered Card Details

```

<Request>
  <Authentication>
    <client>99000001</client>
    <password>fred</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>387545924537037</merchantreference>
      <amount currency="GBP">10.04</amount>
      <ThreeDSecure>
        <merchant_url>http://www.mywebsite.com</merchant_url>
        <purchase_desc>CDs and DVDs</purchase_desc>
        <purchase_datetime>20080808
          23:59:59</purchase_datetime>
        <Browser>
          <device_category>0</device_category>
          <accept_headers>*/*</accept_headers>
          <user_agent>IE/6.0</user_agent>
        </Browser>
      </ThreeDSecure>
    </TxnDetails>
    <MpiTxn>
      <method>mpi</method>
      <card_details
        type="preregistered">4300200042810536</card_details>
    </MpiTxn>
  </Transaction>
</Request>
```

DataCash will use the specified transaction to retrieve the card details to be used for the new transaction.

The process of completing 3-D Secure authentication using the preregistered card service differs only in how the card details are specified in the initial enrolment check transaction. All responses will be as if the card details had been supplied explicitly with the exception of the additional return code of 250 indicating that the details for the supplied reference cannot be found.

2.3.5.6 ACS Simulator & Magic Card Facility

Please refer to the Developer's Guide Appendices (Appendix 2 & Appendix 3) for detailed information on the DataCash ACS Simulator and Magic Card Functionality. It is intended that these are used during merchant integration of the MPI Only service as well as 3DS with DataCash MPI service as described in section 2.3.4.1.1.

2.3.6 3-D Secure, with 3rd party MPI

This service enables you to authorize a payment via the DPG which has already been 3-D Secure checked by a 3rd party MPI. A technical introduction to this Service is available on the website:

<http://www.datacash.com/gettingproducts.php?id=3D-Secure-Third-Party-MPI-Service>

This service is utilised by sending a normal Credit and Debit Card Service Request with up to four pieces of extra information. This section of the documentation assumes the reader is familiar with the Credit and Debit Card Service, as described in section 2.2.1.

The service may also be used in conjunction with the Pre-Registered Card service, as outlined in section 2.2.11

2.3.6.1 Schema Elements for Request

The data for the service is passed in five distinct places in the schema. The additional information for this Service is all passed in the **Secure** element:

2.3.6.1.1 Request

- Authentication – section 2.1.1.2
- Transaction
 - CardTxn – as described in 2.6.1.1.2
 - Card – section 2.2.1.1.1
 - Secure – up to four extra pieces of information about the card transaction, section 2.3.6.1.2
 - TxnDetails – see section 2.2.1.1.3

2.3.6.1.2 Secure

The `Secure` element contains the extra information for the 3-D Secure transaction. The data within this element is obtained from your MPI.

Element Name:	Secure
Position:	Request.Transaction.TxnDetails.CardTxn

Elements of Secure		
Element Name	Description	Values / Limitations
security_code	The Cardholder Authentication Verification Value (CAVV) for Visa cards, or Universal Cardholder Authentication Field (UCAF) for MasterCard, (Amex) for American Express	M
cardholder_registered	Indicates whether the cardholder was registered for 3-D Secure and the PARes / VERes status	yes no attempted - enrolled, PARes status 'A' ch_enrolled_u - VERes status 'U' tx_status_u - enrolled, PARes status 'U'
eci	The Electronic Commerce Indicator (ECI) obtained from your MPI	01 02 05 06 00
transactionID	The XID from PARes.Purchase.xid	As copied from the PARes

Example Secure complex element

```
<Secure>
    <cardholder_registered>no</cardholder_registered>
</Secure>

<Secure>
    <security_code>AAABAwRzYAAAAAAAAAAAAA=</security_code>
    <cardholder_registered>yes</cardholder_registered>
    <eci>06</eci>
    <transactionID>TbwiW8VLThG3TjDuI7KS5wAJCAI=</transactionID>
</Secure>
```

2.3.6.2 XML Example Requests

Example Request XML for 3-D Secure with a 3rd party MPI, for a registered card

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>99000001</client>
        <password>*****</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <merchantreference>123456</merchantreference>
            <amount currency="GBP">10.00</amount>
            <capturemethod>ecomm</capturemethod>
        </TxnDetails>
        <CardTxn>
            <Card>
                <pan>4444*****1111</pan>
                <expirydate>12/09</expirydate>
            </Card>
            <method>pre</method>
            <Secure>
                <security_code>BwABCEYiBQAAAAAGViIFAAA
                AAAA=</security_code>
                <cardholder_registered>yes</cardholder_registered>
                <eci>06</eci>
                <transactionID>TbwiW8VLThG3TjDuI7KS5wAJ
                CAI=</transactionID>
            </Secure>
        </CardTxn>
    </Transaction>
</Request>
```

Example Request XML for 3-D Secure with a 3rd party MPI, for a registered card using the Pre-Registered Card service

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>99000001</client>
        <password>*****</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <merchantreference>123456</merchantreference>
            <amount currency="GBP">10.00</amount>
            <capturemethod>ecomm</capturemethod>
        </TxnDetails>
        <CardTxn>
            <card_details type="preregistered">
                4900200053281123</card_details>
            <method>pre</method>
            <Secure>
                <security_code>BwABCEYiBQAAAAAGViIFAAA
                AAAA=</security_code>
                <cardholder_registered>yes</cardholder_registered>
                <eci>06</eci>
                <transactionID>TbwiW8VLThG3TjDuI7KS5wAJ
                CAI=</transactionID>
            </Secure>
        </CardTxn>
    </Transaction>
</Request>
```

Example Request XML for 3-D Secure with a 3rd party MPI, for an un-registered card

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>99000001</client>
        <password>*****</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <merchantreference>123456</merchantreference>
            <amount currency="GBP">56.26</amount>
            <capturemethod>ecomm</capturemethod>
        </TxnDetails>
        <CardTxn>
            <Card>
                <pan>5374*****0001</pan>
                <expirydate>01/08</expirydate>
            </Card>
            <method>pre</method>
            <Secure>
                <cardholder_registered>no</cardholder_registered>
            </Secure>
        </CardTxn>
    </Transaction>
</Request>
```

Example Request XML for a card scheme which is not supported for the 3-D Secure check

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>99000001</client>
        <password>*****</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <merchantreference>45786906</merchantreference>
            <amount currency="GBP">23.45</amount>
            <capturemethod>ecomm</capturemethod>
        </TxnDetails>
        <CardTxn>
            <Card>
                <pan>6759*****9999</pan>
                <expirydate>12/07</expirydate>
                <startdate>12/04</startdate>
                <issuemumber>03</issuemumber>
            </Card>
            <method>auth</method>
        </CardTxn>
    </Transaction>
</Request>
```

2.3.6.3 Schema Elements for Response

This service does not have any service-specific elements in the response. Please refer to the Bank Card Service (section 2.2.1) and General Response Elements (section 2.1.1.3) for details of the generic elements returned for credit and debit card transaction.

2.3.7 *Batched Fraud Screening*

This service is utilised by sending a normal Credit and Debit Card Service Request with additional information.

This section of documentation assumes the Credit and Debit Card Service has already been integrated and the reader is familiar with it. The Credit and Debit Card Service is described in section 2.2.1

2.3.7.1 Schema Elements for Request

All of these fields described within this section are optional, unless otherwise stated.

Initial Transactions with Card details

The Datacash Fraud Service (DFS) check is carried out for any `pre` and `auth` transactions while your account is subscribed to this service. To perform the check, additional information about the customer needs to be presented with the transaction.

The data is passed within the `The3rdMan` element in the schema:

- `The3rdMan`
 - `CustomerInformation` – additional details about the customer, section 2.3.7.1.2
 - `DeliveryAddress` – details the delivery address for the order, section 2.3.7.1.5
 - `OrderInformation` – details about the order, section 2.3.7.1.18
 - `Products` - section 2.3.7.1.19
 - `Product` – details about the individual products within the order, section 2.3.7.1.20

2.3.7.1.1 The Datacash Fraud Service

Element Name:	The3rdMan
Position:	Request.Transaction.TxnDetails

Elements of The3rdMan		
Element Name	Description	Values / Limitations
CustomerInformation	See section 2.3.7.1.2	
DeliveryAddress	See section 2.3.7.1.5	
BillingAddress	See section 2.3.7.1.4	
OrderInformation	See section 2.3.7.1.18	

Example XML for The3rdMan complex elements

```
<The3rdMan>
  <CustomerInformation>...</CustomerInformation>
  <DeliveryAddress>...</DeliveryAddress>
  <BillingAddress>...</BillingAddress>
  <OrderInformation>...</OrderInformation>
  <Realtime>...</Realtime>
</The3rdMan>

<The3rdMan>
  <CustomerInformation>...</CustomerInformation>
</The3rdMan>
```

2.3.7.1.2 CustomerInformation

This element contains details of the customer. T3M themselves are the authoritative source on what the contents of these fields should be.

Element Name: CustomerInformation
Position: Request.Transaction.TxnDetails.The3rdMan

Elements of CustomerInformation		
Element Name	Description	Values / Limitations
alt_telephone	Alternative Telephone Number	Max 20 characters <i>International numbers:</i> include Country code, exclude access code
customer_dob	Customer Date Of Birth	YYYY-MM-DD format
customer_reference	Customer reference	Max 50 characters
delivery_forename	The first name of the person to which the order is being delivered	Max 50 characters
delivery_phone_number	The phone number of the person to which the order is being delivered	Max 20 characters
delivery_surname	The surname of the person to which the order is being delivered	Max 50 characters
delivery_title	The title of the person to which the order is being delivered	Max 10 characters
driving_licence_number	Driving License Number	Max 30 characters
email	Email Address	Max 50 characters
first_purchase_date	First Purchase Date By Customer	YYYY-MM-DD format
forename	Customer first name	Max 50 characters
introduced_by	Customer Ref Of Introducing Customer	Max 50 characters

ip_address	IP Address	Must be in "dotted-quad" notation
previous_purchases	Previous Purchases	See below
order_number	Order Number	Max 60 characters
sales_channel	Type of sale	1 - Mail order 2 - Telephone order 3 - Internet 4 - Other 5 - In store 6 - Kiosk 7 - Mobile
surname	Customer Surname	Max 50 characters
telephone	Telephone Number	A maximum of 20 characters. <i>International numbers:</i> include Country code, exclude access code.
time_zone	Customer's time zone	Customer's time zone. Should be of the form "GMT", "GMT+1", "CET", etc. Websites such as http://www.greenwichmeantime.com/ list possible time zones.
title	Customer Title (Mr, Mrs etc)	Max 10 characters

Attributes for Elements of CustomerInformation		
Attribute Name	Attribute of element	Value / Limitations
count	previous_purchases	Number of previous purchases. Numeric
amount	previous_purchases	Total value of previous purchases. Max 15 numeric including decimal point

Example XML for CustomerInformation complex elements

```

<CustomerInformation>
  <customer_reference>CUSTREF000001</customer_reference>
  <title>Mr</title>
  <forename>John</forename>
  <surname>Smith</surname>
  <telephone>0131 123 1234</telephone>
  <alt_telephone>0131 123 1234</alt_telephone>
  <email>jsmith@devnull.co.uk</email>
  <ip_address>192.168.0.1</ip_address>
  <customer_dob>1980-03-12</customer_dob>
  <first_purchase_date>2004-02-21</first_purchase_date>
  <previous_purchases count="5" value="58.94"/>
  <introduced_by>CUSTREF0004444</introduced_by>
  <driving_license_number>SMITH0987654321</driving_license_number>
  <time_zone>GMT</time_zone>
</CustomerInformation>

<CustomerInformation>
  <forename>John</forename>
  <surname>Smith</surname>
  <telephone/>
  <email/>
  <delivery_forename>Alice</delivery_forename>
  <delivery_surname>Smith</delivery_surname>
  <delivery_phone_number/>
</CustomerInformation>

```

2.3.7.1.3 DeliveryAddress

Element Name: DeliveryAddress
Position: Request.Transaction.TxnDetails.The3rdMan

Elements of DeliveryAddress		
Element Name	Description	Values / Limitations
street_address_1	First address line	Max 100 characters
street_address_2	Second address line	Max 100 characters
city	City	Max 20 characters
county	County	Max 30 characters
postcode	Postcode	Max 9 characters (or 10 for Amex transactions)
country	Country Code	Use the numeric country codes

Example XML for DeliveryAddress complex elements

```
<DeliveryAddress>
    <street_address_1>10 Stratford Road</street_address_1>
    <street_address_2>Windsor</street_address_2>
    <city>London</city>
    <county>London</county>
    <country>826</country>
    <postcode>AB1 2CD</postcode>
</DeliveryAddress>
```

2.3.7.1.4 BillingAddress

Element Name: BillingAddress

Position: Request.Transaction.TxnDetails.The3rdMan

The child elements of BillingAddress are the same as DeliveryAddress. It is important to note that this element must be populated in addition to the information present in the TxnDetails element.

Example XML for BillingAddress complex elements

```
<BillingAddress>
    <street_address_1>Mulberry House</street_address_1>
    <street_address_2>15 Acacia Place</street_address_2>
    <city>Edinburgh</city>
    <county>Lothian</county>
    <country>826</country>
    <postcode>EH6 7EZ</postcode>
</BillingAddress>

<BillingAddress>
    <street_address_1>10 Banana Gardens</street_address_1>
    <city>Chepstowe</city>
    <county>826</county>
    <postcode>CH11 4XX</postcode>
</BillingAddress>
```

2.3.7.1.5 DeliveryAddress

Element Name: DeliveryAddress

Position: Request.Transaction.TxnDetails.The3rdMan.AdditionalDeliveryAddresses

Up to four additional delivery addresses can be provided within the AdditionalDeliveryAddresses element. Each DeliveryAddress element should have the "number" attribute set with a value of 1 to 4. Please refer to the example XML at the end of this section.

Elements of DeliveryAddress		
Element Name	Description	Values / Limitations
title		Max 10 characters
forename		Max 50 characters
surname		Max 50 characters
street_address_1	First address line	Max 100 characters
street_address_2	Second address line	Max 100 characters
city	City	Max 20 characters
county	County	Max 30 characters
country	Country Code	Use the numeric country codes
postcode	Postcode	Max 9 characters (or 10 for Amex transactions)

Example XML for DeliveryAddress complex elements

```

<AdditionalDeliveryAddresses>
  <DeliveryAddress number="1">
    <title>MR</title>
    <forename>D</forename>
    <surname>Jones</surname>
    <street_address_1>10 Stratford Road</street_address_1>
    <street_address_2>Windsor</street_address_2>
    <city>London</city>
    <country>826</country>
    <postcode>AB1 2CD</postcode>
  </DeliveryAddress>
  <DeliveryAddress number="2">
    <title>MR</title>
    <forename>P</forename>
    <surname>Sutherland</surname>
    <street_address_1>12 Stratford Road</street_address_1>
    <street_address_2>Windsor</street_address_2>
    <city>London</city>
    <country>826</country>
    <postcode>AB1 2CD</postcode>
  </DeliveryAddress>
</AdditionalDeliveryAddresses>

```

2.3.7.1.6 AccountInformation

Element Name:	AccountInformation
Position:	Request.Transaction.TxnDetails.The3rdMan

Elements of AccountInformation		
Element Name	Description	Values / Limitations
Address	See section 2.3.3.1.3	
BankInformation	See section 2.3.7.1.8	
PreviousDetails	See section 2.3.7.1.9	
PurchaseInformation	See section 2.3.7.1.10	
UserDetails	See section 2.3.7.1.11	
title		Max 10 characters
forename		Max 50 characters
surname		Max 50 characters
phone		Max 20 characters
loyalty		Max 50 characters
numbers		Max 50 characters

open_date		Format YYYY-MM-DD
current_use		Max 30 characters
previous_use		Max 30 characters
remaining		Max 30 characters
balance	This field supports the currency attribute. See example below.	Format NNNNNNNNNNNN.NN
account_profit_loss		Format NNNNNNNNNNNN.NN
account_status		Max 50 characters
alt_email		Max 50 characters.
authentication_method	KYC check – refer to DFS	Max 30 characters.
browser_language		Max 30 characters
derived_language		Max 30 characters
discount_amount		Format NNNNNNNNNNNN.NN
exposure_amount		Format NNNNNNNNNNNN.NN
fraud_status		Max 30 characters
games_most_played		Max 50 characters
number_bonus_taken		Numeric
number_cards_registered		Numeric
order_status		Max 30 characters
previous_guest		Max 10 characters
proposition_date		Format YYYY-MM-DD
rake		Max 30 characters
session_id		Max 100 characters
total_account_withdrawal		Format NNNNNNNNNNNN.NN
vendor_id		Max 30 characters

Example XML for AccountInformation complex elements

```
<AccountInformation>
    <Address>...</Address>
    <BankInformation>...</BankInformation >
    <PreviousDetails>...</PreviousDetails >
    <PurchaseInformation>...</PurchaseInformation >
    <UserDetails>...</UserDetails>
    <title>Mr</title>
    <forename>Steven</forename>
    <surname>Benson</surname>
    <phone>01505000000</phone>
    <loyalty>Tier 1</loyalty>
    <numbers>1235</numbers>
    <open_date>2010-12-02</open_date>
    <current_use>usage</current_use>
    <previous_use>previous</previous_use>
    <remaining>remaining</remaining>
    <balance currency="EUR">22.55</balance>
    <account_profit_loss>600.53</account_profit_loss>
    <account_status>Active</account_status>
    <alt_email>test@datacash.com</alt_email>
    <authentication_method>POST</authentication_method>
    <browser_language>EN</browser_language>
    <derived_language>EN</derived_language>
    <discount_amount>5.50</discount_amount>
    <exposure_amount>2.50</exposure_amount>
    <fraud_status>OK</fraud_status>
    <games_most_played>Bejewelled</games_most_played>
    <number_bonus_taken>5</number_bonus_taken>
    <number_cards_registered>2</number_cards_registered>
    <order_status>SENT</order_status>
    <previous_guest>Yes</previous_guest>
    <proposition_date>2010-11-15</proposition_date>
    <rake>rake</rake>
    <session_id>1IHKD32</session_id>
    <total_account_withdrawal>55.55</total_account_withdrawal>
    <vendor_id>123</vendor_id>
</AccountInformation>
```

2.3.7.1.7 Address

Element Name: Address
Position: Request.Transaction.TxnDetails.The3rdMan.AccountInformation

The child elements of Address are the same as DeliveryAddress. See section 0 for a description of these elements.

2.3.7.1.8 BankInformation

Element Name: BankInformation
Position: Request.Transaction.TxnDetails.The3rdMan.AccountInformation

Elements of BankInformation		
Element Name	Description	Values / Limitations
account_number		Max 30 characters
bank_address		Max 100 characters
bank_country		
bank_name		Max 50 characters
customer_name		Max 50 characters
sort_code		Max 20 characters

Example XML for BankInformation element

```
<BankInformation>
  <account_number>00000000</account_number>
  <bank_address>12 Bank Street</bank_address>
  <bank_country>Scotland</bank_country>
  <bank_name>Santander</bank_name>
  <customer_name>MR J K BANK</customer_name>
  <sort_code>00-00-00</sort_code>
</BankInformation>
```

2.3.7.1.9 PreviousDetails

Element Name:	PreviousDetails
Position:	Request.Transaction.TxnDetails.The3rdMan.AccountInformation

This container element holds up to five PreviousDetail elements. This element is described in the table below. The "number" attribute should be provided on each PreviousDetail element. Please see the example XML at the end of this section for typical usage.

Elements of PreviousDetail		
Element Name	Description	Values / Limitations
amount_in		Format NNNNNNNNNNNN.NN
amount_out		Format NNNNNNNNNNNN.NN
currency	ISO currency code standard (ISO 4217)	(e.g. GBP, EUR, USD). Max 3 characters
date		YYYY-MM-DD HH:MM:SS
details		Max 100 characters

Example XML for PreviousDetails complex element

```
<PreviousDetails>
  <PreviousDetail number="1">
    <amount_in>52.52</amount_in>
    <amount_out>30.65</amount_out>
    <currency>GBP</currency>
    <date>2010-10-12 11:12:58</date>
    <details>Detail text</details>
  </PreviousDetail>
  <PreviousDetail number="2">
    <amount_in>552.63</amount_in>
    <amount_out>125.25</amount_out>
    <currency>EUR</currency>
    <date>2010-12-01 15:27:46</date>
    <details>Detail text</details>
  </PreviousDetail>
</PreviousDetails>
```

2.3.7.1.10 PurchaseInformation

Element Name: PurchaseInformation
Position: Request.Transaction.TxnDetails.The3rdMan.AccountInformation

Elements of PurchaseInformation		
Element Name	Description	Values / Limitations
avg		Format NNNNNNNNNNNN.NN
max		Format NNNNNNNNNNNN.NN
min		Format NNNNNNNNNNNN.NN

Example XML for PreviousDetails element

```
<AccountInformation>
  <PurchaseInformation>
    <avg>52.47</avg>
    <max>100.52</max>
    <min>10.28</min>
  </PurchaseInformation>
</AccountInformation>
```

2.3.7.1.11 UserDetails

Element Name: UserDetails
Position: Request.Transaction.TxnDetails.The3rdMan.AccountInformation

Elements of UserDetails		
Element Name	Description	Values / Limitations
alias		Max 50 characters
id		Max 50 characters
profile	This field supports the "number" attribute which allows the profile number to be specified (1-3). See example below.	Max 100 characters
machine_id		Max 100 characters

Example XML for UserDetails element

```
<AccountInformation>
  <UserDetails>
    <alias>rtr</alias>
    <id>testid</id>
    <profile number="1">profile1</profile>
    <profile number="2">profile2</profile>
    <profile number="3">profile3</profile>
    <machine_id>xc334sv</machine_id>
  </UserDetails>
</AccountInformation>
```

2.3.7.1.12 AltPayment

Element Name: AltPayment

Position: Request.Transaction.TxnDetails.The3rdMan

Elements of AltPayment		
Element Name	Description	Values / Limitations
BillingAddress	See section 2.3.7.1.13	
Card	See section 2.3.7.1.14	
CustomerInformation	See section 2.3.7.1.15	
Response	See section 2.3.7.1.16	
Secure	See section 2.3.7.1.17	
amount	The transaction amount.	Format NNNNNNNNNNNN.NN
payment_method		

Example XML for AltPayment element

```
<AltPayment>
  <BillingAddress>...</BillingAddress>
  <Card>...</Card>
  <CustomerInformation>...</CustomerInformation>
  <Card>...</Card>
  <Response>...</Response>
  <Secure>...</Secure>
  <amount>55.55</amount>
</AltPayment>
```

2.3.7.1.13 BillingAddress

Element Name: BillingAddress

Position: Request.Transaction.TxnDetails.The3rdMan.AltPayment

The child elements of BillingAddress are the same as DeliveryAddress. See section 0 for a description of these elements.

2.3.7.1.14 Card

Element Name: Card

Position: Request.Transaction.TxnDetails.The3rdMan.AltPayment

Elements of Card		
Element Name	Description	Values / Limitations
alt_pan	The full card number. If you prefer not to provide this you can provide alt_bin, alt_last_4_digits and alt_pan_sha1 instead.	Numeric. Max length 19
alt_bin	First 8 digits of card number	Numeric. Max length 8.
alt_last_4_digits	Last 4 digits of card number	Numeric. Max length 4.
alt_pan_sha1	May be provided as an alternative to the alt_pan field. Please contact support for additional data to be added to each card prior to SHA-1 hashing.	Upper case Hex. Max length 50
alt_expirydate	The card expiry date.	MMYY Format
alt_issuenumber_or_startdate	The card issue number or start_date	Numberic. Max length 2

Example XML for Card element

```

<AltPayment>
  <Card>
    <alt_pan>0000000000000000</alt_pan>
    <alt_expirydate>1012</alt_expirydate>
  </Card>
</AltPayment>

```

2.3.7.1.15 CustomerInformation

Element Name: CustomerInformation
Position: Request.Transaction.TxnDetails.The3rdMan.AltPayment

Elements of CustomerInformation		
Element Name	Description	Values / Limitations
title	If Cardholder name cannot be split then put all details in Cardholder Surname field.	Max 10 characters.
forename	The cardholder forename.	Max 50 characters.
surname	The cardholder surname.	Max 50 characters.

Example XML for CustomerInformation element

```
<AltPayment>
  <CustomerInformation>
    <surname>Mr J Hamilton</surname>
  </CustomerInformation>
</AltPayment>
```

2.3.7.1.16 Response

Element Name: Response

Position: Request.Transaction.TxnDetails.The3rdMan.AltPayment

Elements of Response		
Element Name	Description	Values / Limitations
auth_code	The bank authorisation code may be null if not applicable and is represented as the empty string. If the transaction type is Update, bank authorisation code should contain the status of the transaction (either OK, Rejected, Chargeback or Chargeback Fraud)	Max 30 characters
avs_address_response		0 = not given 1 = not checked 2 = pass 4 = fail
avs_postcode_response		0 = not given 1 = not checked 2 = pass 4 = fail
bank_response_code	The response code as received from the bank	Numeric. Max length 2
bank_response_message	The authorisation message as received from the bank	Max 100 characters
cv2_response	Security code response	0 = not given 1 = not checked 2 = pass 4 = fail

2.3.7.1.17 Secure

Element Name: Secure**Position:** Request.Transaction.TxnDetails.The3rdMan.AltPayment

Elements of Response		
Element Name	Description	Values / Limitations
eci	Transaction ECI Status.	Numeric. Max length 2
security_code	Visa transactions – CAVV Mastercard transactions –AVV	Max 50 characters

2.3.7.1.18 OrderInformation

Element Name: OrderInformation
Position: Request.Transaction.TxnDetails.The3rdMan

Elements of OrderInformation		
Element Name	Description	Values / Limitations
Products	See section 2.3.7.1.19	
brand	The brand, if the sales channel supports multiple brands	Max 30 characters
consumer_opt_in	Customer opt-in flag	0 - no 1 - yes
destination_location	Used for travel and ticketing. Route codes may be used and defined as required with the DFS.	Max 50 characters
distribution_channel	The distribution channel	Max 30 characters
event_date	The date the event takes place, e.g. date of travel, or date of show	YYYY-MM-DD format
event_location	The event location	Max 50 characters
gift_message	Message entered on gift card	Max 100 characters
installation_requested	Whether goods are to be supplied together with physical installation	Yes No
loyalty_card_number	The loyalty card number, if held by customer	Max 50 characters
operator_id		Max 30 characters
payment_method		
route_via_location	Routing information	Max 50 characters

Example XML for OrderInformation complex elements

```
<OrderInformation>
  <distribution_channel>First Class Post</distribution_channel>
  <Products count="6">...</Products>
</OrderInformation>
```

2.3.7.1.19 Products

This element enables details of the items within the order to be passed.

Element Name:	Products
Position:	Request.Transaction.TxnDetails.The3rdMan.OrderInformation

Elements of Products		
Element Name	Description	Values / Limitations
Product	See section 2.3.7.1.19	One element for each distinct item ordered

Attributes of Products		
Attribute name	Description	Values / Limitations
count	Total number of items in the order	Must match the total of the items specified in the <code>Product.count</code> fields.

Example XML for Products complex elements

```

<Products count="1">
    <Product>...</Product>
</Products>

<Products count="2">
    <Product>...</Product>
    <Product>...</Product>
</Products>

```

2.3.7.1.20 Product

This enables specific information about each distinct item within the order to be presented. If the Products element is specified, at least one Product element must be present.

Element Product
Name:

Position: Request.Transaction.TxnDetails.The3rdMan.OrderInformation.Products

Elements of Product		
Element Name	Description	Values / Limitations
AirlineInformation	See section 2.3.7.1.21	
code	The product code for the item	Max 30 characters
price	The unit cost of a single item	Numeric with decimal point
prod_category	The product category	Max 50 characters
prod_description	The product description, For travel use DD MMM YYYY HH:MM (e.g. 27 Feb 2008 17:50)	Max 100 characters
prod_id	The product ID, supports the type attribute. See example below.	
prod_risk	The level of risk associated with the product	Max 50 characters
prod_type	The type of product. For airlines please use 3 character IATA airport code (E.g. CDG)	Max 50 characters
quantity	The number of items of this product ordered	Non-negative integer

Example XML for Product complex element

```
<Product>
  <code>32231</code>
  <prod_id type="a">125</prod_id>
  <quantity>2</quantity>
  <price>22.99</price>
</Product>
```

2.3.7.1.21 AirlineInformation

Element Name: AirlineInformation
Position: ...The3rdMan.OrderInformation.Products.Product

Elements of AirlineInformation		
Element Name	Description	Values / Limitations
flight_carrier	This should be a 2 character IATA airline code.	Max 30 characters
dob	The passengers date of birth.	YYYY-MM-DD
fare_class	The fare class for the airline ticket.	Max 30 characters
flight_number	The flight number.	Max 30 characters
loyalty_indicator	The loyalty indicator for the passenger.	Max 30 characters
loyalty_number	The passenger loyalty number	Max 30 characters
nationality	The passenger nationality.	Max 30 characters
passport_or_ssn	The passengers passport or social security number.	Max 50 characters
route_via	If multiple points then separate with - (e.g. LHR-DUB)	Max 50 characters

Example XML for AirlineInformation elements

```
<AirlineInformation>
  <flight_carrier>BA</flight_carrier>
  <dob>1970-06-16</dob>
  <fare_class>Y</fare_class>
  <flight_number>059</flight_number>
  <loyalty_indicator>Y</loyalty_indicator>
  <loyalty_number>9876</loyalty_number>
  <nationality>SA</nationality>
  <passport_or_ssn>456789032</passport_or_ssn>
  <route_via>LHR-DUB</route_via>
</AirlineInformation>
```

2.3.7.1.22 AdditionalInformation

Element Name: AdditionalInformation
Position: Request.Transaction.TxnDetails.The3rdMan

The following additional elements can be used to send additional information to the DFS. The contents of these fields should be agreed with the DFS before they are used.

Elements of AdditionalInformation		
Element Name	Description	Values / Limitations
count	To be agreed with the DFS. Requires the "number" attribute to be set (see example). Up to 3 separate counts can be provided (1-3).	Numeric
generic	To be agreed with the DFS. Requires the "number" attribute to be set (see example). Up to 18 generic fields can be provided (1-18).	Max 100 characters

Example XML for AdditionalInformation elements

```

<AdditionalInformation>
    <count number="1">155576</count>
    <count number="2">18521</count>
    <count number="3">12568</count>
    <generic number="1">Data agreed with The 3rd Man</generic>
    <generic number="2">Data agreed with The 3rd Man</generic>
    <generic number="3">Data agreed with The 3rd Man</generic>
    <generic number="4">Data agreed with The 3rd Man</generic>
    <generic number="5">Data agreed with The 3rd Man</generic>
    ...
    <generic number="17">Data agreed with The 3rd Man</generic>
    <generic number="18">Data agreed with The 3rd Man</generic>
</AdditionalInformation>
```

2.3.7.2 Schema Elements for Response

As the results of the service will be returned to you directly by the DFS, there are no additional XML Response elements for this service.

Please refer to section 2.1.1.3 for details of the general response elements.

2.3.7.3 HTTPS Responses

See section 2.3.11

2.3.7.4 Example XML Responses

A complete list of Response Codes for this service is available on the website. The Support Centre also contains extensive examples for most error codes – including XML Responses - and also contains suggestions to prevent them occurring.

- [Support Centre](#)
- [Developers Area](#)

2.3.8 Fraud Only Transactions

A Fraud Only transaction type exists which allows information to be sent to the DFS. As the name suggests, this transaction type exists purely to send information to The DFS, and will not result in any bank communication / authorisations taking place.

Element Name: FraudOnlyTxn
Position: Request.Transaction.FraudOnlyTxn

Elements of FraudOnlyTxn		
Attribute name	Description	Values / Limitations
Card	See section 2.6.1.1.5	
response	See section 2.3.7.1.16	
secure	See section 2.3.7.1.17	
amount	The transaction amount.	Supports the currency attribute.
tran_type	The payment transaction type associated with a transaction	FraudOnly Auth Refund Pre-auth Deferred Manual Repeat Repeat Deferred Void Cancel Update SuperSearch Account Registration Account Amendment Buyer Registration Seller Registration KYC Withdrawal Amendment Bank Response Update Other

		FraudOnly Auth Refund Pre-auth Deferred Manual Repeat Repeat Deferred Void Cancel Update SuperSearch Account Registration Account Amendment Buyer Registration Seller Registration KYC Withdrawal Amendment Bank Response Update Other
alt_tran_type	The payment transaction type associated with a transaction. If the transaction is being passed to T3M to notify them of a status (e.g. an order has been set to Rejected, OK, Chargeback or Chargeback Fraud - a transaction type of Update should be given)	
acquirer	The merchant acquirer's name	Max 30

Example XML for FraudOnlyTxn element

```
<Request>
...
<Transaction>
    <TxnDetails>
        <merchantreference>12345</merchantreference>
    </TxnDetails>
    <FraudOnlyTxn>
        <Card>...</Card>
        <Response>...</Response>
        <Secure>...</Secure>
        <amount currency="GBP">12.50</amount>
        <tran_type>...</tran_type>
        <acquirer>...</acquirer>
    </FraudOnlyTxn>
</Transaction>
...
</Request>
```

2.3.8.1 Card

Element Name:	Card
Position:	Request.Transaction.FraudOnlyTxn

Elements of Card		
Element Name	Description	Values / Limitations
Pan	The full card number. If you prefer not to provide this you can provide alt_bin, alt_last_4_digits and alt_pan_sha1 instead.	Numeric. Max length 19
Bin	First 8 digits of card number	Numeric. Max length 8.
last_4_digits	Last 4 digits of card number	Numeric. Max length 4.
pan_shal	May be provided as an alternative to the alt_pan field. Please contact support for additional data to be added to each card prior to SHA-1 hashing.	Upper case Hex. Max length 50
expirydate	The card expiry date.	MMYY Format
issuenumber_or_startdate	The card issue number or start_date	Issue Number (Numeric 2) Start Date (MMYY)(Numeric 4)

[Back to Contents](#)

2.3.9 Risk Services Gateway (RSG)

Note that in order to use the RSG service, you must first be subscribed. If you require additional information, please contact support. The RSG service cannot be used in conjunction with any other fraud screening services that have been enabled for your vTID, such as ReD or T3M.

This section of documentation assumes the reader is familiar with the Credit and Debit Card Service, which is discussed in section 2.2.1.

2.3.9.1 Service Overview

The RSG service provides the merchant with the ability to send their transaction information to Risk Systems for fraud screening. It is also possible to send a number of additional data fields containing information relevant to the transaction in order to aid the screening process. These additional fields will be explained in detail throughout the sections that follow.

2.3.9.2 Formulating a Request

The following sections explain the additional XML Elements that should be sent in the request in order to use the RSG service.

2.3.9.2.1 Risk Element

2.3.9.2.2 The Risk element acts as a container for all additional transaction information which will be passed to Risk Systems for fraud screening. This element can be presented to the DPG within the TxnDetails Element of a transaction. Note that the Risk element can only be provided when using the "pre" or "auth" methods. No other methods are supported. Additionally, Fire and Forget Continuous Authority and Historic Continuous Authority transactions cannot be fraud screened. The initial Continuous Authority setup is the only exception to this rule.

Element Name: Risk

Position: Request.Transaction.TxnDetails

2.3.9.2.3 Action Element

The Action container is defined as containing the service instruction, as well as all relevant data containers in order to execute the actions related to the services requested.

Element Name: Action
Position: Request.Transaction.TxnDetails.Risk

Attributes of Action		
Attribute name	Description	Values / Limitations
Service	<p>This will indicate the action(s) to be performed, based on a set of predefined services.</p> <p>Values can be:</p> <ul style="list-style-type: none"> • “1” for pre-authorisation fraud screening request • “2” for post-authorisation fraud screening request 	Must be either “1” or “2”

2.3.9.2.4 Modes of Operation / Service Instruction

The DPG supports two modes of operation. Pre-Auth or Post-Auth fraud checking. There is an important distinction in behaviour between these two modes which will be explained below.

2.3.9.2.4.1 Post-Auth Fraud Checking Mode

In Post-Auth fraud checking mode, your transaction will be sent to the bank for authorisation in the normal manner. Your transaction details including any additional data fields are then sent to the Risk Services Gateway along with the bank response for the authorisation. The transaction response XML returned to you will include the screening response generated by the RSG.

You can examine this response and determine whether or not to continue with the transaction. It is important to note that in Post-Auth fraud checking mode, the RSG response acts as an aid in determining the best course of action for a particular transaction. If you decide that a particular transaction is too risky based on the information returned in the RSG response, the responsibility lies with you and/or your systems to correctly follow up on any decisions (e.g. cancel the transaction by sending a “cancel” request to the DPG).

2.3.9.2.4.2 Pre-Auth Fraud Checking Mode

In Pre-Auth fraud checking mode, the DPG takes an active role in managing potentially risky transactions. Your transaction details, including any additional data fields will be sent to the Risk Services Gateway **before** it is sent to the bank for authorisation.

The DPG will then analyse the response from the Risk Services Gateway and take 1 of

3 actions:

If the response from the RSG indicates that the transaction is of relatively low risk. The DPG will continue as normal by sending the transaction to the bank for authorisation. The RSG screening response will be present in the DPG response XML, which you can store for your records.

If the response from the RSG indicates that the transaction is of a medium risk, the DPG will mark the transaction for review and it will **not** be sent for authorisation at this time. You will receive response code 1127 "RSG: review" in the DPG response XML along with the screening response from the RSG. Transactions which have been marked for review can be inspected manually by yourself and/or systems and continued by sending a follow up "accept_review" transaction. The DPG will then proceed to send an authorisation message to the bank in the usual manner. It should be noted that this behaviour can be overridden by our support team on a merchants request , preventing transactions being marked for review, thus requiring no manual intervention.

If the RSG has indicated in its response that the transaction is extremely risky, the DPG will outright reject the transaction and it will **not** be sent for authorisation. You will receive response code 1126 "RSG: rejected" in the DPG response XML. It is important to note that transactions which have been rejected with this response cannot be continued.

2.3.9.2.5 Data Elements / Containers

Element Name(s):	MerchantConfiguration CustomerDetails
Position:	Request.Transaction.TxnDetails.Risk.Action

Each message type has distinct data requirements needed to process the requested message.

The current API supports 3 different types of request message structures for service requests 1 and 2, these message structures depend on the industry which the merchant has been setup for, i.e. gaming, telephony, e-wallet funding, retail or airline.

The gaming message structure is a basic, generic structure and also serves to be used by telephony merchants and e-wallet providers where the funding of the e-wallet is screened (from here on referred to as e-wallet funding).

The gaming model is the most generic and serves as a base model for retail and airline, where additional data containers have been added that are retail and airline specific.

The data for screening service requests ("1" & "2") must be supplied under the following headings:

- Merchant Configured Details
- Customer Details
 - Risk Details

- Personal Details
- Address Details
- **Shipping Details ****
- Payment Details
- Order Details
- **Line Item Details ****
- **Journey Details *****
- **Leg Items *****
- **Passenger Items *****

Note: The data sections displayed in black above are the basic data structures that should be used for all screening requests (service 1 and 2) for gaming, telephony, e-wallet funding, retail and airline, unless otherwise indicated in the field descriptions, where it might state that a field is only required for airline or retail.

****Note:** These data sections must only be included when sending screening requests for merchants that have been configured as **retail merchants** (in addition to the basic data containers shown in black above). Where multiple items are ordered, multiple item detail headings should be sent, one for each distinct line item, as appropriate.

*****Note:** These sections are airline industry specific message sections, and only need to be included when sending a screening request for a merchant that has been configured as an **airline merchant** (in addition to the basic data containers shown in black above).

The following table describes all the data fields that can be used across all actions and is applicable to incoming messages. The table also indicates the required data fields for each incoming message.

Additionally, there is an indication of requirements around each field, where the following criteria were applied:

Key	Description
•	Tag is required and needs to be populated with a value.
◦	Tag or value is not required.

2.3.10 Pre & Post Authorisation Fraud Screening

Field	Type	Industry			
		Gaming	Telephony	Retail	Airline
Merchant Configuration		•	•	•	•
Field	Type				

<p>tag name: channel</p> <p>definition: This reflects the manner in which the customer transaction has been captured. (This will be used to distinguish the 'ticketless' on-line purchaser, from Moto taken orders directly by a booking system)</p> <p>possible values:</p> <ul style="list-style-type: none"> P = Physical M = Moto W = Web K = Kiosk I = In-Store S = Mail Order O = Other 	Char (1)				
<p>tag name: merchant_location</p> <p>definition: This represents the location of stores or outlets for a reseller or airline.</p>	Varchar (30)			o	•
Merchant Configuration		•	•	•	•
- Callback Configuration			o		
Field	Type				
<p>tag name: callback_format</p> <p>definition:</p> <p>possible values:</p> <ul style="list-style-type: none"> HTTP XML SOAP 	Varchar(4)			o	
<p>tag name: callback_url</p> <p>definition: The callback URL used by this transaction if applicable.</p>	Varchar(200)			o	

<p>tag name: callback_options definition: Use either/or both the response and the customer/aggregator registered callback.</p> <p>possible values: 0 = Immediate real-time callback or 1 = Customer 2 = This callback (default) 3 = Both 4 = Polled when not available or combination of the two. i.e. 02 would be realtime callback and callback on backend response.</p>	Char(2)				
Customer Details	•	•	•	•	
- Risk Details	•	•	•	•	
Field	Type				
<p>tag name: account_number definition: An account number/unique ref number at the Merchant associated with the individual who is transacting with the merchant.</p>	Varchar (30)	•	•	◦	•
<p>tag name: email_address definition: Customer's Email Address. Note: These are generally associated with the individual who is transacting with the merchant.</p>	Varchar (64)	◦	◦	◦	◦
<p>tag name: alt_email_address definition: Alternative e-mail address Note: These are generally associated with the individual who is transacting with the merchant.</p>	Varchar(64)			◦	

tag name: session_id definition: a web server generated session id.	Varchar (255)			o	o
tag name: ip_address definition: Customer's IP Address.	Varchar (15) Format 255.255.255.2	o	o	o	o
tag name: user_id definition: This is a Merchant configurable data field typically used to identify a customer uniquely within the system, regardless of the amount of accounts that the customer has.	Varchar (36)	o	o		o
tag name: usermachine_id definition: This is an ID used to identify the computer (actual hardware) uniquely within the system, regardless of the amount of customers or accounts that's using the computer. Examples of the User Machine ID are Motherboard ID, Hard drive ID, CPU ID, etc.	Varchar (255)	o	o		o
tag name: user_profile definition: This is a customizable field that may be used by the merchant to supplement fraud screening processes according to a risk classification set by the merchant. (e.g. VIP flag, Indication whether the transaction is part of a promotion, a confidence level around the identity of the customer, etc.)	Varchar (20)	o	o		o

tag name: user_profile_2 definition: This is a customizable field that may be used by the merchant to supplement fraud screening processes according to a risk classification set by the merchant. (See above examples.)	Varchar (20)	◦	◦		◦
tag name: user_profile_3 definition: This is a customizable field that may be used by the merchant to supplement fraud screening processes according to a risk classification set by the merchant. (See above examples.)	Varchar (20)	◦	◦		◦
tag name: register_consumer_watch definition: Register the consumer associated with this transaction for the consumer product. possible values: Y = Yes N = No	Char(1)				◦
tag name: browser_language definition: Language setting of the respective browser	Varchar(30)			◦	
Customer Details		•	•	•	•
- Personal Details		•	•	•	•
Note - these are generally associated with the individual who consents to the merchant's customer terms & conditions and is transacting with the merchant.					
Field	Type				
tag name: first_name definition: Customer's name. Note: The first_name field and the surname field combined should not exceed 32 characters.	Varchar (32)	•	•	•	•

tag name: surname definition: Customer's surname Note: The <i>first_name</i> field and the <i>surname</i> field combined should not exceed 32 characters.	Varchar (32)	•	•	•	•
tag name: telephone definition: Customer's telephone number.	Varchar (20)	•	•	◦	•
tag name: telephone_2 definition: Alternative telephone number.	Varchar (20)	◦	◦	◦	◦
tag name: date_of_birth definition: Customer's date of birth.	Datetime Format yyyy-mm-dd	◦	◦	◦	◦
tag name: nationality definition: Customer's nationality, represented by a 2-character code. Please reference Appendix 11 for further details. Note: This is generally used in respect of certain 3rd party electronic ID verification services.	Char (2)	◦	◦	◦	◦
tag name: id_number definition: Passport, ID, or other verification document number. Note: This is generally used in respect of certain 3rd party electronic ID verification services.	Varchar (40)	◦	◦	◦	◦

<p>tag name: id_type</p> <p>definition: A code that defines the type of verification document supplied.</p> <p>possible values:</p> <ul style="list-style-type: none"> 1 = Passport 2 = Personal ID 3 = Identity Card 4 = Driver's License 8 = Travel Document 12 = Residence Permit 13 = Identity Certificate 16 = Registro Federal de Contribuyentes 17 = Credential de Elector <p>Note: This is generally used in respect of certain 3rd party electronic ID verification services.</p>	Int				
<p>tag name: ssn</p> <p>definition: Last 4 digits of the SSN number (US residents).</p> <p>Note: This is generally used in respect of certain 3rd party electronic ID verification services.</p>	Int	o	o	o	o
Customer Details		•	•	•	•
- Address Details		□	□	□	□
<i>Note - these are generally associated with the individual who consents to the merchant's customer terms & conditions and is transacting with the merchant.</i>					
Field	Type				
<p>tag name: address_line1</p> <p>definition: Customer's address line 1.</p>	Varchar (60)	•	•	•	•
<p>tag name: address_line2</p> <p>definition: Customer's address line 2.</p>	Varchar (60)	o	o	o	o
<p>tag name: city</p> <p>definition: Customer's City.</p>	Varchar (25)	•	•	•	•

tag name: state_province definition: Customer's State/Province/District/ County. Note: For USA and Ireland, Canada and the UK, a 2-character code needs to be supplied where the Merchant requests verification services in these countries. Please reference Appendix 24.	Varchar (25)	o	o	o	o
tag name: country definition: Customer's Country. Note: This is a 2-character code that needs to be supplied. Please reference Appendix 11 for further details.	Char (2)	•	•	o	o
tag name: zip_code definition: Customer's Zip/Postal code.	Varchar (10)	o	o	o	o
Customer Details		•	•	•	•
- Shipping Details				•	
Note - these are generally associated with the dispatch of goods/services to intended recipients.					
Field	Type				
tag name: title definition: Recipient's title	Varchar(10)			o	
tag name: first_name definition: Recipient's first name.	Varchar (50)			o	
tag name: surname definition: Recipient's surname.	Varchar (50)			o	
tag name: address_line1 definition: Customer's shipping address line 1.	Varchar (60)			o	
tag name: address_line2 definition: Customer's shipping address line 2.	Varchar (60)			o	

tag name: city definition: Customer's shipping city.	Varchar (25)			o	
tag name: state_province definition: Customer's shipping state or province. Note: For USA, Ireland, Canada and UK a 2-character code can be supplied for verification purposes, if additional verification checks are required on billing address (verification checks use primary address details). Please reference Appendix 24.	Varchar (25)			o	
tag name: country definition: Customer's shipping country. Note: This is a 2-character code that needs to be supplied. Please reference Appendix 11 for further details.	Char (2)			o	
tag name: zip_code definition: Customer's shipping zip code.	Varchar (10)			o	
tag name: delivery_date definition: Delivery date	Datetime Format yyyy-mm-dd			o	
tag name: delivery_method definition: Method of delivery	Varchar(30)			o	
tag name: installation_requested definition: Indicates whether goods are supplied together with a physical installation. possible values: Y = Yes N = No	Char(1)			o	

Customer Details		•	•	•	•
- Payment Details		•	•	•	•
Field	Type				
tag name: transaction_type definition: The payment transaction type associated with a transaction. possible values: Auth Blank Instant Other	Varchar(20)				◦
tag name: payment_method definition: This is the mechanism with which the client chooses to purchase. Note: For the screening of non-card based payments, please consult with DataCash as new options will be added. possible values: please refer to Appendix 25 Payment Method Codes.	Varchar (2)	•	•	•	•
Customer Details		•	•	•	•
- Payment Details		•	•	•	•
-- Wallet Details		•	•	•	•
Note – this is for any funding account details other than cards.					
Field	Type				
tag name: acquirer_number definition: Customer's electronic wallet account number or bank account used for purchasing. The value will be associated with the payment_method used for the transaction.	Varchar (60)	◦	◦	◦	◦
Customer Details		•	•	•	•
- Order Details		•	•	•	•
Field	Type				
tag name: discount_value definition: The discount amount relevant to the transaction in the smallest unit of the currency i.e. cents or pence, e.g. US\$100.01 = 10001 or ¥100 = 100.	Int			◦	

tag name: time_zone definition: Local time zone, GMT +/- hours (GMT = 0) example: +01:00	Time format (+/-hh:mm)			o	
tag name: proposition_date definition: Similar to a picking date, can be used to prioritise the referral queue so that delivery deadlines are met.	Datetime Format yyyy-mm-dd			o	
Customer Details		•	•	•	•
- Order Details		•	•	•	•
-- Billing Details		•	•	•	•
<i>Note - these are associated with the owner of the card number or acquirer number used to fund the respective purchase.</i>					
Field	Type				
tag name: name definition: The billing account owner's name associated with the transaction.	Varchar (50)	o	o	o	o
tag name: address_line1 definition: The billing account owner's billing address line 1.	Varchar (60)	o	o	o	o
tag name: address_line2 definition: The billing account owner's billing address line 2.	Varchar (60)	o	o	o	o
tag name: city definition: The billing account owner's billing city.	Varchar (25)	o	o	o	o

tag name: state_province definition: The billing account owner's state or province. Note: For USA, Ireland, Canada and UK a 2-character code can be supplied for verification purposes, if additional verification checks are required on billing (verification checks use primary address details). Please reference Appendix 24.	Varchar (25)	o	o	o	o
tag name: country definition: The billing account owner's billing country. Note: This is a 2-character code that needs to be supplied. Please reference Appendix 11 for further details.	Char (2)	•	•	o	•
tag name: zip_code definition: The billing account owner's billing zip code	Varchar (10)	o	o	o	o
Customer Details		•	•	•	•
- Order Details		•	•	•	•
-- Line Items				o	
Note - these are generally associated with shopping cart details within the retail sector where it is important to screen the items being purchased.					
--- Item				•	
Field	Type				
tag name: product_code definition: Line item product code. Note: If any one of the Line Item fields is submitted, then all Line Item fields become mandatory.	Varchar (50)			o	
tag name: product_description definition: Line item description.	Varchar (50)			o	
tag name: product_category definition: Type of item, as classified by the Merchant.	Varchar (80)			o	

tag name: product_risk definition: Indication of how susceptible the product is to fraud. possible values: High Medium Low	Varchar(50)			◦	
tag name: order_quantity definition: Number of items ordered. Note: If any one of the Line Item fields is submitted, then all Line Item fields become mandatory.	Int			◦	
tag name: unit_price definition: The amount of the transaction in the smallest unit of the currency i.e. cents or pence, e.g. US\$100.01 = 10001 or ¥100 = 100.	Int			◦	
Customer Details		•	•	•	•
- Airline Details					•
-- Journey					•
Field	Type				
tag name: ticket_number definition: the ticket number assigned to the journey.	Varchar (30)				◦
tag name: payer_travelling definition: an indication whether the person paying is also travelling on the ticket. possible values: Y = Yes N = No	Char(1)				◦
tag name: pnr definition: Passenger Name Record	Varchar(10)				◦
Customer Details		•	•	•	•
- Airline Details					•

-- Journey					•
Note - A journey is a connected series of one or more legs that are sold as a unit. A one-way trip consists of a single leg. A round trip consists of two legs (outbound and return). There may be more than two legs. Legs should be presented in the sequence of travel, starting with the first leg of the journey.					
- - - Legs					•
- - - Leg					•
Field	Type				
tag name: depart_airport definition: Departure point airport. Note: this is a 3 character code IATA airport code that needs to be supplied. http://www.iata.org	Char (3)				•
tag name: depart_country definition: Departure Country. Note: This is a 2-character code that needs to be supplied. Please reference Appendix 11 for further details.	Char (2)				◦
tag name: depart_datetime definition: Local Date & time of scheduled departure.	Datetime Format yyyy-mm-dd hh:mm:ss				•
tag name: depart_airport_timezone definition: Local time zone, GMT +/- hours. example: +01:00	Time format (+/- hh:mm)				•
tag name: arrival_airport definition: Arrival point. Note: this is a 3 character code IATA airport code that needs to be supplied. For more information on these codes visit http://www.iata.org	Char (3)				•

<p>tag name: carrier</p> <p>definition: Flight carrier for the journey.</p> <p>Note: this is a 2 character IATA airline code that needs to be supplied. For more information on these codes visit http://www.iata.org</p>	Char (2)				
<p>tag name: flight_number</p> <p>definition: Flight number for the journey.</p> <p>(Data Capture – not fed into rules engine)</p>	Char (4)				•
<p>tag name: fare_basiscode</p> <p>definition: The fare basis code provides information about the specific fare in addition to the class of service required for booking.</p> <p>example: HL7LNR</p>	Char (10)				•
<p>tag name: fare_class</p> <p>definition: Class of flight.</p> <p>example:</p> <p>F (=Unrestricted First Class)</p> <p>FR (=Restricted First Class)</p> <p>F (=Unrestricted Business Class)</p> <p>CR (=Restricted Business Class)</p> <p>Y (=Unrestricted Coach/Economy Class)</p> <p>YD (=Restricted Coach/Economy Class)</p> <p>(Data Capture – not fed into rules engine)</p>	Char (3)				•

tag name: base_fare definition: The amount of the transaction, excluding taxes and fees, in the smallest unit of the currency i.e. cents or pence. (e.g. US\$100.01 = 10001 OR ¥100 = 100). (Data Capture – not fed into rules engine. For reporting/review purposes. Fraud screening and related velocity calculations will initially be based upon the total value of the purchase transaction)	Int					
tag name: currency_code definition: This is the ISO numeric code for the currency in which the transaction should be processed. See Appendix 10 for applicable codes. This is mandatory if a <i>base_fare</i> is supplied. (Data Capture – not fed into rules engine. For reporting/review purposes)	Char (3)					
Customer Details		•	•	•	•	
- Airline Details						•
-- Journey						•
--- Passengers						•
---- Passenger						•
Field	Type					
tag name: first_name definition: Passenger's name. Note: The first-name field and the surname field combined should not exceed 32 characters. If any passenger details are submitted, then the first-name field become Mandatory.	Varchar (32)					o

<p>tag name: surname definition: Passenger's surname Note: The first-name field and the surname field combined should not exceed 32 characters. If any passenger details are submitted, then the surname field become Mandatory.</p>	Varchar (32)				o
<p>tag name: passenger_type definition: The type of passenger travelling, e.g. Adult, Child, Infant. possible values: A = Adult C = Child I = Infant</p>	Char (1)				•
<p>tag name: nationality definition Customer's Nationality, represented by a 2-character code. Please reference Appendix 11 for further details.</p>	Char (2)				o
<p>tag name: id_number definition: Passport, ID, or other verification document number. Note: fraud screening block lists may include a composite nationality and <i>id_number</i> value.</p>	Varchar (40)				o
<p>tag name: loyalty_number definition: Passenger's loyalty scheme number.</p>	Varchar (20)				o
<p>tag name: loyalty_type definition: Passenger's loyalty scheme (e.g. Voyager,)</p>	Varchar (20)				o

tag name: loyalty_tier definition: Passenger's loyalty scheme status or level possible values: 1 = highest tier level e.g. platinum 2 = second highest tier level e.g. gold 3 = third highest tier level e.g. silver...,etc	Int					
--	-----	--	--	--	--	--

2.3.10.1.1 Example XML Request

The XML below serves as an example of sending all possible information fields to be risk screened.

Example XML Element Risk for an RSG Transaction

```

<Risk>
  <Action service="1">
    <MerchantConfiguration>
      <merchant_location>Edinburgh</merchant_location>
      <channel>W</channel>
    </MerchantConfiguration>
    <CustomerDetails>
      <RiskDetails>
        <account_number>24972343</account_number>
        <email_address>email@example.com</email_address>
        <session_id>FJW35798FHI</session_id>
        <ip_address>127.0.0.1</ip_address>
        <user_id>333</user_id>
        <usermachine_id>38744</usermachine_id>
        <user_profile>profile</user_profile>
        <user_profile_2>profile1</user_profile_2>
        <user_profile_3>profile2</user_profile_3>
      </RiskDetails>
      <PersonalDetails>
        <first_name>Steven</first_name>
        <surname>Partridge</surname>
        <telephone>555-3937</telephone>
        <telephone_2>032749874738</telephone_2>
        <date_of_birth>1978-03-10</date_of_birth>
        <nationality>British</nationality>
        <id_number>43579</id_number>
        <id_type>Passport</id_type>
        <ssn>000-00-0000</ssn>
      </PersonalDetails>
      <AddressDetails>
        <address_line1>123 Brick Lane</address_line1>
        <address_line2>South Brickford</address_line2>
        <city>Livingston</city>
      </AddressDetails>
    </CustomerDetails>
  </Action>
</Risk>

```

```

<state_province>West Lothian</state_province>
<country>UK</country>
<zip_code>LI43 0SA</zip_code>
</AddressDetails>
<ShippingDetails>
<address_line1>123 Brick Lane</address_line1>
<address_line2>Brickford South</address_line2>
<city>Livingston</city>
<state_province>West Lothian</state_province>
<country>UK</country>
<zip_code>EH7 8EA</zip_code>
</ShippingDetails>
<PaymentDetails>
<payment_method>CC</payment_method>
<WalletDetails>
<acquirer_number>458</acquirer_number>
</WalletDetails>
</PaymentDetails>
<OrderDetails>
<BillingDetails>
<name>Mr Steven Partridge</name>
<address_line1>34 Fake Street</address_line1>
<address_line2>District 7</address_line2>
<city>Terminalville</city>
<state_province>Western Docklands</state_province>
<country>Tanzania</country>
<zip_code>347595</zip_code>
</BillingDetails>
<LineItems>
<Item>
<product_code>PRN383</product_code>
<product_description>Pair of
socks</product_description>
<product_category>Clothing</product_category>
<order_quantity>11</order_quantity>
<unit_price>4.11</unit_price>
</Item>
<Item>
<product_code>FHEUF</product_code>
<product_description>Model dog</product_description>
<product_category>Keepsake</product_category>
<order_quantity>2</order_quantity>
<unit_price>5.32</unit_price>
<line_total>10.64</line_total>
</Item>
</LineItems>
</OrderDetails>
<AirlineDetails>
<Journey>
<ticket_number>478935793</ticket_number>
<Legs>
<Leg>
<depart_airport>EDI</depart_airport>
<depart_country>UK</depart_country>
<depart_datetime>2030-10-01
13:23:11</depart_datetime>
<depart_airport_timezone>+00:00</depart_airport_timezone>

```

```

<arrival_airport>WRO</arrival_airport>
<carrier>CNB</carrier>
<flight_number>508</flight_number>
<fare_basiscode>LWC</fare_basiscode>
<fare_class>ECONOMY</fare_class>
<base_fare>30.00</base_fare>
<currency_code>GBP</currency_code>
</Leg>
<Leg>
<depart_airport>JHB</depart_airport>
<depart_country>SA</depart_country>
<depart_datetime>2030-08-01
13:23:11</depart_datetime>

<depart_airport_timezone>+01:00</depart_airport_timezone>
<arrival_airport>LHW</arrival_airport>
<carrier>SuperJet</carrier>
<flight_number>404</flight_number>
<fare_basiscode>FR</fare_basiscode>
<fare_class>Business</fare_class>
<base_fare>7844.00</base_fare>
<currency_code>RND</currency_code>
</Leg>
</Legs>
<Passengers>
<Passenger>
<first_name>Daniel</first_name>
<surname>Moore</surname>
<passenger_type>Business</passenger_type>
<nationality>British</nationality>
<id_number>666</id_number>
<loyalty_number>32974</loyalty_number>
<loyalty_type>Super Loyal</loyalty_type>
<loyalty_tier>Tier 3</loyalty_tier>
</Passenger>
<Passenger>
<first_name>Fed</first_name>
<surname>McFed</surname>
<passenger_type>Type</passenger_type>
<nationality>Canadian</nationality>
<id_number>574</id_number>
<loyalty_number>433</loyalty_number>
<loyalty_type>Disloyal</loyalty_type>
<loyalty_tier>Tier 1</loyalty_tier>
</Passenger>
</Passengers>
</Journey>
</AirlineDetails>
</CustomerDetails>
</Action>
</Risk>
```

2.3.10.2 XML Response

The following sections detail the response returned from the DPG and RSG respectively.

2.3.10.2.1 Pre-Authorisation Fraud Screening RSG Response

On receiving a pre-auth fraud screening request message, the Risk Services Gateway will send a response message back to the Merchant. The following table describes the return values of the response message:

Field Name	Description
Screening Response	
transaction_id	This is the DataCash reference of the original transaction which was sent for risk screening.
response_code	<p>This is a status/return code. It indicates whether the transaction was received and/or the result of the screening process.</p> <p>00 – reflecting that the transaction was accepted by Risk Services 01 – reflecting that the transaction was rejected by Risk Services 02 – reflecting that the transaction was flagged for review by Risk Services</p> <p>For the full set of response codes, refer to Appendix 24</p>
response_message	<p>This is a text field used to give a short text description of the status/return code.</p> <p>Transaction Approved Transaction not Approved Transaction marked for Review</p> <p>For the full set of response codes, refer to Appendix 24</p>
cpi_value	This is a Risk assessment score for a customer, available within select industries. Values can be 1-5 and will be based on definition discussed between the Merchant and DataCash.
Additional Messages	
message	This field will display any additional relevant response information. There can be more than one message tag for this purpose.

2.3.10.2.2 Example Pre-Auth Response XML

Example XML Response for an RSG transaction.

```
<Response>
  <CardTxn>
    <authcode>640519</authcode>
    <card_scheme>VISA</card_scheme>
  </CardTxn>
  <Risk>
    <action_response>
      <bankresult_response>
        <cpi_value>-1</cpi_value>
        <response_code>00</response_code>
        <response_message>Successful</response_message>
        <transaction_id>3200900010000275</transaction_id>
      </bankresult_response>
      <screening_response>
        <cpi_value>0</cpi_value>
        <response_code>00</response_code>
        <response_message>Transaction
Approved</response_message>
        <transaction_id>3200900010000275</transaction_id>
      </screening_response>
    </action_response>
  </Risk>
  <datacash_reference>3200900010000275</datacash_reference>
  <merchantreference>1aa5c31170306f1504</merchantreference>
  <mode>LIVE</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1288683112</time>
</Response>
```

2.3.10.2.3 Post- Authorisation Fraud Screening RSG Response

On receiving a post-auth fraud screening request message, the Risk Services Gateway will send a response message back to the Merchant. The following table describes the return values of the response message:

Field Name	Description
Screening Response	
transaction_id	This is the DataCash reference of the original transaction which was sent for risk screening.
response_code	<p>This is a status/return code. It indicates whether the transaction was received and/or the result of the screening process.</p> <p>00 – reflecting that the transaction was accepted by Risk Services 01 – reflecting that the transaction was rejected by Risk Services 02 – reflecting that the transaction was flagged for review by Risk Services</p> <p>For the full set of response codes, refer to Appendix 24</p>
response_message	<p>This is a text field used to give a short text description of the status/return code.</p> <p>Transaction Approved Transaction not Approved Transaction marked for Review</p>
cpi_value	This is a Risk assessment score for a customer, available within select industries. Values can be 1-5 and will be based on definition discussed between the Merchant and DataCash.
Additional Messages	
message	This field will display any additional relevant response information. There can be more than one message tag for this purpose.
Bank Result Response	
response_code	<p>This is a status/return code. It indicates whether the transaction was received and/or the result of the screening process.</p> <p>00 – reflecting that the bank result was updated successfully 01 – reflecting that the bank result was not updated successfully</p>
response_message	<p>This is a text field used to give a short text description of the status/return code.</p> <p>Successful Unsuccessful</p>
Additional Messages	
message	This field will display any additional relevant response information. There can be more than one message tag for this purpose.

2.3.10.2.4 Accepting a Transaction Marked for Review

If you have sent a Pre-Authorisation service instruction and receive response code 1127 "RSG: review", you can accept/continue this transaction by sending a follow up "accept_review" transaction.

A typical response indicating a transaction has been marked for review is shown below:

Example XML Response for an RSG transaction which has been marked for review.

```
<Response>
  <CardTxn>
    <card_scheme>VISA</card_scheme>
  </CardTxn>
  <Risk>
    <action_response>
      <screening_response>
        <cpi_value>0</cpi_value>
        <response_code>02</response_code>
        <response_message>Transaction marked for
Review</response_message>
        <transaction_id>3200900010000124</transaction_id>
      </screening_response>
    </action_response>
  </Risk>
  <datacash_reference>3200900010000124</datacash_reference>
  <information>Transaction marked for review by the
RSG</information>
  <merchantreference>3cb4b5ca105c388ea6</merchantreference>
  <mode>LIVE</mode>
  <reason>Transaction marked for review</reason>
  <status>1127</status>
  <time>1288167601</time>
</Response>
```

A typical “accept_review” request is shown below:

Example XML Request for an “accept_fraud” transaction.

```
<Request>
    <Authentication>
        <password>*****</password>
        <client>99000001</client>
    </Authentication>
    <Transaction>
        <HistoricTxn>
            <method>accept_review</method>
            <reference>123412345</reference>
        </HistoricTxn>
    </Transaction>
</Request>
```

2.3.11 HTTPS Responses

Responses for each transaction are posted back to your server on a transaction-by-transaction basis. This is referred to as a callback.

This will be in the form of an HTTP POST which contains the following fields in XML or as Name=Value fields separated by '&'.

	Description	Format (maximum length)
aggregator_identifier	Unique identifier for an aggregator (for example a PSP).	15
merchant_identifier	gatekeeper client ID	15
merchant_order_ref	The reference number the merchant gives to the order	250
t3m_id	Third Man internal identifier. Used for later calls such as the rules string request.	
Score	Score	-999 to +999
Recommendation	Recommendation	0=Release 1=Hold 2=Reject 9=Under Investigation

CallBack response will look as follows:

2.3.11.1 XML Response

```
<?xml version="1.0" ?>
<RealTimeResponse xmlns="T3MCallback">
    <aggregator_identifier/>
    <merchant_identifier>5567</merchant_identifier>
    <merchant_order_ref>12345</merchant_order_ref>
    <t3m_id>333333333</t3m_id>
    <score>0</score>
    <recommendation>1</recommendation>
    <message_digest></message_digest>
</RealTimeResponse>
```

2.3.11.2 Key Value Pair Response

```
merchant_identifier=30000&merchant_order_ref=12345&t3m_id=333333333&score=0&recommendation=1&message_digest=
```

2.3.11.3 Considerations

Customer systems must acknowledge the callback response within 1 second. This will then enable Datacash Fraud Systems to record a successful post. An acknowledge mean can take the form of simply passing back the word "ok" or "successful"

If the initial post does fail then the system will re-try 10 times with a 2 minute pause in-between. If the system still doesn't record a successful post after the 10th try, a report will be generated to the support team to try a manual re-send at a later date.

If you wish to use HTTPS you must have a signed certificate to accept SSL authentication.

Additional fields can be added to the existing response – please request these before starting the implementation process or as soon into the process as possible.

2.3.11.4 Additional Information

A full list of RSG response codes can be found in the appendix.

[Back to Contents](#)

2.4 BACS SERVICE

2.4.1 Direct Debit

Additional non-technical information about this Service is available on the website:
<http://www.datacash.com/gettingproducts.php?id=Direct-Credit-Direct-Credit-Overview>

2.4.1.1 Schema Elements for Request

In this section the required fields for each stage in the Direct Debit process will be presented, along with example XML for those fields. The XML is presented in italics for those fields that are not required for all situations. In the XML examples where a field has been **highlighted** in red, this indicates a situation in which other values can be presented in its place – for example `setup` in place of `presetup`.

Optional fields for each transaction type are indicated with an *O*, required fields with an *R*.

Please refer to the [website](#) for definitions of the transaction types and examples of when you may wish to implement them.

The Initial Setup –Setup and Presetup

The `setup` and `presetup` methods both require the same fields to be populated. This information is passed in two distinct places in the schema:

- Request
 - Authentication – section 2.1.1.2
 - Transaction
 - `DirectDebitTxn` – contains all the information about the customer and the method, section 2.4.1.1.1
 - `TxnDetails` - section 2.4.1.1.2

Updating Setups – Confirm and Revoke

If you are using the two stage model, you will need to confirm the `presetup` in order to activate it. Once all payments on a DDI have been completed, the DDI can be cancelled by submitting a `revoke` request.

Both of these transaction types update the existing DDI and the information is passed in two distinct schema paths:

- Request
 - Authentication – section 2.1.1.2
 - Transaction
 - `HistoricTxn` – the `datacash_reference` of the original setup or `presetup` plus the method, section 2.4.1.1.3
 - `TxnDetails` - contains the reference number, section 2.4.1.1.2

Taking Payments

Once a DDI is active, a drawdown can then be processed against it. A drawdown may also be cancelled, if required, using a cancel.

- Request
 - Authentication – section 2.1.1.2
 - Transaction
 - HistoricTxn – the datacash_reference of the original transaction, plus the method and for drawdowns (optionally) the due date and transaction code, section 2.4.1.1.3
 - TxnDetails - for drawdowns only, contains the reference number and the amount, section 2.4.1.1.2

2.4.1.1.1 DirectDebitTxn

The DirectDebitTxn element contains the details of the customer and their bank account. It also contains the method which allows you to choose the one stage setup or two stage presetup models. Each element is required for both models.

Element Name:	DirectDebitTxn
Position(s)	Request.Transaction

Elements of DirectDebitTxn			
Element Name	Description	Values / Limitations	
sortcode	The sort code of the customer's bank	Six digits Can contain additional hyphens and spaces	R
accountnumber	The customer's account number	Eight digits Can contain additional hyphens and spaces	R
accountname	The name of the account holder	A maximum of 18 alphanumerics	R
method	determines the processing model to be used	setup - for one stage processing presetup - for two stage processing	R

In addition to these elements, there are also two attributes that may be populated - these are both optional:

Optional Attributes of DirectDebitTxn			
Attribute Name	Description	Values / Limitations	
active	Enables an existing setup to be transferred to the DataCash system	true	O

type	Enables a non_AUDDIS DDI to be converted into an AUDDIS DDI	conversion	0
------	---	------------	---

Example XML for DirectDebitTxn elements

```

<DirectDebitTxn type="conversion">
  <sortcode>123456</sortcode>
  <accountnumber>12345678</accountnumber>
  <accountname>Mr A. N. Other</accountname>
  <method>setup</method>
</DirectDebitTxn>

<DirectDebitTxn>
  <sortcode>123457</sortcode>
  <accountnumber>12345678</accountnumber>
  <accountname>Mr A. N. Other</accountname>
  <method>presetup</method>
</DirectDebitTxn>

```

2.4.1.1.2 TxnDetails

Element Name:	TxnDetails
Position(s)	Request.Transaction

Elements of TxnDetails							
Element Name	Description	Values / Limitations	Confirm	Draw down	Revoke	Setup & Preset up	Cancel
merchantreference	The reference number of the DDI mandate	Must be of the format agreed with your Sponsoring Bank. Character limitations apply	R	R	R	R	-
amount	The drawdown value		-	R	-	-	-

Example XML Request for TxnDetails elements

```

<TxnDetails>
  <merchantreference>123402</merchantreference>
  <amount>30.00</amount>
</TxnDetails>

```

2.4.1.1.3 HistoricTxn

Element Name:	HistoricTxn
Position(s)	Request.Transaction

Elements of HistoricTxn						
Element Name	Description	Values / Limitations	Confirm	Revoke	Drawdown	Cancel
method	Allows the type of HistoricTxn to be set	confirm revoke drawdown	R	R	R	R
reference	The datacash_reference of the original transaction		R	R	R	R
duedate	Allows a future date for the drawdown to be set if required. If this field is not presented, the next available date will be used	yyyyymmdd format Must be at least 3 working days in the future	-	-	O	-

The DataCash service automatically calculates the BACS Transaction Code for all drawdown transactions. If the drawdown is the first made against a new DDI setup, the transaction code is set to 01, otherwise the transaction code is set to 17.

You may need to set the transaction code yourself however, depending upon whether the drawdown is a representation of a previously rejected drawdown, or if this is the last drawdown to be made against a DDI setup.

Optional Attribute of duedate		
Attribute Name	Description	Values / Limitations
tran_code	The Transaction Code for the drawdown	01 - the first drawdown 17 18 - a drawdown that is being re-presented 19 - the final drawdown. The drawdown will be processed and then the mandate will be cancelled

Example XML Request for HistoricTxn elements

```

<HistoricTxn>
    <method>confirm</method>
    <reference>12345679</reference>
</HistoricTxn>

<HistoricTxn>
    <method>drawdown</method>
    <reference>12345678</reference>
</HistoricTxn>

<HistoricTxn>
    <method>drawdown</method>
    <reference>12345678</reference>
    <duedate tran_code="18">20070101</duedate>
</HistoricTxn>

<HistoricTxn>
    <method>drawdown</method>
    <reference>12345678</reference>
    <duedate>20070131</duedate>
</HistoricTxn>

<HistoricTxn>
    <method>cancel</method>
    <reference>42912928</reference>
</HistoricTxn>

```

2.4.1.2 XML Example Requests

2.4.1.2.1 The Initial DataCash Setup

Example XML Request for presetup

```

<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <password>*****</password>
        <client>99000000</client>
    </Authentication>
    <Transaction>
        <DirectDebitTxn>
            <sortcode>123456</sortcode>
            <accountnumber>12345678</accountnumber>
            <accountname>Mr A. N. Other</accountname>
            <method>presetup</method>
        </DirectDebitTxn>
        <TxnDetails>
            <merchantreference>123402</merchantreference>
        </TxnDetails>
    </Transaction>
</Request>

```

Example XML Request for setup transferring an existing mandate to DataCash

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000000</client>
  </Authentication>
  <Transaction>
    <DirectDebitTxn active="true">
      <sortcode>123456</sortcode>
      <accountnumber>12345678</accountnumber>
      <accountname>Mr A. N. Other</accountname>
      <method>setup</method>
    </DirectDebitTxn>
    <TxnDetails>
      <merchantreference>123402</merchantreference>
    </TxnDetails>
  </Transaction>
</Request>
```

Example XML Request for a setup

This will convert an existing non-AUDDIS mandate to AUDDIS

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000000</client>
  </Authentication>
  <Transaction>
    <DirectDebitTxn type="conversion">
      <sortcode>123456</sortcode>
      <accountnumber>12345678</accountnumber>
      <accountname>Mr A. N. Other</accountname>
      <method>setup</method>
    </DirectDebitTxn>
    <TxnDetails>
      <merchantreference>123402</merchantreference>
    </TxnDetails>
  </Transaction>
</Request>
```

2.4.1.2.2 Updating Setups

Example XML Request for confirm

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000000</client>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>confirm</method>
      <reference>12345679</reference>
    </HistoricTxn>
    <TxnDetails>
      <merchantreference>123401</merchantreference>
    </TxnDetails>
  </Transaction>
</Request>
```

If the method was changed, the same XML could be used for a [revoke](#).

2.4.1.2.3 Taking Payments

Example XML Request for a drawdown

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>drawdown</method>
      <reference>12345678</reference>>
    </HistoricTxn>
    <TxnDetails>
      <merchantreference>123401</merchantreference>
      <amount>10.00</amount>
    </TxnDetails>
  </Transaction>
</Request>
```

Example XML Request for a drawdown with a duedate

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <password>*****</password>
        <client>99000001</client>
    </Authentication>
    <Transaction>
        <HistoricTxn>
            <method>drawdown</method>
            <reference>12345678</reference>
            <duedate>20070922</duedate>
        </HistoricTxn>
        <TxnDetails>
            <merchantreference>123401</merchantreference>
            <amount>10.00</amount>
        </TxnDetails>
    </Transaction>
</Request>
```

Example XML Request for a drawdown with a tran_code

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <password>*****</password>
        <client>99000001</client>
    </Authentication>
    <Transaction>
        <HistoricTxn>
            <method>drawdown</method>
            <reference>12345678</reference>
            <duedate tran_code="18">20070815</duedate>
        </HistoricTxn>
        <TxnDetails>
            <merchantreference>123401</merchantreference>
            <amount>10.00</amount>
        </TxnDetails>
    </Transaction>
</Request>
```

2.4.1.2.4 Cancelling Payments

Example XML Request for a drawdown cancellation

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <password>*****</password>
        <client>99000001</client>
    </Authentication>
    <Transaction>
        <HistoricTxn>
            <method>cancel</method>
            <reference>123412345</reference>
        </HistoricTxn>
    </Transaction>
</Request>
```

2.4.1.3 XML Example Responses

In addition to the standard fields, each Response will contain a `DirectDebitTxn` element.

Element Name:	<code>DirectDebitTxn</code>
Position(s)	Response

Not all elements will be returned for all transaction types, for successful Responses these are outlined below.

Failures will generally contain less of these elements - additional information about the reason for failure will be returned in the `reason` and `information` elements. Examples of failures are available in the [Support Centre](#).

Elements of DirectDebitTxn Response								
Element Name	Description	Values / Limitations	Setup	Presetup	Confirm	Revoke	Drawdown	Cancel
stage	The current stage of the mandate	See Support Centre	yes	yes	yes	yes	n/a	n/a
method	The method sent through with the Request		yes	yes	yes	yes	yes	yes
startdate	Indicates the first date that a drawdown can be submitted	Will be five working days hence	yes	n/a	yes	n/a	n/a	n/a
duedate	The date the drawdown will be taken from the account		n/a	n/a	n/a	n/a	yes	n/a

Once a successful Response for a `setup` or `presetup` has been received, the `datacash_reference` must be stored, as this must be presented with any future transactions referring to that mandate.

2.4.1.3.1 The Initial DataCash Setup

Example XML Response for a successful setup

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <datacash_reference>12345672</datacash_reference>
  <DirectDebitTxn>
    <method>setup</method>
    <stage>active DDI</stage>
    <startdate>20020313</startdate>
  </DirectDebitTxn>
  <merchantreference>123402</merchantreference>
  <mode>TEST</mode>
  <reason>ACKNOWLEDGEMENT</reason>
  <status>1</status>
  <time>...</time>
</Response>
```

Example XML Response for a successful presetup

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <datacash_reference>12345672</datacash_reference>
  <DirectDebitTxn>
    <method>presetup</method>
    <stage>confirm required</stage>
  </DirectDebitTxn>
  <merchantreference>123402</merchantreference>
  <mode>ACCREDITATION</mode>
  <reason>ACKNOWLEDGEMENT</reason>
  <status>1</status>
  <time>...</time>
</Response>
```

Example XML Response for a successful setup of an existing DDI

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <datacash_reference>12345672</datacash_reference>
  <DirectDebitTxn>
    <method>setup</method>
    <stage>existing</stage>
    <startdate>20020306</startdate>
  </DirectDebitTxn>
  <merchantreference>123402</merchantreference>
  <mode>LIVE</mode>
  <reason>ACKNOWLEDGEMENT</reason>
  <status>1</status>
  <time>...</time>
</Response>
```

2.4.1.3.2 Updating Setups

Example XML Response for a successful confirm

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <DirectDebitTxn>
    <method>confirm</method>
    <stage>active DDI</stage>
    <startdate>20020315</startdate>
  </DirectDebitTxn>
  <merchantreference>123401</merchantreference>
  <reason>ACKNOWLEDGEMENT</reason>
  <status>1</status>
</Response>
```

Example XML Response for a successful revoke

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <DirectDebitTxn>
    <method>revoke</method>
    <stage>revoked</stage>
  </DirectDebitTxn>
  <merchantreference>123402</merchantreference>
  <reason>ACKNOWLEDGEMENT</reason>
  <status>1</status>
</Response>
```

2.4.1.3.3 Taking Payments

Example XML Response for a successful drawdown

```
<Response>
  <datacash_reference>12345672</datacash_reference>
  <DirectDebitTxn>
    <duedate tran_code="01">20070130</duedate>
    <method>drawdown</method>
  </DirectDebitTxn>
  <merchantreference>123402</merchantreference>
  <mode>TEST</mode>
  <reason>OK</reason>
  <status>1</status>
  <time>...</time>
</Response>
```

Example XML Response for a successful drawdown

```
<Response>
  <datacash_reference>12345672</datacash_reference>
  <DirectDebitTxn>
    <duedate tran_code="19">20060924</duedate>
    <method>drawdown</method>
  </DirectDebitTxn>
  <merchantreference>123402</merchantreference>
  <mode>TEST</mode>
  <reason>OK</reason>
  <status>1</status>
  <time>...</time>
</Response>
```

2.4.1.3.4 Cancelling Payments

Example XML Response for a successful drawdown cancellation

```
<Response>
  <datacash_reference>12345671</datacash_reference>
  <DirectDebitTxn>
    <method>cancel</method>
  </DirectDebitTxn>
  <merchantreference>12345671</merchantreference>
  <mode>LIVE</mode>
  <reason>CANCELLED OK</reason>
  <status>1</status>
  <time>...</time>
</Response>
```

2.4.1.4 Notifications

This section is only applicable if you have completed the relevant paperwork at your Sponsoring Bank, enabling DataCash to collect notifications of any rejections, cancellations and changes of your setups. These changes will be automatically reflected in the DataCash System.

To allow your system to also be kept up to date and prevent drawdown submission on cancelled mandates, these will be mailed to you in a standard format attachment.

The following information will be included in the attachment:

- merchant_ref – your setup reference
- uniq_ref – the datacash_reference of the mandate
- reason_code – the ADDACS / AUDDIS reason code
- reason_code_text – further information about the reason_code
- effective_date – the date on which the notification takes affect

Full details about the reason codes are available in the Originators Guide and Rules to the Direct Debit Scheme which is available from your bank. These are also detailed in the [Support Centre](#).

Email notifying of nine modifications to Direct Debits

Direct Debit Advices: BACS ID 556622

Please find attached your CSV file containing Direct Debit advices.

Total updated: 9
 Total cancelled: 6
 Total changed: 2
 Total reinstated: 1

Example attachment with six cancellations, two changes and one reinstatement - the changes are reason_codes C and E

```
merchant_ref,uniq_ref,reason_code,reason_code_text,effective_date
"ABC001","99990001","1","Instruction Cancelled by Payer","2010-01-30"
"ABCDE1","99990000","0","Instruction Cancelled - Refer to
Payer","2010-01-30"
"ABC053","99992222","2","Payer deceased","2010-02-01"
"ABC296","89898989","1","Instruction Cancelled by Payer","2010-02-03"
"ABC00A","99995555","1","Instruction Cancelled by Payer","2010-01-31"
"ABC009","99991111","C","Account transferred to a different
account/branch of Bank/Building Society","2010-01-30"
"ABCDE9","99993333","G","Bank will not accept Direct Debit on
Account","2010-02-22"
"ABC921","99966633","E","Instruction Amended","2010-01-31"
"ABC828","99779977","R","Instruction Reinstated","2010-01-30"
```

2.4.1.5 Returned Drawdown Notifications

This section is only applicable if you have completed the relevant paperwork at your Sponsoring Bank, enabling DataCash to collect ARUDD notifications (Automated Return of Unpaid Direct Debits) on your behalf.

To allow your system to also be kept up to date, these will be mailed to you in a standard format attachment.

The following information will be included in the attachment:

- Reference – merchant's reference
- datacash_ref – full 16-digit datacash_reference (please refer to 2.1.1.3)
- uniq_ref – the DataCash OTR (please refer to the [Support Centre](#))
- return_code – ARUDD reason code
- return_code_text – further information about the return_code
- original_processing_date – original BACS processing date
- due_date – The date the drawdown was due to be taken from the account
- amount – the drawdown value
- currency – the drawdown currency

Full details about the reason codes are available in the Originators Guide and Rules to the Direct Debit Scheme which is available from your bank. These are also detailed in the [Support Centre](#).

Email notifying of nine returned debit items

Direct Debit ARUDDS Reports: BACS ID 766000

Please find attached your CSV file containing a list of returned debit items.

Total Returned Debit Items: 9

Please contact our Technical Support desk if you have any queries relating to this advice.

Example attachment

```
Reference,datacash_ref,uniq_ref,return_code,return_code_tex,original_processing_date,due_date,amount,currency
"ABCDEF368","3200100963845000","963845000","1","Instruction Cancelled","06/25/2010","06/28/2010","238.19","GBP"
"ABCDEF956","3500100963845001","963845001","0","Refer to Payer","06/25/2010","06/28/2010","238.19","GBP"
"ABCDEF905","3300100963845002","963845002","0","Refer to Payer","06/25/2010","06/28/2010","87.76","GBP"
"ABDFR3749","3500100973919000","973919000","0","Refer to Payer","07/02/2010","07/09/2010","84.01","GBP"
"ABCR93037","3800100969104000","969104000","0","Refer to Payer","07/08/2010","07/05/2010","274.26","GBP"
"ABCD38303","3900100969104001","969104001","B","Account Closed","07/08/2010","07/05/2010","137","GBP"
"3802JFDEW","3300100974762000","974762000","1","Instruction Cancelled","06/29/2010","07/12/2010","74.07","GBP"
"84ABC3935","3800100974762001","974762001","0","Refer to Payer","06/29/2010","07/12/2010","80.63","GBP"
"23445ABC2","3200100980386000","980386000","1","Instruction Cancelled","07/12/2010","07/13/2010","82","GBP"
```

2.4.1.6 Direct Debit Continuous Authority

This Service allows drawdowns to be automatically created against a DDI.

A technical introduction to this Service is available on the website:

<http://www.datacash.com/gettingproducts.php?id=Direct-Debit>

2.4.1.7 Schema Elements for Request

In this section the required fields for Direct Debit refunds will be presented, along with example XML for those fields. The XML is presented in italics for those fields that are not required for all situations. In the XML examples where a field has been highlighted in blue, this indicates a situation in which other values can be presented in its place – for example `weekly` in place of `monthly`.

Optional fields for each transaction type are indicated with an O, required fields with an R.

Setting up a Direct Debit Continuous Authority Account

The information required to set up a Direct Debit Continuous Authority Account is passed in these distinct places in the schema:

- Request
 - Authentication – section 2.1.1.2
 - Transaction
 - ContAuthTxn – details of the account to be set up and its regular payments, section 2.4.1.7.1
 - FirstPayment – optional first payment date & amount, section 2.4.1.7.2
 - LastPayment – optional last payment date & amount, section 2.4.1.7.3
 - TxnDetails – your reference number for the account, section 2.4.1.7.4
 - HistoricTxn – details of the DDI to create an account for, section 2.4.1.1.3

Cancelling a Continuous Authority Account

The Direct Debit Continuous Authority Account can also be cancelled by using the cancel transaction type. This transaction type requires information passed in one distinct place in the schema:

- Request
 - Authentication – [section 2.1.1.2](#)
 - Transaction
 - HistoricTxn – the `datacash_reference` of the account, plus the method, [section 2.4.1.1.3](#)

2.4.1.7.1 ContAuthTxn

The elements within this parent may only be presented for drawdown transactions. They are excluded for cancel transactions.

Element Name:	ContAuthTxn
Position(s)	Request.Transaction

Elements of ContAuthTxn		
Element Name	Description	Values / Limitations
start_date	The date on which the first regular payment will be taken. Subsequent payments will be based on this date.	Must be more than three days after the transaction is sent. Date in the format dd/mm/yyyy R
number_of_payments	Number of regular payments to be collected.	Payments will be taken until the Continuous Authority account is cancelled if a value is not specified O
FirstPayment	Details of any initial payment to be taken. This is in addition to the regular payments. See section 2.4.1.7.2	O
LastPayment	Details of any final payment to be taken. This is in addition to the regular payments. See section 2.4.1.7.3	O
period	The frequency of payments	weekly monthly quarterly annually R
method	The transaction method	drawdown R
amount	The value of the drawdown	Must be specified to two decimal places. GBP only R
flagging	This element has two attributes. If the element is supplied, both attributes must be supplied. See table below for more information	O

Attributes of Flagging			
Element Name	Description	Values / Limitations	
first	Indicates whether the first payment made using the Cont Auth account is to be flagged as the first drawdown	yes no	O
last	Indicates whether the last payment made using the Cont Auth account is to be flagged as the last drawdown	yes no	O

For further information about drawdown flagging, please refer to section 0.

Example XML for ContAuthTxn elements

```

<ContAuthTxn>
  <method>drawdown</method>
  <start_date>26/02/2006</start_date>
  <period>weekly</period>
  <amount>35.00</amount>
</ContAuthTxn>

<ContAuthTxn>
  <method>drawdown</method>
  <start_date>01/04/2005</start_date>
  <period>annually</period>
  <amount>100.00</amount>
  <number_of_payments>3</number_of_payments>
  <flagging first="yes" last="no"/>
  <FirstPayment>...</FirstPayment>
</ContAuthTxn>

```

2.4.1.7.2 FirstPayment

The elements within this parent may only be presented for drawdown transactions, in which case both child elements must be completed.

This element is excluded for cancel transactions.

Element Name: FirstPayment
Position(s) Request.Transaction.ContAuthTxn

Elements of FirstPayment		
Element Name	Description	Values / Limitations
date	The date the first payment is to be taken	Must be in dd/mm/yyyy format. Must be at least three days from the current date. It must also be at least one day ahead of the start_date
amount	The value of the first payment	If a currency is supplied, it must be GBP

Example XML for FirstPayment element

```
<FirstPayment>
  <amount>50.00</amount>
  <date>25/02/2006</date>
</FirstPayment>
```

2.4.1.7.3 LastPayment

The elements within this parent may only be presented for drawdown transactions, in which case both child elements must be completed.

This element is excluded for cancel transactions.

The LastPayment element can only be used if a set number of drawdowns have been specified.

Element Name:	LastPayment
Position(s)	Request.Transaction.ContAuthTxn

Elements of LastPayment		
Element Name	Description	Values / Limitations
date	The date the last payment is to be taken	Must be in dd/mm/yyyy format; must be a minimum of one date after the last regular payment
amount	The value of the last payment	If a currency is supplied, it must be GBP

Example XML for LastPayment

```
<LastPayment>
  <amount>50.00</amount>
  <date>25/03/2008</date>
</LastPayment>
```

2.4.1.7.4 TxnDetails

The elements within this parent may only be presented for drawdown transactions. They are excluded for cancel transactions.

Element Name:	TxnDetails
Position(s)	Request.Transaction

Elements of TxnDetails		
Element Name	Description	Values / Limitations
merchantreference	A unique reference number to identify the individual Direct Debit Continuous Authority account	Between 6 and 26 alphanumeric characters

Example XML for TxnDetails elements

```
<TxnDetails>
    <merchantreference>123402</merchantreference>
</TxnDetails>
```

2.4.1.7.5 HistoricTxn

Element Name:	HistoricTxn
Position(s)	Request.Transaction

Elements of HistoricTxn

Element Name	Description	Values / Limitations	Drawdown	Cancel
method	cancellations only	cancel	-	R
reference	The datacash_reference of the original transaction		R	R

Example XML element for cancel

```
<HistoricTxn>
    <method>cancel</method>
    <reference>3000123412345678</reference>
</HistoricTxn>
```

Example XML element for drawdown

```
<HistoricTxn>
    <reference>12345678</reference>
</HistoricTxn>
```

2.4.1.8 XML Example Requests

2.4.1.8.1 Payments

Example XML Request for a five weekly payments of 10.00

```

<Request>
  <Authentication>
    <client>21859999</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <ContAuthTxn>
      <method>drawdown</method>
      <start_date>10/02/2005</start_date>
      <period>weekly</period>
      <number_of_payments>5</number_of_payments>
      <amount>10.00</amount>
    </ContAuthTxn>
    <TxnDetails>
      <merchantreference>12345678</merchantreference>
    </TxnDetails>
    <HistoricTxn>
      <reference>12345678</reference>
    </HistoricTxn>
  </Transaction>
</Request>

```

Example XML Request for an unknown number of payments to be taken on the 31st of each month

```

<Request>
  <Authentication>
    <client>21859999</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <ContAuthTxn>
      <method>drawdown</method>
      <start_date>31/03/2005</start_date>
      <period>monthly</period>
      <amount>25.00</amount>
    </ContAuthTxn>
    <TxnDetails>
      <merchantreference>12345678</merchantreference>
    </TxnDetails>
    <HistoricTxn>
      <reference>12345678</reference>
    </HistoricTxn>
  </Transaction>
</Request>

```

Example XML Request for eight payments: six regular payments, and separate first and last payments

The first payment is a one off administration fee of £30, followed by 6 monthly payments of £25 and a final payment of £20. The first and last payments are to be flagged with "01" and "19" respectively, which will cancel the underlying DDI after the final payment.

```
<Request>
  <Authentication>
    <client>21859999</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <ContAuthTxn>
      <method>drawdown</method>
      <start_date>31/05/2005</start_date>
      <period>monthly</period>
      <amount>25.00</amount>
      <number_of_payments>6</number_of_payments>
      <FirstPayment>
        <amount>30.00</amount>
        <date>20/04/2005</date>
      </FirstPayment>
      <LastPayment>
        <amount>20.00</amount>
        <date>30/11/2005</date>
      </LastPayment>
      <flagging first="yes" last="yes"/>
    </ContAuthTxn>
    <TxnDetails>
      <merchantreference>12345678</merchantreference>
    </TxnDetails>
    <HistoricTxn>
      <reference>12345678</reference>
    </HistoricTxn>
  </Transaction>
</Request>
```

2.4.1.8.2 Cancellations

Example XML Request for cancel

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>cancel</method>
      <reference>4900200000000001</reference>
    </HistoricTxn>
  </Transaction>
</Request>
```

Merchants using the Credit and Debit Card Services may note that this XML structure is the same as for cancelling a card payment – section 2.2.1.4.2.

2.4.1.9 Schema Elements for Response

There are no Response elements specific to this service. The general Response elements are returned, these are discussed in section 2.1.1.3.

2.4.1.10 XML Example Responses

Example XML Response for a successful account setup

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <datacash_reference>9910670039375669</datacash_reference>
  <information>Direct Debit Continuous Authority account
    set up</information>
  <merchantreference>ABC00005</merchantreference>
  <mode>LIVE</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
</Response>
```

2.4.1.11 Notifications

There are two types of notifications, one for batched drawdowns and one for account cancellations.

2.4.1.11.1 Batched Drawdowns

Each day a payment is taken from an account, you will receive an email summarising the payments taken that day, in a standard CSV format attachment.

The following information will be included in the attachment:

- Account Reference – the last 11 digits of the `datacash_reference` number of the account
- Merchant Reference – the DDI mandate number (`merchantreference`) against which the drawdown will be processed
- DataCash Reference – the `datacash_reference` of the drawdown

Email notifying of batched drawdowns

Drawdown Results for Mon 17-Oct-2010

Attachments: batched.csv

This email contains details of drawdowns which have been processed on Live production site by DataCash on your behalf, for 2185**** Mon 17-Oct-2010.

If you have any questions please contact our support team.

Example attachment with two drawdowns

Account Reference, Merchant Reference, DataCash Reference
 41740256, 1823362AHC, 4200200040467707
 41756562, 1825472RHP, 4400200040373171

2.4.1.11.2 Account Cancellations

If a DDI is cancelled or revoked, the account will be automatically cancelled.

To allow your system to also be kept up to date, details of these will be mailed to you in a standard format CSV attachment.

The following information will be included in the attachment:

- Account reference – the last 11 digits of the `datacash_reference` number of the account
- Merchant Reference – the `merchantreference` of the account
- Cancel DataCash Reference – the `datacash_reference` number of the transaction which cancelled the account. If the account was automatically cancelled when a DDI `revoke` is submitted, this field will be the same as `DDI`

Setup Reference

- DDI Setup Reference – the datacash_reference number of the DDI setup
- DDI Merchant Reference – your mandate number
- DDI Stage – the stage of the DDI

Email notifying of account cancellations

Account Cancellations for Mon 17-Oct-2010

Attachments: cancelled.csv

This email contains details of Direct Debit Recurring Transaction accounts which have been cancelled on Live production site by DataCash on your behalf, for 2185*** Mon 17-Oct-2010, where the associated DDI setup is no longer active.

Example attachment with four cancellation notifications

Account Reference,Merchant Reference,Cancel DataCash Reference,DDI Setup Reference,DDI Merchant Reference,DDI Stage
46453486,EFG987456,4600200040488736,4600200040488741,ABC1234562,revoked
47453828,EFG453454,4200200040493495,4000200040493496,ABC483,cancelled
at source
47455267,EFG456454,4300200040493522,4400200040493526,SJI451,cancelled
at source
47511296,EFG15648,4100200041248347,4800200041260637,HUR45423,revoked

2.4.2 Direct Credit – Direct Debit Refunds

This Service allows funds to be returned to any DDI held within the DPG.

A technical introduction to this service is available on the website:

<http://www.datacash.com/gettingproducts.php?id=Direct-Debit-Refunds>

2.4.2.1 Schema Elements for Request

In this section the required fields for Direct Debit refunds will be presented, along with example XML for those fields.

Refunding Direct Debits

The information required to process a direct debit refund - ddrefund - is passed in these distinct places in the schema:

2.4.2.1.1 Request

- o Authentication – [section 2.1.1.2](#)
- o Transaction
 - TxnDetails – contains the merchantreference number and the amount, [section 2.4.1.7.4](#)
 - HistoricTxn – the datacash_reference of the original mandate and the method, [section 2.4.1.1.3](#).

Cancelling Direct Debit Refunds

The information required to cancel a ddrefund is passed in these distinct places in the schema:

2.4.2.1.2 Request

- o Authentication – [section 2.1.1.2](#)
- o Transaction
 - HistoricTxn – the datacash_reference of the original ddrefund and the method, [section 2.4.1.1.3](#).

2.4.2.1.3 TxnDetails

The elements within this parent are required for ddrefund transactions. They are excluded for cancel transactions.

Element Name:	TxnDetails
Position(s)	Request.Transaction

Elements of TxnDetails		
Element Name	Description	Values / Limitations
merchantreference	The reference number of the DDI mandate	Between six and eighteen alphanumeric characters
amount	The value of the refund	

Example XML for TxnDetails elements

```
<TxnDetails>
  <merchantreference>123402</merchantreference>
  <amount>30.00</amount>
</TxnDetails>
```

2.4.2.1.4 HistoricTxn

The elements within this parent are required for both cancel and ddrefund transaction types.

Element Name:	HistoricTxn
Position(s)	Request.Transaction

Elements of HistoricTxn		
Element Name	Description	Values / Limitations
method	Identifies the transaction type as a refund of a direct debit	ddrefund cancel
reference	The datacash_reference of the original transaction	

Example XML for HistoricTxn element for a ddrefund

```
<HistoricTxn>
  <method>ddrefund</method>
  <reference>12345678</reference>
</HistoricTxn>
```

Example XML for HistoricTxn element for a cancel

```
<HistoricTxn>
  <method>cancel</method>
  <reference>4000000012345678</reference>
</HistoricTxn>
```

2.4.2.2 XML Example Requests

Example XML Request for a ddrefund

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>ddrefund</method>
      <reference>12345678</reference>
    </HistoricTxn>
    <TxnDetails>
      <merchantreference>ABCDEF456</merchantreference>
      <amount>10.00</amount>
    </TxnDetails>
  </Transaction>
</Request>
```

Example XML Request for a cancel

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>cancel</method>
      <reference>4000000059832490</reference>
    </HistoricTxn>
  </Transaction>
</Request>
```

2.4.2.3 Schema Elements for Response

There are no additional XML elements returned for specifically for this Service. The general XML elements in the Response are covered in section 2.1.1.3.

2.4.2.4 XML Example Responses

2.4.2.4.1 Refunds

Example XML Response for a successful ddrefund

```
<Response>
  <datacash_reference>4800200040644359</datacash_reference>
  <merchantreference>ABCDEF456</merchantreference>
  <mode>LIVE</mode>
  <reason>Accepted</reason>
  <status>1</status>
  <time>...</time>
</Response>
```

Example XML Response for an unsuccessful ddrefund

```
<Response>
  <datacash_reference>4000000059832491</datacash_reference>
  <information>Unable to locate matching DDI
    transaction</information>
  <merchantreference>123411</merchantreference>
  <mode>TEST</mode>
  <reason>Unable to locate matching DDI transaction</reason>
  <status>137</status>
  <time>...</time>
</Response>
```

2.4.2.4.2 Cancellations

Example XML Response for a successfully cancelled Direct Credit

```
<Response>
  <datacash_reference>39375274</datacash_reference>
  <merchantreference>4800200040644359</merchantreference>
  <reason>CANCELLED OK</reason>
  <status>1</status>
  <time>...</time>
</Response>
```

Example XML Response for an unsuccessful Direct Credit cancellation

```
<Response>
  <information>Direct Credit has already been
    cancelled</information>
  <merchantreference>4800200040644359</merchantreference>
  <mode>TEST</mode>
  <reason>Already Cancelled</reason>
  <status>128</status>
  <time>...</time>
</Response>
```

2.4.3 *Direct Credit – Standard*

This Service allows funds to be returned to a customer by placing them directly into their bank account.

A technical introduction to this service is available on the website:
<http://www.datacash.com/gettingproducts.php?id=Direct-Credit>

2.4.3.1 Schema Elements for Request

In this section the required fields for Standard Direct Credit will be presented, along with example XML for those fields. The XML is presented in italics for those fields that are not required for all situations.

Optional fields for each transaction type are indicated with an O, required fields with an R.

Performing Direct Credits

The information required to process a direct credit - `directcredit` - is passed in these distinct places in the schema:

2.4.3.1.1 Request

- o Authentication – section 2.1.1.2
- o Transaction
 - TxnDetails – contains the merchantreference number and the amount, section 2.4.1.7.4
 - DirectCreditTxn - the details of the customers bank account, plus the method, section 2.4.1.1.1

Cancelling Direct Debit Refunds

The information required to cancel a directcredit is passed in these distinct places in the schema:

2.4.3.1.2 Request

- o Authentication – section 2.1.1.2
- o Transaction
 - HistoricTxn - the datacash_reference of the original directcredit and the method, section 2.4.1.1.3

2.4.3.1.3 DirectCreditTxn

The elements within this parent may only be presented for directcredit transactions. They are excluded for cancel transactions.

Element Name: DirectCreditTxn
Position(s) Request.Transaction

Elements of DirectCreditTxn			
Element Name	Description	Values / Limitations	Required?
method	Identifies the transaction type as a standard Direct Credit	directcredit	R
sortcode	The sort code of the account to be credited	Valid UK sortcode	R
accountnumber	The account number to be credited	Valid UK account number	R
accountname	The name of the account holder(s)	Up to 18 characters, alpha numerics and spaces only	R
bankreference		Up to 18 characters, alpha numerics and spaces only	O

Example XML Element DirectCredit

```

<DirectCreditTxn>
  <sortcode>000001</sortcode>
  <accountnumber>12345609</accountnumber>
  <accountname>Julia McDonald</accountname>
  <method>directcredit</method>
</DirectCreditTxn>

<DirectCreditTxn>
  <sortcode>000222</sortcode>
  <accountnumber>12340000</accountnumber>
  <accountname>Michael Johnson</accountname>
  <bankreference>return of goods</bankreference>
  <method>directcredit</method>
</DirectCreditTxn>

```

2.4.3.1.4 TxnDetails

The elements within this parent are required for directcredit transactions, but excluded for cancel transactions.

Element Name:	TxnDetails
Position(s)	Request.Transaction

Elements of TxnDetails		
Element Name	Description	Values / Limitations
merchantreference	A unique reference number for each transaction	Up to 30 characters
amount	The value of the refund	

Example XML for TxnDetails elements

```

<TxnDetails>
  <merchantreference>123402</merchantreference>
  <amount>30.00</amount>
</TxnDetails>

```

2.4.3.1.5 HistoricTxn

The elements within this parent must be presented for `cancel` transactions. They are excluded for `directcredit` transactions.

Element Name: HistoricTxn
Position(s) Request.Transaction.HistoricTxn

Elements of HistoricTxn		
Element Name	Description	Values / Limitations
method	Identifies the transaction type as a cancellation	cancel
reference	The <code>datacash_reference</code> of the original transaction	16 digits

Example XML Element HistoricTxn for a cancel

```
<HistoricTxn>
  <method>cancel</method>
  <reference>9999999912345678</reference>
</HistoricTxn>
```

2.4.3.2 XML Example Requests

Example XML Request

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>123402</merchantreference>
      <amount>1000.01</amount>
    </TxnDetails>
    <DirectCreditTxn>
      <method>directcredit</method>
      <sortcode>123456</sortcode>
      <accountnumber>12345678</accountnumber>
      <accountname>Greg Kane</accountname>
    </DirectCreditTxn>
  </Transaction>
</Request>
```

Example XML Request with optional bankreference

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>ABDYC003</merchantreference>
      <amount>49.68</amount>
    </TxnDetails>
    <DirectCreditTxn>
      <method>directcredit</method>
      <sortcode>123456</sortcode>
      <accountnumber>12340000</accountnumber>
      <accountname>Jane Smith</accountname>
      <bankreference>Money for Petrol</bankreference>
    </DirectCreditTxn>
  </Transaction>
</Request>
```

Example XML Request for a cancel

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>cancel</method>
      <reference>4000000059832490</reference>
    </HistoricTxn>
  </Transaction>
</Request>
```

2.4.3.3 Schema Elements for Response

The XML Responses for this Service only contain those elements that are discussed in section 2.1.1.3. There are no additional XML elements returned specifically for this Service

2.4.3.4 XML Example Responses

Example XML Response for a successful directcredit

```
<Response>
  <datacash_reference>4803000012345672</datacash_reference>
  <merchantreference>ABDYC003</merchantreference>
  <mode>TEST</mode>
  <reason>Accepted</reason>
  <status>1</status>
  <time>...</time>
</Response>
```

Example XML Response for an unsuccessful directcredit

```
<Response>
  <information>Invalid format for Account Name</information>
  <merchantreference>384086694791667</merchantreference>
  <mode>TEST</mode>
  <reason>Invalid format for Account Name</reason>
  <status>134</status>
  <time>...</time>
</Response>
```

Examples of directcredit cancellations are available in section 2.4.2.4.2.

2.4.4 Direct Credit – Card Collection Accounts

This Service allows funds to be returned to a cardholder by using their card collection account.

A technical introduction to this service is available on the website:

<http://www.datacash.com/gettingproducts.php?id=Payments-to-Card-Collection-Accounts>

2.4.4.1 Schema Elements for Request

In this section the required fields for Direct Credit to Card Collection Accounts will be presented, along with example XML for those fields. The XML is presented in italics for those fields that are not required for all situations.

Optional fields for each transaction type are indicated with an *O*, required fields with an R.

Performing Card Account Payments

The information required to process a direct debit refund - cardaccountpayment - is passed in these distinct places in the schema:

2.4.4.1.1 Request

- o Authentication – section 2.1.1.2
- o Transaction
 - TxnDetails – contains the merchantreference number and the amount, section 2.4.4.1.4
 - DirectCreditTxn – details of the customers card, plus the method, section 2.4.1.1.1.

Cancelling Direct Debit Refunds

The information required to cancel a cardaccountpayment is passed in these distinct places in the schema:

2.4.4.1.2 Request

- o Authentication – section 2.1.1.2
- o Transaction
 - HistoricTxn – the datacash_reference of the cardaccountpayment and the method, section 2.4.4.1.5.

2.4.4.1.3 DirectCreditTxn

The elements within this parent may only be presented for cardaccountpayment transactions. They are excluded for cancel transactions.

Element Name:	DirectCreditTxn
Position(s)	Request.Transaction

Elements of DirectCreditTxn			
Element Name	Description	Values / Limitations	Required?
method	Identifies the transaction type as a refund to a card collection account	cardaccountpayment	R
pan	The card number	Card must be Visa or MasterCard branded and issued in the UK	R
expirydate	The card expiry date	MM/YY format	O
cardname	The card holder's name	Up to 18 characters. "COLLECTION ACCOUNT" will be automatically used if this element is not presented	O

2.4.4.1.4 TxnDetails

Element Name:	TxnDetails
Position(s)	Request.Transaction

Elements of TxnDetails		
Element Name	Description	Values / Limitations
merchantreference	A unique reference number for each transaction	Up to 30 characters
amount	The value of the refund	

Example XML for TxnDetails elements

```

<TxnDetails>
    <merchantreference>123402</merchantreference>
    <amount>30.00</amount>
</TxnDetails>

```

2.4.4.1.5 HistoricTxn

Element Name:	HistoricTxn
Position(s)	Request.Transaction.HistoricTxn

Elements of HistoricTxn		
Element Name	Description	Values / Limitations
method	Identifies the transaction type as a cancellation	cancel
reference	The datacash_reference of the original transaction	16 digits

Example XML Element HistoricTxn for a cancel

```
<HistoricTxn>
  <method>cancel</method>
  <reference>3009999912345678</reference>
</HistoricTxn>
```

2.4.4.2 XML Example Requests

Example XML Request for cardaccountpayment

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>

      <merchantreference>10397897589</merchantreference>
        <amount>59.00</amount>
      </TxnDetails>
      <DirectCreditTxn>
        <method>cardaccountpayment</method>
        <pan>5473*****0007</pan>
      </DirectCreditTxn>
    </Transaction>
  </Request>
```

Example XML Request for cardaccountpayment

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <TxnDetails>

      <merchantreference>B00123895</merchantreference>
        <amount>23.50</amount>
      </TxnDetails>
      <DirectCreditTxn>
        <method>cardaccountpayment</method>
        <pan>4444*****0001</pan>
        <cardname>Wes Scantlin</cardname>
        <expirydate>03/09</expirydate>
      </DirectCreditTxn>
    </Transaction>
  </Request>
```

Example XML Request for a cancel

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>cancel</method>
      <reference>4000000059832490</reference>
    </HistoricTxn>
  </Transaction>
</Request>
```

2.4.4.3 Schema Elements for Response

The XML Responses for this Service only contain those elements that are discussed in section 2.1.1.3. There are no additional XML elements returned specifically for this Service.

2.4.4.4 XML Example Responses

Example XML Response for a successful cardaccountpayment

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000200045345250</datacash_reference>
    <merchantreference>390726429629630</merchantreference>
    <mode>...</mode>
    <reason>Accepted</reason>
    <status>1</status>
    <time>...</time>
</Response>
```

Example XML Requests for unsuccessful cardaccountpayment

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4900200043479577</datacash_reference>
    <information>The card number does not pass the standard
                Luhn checksum test</information>
    <merchantreference>384084003240741</merchantreference>
    <mode>...</mode>
    <reason>Bad checksum</reason>
    <status>25</status>
    <time>...</time>
</Response>

<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>3600200044366081</datacash_reference>
    <information>No payment details for this card</information>
    <merchantreference>389377230555556</merchantreference>
    <mode>...</mode>
    <reason>Unknown Payment Details</reason>
    <status>126</status>
    <time>...</time>
</Response>
```

Examples of cancellations are available in section 2.4.2.4.2.

[Back to Contents](#)

2.5 BATCH INPUT

The Batch Input Service allows many of the DPG Services to be utilised via batch process, instead of submitting the transactions in real time. Further non-technical information about this service is available on the website:

<http://www.datacash.com/services/batch/overview.shtml>

The transaction data may be presented in either a XML or a CSV file format. This file will be referred to in this document as the *Batch Request XML* (or CSV). This file must then be Zlib-compressed and Base64 encoded.

The compressed and encoded *Batch Request* file is then placed within the *Batch Submission* Request document, which is itself an XML document. The *Batch Submission* Request is then sent to DataCash and a *Batch Submission* Response is returned.

To collect the results of the batch, a *Batch Query* Request is sent to DataCash. A *Batch Query* Response will be returned. This will contain the results of the batch, which have been Zlib –compressed and Base64 encoded. These results will be referred to as the *Batch Response XML* (or CSV). The format of the *Batch Response* document will match that of the *Batch Request*.

Naturally, the data you need to supply will depend upon the particular service you are using. However there are certain aspects of Batch Input that are common to each service that it can be used with. The features common to all services will be covered first

- Available DPG Services – section 2.5.1
- Common Elements - each of these elements are independent of the service used, section 2.5.2
 - File Formats Attributes, section 2.5.2.1
 - *Batch Submission* elements - also independent of format, section 2.5.2.2
 - *Batch Query* elements - also independent of format, section 2.5.2.3
 - *Batch Request* elements - for both XML and CSV, section 2.5.2.5
 - *Batch Response* elements - for both XML and CSV, section 2.5.2.6
- SERVIce Specific Elements – these are specific to both the service and the file format used, section 2.5.3.1

The following key will be used for this section:

- *O* - Optional
- *R* - Required, field must be presented
- *M* - Mandatory if Available, if the information is available, it should be presented

2.5.1 Available DPG Services

Various DPG Services can be accessed via the Batch Input Service using both the XML and CSV file formats. These are outlined in the table below.

Service Type	DPG Service Name	Restrictions	Additional Information
Credit and Debit Cards	Bank Card	-	Section 2.2.1
	Pre-Registered Cards	Account must be configured for this service	Section 2.2.11
	Recurring Transactions – Capture Method	Account must be configured for this service	Section 2.2.12
	Chip and PIN	Account must be configured for this service. Transaction method must be auth or refund.	Section 2.6.1.1
BACS Services	Standard Direct Debit	Account must be configured for this service	Section 2.4.1.1
	Direct Credit: DDrefunds	Account must be configured for this service	Section 2.4.2
	Standard Direct Credit	Account must be configured for this service	Section 2.4.3
	Direct Credit: Card Collection Accounts	Account must be configured for this service	Section 2.4.4

In addition, the following Fraud Screening Services may be used in conjunction with the Credit and Debit Card Services:

Service Type	DPG Service Name	Restrictions	Additional Information
Fraud Prevention for Cards	ReD	Account must be configured for this service. The extended dataset cannot be provided.	Section 2.3.3
	AVSCV2	AVS check only. Not available for Chip and PIN	Section 2.3.1

2.5.2 Overview of Common Elements and Attributes

2.5.2.1 File Formats Attributes

The individual transactions within each file may be presented in either XML or CSV format. Each format has a name, which is passed within both the Batch Submission document and the Batch Request. The file names are presented below.

XML File Formats		
Format	Service	Details in
xml_cardtxn	Credit and Debit Cards	Section 2.5.3.1
xml_chp_cardtxn	Cardholder Present Cards	Section 2.5.3.2
xml_directcredit	BACS – Direct Credit	Section 2.5.3.3
xml_directdebit	BACS - Direct Debit	Section 2.5.3.4

CSV file Formats			
Format	Service	Used for	Details in
csv_avs_card_txn	Credit and Debit Cards	Authorisations with standard AVS checking	Section 2.5.3.6
csv_card_txn	Credit and Debit Cards	Authorisations	Section 2.5.3.5
csv_chp_card_txn	Cardholder Present	auth and refund transactions	Section 2.5.3.7
csv_historic	Credit and Debit Cards	Historic transactions: fulfill, txn_refund, cancel, accept_fraud authorize_referral_request	Section 2.5.3.8
csv_prereg	Credit and Debit Cards	Pre-Registered Cards	Section 2.5.3.9
csv_dc_cc_account	BACS Services	Direct Credit cardaccountpayment	Section 2.5.3.10
csv_dc_ddrefund	BACS Services	Direct Credit ddrefunds	Section 2.5.3.11
csv_dc_std	BACS Services	Standard Direct Credit	Section 2.5.3.12
csv_dd_drawdown	BACS Services	Direct Debit drawdown	Section 2.5.3.13
csv_dd_setup	BACS Services	Direct Debit setup and presetup	Section 2.5.3.14
csv_dd_setup_edit	BACS Services	Direct Debit confirm and revoke	Section 2.5.3.15

2.5.2.2 Batch Submission Elements

These elements are common to all of the Batch Input Services, regardless of the service or format used.

2.5.2.2.1 Schema Elements for Batch Submission

Each batch contains header information:

2.5.2.2.2 Request

- Authentication – each Batch Submission can contain transactions for only one DataCash account (vTID). The Authentication element is covered in section 2.1.1.2
- Transaction
 - BatchInputTxn – section 2.5.2.2.2.1

2.5.2.2.2.1 BatchInputTxn

Element Name:	BatchInputTxn
Position:	Request.Transaction

Elements of BatchInputTxn			
Element Name	Description	Values / Limitations	
batchfile	The Batch Request, compressed using Zlib compression, then Base64 encoded	Uncompressed Batch Request must be less than 1MB	R
txn_count	The total number of transactions in the batchfile	Must be an integer > 0.	R
total_amount	The total gross value of transactions in the batchfile, regardless of currency and transaction type	Tolerant to 0.001. If the format of the transactions within the Batch Request does not contain an amount, the <code>total_amount</code> must be set to zero	R

Attributes of batchfile			
Element Name	Description	Values / Limitations	
format	The format of the file you are submitting	Please refer to section 0	R

XML Examples of Batch Submission for BatchInputTxn complex elements

```
<BatchInputTxn>
    <batchfile format="xml_cardtxn">...</batchfile>
    <txn_count>3</txn_count>
    <total_amount>395.99</total_amount>
</BatchInputTxn>

<BatchInputTxn>
    <batchfile format="csv_historic">...</batchfile>
    <txn_count>16</txn_count>
    <total_amount>0</total_amount>
</BatchInputTxn>

<BatchInputTxn>
    <batchfile format="xml_directdebit">...</batchfile>
    <txn_count>4</txn_count>
    <total_amount>75.60</total_amount>
</BatchInputTxn>

<BatchInputTxn>
    <batchfile format="csv_dd_setup">...</batchfile>
    <txn_count>29</txn_count>
    <total_amount>0</total_amount>
</BatchInputTxn>

<BatchInputTxn>
    <batchfile format="csv_dd_drawdown">...</batchfile>
    <txn_count>63</txn_count>
    <total_amount>1565.09</total_amount>
</BatchInputTxn>
```

2.5.2.2.3 XML Example Batch Submission

Example XML Batch Submission for a Batch Request in XML format containing three transactions

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <BatchInputTxn>
      <batchfile format="xml_cardtxn">...</batchfile>
      <txn_count>3</txn_count>
      <total_amount>395.99</total_amount>
    </BatchInputTxn>
  </Transaction>
</Request>
```

Example XML Batch Submission for a Batch Request in CSV format containing five transactions

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <BatchInputTxn>
      <batchfile format="csv_historic">...</batchfile>
      <txn_count>5</txn_count>
      <total_amount>2501.00</total_amount>
    </BatchInputTxn>
  </Transaction>
</Request>
```

Example XML Batch Submission for a Batch Request in XML format containing five transactions

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <BatchInputTxn>
      <batchfile format="xml_directdebit">...</batchfile>
      <txn_count>5</txn_count>
      <total_amount>395.99</total_amount>
    </BatchInputTxn>
  </Transaction>
</Request>
```

Example XML Batch Submission for a Batch Request in CSV format containing eighteen transactions

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <BatchInputTxn>
      <batchfile format="csv_dd_drawdown">...</batchfile>
      <txn_count>18</txn_count>
      <total_amount>156.00</total_amount>
    </BatchInputTxn>
  </Transaction>
</Request>
```

2.5.2.2.4 Schema Elements for Response

The Response for Batch Submission contains the normal Response fields. These are covered in section 2.1.1.3

The `datacash_reference` returned is the reference number for the Batch Submission. This should be stored to allow a Batch Query (section 2.5.2.3) to be performed. The `merchantreference` returned is extracted from within the Batch Request.

2.5.2.2.5 XML Example Batch Submission Response

Two Batch Submission Responses are shown below.

A full list of return codes for this service is available on the website [here](#).

Example XML Batch Submission Responses for a successful and a rejected batch submission

```
<Response>
    <datacash_reference>3900900100010001</datacash_reference>
    <merchantreference>batch_ABCDEF</merchantreference>
    <mode>LIVE</mode>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <time>...</time>
</Response>

<Response>
    <datacash_reference>3800900100010006</datacash_reference>
    <information>Supplied and actual transaction counts do
        not match</information>
    <merchantreference>batch_XYZ</merchantreference>
    <mode>LIVE</mode>
    <reason>BatchInput: Error in batch data</reason>
    <status>262</status>
    <time>...</time>
</Response>
```

2.5.2.3 Batch Query

The Batch Query allows you to receive the results of each transaction within the Batch Submission.

2.5.2.3.1 Schema Elements for Batch Submission Query

2.5.2.3.2 Request

- o Authentication – section 2.1.1.2
- o Transaction
 - HistoricTxn – see section 2.5.2.3.2.1.

2.5.2.3.2.1 HistoricTxn

Element Name: HistoricTxn
Position: Request.Transaction

Elements of HistoricTxn			
Element Name	Description	Values / Limitations	
method	The transaction type	query	R
reference	The datacash_reference of the Batch Submission		R

Example XML Request for HistoricTxn complex elements

```
<HistoricTxn>
  <reference>3900900100010001</reference>
  <method>query</method>
</HistoricTxn>
```

2.5.2.3.3 XML Example Batch Submission Query

Example XML Batch Query

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>query</method>
      <reference>3900900100010001</reference>
    </HistoricTxn>
  </Transaction>
</Request>
```

2.5.2.4 Schema Elements for Query Response

2.5.2.4.1.1 BatchInputTxn

Element Name:	BatchInputTxn
Position:	Response

The element `BatchInputTxn` contains the Zlib compressed and Base64 encoded results of the Batch Request.

2.5.2.4.2 XML Example Query Response

Two Example XML Batch Query Responses

```

<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <BatchInputTxn>...</BatchInputTxn>
    <datacash_reference>3900900100010001</datacash_reference>
    <merchantreference>batch_ABCDEF</merchantreference>
    <mode>LIVE</mode>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <time>...</time>
</Response>

<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>3400900100010008</datacash_reference>
    <information>You have queried a Batch Input Transaction,
        which is currently being processed</information>
    <merchantreference>batch_ABCDEG</merchantreference>
    <mode>LIVE</mode>
    <reason>BatchInput: Processing</reason>
    <status>273</status>
    <time>...</time>
</Response>

```

2.5.2.5 Batch Input Request and Response - Common XML Elements

2.5.2.5.1 Schema Elements for Batch Request

2.5.2.5.2 BatchInputRequest

- o Header – see section 2.5.2.5.2.1
- o Transactions – see section 2.5.2.5.2.2

2.5.2.5.2.1 Header

Element Name: Header
Position: BatchInputRequest

Elements of Header			
Element Name	Description	Values / Limitations	
format	The name of the file format used	See section 0 for list. Must match that listed in the Batch Submission	R
reference	A reference number for the entire Batch	Must be between six and thirty alphanumeric in length	R

Example XML Request for Header complex elements

```

<Header>
  <format>xml_cardtxn</format>
  <reference>mybatch0002</reference>
</Header>

<Header>
  <format>xml_directdebit</format>
  <reference>mybatch_0003</reference>
</Header>

```

2.5.2.5.2.2 Transactions

Element Name:	Transactions
Position:	BatchInputRequest

Elements of Transactions		
Element Name	Description	Values / Limitations
Request	One Request element per transaction	Each Request must have identical Authentication elements, which must also be the same as those supplied in the Authentication element of the Batch Submission

Example XML for Transactions complex elements. There are 3 transactions.

```

<Transactions>
  <Request>
    ...
  </Request>
  <Request>
    ...
  </Request>
  <Request>
    ...
  </Request>
</Transactions>

```

2.5.2.5.3 Example XML Batch Request

```

<BatchInputRequest>
  <Header>
    <format>xml_cardtxn</format>
    <reference>batchref1234</reference>
  </Header>
  <Transactions>
    <Request>
      ...
    </Request>
    <Request>
      ...
    </Request>
    <Request>
      ...
    </Request>
    <Request>
      ...
    </Request>
  </Transactions>
</BatchInputRequest>

```

2.5.2.5.4 Schema Elements for Batch Response

2.5.2.5.5 BatchInputResponse

- o Header – this element is currently unpopulated
- o Transactions – see section 2.5.2.5.2.2

2.5.2.5.5.1 Transactions

Element Name:	Transactions
Position:	BatchInputResponse

Elements of Transactions	
Element Name	description
Response	One Response element per original transaction

Example XML for Transactions complex elements.
There are 2 transactions present

```

<Transactions>
  <Response>
    ...
  </Response>
  <Response>
    ...
  </Response>
</Transactions>

```

2.5.2.5.6 Example XML Batch Response

Example XML for a Batch Response, without the transaction details shown

```
<BatchInputResponse>
  <Header/>
  <Transactions>
    <Response>...</Response>
    <Response>...</Response>
  </Transactions>
</BatchInputResponse>
```

An Example XML Response, with 3 transactions

```
<BatchInputResponse>
  <Header/>
  <Transactions>
    <Response>
      <CardTxn>
        <authcode>183425</authcode>
        <card_scheme>Mastercard</card_scheme>
        <country>United Kingdom</country>
        <Cv2Avs>
          <cv2avs_status>ADDRESS MATCH ONLY</cv2avs_status>
          <policy>1</policy>
        </Cv2Avs>
        <issuer>BARCLAYS BANK PLC</issuer>
      </CardTxn>
      <datacash_reference>3300900100010037</datacash_reference>
      <merchantreference>1000001</merchantreference>
      <mode>LIVE</mode>
      <reason>ACCEPTED</reason>
      <status>1</status>
      <time>1067274153</time>
    </Response>
    <Response>
      <datacash_reference>3300900100010037</datacash_reference>
      <merchantreference>123402</merchantreference>
      <reason>Prereg: Merchant Not Subscribed</reason>
      <status>251</status>
      <time>1067274153</time>
    </Response>
    <Response>
      <CardTxn>
        <authcode>100000</authcode>
      </CardTxn>
      <datacash_reference>3300900100010037</datacash_reference>
      <merchantreference>123402</merchantreference>
      <reason>ACCEPTED</reason>
      <status>1</status>
      <time>1067274153</time>
    </Response>
  </Transactions>
</BatchInputResponse>
```

2.5.2.6 Batch Request and Response - Common CSV Elements

2.5.2.6.1 CSV File Format Definition

The definition of CSV format used is as follows

- Allowable characters within a CSV field include 0x09 (tab) and the inclusive range of 0x20 (space) through 0x7E (tilde).
- A field within CSV may be surrounded by double-quotes. Double quoting is used by default to surround fields in CSV format responses.
- A field within CSV must be surrounded by double-quotes to contain a comma.
- A field within CSV must be surrounded by double-quotes to contain an embedded double-quote, represented by a pair of consecutive double-quotes.
- A CSV field may be terminated by 0x0A (line feed) or by 0x0D, 0x0A (carriage return, line feed).
- Blank lines and lines starting with 0x23 (#) are ignored.

2.5.2.6.2 CSV Request Headers

Each CSV file contains should contain a header line containing these fields:

Headers for CSV format, in order			
Element Name	Description	Values / Limitations	
format	The name of the file format used	See section 2.5.2.6.1 for list. Must match that listed in the Batch Submission	R
reference	A reference number for the entire Batch	Must be between six and thirty alphanumeric in length	R
client	The vTID of the account the batch is to be processed against	Must be the same as those supplied in the Authentication element of the Batch Submission	R
password	The password of the account the batch is to be processed against		R

Two CSV Examples for a Batch Request Headers

```
csv_card_txn,batch_csv_card_txn_6,99000001,mypasswd
csv prereg,batch csv prereg 3,99000001,mypasswd
```

2.5.2.6.3 CSV Response

The Response elements returned will be the same for each file format. These are:

Fields for CSV responses			
	Element Name	Description	XML equivalent
1	merchant_ref	The merchant_ref of the original transaction	merchant_reference
2	datacash_ref	The datacash_reference of the transaction	datacash_reference
3	response_code	The DataCash Response Code	status
4	status_message	Additional information about the response_code	reason
5	authcode	The result of the authorisation request, for transactions submitted for authorisation only -otherwise blank	authcode
6	time	The Unix Timestamp of the transaction	time

Example CSV responses

```
"ABCD9876","3000900100010005","7","REFERRED","CALL AUTH","1080920429"
"3400900100010008","3400900100010008","1","CANCELLED
OK","","1080920429"
"payment_1223","3600900100010889","1","ACCEPTED","896786","1080920430"
"ACBD9877","3600900100010890","1","ACCEPTED","548620","1080920431"
"valid_ddrefund","3000900100010048","1","ACCEPTED","",1080920435"
"ddrefund_no_setup","3800900100010049","137","Unable to locate
matching DDI transaction","",1080920436"
```

2.5.3 Example Batch Requests

This section covers the Batch Request file, for all services.

Please note that the maximum size of a single Batch Request – before compression – is 1MB (1024KB). Any files larger than this will be rejected.

2.5.3.1 xml_cardtxn

An example file with three transactions is shown below. The file contains three transactions, one Pre-Registered and two Credit and Debit Card (a `pre` with AVS data and a `cancel`). The transactions could be screened using either the ReD or T3M services if the account is configured for this at DataCash. A Batch Submission example for this file is shown in section 2.5.2.2.

Example XML Batch Request containing three transactions with a total value of 395.99

```

<BatchInputRequest>
  <Header>
    <format>xml_cardtxn</format>
    <reference>batchref1234</reference>
  </Header>
  <Transactions>
    <Request>
      <Authentication>
        <client>99000001</client>
        <password>*****</password>
      </Authentication>
      <Transaction>
        <CardTxn>
          <Card>
            <expirydate>04/06</expirydate>
            <startdate>01/04</startdate>
            <pan>633300*****1</pan>
            <Cv2Avs>
              <street_address>1 High Street, UK </street_address>
              <postcode>S01 2CD</postcode>
              <policy>1</policy>
            </Cv2Avs>
          </Card>
          <method>pre</method>
        </CardTxn>
        <TxnDetails>
          <amount currency="GBP">146.00</amount>
          <merchantreference>1000001</merchantreference>
        </TxnDetails>
      </Transaction>
    </Request>
    <Request>
      <Authentication>
        <password>*****</password>
        <client>99000001</client>
      </Authentication>
    </Request>
  </Transactions>
</BatchInputRequest>

```

```

<Transaction>
  <TxnDetails>
    <merchantreference>123402</merchantreference>
    <amount currency="USD">249.99</amount>
  </TxnDetails>
  <CardTxn>
    <method>auth</method>
    <card_details type="preregistered">
      310000008888881212</card_details>
    </CardTxn>
  </Transaction>
</Request>
<Request>
  <Authentication>
    <client>99000001</client>
    <password>*****</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <reference>3100000088888888</reference>
      <method>txn_refund</method>
    </HistoricTxn>
  </Transaction>
</Request>
</Transactions>
</BatchInputRequest>

```

2.5.3.2 xml_chp_cardtxn

Example XML Batch Request containing three transactions with a total value of 120.72 in two currencies

```

<BatchInputRequest>
  <Header>
    <format>xml_chp_cardtxn</format>
    <reference>batchref1234</reference>
  </Header>
  <Transactions>
    <Request>
      <Authentication>
        <password>*****</password>
        <client>99000001</client>
      </Authentication>
      <Transaction>
        <TxnDetails>
          <capturemethod>parked</capturemethod>
          <merchantreference>1234567890blahh</merchantreference>
          <amount currency="GBP">54.00</amount>
        </TxnDetails>
        <CardTxn>
          <card_details type="track2_data">
            ****</card_details>
          <Terminal id="82000062">
            <terminal_capabilities ic_reader="true"
              magnetic_stripe_reader="true" manual_card_entry="true"/>
            <features_capabilities pin_pad_available="true"/>
          </Terminal>
        </CardTxn>
      </Transaction>
    </Request>
  </Transactions>
</BatchInputRequest>

```

```

<method>auth</method>
<reason_online_code>10</reason_online_code>
<receipt_no>000123</receipt_no>
<ICC>
    <seq_or_issue_number>*</seq_or_issue_number>
    <auth_response_code>00</auth_response_code>
    <crypto_txn_amount>54.00</crypto_txn_amount>
    <crypto_tran_type>00</crypto_tran_type>
    <term_txn_date>031107</term_txn_date>
    <txn_currency_code>826</txn_currency_code>
    <txn_country_code>826</txn_country_code>
    <arqc>*****</arqc>
    <aip>*****</aip>
    <atc>*****</atc>
    <unpredictable>*****</unpredictable>
    <tvr>*****</tvr>
    <issuer_app_data>*****</issuer_app_data>
    <app_usage_control>****</app_usage_control>
    <crypto_info_data>**</crypto_info_data>
    <cvm>*****</cvm>
    <aid>*****</aid>
    <term_app_ver_no>****</term_app_ver_no>
    <txn_status_info>****</txn_status_info>
    <term_type>**</term_type>
    <term_capabilities>*****</term_capabilities>
    <pos_entry_mode>32</pos_entry_mode>
    <other_card_data>*****</other_card_data>
</ICC>
</CardTxn>
</Transaction>
</Request>
<Request>
    <Authentication>
        <password>*****</password>
        <client>99000001</client>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <capturemethod>keyed</capturemethod>
            <merchantreference>1234560002</merchantreference>
            <amount currency="EUR">30.00</amount>
        </TxnDetails>
        <CardTxn>
            <Terminal id="82000062">
                <terminal_capabilities ic_reader="true" magnetic_stripe_reader="true" manual_card_entry="true"/>
                <features_capabilities pin_pad_available="true"/>
            </Terminal>
            <method>auth</method>
            <receipt_no>000124</receipt_no>
            <ICC>
                <term_type>**</term_type>
            </ICC>
            <Card>
                <pan>444433*****1</pan>
                <expirydate>12/09</expirydate>
            </Card>
        </CardTxn>
    </Transaction>
</Request>

```

```

        </CardTxn>
    </Transaction>
</Request>
<Request>
    <Authentication>
        <password>*****</password>
        <client>99000001</client>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <capturemethod>swiped</capturemethod>
            <merchantreference>1234560002</merchantreference>
            <amount currency="GBP" cashback="20.00">26.23</amount>
        </TxnDetails>
        <CardTxn>
            <card_details type="track2_data">
                *****
            </card_details>
            <Terminal id="82000062">
                <terminal_capabilities ic_reader="true"
                    magnetic_stripe_reader="true" manual_card_entry="true"/>
                <features_capabilities pin_pad_available="true"/>
            </Terminal>
            <method>auth</method>
            <receipt_no>000125</receipt_no>
            <ICC>
                <term_type>**</term_type>
            </ICC>
        </CardTxn>
    </Transaction>
</Request>
<Request>
    <Authentication>
        <password>*****</password>
        <client>99000001</client>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <capturemethod>keyed</capturemethod>
            <merchantreference>1234560002</merchantreference>
            <amount currency="GBP">10.49</amount>
        </TxnDetails>
        <CardTxn>
            <Terminal id="82000062">
                <terminal_capabilities ic_reader="true"
                    magnetic_stripe_reader="true" manual_card_entry="true"/>
                <features_capabilities pin_pad_available="true"/>
            </Terminal>
            <method>refund</method>
            <receipt_no>000126</receipt_no>
            <ICC>
                <term_type>**</term_type>
            </ICC>
            <Card>
                <pan>444433*****1</pan>
                <expirydate>12/09</expirydate>
            </Card>
        </CardTxn>
    </Transaction>
</Request>

```

```

        </Transaction>
    </Request>
</Transactions>
</BatchInputRequest>
```

2.5.3.3 xml_direct_credit

Example XML Batch Request xml_directcredit containing two transactions with a total value of 80.00

```

<BatchInputRequest>
    <Header>
        <format>xml_directcredit</format>
        <reference>45356132</reference>
    </Header>
    <Transactions>
        <Request>
            <Authentication>
                <password>fred</password>
                <client>21850000</client>
            </Authentication>
            <Transaction>
                <TxnDetails>
                    <merchantreference>standardDC0000054</merchantreference>
                    <amount>45.00</amount>
                </TxnDetails>
                <DirectCreditTxn>
                    <method>directcredit</method>
                    <sortcode>826300</sortcode>
                    <accountnumber>80000990</accountnumber>
                    <accountname>Jo Bloggs</accountname>
                </DirectCreditTxn>
            </Transaction>
        </Request>
        <Request>
            <Authentication>
                <password>fred</password>
                <client>21850000</client>
            </Authentication>
            <Transaction>
                <TxnDetails>
                    <merchantreference>ddrefund00434582</merchantreference>
                    <amount>35.00</amount>
                </TxnDetails>
                <HistoricTxn>
                    <method>ddrefund</method>
                    <reference>3600900100010007</reference>
                </HistoricTxn>
            </Transaction>
        </Request>
    </Transactions>
</BatchInputRequest>
```

2.5.3.4 xml_directdebit

An example `xml_directdebit` file with four transactions is shown below. The file contains one setup, one revoke, two drawdowns and a confirm. A Batch Submission example for this file is shown in section 2.5.3.4.

Example XML Batch Request containing five transactions with a total value of 395.99

```

<BatchInputRequest>
  <Header>
    <format>xml_directdebit</format>
    <reference>batchref1236</reference>
  </Header>
  <Transactions>
    <Request>
      <Authentication>
        <client>99000001</client>
        <password>*****</password>
      </Authentication>
      <Transaction>
        <DirectDebitTxn>
          <sortcode>938611</sortcode>
          <accountnumber>02149187</accountnumber>
          <accountname>Mr A. N. Other</accountname>
          <method>setup</method>
        </DirectDebitTxn>
        <TxnDetails>
          <merchantreference>1000001</merchantreference>
        </TxnDetails>
      </Transaction>
    </Request>
    <Request>
      <Authentication>
        <password>*****</password>
        <client>99000001</client>
      </Authentication>
      <Transaction>
        <TxnDetails>
          <merchantreference>123402</merchantreference>
          <amount>100.99</amount>
        </TxnDetails>
        <HistoricTxn>
          <method>drawdown</method>
          <reference>90000005</reference>
        </HistoricTxn>
      </Transaction>
    </Request>
    <Request>
      <Authentication>
        <password>*****</password>
        <client>99000001</client>
      </Authentication>
      <Transaction>
        <TxnDetails>

```

```
<merchantreference>123402</merchantreference>
<amount>295.00</amount>
</TxnDetails>
<HistoricTxn>
    <method>drawdown</method>
    <reference>90000005</reference>
    <duedate>20050929</duedate>
</HistoricTxn>
</Transaction>
</Request>
<Request>
    <Authentication>
        <client>99000001</client>
        <password>*****</password>
    </Authentication>
    <Transaction>
        <HistoricTxn>
            <method>revoke</method>
            <reference>40488707</reference>
        </HistoricTxn>
        <TxnDetails>
            <merchantreference>383285870486111</merchantreference>
        </TxnDetails>
    </Transaction>
</Request>
<Request>
    <Authentication>
        <client>99000001</client>
        <password>*****</password>
    </Authentication>
    <Transaction>
        <HistoricTxn>
            <method>confirm</method>
            <reference>40502431</reference>
        </HistoricTxn>
        <TxnDetails>
            <merchantreference>383366196990741</merchantreference>
        </TxnDetails>
    </Transaction>
</Request>
</Transactions>
</BatchInputRequest>
```

2.5.3.5 csv_card_txn

This CSV file format should be used to obtain authorisation for Credit and Debit Card transactions using the Bank Card Service. It can be used for both the one stage processing model and the first stage of the two stage processing model.

If you are using the Capture Method model of Recurring Transactions, this file format can be used. For this Service, the `capturemethod` should be specified.

Fields for format csv_card_txn			
	Element Name	Description	Values / Limitations
1	pan	As described in section 2.2.1.1.1	R
2	expiry_date		R
3	start_date		M
4	issue		M
5	amount	As described in section 2.5.3.5	R
6	currency		R
7	merchant_ref		R
8	method	The transaction type auth pre refund erp	R
9	capturemethod	As described in section 2.7.3.10.2	M ¹
10	authcode	The authorisation code received from the bank	If presented, must be value received from Banks Authorisation centre

¹Mandatory if the account is configured with multiple merchant ID environments

Example csv_card_txn file – six transactions with a value of 2697.31

```
csv_card_txn,batch_csv_card_txn_6, 99000001,mypasswd
444433332221111,01/09,,,99.99,GBP,myref0000023,pre,ecomm,
5473000000000007,01/09,,,530.01,GBP, myref0000024,auth,ecomm,
343434343434343,01/09,,,13.00,USD, myref0000025,pre.,
493600000000000001,01/09,,2,9.95,GBP,
myref0000026,erp,ecomm,
675900000000026,01/09,01/01,,44.36,GBP,
myref0000027,refund,ecomm,
5473000000000007,01/09,,,2000.00,GBP,
myref123,auth,ecomm,123456
```

2.5.3.6 csv_avs_card_txn

This CSV file format is similar to `csv_card_txn` format (section 2.5.3.5), but it also allows AVS information to be provided with the Credit and Debit Card transactions if required. CV2 information cannot be provided as the CV2 number should not be stored.

Fields for format <code>csv_avs_card_txn</code>			
	Element Name	Description	Values / Limitations
1-10	As described in section 2.5.3.6		
11	address_line_1		O
12	address_line_2		O
13	address_line_3	As described in section 2.3.1.1.2	O
14	address_line_4		O
15	postcode		O
16	policy_number	The Standard Policy against which the transaction is to be checked against	See website ¹ O ²

¹ While all of these values are valid, as only the AVS check can be carried out, only policies 1 or 5 should be chosen.

² The default policy registered against the account will be used if not value is set

Example `csv_avs_card_txn` file – two transactions for a total value of 3.96

```
csv_avs_card_txn,batch_csv_avs_card_txn_2,99000001,mypasswd
4444333322221111,01/09,,,1.98,GBP,ABC123,auth,,,Flat 12,345 Main
Street,Putney,,A9 87XY,5
5473000000000007,01/09,,,1.98,GBP,ABC124,auth,ecomm,,Flat 12,345
Main Street,Putney,,X1 23AB,5
```

```
# Note: the line wrapping is this example should not
# be used in a real file
```

2.5.3.7 csv_chp_card_txn

This CSV file format is used for Cardholder Present transactions.

Fields for format csv_chp_card_txn			
	Element Name	Description	Values / Limitations
1-7	As described in section 2.5.3.7		
8	method	The transaction type	auth refund R
9	capturemethod	Indicates how the transaction was accepted at the POS	keyed swiped R
10	authcode	The authorisation code generated for the transaction	- R
11	authcode_method	Indicates how the authcode was obtained	terminal online telephone O
12	cashback	The amount of cashback required by the customer	Amount in Major.Minor currency units (i.e. 10.98) O
13	tid	The Terminal ID (TID) used for the transaction	DataCash will advise R
14	hot_card_file	Indicates which Hot Card File was used during the transaction	none reserved Switch 400 Switch 8000 O
15	ic_reader	Whether the Terminal has Chip reading capability	true false R
16	magnetic_stripe_reader	Whether the Terminal has magnetic stripe reading capability	true false R
17	manual_card_entry	Whether the Terminal has the facility to enable card details to be manually entered	true false R
18	pin_pad_available	Whether the Terminal has a keypad to enable the PIN to be entered	true false R

Example csv_chp_card_txn file containing three transactions with a value of 643

```
csv_chp_card_txn,batch_csv_chp_card_txn_0008,99000001,mypasswd
444433332221111,01/09,,,99.99,GBP,myref0000023,refund,keyed,482135,terminal,
85621087,,true,true,true
5473000000000007,01/09,,,530.01,GBP,myref0000024,auth,swiped,955AU3,telephone,
85621087,,true,true,true
343434343434343,01/09,,,13.00,USD,myref0000025,auth,swiped,FLDURN,terminal,
85621086,,true,true,true

# Note: the line wrapping is this example should not
# be used in a real file
```

2.5.3.8 csv_historic

This CSV file format is used for the Credit and Debit Card Services to complete two stage processing using the `fulfill` transaction and to perform `cancel`, `txn_refund`, `accept_fraud` and `authorize_referral_request` transactions.

Fields for format csv_historic			
	Element Name	Description	Values / Limitations
1	reference	DataCash unique reference of the original transaction	- R
2	method	The transaction type	fulfill txn_refund cancel accept_fraud authorize_referral_request R
3	authcode	The authorisation code of the original pre or esp transaction, or the value obtained from the Banks Authorisation centre if the transaction was referred.	For fulfill or authorize_referral_request only R
4	amount	The value of the transaction	For fulfill or txn_refund only O ¹

¹ Transactions will be fulfilled/refunded for the full value of the original transaction if this field is empty.

Example csv_historic file containing five transactions with a value of 2501.00

```
csv_historic,batch_csv_historic_0008,99000001,mypasswd
# fulfill supplying amount
3000900100010005,fulfill,100000,1001.00
# cancel
3400900100010008,cancel,,
# fulfill, for the full value of the original txn
3800900100010025, ,100001,,

# txn_refund, two refunds to the same original txn
3600900100010026,txn_refund,,1000.00
3600900100010026,txn_refund,,500.00
```

2.5.3.9 csv_prereg

This CSV file format allows the Pre-Registered Card Service to be utilised.

Fields for format csv_prereg			
	Element Name	Description	Values / Limitations
1	merchant_ref	As described in section 2.6.1.1.2	R
2	reference	DataCash unique reference of the original transaction	R
3	amount		R
4	currency		R
5	method	As described in section 2.6.1.1.5	R
6	capturemethod		O
7	authcode		O

Example csv_prereg file containing three transactions for a total value of 2023.34

```
csv_prereg,batch_csv_prereg_3,99000001,mypasswd

# card txns using card details from previous historic txns
prereg_230006,2185000198760000,1000.00,GBP,auth,ecomm,
prereg_230007,3200900100010014,1000.00,USD,refund,,

# using prereg to carry out manual auth of previous
# referred txn
prereg_230008,3700900100010021,23.34,GBP,pre,ecomm,123123
```

2.5.3.10 csv_dc_cc_account

This CSV file format enables the Direct Credit Card Collection Account Service to be accessed.

Fields for format csv_dc_cc_account			
	Element Name	Description	Values / Limitations
1	merchant_ref	The reference number of the drawdown	See limitations R
2	pan	The card number	R
3	amount	The value to credit	Major.minor currency units (i.e. (10.99)) R
4	card_name	The card holders name	O
5	expiry_date	The card expiry date	- O

Example csv_dc_cc_account file containing two transactions with a value of 20.00

```
csv_dc_cc_account,batch000134,99000001,mypasswd
valid_cardaccountpayment,4560720000000007,10.00,,
optional fields,4560720000000007,10.00,MR FOO BAR,12/09
```

2.5.3.11 csv_dc_ddrefund

This CSV format supports Direct Credit ddrefund transactions.

Fields for format csv_dc_ddrefund			
	Element Name	Description	Values / Limitations
1	merchant_ref	The reference number of the DDI	See limitations R
2	reference	The datacash_reference of the original transaction	- R
3	amount	The amount to credit	Major.minor currency units (i.e. (10.99)) R

Example csv_dc_ddrefund file containing two transactions with a value of 25.00

```
csv_dc_ddrefund,batch000131,99000001,mypasswd
ddrefundABCD000345,10000003,10.00
45347ACBD,3400900100010008,15.00
```

2.5.3.12 csv_dc_std

This CSV format enables the Standard Direct Credit service to be used.

Fields for format csv_dc_standard			
	Element Name	Description	Values / Limitations
1	merchant_ref	The reference number of the drawdown	See limitations R
2	amount	The value to credit	Major.minor currency units (i.e. (10.99) R
3	sort_code	The customer's sort code	6 digits R
4	acc_num	The customer's account number	- R
5	acc_name	The customer's account name	Maximum of 18 characters R
6	bank_ref		See limitations O

Example csv_dc_std file containing two transactions with a value of 60.00

```
csv_dc_standard,batch000130,99000001,mypasswd
directcredit00006,30.00,826300,80000990,Bert Weiss,
credit00009,30.00,826300,80000990,Bert Weiss,Money for petrol
```

2.5.3.13 csv_dd_drawdown

This CSV file format supports drawdown transactions for the Standard Direct Debit Service.

Fields for format csv_dd_drawdown			
	Element Name	Description	Values / Limitations
1	merchant_ref	The reference number of the drawdown	See Limitations R
2	reference	The datacash_reference number of the DDI	- R
3	amount	The value to debit	Major.minor currency units (i.e. (10.99) R
4	duedate	The date to perform the drawdown	yyyymmdd format O
5	bacs_tran_code	Enables the BACS transaction code to be explicitly set	01 17 18 19 O

Example csv_dd_drawdown file containing three transactions with a value of 30.00

```
csv_dd_drawdown,batch000130,99000001,mypasswd
drawdown_BE30006,10000003,10.00,,
drawdown_HI90135,10000004,10.00,20041212,
drawdown_JG00042,10000005,10.00,,18
```

2.5.3.14 csv_dd_setup

This CSV format is used for the Standard Direct Debit Service to perform `setup` and `presetup` transactions.

Fields for format csv_dd_setup			
	Element Name	Description	Values / Limitations
1	merchant_ref	The reference number of the DDI	Must be of the format agreed with your Sponsoring bank. See limitations . R
2	sort_code	The sort code of the customer's bank	6 digits R
3	acc_num	The customer's account number	- R
4	acc_name	The name of the account holder	Maximum of 18 characters R
5	method	Determines the processing model to be used	setup presetup R
6	active	Enables an existing setup to be transferred to the DataCash system	true O
7	type	Enables a non_AUDDIS DDI to be converted to an AUDDIS DDI	conversion O

Example csv_dd_setup file containing four transactions with a value of 0.00 (no amount values present)

```
csv_dd_setup,batch000128,99000001,mypasswd
valid_setup,938611,02149187,MR F BAR,setup,,
setup_active,938612,02149188,MR A N OTHER,setup,true,
setup_conversion,938613,02149189,MR A N
OTHER,setup,,conversion
# DD presetup transaction
valid_presetup,938611,02149187,MR F BAR,presetup,,
valid_presetup456, 938611,02149236,"J, M & P
Smith",presetup,,
```

2.5.3.15 csv_dd_setup_edit

This CSV file format is used with the Direct Debit Service to perform for `confirm` and `revoke` transactions.

Fields for format csv_dd_setup_edit			
	Element Name	Description	Values / Limitations
1	merchant_ref	The reference number of the DDI	As supplied for the original R
2	reference	The <code>datacash_reference</code> of the original transaction	- R
3	method	The operation required	confirm revoke R

Example csv_dd_setup_edit file containing four transactions with a value of 0.00 (no amount values present)

```
csv_dd_setup_edit,batch000129,99000001,mypasswd
# confirm presetup with 8 digit Unique Reference
confirm_presetup,12345678,confirm
# revoke setup with 8 digit Unique Reference
revoke_setup,12345679,revoke
# confirm presetup with 16 digit DataCash Reference returned
by previous query response
confirm_presetup2,3400900100010008,confirm
# revoke previous setup in the same batch
revoke_previous_setup,3400900100010008,revoke
```

[Back to Contents](#)

2.6 CARD HOLDER PRESENT

2.6.1 Chip and Pin

A technical introduction to this service is available on the website:

<http://www.datacash.com/gettingproducts.php?id=Chip-PIN>

2.6.1.1 Schema Elements for Request

As different fields may be presented in different situations, each field will be labelled with the following key:

- O - Optional
- R - Required, field must be presented
- M - Mandatory, field must be presented if it is available
- X - Excluded, field is excluded in specific situations

Transactions with Card Details

The `creditcheck` and `auth` transaction types require certain information to be presented. This is passed in several places in the schema:

- Request
 - Authentication – [section 2.1.1.2](#)
 - Transaction
 - `TxnDetails` – [section 2.6.1.1.1](#)
 - `CardTxn` – [section 2.6.1.1.2](#)
 - `Card` – [section 2.6.1.1.5](#)
 - `ICC` – [section 2.6.1.1.4](#)
 - `Terminal` – [section 2.6.1.1.3](#)

Historic Transactions

The `cancel` and `authorize_referral_request` transaction types require information to be passed in:

- Request
 - Authentication – [section 2.1.1.2](#)
 - Transaction
 - `HistoricTxn` – [section 2.3.3.1.9](#)

2.6.1.1.1 TxnDetails

Element Name:	TxnDetails
Position:	Request.Transaction

Elements of TxnDetails			
Element Name	Description	Values / Limitations	Required?
merchantreference	A unique reference number to distinguish each transaction	See Answer	R
amount	The value of the transaction		R
capturemethod	Indicates how the transaction was accepted at the POS	parked keyed swiped	R

Attributes of TxnDetails			
Attribute Name	Attribute of element	Description	Required?
cashback	amount	The amount of cashback requested by the customer (Cannot be supplied with cashadvance attribute)	O
cashadvance	amount	Indicates whether the transaction was a Cash Advance (Cannot be supplied with cashback attribute)	O

Example XML for TxnDetails element

```

<TxnDetails>
  <merchantreference>123402</merchantreference>
  <amount currency="GBP" cashback="30.00">59.99</amount>
  <capturemethod>parked</capturemethod>
</TxnDetails>

<TxnDetails>
  <merchantreference>AB78324293452</merchantreference>
  <amount currency="EUR">26.50</amount>
  <capturemethod>swiped</capturemethod>
</TxnDetails>

<TxnDetails>
  <merchantreference>AB78324293452</merchantreference>
  <amount currency="GBP" cashadvance="true">26.50</amount>
  <capturemethod>swiped</capturemethod>
</TxnDetails>

```

2.6.1.1.2 CardTxn

Element Name:	CardTxn
Position:	Request.Transaction

Elements of CardTxn					
Element Name	Description	Values / Limitations	Keyed	Swiped	Parked
authcode	The authorisation code, if on-line authorisation is not required	See note below	O	X	X
Card	See section 2.6.1.1.5		R	X	X
card_details	Data read from the magnetic stripe of the card, or equivalent from the ICC	Must be Base64 encoded	-	X	X
hotcardfile	Indicates which Hot Card File was used during the transaction	-	O	O	O
ICC	See section 2.6.1.1.4		-	-	R
method	The transaction method	auth creditcheck	R	R	R
reason_online_code	For on-line transactions: the reason for being on-line. As defined by APACs		O	O	O
Terminal	See section 2.6.1.1.3		R	R	R

Note: For swiped and parked transactions, card details may be presented in either the Card element or the card_details element:

1. The Card element is required if an authorisation code is submitted.
2. The card_details element is required if the authorisation code is not submitted.

Attributes of CardTxn				
Attribute Name	Attribute of element	Description	Values / Limitations	Required?
method	authcode	Indicates how the authcode was obtained.	Must be present if authcode element is presented: online terminal telephone	R
auth_datetime	authcode	The actual date & time of the transaction	As a unixtimestamp	R ¹
type	card_details	Indicates the type of data	track2_data	R

¹ Field is required for merchants using NatWest Streamline. For other Acquiring Banks this field is optional.

Example XML for CardTxn element, for a parked transaction

```
<CardTxn>
    <Terminal id="...">...</Terminal>
    <card_details type="track2_data">
        NDkyOTQ5ODMxMTQwMDAwMj0wODAzMjAxMDA5ODgwMDAwMDAwMR8wMA==
    </card_details>
    <hotcardfile>...</hotcardfile>
    <ICC>...</ICC>
    <method>creditcheck</method>
    <reason_online_code>10</reason_online_code>
</CardTxn>
```

Example XML for CardTxn element, for a swiped transaction without an authcode

```
<CardTxn>
    <Terminal id="...">...</Terminal>
    <card_details type="track2_data">
        NDkyOTQ5ODMxMTQwMDAwMj0wODAzMjAxMDA5ODgwMDAwMDAwMR8wMA==
    </card_details>
    <method>creditcheck</method>
</CardTxn>
```

Example XML for CardTxn element, for a swiped or parked transaction with authcode

```
<CardTxn>
  <Terminal id="...">...</Terminal>
  <Card>...</Card>
  <method>creditcheck</method>
  <authcode method="terminal">
    auth_datetime="1164821283">451576</authcode>
</CardTxn>
```

Example XML for CardTxn element, for a keyed transaction

```
<CardTxn>
  <Terminal id="...">...</Terminal>
  <Card>...</Card>
  <method>auth</method>
</CardTxn>
```

2.6.1.1.3 Terminal

Element Name: Terminal
Position: Request.Transaction.CardTxn

Elements of Terminal	
Element Name	
terminal_capabilities	
features_capabilities	
message_capabilities	

Attributes of Terminal and its children				
Attribute Name	Attribute of element	Description	Values	Required?
id	Terminal	The terminal or TID number	DataCash will advise	R
ic_reader	terminal_capabilities	Chip reading functionality	true false	R
magnetic_stripe_reader	terminal_capabilities	Magnetic stripe reading functionality	true false	R
manual_card_entry	terminal_capabilities	Facility to manually enter card details	true false	R
cardholders_device	features_capabilities	Cardholder's device (eg. personal computer, mobile phone, digital TV or similar device)	true false	O
card_capture_device	features_capabilities	Terminal or operator able to capture cards	true false	O
pin_pad_available	features_capabilities	Keypad to enable PIN to be entered	true false	R
unattended_device	features_capabilities	Indicates whether transaction is processed from an unattended terminal	true false	O
downline_load_floor_limit	message_capabilities	As per APACS standard 70 book 2 appendix A.9	true false	O
downline_load_referral	message_capabilities		true false	O
hold_capability	message_capabilities		true false	O
response_additional_data_support	message_capabilities		true false	O

Example XML for Terminal element

```
<Terminal id="82000062">
  <terminal_capabilities ic_reader="true"
    magnetic_stripe_reader="true"
    manual_card_entry="true"/>
  <features_capabilities pin_pad_available="true"/>
  <message_capabilities downline_load_floor_limit="false"
    downline_load_referral="false"
    hold_capability="false"
    response_additional_data_support="true"/>
</Terminal>
```

2.6.1.1.4 ICC

This parent element contains the data relating to the Chip & Pin functionality. All elements within this parent are mandatory. Please refer to APACS29 v18 (page 77) and APACS30 v18 (page 73) for definitions of these elements.

Element Name:	ICC
Position:	Request.Transaction.CardTxn

2.6.1.1.5 Card

This element is only required for keyed transactions.

Element Name:	Card
Position:	Request.Transaction.CardTxn

Elements of Card			
Element Name	Description	Values / Limitations	
pan	The card number	Must be a valid card number	R
expiry_date	expiry date for the card	MM/YY format	R
start_date	start date for the card	MM/YY format	M
issue_number	issue number of the card	One or two digits	M

Example XML for Card element

```

<Card>
  <expirydate>01/12</expirydate>
  <pan>444433*****1</pan>
</Card>

<Card>
  <expirydate>01/06</expirydate>
  <issuenumber>1</issuenumber>
  <startdate>01/99</startdate>
  <pan>446281*****2</pan>
</Card>

```

2.6.1.1.6 HistoricTxn

This element should be presented for `cancel` or `authorize_referral_request` transactions only.

Element Name:	HistoricTxn
Position:	Request.Transaction

Elements of HistoricTxn				
Element Name	Description	Values / Limitations	<code>cancel</code>	<code>authorize_referral_request</code>
method	The transaction type	<code>cancel</code> <code>authorize_referral_request</code>	R	R
datacash_reference	The datacash_reference of the transaction to be cancelled	Must be valid transaction	R	R
authcode	Value received from Banks Authorisation centre		X	R

Attributes of Elements in HistoricTxn					
Attribute Name	Attribute of element	Description	Values / Limitations	<code>cancel</code>	<code>authorize_referral_request</code>
reversal	method	Indicates whether a reversal should be attempted	true false	O	X
method	authcode	Indicates how the authcode was obtained.	online terminal telephone	X	R
auth_datetime	authcode	The actual date & time of the transaction	As a unixtimestamp	X	R ¹

¹ Field is required for merchants using NatWest Streamline. For other Acquiring Banks this field is optional.

Example XML for HistoricTxn element

```
<HistoricTxn>
    <method reversal="true">cancel</method>
    <reference>4600200040913258</reference>
</HistoricTxn>

<HistoricTxn>
    <method>cancel</method>
    <reference>4500200040913862</reference>
</HistoricTxn>

<HistoricTxn>
    <reference>4500200040913862</reference>
    <authcode method="telephone" auth_datetime="1164821283">A6</authcode>
    <method>authorize_referral_request</method>
</HistoricTxn>
```

2.6.1.2 Example XML Requests

Example XML Request for a parked transaction

```

<Request>
  <Authentication>
    <client>99000001</client>
    <password>mypasswd</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>57483573457gegrreg</merchantreference>
      <amount currency="GBP">54.00</amount>
      <capturemethod>parked</capturemethod>
    </TxnDetails>
    <CardTxn>
      <Terminal id="82000062">
        <terminal_capabilities ic_reader="true"
          magnetic_stripe_reader="true" manual_card_entry="true"/>
        <features_capabilities pin_pad_available="true"/>
      </Terminal>
      <reason_online_code>10</reason_online_code>
      <card_details type="track2_data">NDkyOTQ5ODMxMTQwMDAwMj
        0wODAzMjAxMDA5ODgwMDAwMR8wMA==</card_details>
    <ICC>
      <seq_or_issue_number>00</seq_or_issue_number>
      <crypto_tran_type>00</crypto_tran_type>
      <txn_currency_code>826</txn_currency_code>
      <term_country_code>826</term_country_code>
      <arqc>ED128CAB559EC54C</arqc>
      <aip>5C00</aip>
      <atc>0CF3</atc>
      <unpredictable>81B72731</unpredictable>
      <tvr>0000008000</tvr>
      <issuer_app_data>06FE0A03A4A800</issuer_app_data>
      <crypto_info_data>80</crypto_info_data>
      <cvm>410302</cvm>
      <aid>A0000000031010</aid>
      <term_app_ver_no>0084</term_app_ver_no>
      <term_capabilities>7546</term_capabilities>
    </ICC>
    <method>creditcheck</method>
  </CardTxn>
  </Transaction>
</Request>

```

Example XML Request for a swiped transaction

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>mypasswd</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>924100120050503165050 </merchantreference>
      <amount currency="GBP">40.01</amount>
      <capturemethod>swiped</capturemethod>
    </TxnDetails>
    <CardTxn>
      <Terminal id="82000062">
        <terminal_capabilities magnetic_stripe_reader="true">
          <manual_card_entry="true" ic_reader="true"/>
          <features_capabilities pin_pad_available="true"/>
        </Terminal>
        <card_details type="track2_data">OzQ5Mjk0OTgzMTE0MDAwMDI9MDg
        wMzIwMTAwOTg4MDAwMDAwMDE/Pw==</card_details>
        <method>creditcheck</method>
      </CardTxn>
    </Transaction>
  </Request>
```

Example XML Request for a cancellation

```
<Request>
  <Authentication>
    <client>99000001</client>
    <password>mypasswd</password>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method reversal="true">cancel</method>
      <merchantreference>575600720050517212621</merchantreference>
    </HistoricTxn>
  </Transaction>
</Request>
```

2.6.1.3 Schema Elements for Response

2.6.1.3.1 Response

The following additional elements returned in the Response element. Please refer to section 0 for descriptions of the general elements.

Additional Elements of Response	
Element Name	Description
referral_tel_no	As described in the APACS30 Response
floorlimit	
issuer_auth_data	
issuer_script_data	
authorising_entity	Indicates (where available) the party who provided the authorisation. Only returned for creditcheck transactions. The returned value will be one of: <ul style="list-style-type: none"> • card acceptor • acquirer • card scheme • card issuer

Attributes of Elements in Response		
Attribute Name	Attribute of element	Description
reversal	reason	Indicates the status of reversal requests

2.6.1.4 Example XML Responses

Example XML Response to an auth or creditcheck

```

<Response>
  <acquirer_message>AUTH CODE:100100</acquirer_message>
  <acquirer_response_code>00</acquirer_response_code>
  <CardTxn>
    <authcode>100100</authcode>
    <card_scheme>VISA</card_scheme>
    <country>United Kingdom</country>
  </CardTxn>
  <datacash_reference>...</datacash_reference>
  <floor_limit>020</floor_limit>
  <mode>...</mode>
  <reason>ACCEPTED</reason>
  <referral_telephone_number>MDIyMjQ0=</referral_telephone_number>
  <status>1</status>
  <time>...</time>
</Response>

```

Example XML Responses to Cancellation Requests

```
<Response>
  <datacash_reference>...</datacash_reference>
  <information>Too much time has elapsed between the original
            txn and the reversal request.</information>
  <merchantreference>...</merchantreference>
  <mode>...</mode>
  <reason reversal="not attempted">CANCELLED OK</reason>
  <status>1</status>
  <time>...</time>
</Response>

<Response>
  <datacash_reference>...</datacash_reference>
  <information>Reversal ACCEPTED status=1
            authcode=100100</information>
  <merchantreference>...</merchantreference>
  <mode>...</mode>
  <reason reversal="success">CANCELLED OK</reason>
  <status>1</status>
  <time>...</time>
</Response>
```

Example XML Response for an authorize_referral_request

```
<Response>
  <datacash_reference>...</datacash_reference>
  <merchantreference>...</merchantreference>
  <mode>...</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1071567356</time>
</Response>
```

2.6.2 Preallocated References

Additional non-technical information about this service is available on the website:
<http://www.datacash.com/gettingproducts.php?id=Pre-allocated-Reference-Transactions-Chip-PIN->

2.6.2.1 Schema Elements for Request

In this section the fields required to preallocate references will be given.

Preallocations

The information required to preallocate a DataCash reference is passed in two places in the schema:

- Request
 - Authentication – **section 2.1.1.2**
 - Transaction
 - TxnDetails – contains the merchants reference, **section 2.6.2.1.1**
 - CardTxn – contains the method `allocate_reference`

Performing Transactions

The information required to use a preallocated transaction is very similar to a normal transaction:

- Request
 - Authentication – **section 2.1.1.2**
 - Transaction
 - TxnDetails – **section 2.6.1.1.1**
 - CardTxn – **section 2.6.1.1.2**
 - Card – **section 2.6.1.1.5**
 - ICC – **section 2.6.1.1.4**
 - Terminal – **section 2.6.1.1.3**

Cancellations and Reversals

The `cancel` transaction type is the same as for the normal CHP transactions:

- Request
 - Authentication – **section 2.1.1.2**
 - Transaction
 - HistoricTxn – **section 2.3.3.1.9**

2.6.2.1.1 TxnDetails

These elements are only required for payments made with the allocated reference – they are not required for the `allocate_reference` transactions.

Element Name:	TxnDetails
Position(s)	Request.Transaction

Elements of TxnDetails		
Element Name	Description	Values / Limitations
allocated_reference	The <code>datacash_reference</code> returned from an <code>allocate_reference</code> transaction	16 digit number
amount	The value of the transaction	

Example XML Element TxnDetails for a preallocated reference transaction

```
<TxnDetails>
  <allocated_reference>3000100012345671</allocated_reference>
  <amount>100.00</amount>
</TxnDetails>
```

2.6.2.1.2 CardTxn

This section shows the fields required for the `allocate_reference` transactions. Please refer to section 0 for those required for `auth` and `creditcheck` transactions.

Element Name:	CardTxn
Position(s)	Request.Transaction

Elements of CardTxn		
Element Name	Description	Values / Limitations
method	The transaction type	allocate_reference

Example XML CardTxn element

```
<CardTxn>
  <method>allocate_reference</method>
</CardTxn>
```

2.6.2.2 Example Request

Example XML Request for allocate_reference

```
<Request>
  <Transaction>
    <TxnDetails>
      <merchantreference>123402</merchantreference>
    </TxnDetails>
    <CardTxn>
      <method>allocate_reference</method>
    </CardTxn>
  </Transaction>
</Request>
```

Example XML Request

```
<Request>
  <Authentication>...</Authentication>
  <Transaction>
    <TxnDetails>
      <allocated_reference>4800200040913907</allocated_reference>
      <amount currency="GBP">54.00</amount>
      <capturemethod>parked</capturemethod>
    </TxnDetails>
    <CardTxn>
      <Terminal id="82000062">
        <terminal_capabilities ic_reader="true"
          magnetic_stripe_reader="true" manual_card_entry="true"/>
        <features_capabilities pin_pad_available="true"/>
      </Terminal>
      <card_details type="track2_data">NDkyOTQ5ODMxMTQwMDAwMj
        0wODAzMjAxMDA50DgwMDAwMR8wMA==</card_details>
      <ICC>
        <reason_online_code>10</reason_online_code>
        <seq_or_issue_number>00</seq_or_issue_number>
        <crypto_tran_type>00</crypto_tran_type>
        <txn_currency_code>826</txn_currency_code>
        <term_country_code>826</term_country_code>
        <arqc>ED128CAB559EC54C</arqc>
        <aip>5C00</aip>
        <atc>0CF3</atc>
        <unpredictable>81B72731</unpredictable>
        <tvr>0000008000</tvr>
        <issuer_app_data>06FE0A03A4A800</issuer_app_data>
        <crypto_info_data>80</crypto_info_data>
        <cvm>410302</cvm>
        <aid>A0000000031010</aid>
        <term_app_ver_no>0084</term_app_ver_no>
        <term_capabilities>7546</term_capabilities>
      </ICC>
      <method>creditcheck</method>
    </CardTxn>
  </Transaction>
</Request>
```

2.6.2.3 Schema Elements for Response

The response to transactions performed using this service contains no fields beyond the CHP response elements, covered in section 2.1.1.3

2.6.2.4 Example Response

Example XML Request for an allocate_reference request

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>3000100012345671</datacash_reference>
    <merchantreference>123402</merchantreference>
    <mode>LIVE</mode>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <time>1107190133</time>
</Response>
```

[Back to Contents](#)

2.7 OTHER DPG SERVICES

2.7.1 PayPal

This service allows you to process Express Checkout and Auth and Capture PayPal transactions via your DataCash account. Successful payments can be refunded back to the customer's PayPal account. The service can be seamlessly integrated into your systems, enabling your customers and Customer Service teams to experience fast and efficient processing and management of transactions.

An introduction to this service, including the transaction types, is available on the website:

<http://www.datacash.com/gettingproducts.php?id=PayPal---Express-Checkout>

2.7.1.1 Schema Elements for Request

In this section the required fields for each transaction type will be presented, along with example XML for those fields.

The data for the PayPal Service is passed in these distinct places in the schema:

- Request
- Authentication – section 2.1.1.2
- Transaction
- TxnDetails – section 2.6.1.1.1
- PayPalTxn – this section contains all of the details specific to the PayPal transaction, see section 2.7.1.1.2
- Items – details of all items in the order, section 2.7.1.1.4
- Item – details of each item within the order, section 2.7.1.1.5
- ShippingAddress – details of the shipping address, section 2.7.1.1.6
- AirlineItineraryData – details of flights for Airline transactions, section 2.7.1.1.7
- FlightDetails – details of flight legs, section 2.7.1.1.8

2.7.1.1.1 TxnDetails

Note that most fields are only available for specific transaction types.

- S – set_express_checkout
- G – get_express_checkout_details
- D – do_express_checkout_payment
- X – txn_refund, used to perform refunds of existing payments
- A – do_authorization
- C – do_capture
- CB – create_billing_agreement
- UB – update_billing_agreement
- DR – do_reference_transaction

Element Name: TxnDetails
Position(s) Request.Transaction

Elements of TxnDetails											
Element Name	Description	Values / Limitations	Required								
			S	G	D	X	A	C	CB	UB	DR
amount	The value of the transaction	May be '0' only for D , where a Billing Agreement is to be created with no initial payment taken. For DR it may be '0' for S if no initial payment is to be taken.	R	-	R	O	R	R	-	-	R
merchantreference	A unique reference number for each transaction	Minimum 6, maximum 30 alphanumeric characters . Must be unique	R	R	R	O	-	-	R	R	R

For a `txn_refund`, an amount may be specified if the refund is for a smaller amount than the original transaction. The value must be less than or equal to the value of the original transactions, minus any previous partial refunds.

Attributes of elements of TxnDetails						
Attribute name	Attribute of Element	Description	Values / Limitations			
currency	amount	The currency for the payment	AUD CAD CHF CZK	DKK EUR GBP HKD	HUF JPY NOK NZD	PLN SEK SGD USD

May not be specified for `txn_refund`, `do_authorization` or `do_capture`

Example XML for TxnDetails element

```

<TxnDetails>
    <amount currency="GBP">19.26</amount>
    <merchantreference>123ABC</merchantreference>
</TxnDetails>

<TxnDetails>
    <merchantreference>46548tretr</merchantreference>
</TxnDetails>

```

2.7.1.1.2 PayPalTxn

All data specifically relating to PayPal transactions are submitted in this element and associated sub-elements. Note that most fields are only available for specific transaction types.

- S – `set_express_checkout`
- G – `get_express_checkout_details`
- D – `do_express_checkout_payment`
- R – `txn_refund`
- A – `do_authorization`
- C – `do_capture`
- V – `do_void`
- CB – `create_billing_agreement`
- UB – `update_billing_agreement`
- DR – `do_reference_transaction`

Element Name: PayPalTxn
Position(s) Request.Transaction

Elements of PayPalTxn

Element Name	Description	Values / Limitations	Required									
			S	G	D	R	A	C	V	C B	UB	D R
cancel_url	URL to which the customer is returned if (s)he does not approve the use of PayPal	URL should be URL-encoded	R	-	-	-	-	-	-	-	-	-
method	The transaction type	set_express_checkout get_express_checkout_details do_express_checkout_payment txn_refund do_authorization do_capture do_void create_billing_agreement update_billing_agreement do_reference_transaction	R	R	R	R	R	R	R	R	R	R
reference	The 16 digit datacash_reference, referring to a previous successfully processed transaction	D uses ref from S G uses ref from S R uses ref from either D or C A uses ref from D C uses ref from A V uses ref from A (or D when payment_action is order) CB uses ref from S UB uses ref from D or CB DR uses ref from D or CB A and S use ref from D or DB	-	R	R	R	R	R	R	R	R	R

return_url	URL to which the customer's browser is returned. Recommended to be the final review page on which the customer confirms the order and payment/billing agreement	URL should be URL-encoded	R	-	-	-	-	-	-	-	-	-	-
billing_type	Type of billing agreement.	MerchantInitiatedBilling - creates a billing agreement even if there is an existing billing agreement MerchantInitiatedBillingSingleSource - creates a billing agreement only if there is no existing billing agreement with the buyer	R	-	-	-	-	-	-	-	-	-	-
billing_agreement_status	The new/current status of the billing agreement	Active Cancelled Note: an agreement which has a status of Cancelled can not be further updated.	-	-	-	-	-	-	-	-	O	-	-
billing_agreement_description	Description of goods or services associated with the billing agreement	127 single-byte characters	O	-	-	-	-	-	-	-	O	-	-
billing_agreement_custom	Custom annotation field for your own use	255 single-byte characters	O	-	-	-	-	-	-	-	O	-	-
buttonsource	An identification code for use by third party applications to identify transactions	32 single-byte alphanumeric characters	-	-	O	-	-	-	-	-	-	-	-
channel_type	Type of channel	Merchant – non-auction seller eBayItem - ebay auction	O	-	-	-	-	-	-	-	-	-	-

completed	Indicates the final do_capture being processed against a successful do_authorization, thus completing the authorisation	Yes	-	-	-	-	-	O	-	-	-	-	-	-	-	-	-	-
custom	A free-form field for your own use, such as a tracking number or other value you want PayPal to return on get_express_checkout_details and do_express_checkout_payment	255 single-byte alphanumeric characters	O	-	O	-	-	-	-	-	-	-	-	-	-	-	O	-
description	Description of the items the customer is purchasing	127 single-byte characters	O	-	O	-	-	-	-	-	-	-	-	-	-	-	O	-
email	Email address of the buyer. Used to pre-fill the PayPal membership sign-up portion of the PayPal login page		O	-	-	-	-	-	-	-	-	-	-	-	-	-	O	-
handling_total	Total handling costs for this order		-	-	O	-	-	-	-	-	-	-	-	-	-	-	O	-
header_style	Defines the image, border colour and background colours for the image in the header of the payment page. Header space is 750px wide by 90 px high.	See attributes table below	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

invnum	Your own unique invoice or tracking number. Is returned in the do_express_checkout_payment response		O	-	O	-	-	O	-	-	-	O
Items	Please refer to section 2.7.1.1.5	Alphanumerics - (dash) * (asterisk) . (period) {space}	-	-	O	-	-	-	-	-	-	-
item_total	Sum of cost of all items in this order	Numeric with two decimal places. Must not exceed \$10,000 in any currency. Max value 9,999,999.99	-	-	O	-	-	-	-	-	-	O

localecode	<p>Locale of pages displayed by PayPal during checkout</p>	<ul style="list-style-type: none"> • AU – Australia • AT – Austria • BE – Belgium • BR – Brazil • CA – Canada • CH – Switzerland • CN – China • DE – Germany • ES – Spain • GB – United Kingdom • FR – France • IT – Italy • NL – Netherlands • PL – Poland • PT – Portugal • RU – Russia • US – United States <p>The following 5-character codes are also supported for languages in specific countries:</p> <ul style="list-style-type: none"> ○ da_DK – Danish (for Denmark only) ○ he_IL – Hebrew (all) ○ id_ID – Indonesian (for Indonesia only) ○ jp_JP – Japanese (for Japan only) ○ no_NO – Norwegian (for Norway only) ○ pt_BR – Brazilian Portuguese (for Portugal and Brazil only) ○ ru_RU – Russian (for Lithuania, Latvia, and Ukraine only) ○ sv_SE – Swedish (for Sweden only) ○ th_TH – Thai (for Thailand only)
------------	--	--

		<ul style="list-style-type: none"> ○ tr_TR – Turkish (for Turkey only) ○ zh_CN – Simplified Chinese (for China only) ○ zh_HK – Traditional Chinese (for Hong Kong only) ○ zh_TW – Traditional Chinese (for Taiwan only) <p>Character length and limitations: 2-character country code</p> <p>For the most current list, please refer to the following link:</p> <p>https://cms.paypal.com/us/cgi-bin/?cmd=_render-content&content_ID=developer/e_howto_api_nvp_r_SetExpressCheckout</p>									
max_amount	The expected maximum total amount of the complete order, including shipping and tax	Numeric with two decimal places. Must not exceed \$10,000 in any currency. Max value 9,999,999.99	O	-	-	-	-	-	-	-	-
no_shipping	Indicates whether the shipping address fields should be displayed on the PayPal pages	0 - fields displayed (default) 1 - fields not displayed	O	-	-	-	-	-	-	-	-

note	Custom memo about the refund	255 single-byte alphanumeric characters 22 single-byte characters comprised of: Alphanumerics - (dash) * (asterisk) . (period) {space}	-	-	-	O	-	O	O	-	-	-	-	-	-	-	-
notify_url	Your URL for receiving Instant Payment Notification (IPN) about this transaction	2048 single-byte alphanumeric characters	-	-	O	-	-	-	-	-	-	-	-	-	-	O	
override_address	Indicates whether the shipping address fields (as set by you) should be displayed on the PayPal pages, rather than the shipping address details held on file at PayPal	0 – displays details held on file (default) 1 – displays details set by you	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-
payment_action	Specifies the type of PayPal service to use 'sale' - will use Express Checkout instant payment (default) 'order' - will use Auth Capture	sale (default) order order must be supplied during S for Auth and Capture txns. An order may be converted to a sale during the D txn. For Auth and Capture must be order for S may be order for D , or sale if wishing to convert the existing txn	O	-	O	-	-	-	-	-	-	-	-	-	-	O	
page_style	Sets the Custom Payment Page Style of the payment pages. This corresponds to the HTML variable page_style for customising payment pages	Value must be equal to the Page Style Name as chosen by you when adding/editing the page style from the My Profile → Profile subtab of your PayPal Account	O	-	-	-	-	-	-	-	-	-	-	-	-	-	-

payment_type	Specifies type of PayPal payment you require for the billing agreement	Any InstantOnly	O - - - - - - - - - - - - - - - - - -
req_billing_address	Specifies whether you require the customers billing address from PayPal. This is returned the <code>get_express_checkout_details</code> response	0 – Billing address not required (default) 1 – Billing address required	O - - - - - - - - - - - - - - - - - -
req_confirmed_shipping	Indicates whether the customer's shipping address on file with PayPal must be a confirmed address	0 – need not be confirmed (default) 1 – must be confirmed	O - - - - - - - - - - - - - - - - - -
shipping_total	Total shipping cost for this order		- - O - - - - - - - - - - - O
soft_descriptor	Per transaction description of the payment that is passed to the consumer's credit card	255 single-byte alphanumeric characters 22 single-byte characters comprised of: Alphanumerics - (dash) * (asterisk) . (period) {space}	- - - - - - - O - - - - - O
solution_type	Type of checkout flow	Sole – for Express Checkout auctions Mark – normal Express Checkout	O - - - - - - - - - - - - - - - - - -

tax_total	Sum of tax for all items in this order	255 single-byte alphanumeric characters 22 single-byte characters comprised of: Alphanumeric - (dash) * (asterisk) . (period) {space}	-	-	O	-	-	-	-	-	-	-	O
token	Time stamped token – token expires after three hours Normally this is generated by PayPal and does not need to be supplied.	20 single-byte characters max	O	-	-	-	-	-	-	-	-	-	-
ShippingAddress	Please refer to section 2.7.1.1.6	255 single-byte alphanumeric characters 22 single-byte characters comprised of: Alphanumeric - (dash) * (asterisk) . (period) {space}	-	-	O	-	-	-	-	-	-	-	-
AirlineItineraryData	Please refer to section 2.7.1.1.7	255 single-byte alphanumeric characters 22 single-byte characters comprised of: Alphanumeric - (dash) * (asterisk) . (period) {space}	-	-	-	-	-	O	-	-	-	-	O

Attributes for Elements of PayPalTxn

Attribute Name	Attribute of element	Value / limitations	Required
bgcolor	header_style	Background colour for the payment page header. Six character HTML hexadecimal colour code in ASCII.	O
bordercolor	header_style	Sets the colour of the 2px border around the header of the payment page. Six character HTML hexadecimal colour code in ASCII.	O
img	header_style	URL of the image to appear on the top left of the payment page. Max image size 750px wide by 90px high. Recommended to be HTTPS link. 127 single-byte characters max, URL-encoded.	O

Example XML for PayPal elements for get_express_checkout_details

```
<PayPalTxn>
  <method>get_express_checkout_details</method>
  <reference>4000900012345671</reference>
</PayPalTxn>
```

stylename	page_style	Used to select the style of checkout pages. Styles are configurable in PayPal account settings.	O
-----------	------------	---	---

Example XML for PayPal elements

```
<PayPalTxn>
  <method>set_express_checkout</method>
  <return_url>https://www.example.com/myshoppingcard?action=complete&sale_id=xyz123</return_url>
  <cancel_url>https://www.example.com/myshoppingcard?action=cancel&sale_id=xyz123</cancel_url>
  <payment_action>order</payment_action>
</PayPalTxn>
```

Example XML for PayPal elements for do_express_checkout_payment

```
<PayPalTxn>
  <method>do_express_checkout_payment</method>
  <reference>4000900012345671</reference>
</PayPalTxn>

<PayPalTxn>
  <method>do express checkout payment</method>
  <reference>4000900012345671</reference>
  <description>Twenty NeverFail(tm) Widgets</description>
  <custom>UPS_ID=12345678</custom>
  <invnum>abc123</invnum>
  <buttonsource>foobar</buttonsource>
  <notify_url>http://www.example.com/mynotifyurl?id=abc123
    </notify_url>
  <item_total>85.00</item_total>
  <shipping_total>10.00</shipping_total>
  <handling_total>5.00</handling_total>
  <tax_total>11.97</tax_total>
  <Items>...</Items>
  <ShippingAddress>...</ShippingAddress>
</PayPalTxn>

<PayPalTxn>
  <method>do_express_checkout_payment</method>
  <reference>4000900012345671</reference>
  <Items>...</Items>
</PayPalTxn>
```

Example XML for PayPal elements for txn_refund

```
<PayPalTxn>
  <method>txn_refund</method>
  <reference>4000900012345671</reference>
</PayPalTxn>

<PayPalTxn>
  <method>txn_refund</method>
  <reference>4000900012345671</reference>
  <note>Refund of half the order, as promised</note>
</PayPalTxn>
```

Example XML for PayPal elements for do_authorization

```
<PayPalTxn>
  <method>do_authorization</method>
  <reference>4000900012345671</reference>
</PayPalTxn>
```

Example XML for PayPal elements for do_capture

```
<PayPalTxn>
  <method>do_capture</method>
  <reference>4000900012345671</reference>
</PayPalTxn>

<PayPalTxn>
  <method>do_capture</method>
  <reference>4000900012345671</reference>
  <invnum>inv1234</invnum>
  <note>Finally, I have captured some funds</note>
  <soft_descriptor>FooCorp LLC</soft_descriptor>
</PayPalTxn>

<PayPalTxn>
  <method>do_capture</method>
  <reference>4000900012345671</reference>
  <completed>yes</completed>
  <invnum>inv1234</invnum>
  <note>This is the last capture for this
  authorization</note>
  <soft_descriptor>FooCorp LLC</soft_descriptor>
</PayPalTxn>
```

Example XML for PayPal elements for do_void

```
<PayPalTxn>
  <method>do_void</method>
  <reference>4000900012345671</reference>
</PayPalTxn>

<PayPalTxn>
  <method>do_void</method>
  <reference>4000900012345671</reference>
  <note>Excess funds released</note>
</PayPalTxn>
```

Example XML for PayPal elements for create_billing_agreement

```
<PayPalTxn>
  <method>create_billing_agreement</method>
  <reference>4000900012345671</reference>
</PayPalTxn>
```

Example XML for PayPal elements for update_billing_agreement

```
<PayPalTxn>
  <method>update_billing_agreement</method>
  <reference>4000900012345671</reference>
  <billing_agreement_status>Cancelled
  </billing_agreement_status>
</PayPalTxn>

<PayPalTxn>
  <method>update_billing_agreement</method>
  <reference>4000900012345671</reference>
  <billing_agreement_description>Subscription to FOO
  Magazine</billing_agreement_description>
  <billing_agreement_custom>Part of our CHEAP09
  promotion</billing_agreement_custom>
</PayPalTxn>
```

2.7.1.1.3

Example XML for PayPal elements for do_reference_transaction

```

<PayPalTxn>
  <method>do_reference_transaction</method>
  <reference>4000900012345671</reference>
  <description>Twenty NeverFail(tm) Widgets</description>
  <custom>UPS_ID=12345678</custom>
  <invnum>abc123</invnum>
  <buttonsource>foobar</buttonsource>
  <notify_url>http://www.example.com/mynotifyurl?id=abc123
    </notify_url>
  <item_total>85.00</item_total>
  <shipping_total>10.00</shipping_total>
  <handling_total>5.00</handling_total>
  <tax_total>11.97</tax_total>
  <Items>...</Items>
  <ShippingAddress>...</ShippingAddress>
</PayPalTxn>

```

2.7.1.1.4 Items

The Items element is optional. It contains an Item child element for each individual product or service purchased.

Element Name: Items
 Position(s) Request.Transaction.PayPalTxn

Elements of Items						
Element Name	Description	Values / Limitations	Required			
			S	G	D	R
Item	See section 2.7.1.1.4		-	-	O	- O

Example XML for Items element

```

<Items>
  <Item id="0">...</Item>
  <Item id="1">...</Item>
  <Item id="2">...</Item>
  <Item id="3">...</Item>
</Items>

```

2.7.1.1.5 Item

The Item elements contain details of each product within the order.

Element Name:	Item
Position(s)	Request.Transaction.PayPalTxn.Items

Element Name	Description	Values / Limitations	Required				
			S	G	D	R	DR
amount	Cost of item	Must not exceed \$10,000 in any currency. Max value 9,999,999.99	-	-	O	-	O
ebay_item_auction_txnid	Auction transaction identification number	225 single-byte characters	-	-	O	-	O
ebay_item_number	Auction item number	765 single-byte characters	-	-	O	-	O
ebay_item_order_id	Auction identification number	64 single-byte characters	-	-	O	-	O
name	The item name	127 single-byte characters	-	-	O	-	O
number	The item number		-	-	O	-	O
quantity	The item quantity	Positive integer	-	-	O	-	O
taxamt	Item sales tax	Must not exceed \$10,000 in any currency. Max value 9,999,999.99	-	-	O	-	O

Attributes for Elements of Item			
Attribute Name	Attribute of element	Value / Limitations	Required
id	Item	Must start from 0 and be contiguous	R

Example XML for Item elements for do_express_checkout_payment

```
<Item id="0">
  <name>Widgets</name>
  <number>1230212-A</number>
  <quantity>20</quantity>
  <amount>73.03</amount>
  <taxamt>11.97</taxamt>
</Item>
```

Example XML for Item elements for do_express_checkout_payment

```
<Item id="0">
  <ebay_item_number>9988775544</ebay_item_number>
  <ebay_item_auction_txn_id>123456123</ebay_item_auction_txn_id>
  <ebay_item_order_id>abc123def890</ebay_item_order_id>
</Item>
```

2.7.1.1.6 ShippingAddress

This parent element is used to present information about the shipping address. While the parent element itself is optional, if it is sent certain elements must be provided.

Element Name:	ShippingAddress
Position(s)	Request.Transaction.PayPalTxn

Elements of ShippingAddress			
Element Name	Description	values / limitations	Required
city	The name of the city	40 single-byte characters max	R
country_code	The country code, as defined in ISO 3166-1	2 character	R
name	The person's name associated with the shipping address	32 single-byte characters max	R
postcode	UK postcode, US ZIP code or other country-specific postal code	20 single-byte characters max	R
region	The state, province or region	40 single-byte characters max	R for US addresses only
street_address1	The first line of the address	100 single-byte characters max	R
street_address2	The second line of the address	100 single-byte characters max	O
telephone_number	Phone number	20 single-byte characters max	O

Example XML for ShippingAddress elements

```
<ShippingAddress>
  <name>Joe Bloggs</name>
  <street_address1>3 Fish Street</street_address1>
  <street_address2>Castle Street</street_address2>
  <city>Hull</city>
  <country_code>GB</country_code>
  <postcode>HU1 1AA</postcode>
  <telephone_number>01234 345 6789</telephone_number>
</ShippingAddress>

<ShippingAddress>
  <name>John Smith</name>
  <street_address1>144 Main Street</street_address1>
  <city>San Jose</city>
  <region>CA</region>
  <country_code>US</country_code>
  <postcode>99221</postcode>
</ShippingAddress>
```

2.7.1.1.7 AirlineItineraryData

This parent element is used to present flight data for Airline transactions. While this element itself is optional, if it is sent then certain elements must be provided.

This element is supported for the following transaction types, depending on the value of the `payment_action` element.

- D – `do_express_checkout_payment`
- C – `do_capture`

Where the `payment_action` of a transaction is sale, Airline data should be provided in the `do_express_checkout_payment` request.

Where the `payment_action` is order, Airline Data should be provided in the `do_capture` request.

Element Name: AirlineItineraryData
Position(s) Request.Transaction.PayPalTxn

Elements of AirlineItineraryDetails				
Element Name	Description	Values / Limitations	Required	
			D	C
clearing_count	Numeric code to identify each clearing record message in those cases where multiple clearing messages are allowed per authorized transaction. Applicable to multiple captures against an authorization. In the case of single capture against an authorization, the value should be 1	Positive integer	R	R
clearing_sequence	Numeric code to identify each clearing record message in cases where multiple clearing messages are allowed per authorised transaction. Applicable to multiple captures against an authorisation. In the case of single capture against an authorisation, the value should be 1	Positive integer	R	R
FlightDetails	Please refer to section 2.7.1.1.8			- R

issuing_carrier_code	Airline code for the airline issuing the ticket. Obtain the airline code from the official <i>Airline Guide</i> or its equivalent	4 single-byte characters max	R	R
passenger_name	Name of the passenger	25 single-byte characters max	R	R
restricted_ticket	Indicated whether the ticket is restricted (refundable)	0 – Ticket is not restricted 1 – Ticket is restricted	R	R
ticket_number	The ticket number. If multiple tickets are purchased with one transaction, you should provide the primary ticket number	16 single-byte characters max	R	R
customer_code	A code that the cardholder supplied to you. Can be used for passing in the frequent flyer number of the customer	17 single-byte characters max	O	O
issue_date	Date of issue recorded in the airline system. In case of multiple issuances of the same ticket, use the last ticket date	YYYYMMDD format	O	O
total_fare	Total fare for all legs on this ticket, excluding taxes and fees	Numeric with two decimal places. Must not exceed \$10,000 in any currency. Max value 9,999,999.99	O	O
total_fee	Total fee for all legs on the ticket		O	O
total_taxes	Total taxes for all legs on the ticket		O	O
travel_agency_code	The travel agency code. In an integration by the airline, this is the airline code from the official <i>Airline Guide</i> or its equivalent	8 single-byte characters max	O	O
travel_agency_name	Name of the travel agency issuing the ticket. In an integration by an airline, this is the airline name	25 single-byte characters max	O	O

Example XML for AirlineItineraryData elements

```
<AirlineItineraryData>
  <passenger_name>Joseph Bloggs</passenger_name>
  <issue_date>20090701</issue_date>
  <travel_agency_name>Super Agents</travel_agency_name>
  <travel_agency_code>SUPER1</travel_agency_code>
  <ticket_number>BLAH1234567890</ticket_number>
  <issuing_carrier_code>ABCD</issuing_carrier_code>
  <customer_code>JBLOGGS09713</customer_code>
  <total_fare>85.00</total_fare>
  <total_taxes>10.00</total_taxes>
  <total_fee>5.00</total_fee>
  <restricted_ticket>0</restricted_ticket>
  <clearing_sequence>1</clearing_sequence>
  <clearing_count>1</clearing_count>
  <FlightDetails leg_id="0">
    ...
  </FlightDetails>
  <FlightDetails leg_id="1">
    ...
  </FlightDetails>
</AirlineItineraryData>

<AirlineItineraryData>
  <passenger_name>Joseph Bloggs</passenger_name>
  <ticket_number>BLAH1234567890</ticket_number>
  <issuing_carrier_code>ABCD</issuing_carrier_code>
  <restricted_ticket>0</restricted_ticket>
  <clearing_sequence>1</clearing_sequence>
  <clearing_count>1</clearing_count>
  <FlightDetails leg_id="0">
    ...
  </FlightDetails>
</AirlineItineraryData>
```

2.7.1.1.8 FlightDetails

The FlightDetails elements contain details of each leg of the trip.

Element Name:	FlightDetails
Position(s)	Request.Transaction.PayPalTxn.AirlineItineraryData

Elements of FlightDetails				
Element Name	Description	Values / Limitations	Required	
			D	C
arrival_airport_code	The arrival airport code. Obtain the code from the official <i>Airline Guide</i> or its equivalent	5 single-byte characters max	R	R
carrier_code	Standard abbreviation for airline carrier. Obtain the code from the official <i>Airline Guide</i> or its equivalent	2 single-byte characters	R	R
departure_airport_code	The departure airport code. Obtain the code from the official <i>Airline Guide</i> or its equivalent	5 single-byte characters max	R	R
departure_time	The departure time in local time at the departure airport	HH:MM format, between 00:00 and 23:59	R	R
fare_basis_code	The alphanumeric code that carriers assign to a particular ticket type, such as business class, discounted, or non-refundable	15 single-byte characters max	R	R
flight_number	The flight number assigned by the airline carrier	5 single-byte characters max	R	R
service_class	The type of service; for example, first class or coach. Obtain the service class from the official <i>Airline Guide</i> or equivalent	2 single-byte characters	R	R
stopover_code	A code indicating a non-direct flight or route on the same ticket number	0 – Stopover not permitted 1 – Stopover permitted	R	R
travel_date	The date of travel in local time at the departure airport	YYYYMMDD format	R	R

conjunction_ticket	Ticket issued to a passenger in conjunction with another ticket that constitutes a single contract of carriage	14 single-byte characters max	O	O
coupon_number	The coupon number associated with this leg of the trip. A ticket can contain several legs of travel, and each leg of travel requires a separate coupon	1 single-byte character	O	O
endorsement_or_restrictions	An endorsement can be an agency-added notation or a mandatory government required notation such as value added tax. A restriction is a limitation based on the type of fare such as a ticket with a non-refundable or 3-day minimum stay.	20 single-byte characters max	O	O
exchange_ticket	New ticket number that is issued when a ticket is exchanged	15 single-byte characters max	O	O
fare	Amount of the ticket for this leg of the trip excluding taxes and fees	Numeric with two decimal places. Must not exceed \$10,000 in any currency. Max value 9,999,999.99	O	O
fee	Fee amount for this leg of the trip		O	O
taxes	Amount of the taxes for this leg of the trip		O	O

Attributes for Elements of FlightDetails			
Attribute Name	Attribute of element	Value / limitations	Required
leg_id	FlightDetails	Must start from 0 and be contiguous	R

Example XML for FlightDetails elements

```

<FlightDetails leg_id="0">
  <conjunction_ticket>CONTKT12345678</conjunction_ticket>
  <exchange_ticket>EXTKT123456789</exchange_ticket>
  <coupon_number>X</coupon_number>
  <service_class>A1</service_class>
  <travel_date>20090801</travel_date>
  <carrier_code>AB</carrier_code>
  <stopover_code>0</stopover_code>
  <departure_airport_code>EDI</departure_airport_code>
  <arrival_airport_code>LHR</arrival_airport_code>
  <flight_number>FOO12</flight_number>
  <departure_time>12:34</departure_time>
  <arrival_time>13:37</arrival_time>
  <fare basis code>CHEAP1</fare basis code>
  <fare>15.00</fare>
  <taxes>5.00</taxes>
  <fee>2.00</fee>
  <endorsement_or_restrictions>non-
  refundable</endorsement_or_restrictions>
</FlightDetails>

<FlightDetails leg_id="1">
  <service_class>A1</service_class>
  <travel date>20090801</travel date>
  <carrier_code>AB</carrier_code>
  <stopover_code>0</stopover_code>
  <departure_airport_code>EDI</departure_airport_code>
  <arrival_airport_code>LHR</arrival_airport_code>
  <flight_number>FOO12</flight_number>
  <departure_time>12:34</departure_time>
  <fare_basis_code>CHEAP1</fare_basis_code>
</FlightDetails>

```

2.7.1.2 XML Example Requests

This section provides full XML examples for each transaction type.

2.7.1.2.1 set_express_checkout

Example XML Request

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">100.00</amount>
            <merchantreference>12345601</merchantreference>
        </TxnDetails>
        <PayPalTxn>
            <method>set_express_checkout</method>
            <return_url>https://www.example.com/myshoppingcard?
action=complete&sale_id=xyz123</return_url>
            <cancel_url>https://www.example.com/myshoppingcard?
action=cancel&sale_id=xyz123</cancel_url>
            <email>homer@example.com</email>
            <max_amount>1000.00</max_amount>
            <description>Twenty NeverFail(tm) Widgets</description>
            <custom>UPS_ID=12345678</custom>
            <invnum>abc123</invnum>
            <req_confirmed_shipping>0</req_confirmed_shipping>
            <no_shipping>0</no_shipping>
            <override_address>0</override_address>
            <localecode>GB</localecode>
            <header_style img="https://www.example.com/header.png"
bordercolor="black" bgcolor="red"/>
            <page_style stylename="My Style" bgcolor="white"/>
            <ShippingAddress>
                <name>Test User</name>
                <street_address1>1 Main Terrace</street_address1>
                <street_address2>Highgate</street_address2>
                <city>Wolverhampton</city>
                <region>West Midlands</region>
                <country_code>GB</country_code>
                <postcode>E12 4LQ</postcode>
                <telephone_number>0121 231 3122</telephone_number>
            </ShippingAddress>
        </PayPalTxn>
    </Transaction>
</Request>
```

Example XML for Request

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">100.00</amount>
            <merchantreference>12345601</merchantreference>
        </TxnDetails>
        <PayPalTxn>
            <method>set_express_checkout</method>
            <return_url>https://www.example.com/myshoppingcard?
                action=complete&sale_id=xyz123</return_url>
            <cancel_url>https://www.example.com/myshoppingcard?
                action=cancel&sale_id=xyz123</cancel_url>
            <header_style img="https://www.example.com/header.png"
                bordercolor="black" bgcolor="red"/>
            <page style stylename="My Style" bgcolor="white"/>
        </PayPalTxn>
    </Transaction>
</Request>
```

Example XML for Request

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">100.00</amount>
            <merchantreference>12345601</merchantreference>
        </TxnDetails>
        <PayPalTxn>
            <method>set_express_checkout</method>
            <return_url>https://www.example.com/myshoppingcard?
                action=complete&sale_id=xyz123</return_url>
            <cancel_url>https://www.example.com/myshoppingcard?
                action=cancel&sale_id=xyz123</cancel_url>
            <channel_type>Merchant</channel_type>
            <solution_type>Mark</solution_type>
            <billing_type>MerchantInitiatedBilling</billing_type>
            <billing_agreement_description>Delivery by 10pm
                Tuesday</billing_agreement_description>
            <billing_agreement_custom>Remember to tell
                Bob</billing_agreement_custom>
            <payment_type>InstantOnly</payment_type>
        </PayPalTxn>
    </Transaction>
</Request>
```

2.7.1.2.2 get_express_checkout_details

Example XML for Request

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <merchantreference>46548tretr</merchantreference>
        </TxnDetails>
        <PayPalTxn>
            <method>get_express_checkout_details</method>
            <reference>4000900012345671</reference>
        </PayPalTxn>
    </Transaction>
</Request>
```

2.7.1.2.3 do_express_checkout_payment

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">100.00</amount>
            <merchantreference>12345601</merchantreference>
        </TxnDetails>
        <PayPalTxn>
            <method>do_express_checkout</method>
            <reference>4000900012345671</reference>
            <Items>
                <Item id="0">
                    <ebay_item_number>9988775544</ebay_item_number>
                    <ebay_item_auction_txn_id>123456123
                    </ebay_item_auction_txn_id>
                    <ebay_item_order_id>abc123def890</ebay_item_order_id>
                </Item>
            </Items>
            <ShippingAddress>
                <name>Test User</name>
                <street_address1>1 Main Terrace</street_address1>
                <city>Wolverhampton</city>
                <country_code>GB</country_code>
                <postcode>E12 4LQ</postcode>
            </ShippingAddress>
        </PayPalTxn>
    </Transaction>
</Request>
```

Example XML for Request

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">100.00</amount>
      <merchantreference>12345601</merchantreference>
    </TxnDetails>
    <PayPalTxn>
      <method>do_express_checkout_payment</method>
      <reference>4000900012345671</reference>
    </PayPalTxn>
  </Transaction>
</Request>
```

Example XML for Request

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">100.00</amount>
      <merchantreference>12345601</merchantreference>
    </TxnDetails>
    <PayPalTxn>
      <method>do_express_checkout_payment</method>
      <reference>4000900012345671</reference>
      <description>Twenty NeverFail(tm) Widgets</description>
      <custom>UPS_ID=12345678</custom>
      <invnum>abc123</invnum>
      <buttonsource>foobar</buttonsource>
      <notify_url>http://www.example.com/mynotifyurl?id=
      abc123</notify_url>
      <item_total>85.00</item_total>
      <shipping_total>10.00</shipping_total>
      <handling_total>5.00</handling_total>
      <tax_total>11.97</tax_total>
    </PayPalTxn>
  </Transaction>
</Request>
```

Example XML for Request

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">100.00</amount>
            <merchantreference>12345601</merchantreference>
        </TxnDetails>
        <PayPalTxn>
            <method>do_express_checkout_payment</method>
            <reference>4000900012345671</reference>
            <AirlineItineraryData>
                <passenger_name>Joseph Bloggs</passenger_name>
                <ticket_number>BLAH1234567890</ticket_number>
                <issuing_carrier_code>ABCD</issuing_carrier_code>
                <restricted_ticket>0</restricted_ticket>
                <clearing_sequence>1</clearing_sequence>
                <clearing_count>1</clearing_count>
                <FlightDetails leg_id="0">
                    <service_class>A1</service_class>
                    <travel_date>20090801</travel_date>
                    <carrier_code>AB</carrier_code>
                    <stopover_code>0</stopover_code>
                    <departure_airport_code>EDI</departure_airport_code>
                    <arrival_airport_code>LHR</arrival_airport_code>
                    <flight number>FO012</flight number>
                    <departure_time>12:34</departure_time>
                    <fare_basis_code>CHEAP1</fare_basis_code>
                </FlightDetails>
            </AirlineItineraryData>
        </PayPalTxn>
    </Transaction>
</Request>
```

Example XML for Request without initial payment to be processed via the PayPal Auth Capture Service

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">0</amount>
            <merchantreference>12345601</merchantreference>
        </TxnDetails>
        <PayPalTxn>
            <method>set_express_checkout</method>
            <return url>https://www.example.com/myshoppingcard?
action=complete&sale_id=xyz123</return_url>
            <cancel_url>https://www.example.com/myshoppingcard?
action=cancel&sale_id=xyz123</cancel_url>
            <billing_type>MerchantInitiatedBilling</billing_type>
            <billing_type>MerchantInitiatedBilling</billing_type>
            <billing_agreement_description>Subscription to
Magazine</billing_agreement_subscription>
        </PayPalTxn>
    </Transaction>
</Request>
```

2.7.1.2.4

Example XML for Request with initial payment of £45.99 to be processed via the PayPal Auth Capture Service

```
<?xml version="1.0" encoding="UTF-8"?>

<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">45.99</amount>
            <merchantreference>12345601</merchantreference>
        </TxnDetails>
        <PayPalTxn>
            <method>set_express_checkout</method>
            <return_url>https://www.example.com/myshoppingcard?
action=complete&sale_id=xyz123</return_url>
            <cancel_url>https://www.example.com/myshoppingcard?
action=cancel&sale_id=xyz123</cancel_url>
            <billing_type>MerchantInitiatedBillingSingleSource</billing_type>
        </PayPalTxn>
    </Transaction>
</Request>
```

2.7.1.2.6

Example XML for Request with initial payment of £45.99 to be processed via the PayPal Auth Capture Service

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">45.99</amount>
            <merchantereference>12345601</merchantereference>
        </TxnDetails>
        <PayPalTxn>
            <method>set_express_checkout</method>
            <return_url>https://www.example.com/myshoppingcard?
action=complete&sale_id=xyz123</return_url>
            <cancel_url>https://www.example.com/myshoppingcard?
action=cancel&sale_id=xyz123</cancel_url>

        <billing_type>MerchantInitiatedBillingSingleSource</billing_type>
            <payment_action>order</payment_action>
        </PayPalTxn>
    </Transaction>
</Request>
```

2.7.1.2.7 txn_refund

Example XML for a full refund

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <PayPalTxn>
      <method>txn_refund</method>
      <reference>4000900012345671</reference>
    </PayPalTxn>
    <TxnDetails>
      <merchantreference>4655eubtr</merchantreference>
    </TxnDetails>
  </Transaction>
</Request>
```

Example XML for a partial refund

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount>19.00</amount>
      <merchantreference>45gy7byrty</merchantreference>
    </TxnDetails>
    <PayPalTxn>
      <method>txn_refund</method>
      <reference>4000900012345671</reference>
      <note>Here's the refund I promised</note>
    </PayPalTxn>
  </Transaction>
</Request>
```

2.7.1.2.8 do_authorization

Example XML for do_authorization

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount>100.00</amount>
        </TxnDetails>
        <PayPalTxn>
            <method>do_authorization</method>
            <reference>4000900012345671</reference>
        </PayPalTxn>
    </Transaction>
</Request>
```

2.7.1.2.9 do_capture

Example XML for do_capture

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount>100.00</amount>
        </TxnDetails>
        <PayPalTxn>
            <method>do_capture</method>
            <reference>4000900012345671</reference>
        </PayPalTxn>
    </Transaction>
</Request>
```

Example XML for do_capture

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount>100.00</amount>
        </TxnDetails>
        <PayPalTxn>
            <method>do_capture</method>
            <reference>4000900012345671</reference>
            <completed>yes</completed>
            <invnum>inv1234</invnum>
            <note>Finally, I have captured some funds</note>
            <soft_descriptor>FooCorp LLC</soft_descriptor>
        </PayPalTxn>
    </Transaction>
</Request>
```

2.7.1.2.10 do_void

Example XML for do_void

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <PayPalTxn>
            <method>do_void</method>
            <reference>4000900012345671</reference>
        </PayPalTxn>
    </Transaction>
</Request>
```

Example XML for do_void

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <PayPalTxn>
      <method>do_void</method>
      <reference>4000900012345671</reference>
      <note>Excess funds released</note>
    </PayPalTxn>
  </Transaction>
</Request>
```

2.7.1.2.11 create_billing_agreement**Example XML for create_billing_agreement**

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <PayPalTxn>
      <method>create_billing_agreement</method>
      <reference>4000900012345671</reference>
    </PayPalTxn>
  </Transaction>
</Request>
```

2.7.1.2.12 update_billing_agreement

Example XML for update_billing_agreement for cancelling a billing agreement

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <PayPalTxn>
      <method>update_billing_agreement</method>
      <reference>4000900012345671</reference>
      <billing_agreement_status>Cancelled</billing_agreement_status>
    </PayPalTxn>
  </Transaction>
</Request>
```

Example XML for update_billing_agreement to update the description and custom field

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <PayPalTxn>
      <method>update_billing_agreement</method>
      <reference>4000900012345671</reference>
      <billing_agreement_description>Subscription to FOO Magazine
      </billing_agreement_description>
      <billing_agreement_custom>Part of our CHEAP09 prom
      </billing_agreement_custom>
    </PayPalTxn>
  </Transaction>
</Request>
```

2.7.1.2.13 do_reference_transaction

Example XML for do_reference_transaction, with immediate settlement of funds.

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <PayPalTxn>
      <method>do_reference_transaction</method>
      <reference>4000900012345671</reference>
      <description>Twenty NeverFail(tm) Widgets</description>
      <custom>UPS_ID=12345678</custom>
      <invnum>abc123</invnum>
      <buttonsource>foobar</buttonsource>
      <notify_url>http://www.example.com/mynotifyurl?id=abc123
        </notify_url>
      <item_total>85.00</item_total>
      <shipping_total>10.00</shipping_total>
      <handling_total>5.00</handling_total>
      <tax_total>11.97</tax_total>
      <Items>...</Items>
      <ShippingAddress>...</ShippingAddress>
    </PayPalTxn>
  </Transaction>
</Request>
```

Example XML for do_reference_transaction, with delayed settlement of funds

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
    <Authentication>
        <client>xxxx</client>
        <password>xxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">100.00</amount>
        </TxnDetails>
        <PayPalTxn>
            <method>do_reference_transaction</method>
            <reference>4000900012345671</reference>
            <payment action>order</payment action>
        </PayPalTxn>
    </Transaction>
</Request>
```

2.7.1.3 Schema Elements for Response

The following section will highlight elements which are specific to PayPal transactions. In addition to these, some general elements will be returned. Please refer to section 2.1.1.3 of the Developers Guide for descriptions of these elements.

2.7.1.3.1 PayPalTxn

The PayPalTxn element returns all of the details which are specific to the PayPal Service. It should be noted that this document shows all possible fields which may be returned by PayPal. The fields that are actually returned will depend upon the data provided by your customer to PayPal, plus how your own merchant account at PayPal is set up. The following key will be used:

- R – always returned
- A – returned if available

Many fields are only available for specific transaction types:

- S – set_express_checkout
- G – get_express_checkout_details
- D – do_express_checkout_payment
- R – txn_refund
- A – do_authorization
- C – do_capture
- V – do_void
- CB – create_billing_agreement
- UB – update_billing_agreement
- DR – do_reference_transaction
- E – error messages

Element Name: PayPalTxn
Position(s) Response.Transaction

Elements of PayPalTxn												
Element Name	Description	Values / Limitations	Provided (if available)?									
			S	G	D	R	A	C	V	C B	U B	D R
ack		Success SuccessWithWarning Failure FailureWithWarning Warning	A	A	A	A	A	A	A	A	A	A
addressid			-	A	-	-	-	-	-	-	-	-

addressstatus	Status of street address on file with PayPal	Confirmed None Unconfirmed	- A -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -
amt	The final amount charged, including any shipping and taxes from your PayPal Merchant Profile	Must not exceed \$10,000 in any currency. Max value 9,999,999.99	- - A - A A A - - A -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -
billing_agreement_accepted	Whether or not the customer accepted the billing agreement	1 – the customer accepted the billing agreement 0 – the customer did not agree to the billing agreement	- A -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -
billing_agreement_id	The value of the Billing Agreement ID originally returned from do_express_checkout_payment or create_billing_agreement		- - A - - - -	A A - - -	- - - - -	- - - - -	- - - - -	- - - - -
billing_agreement_description	The description held by PayPal for the Billing Agreement	127 single-byte characters	- - - - - - - -	- A - - -	- - - - -	- - - - -	- - - - -	- - - - -
billing_agreement_custom	The custom annotation field held by PayPal for the Billing Agreement	255 single-byte characters	- - - - - - - -	- - A - - -	- - - - -	- - - - -	- - - - -	- - - - -
build	A string representing the specific software build that processes the request and generated the response			A A A A A A A A A A A A				
business	Payer's business name	127 single-byte characters	- A -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -
correlationid	A string that identifies the exact API request that produced this response	Max 41 characters	A A A A - - -	A A A A A A A A A A				

countrycode	Payer's country of residence	As defined in ISO 3166-1	- A -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -
currencycode	The currency of the transaction			- - A - - - - -	- - - - -	- - - - -	- - - - -	- - - - -	A -
custom	As section 2.7.1.1.1		- A -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -
email	As section 2.7.1.1.1		- A -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -
Errors	Returned for error and warning transactions. Please refer to section 2.7.1.3.4								
exchangerate	Exchange rate if a currency conversion occurred. Relevant only if you are billing in the customer's non-primary currency. If the customer chooses to pay with a currency other than the primary currency, the conversion occurs in the customer's account.	Decimal. Max size 17 characters including decimal point	- - A - - A - - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -
feeamt	PayPal fee amount charged for the transaction	Must not exceed \$10,000 in any currency.	- - A - - A - - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	A -
feerefundamt	Transaction fee refunded to original recipient of payment	Max value 9,999,999.99	- - - - A - - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -
firstname	Payer's first name	25 single-byte characters	- A -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -
grossrefundamt	Amount of money refunded to original payer	Must not exceed \$10,000 in any currency.	- - - - A - - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -
invnum	As section 2.7.1.1.1		- A -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -	- - - - -

lastname	Payer's last name	25 single-byte characters	-	A	-	-	-	-	-	-	-	-	-
middlename	Payer's middle name	25 single-byte characters	-	A	-	-	-	-	-	-	-	-	-
netrefundamt	Amount subtracted from PayPal balance of original recipient payment to make this refund	Must not exceed \$10,000 in any currency. Max value 9,999,999.99	-	-	-	A	-	-	-	-	-	-	-
ordertime	Time/date stamp of the payment, in ISO 8601 (UTC/Zulu time)	YYYY-MM-DDTHH:MM:SSZ	-	-	A	-	-	A	-	-	-	A	-
payerid	Unique PayPal customer account identification number	13 single-byte characters	-	A	-	-	-	-	-	-	-	-	-
payerstatus	The status of the payer	verified unverified	-	A	-	-	-	-	-	-	-	-	-
paymentstatus	Status of the payment	See table below for values	-	-	A	-	-	A	-	-	-	A	-
paymenttype	Indicates whether the payment is instant or delayed	none echeck instant	-	-	A	-	-	A	-	-	-	A	-
pendingreason	The reason the payment is pending	See table below for possible values	-	-	A	-	-	-	-	-	-	A	-
phonenum	Payer's contact telephone number		-	A	-	-	-	-	-	-	-	-	-
reasoncode	The reason for the reversal, if transactiontype is reversal	See table below	-	-	A	-	-	-	-	-	-	-	-
refundtransactionid	Unique transaction ID for the refund	17 single-byte characters	-	-	-	A	-	-	-	-	-	-	-
salutation	Payer's salutation	20 single-byte characters	-	A	-	-	-	-	-	-	-	-	-

settlemnt	Amount deposited in your PayPal account after currency conversion	Must not exceed \$10,000 in any currency. Max value 9,999,999.99	-	-	A	-	-	A	-	-	-	A	-
ShippingAddress	If returned, this data will be in the same format as described in section 2.7.1.1.6												
suffix	Payer's suffix	12 single-byte characters	-	A	-	-	-	-	-	-	-	-	-
taxamt	Tax charged on the transaction	Must not exceed \$10,000 in any currency. Max value 9,999,999.99	-	-	A	-	-	A	-	-	-	A	-
timestamp	Date/time of response in ISO 8601 (UTC/Zulu time)	YYYY-MM-DDTHH:MM:SS Z	A	A	A	A	A	A	A	A	A	A	A
token	A timestamped token by which you identify to PayPal that you are processing this payment with Express Checkout. Token expires after three hours	20 single-byte characters	R	A	A	-	-	-	-	-	-	-	-
transactionid	Unique transaction ID for the payment	19 single-byte characters	-	-	A	-	A	A	-	-	A	-	-
transactiontype	The type of transaction	express-checkout reversal merchtpmt	-	-	A	-	-	A	-	-	-	A	-
version	PayPal API service version number		A	A	A	A	A	A	A	A	A	A	A
authorizationid	Value of transactionid returned in response of do_authorization	19 single-byte characters	-	-	-	-	-	A	-	-	-	-	-

parenttransactionid	Value of transactionid returned in response of do_authorization	19 single-byte characters	-	-	-	-	-	A	-	-	-	-	-	-	-	-	-	-	-	-	-
receipt	Receipt identification number	19 single-byte characters	-	-	-	-	-	A	-	-	-	-	-	-	-	-	-	-	-	-	-

Values for paymentstatus	
Value	Meaning
Completed	The payment has been completed, and the funds have been added successfully to your account balance.
Pending	The payment is pending. See the pendingreason element for more information.

Values for pendingreason	
Value	Meaning
none	No pending reason.
address	The payment is pending because your customer did not include a confirmed shipping address and your Payment Receiving Preferences is set such that you want to manually accept or deny each of these payments. To change your preference, go to the Preferences section of your Profile.
echeck	The payment is pending because it was made by an eCheck that has not yet cleared.
intl	The payment is pending because you hold a non-U.S. account and do not have a withdrawal mechanism. You must manually accept or deny this payment from your Account Overview.
multi-currency	You do not have a balance in the currency sent, and you do not have your Payment Receiving Preferences set to automatically convert and accept this payment. You must manually accept or deny this payment.
verify	The payment is pending because you are not yet verified. You must verify your account before you can accept this payment.
other	The payment is pending for a reason other than those listed above. For more information, contact PayPal customer service.

Values for reasoncode	
Value	Meaning
none	No reason code
chargeback	A reversal has occurred on this transaction due to a chargeback by your customer.
guarentee	A reversal has occurred on this transaction due to your customer triggering a money-back guarantee.
buyer-complaint	A reversal has occurred on this transaction due to a complaint about the transaction from your customer.
refund	A reversal has occurred on this transaction because you have given the customer a refund.
other	A reversal has occurred on this transaction due to a reason not listed above.

Example XML for PayPal elements, for set_express_checkout

```
<PayPalTxn>
    <method>set_express_checkout</method>
    <return_url>https://www.example.com/myshoppingcard?action=
complete&sale_id=xyz123</return_url>
    <cancel_url>https://www.example.com/myshoppingcard?action=
cancel&sale_id=xyz123</cancel_url>
    <payment_action>order</payment_action>
</PayPalTxn>
```

Example XML for PayPal elements, for set_express_checkout

```
<PayPalTxn>
    <ack>Success</ack>
    <build>1.0006</build>
    <timestmp>2010-10-03T13:44:56Z</timestmp>
    <token>EC-4VL78907RS990801R</token>
    <version>2.300000</version>
</PayPalTxn>
```

Example XML for PayPal elements, for get_express_checkout_details

```

<PayPalTxn>
  <ack>Success</ack>
  <addressid>PayPal</addressid>
  <addressstatus>Confirmed</addressstatus>
  <billing_agreement_accepted>1
    </billing_agreement_accepted>
  <build>1.0006</build>
  <correlationid>6389c035b7785</correlationid>
  <countrycode>GB</countrycode>
  <custom>my_custom_data</custom>
  <email>abc_11_1191417836_per@example.com</email>
  <firstname>Test</firstname>
  <invnum>123-456-abc</invnum>
  <middlename>Arther</middlename>
  <lastname>User</lastname>
  <payerid>8PN3BJPN2WNDE</payerid>
  <payerstatus>unverified</payerstatus>
  <phonenum>+44 123 4567</phonenum>
  <salutation>Mr</salutation>
  <ShippingAddress>...</ShippingAddress>
  <suffix>Esq</suffix>
  <timestamp>2010-10-03T13:44:56Z</timestamp>
  <token>EC-4VL78907RS990801R</token>
  <version>2.300000</version>
</PayPalTxn>

```

Example XML for PayPal elements, for do_express_checkout_payment

```

<PayPalTxn>
  <ack>Success</ack>
  <amt>10.00</amt>
  <billing_agreement_id>billing123</billing_agreement_id>
  <build>1.0006</build>
  <correlationid>889dd330d54f0</correlationid>
  <currencycode>GBP</currencycode>
  <exchangerate>1.2222</exchangerate>
  <feeamt>0.54</feeamt>
  <ordertime>2010-10-03T13:45:17Z</ordertime>
  <paymentstatus>Completed</paymentstatus>
  <paymenttype>instant</paymenttype>
  <pendingreason>None</pendingreason>
  <reasoncode>None</reasoncode>
  <settleamt>13.22</settleamt>
  <taxamt>0.00</taxamt>
  <tstamp>2010-10-03T13:45:26Z</tstamp>
  <token>EC-4VL78907RS990801R</token>
  <transactionid>3YT66098JE5874747</transactionid>
  <transactiontype>expresscheckout</transactiontype>
  <version>2.300000</version>
</PayPalTxn>

```

Example XML for PayPal elements, for refund

```
<PayPalTxn>
  <build>1.0006</build>
  <feerefundamt>0.54</feerefundamt>
  <version>2.300000</version>
  <timestampl>2010-10-05T13:09:42Z</timestampl>
  <correlationid>5db496bf4454</correlationid>
  <refundtransactionid>83V1269821897443H</refundtransactionid>
  <grossrefundamt>10.00</grossrefundamt>
  <ack>Success</ack>
  <netrefundamt>9.4</netrefundamt>
</PayPalTxn>
```

Example XML for PayPal elements, for do_authorization

```
<PayPalTxn>
  <ack>Success</ack>
  <amt>10.00</amt>
  <build>1.0006</build>
  <timestampl>2010-10-03T13:45:26Z</timestampl>
  <transactionid>3YT66098JE5874747</transactionid>
  <version>2.300000</version>
</PayPalTxn>
```

Example XML for PayPal elements, for do_capture

```
<PayPalTxn>
  <ack>Success</ack>
  <amt>10.00</amt>
  <authorizationid>123456</authorizationid>
  <build>1.0006</build>
  <exchangerate>1.2222</exchangerate>
  <feeamt>0.54</feeamt>
  <ordertime>2010-10-03T13:45:17Z</ordertime>
  <parenttransactionid>J660-98E5-8747-473YT</parenttransactionid>
  <paymentstatus>Completed</paymentstatus>
  <paymenttype>instant</paymenttype>
  <receipt>34FB-SU1K-LAD9-SDA8</receipt>
  <settleamt>13.22</settleamt>
  <taxamt>0.00</taxamt>
  <timestampl>2010-10-03T13:45:26Z</timestampl>
  <transactionid>JE587-4747-3YT6-6098</transactionid>
  <transactiontype>express-checkout</transactiontype>
  <version>2.300000</version>
</PayPalTxn>
```

Example XML for PayPal elements, for do_void

```
<PayPalTxn>
  <ack>Success</ack>
  <amt>10.00</amt>
  <build>1.0006</build>
  <timestamp>2010-10-03T13:45:26Z</timestamp>
  <transactionid>3YT66098JE5874747</transactionid>
  <version>2.300000</version>
</PayPalTxn>
```

Example XML for PayPal elements, for create_billing_agreement

```
<PayPalTxn>
  <ack>Success</ack>
  <billing_agreement_id>B-4C465615S44462100
    </billing_agreement_id>
  <build>1.0006</build>
  <correlationid>3YT66098JE5874747</correlationid>
  <timestamp>2010-10-03T13:45:26Z</timestamp>
  <version>2.300000</version>
</PayPalTxn>
```

Example XML for PayPal elements, for update_billing_agreement

```
<PayPalTxn>
  <ack>Success</ack>
  <billing_agreement_custom>Part of our CHEAP09 promo
    </billing_agreement_custom>
  <billing_agreement_description>9.99 per month subscription to
    FOO Magazine</billing_agreement_description>
  <billing_agreement_id>B-4C465615S44462100
    </billing_agreement_id>
  <build>1.0006</build>
  <correlationid>3YT66098JE5874747</correlationid>
  <timestamp>2010-10-03T13:45:26Z</timestamp>
  <version>2.300000</version>
</PayPalTxn>
```

Example XML for PayPal elements, for do_reference_transaction

```
<PayPalTxn>
  <ack>Success</ack>
  <amt>30.00</amt>
  <authorizationid>HOFAOW8973H4FA</authorizationid>
  <build>1.0006</build>
  <correlationid>ABCD123EFGH456</correlationid>
  <feeamt>5.00</feeamt>
  <paymentstatus>Completed</paymentstatus>
  <paymenttype>instant</paymenttype>
  <settleamt>25.00</settleamt>
  <taxamt>5.00</taxamt>
  <timestampl>2010-11-01T12:34:56Z</timestampl>
  <transactionid>AUWEHRGLUH92374</transactionid>
  <transactiontype>merchtpmt</transactiontype>
  <version>58.000000</version>
</PayPalTxn>
```

2.7.1.3.2

Additional XML Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">100.00</amount>
      <merchantreference>12345601</merchantreference>
    </TxnDetails>
    <PayPalTxn>
      <method>do_express_checkout_payment</method>
      <payment_action>order</payment_action>
      <reference>4000900012345671</reference>
    </PayPalTxn>
  </Transaction>
</Request>
```

2.7.1.3.3 Errors

This element will only be returned if a PayPal error or warning occurs.

Elements of Errors	
Value	Description
Error	See section 2.7.1.3.4

Example XML for Errors element

```
<Errors>
  <Error id="0">...</Error>
  <Error id="1">...</Error>
  ...
</Errors>
```

2.7.1.3.4 Error

This element will only be returned if a PayPal error or warning occurs. PayPal can return multiple Errors for a single transaction. The ID numbering of these errors will correspond to the number used by PayPal in their message response.

Elements of Errors
error_code
short_message
long_message
severity_code

Each error code which is returned will contain the short and long message. Please refer to Appendix A (page 159) of the [Name-Value Pair API Developer Guide and Reference](#) for a full list of error codes and their explanations.

Example XML for Error element

```
<Error id="0">
  <error_code>81104</error_code>
  <short_message>Missing Parameter</short_message>
  <long_message>CancelURL : Required parameter
    missing</long_message>
  <severity_code>severityCode</severity_code>
</Error>
<Error id="1">
  <error_code>10431</error_code>
  <short_message>Item amount is invalid</short_message>
  <long_message>Item amount is invalid</long_message>
</Error>
```

2.7.1.4 XML Example Responses

2.7.1.4.1 Successful Transactions

Example XML Response for a set_express_checkout transaction

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <status>1</status>
    <merchantreference>12345601</merchantreference>
    <reason>ACCEPTED</reason>
    <mode>LIVE</mode>
    <time>1132843641</time>
    <PayPalTxn>
        <ack>Success</ack>
        <build>1.0006</build>
        <timestamP>2010-10-03T13:44:56Z</timestamP>
        <token>EC-4VL78907RS990801R</token>
        <version>2.300000</version>
    </PayPalTxn>
</Response>
```

Example XML Response for a get_express_checkout_details transaction

```

<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <status>1</status>
    <merchantreference>12345601</merchantreference>
    <reason>ACCEPTED</reason>
    <mode>LIVE</mode>
    <time>1132843641</time>
    <PayPalTxn>
        <ack>Success</ack>
        <addressid>PayPal</addressid>
        <addressstatus>Confirmed</addressstatus>
        <BillingAddress>
            <name>Test User</name>
            <street_address1>12 Boddingtons Road</street_address1>
            <street_address2>Regeant's Park</street_address2>
            <city>Wolverhampton</city>
            <region>West Midlands</region>
            <country>United Kingdom</country>
            <country_code>GB</country_code>
            <address_owner>Foobar Ltd</address_owner>
            <address_status>Confirmed</address_status>
        </BillingAddress>
        <billing_agreement_accepted>1
            </billing_agreement_accepted>
        <build>1.0006</build>
        <correlationid>6389c035b7785</correlationid>
        <countrycode>GB</countrycode>
        <custom>my_custom_data</custom>
        <email>abc_11_1191417836_per@example.com</email>
        <firstname>Test</firstname>
        <invnum>123-456-abc</invnum>
        <middlename>Arther</middlename>
        <lastname>User</lastname>
        <payerid>8PN3BJPN2WNDE</payerid>
        <payerstatus>unverified</payerstatus>
        <phonenum>+44 123 4567</phonenum>
        <salutation>Mr</salutation>
        <ShippingAddress>
            <name>Test User</name>
            <street_address1>1 Main Terrace</street_address1>
            <street_address2>Highgate</street_address2>
            <city>Wolverhampton</city>
            <region>West Midlands</region>
            <country>United Kingdom</country>
            <country_code>GB</country_code>
            <postcode>E12 4LQ</postcode>
            <telephone_number>0121 231 3122</telephone_number>
        </ShippingAddress>
        <suffix>Esq</suffix>
        <timestamp>2010-10-03T13:44:56Z</timestamp>
        <token>EC-4VL78907RS990801R</token>
        <version>2.300000</version>
    </PayPalTxn>

```

```
</Response>
```

Example XML Response for do_express_checkout_payment transaction

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash reference>4000900012345671</datacash reference>
    <status>1</status>
    <merchantreference>12345601</merchantreference>
    <reason>ACCEPTED</reason>
    <mode>LIVE</mode>
    <time>1132843641</time>
    <PayPalTxn>
        <ack>Success</ack>
        <amt>10.00</amt>
        <billing_agreement_id>billing123</billing_agreement_id>
        <build>1.0006</build>
        <correlationid>889dd330d54f0</correlationid>
        <currencycode>GBP</currencycode>
        <exchangerate>1.2222</exchangerate>
        <feeamt>0.54</feeamt>
        <ordertime>2010-10-03T13:45:17Z</ordertime>
        <paymentstatus>Completed</paymentstatus>
        <paymenttype>instant</paymenttype>
        <pendingreason>None</pendingreason>
        <reasoncode>None</reasoncode>
        <settleamt>13.22</settleamt>
        <taxamt>0.00</taxamt>
        <timestamp>2010-10-03T13:45:26Z</timestamp>
        <token>EC-4VL78907RS990801R</token>
        <transactionid>3YT66098JE5874747</transactionid>
        <transactiontype>expresscheckout</transactiontype>
        <version>2.300000</version>
    </PayPalTxn>
</Response>
```

Example XML Response for a refund

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <status>1</status>
    <merchantreference>12345601</merchantreference>
    <reason>ACCEPTED</reason>
    <mode>LIVE</mode>
    <time>1132843641</time>
    <PayPalTxn>
        <build>1.0006</build>
        <feerefundamt>0.54</feerefundamt>
        <version>2.300000</version>
        <timestamp>2010-10-05T13:09:42Z</timestamp>
        <correlationid>5db496bf4454</correlationid>
        <refundtransactionid>83V1269821897443H</refundtransactionid>
        <grossrefundamt>10.00</grossrefundamt>
        <ack>Success</ack>
        <netrefundamt>9.4</netrefundamt>
    </PayPalTxn>
</Response>
```

Example XML Response for a do_authorization

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <status>1</status>
    <merchantreference>12345601</merchantreference>
    <reason>ACCEPTED</reason>
    <mode>LIVE</mode>
    <time>1132843641</time>
    <PayPalTxn>
        <ack>Success</ack>
        <amt>10.00</amt>
        <build>1.0006</build>
        <timestamp>2010-10-03T13:45:26Z</timestamp>
        <transactionid>3YT66098JE5874747</transactionid>
        <version>2.300000</version>
    </PayPalTxn>
</Response>
```

Example XML Response for a do_capture which cancels any remaining funds received against the payers account

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <status>1</status>
    <merchantreference>12345601</merchantreference>
    <reason>ACCEPTED</reason>
    <mode>LIVE</mode>
    <time>1132843641</time>
    <PayPalTxn>
        <ack>Success</ack>
        <amt>10.00</amt>
        <authorizationid>123456</authorizationid>
        <build>1.0006</build>
        <exchangerate>1.2222</exchangerate>
        <feeamt>0.54</feeamt>
        <ordertime>2010-10-03T13:45:17Z</ordertime>
        <parenttransactionid>JE660-98E5-8747-473YT</parenttransactionid>
        <paymentstatus>Completed</paymentstatus>
        <paymenttype>instant</paymenttype>
        <receipt>34FB-SU1K-LAD9-SDA8</receipt>
        <settleamt>13.22</settleamt>
        <taxamt>0.00</taxamt>
        <timestamp>2010-10-03T13:45:26Z</timestamp>
        <transactionid>JE587-4747-3YT6-6098</transactionid>
        <transactiontype>express-checkout</transactiontype>
        <version>2.300000</version>
    </PayPalTxn>
</Response>
```

Example XML Response for a do_void

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <status>1</status>
    <merchantreference>12345601</merchantreference>
    <reason>ACCEPTED</reason>
    <mode>LIVE</mode>
    <time>1132843641</time>
    <PayPalTxn>
        <ack>Success</ack>
        <amt>10.00</amt>
        <build>1.0006</build>
        <timestamp>2010-10-03T13:45:26Z</timestamp>
        <transactionid>3YT66098JE5874747</transactionid>
        <version>2.300000</version>
    </PayPalTxn>
</Response>
```

Example XML Response for a create_billing_agreement

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <datacash_reference>4000900012345671</datacash_reference>
  <status>1</status>
  <merchantreference>12345601</merchantreference>
  <reason>ACCEPTED</reason>
  <mode>LIVE</mode>
  <time>1132843641</time>
  <PayPalTxn>
    <ack>Success</ack>
    <billing_agreement_id>B-4C465615S44462100
      </billing_agreement_id>
    <build>1.0006</build>
    <correlationid>abc123correlationIDftw</correlationid>
    <timestampl>2010-11-01T12:34:56Z</timestampl>
    <version>58.000000</version>
  </PayPalTxn>
</Response>
```

Example XML Response for an update_billing_agreement

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <datacash_reference>4000900012345671</datacash_reference>
  <status>1</status>
  <merchantreference>12345601</merchantreference>
  <reason>ACCEPTED</reason>
  <mode>LIVE</mode>
  <time>1132843641</time>
  <PayPalTxn>
    <ack>Success</ack>
    <billing_agreement_custom>Part of our CHEAP09 promo
      </billing_agreement_custom>
    <billing_agreement_description>9.99 per month subscription
      to FOO Magazine</billing_agreement_description>
    <billing_agreement_id>B-4C465615S44462100
      </billing_agreement_id>
    <billing_agreement_status>Active</billing_agreement_status>
    <build>1.0006</build>
    <correlationid>abc123correlationIDftw</correlationid>
    <timestampl>2010-11-01T12:34:56Z</timestampl>
    <version>58.000000</version>
  </PayPalTxn>
</Response>
```

Example XML Response for a do_reference_transaction

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <status>1</status>
    <merchantreference>12345601</merchantreference>
    <reason>ACCEPTED</reason>
    <mode>LIVE</mode>
    <time>1132843641</time>
    <PayPalTxn>
        <ack>Success</ack>
        <amt>30.00</amt>
        <authorizationid>HOFAOW8973H4FA</authorizationid>
        <build>1.0006</build>
        <correlationid>ABCD123EFGH456</correlationid>
        <feeamt>5.00</feeamt>
        <paymentstatus>Completed</paymentstatus>
        <paymenttype>instant</paymenttype>
        <settleamt>25.00</settleamt>
        <taxamt>5.00</taxamt>
        <timestampl>2010-11-01T12:34:56Z</timestampl>
        <transactionid>AUWEHRLUH92374</transactionid>
        <transactiontype>merchtpmt</transactiontype>
        <version>58.000000</version>
    </PayPalTxn>
</Response>
```

2.7.1.4.2 PayPal Warning and Errors

Example XML Response for an error

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <merchantereference>12345601</merchantereference>
    <mode>LIVE</mode>
    <time>1132843641</time>
    <status>561</status>
    <reason>PayPal: Error returned in response</reason>
    <information>The transaction response from PayPal indicated that
        an error has occurred - see the extended PayPal
        information</information>
    <PayPalTxn>
        <ack>Failure</ack>
        <timestamp>2010-10-03T13:45:26Z</timestamp>
        <correlationid>889dd330d54f0</correlationid>
        <version>2.300000</version>
        <build>1.0006</build>
        <Errors>
            <Error id="0">
                <error_code>81104</error_code>
                <short_message>Missing Parameter</short_message>
                <long_message>CancelURL : Required parameter
                    missing</long_message>
                <severity_code>severityCode</severity_code>
            </Error>
        </Errors>
    </PayPalTxn>
</Response>
```

2.7.1.4.3 Invalid Transactions

When DataCash is sent a transaction, it will be validated before forwarding the information to PayPal. If a transaction fails this validation, the response will not have a `PayPalTxn` element. Examples are shown below.

A full list of error codes is available in the [Developers Area](#). Additional examples - and suggestions for how to prevent the errors re-occurring - are published in the [Support Centre](#).

Example XML Response for an invalid transaction

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <merchantreference>12345601</merchantreference>
    <status>560</status>
    <reason>PayPal: Not configured for service</reason>
    <information>Merchant is not configured for the PayPal
        service</information>
    <mode>LIVE</mode>
    <time>1132843641</time>
</Response>
```

Example XML Response for an invalid transaction

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <merchantreference>12345601</merchantreference>
    <status>565</status>
    <reason>PayPal: Invalid reference</reason>
    <information>Referenced transactions is not a
        PayPalTxn</information>
    <mode>LIVE</mode>
    <time>1132843641</time>
</Response>
```

2.7.1.5 Re-Directing the Customer

This section describes how to re-direct your customer from your website to PayPal.

In order to complete this process, you must have submitted a `set_express_checkout` transaction to DataCash, and received a successful response. This response will provide you with a token (section 2.1.1.3):

Example Successful XML Response for `set_express_checkout`, highlighting the token

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>...</datacash_reference>
    <status>1</status>
    <merchantereference>...</merchantereference>
    <reason>ACCEPTED</reason>
    <mode>LIVE</mode>
    <time>...</time>
    <PayPalTxn>
        <token>EC-4VL78907RS990801R</token>
    </PayPalTxn>
</Response>
```

This token is used to construct the URL to which you will re-direct your customer, as shown below:

Example URL for re-direct

```
https://www.paypal.com/cgi-bin/webscr?cmd=_express-checkout
&token=EC-4VL78907RS990801R
https://www.sandbox.paypal
```

When redirecting the customer's browser to the PayPal login page, PayPal recommends that you use the HTTPS response 302 "Object Moved" with the URL above as the value of the Location header in the HTTPS response. Ensure that you use an SSL-enabled server to prevent browser warnings about a mix of secure and insecure graphics.

2.7.1.6 Query Transaction for PayPal

Query transactions can be submitted against the DataCash reference of a PayPal transaction to obtain information about that transaction.

2.7.1.6.1 Schema Elements For Request

- Request
- Authentication – section 2.2.1.1
- Transaction
- HistoricTxn – see section 2.5.2.3.2.1.

Elements of HistoricTxn			
Element Name	Description	Values / Limitations	
method	The transaction type	query	R
reference	The reference of the transaction to be queried	DataCash Reference of a previous PayPal transaction on the same client.	R

Example XML Request for HistoricTxn complex elements

```
<HistoricTxn>
  <reference>3900900100010001</reference>
  <method>query</method>
</HistoricTxn>
```

2.7.1.6.2 XML Example - PayPal Query Transaction

Example XML PayPal Query

```
<Request>
  <Authentication>
    <password>*****</password>
    <client>99000001</client>
  </Authentication>
  <Transaction>
    <HistoricTxn>
      <method>query</method>
      <reference>3900900100010001</reference>
    </HistoricTxn>
  </Transaction>
</Request>
```

2.7.1.6.3 Schema Elements for PayPal Query Response

Element Name: QueryTxnResult
Position: Response

The `QueryTxnResult` complex element contains the retrieved details for the transaction reference provided in the request. The contents of the following elements all relate to the original transaction being queried, and not the query transaction itself.

Elements of QueryTxnResult	
Element Name	Description
PayPalTxn	Complex element containing details about the PayPal transaction being queried
datacash_reference	DataCash reference
merchant_reference	Merchant reference
reason	DataCash response message.
status	DataCash response code
transaction_date	Date of processing by DPG

Elements of PayPalTxn	
Element Name	Description
method	The method of the PayPal transaction being queried
status_update	Indicates whether the transaction status was updated as a result of the query (see 2.7.1.6.5)
Captures	Complex element returned when the queried transaction was an Authorization (method <code>do_authorization</code>), containing details of any captures (method <code>do_capture</code>) that were associated with that transaction

Elements of Captures	
Element Name	Description
Capture	Complex element containing details of a single capture. An element of this type will appear for each capture associated with the authorization.

Elements of Capture	
Element Name	Description
datacash_reference	The DataCash reference of the capture transaction. This can be used to query the capture transaction directly.
dc_response	The DataCash response code of the capture transaction.

2.7.1.6.4 XML Example - PayPal Query Response

Example XML Response for basic PayPal Query Txn

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <QueryTxnResult>
    <PayPalTxn>
      <method>do_express_checkout_payment</method>
    </PayPalTxn>
    <datacash_reference>4600900012345673</datacash_reference>
    <merchant_reference>mref12345680</merchant_reference>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <transaction_date>2011-07-04 14:57:11</transaction_date>
  </QueryTxnResult>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1309787852</time>
</Response>
```

Example XML Response for PayPal Query Txn on an Authorization

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <QueryTxnResult>
    <PayPalTxn>
      <Captures>
        <Capture>
          <datacash_reference>4100900012345675</datacash_reference>
          <dc_response>1</dc_response>
        </Capture>
        <Capture>
          <datacash_reference>4900900012345676</datacash_reference>
          <dc_response>1</dc_response>
        </Capture>
      </Captures>
      <method>do_authorization</method>
    </PayPalTxn>
    <datacash_reference>4400900012345674</datacash_reference>
    <merchant_reference>mref12345678</merchant_reference>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <transaction_date>2011-07-04 15:01:40</transaction_date>
  </QueryTxnResult>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1309788159</time>
</Response>
```

2.7.1.6.5 Status Updates

In the event that a PayPal transaction fails with a reason indicating that no response was received from PayPal, if the transaction had method

do_express_checkout_payment, txn_refund, do_authorization or do_capture then submitting a Query against that transaction will cause the payment gateway to attempt to determine an updated status for the transaction by communicating with PayPal. In the response to the Query the element `status_update` will indicate the outcome of this attempt.

Example XML Response for PayPal Query Txn with status update

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <QueryTxnResult>
    <PayPalTxn>
      <method>txn_refund</method>
      <status_update>Transaction status has been updated</status_update>
    </PayPalTxn>
    <datacash_reference>4100900012345675</datacash_reference>
    <merchant_reference>mref12345677</merchant_reference>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <transaction_date>2011-07-04 15:06:12</transaction_date>
  </QueryTxnResult>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1309788432</time>
</Response>
```

2.7.2 Variable End of Day

This section describes the Variable End of Day service for the dynamic processing model only. If you are using the Static processing model, no transactions need to be sent to the DPG.

2.7.2.1 Schema Elements for Request

In this section, the fields that can be submitted when using the variable end of day (dynamic) service will be presented, along with sample XML for those fields.

When presenting an eod transaction, the following schema elements should be presented:

- Request
 - Authentication – section 2.1.1.1
 - Transaction
 - TxnDetails – the merchant reference – section 2.7.2.1.1
 - EoDTxn – the method eod – section 2.7.2.1.2

All elements are required, unless otherwise stated.

2.7.2.1.1 TxnDetails

There is one element to be presented in this parent element

Elements in TxnDetails		
Element Name	Description	Values / Limitations
merchantreference	A unique reference number for each transaction	Minimum 6, maximum 30 alphanumeric characters. Must be unique

Example XML for EoDTxn elements

```
<TxnDetails>
  <merchantreference>48tv3498wy3c4dt</merchantreference>
</TxnDetails>
```

2.7.2.1.2 EoDTxn

There is one element to be presented in this parent element.

1Elements in EoDTxn		
Element Name	Description	Values / Limitations
method	The transaction type	eod

Example XML for EoDTxn elements

```
<EoDTxn>
  <method>eod</method>
</EoDTxn>
```

2.7.2.2 Example Requests

Example XML Request

```
<Request>
  <Authentication>
    <client>...</client>
    <password>...</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <merchantreference>...</merchantreference>
    </TxnDetails>
    <EoDTxn>
      <method>eod</method>
    </EoDTxn>
  </Transaction>
</Request>
```

2.7.2.3 Schema Elements for Response

Please refer to section 2.1.1.3. There are no additional elements which may be returned in the XML response for this service.

2.7.2.4 Example Responses

Example XML Response for a successful transaction

```
<Response>
  <datacash_reference>4900200047685356</datacash_reference>
  <merchantreference>117990881318233</merchantreference>
  <mode>LIVE</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1179908813</time>
</Response>
```

Example XML Response for a rejected transaction

```
<Response>
  <datacash_reference>4000200047685351</datacash_reference>
  <merchantreference>117990875018165</merchantreference>
  <mode>LIVE</mode>
  <reason>VEoD: already received an EoD</reason>
  <status>532</status>
  <time>1179908750</time>
</Response>
```

2.7.3 RBS Gift Card

2.7.3.1 Summary of Service

The RBS Gift Card service provides the ability to perform a number of transaction types against RBS Gift Cards:

- Balance Enquiries: To determine the remaining balance
- Redemptions: To perform an authorisation against the card
- Refunds: To refund an amount to the card
- txn_refunds: To refund an amount to the card based on a previously authorized redemption.

2.7.3.2 Schema Elements for Request

Element Name: RbsGiftCardTxn
Position: Request.Transaction

Elements of RbsGiftCardTxn			
Element Name	Description	Values / Limitations	Required
method	The method of the transaction.	balance_enquiry redeem refund txn_refund	R
Card	The card details of the transaction to be authenticated. This element contains sub elements. See 2.2.1.1.1 for more information. This is not used in the case of a txn_refund.	See 2.2.1.1.1 for more information	M
Reference	When performing a txn_refund, this element should be used to indicate the transaction that is being refunded.	The value contained should be the datacash_reference of the original RBS gift card transaction to refund.	M

2.7.3.3 Balance Enquiry

The balance enquiry transaction is provided to allow the merchant to check the existing balance of an RBS Gift Card prior to performing a redemption request.

An example balance enquiry transaction is illustrated below:

Example Balance Enquiry Request

```
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>

  <Transaction>
    <TxnDetails>
      <merchantreference>12345601</merchantreference>
      <capturemethod>ecomm</capturemethod>
    </TxnDetails>

    <RbsGiftCardTxn>
      <method>balance_enquiry</method>
      <Card>
        <pan>xxxxxxxxxxxxxxxx</pan>
        <accesscode>xxxxxxxx</accesscode>
        <expirydate>04/06</expirydate>
        <startdate>01/99</startdate>
      </Card>
    </RbsGiftCardTxn>
  </Transaction>
</Request>
```

An example response to a balance enquiry is shown below:

Example Balance Enquiry Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <status>1</status>
    <merchantreference>12345601</merchantreference>
    <reason>ACCEPTED</reason>

    <RbsGiftCardTxn>
        <authcode>100000</authcode>
        <balance>100.00</balance>
        <currency>GBP</currency>
    </RbsGiftCardTxn>

    <mode>TEST</mode>
    <time>1132843641</time>
</Response>
```

2.7.3.4 Redemption

The redemption transaction allows the merchant to collect funds from the gift card. An example redemption transaction is shown below:

Example Redemption Request

```
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">100.00</amount>
            <merchantreference>12345601</merchantreference>
            <capturemethod>ecomm</capturemethod>
        </TxnDetails>

        <RbsGiftCardTxn>
            <method>redeem</method>
            <Card>
                <pan>xxxxxxxxxxxxxxxx</pan>
                <accesscode>xxxxxxxx</accesscode>
                <expirydate>04/06</expirydate>
                <startdate>01/99</startdate>
                <Cv2Avs>
                    <cv2>123</cv2>
                </Cv2Avs>
            </Card>
        </RbsGiftCardTxn>
    </Transaction>
</Request>
```

An example response to a redeem transaction where the card has sufficient funds to complete the transaction is shown below:

Example Redemption Response where card has sufficient funds

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <status>1</status>
    <merchantreference>12345601</merchantreference>
    <reason>ACCEPTED</reason>

    <RbsGiftCardTxn>
        <authcode>100000</authcode>
        <balance>100.00</balance>
    </RbsGiftCardTxn>

    <mode>TEST</mode>
    <time>1132843641</time>
</Response>
```

The remaining balance is presented in the response XML as shown above.

Where the card contains insufficient funds, the response is slightly different. The available funds on the Gift Card will be deducted and the XML response will contain a negative balance. The negative balance will specify the difference between the requested amount and the card balance i.e. the outstanding balance required to complete the transaction.

The response below illustrates a negative balance being returned:

Example Redemption Response where card has insufficient funds

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <status>1</status>
    <merchantreference>12345601</merchantreference>
    <reason>ACCEPTED</reason>

    <RbsGiftCardTxn>
        <authcode>100000</authcode>
        <balance>-41.00</balance>
    </RbsGiftCardTxn>

    <mode>TEST</mode>
    <time>1132843641</time>
</Response>
```

Where a negative balance is returned, the merchant must submit another transaction request using another card (either Gift Card or Credit/Debit card) to secure the outstanding balance.

See the developers guide for information on processing Credit / Debit card transactions.

2.7.3.5 Refunds

The refund transaction allows the merchant to refund the gift card. An example refund transaction is shown below:

Example Refund Request

```
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">10.59</amount>
      <merchantreference>12345601</merchantreference>
      <capturemethod>ecomm</capturemethod>
    </TxnDetails>

    <RbsGiftCardTxn>
      <method>refund</method>
      <Card>
        <pan>xxxxxxxxxxxxxxxx</pan>
        <accesscode>xxxxxxxx</accesscode>
        <expirydate>04/06</expirydate>
        <startdate>01/99</startdate>
      </Card>
    </RbsGiftCardTxn>

  </Transaction>
</Request>
```

An example response to a successful refund transaction is shown below:

Example Refund Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <datacash_reference>4000900012345671</datacash_reference>
  <status>1</status>
  <merchantreference>12345601</merchantreference>
  <reason>ACCEPTED</reason>

  <RbsGiftCardTxn>
    <authcode>100000</authcode>
    <balance>100.00</balance>
  </RbsGiftCardTxn>

  <mode>TEST</mode>
  <time>1132843641</time>
</Response>
```

Note: the response formatting is identical to that for redemption transactions.

2.7.3.6 Txn Refunds

The txn_refund transaction allows the merchant to refund the gift card with a maximum amount equal to that contained within the original redemption transaction as identified by the 'reference' provided in the XML.

Note that where multiple txn_refund transactions are submitted against the same redemption, the sum of the refunded amounts must not exceed the originally authorised amount.

An example txn_refund transaction is shown below:

Example txn_refund Request

```
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount>10.59</amount>
      <capturemethod>ecomm</capturemethod>
    </TxnDetails>

    <RbsGiftCardTxn>
      <method>txn_refund</method>
      <reference>4100900012345675</reference>
    </RbsGiftCardTxn>

  </Transaction>
</Request>
```

If the amount is not provided, the amount refunded will be that of the original redemption transaction. Note that 'currency' is not permitted in a txn_refund.

An example response to a `txn_refund` is provided below:

Example `txn_refund` Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345671</datacash_reference>
    <status>1</status>
    <merchantreference>4000900012345670</merchantreference>
    <reason>ACCEPTED</reason>

    <RbsGiftCardTxn>
        <authcode>100000</authcode>
        <balance>100.00</balance>
    </RbsGiftCardTxn>

    <mode>TEST</mode>
    <time>1132843641</time>
</Response>
```

The 'merchantreference' returned in response to a `txn_refund` is the 'datacash_reference' of the original redemption transaction.

2.7.3.7 Unsuccessful Transactions

In the event of an unsuccessful transaction (where a valid response is received indicating failure at RBS), the balance may be returned, and additional data provided in the response message may be returned in the `information` field.

An example of an unsuccessful transaction can be seen below:

Example Response for a Unsuccessful Transaction

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345672</datacash_reference>
    <status>7</status>
    <merchantreference>12345602</merchantreference>
    <reason>DECLINED</reason>

    <RbsGiftCardTxn>
        <balance>5.25</balance>
    </RbsGiftCardTxn>

    <information>CARD EXPIRED</information>
    <mode>TEST</mode>
    <time>1132843641</time>
</Response>
```

2.7.3.8 Invalid Transactions

In the event of a transaction being rejected by the DPG without going to RBS for processing, the `RbsGiftCardTxn` block will not be returned. A suitable status code and simple message will be returned, with additional details in the `information` field. This is in-line with the existing DataCash response format.

For example:

Example Response for a Invalid Transaction

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345672</datacash_reference>
    <status>631</status>
    <merchantreference>12345602</merchantreference>
    <reason>Inappropriate card number</reason>
    <information>The card provided is not an RBS Gift
        Card</information>
    <mode>TEST</mode>
    <time>1132843641</time>
</Response>
```

Such responses will be received if the XML submitted contains unexpected elements, has missing elements or the data contained within those elements has failed to meet the validation rules.

2.7.3.9 Failed Transactions

A failed transaction is one where an attempt to submit a request to RBS was made, but a communication error has rendered the outcome either failed or unknown.

An example of a failed transaction is given below for completeness:

Example Response for a Failed Transaction

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345672</datacash_reference>
    <status>6</status>
    <merchantreference>12345602</merchantreference>
    <reason>BANK DID NOT REPLY</reason>
    <mode>TEST</mode>
    <time>1132843641</time>
</Response>
```

2.7.3.10 Notes

2.7.3.10.1 Accesscode

This element is optional and if presented must be 8 digits in length. If provided, the 'accesscode' must match that held against the account otherwise the transaction will be declined.

2.7.3.10.2 Capturemethod

This can be one of 'ecomm' or 'cnp' for RBS Gift Card transactions. If this is not specified, the capturemethod will be set by determining the environment of a terminal configured on your account which is capable of processing the card and currency combination specified in the transaction.

Note that the 'capturemethod' is mandatory if the account is configured with multiple merchant ID environments.

2.7.4 *PrePay Technology (PPT)*

2.7.4.1 Summary of Service

The PrePay Technology service (hereafter referred to as PPT) provides the ability to perform a number of transaction types against PPT gift cards:

- Top Ups: To load the PPT gift card with funds
- Redemptions: To perform an authorization against the PPT gift card
- Refunds: To refund an amount to the PPT gift card
- Balance Enquiries: To determine the remaining balance on the PPT gift card
- Reversals: Reverse a previous top up, redemption or refund on the PPT gift card
- txn_refunds: To refund an amount to the card based on a previously authorized redemption.

PPT transactions can be performed in ecommerce or cardholder not present environments. Alternatively PPT cards can be swiped or keyed into a card terminal.

2.7.4.2 Schema Elements for Request

Element Name: PPTCardTxn
Position: Request.Transaction

Elements of PPTCardTxn			
Element Name	Description	Values / Limitations	Required
method	The method of the transaction	top_up redeem refund balance_enquiry reversal txn_refund	R
pan	The PPT card number. Element used for "ecomm", "cnp" and "keyed" capturemethods.	The PPT card number	M
pin	A Pin number can optionally be supplied when performing "top-up", "redeem" and "balance_enquiry" transactions.	4 digit pin number	O
reference	The datacash reference of the related transaction. This element is only valid for "reversal" or "txn_refund" methods.	Datacash reference of the related transaction	M
merchantreference	If subscribed to the merchant reference cancellations service, a "reversal" may be invoked with this element instead of reference	Merchant reference of the related transaction	O
card_details	For the "swiped" capturemethod only, this element is used instead of pan.	The track2 information from the PPT gift card	M
Terminal	For PPT transactions using a terminal ("keyed", "swiped" capturemethods), this element should be present.	See section 0	M

2.7.4.3 Schema Elements for Response

Element Name: PPTCardTxn

Position: Response

Elements of PPTCardTxn		
Element Name	Description	Values / Limitations
Authcode	The authcode of the transaction, this element can also contain messages from PPT e.g "INVALID PIN".	
card_activated	PPT may return information in their authorisation response that indicates that a card has been activated as a result of the transaction. This element will only be included where card activation has occurred.	"yes"
Balance	The balance remaining on the PPT gift card following the transaction, returned with major and minor units (i.e 0.00).	

2.7.4.4 Top Ups

Top up transactions are used to load funds onto the PPT gift card. An example of a "swiped" top up request is shown below:

Example PPT Top Up Request

```
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">100.00</amount>
      <merchantreference>12345601</merchantreference>
      <capturemethod>swiped</capturemethod>
    </TxnDetails>

    <PPTCardTxn>
      <Terminal id="xxxxxxxx">
        <terminal_capabilities ic_reader="true"
          magnetic_stripe_reader="true"
          manual_card_entry="true"/>
        <features_capabilities pin_pad_available="true"/>
      </Terminal>
      <method>top_up</method>
      <card_details
        type="track2">xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx</card_det
        ails>
        <pin>xxxx</pin>
      </PPTCardTxn>

    </Transaction>
  </Request>
```

An example of a DataCash response to a top up request is shown below. Note that in this case, this was the first use of the gift card and so the "card_activated" element is present.

Example PPT Top Up Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <currency>GBP</currency>
    <datacash_reference>4000900012345671</datacash_reference>
    <merchantreference>12345601</merchantreference>
    <PPTCardTxn>
        <authcode>100000</authcode>
        <balance>100.00</balance>
        <card_activated>yes</card_activated>
    </PPTCardTxn>
    <mode>TEST</mode>
    <time>_UNIX_TIME_</time>
</Response>
```

2.7.4.5 Redemption

Redeem transactions are used to authorise fund removal from the gift card. An example of a "keyed" redemption request is shown below:

Example PPT Redemption Request

```
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">40.00</amount>
            <merchantreference>12345602</merchantreference>
            <capturemethod>keyed</capturemethod>
        </TxnDetails>

        <PPTCardTxn>
            <Terminal id="xxxxxxxx">
                <terminal_capabilities ic_reader="true"
magnetic_stripe_reader="true" manual_card_entry="true"/>
                <features_capabilities pin_pad_available="true"/>
            </Terminal>
            <method>redeem</method>
            <pan>xxxxxxxxxxxxxx</pan>
            <pin>xxxx</pin>
        </PPTCardTxn>
    </Transaction>
</Request>
```

Note that the “pin” element is optional.

An example of a response to the above request is shown below:

Example PPT Redemption Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <currency>GBP</currency>
    <datacash_reference>4800900012345672</datacash_reference>
    <merchantreference>12345602</merchantreference>

    <PPTCardTxn>
        <authcode>100000</authcode>
        <balance>60.00</balance>
    </PPTCardTxn>

    <mode>TEST</mode>
    <time>_UNIX_TIME_</time>
</Response>
```

2.7.4.6 Refunds

Refund transactions are used to refund an amount to the gift card, where the refund doesn't relate to an existing redemption on that gift card. An example of an e-commerce refund request is shown below:

Example PPT Refund Request

```
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">30.00</amount>
            <merchantreference>12345603</merchantreference>
            <capturemethod>ecomm</capturemethod>
        </TxnDetails>

        <PPTCardTxn>
            <method>refund</method>
            <pan>xxxxxxxxxxxxxx</pan>
        </PPTCardTxn>

    </Transaction>
</Request>
```

An example response to the above transaction is shown below:

Example PPT Refund Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <currency>GBP</currency>
    <datacash_reference>4800900012345673</datacash_reference>
    <merchantreference>12345603</merchantreference>

    <PPTCardTxn>
        <authcode>100000</authcode>
        <balance>90.00</balance>
    </PPTCardTxn>

    <mode>TEST</mode>
    <time>_UNIX_TIME_</time>
</Response>
```

2.7.4.7 Balance Enquiries

Balance enquiries are used query the amount of funds on the gift card. An example of an e-commerce balance enquiry request is shown below:

Example PPT Balance Enquiry Request

```
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <merchantreference>12345604</merchantreference>
            <capturemethod>ecomm</capturemethod>
        </TxnDetails>

        <PPTCardTxn>
            <method>balance_enquiry</method>
            <pan>xxxxxxxxxxxxxx</pan>
        </PPTCardTxn>

    </Transaction>
</Request>
```

An example response to the above transaction is shown below:

Example PPT Balance Enquiry Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <currency>GBP</currency>
    <datacash_reference>4800900012345674</datacash_reference>
    <merchantreference>12345604</merchantreference>

    <PPTCardTxn>
        <authcode>100000</authcode>
        <balance>90.00</balance>
    </PPTCardTxn>

    <mode>TEST</mode>
    <time>_UNIX_TIME_</time>
</Response>
```

2.7.4.8 Reversals

Reversals can be used to reverse a previous top up, redemption or refund on the PPT gift card, by using the `datacash_reference` (supplied in the response of the previous transaction). An example of an e-commerce reversal request is shown below:

Example PPT Reversal Request

```
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>

        <PPTCardTxn>
            <method>reversal</method>
            <reference>4800900012345673</reference>
        </PPTCardTxn>

    </Transaction>
</Request>
```

An example response to the above transaction is shown below:

Example PPT Reversal Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <reason reversal="success">ACCEPTED</reason>
    <status>1</status>
    <currency>GBP</currency>
    <datacash_reference>4800900012345675</datacash_reference>
    <information>Reversal ACCEPTED status=1
authcode=100000</information>

    <PPTCardTxn>
        <authcode>100000</authcode>
        <balance>60.00</balance>
    </PPTCardTxn>

    <mode>TEST</mode>
    <time>_UNIX_TIME_</time>
</Response>
```

2.7.4.9 Txn_refunds

Txn_Refund transactions are used to refund an amount to the gift card, where the refund relates to an existing redemption on that gift card. The refund may be for the full amount of the transaction, or a partial amount (if the amount element is missing then the full amount will be assumed).

An example of an e-commerce txn_refund request is shown below:

Example PPT Txn_Refund Request

```
<Request>
    <Authentication>
        <client>xxxxxxxx</client>
        <password>xxxxxxxx</password>
    </Authentication>
    <Transaction>
        <TxnDetails>
            <amount currency="GBP">10.59</amount>
        </TxnDetails>

        <PPTCardTxn>
            <method>txn_refund</method>
            <reference>4800900012345672</reference>
        </PPTCardTxn>

    </Transaction>
</Request>
```

An example response to the above transaction is shown below:

Example PPT Txn_Refund Response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <reason>ACCEPTED</reason>
    <status>1</status>
    <currency>GBP</currency>
    <datacash_reference>4800900012345675</datacash_reference>
    <merchantreference>12345602</merchantreference>

    <PPTCardTxn>
        <authcode>100000</authcode>
        <balance>79.41</balance>
    </PPTCardTxn>

    <mode>TEST</mode>
    <time>_UNIX_TIME_</time>
</Response>
```

2.7.4.10 Unsuccessful Transactions

2.7.4.10.1 Declined

In the event of a decline by PPT, the balance may or may not be returned, for example:

Example of a PPT Decline response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <reason>DECLINED</reason>
    <status>7</status>
    <currency>GBP</currency>
    <datacash_reference>4800900012345672</datacash_reference>
    <merchantreference>12345602</merchantreference>

    <PPTCardTxn>
        <authcode>DECLINED</authcode>
        <balance>5.25</balance>
    </PPTCardTxn>

    <mode>TEST</mode>
    <time>_UNIX_TIME_</time>
</Response>
```

2.7.4.10.2 Pin Mismatch

This scenario follows the same format as a Decline above, however the reason field indicates why the transaction was declined, for example:

Example of a PPT PIN mismatch response

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <reason>PIN MISMATCH</reason>
    <status>7</status>
    <currency>GBP</currency>
    <datacash_reference>4800900012345672</datacash_reference>
    <merchantreference>12345602</merchantreference>

    <PPTCardTxn>
        <authcode>PIN MISMATCH 010000010000</authcode>
        <balance>5.25</balance>
    </PPTCardTxn>

    <mode>TEST</mode>
    <time>_UNIX_TIME_</time>
</Response>
```

2.7.4.11 Invalid Transactions

In the event of a transaction being rejected by the DPG without going to PPT for processing, the PPTCardTxn block will not be returned. A suitable status code and simple message will be returned, with additional details in the information field. It may not be possible to include currency information in such transactions. For example:

Example of a response to an Invalid PPT transaction

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
    <datacash_reference>4000900012345672</datacash_reference>
    <status>492</status>
    <merchantreference>12345602</merchantreference>
    <reason>PPT: Inappropriate Request Element</reason>
    <information>Field LINEITEMDETAIL may not be included in a PPT
transaction</information>
    <mode>TEST</mode>
    <time>1132843641</time>
</Response>
```

2.7.4.12 Failed Transactions

A failed transaction will be one where an attempt to submit a request to PPT was made, but a communication error has rendered the outcome either failed or unknown.

On behalf of a merchant, DataCash issue an automatic reversal in this case, and the reversed element is used to indicate whether the reversal was successful. This will only be applicable only to error 6 responses.

Elements of Response		
Element Name	Description	Values / Limitations
reversed	Indicates whether an automatic reversal issued by DataCash in the event of a communications failure was successful.	Y – The original transaction was reversed N – The original transaction did not reach the acquirer so was not reversed F – A reversal was attempted but failed

In the event of a reversed flag of F, it is not possible for DataCash to know the status of the transaction, and the merchant will have to investigate further with PPT. For example:

Example of DataCash Response on PPT communications failure

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <datacash_reference>4000900012345672</datacash_reference>
  <status>6</status>
  <merchantreference>12345602</merchantreference>
  <reason>BANK DID NOT REPLY</reason>
  <information>Reversal attempt FAILED</information>
  <reversed>F</reversed>
  <currency>EUR</currency>
  <mode>TEST</mode>
  <time>1132843641</time>
</Response>
```

2.7.5 Fexco Dynamic Currency Conversion

2.7.5.1 Summary of service

This service allows merchants to take international payments via Fexco's third party Dynamic Currency Conversion (herin referred to as DCC) product.

Transactions placed using the DCC service can charge a bank card in the currency of the account that the cardholder uses rather than the currency of the merchant processing the txn. The amount changed to the customer is calculated using exchange rates provided by Fexco to the merchant.

2.7.5.2 Schema Elements for Request

Element Name: FexcoDCC
Position: Request.Transaction.TxnDetails

Elements of FexcoDCC			
Element Name	Description	Values / Limitations	Required
BaseCurrency	The currency that the merchant usually trades in	Valid ISO 4217 alphabetic currency code	R
BaseAmount	The amount that the merchant has requested for the transaction		R
ExchangeRate	The exchange rate provided by FEXCO for the transaction	Must contain a decimal point, and contain no more than 5 decimal places	R

Whenever the FexcoDCC block is present in an XML transaction, the currency attribute of Request.Transaction.TxnDetails.amount is mandatory.

Note that Fexco DCC transactions can only be supplied in bank card transactions where the transaction method is either "auth" or "pre".

The Fexco DCC service is compatiable with both DataCash MPI and Third Party MPI 3-D Secure transactions. In order to use the Fexco DCC service with these services, the FexcoDCC element must only be supplied in the initial stage (Where the transaction method is "auth" or "pre").

In the case of Airlines transactions, the per-passenger ticket prices can be in either the DCC (Post conversion) currency or the Base (Pre conversion) currency. No cross check on the totalling of these values against the transaction amount will be made for DCC transactions in the DPG.

2.7.5.3 Example Transaction

The following is an example authorisation. Note that the post conversion amount and currency is specified in the amount element.

For example:

Example of a Fexco DCC Authorisation Request

```
<Request>
  <Authentication>
    <client>xxxxxxxx</client>
    <password>xxxxxxxx</password>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="USD">200.00</amount>
      <merchantreference>12345601</merchantreference>
      <capturemethod>ecomm</capturemethod>

      <FexcoDCC>
        <BaseCurrency>GBP</BaseCurrency>
        <BaseAmount>100.00</BaseAmount>
        <ExchangeRate>2.0000</ExchangeRate>
      </FexcoDCC>

    </TxnDetails>
    <CardTxn>
      <method>auth</method>
      <Card>
        <pan>xxxxxxxxxxxxxxxx</pan>
        <expirydate>12/15</expirydate>
      </Card>
    </CardTxn>
  </Transaction>
</Request>
```

2.7.6 Bankcard Query Transactions

2.7.6.1 Service Overview

The Query Transaction service makes it possible to retrieve details of a previous Bankcard transaction by sending a request to the DataCash Payment Gateway. In addition to regular bankcard transactions, the service may be used to retrieve details for Fuels, RBS Giftcard and PPT transactions.

No additional subscription is required by merchants to use this service.

See also:

- BatchInput Query Transactions – section 2.5.2.3
- HCC Query Transactions – section 2.2.9.4.2
- HPS Query Transactions – section 2.2.10

Additional uses for the `query` transaction are outlined in the FAQ:

https://datacash.custhelp.com/cgi-bin/datacash.cfg/php/enduser/std_adp.php?p_faqid=1037

2.7.6.2 Schema Elements For Request

- Request
 - Authentication – [section 2.2.1.1](#)
 - Transaction
 - HistoricTxn – [see section 2.5.2.3.2.1](#).

Elements of HistoricTxn		
Element Name	Description	Values / Limitations
method	The transaction type	query R
reference	The reference of the transaction to be queried	DataCash or Merchant Reference of a previous Bankcard transaction on the same client. If the Merchant Reference is used, only the details for the most recent transaction with that reference will be returned. R

Attributes for Elements of HistoricTxn			
Attribute Name	Attribute of Element	Values / Limitations	
type	reference	datacash (default) merchant	O

Example XML Request for HistoricTxn complex elements

```

<HistoricTxn>
    <reference>3900900100010001</reference>
    <method>query</method>
</HistoricTxn>

<HistoricTxn>
    <reference type="merchant">abc12345678</reference>
    <method>query</method>
</HistoricTxn>

```

2.7.6.3 XML Example - Bankcard Query Transaction

Example XML Bankcard Query

```

<Request>
    <Authentication>
        <password>*****</password>
        <client>99000001</client>
    </Authentication>
    <Transaction>
        <HistoricTxn>
            <method>query</method>
            <reference>3900900100010001</reference>
        </HistoricTxn>
    </Transaction>
</Request>

```

2.7.6.4 Schema Elements for Bankcard Query Response

Element Name: QueryTxnResult
Position: Response

The QueryTxnResult complex element contains the retrieved details for the transaction reference provided in the request. The contents of the following elements all relate to the original transaction being queried, and not the query transaction itself.

Elements of QueryTxnResult	
Element Name	Description
Card	Complex element containing details of the payment card used
ThreeDSecure	Complex element containing 3-D Secure details from the original transaction – only present where the original transaction used 3-D Secure
acquirer	Acquiring bank
authcode	Bank authorisation response
datacash_reference	DataCash reference
environment	Processing environment
fulfill_date	Date of fulfillment
merchant_reference	Merchant reference
reason	DataCash response message. For a 'duplicate transaction' reason this will also contain the DataCash reference of the original, which may then be queried.
sent	Sent status – indicates whether the transaction was sent for settlement
status	DataCash response code
transaction_date	Date of processing by DPG

Elements of Card	
Element Name	Description
expirydate	Expiry date
issuemumber	Issue number or start date if available
pan	Card number (masked)

Elements of ThreeDSecure	
Element Name	Description
eci	ECI
security_code	CAVV
transactionID	XID

Example XML Response for QueryTxnResult complex elements

```
<QueryTxnResult>
  <Card>
    <expirydate>1214</expirydate>
    <pan>444433*****1111</pan>
  </Card>
  <acquirer>natwest</acquirer>
  <authcode>100000</authcode>
  <datacash_reference>4000900012345671</datacash_reference>
  <environment>ecommm</environment>
  <fulfill_date>2009-12-08 11:35:48</fulfill_date>
  <merchant_reference>123401</merchant_reference>
  <reason>ACCEPTED</reason>
  <sent>1</sent>
  <status>1</status>
  <transaction_date>2009-12-08 11:35:48</transaction_date>
</QueryTxnResult>

<QueryTxnResult>
  <Card>
    <expirydate>0120</expirydate>
    <issuenumber>0101</issuenumber>
    <pan>633300*****0005</pan>
  </Card>
  <ThreeDSecure>
    <eci>11</eci>
    <security_code>Q0FWVkBv1ZDQVZWQ0FWVkBv1Y=</security_code>
    <transactionID>MDAwMDAwMDAwMTIzNDU2NzE=</transactionID>
  </ThreeDSecure>
  <acquirer>natwest</acquirer>
  <authcode>100000</authcode>
  <datacash_reference>4000900012345671</datacash_reference>
  <environment>ecommm</environment>
  <fulfill_date>2009-12-08 11:56:53</fulfill_date>
  <merchant_reference>4421000008</merchant_reference>
  <reason>ACCEPTED</reason>
  <sent>1</sent>
  <status>1</status>
  <transaction_date>2009-12-08 11:56:53</transaction_date>
</QueryTxnResult>
```

2.7.6.5 XML Example - Bankcard Query Responses

Example XML Response for successful Query Txn

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <QueryTxnResult>
    <Card>
      <expirydate>0120</expirydate>
      <issuenumber>0101</issuenumber>
      <pan>633300*****0005</pan>
    </Card>
    <ThreeDSecure>
      <eci>11</eci>
      <security_code>Q0FWVkBVB1ZDQVZWQ0FWVkBVB1Y=</security_code>
      <transactionID>MDAwMDAwMDAwMTIzNDU2NzE=</transactionID>
    </ThreeDSecure>
    <acquirer>natwest</acquirer>
    <authcode>100000</authcode>
    <datacash_reference>4000900012345671</datacash_reference>
    <environment>ecommm</environment>
    <fulfill_date>2009-12-08 11:56:53</fulfill_date>
    <merchant_reference>4421000008</merchant_reference>
    <reason>ACCEPTED</reason>
    <sent>1</sent>
    <status>1</status>
    <transaction_date>2009-12-08 11:56:53</transaction_date>
  </QueryTxnResult>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1260273415</time>
</Response>
```

Example XML Response for unsuccessful Query Txn

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <datacash_reference>4100900012345675</datacash_reference>
  <information>The transaction you are querying
  cannot be located</information>
  <mode>TEST</mode>
  <reason>Cannot locate transaction to query</reason>
  <status>274</status>
  <time>1260274296</time>
</Response>
```

2.7.7 vTID Configuration Transactions

2.7.7.1 Service Overview

The vTID Configuration service allows for the changing or generation of a new password for a vTID by submitting a transaction to the DataCash Payment Gateway. This service is constrained to transactions received from IP addresses specified in vTID IP filtering for the vTID being configured.

No additional subscription is required by merchants to use this service.

2.7.7.2 Schema Elements For Request

- Request
 - Authentication – [section 2.1.1.2](#)
 - Transaction
 - VtidConfigurationTxn
 - Password

Elements of VtidConfigurationTxn		
Element Name	Description	Values / Limitations
method	The transaction type	Vtidconfiguration R

Elements of Password		
Element Name	Description	Values / Limitations
new_password	The password to be assigned to the vTID.	<p>May be blank in order to have the DataCash Payment Gateway generate a vTID password.</p> <p>Alternatively, a password may be specified and must meet the following specifications.</p> <ul style="list-style-type: none"> • More than 8 characters • Made up of a combination of at least 2 of lowercase, uppercase, numeric and special characters. • Dissimilar to the current password. • Not too simplistic or repetitive. • Not contain whole or partial words from the dictionary. R

Example XML Request for VtidConfigurationTxn with specified password

```
<VtidConfigurationTxn>
  <method>vtidconfiguration</method>
  <Password>
    <new_password>#####</new_password>
  </Password>
</VtidConfigurationTxn>
```

Example XML Request for VtidConfigurationTxn for password generation

```
<VtidConfigurationTxn>
  <method>vtidconfiguration</method>
  <Password>
    <new_password/>
  </Password>
</VtidConfigurationTxn>
```

2.7.7.3 Schema Elements for Response

- Response

Elements of Response	
Element Name	Description
new_password	Contains the specified or generated password as acknowledgement that the password has been modified.

Example XML Response for VtidConfigurationTxn

```
<Response>
  <datacash_reference>4000900012345671</datacash_reference>
  <merchantreference>123400</merchantreference>
  <new_password>#####</new_password>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1286959241</time>
</Response>
```

2.7.8 PreCall Transaction

2.7.8.1 Service Overview

The PreCall service analyses the input field (detailed below) and applies a set of rules to this information in order to determine the payment methods that are available for this particular transaction. The supported methods of payment are returned in the response. The rules applied to the input are configurable. Contact DataCash for more information regarding this service and to discuss the configuration of these rules.

2.7.8.2 Schema Elements for Request

2.7.8.2.1 Authentication

Element Name: Authentication
Position: Request

Elements of Authentication	
Element Name	Description
client	This is the vTID which has been configured for you by DataCash support.
password	This is the SHA1 hash, generated using the shared secret agreed with DataCash support. The SHA1 hash is generated by concatenating your vTID, client_reference and shared secret together and hashing the resultant string using SHA1.

2.7.8.2.2 Txn Details

Element Name: TxnDetails
Position: Request.Transaction

Elements of TxnDetails	
Element Name	Description
amount	This is the decimal amount of the transaction. The currency attribute must be provided as a 3 letter currency code. (See Example XML)
merchantreference	This is your unique reference for the transaction which can be up to 30 characters.

2.7.8.2.3 PreCall Txn

Element Name:	PreCallTxn
Position:	Request.Transaction

Elements of PreCallTxn	
Element Name	Description
method	Currently, the only supported method is “precall”, and you should populate this field with the latter value.

2.7.8.2.4 Client Data

Element Name:	ClientData
Position:	Request.Transaction.PreCallTxn

Elements of ClientData	
Element Name	Description
client_reference	This is your reference for the PreCall transaction which can be used as part of the rule application process.
client_order_id	This is your reference of the customer order for which this transaction is being submitted.
client_billing_id	This is your client billing id reference.
client_pos	This is the point of sale or office identifier for the location in which the transaction is taking place.
client_country	The 2 letter country code in which the transaction is taking place.
client_site	The URI for the website from which the transaction originated. E.g. www.example.com

2.7.8.2.5 Custom Client Fields

Element Name: CustomClientFields
Position: Request.Transaction.PreCallTxn.ClientData

This block can be used to provide additional / custom client data on agreement with DataCash support.

Elements of CustomClientFields	
Element Name	Description
reference_1	To be agreed with DataCash.
reference_2	To be agreed with DataCash.
reference_3	To be agreed with DataCash.
reference_4	To be agreed with DataCash.

2.7.8.2.6 Order Details

Element Name: OrderDetails
Position: Request.Transaction.PreCallTxn

Elements of OrderDetails	
Element Name	Description
order_date_time	This is the date and time when the order took place. This should be presented as an ISO8601 formatted timestamp. E.g. 2011-01-14T14:22:23+01:00
delivery_date_time	This is the date and time when the goods/service will be delivered. If this transaction is related to the sale of Airline tickets, this is the time of flight. This should be presented as an ISO8601 formatted timestamp. E.g. E.g. 2011-01-14T14:22:23+01:00
billing_country	This is the country in which the customer will be billed.
customer_logged_in	Indicates whether or not the customer is an existing account holder, and was logged into this account at the time of the transaction.
sales_channel	The channel through which this transaction took place. E.g. ecommerce.
customer_ip_address	The IP address of the customer.
customer_language	The customers preferred language.

2.7.8.2.7 Airline Details

Element Name: AirlineDetails
Position: Request.Transaction.PreCallTxn

Elements of AirlineDetails	
Element Name	Description
airline_country	This is the two letter country code of the country in which the customer will be billed.
airline_booking_type	E.g. BOOK
customer_level	The customer loyalty level.
Legs	Contains up to 16 Leg elements representing each individual Leg of a journey.

2.7.8.2.8 Legs

The Legs complex element can contain up to 16 Leg elements. Each Leg element represents an individual leg of the overall journey. The Leg element is described below:

Element Name: Leg
Position: Request.Transaction.PreCallTxn.AirlineDetails.Legs

Elements of Leg	
Element Name	Description
depart_airport	This is the 3 letter IATA airport code for the departure airport.
depart_city	This is the 3 letter city code for the departure city.
depart_state	The departure state if applicable.
depart_country	This is the two letter country code for the departure country.
carrier	This is the 2 letter IATA airline code for the carrier.
arrival_airport	This is the 3 letter IATA airport code for the arrival/destination airport.
arrival_city	This is the 3 letter city code for the arrival/destination city.
arrival_state	The arrival/destination state if applicable.
arrival_country	This is the two letter country code for the arrival/destination country.

2.7.8.3 Example Request / Response XML

The following represents an example PreCall request and response.

Example XML Request for PreCallTxn

```

<Request>
  <Authentication>
    <password
      type="SHA1">df855a23df855a23df855a23df855a23df855a23</password>
      <client>88008800</client>
    </Authentication>
    <Transaction>
      <TxnDetails>
        <amount currency="EUR">1000.00</amount>
        <merchantreference>00001239</merchantreference>
      </TxnDetails>
      <PreCallTxn>
        <method>precall</method>
        <ClientData>
          <client_reference>00001239</client_reference>
          <client_order_id>order-15-2011</client_order_id>
          <client_billing_id>cust12358</client_billing_id>
          <client_pos>FRALH010Z</client_pos>
          <client_country>IT</client_country>
          <client_site>agent.com</client_site>
          <CustomClientFields>
            <reference_1>xyz</reference_1>
            <reference_2>xyz</reference_2>
            <reference_3>xyz</reference_3>
            <reference_4>xyz</reference_4>
          </CustomClientFields>
        </ClientData>
        <OrderDetails>
          <order_date_time>2011-01-14T14:22:23+01:00</order_date_time>
          <delivery_date_time>2011-03-
            25T14:30:00+01:00</delivery_date_time>
          <billing_country>DE</billing_country>
          <customer_logged_in>true</customer_logged_in>
          <sales_channel>ecommerce</sales_channel>
          <customer_ip_address>112.118.128.253</customer_ip_address>
          <customer_language>fr</customer_language>
        </OrderDetails>
        <AirlineDetails>
          <airline_country>DE</airline_country>
          <airline_booking_type>BOOK</airline_booking_type>
          <customer_level>SEN</customer_level>
          <Legs>
            <Leg>
              <depart_airport>JFK</depart_airport>
              <depart_city>NYC</depart_city>
              <depart_state>NY</depart_state>
              <depart_country>US</depart_country>
              <carrier>LH</carrier>
              <arrival_airport>HAM</arrival_airport>
              <arrival_city>HH</arrival_city>
              <arrival_state>HH</arrival_state>
              <arrival_country>DE</arrival_country>
            </Leg>
          </Legs>
        </AirlineDetails>
      </PreCallTxn>
    </Transaction>
  </Authentication>
</Request>

```

```

    </PreCallTxn>
  </Transaction>
</Request>

```

Example XML Response for PreCallTxn

```

<Response>
  <PreCallTxn>
    <ClientData>
      <client_reference>00001239</client_reference>
      <client_order_id>order-15-2011</client_order_id>
      <client_billing_id>cust12358</client_billing_id>
      <CustomClientFields>
        <reference_1>xyz</reference_1>
        <reference_2>xyz</reference_2>
        <reference_3>xyz</reference_3>
        <reference_4>xyz</reference_4>
      </CustomClientFields>
    </ClientData>
    <PaymentMethods>
      <payment_method type="wallet" valid_thru="2011-01-
        14T22:50:00+01:00" airline_fee_level="A001">PL</payment_method>
      <payment_method type="creditCard" valid_thru="2011-01-
        14T22:50:00+01:00" airline_fee_level="A002">CC</payment_method>
    </PaymentMethods>
  </PreCallTxn>
  <datacash_reference>4600900012345673</datacash_reference>
  <merchantreference>00001239</merchantreference>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>_UNIX_TIME_</time>
</Response>

```

2.8 CARD TOKENIZATION

2.8.1 Introduction

For further details on Tokenization transactions and their use, please refer to the Tokenization transaction appendix 2.27 in the Datacash Developers Guide Appendices

And the website: <http://www.datacash.com/gettingproducts.php?id=Card-Tokenization>

2.8.2 Token generation

Tokens can be generated in two ways.

2.8.2.1 Tokenize transaction

By issuing a tokenize transaction:

Element Name:	TokenizeTxn
Position:	Request.Transaction

Elements of TxnDetails			
Element Name	Description	Values/Limitations	
pan	Card Number	must be a valid card number between 13 and 19 digits in length	R
method	The type of Token generation	tokenize retokenize	R
shared_secret	Overrides the VTIDs shared secret	20 alphanumeric characters	O

Attributes for Elements of TxnDetails			
Attribute Name	Attribute of Element	Values / Limitations	
type	pan	Token – used to indicate a token is being provided. Used in “retokenize” transactions.	O

2.8.2.2 Standard transaction

MERCHANTS that are signed up to the Tokenization service will receive the "token" node in response to normal card transactions. This provided token can be used as if generated by the specific tokenize transaction

Example XML for TxnDetails complex elements:

Tokenize transaction

```
<?xml version="1.0"?>
<Request>
  <Transaction>
    <TxnDetails>...</TxnDetails>
    <TokenizeTxn>
      <method>tokenize</method>
      <Card>
        <pan>#####</pan>
      </Card>
    </TokenizeTxn>
  </Transaction>
  <Authentication>...</Authentication>
</Request>
```

Response

```
<Response>
  ...
  <datacash_reference>3900100821792328</datacash_reference>
  <CardTxn>
    <card_scheme>Visa</card_scheme>
    <country>France</country>
    <issuer>Credit Lyonnais</issuer>
    <token>8466A4B11D4F34227E50FF51A6B9CA946B4871FB</token>
  </CardTxn>
  ...
</Response>
```

2.9 SWITCH TO ISSUER TRANSACTIONS

2.9.1 Versioning

All Switch to Issuer transactions are version 2.

2.9.2 Standard transaction - response elements

When submitting a standard transaction through Switch to Issuer, five extra response elements are provided by the DPG. These are:

Elements of Response	
Element Name	Description
response_code	Contains the response code returned by Switch to Issuer.
response_text	Contains the response message returned by Switch to Issuer.
aiic	Acquiring Institution ID Code – 6 digits
stan	System Trace Audit Number – 6 digits
rrn	Retrieval Reference Number – 12 digits

Example XML:

Response

```
<Response version="2">
  ...
  <datacash_reference>3900100821792328</datacash_reference>
  <CardTxn>
    ...
    <response_code>0</response_code>
    <response_text>Approved or completed successfully</response_text>
  </CardTxn>
  ...
  <aiic>55555</aiic>
  <stan>000001</stan>
  <rrn>115010000001</rrn>
</Response>
```

2.9.3 Query transaction – response elements

When making a query transaction against Switch to Issuer, three new response elements are provided by the DPG. These are:

Elements of Response	
Element Name	Description
aiic	Acquiring Institution ID Code – 6 digits
stan	System Trace Audit Number – 6 digits
rrn	Retrieval Reference Number – 12 digits

Example XML:

Query Response

```
<Response version="2">
  ...
  <QueryTxnResult>
    <datacash_reference>3900100821792328</datacash_reference>
    ...
  </QueryTxnResult>
  ...
  <aiic>55555</aiic>
  <stan>000001</stan>
  <rrn>115010000001</rrn>
</Response>
```

2.9.4 Transactions including CSC data

For Switch to Issuer, CSC (CV2) data is to be supplied in an alternative structure, under a CSC block (part of CardTxn).

As well as the CSC (CV2) number, a policy can optionally be supplied, or an ExtendedPolicy (unlike non-Switch to Issuer transactions, CSC.ExtendedPolicy can only contain a single element, csc_policy).

Example XML:

```
<Request version="2">
  ...
  <CardTxn>
    <Card>
      <CSC>
        <csc>123</csc>
      </CSC>
    </Card>
  ...
</CardTxn>
...
</Request>

<Request version="2">
  ...
  <CardTxn>
    <Card>
      <CSC>
        <csc>123</csc>
        <policy>123</policy>
      </CSC>
    </Card>
  ...
</CardTxn>
...
</Request>

<Request version="2">
  ...
  <CardTxn>
    <Card>
      <CSC>
        <csc>123</csc>
        <ExtendedPolicy>
          <csc_policy notprovided='reject' notchecked='reject'
matched='accept' notmatched='reject' partialmatch='reject' />
        </ExtendedPolicy>
      </CSC>
    </Card>
  ...
</CardTxn>
...
</Request>
```

2.9.5 *Transactions including AVS data*

For Switch to Issuer, AVS data is to be supplied in an alternative structure, under an AVS block (part of CardTxn).

As well as the address data, a policy can optionally be supplied, or an ExtendedPolicy (unlike non-Switch to Issuer transactions, AVS.ExtendedPolicy can only contain the two elements, postcode_policy and address_policy).

Example XML:

```
<Request version="2">
  ...
  <CardTxn>
    <Card>
      <AVS>
        <street_address1>1 High Street</street_address1>
        <street_address2>Town</street_address2>
        <street_address3>County</street_address3>
        <street_address4>United Kingdom</street_address4>
        <postcode>AA1 1AA</postcode>
      </AVS>
    </Card>
    ...
  </CardTxn>
  ...
</Request>

<Request version="2">
  ...
  <CardTxn>
    <Card>
      <AVS>
        <street_address1>1 High Street</street_address1>
        <street_address2>Town</street_address2>
        <street_address3>County</street_address3>
        <street_address4>United Kingdom</street_address4>
        <postcode>AA1 1AA</postcode>
        <policy>1</policy>
      </AVS>
    </Card>
    ...
  </CardTxn>
  ...
</Request>
```

```
<Request version="2">
  ...
  <CardTxn>
    <Card>
      <AVS>
        <street_address1>1 High Street</street_address1>
        <street_address2>Town</street_address2>
        <street_address3>County</street_address3>
        <street_address4>United Kingdom</street_address4>
        <postcode>AA1 1AA</postcode>

        <ExtendedPolicy>
          <postcode_policy notprovided='reject' notchecked='reject'
            matched='accept' notmatched='reject' partialmatch='reject' />
          <address_policy notprovided='reject' notchecked='reject'
            matched='accept' notmatched='reject' partialmatch='reject' />
        </ExtendedPolicy>
      </AVS>
    </Card>
    ...
  </CardTxn>
  ...
</Request>
```

[Back to Contents](#)

2.11 LOCAL DIRECT DEBIT

2.11.1 Service Overview

In Germany the ELV system (Elektronisches Lastschriftverfahren - Direct Debit) is the most popular method of payment among customers in an online environment. The ELV process is a simple one and can be utilised by anyone who is in possession of a German cheque account. This payment method allows the customer to authorise a merchant to withdraw money directly from their bank account.

Schema Elements for Request

- Request
 - Authentication – section 2.1.1.2
 - Transaction
 - LocalDDTxn – section 2.11.2
 - TxnDetails – section 2.11.3

The following keys will be used for each transaction type:

- B – localdd_buy
- A – localdd_authorise
- C – localdd_capture
- L – localdd_cancel
- R – localdd_refund

Each field will be labelled with the following key:

- O – Optional.
- R – Required, field must be presented.
- X – Excluded, presenting this field will cause the transaction to fail.

2.11.2 LocalDDTxn

Element Name:	LocalDDTxn
Position(s)	Request.Transaction

Attributes of LocalDDTxn							
Attribute Name	Attribute of Element	Values / Limitations	B	A	C	L	R
type	LocalDDTxn	Currently this should only be 'elv'.	R	R	X	X	X

Elements of LocalDDTxn							
Element Name	Description	Values/ Limitations	B	A	C	L	R

billing_name	The name of the account holder	Alphanumeric	R	R	X	X	X
method	Determines the processing model to be used	localdd_buy - for one stage processing localdd_authorise - for two stage processing	R	R	X	X	X
profile	The respective profile controls which payment and transaction methods can be selected.	Numeric	R	R	X	X	X
customer_ip	The customer's IP address.	Valid IP address	R	R	X	X	X

One of the following payment elements must be included. Please note, sortcode and accountnumber are supplied as a pair.

Elements of LocalDDTxn		
Attribute Name	Description	Values / Limitations
sortcode	The sort code of the customer's bank	Six digits
accountnumber	The customer's account number	Eight digits
iban	An International Bank Account Number. Standard for identifying bank accounts across national borders.	A valid IBAN
paymentreference	Merchant configurable data field typically used to identify a customer uniquely within the system, regardless of the amount of accounts that the customer has.	A valid Payment Reference

When the method in use is localdd_buy one of the following elements is required.

Elements of LocalDDTxn		
Attribute Name	Description	Values / Limitations
delaydays	The number of days on which the direct debit must be actioned.	Numeric
debitdate	The date on which the direct debit must be auctioned.	yyyy-mm-dd

Example XML for LocalDDTxn complex elements:
localdd_buy

```
<LocalDDTxn>
  <customer_ip>192.168.1.1</customer_ip>
  <profile>1</profile>
  <method>localdd_buy</method>
  <delaydays>2</delaydays>
  <sortcode>123456</sortcode>
  <accountnumber>87654321</accountnumber>
  <billing_name>A Customer</method>
</LocalDDTxn>
```

localdd_authorise

```
<LocalDDTxn>
  <customer_ip>192.168.1.1</customer_ip>
  <profile>1</profile>
  <method>localdd_authorise</method>
  <sortcode>123456</sortcode>
  <accountnumber>87654321</accountnumber>
  <billing_name>A Customer</method>
</LocalDDTxn>
```

2.11.3 TxnDetails

Element Name:	TxnDetails
Position:	Request.Transaction

Elements of TxnDetails							
Element Name	Description	Values/ Limitations	B	A	C	L	R
merchantreference	A unique reference number for each transaction	Min 6, max 30 alphanumeric characters . Must be unique .	R	R	X	X	X
amount	The value of the transaction		R	R	O	O	O

Attributes for Elements of TxnDetails							
Attribute Name	Attribute of Element	Values / Limitations	B	A	C	L	R
currency	amount	Currency is passed in the 3 character ISO 4217 Alphabetic format e.g. GBP, USD, EUR.	R	R	X	X	X

Example XML for TxnDetails complex elements:

Transaction

```
<TxnDetails>
    <merchantreference>myreference123459988</merchantreference>
    <amount currency="EUR">146.99</amount>
</TxnDetails>
```

Historic Transaction

```
<TxnDetails>
    <amount>146.99</amount>
</TxnDetails>
```

2.11.4 HistoricTxn

This element is the direct equivalent of the Card element (for transactions using card details).

Element Name:	HistoricTxn
Position:	Request.Transaction

Elements of HistoricTxn					
Element Name	Description	Values / Limitations	C	L	R
reference	DataCash unique reference of the original transaction	must be a valid transaction	R	R	R
method	The transaction type	localdd_capture localdd_cancel localdd_refund	R	R	R

Example XML Request for HistoricTxn complex elements

```
<HistoricTxn>
    <reference>4100200039275407</reference>
    <method>localdd_capture</method>
</HistoricTxn>
```

Successful localdd_authorise

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <LocalDDTxn type="elv">
    <apg_reference_id>150968437</apg_reference_id>
    <apg_response_code>100</apg_response_code>
    <apg_response_desc>Transaction successfull</apg_response_desc>
    <method>localdd_authorise</method>
  </LocalDDTxn>
  <datacash_reference>3300900010000208</datacash_reference>
  <merchantreference>1c464467c61b2321</merchantreference>
  <mode>LIVE</mode>
  <reason>The APG accepted the transaction</reason>
  <status>1448</status>
  <time>1321541659</time>
</Response>
```

Successful localdd_capture

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <LocalDDTxn type="elv">
    <apg_payment_id>DD1354319267000</apg_payment_id>
    <apg_reference_id>150968507</apg_reference_id>
    <apg_response_code>100</apg_response_code>
    <apg_response_desc>Transaction successfull</apg_response_desc>
    <method>localdd_capture</method>
  </LocalDDTxn>
  <datacash_reference>3300900010000208</datacash_reference>
  <merchantreference>1c464467c61b2321</merchantreference>
  <mode>LIVE</mode>
  <reason>CAPTURED OK</reason>
  <status>1</status>
  <time>1321541698</time>
</Response>
```

Unsuccessful localdd_buy

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <LocalDDTxn type="elv">
    <method>localdd_buy</method>
  </LocalDDTxn>
  <datacash_reference>3900900010000210</datacash_reference>
  <information>Field 'delaydays' or 'debitdate' are required for localdd_buy txns</information>
  <merchantreference>94ed40155dd418f6</merchantreference>
  <mode>LIVE</mode>
  <reason>Field not supported in APG transaction</reason>
  <status>1442</status>
  <time>1321541775</time>
</Response>
```

Unsuccessful localdd_capture

```
<?xml version="1.0" encoding="UTF-8"?>
<Response>
  <LocalDDTxn type="elv">
    <apg_error_code>211</apg_error_code>
    <apg_error_description>Modification.DPGReferenceId refers to
transaction that cannot be captured</apg_error_description>
    <method>localdd_capture</method>
  </LocalDDTxn>
  <datacash_reference>3500900010000207</datacash_reference>
  <mode>LIVE</mode>
  <reason>The APG rejected the transaction</reason>
  <status>1447</status>
  <time>1321541644</time>
</Response>
```

[Back to Contents](#)

2.12 ALTERNATIVE PAYMENT METHODS

The DataCash Payment Gateway has built in support for the following Alternative Payment Methods:

- iDeal
- Sofort
- La Caixa
- Nordea
- Web Money
- MisterCash

If you are interested in using one or more of the supported alternative payment methods, please contact DataCash support.

The following sections detail the request / response XML elements used to process alternative payments using the DataCash Payment Gateway:

2.12.1 Common Request Elements

The following request elements are common across all of the alternative payment methods listed above. Billing information is optional (O), whereas all other common fields are required / mandatory (R).

Element Names:	IdealTxn SofortTxn LaCaixaTxn NordeaTxn WebMoneyTxn MisterCashTxn
Position:	Request.Transaction

Elements of [PaymentMethod]Txn			
Element Name	Description	Values/Limitations	
method	The type of request.	A constant value of "payment" (this is currently the only method supported)	R
language	The customer's language	IETF Language Tag	R
customer_email	The customer's E-Mail address.	64 Characters Maximum, must contain @ and .	R
customer_ip	The IPv4 address of the customer.	nnn.nnn.nnn.nnn	R
title	The customer's title/honorific.	E.g. Mr, Mrs, Sir, Miss, Lord	R
first_name	The customer's first name.	Alphanumeric.	R
last_name	The customer's last name / surname.	Alphanumeric.	R

country	The customer's country. In the case of an airlines transaction, this would usually be the country of departure.	(ISO 3166-1 Alpha-2)	R
billing_street1	The billing address (1 st line)	Alphanumeric	O
billing_street2	The billing address (2 nd line)	Alphanumeric	O
billing_city	The billing city	Alphanumeric	O
billing_postcode	The billing postal code	Alphanumeric	O
billing_state_province	The billing state / province	Alphanumeric	O
billing_country	The billing country.	Alphanumeric	O

2.12.2 iDeal Transactions

The following request elements are specific to iDeal transactions (in addition to the common elements described above).

Element Names: IdealTxn
 Position: Request.Transaction

Elements of IdealTxn			
Element Name	Description	Values/Limitations	
issuer_id	The issuer id	Alphanumeric	R
success_url	The success URL (customer will be redirected in the event of a successful payment)	Alphanumeric	R
failure_url	The failure URL (customer will be redirected in the event of a failed payment / error)	Alphanumeric	R

Example IdealTxn Request:

```

<Request>
  <Authentication>
    <password>#####</password>
    <client>14000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">119.76</amount>
      <merchantreference>830100000000259</merchantreference>
    </TxnDetails>
    <IdealTxn>
      <method>payment</method>
      <language>en</language>
      <customer_email>john@mail.com</customer_email>
      <customer_ip>127.0.0.1</customer_ip>
      <title>MR</title>
      <first_name>JOHN</first_name>
      <last_name>KENT</last_name>
      <country>FR</country>
      <billing_street1>Address Line 1</billing_street1>
      <billing_city>CA</billing_city>
      <billing_postcode>12345</billing_postcode>
      <billing_country>DE</billing_country>
      <issuer_id>2</issuer_id>
      <success_url>http://success</success_url>
      <failure_url>http://failure</failure_url>
    </IdealTxn>
  </Transaction>
</Request>

```

Example IdealTxn Response:

```

<Response>
  <IdealTxn>
    <RedirectParams>
      <redirect_param param_name="param1">value1</redirect_param>
      <redirect_param param_name="param2">value2</redirect_param>
    </RedirectParams>
    <apg_ref>123456789</apg_ref>
    <apg_status>REDIRECT</apg_status>
    <content_type>application/x-www-form-urlencoded</content_type>
    <display_message>Please wait</display_message>
    <http_method>POST</http_method>
    <redirect_url>http://redirecthost</redirect_url>
  </IdealTxn>
  <datacash_reference>420090300000481</datacash_reference>
  <merchantreference>830100000000259</merchantreference>
  <mode>TEST</mode>
  <reason>The APG accepted the transaction</reason>
  <status>1448</status>
  <time>1320327371</time>
</Response>

```

2.12.3 Sofort Transactions

The following request elements are specific to Sofort transactions (in addition to the common elements described above).

Element Names: SofortTxn
Position: Request.Transaction

Elements of SofortTxn			
Element Name	Description	Values/Limitations	
issuer_id	The issuer id	Alphanumeric	R
success_url	The success URL (customer will be redirected in the event of a successful payment)	Alphanumeric	R
failure_url	The failure URL (customer will be redirected in the event of a failed payment / error)	Alphanumeric	R

Example Sofort Request:

```
<Request>
  <Authentication>
    <password>#####</password>
    <client>14000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">119.76</amount>
      <merchantreference>830300000000258</merchantreference>
    </TxnDetails>
    <SofortTxn>
      <method>payment</method>
      <language>en</language>
      <customer_email>john@mail.com</customer_email>
      <customer_ip>127.0.0.1</customer_ip>
      <title>MR</title>
      <first_name>JOHN</first_name>
      <last_name>KENT</last_name>
      <country>FR</country>
      <billing_street1>Address Line 1</billing_street1>
      <billing_city>CA</billing_city>
      <billing_postcode>12345</billing_postcode>
      <billing_country>DE</billing_country>
      <profile>1</profile>
      <success_url>http://success</success_url>
      <failure_url>http://failure</failure_url>
    </SofortTxn>
  </Transaction>
</Request>
```

Example Sofort Response:

```
<Response>
  <SofortTxn>
    <RedirectParams>
      <redirect_param param_name="user_id">12345</redirect_param>
      <redirect_param param_name="project_id">56789</redirect_param>
      <redirect_param param_name="amount">119.76</redirect_param>
      <redirect_param param_name="currency_id">GBP</redirect_param>
      <redirect_param
param_name="reason_1">SU1234567890123</redirect_param>
        <redirect_param param_name="reason_2">12345</redirect_param>
        <redirect_param
param_name="user_variable_0">123456789</redirect_param>
        <redirect_param
param_name="user_variable_1">X4M75U</redirect_param>
        <redirect_param
param_name="user_variable_2">127.0.0.1</redirect_param>
        <redirect_param
param_name="user_variable_4">https://epi.expercash.net/3rdPartyNotifica
tion/su_success.php?taid=123456789</redirect_param>
        <redirect_param
param_name="user_variable_5">https://epi.expercash.net/3rdPartyNotifica
tion/su_failure.php</redirect_param>
        <redirect_param
param_name="hash">0123456789ABCDEF0123456789ABCDEF01234567</redirect_pa
ram>
    </RedirectParams>
    <apg_ref>123456789</apg_ref>
    <apg_status>REDIRECT</apg_status>
    <content_type>application/x-www-form-urlencoded</content_type>
    <http_method>GET</http_method>
    <redirect_url>http://testdb:8180/dc-apg-
simulator/payment</redirect_url>
  </SofortTxn>
  <datacash_reference>4600903000000479</datacash_reference>
  <merchantreference>830300000000258</merchantreference>
  <mode>TEST</mode>
  <reason>The APG accepted the transaction</reason>
  <status>1448</status>
  <time>1320327367</time>
</Response>
```

2.12.4 WebMoney Transactions

The following request elements are specific to WebMoney transactions (in addition to the common elements described above).

Element Names: WebMoneyTxn
Position: Request.Transaction

Elements of WebMoneyTxn			
Element Name	Description	Values/Limitations	
success_url	The success URL (customer will be redirected in the event of a successful payment)	Alphanumeric	R
failure_url	The failure URL (customer will be redirected in the event of a failed payment / error)	Alphanumeric	R

Example WebMoneyTxn Request:

```
<Request>
  <Authentication>
    <password>#####</password>
    <client>14000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">119.76</amount>
      <merchantreference>850100000000261</merchantreference>
    </TxnDetails>
    <WebMoneyTxn>
      <method>payment</method>
      <language>en</language>
      <customer_email>john@mail.com</customer_email>
      <customer_ip>127.0.0.1</customer_ip>
      <title>MR</title>
      <first_name>JOHN</first_name>
      <last_name>KENT</last_name>
      <country>FR</country>
      <billing_street1>Address Line 1</billing_street1>
      <billing_city>CA</billing_city>
      <billing_postcode>12345</billing_postcode>
      <billing_country>DE</billing_country>
      <success_url>http://success</success_url>
      <failure_url>http://failure</failure_url>
    </WebMoneyTxn>
  </Transaction>
</Request>
```

Example WebMoneyTxn Response:

```

<Response>
  <WebMoneyTxn>
    <RedirectParams>
      <redirect_param
param_name="LMI_PAYMENT_AMOUNT">119.76</redirect_param>
      <redirect_param
param_name="LMI_PAYEE_PURSE">pursesname</redirect_param>
        <redirect_param param_name="LMI_PAYMENT_NO">payment
number</redirect_param>
      <redirect_param param_name="LMI_SIM_MODE">sim
mode</redirect_param>
    </RedirectParams>
    <apg_ref>123456789</apg_ref>
    <apg_status>REDIRECT</apg_status>
    <content_type>application/x-www-form-urlencoded</content_type>
    <display_message>Please wait while you are redirected to
WebMoney</display_message>
    <http_method>GET</http_method>
    <redirect url>http://webmoneyurl</redirect url>
  </WebMoneyTxn>
  <datacash_reference>4300903000000485</datacash_reference>
  <merchantreference>8501000000000261</merchantreference>
  <mode>TEST</mode>
  <reason>The APG accepted the transaction</reason>
  <status>1448</status>
  <time>1320327377</time>
</Response>

```

2.12.5 MisterCash Transactions

The following request elements are specific to MisterCash transactions (in addition to the common elements described above).

Element Names:	MisterCashTxn
Position:	Request.Transaction

Elements of MisterCashTxn			
Element Name	Description	Values/Limitations	
success_url	The success URL (customer will be redirected in the event of a successful payment)	Alphanumeric	R

failure_url	The failure URL (customer will be redirected in the event of a failed payment / error)	Alphanumeric	R
-------------	---	--------------	---

Example MisterCashTxn Request:

```

<Request>
  <Authentication>
    <password>#####</password>
    <client>14000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">119.76</amount>
      <merchantreference>830200000000260</merchantreference>
    </TxnDetails>
    <MisterCashTxn>
      <method>payment</method>
      <language>en</language>
      <customer_email>john@mail.com</customer_email>
      <customer_ip>127.0.0.1</customer_ip>
      <title>MR</title>
      <first_name>JOHN</first_name>
      <last_name>KENT</last_name>
      <country>FR</country>
      <billing_street1>Address Line 1</billing_street1>
      <billing_city>CA</billing_city>
      <billing_postcode>12345</billing_postcode>
      <billing_country>DE</billing_country>
      <success_url>http://success</success_url>
      <failure_url>http://failure</failure_url>
    </MisterCashTxn>
  </Transaction>
</Request>

```

Example MisterCashTxn Response:

```
<Response>
  <MisterCashTxn>
    <RedirectParams>
      <redirect_param
param_name="merchantID">123456789</redirect_param>
      <redirect_param param_name="customerRef">X4M75U</redirect_param>
      <redirect_param param_name="country">UK</redirect_param>
      <redirect_param param_name="currency">GBP</redirect_param>
      <redirect_param param_name="recieveCurrency">GBP</redirect_param>
      <redirect_param param_name="email">john@mail.com</redirect_param>
      <redirect_param param_name="service">MISTERCASH</redirect_param>
      <redirect_param param_name="language">en</redirect_param>
    </RedirectParams>
    <apg_ref>123456789</apg_ref>
    <apg_status>REDIRECT</apg_status>
    <content_type>application/x-www-form-urlencoded</content_type>
    <display_message>Please wait while you are redirected to the Envoy
OneClick site</display_message>
    <http method>GET</http method>
    <redirect_url>http://envoy</redirect_url>
  </MisterCashTxn>
  <datacash_reference>4800903000000483</datacash_reference>
  <merchantreference>830200000000260</merchantreference>
  <mode>TEST</mode>
  <reason>The APG accepted the transaction</reason>
  <status>1448</status>
  <time>1320327373</time>
</Response>
```

2.12.6 La Caixa Transactions

Only the common elements described above are applicable to La Caixa transactions.

Element Names:	LaCaixaTxn
Position:	Request.Transaction

Example LaCaixaTxn Request:

```
<Request>
  <Authentication>
    <password>#####</password>
    <client>14000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">119.76</amount>
      <merchantreference>840100000000256</merchantreference>
    </TxnDetails>
    <LaCaixaTxn>
      <method>payment</method>
      <language>es</language>
      <customer_email>john@mail.com</customer_email>
      <customer_ip>127.0.0.1</customer_ip>
      <title>MR</title>
      <first_name>JOHN</first_name>
      <last_name>KENT</last_name>
      <country>FR</country>
      <billing_street1>Address Line 1</billing_street1>
      <billing_city>CA</billing_city>
      <billing_postcode>12345</billing_postcode>
      <billing_country>DE</billing_country>
    </LaCaixaTxn>
  </Transaction>
</Request>
```

Example LaCaixaTxn Response:

```
<Response>
  <LaCaixaTxn>
    <apg_account_name>Merchant account name</apg_account_name>
    <apg_account_number>1234567890</apg_account_number>
    <apg_bank>Bank of Bankeria</apg_bank>
    <apg_branch>BranchName</apg_branch>
    <apg_branch_address>123 Bank St, Banktown, Republic of
Bankeria</apg_branch_address>
    <apg_iban>RB01 CBCO 1234 1234 1234 56</apg_iban>
    <apg_ref>123456789</apg_ref>
    <apg_status>PENDING</apg_status>
    <apg_swift>CBCORB00XXX</apg_swift>
    <display_message>Display message</display_message>
  </LaCaixaTxn>
  <datacash_reference>4400903000000475</datacash_reference>
  <merchantreference>840100000000256</merchantreference>
  <mode>TEST</mode>
  <reason>The APG accepted the transaction</reason>
  <status>1448</status>
  <time>1320327363</time>
</Response>
```

2.12.7 Nordea Transactions

Only the common elements described above are applicable to Nordea transactions.

Element Names:	NordeaTxn
Position:	Request.Transaction

Example NordeaTxn Request:

```
<Request>
  <Authentication>
    <password>#####</password>
    <client>14000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">119.76</amount>
      <merchantreference>840200000000255</merchantreference>
    </TxnDetails>
    <NordeaTxn>
      <method>payment</method>
      <language>en</language>
      <customer_email>john@mail.com</customer_email>
      <customer_ip>127.0.0.1</customer_ip>
      <title>MR</title>
      <first_name>JOHN</first_name>
      <last_name>KENT</last_name>
      <country>FR</country>
      <billing_street1>Address Line 1</billing_street1>
      <billing_city>CA</billing_city>
      <billing_postcode>12345</billing_postcode>
      <billing_country>DE</billing_country>
    </NordeaTxn>
  </Transaction>
</Request>
```

Example NordeaTxn Response:

```

<Response>
  <NordeaTxn>
    <apg_account_name>Merchant account name</apg_account_name>
    <apg_account_number>1234567890</apg_account_number>
    <apg_bank>Bank of Bankeria</apg_bank>
    <apg_branch>BranchName</apg_branch>
    <apg_branch_address>123 Bank St, Banktown, Republic of
Bankeria</apg_branch_address>
    <apg_ibanc>RB01 CBCO 1234 1234 1234 56</apg_ibanc>
    <apg_ref>123456789</apg_ref>
    <apg_status>PENDING</apg_status>
    <apg_swift>CBCORB00XXX</apg_swift>
    <display_message>Display message</display_message>
  </NordeaTxn>
  <datacash_reference>4900903000000473</datacash_reference>
  <merchantreference>840200000000255</merchantreference>
  <mode>TEST</mode>
  <reason>The APG accepted the transaction</reason>
  <status>1448</status>
  <time>1320327362</time>
</Response>

```

2.12.8 Common Response Elements

The following response elements are common across all Alternative Payment Methods. The transaction type specific response fields will be contained in the [PaymentMethod]Txn complex element. For example, a response for a Sofort txn would contain a "Response.SofortTxn" block. See the examples above for more detail.

Element Names:	[PaymentMethod]Txn
Position:	Response

Elements of [PaymentMethod]Txn

Element Name	Description
apg_ref	The apg reference for this transaction.
apg_status	The status of this transaction.
display_message	The recommended display message that should be displayed to the customer.

2.12.9 Offline Bank Transfer (OBT) Response Elements

The following elements are specific to OBT payment methods:

- Nordea
- La Caixa

The response elements detailed below should be displayed to the customer to enable them to complete the payment offline. The APG reference detailed above should also be displayed.

Element Names: [OBTPaymentMethod]Txn
Position: Response

Elements of [PaymentMethod]Txn	
Element Name	Description
apg_account_name	The merchant account name
apg_account_number	The account number with the APG (Alternative Payment Gateway).
apg_bank	
apg_branch	The name of the branch.
apg_branch_address	The address of the branch.
apg_iban	The International Bank Account Number (IBAN)
apg_swift	The SWIFT code for the bank.

2.12.10 Redirect Response Elements

The following response elements are specific to redirect payment types (where the customer is redirected to a 3rd party processor in order to complete the payment). Supported redirect payment types are:

- iDeal
- Sofort
- Web Money
- MisterCash

Element Names: [RedirectPaymentMethod]Txn
Position: Response

Elements of [RedirectPaymentMethod]Txn	
Element Name	Description
redirect_url	This is a URL on the 3 rd party payment processors website to which the customer should be redirected.

http_method	This is the HTTP method that should be used to redirect the customer (GET or POST). In the case of a POST redirect, you might need to consider using JavaScript to automatically submit a form (and providing a button in the case where the customer has disabled / does not support JavaScript).
content_type	The content type that should be used.
RedirectParams	This complex element contains an arbitrary number of redirect parameters. These parameters must be used when redirecting the customer to the 3 rd party payment processor.

2.12.10.1 Redirect Handling (Redirect Payment Types)

The XML response for a redirect type payment request will contain a `RedirectParameters` complex element containing an arbitrary number of redirect parameters. See the example snippet below:

Example of `RedirectParams` complex element:

```

<Response>
  <MisterCashTxn>
    <RedirectParams>
      <redirect_param param_name="customerRef">X4M75U</redirect_param>
      <redirect_param param_name="country">UK</redirect_param>
      <redirect_param param_name="currency">GBP</redirect_param>
      <redirect_param param_name="recieveCurrency">GBP</redirect_param>
      <redirect_param param_name="email">john@mail.com</redirect_param>
      <redirect_param param_name="service">MISTERCASH</redirect_param>
      <redirect_param param_name="language">en</redirect_param>
    </RedirectParams>
    ...
    <content_type>application/x-www-form-urlencoded</content_type>
    <http_method>GET</http_method>
    <redirect_url>http://envoy</redirect_url>
  </MisterCashTxn>
  ...
</Response>

```

When redirecting the customer to the 3rd party payment processor URL (`redirect_url`), you must extract the redirect parameters from the `RedirectParams` complex element and include these in the redirect (paying attention to the `content_type` and `http_method`). Note that the redirect parameters above are only an example.

2.12.11 Compatibility With HPS

The alternative payment methods described in the previous section can also be used in conjunction with HPS (Hosted Payment Service). Some examples of requests / responses are shown below. Note that the *success_url* and *failure_url* elements described in the sections above are **optional**. In the case where no *success_url* or *failure_url* is supplied, HPS will use the *return_url* instead.

For more information on HPS, please see the relevant section of this guide.

Example of HPS Setup Session Request for Sofort Txn:

```

<Request>
  <Authentication>
    <password>#####</password>
    <client>14000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">119.76</amount>
      <merchantreference>830300000000265</merchantreference>
    </TxnDetails>
    <HpsTxn>
      <method>setup_full</method>
      <page_set_id>99999</page_set_id>
      <return_url>http://returnurl</return_url>
    </HpsTxn>
    <SofortTxn>
      <method>payment</method>
      <language>en</language>
      <customer_email>john@mail.com</customer_email>
      <customer_ip>127.0.0.1</customer_ip>
      <title>MR</title>
      <first_name>JOHN</first_name>
      <last_name>KENT</last_name>
      <country>FR</country>
      <billing_street1>Address Line 1</billing_street1>
      <billing_city>CA</billing_city>
      <billing_postcode>12345</billing_postcode>
      <billing_country>DE</billing_country>
      <profile>1</profile>
    </SofortTxn>
  </Transaction>
</Request>

```

Example of HPS Setup Session Response for Sofort Txn:

```
<Response>
  <HpsTxn>
    <hps_url>http://localhost:8180/hps/</hps_url>
    <session_id>ce0be36b-15c1-46e5-9f67-33ce39ad5dc7</session_id>
  </HpsTxn>
  <datacash_reference>4900903000000492</datacash_reference>
  <merchantreference>830300000000265</merchantreference>
  <mode>TEST</mode>
  <reason>ACCEPTED</reason>
  <status>1</status>
  <time>1320328631</time>
</Response>
```

Multiple Txn blocks can also be supplied in the context of an HPS Setup session. In this scenario, when the customer is redirected to HPS, they will be given a choice of payment methods. See the below request example below:

Example of HPS Setup Session Request with Multiple Txn Blocks:

```

<Request>
  <Authentication>
    <password>#####</password>
    <client>14000001</client>
  </Authentication>
  <Transaction>
    <TxnDetails>
      <amount currency="GBP">119.76</amount>
      <merchantreference>830300000000265</merchantreference>
    </TxnDetails>
    <HpsTxn>
      <method>setup_full</method>
      <page_set_id>99999</page_set_id>
      <return_url>http://returnurl</return_url>
    </HpsTxn>
    <SofortTxn>
      <method>payment</method>
      <language>en</language>
      <customer_email>john@mail.com</customer_email>
      <customer_ip>127.0.0.1</customer_ip>
      <title>MR</title>
      <first_name>JOHN</first_name>
      <last_name>KENT</last_name>
      <country>FR</country>
      <billing_street1>Address Line 1</billing_street1>
      <billing_city>CA</billing_city>
      <billing_postcode>12345</billing_postcode>
      <billing_country>DE</billing_country>
      <profile>1</profile>
    </SofortTxn>
    <MisterCashTxn>
      <method>payment</method>
      <language>en</language>
      <customer_email>john@mail.com</customer_email>
      <customer_ip>127.0.0.1</customer_ip>
      <title>MR</title>
      <first_name>JOHN</first_name>
      <last_name>KENT</last_name>
      <country>FR</country>
      <billing_street1>Address Line 1</billing_street1>
      <billing_city>CA</billing_city>
      <billing_postcode>12345</billing_postcode>
      <billing_country>DE</billing_country>
    </MisterCashTxn>
  </Transaction>
</Request>

```

3. APG – ALTERNATIVE PAYMENT GATEWAY

3.1 INTRODUCTION To APG

This section of the DataCash Developers Guide describes the Alternative Payment Gateway, which provides non card purchase and payout method processing (including Risk Screening) functionality.

The APG section of the document includes detailed descriptions of the process flow involved in the various Services provided by the APG.

3.1.1 *Glossary*

Abbreviation / Term	Description
APG	Alternative Payment Gateway
SOAP	Simple Object Access Protocol
WSDL	Web Service Definition Language
RSG	Risk Services Gateway
Processor	A generic reference to an electronic wallet provider / processing institution. The term “payment method” is also sometimes used interchangeably with the term “processor” in the APG. See Appendix 7 for a list of all the Processors that the APG provides access to.

[Back to Contents](#)

3.2 APG OVERVIEW

The APG exposes two interfaces that facilitate transaction flow: the SOAP Web Service and the Redirect Solution. A successful integration will involve using one or both of these.

The APG supports the UTF-8 Unicode encoding standard internally, but please note that not all purchase and payout methods support this standard.

3.2.1 APG SOAP Web Service

The APG Web Service is a standard SOAP web service to be called directly from the merchant’s server.

The APG web service is RPC-based (Remote Procedure Call), which allows clients to make the synchronous invocation of DataCash’s gateway seem like a local procedure call.

The web service integration offers several benefits:

- Service oriented standards based architecture;
- SOAP 1.1 and 1.2 based data encoding encapsulates transportation protocols;
- WSDL descriptor file allows easier integration and client side validation;
- The RPC-based interface allows developers to generate a web service client without having to manually encode and decode XML data.

This web service offers 7 methods:

- Purchase - see section 3.3.2
- Payout – see section 3.3.3
- TransactionQuery – see section 3.3.4.1
- TransactionQueryByTransactionId – see section 3.3.4.2
- RetrieveSettlementValue – see section 3.3.5
- GetInformation – see section 3.3.6
- Modification – see section 3.3.7.1

Web Service locations:

Live URL	https://DPG-CPT.datacash.com/Service/Production.wsdl
Staging URL	https://DPG-CPT.staging.datacash.com/Service/Staging.wsdl

3.2.2 APG Redirect Solution

Certain Processors' transactions make use of this interface. DataCash will provide the necessary information that needs to be sent to the Processor. When processing has been completed, the Processor will then redirect the end user to the redirect solution for the final step of the transaction.

Note that the Redirect Solution of the APG provides an interface between the APG and Processors, so Merchants do not need to integrate into the Redirect Solution (as they do into the Web Service).

Refer to the Transaction Flow descriptions (section 1) for more information.

3.2.3 Security

To keep the merchant information safe between the merchant and the DataCash Gateway, the following methods of security are in place:

- Communication Encryption.
- Digital certificates.
- Merchant Account.

In order to have a secure communication channel between the merchant site and the APG, use SSL version 3, with 1028-bit encryption.

For merchants integrating using Java, it is recommended that you install the client certification. Note: Java users can read more at
<http://www.oracle.com/technetwork/java/api-141528.html>

DataCash will also supply the merchant with their account details: an Account Name, Company ID and PIN, which must be sent to the APG on a transactional basis in order to process successfully.

3.2.4 *Transaction Flow*

The flow of a transaction from a merchant's perspective may differ depending on whether the payment method requires a redirect, partial redirect or no redirect.

3.2.4.1 Non-Redirect Flow (Figure 1 - Purchase)

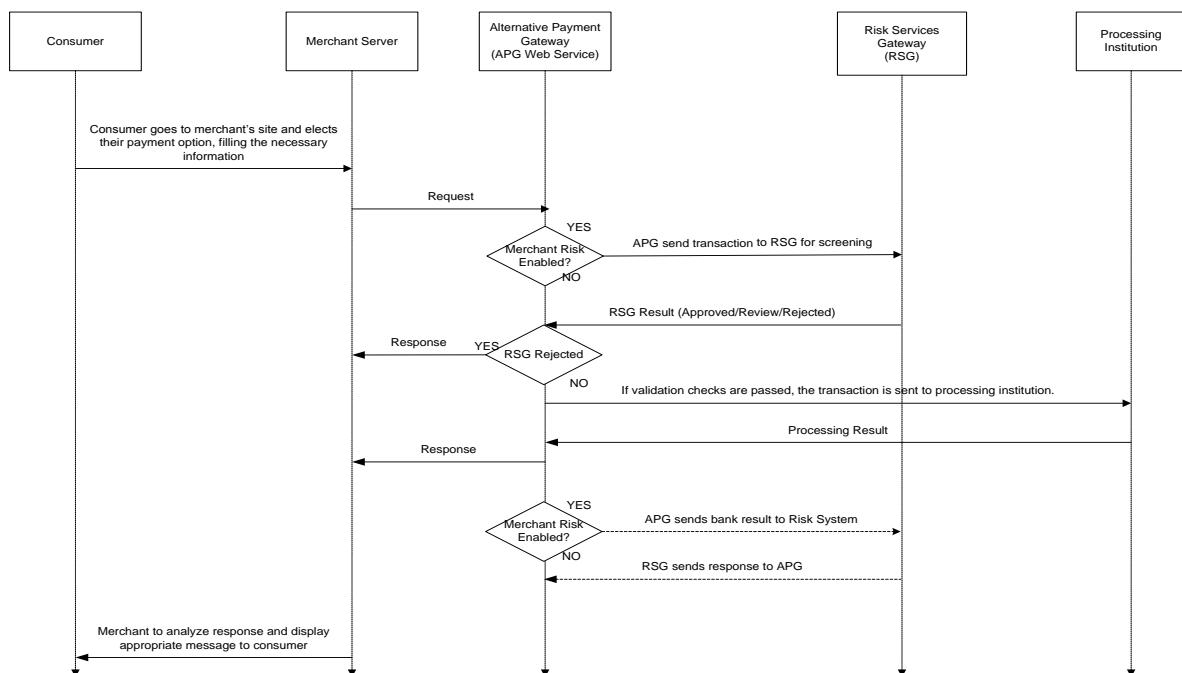
1. The client enters their purchase details on the merchant's site.
2. The merchant sends the relevant information to the APG using the Purchasemethod of the APG Web Service (see 3.3)
3. If the merchant is enabled for Risk, APG sends the transaction to RSG for screening – if the transaction is rejected by RSG, there is no further processing of the transaction so continue at step 6; if the transaction is not rejected by RSG, continue at the next step.
4. APG processes the transaction.
5. If the merchant is enabled for Risk, APG sends the bank result to RSG.
6. APG responds to the Purchase method request with the transaction's result details.
7. The result details must be analyzed and an appropriate message be displayed to the client on the merchant's site.

The following Processors use the non-redirect flow for Purchase transactions:

- AM – Amazon
- CL – Click2Pay
- EL – eCheck
- EV – ExperCash (ELV)
- NT – Neteller
- O3 – Offline Bank Transfer
- UK – Ukash

(Refer to Appendix 7 for descriptions of the various Processors.)

Figure 1



3.2.4.2 Non-Redirect Flow (Figure 2 - Payout)

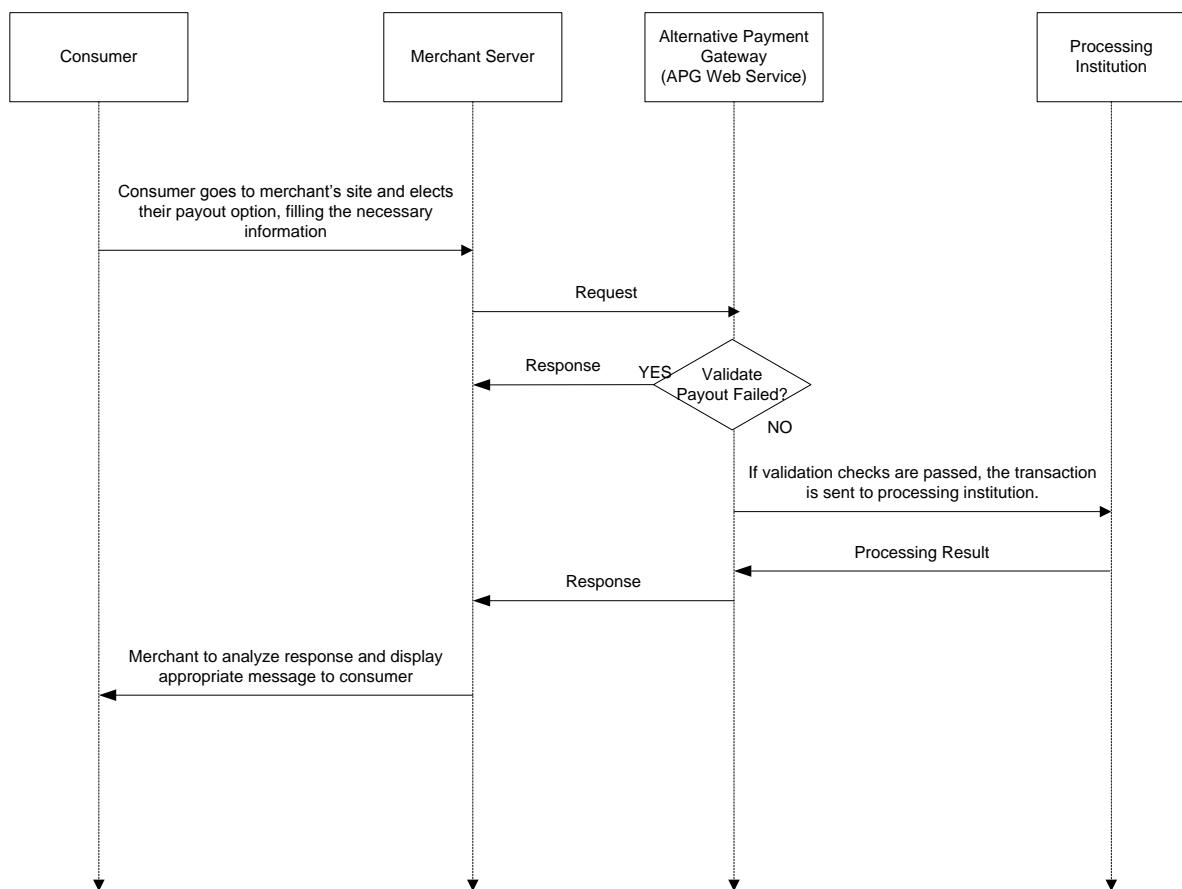
1. The client enters their payout details on the merchant's site.
2. The merchant sends the relevant information to the APG using the Payout method of the APG Web Service (see 3.3)
3. APG validates the payout request. If the transaction is rejected, there is no further processing of the transaction so continue at step 5; if the transaction is not rejected, continue at the next step.
4. APG processes the transaction.
5. APG responds to the Payout method request with the transaction's result details.
6. The result details must be analyzed and an appropriate message be displayed to the client on the merchant's site.

The following Processors use the non-redirect flow for Payout transactions:

- BU – ClickandBuy
- CL – Click2Pay
- EP – EntroPay
- IN – InstaDebit
- MB – MoneyBookers
- NT – Neteller
- OC – Envoy OneClick
- UK – Ukash

(Refer to Appendix 7 for descriptions of the various Processors.)

Figure 2



3.2.4.3 Redirect Flow (Figure 3)

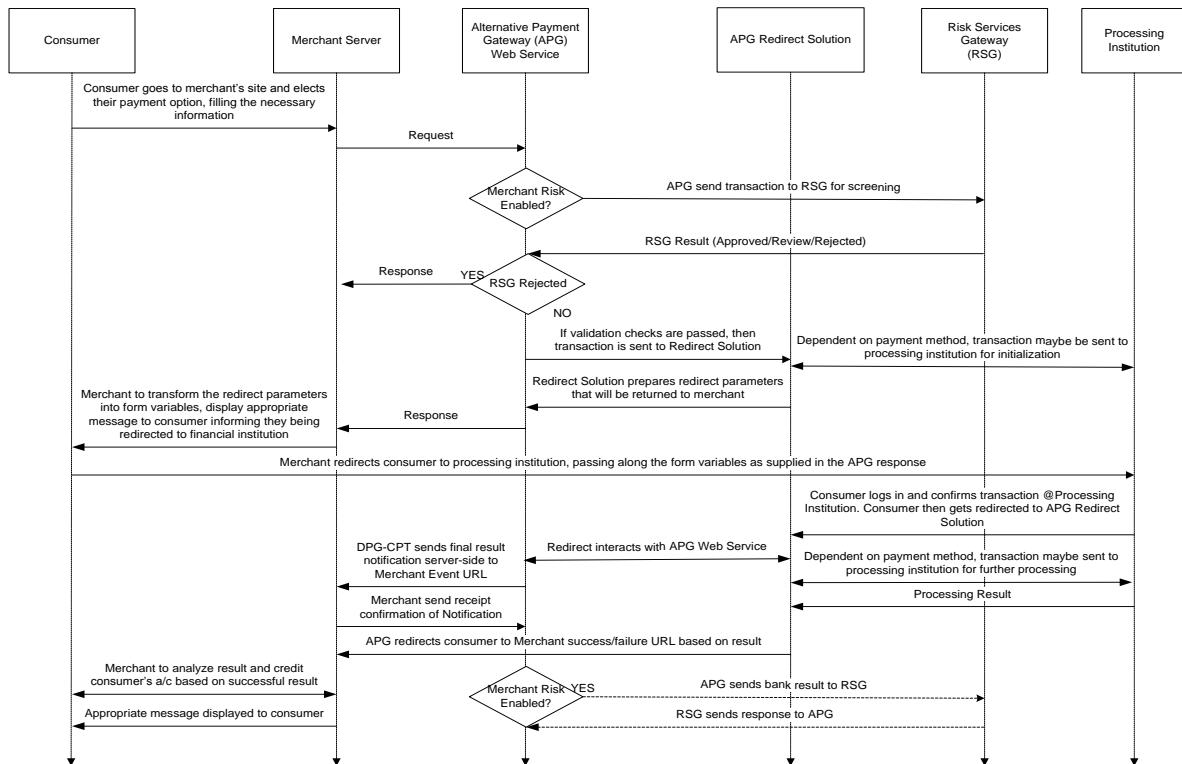
1. The client enters their purchase details on the merchant's site.
2. The merchant sends the relevant information to the APG using the Purchase method of the APG Web Service (see 3.3).
3. If the merchant is enabled for Risk, APG sends the transaction to RSG for screening – if the transaction is rejected by RSG, there is no further processing of the transaction and APG responds to merchant; if the transaction is not rejected by RSG, continue at the next step.
4. APG responds to the Purchase method request by returning the relevant set of redirect parameters.
5. Merchant needs to transform the redirect parameters into form variables and then redirect client to the Processor.
6. Client enters necessary information at the Processor's site and confirms transaction.
7. Processor will redirect the client to the Redirect Solution of the APG.
8. If the merchant is enabled for Risk, APG sends the bank result to RSG.
9. APG sends transaction result details notification to merchant's URL via a server-side post as detailed in Section 0.
10. The merchant acknowledges receipt of transaction result notification.
11. The result details must be analyzed by the merchant.
12. APG redirects the client to the merchant's success or failure page depending on the result.

Note: Steps 9-10 happens simultaneously with step 12.

The following Processors use the redirect flow:

- BU – ClickAndBuy
- EN – eNETS
- EV – ExperCash (GiroPay and Sofort)
- I3 – iDeal
- IN – InstaDebit
- MB – Moneybookers
- NS – NPS
- OC – OneClick
- PF – PaysafeCard
- W3 – WebMoney

(Refer to Appendix 7 for descriptions of the various Processors.)

Figure 3

3.2.4.4 PARTIAL-Redirect Flow (Figure 4)

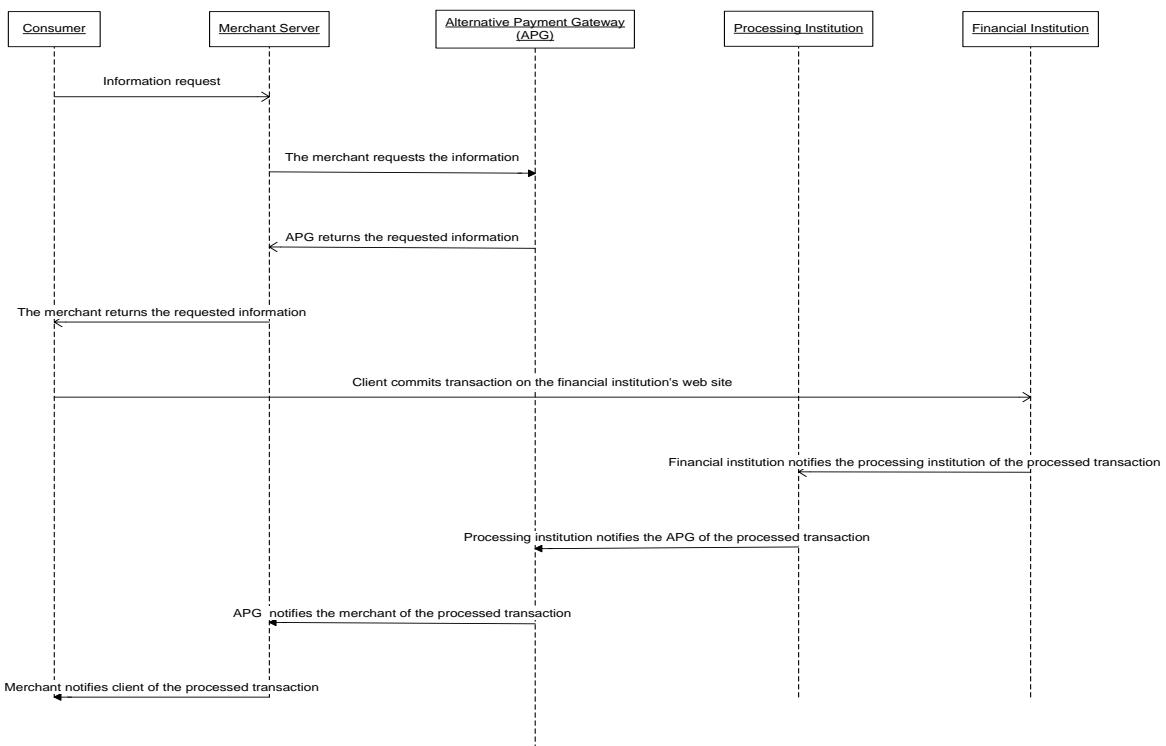
1. The client requests certain information (to be used for submitting a transaction) from the merchant.
2. The merchant requests the information from APG. APG gathers the requested information, and returns it to the merchant, who in turn provides it to the client.
3. The client enters the necessary information at the financial institution's site and confirms the transaction.
4. Financial institution will notify the Processor of the processed transaction.
5. The Processor notifies APG of the processed transaction.
6. APG receives this notification of the successful processed transaction.
7. APG sends transaction result details notification to merchant's URL via a server-side post.

The following payment methods / processors use the partial-redirect flow:

- BT – Direct Bank Transfer

(Refer to Appendix 7 for descriptions of the various payment methods / processors.)

Figure 4



[Back to Contents](#)

3.3 APG SOAP WEB SERVICE

In order to call one of the web methods available in the web service, the following needs to happen:

- Specify the DataCash URL specified in section 3.2.1 as the web reference
- Create the proxy class
- Instantiate the proxy class
- Pass along the parameters to the specific web method

All the Web Service's Web Methods are discussed in detail in the following sections.

3.3.1 Shared XML Request and Response elements

There are certain XML complex structures in the Requests and Responses that are common to more than one of the APG Web Methods – these elements are covered in this section. Others are only used for a particular Web Methods – these elements will be covered within the documentation for those Web Methods.

3.3.1.1 Shared Request Structures

3.3.1.1.1 AccountDetails

This is a set of the merchant's authentication details which is required in order to access the APG.

Element Name	AccountDetails
Type	AccountItem
Position	Purchase.purchaseDetails Payout.payoutDetails TransactionQuery RetrieveSettlementValue InformationRequest Modification
Children	Yes
Attributes	No

Elements of AccountDetails			
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details
AccountName	The DataCash Account Name is the unique gateway-supplied identifier for the Merchant (applicable for all Merchants).	String (20)	Mandatory for all Payment Methods and Risk
AccountPIN	The DataCash Account PIN is the unique Gateway supplied password for the Merchant.	String (50)	Mandatory for all Payment Methods
CompanyId	The DataCash Company Id is the unique Gateway supplied identifier for the Merchant.	String (20)	Mandatory for all Payment Methods

Example XML for AccountDetails complex elements

```
<AccountDetails>
  <AccountName>TestAccount</AccountName>
  <AccountPIN>TestPIN</AccountPIN>
  <CompanyId>A99</CompanyId>
</AccountDetails>
```

3.3.1.1.2 AmountDetails

Element Name	AmountDetails
Type	AmountItem
Position	Purchase.purchaseDetails.BillingDetails Purchase.purchaseDetails.ShippingDetails Payout.payoutDetails.BillingDetails RetrieveSettlementValue Modification
Children	Yes
Attributes	No

Elements of AmountDetails			
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details
Amount	This is the transaction amount.	Int This value must be sent as cents (e.g. US \$100.01 = 10001).	Mandatory for all Payment Methods and Risk
Exponent	This is the transaction currency exponent. Please refer to Appendix 10.	Short	Mandatory for all Payment Methods and Risk
CurrencyCode	This is the transaction currency.	Short This value must be sent as the numeric ISO currency code (Please reference Appendix 10).	Mandatory for all Payment Methods and Risk

Example XML for AmountDetails complex elements

```
<AmountDetails>
  <Amount>725</Amount>
  <Exponent>2</Exponent>
  <CurrencyCode>826</CurrencyCode>
</AmountDetails>
```

3.3.1.1.3 SaleAmountDetails

Element Name	SaleAmountDetails
Type	AmountItem
Position	Purchase.purchaseDetails.BillingDetails
Children	Yes
Attributes	No

Elements of SaleAmountDetails			
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details
Amount	This is the transaction item sale amount.	Int This value must be sent as cents (e.g. US \$100.01 = 10001).	# Optional for OC. N/A for all other Payment Methods
Exponent	This is the transaction item sale currency exponent. Please refer to Appendix 10.	Short	# Optional for OC. N/A for all other Payment Methods
CurrencyCode	This is the transaction item sale currency.	Short This value must be sent as the numeric ISO currency code (Please reference Appendix 10).	# Optional for OC. N/A for all other Payment Methods

SaleAmount can be used if the merchant requires the APG to do a currency conversion from the Sale Amount to the merchant currency.

Example XML for AmountDetails complex elements

```
<SaleAmountDetails>
  <Amount>100</Amount>
  <Exponent>2</Exponent>
  <CurrencyCode>978</CurrencyCode>
</SaleAmountDetails>
```

3.3.1.2 Shared Response structures

3.3.1.2.1 Error

The result of an error encountered during processing will be displayed here.

Element Name: Error	
Position	Response.Event.Purchase Response.Event.Payout Response.Event.Settlement Response.Event.Information Response.Event.Modification
Children	Yes
Attributes	No

Elements of Error		
Element Name	Description	Data type / Limitations
Code	This is a return code which indicates the rejection reason, if applicable.	String (10)
Description	This is a text field used to give a short description of Code.	String (255)

Example XML for Error complex elements

```

<Error>
  <Code>020</Code>
  <Description>PurchaseDetails.AccountDetails.AccountPIN value
    invalid</Description>
</Error>

```

3.3.1.2.2 Event

Top-level container for the type of event that this Response refers to. This element applies to all transaction types.

Element Name:	Event
Position	Response
Children	Yes
Attributes	No

Elements of Event		
Element Name	Description	Data type / Limitations
Directly below the Event element will be one of the following complex XML structures:		
<ul style="list-style-type: none"> • Payout – see section 3.3.3.2 • Purchase – see section 3.3.2.3.4 • Settlement – see section 3.3.5.3.1 • Information – see section 3.3.6.3.6 • Modification – see section 3.3.7.1 		

3.3.1.2.3 Response

The complete XML structure that will be returned to the Merchant during a transaction.

Element Name:	Response
Position	Top-level
Children	Yes
Attributes	Yes

Attributes involved in the Response element			
Attribute Name	Located in which Element	Description	Data type / Limitations
version	Response	Version of the APG response complex structure.	String
AccountName	Response	This is the account name that the merchant holds with the processor.	String (50)

Elements of Response		
Element Name	Description	Data type / Limitations
Event	Top-level container for the type of event that this Response refers to. This element appears in this position for all transaction types.	n/a

Example XML for Response complex elements

```
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    ...other complex structures here, e.g. <Purchase...> etc.
```

3.3.2 Purchase

A call to the Purchase web method of the APG Web Service is the first step in the lifecycle of a transaction.

Validation will occur during the call to the web method - see Appendix 6 for a list of possible validation errors.

The result of this request will determine if the transaction has completed processing, or whether the merchant should continue to process the transaction (by redirecting to the Processor, and passing along the redirect parameters returned by the response to the Purchase request).

A transaction that adheres to the partial-redirect flow is an exception to the standard Purchase web method arrangement, since the merchant will never call the Purchase web method in this scenario.

The Purchase web method accepts two parameters: `purchaseDetails` of type `PurchaseItem`, and `methodDetails` of type `PaymentMethodItem`. The two types are custom complex structures. For more information on the composition of these structures, please refer to the WSDL which can be downloaded from the URLs specified in section 0.

3.3.2.1 Schema Elements for Purchase Request

In this section, the fields that can be presented with Purchase transaction Requests will be presented along with example XML for those fields.

To submit Processor transaction information for a Purchase transaction, one should use the relevant `PaymentMethodItem` within the `Purchase` schema element:

- `Purchase`
 - `PurchaseDetails` – see section 3.3.2.1.1
 - `AccountDetails` – see section 3.3.1.1.1
 - `BillingDetails` – see section 3.3.2.1.3
 - `AddressDetails` – see section 3.3.2.1.2
 - `AmountDetails` – see section 3.3.1.1.2
 - `SaleAmountDetails` – see section 3.3.1.1.3
 - `PersonalDetails` – see section 3.3.2.1.8
 - `ShippingDetails` – see section 3.3.2.1.4
 - `AddressDetails` – see section 3.3.2.1.2
 - `AmountDetails` – see section 3.3.1.1.2
 - `RiskDetails` – see section 3.3.2.1.5
 - `LineItems` – see section 3.3.2.1.6
 - `CustomMerchantDetails` – see section 3.3.2.1.7
 - `MethodDetails` – see section 3.3.2.1.26
 - One of the following:

<ul style="list-style-type: none"> • <code>AmazonItem</code> • <code>Click2PayItem</code> • <code>ClickAndBuyItem</code> • <code>ElectronicCheckItem</code> • <code>eNetsItem</code> • <code>ExperCashItem</code> 	<ul style="list-style-type: none"> – section 3.3.2.1.21 – section 3.3.2.1.11 – section 3.3.2.1.14 – section 3.3.2.1.20 – section 3.3.2.1.16 – section 3.3.2.1.19
---	--

- | | |
|---|---|
| <ul style="list-style-type: none"> • <i>iDealItem</i> • <i>InstaDebitItem</i> • <i>MoneybookersItem</i> • <i>NetellerItem</i> • <i>NPSItem</i> • <i>OBTItem</i> • <i>OneClickItem</i> • <i>PaySafeCardItem</i> • <i>UKashItem</i> • <i>WebMoneyItem</i> | <ul style="list-style-type: none"> - section 3.3.2.1.22 - section 3.3.2.1.18 - section 3.3.2.1.12 - section 3.3.2.1.10 - section 3.3.2.1.9 - section 3.3.2.1.25 - section 3.3.2.1.17 - section 3.3.2.1.13 - section 3.3.2.1.15 - section 3.3.2.1.24 |
|---|---|

The "Required / Payment Method Specific" column lists the following details:

- *Mandatory* - This means that field cannot be empty.
- *Recommended* - The data is highly recommended. The fields are not mandated by APG since not all processors mandate them. Please note that if these fields are not supplied and the transaction is processed through a processor which does mandate them, the transaction will be rejected.
- *Optional* – The data is not required for processing but should be supplied if available.
- *Not Applicable* – The data is not applicable for this payment method and will be discarded if supplied.

If no data is available for Recommended or Optional fields, the field can simply be left out completely. Under no circumstances is the word 'NULL' to be used as a default value for blank fields; merchants should rather leave the field blank.

NOTE: If the value sent for a field should exceed the maximum length then the transaction may be rejected by the risk system or risk screening may be bypassed.

SOAP XML Purchase Request document:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">
            <purchaseDetails>
                <TransactionId>string</TransactionId>
                <Description>string</Description>
                <TransactionDateTime>string</TransactionDateTime>
                <WebBrowserSessionId>string</WebBrowserSessionId>
                <SuccessURL>string</SuccessURL>
                <FailureURL>string</FailureURL>
                <UserIP>string</UserIP>
                <UserAgentHeader>string</UserAgentHeader>
                <UserAcceptHeader>string</UserAcceptHeader>
                <CustomerReference>string</CustomerReference>
                <AccountDetails>
                    <AccountName>string</AccountName>
                    <AccountPIN>string</AccountPIN>
                    <CompanyId>string</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>string</FirstName>
                    <LastName>string</LastName>
                    <Telephone1>string</Telephone1>
                    <Telephone1_CountryCode>short</Telephone1_CountryCode>
                    <Telephone1_AreaCode>short</Telephone1_AreaCode>
                    <Telephone2>string</Telephone2>
                    <Telephone2_CountryCode>short</Telephone2_CountryCode>
                    <Telephone2_AreaCode>short</Telephone2_AreaCode>
                    <DateOfBirth>dateTime</DateOfBirth>
                    <Email>string</Email>
                    <IDNumber>string</IDNumber>
                    <TypeOfID>NONE or PASSPORT or PERSONAL_ID or IDENTITY_CARD or
                        DRIVERS_LICENSE or TRAVEL_DOCUMENT or RESIDENCE_PERMIT or
                        IDENTITY_CERTIFICATE_NUMBER or
                        REGISTRO_FEDERAL_DE_CONTRIBUYENTES or
                        CREDENTIAL_DE_ELECTOR</TypeOfID>
                    <SSN>int</SSN>
                    <CPF_CNPJ>string</CPF_CNPJ>
                    <Nationality>string</Nationality>
                </PersonalDetails>
                <BillingDetails>
                    <Name>string</Name>
                    <AddressDetails>
                        <StreetLine1>string</StreetLine1>
                        <StreetLine2>string</StreetLine2>
                        <City>string</City>
                        <PostalCode>string</PostalCode>
                        <StateProvince>string</StateProvince>
                        <Country>string</Country>
                    </AddressDetails>
                    <AmountDetails>
                        <Amount>int</Amount>
                        <Exponent>short</Exponent>
                    </AmountDetails>
                </BillingDetails>
            </purchaseDetails>
        </Purchase>
    </soap:Body>
</soap:Envelope>

```

```
    <CurrencyCode>short</CurrencyCode>
    </AmountDetails>
    <SaleAmountDetails>
        <Amount>int</Amount>
        <Exponent>short</Exponent>
        <CurrencyCode>short</CurrencyCode>
    </SaleAmountDetails>
</BillingDetails>
<ShippingDetails>
    <Name>string</Name>
    <AddressDetails>
        <StreetLine1>string</StreetLine1>
        <StreetLine2>string</StreetLine2>
        <City>string</City>
        <PostalCode>string</PostalCode>
        <StateProvince>string</StateProvince>
        <Country>string</Country>
    </AddressDetails>
    <AmountDetails>
        <Amount>int</Amount>
        <Exponent>short</Exponent>
        <CurrencyCode>short</CurrencyCode>
    </AmountDetails>
</ShippingDetails>
<RiskDetails>
    <MerchantId>short</MerchantId>
    <AccountName>string</AccountName>
    <AccountPassword>string</AccountPassword>
    <AccountNumber>string</AccountNumber>
    <UserId>string</UserId>
    <UserMachineId>string</UserMachineId>
    <UserProfile1>string</UserProfile1>
    <UserProfile2>string</UserProfile2>
    <UserProfile3>string</UserProfile3>
</RiskDetails>
    <Language>string</Language>
</purchaseDetails>
<methodDetails/>
</Purchase>
</soap:Body>
</soap:Envelope>
```

3.3.2.1.1 PurchaseDetails

This is a set of fields and structures required for a transaction.

Element Name	PurchaseDetails
Type	PurchaseItem
Position	Purchase
Children	Yes
Attributes	No

Elements of PurchaseDetails				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
TransactionId	The TransactionId is a unique identifier generated by the integrating (Merchant) system. Usually this is the integrating system's internal payment order reference number.	String (16); We Recommend this value be be prefixed with the CompanyId plus an underscore e.g. ABC_1234567 (where the CompanyId is a value that will be supplied by DataCash).	All	Mandatory
Description	Use this field to supply descriptive text for the transaction.	String (255)	UK	Optional
TransactionDateTime	The Merchant's time stamp of the transaction.	String (19); Format "yyyy-mm-dd hh:mm:ss"	EN EV Risk UK	Mandatory Mandatory Mandatory Mandatory
UserAgentHeader	This is a value indicating the browser type and operating system details of the user. This information can be retrieved from the HTTP_USER_AGENT header.	String (255)	All	Recommended

UserAcceptHeader	This is a value that contains the MIME types that a browser can accept. This information can be retrieved from the HTTP_ACCEPT header.	String (255)	All	Not Applicable
SuccessURL	This is the URL to which the merchant's consumer will be redirected to after a successful transaction.	String (255)	BU	Mandatory
			EN	Mandatory
			EV	Mandatory: for Giropay and Sofort only
			I3	Mandatory
			IN	Mandatory
			MB	Mandatory
			NS	Mandatory
			OC	Mandatory
			PF	Mandatory
FailureURL	This is the URL to which the merchant's consumer will be redirected to after an unsuccessful transaction.	String (255)	W3	Mandatory
			BU	Mandatory
			EN	Mandatory
			EV	Mandatory: for Giropay and Sofort only
			I3	Mandatory
			IN	Mandatory
			MB	Mandatory
			NS	Mandatory
			OC	Mandatory
Language	<p>This field contains the language in which the client's user experience will take place. Please refer to Appendix 13 for supported languages.</p> <p>If no language is supplied, the value defaults to "en" (English).</p>	String (20)	All	Recommended
			Risk	Not applicable
UserIP	Consumer's IP address.	String (26); (Format: 255.255.255.255	CL EL EV	Mandatory Mandatory Mandatory

)	Risk	Recommended
CustomerReference	This is the generic identifier of a customer.	String(70)	EV	Optional
			IN	Mandatory
			O3	Mandatory
			OC	Mandatory for InstantBank Purchase
			All	Mandatory
AccountDetails	This is a set of the merchant's authentication details which is required in order to access the APG.	See section 3.3.1.1.1	All	Mandatory
PersonalDetails	This is a set of consumer specific details that are required by the Processors.	See section 3.3.2.1.8	All	Mandatory
			EV	Recommended see PersonalDetailName PersonalDetailtName
BillingDetails	This is a set of the consumer's billing details that are required by the Processors.	See section 3.3.2.1.3	All	Mandatory
ShippingDetails	This is a set of consumer's shipping details that are required by the Processors.	See section 3.3.2.1.4	AM	Mandatory
			NS	Mandatory
RiskDetails	This is a set of specific details that are required by the Processors.	See section 3.3.2.1.5	All	If Risk is enabled on the merchant this element is mandatory
MerchantCustomDetails	Optional structure allowing Merchants to pass custom fields and values.	See section 3.3.2.1.7	All	Optional
LineItems	Merchants can provide detail on the purchased items. The items are passed as Line Items.	See Section 3.3.2.1.6	AM	Mandatory

Example XML for PurchaseDetails with all complex elements

```

<purchaseDetails>
    <TransactionId>string</TransactionId>
    <Description>string</Description>
    <TransactionDateTime>string</TransactionDateTime>
    <WebBrowserSessionId>string</WebBrowserSessionId>
    <SuccessURL>string</SuccessURL>
    <FailureURL>string</FailureURL>
    <UserIP>string</UserIP>
    <UserAgentHeader>string</UserAgentHeader>
    <UserAcceptHeader>string</UserAcceptHeader>
    <Language>string</Language>
    <AccountDetails>
        <AccountName>TestAccount</AccountName>
        <AccountPIN>TestPIN</AccountPIN>
        <CompanyId>A99</CompanyId>
    </AccountDetails>
    <PersonalDetails>
        <FirstName>Jerry</FirstName>
        <LastName>Anderson</LastName>
        <Telephone1>6149135</Telephone1>
        <Telephone1_CountryCode>44</Telephone1_CountryCode>
        <Telephone1_AreaCode>105</Telephone1_AreaCode>
        <Telephone2/>
        <Telephone2_CountryCode/>
        <Telephone2_AreaCode/>
        <DateOfBirth>1982-09-14</DateOfBirth>
        <Email>customer@email.com</Email>
        <IDNumber>123456789</IDNumber>
        <TypeOfID>PASSPORT</TypeOfID>
        <SSN/>
        <CPF_CNPJ/>
        <Nationality>GB</Nationality>
        <Gender>M</Gender>
    </PersonalDetails>
    <BillingDetails>
        <Name>Home</Name>
        <AddressDetails>
            <StreetLine1>404 Mulberry Court</StreetLine1>
            <StreetLine2>78 Praed Street</StreetLine2>
            <City>London</City>
            <PostalCode>W2 1NT</PostalCode>
            <StateProvince>EN</StateProvince>
            <Country>GB</Country>
        </AddressDetails>
        <AmountDetails>
            <Amount>725</Amount>
            <Exponent>2</Exponent>
            <CurrencyCode>826</CurrencyCode>
        </AmountDetails>
        <SaleAmountDetails>
            <Amount>650</Amount>
        </SaleAmountDetails>
    </BillingDetails>
</purchaseDetails>

```

```
<Exponent>2</Exponent>
<CurrencyCode>826</CurrencyCode>
</SaleAmountDetails>
</BillingDetails>
<ShippingDetails>
<Name>Home</Name>
<AddressDetails>
<StreetLine1>404 Mulberry Court</StreetLine1>
<StreetLine2>78 Praed Street</StreetLine2>
<City>London</City>
<PostalCode>W2 1NT</PostalCode>
<StateProvince>EN</StateProvince>
<Country>GB</Country>
</AddressDetails>
<AmountDetails>
<Amount>725</Amount>
<Exponent>2</Exponent>
<CurrencyCode>826</CurrencyCode>
</AmountDetails>
</ShippingDetails>
<RiskDetails>
<MerchantId>123</MerchantId>
<AccountName>TestAccount</AccountName>
<AccountPassword>TestPwd</AccountPassword>
<AccountNumber></AccountNumber>
<UserId>CUST2468</UserId>
<UserMachineId>DESKTOP13579</UserMachineId>
<UserProfile1>is_vip_customer</UserProfile1>
<UserProfile2>is_promotion</UserProfile2>
<UserProfile3>is_good_customer</UserProfile3>
<LineItems/>
</RiskDetails>
<MerchantCustomDetails>
<CustomDetail>
<FieldName>PaymentId</FieldName>
<FieldValue>1234</FieldValue>
</CustomDetail>
</MerchantCustomDetails>
<LineItems/>
</purchaseDetails>
```

3.3.2.1.2 AddressDetails

Element Name	AddressDetails
Type	AddressItem
Position	Purchase.purchaseDetails.BillingDetails Purchase.purchaseDetails.ShippingDetails Payout.payoutDetails.BillingDetails
Children	Yes
Attributes	No

Elements of AddressDetails				
Element Name	Description	Data type / Limitations	Payment Applicable Requirements	
StreetLine1	Consumer's address line1	String (60)	BU	Mandatory
			EP	Optional
			EV	Optional (refer to note below)
			IN	Optional
			MB	Optional
			NS	Mandatory
			O3	Optional
			OC	Optional – However is mandatory for some of the OC products.
			Risk	Mandatory
StreetLine2	Consumer's address line2	String (60)	BU	Optional
			EV	Optional (refer to note below)
			NS	Recommended
			O3	Optional
			OC	Optional – However is mandatory for some of the OC products.
			Risk	Recommended
City	Consumer's city	String (60) Only alpha characters (A – Z) and spaces are allowed in this field.	BU	Optional
			EP	Optional
			EV	Optional (refer to note below)
			IN	Optional
			MB	Optional
			NS	Mandatory
			O3	Optional
			OC	Optional – However is mandatory for some of the OC products.
			Risk	Mandatory

PostalCode	Consumer's Postal code/zip.	String (10)	BU	Optional
			EP	Optional
			EV	Optional (refer to note below)
			IN	Optional
StateProvince	Consumer's State/Province	String (25) Please reference Appendix 8. For countries listed in Appendix 8, please send the value listed in the Code Column. For any other country, please send the full state name up to the max length specified.	MB	Optional
			NS	Mandatory
			O3	Optional
			OC	Optional – However is mandatory for some of the OC products.
Country	Consumer's Country	String (60) This is a 2-character code that needs to be supplied. Please reference Appendix 9.	Risk	Mandatory
			CL	Mandatory
			EP	Optional
			IN	Mandatory
Country	Consumer's Country	String (60) This is a 2-character code that needs to be supplied. Please reference Appendix 9.	MB	Optional
			NS	Mandatory
			NT	Optional
			O3	Mandatory
Country	Consumer's Country	String (60) This is a 2-character code that needs to be supplied. Please reference Appendix 9.	OC	Optional – However is mandatory for some of the OC products.
			Risk	Mandatory

ExperCash Note: When submitting an ExperCash (ELV) transaction, either of the below combinations of customer data may be sent to APG or nothing at all.

Field	Combo 1	Combo 2	Combo 3	Combo 4	Combo 5	Combo 6
PersonalDetails.FirstName	x	x			x	
PersonalDetails.LastName	x	x	x	x	x	
StreetLine1	x	x	x			
StreetLine2	x			x	x	
City	x	x	x	x	x	
PostalCode	x	x	x	x	x	
Country	x	x	x	x	x	

If only partial data is supplied, then the transaction will be rejected.

The above fields are only applicable for ELV (Buy and Authorise). For GiroPay and Sofort, these can be left out completely or sent through with blank values. In the case where Risk is enabled, these values must be sent through.

Example XML for AddressDetails complex elements

```
<AddressDetails>
    <StreetLine1>404 Mulberry Court</StreetLine1>
    <StreetLine2>78 Praed Street</StreetLine2>
    <City>London</City>
    <PostalCode>W2 1NT</PostalCode>
    <StateProvince>EN</StateProvince>
    <Country>GB</Country>
</AddressDetails>
```

3.3.2.1.3 BillingDetails

This is a set of the consumer's billing details that are required by the Processors.

Element Name	BillingDetails
Type	BillingItem
Position	Purchase.purchaseDetails Payout.payoutDetails
Children	Yes
Attributes	No

Elements of BillingDetails					
Element Name	Description	Purchase / Payout	Data type / Limitations	Payment Applicable Requirements	
Name	This is descriptor for the billing details.	Purchase, Payout	String (50)	EV	Mandatory: Only for ELV (Buy and Authorise)
				EL	Mandatory
				O3	Optional
				OC	Optional

			Risk	Recommended
AddressDetails	These are line items of the address.	Purchase, Payout		See section 3.3.2.1.2
AmountDetails	This is a set of billing amount details.	Purchase, Payout		See section 3.3.1.1.2
SaleAmountDetails	This is a set of sale amount details.	Purchase		See section 3.3.1.1.3

Example XML for BillingDetails complex elements

```

<BillingDetails>
  <Name>Home</Name>
  <AddressDetails>
    <StreetLine1>404 Mulberry Court</StreetLine1>
    <StreetLine2>78 Praed Street</StreetLine2>
    <City>London</City>
    <PostalCode>W2 1NT</PostalCode>
    <StateProvince>EN</StateProvince>
    <Country>GB</Country>
  </AddressDetails>
  <AmountDetails>
    <Amount>725</Amount>
    <Exponent>2</Exponent>
    <CurrencyCode>826</CurrencyCode>
  </AmountDetails>
  <SaleAmountDetails>
    <Amount>650</Amount>
    <Exponent>2</Exponent>
    <CurrencyCode>826</CurrencyCode>
  </SaleAmountDetails>
</BillingDetails>

```

3.3.2.1.4 ShippingDetails

This is a set of the consumer's shipping details that are required by the Processors.

Element Name	ShippingDetails
Type	ShippingItem
Position	Purchase.purchaseDetails
Children	Yes
Attributes	No

Elements of ShippingDetails				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
Name	This is descriptor for the shipping details.	String (50)	NS	Optional
			O3	Optional
			Risk	Optional
AddressDetails	These are line items of the address.	See section 3.3.2.1.2		
AmountDetails	This is a set of billing amount details.	See section 3.3.1.1.2		

Example XML for ShippingDetails complex elements

```

<ShippingDetails>
    <Name>Home</Name>
    <AddressDetails>
        <StreetLine1>404 Mulberry Court</StreetLine1>
        <StreetLine2>78 Praed Street</StreetLine2>
        <City>London</City>
        <PostalCode>W2 1NT</PostalCode>
        <StateProvince>EN</StateProvince>
        <Country>GB</Country>
    </AddressDetails>
    <AmountDetails>
        <Amount>725</Amount>
        <Exponent>2</Exponent>
        <CurrencyCode>826</CurrencyCode>
    </AmountDetails>
</ShippingDetails>

```

3.3.2.1.5 RiskDetails

This is a set of specific details that are required by for Risk Screening.

Element Name	RiskDetails
Type	RiskItem
Position	Purchase.purchaseDetails
Children	Yes
Attributes	No

Elements of RiskDetails			
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details

MerchantId	This is a unique gateway supplied risk identifier for merchants who require risk screening.	Short	Risk	#Optional
AccountName	This is the associated account name that goes with the MerchantId supplied by DataCash.	String (30)	Risk	#Optional
AccountPassword	This is the associated account password that goes with the MerchantId supplied by DataCash.	String (50)	Risk	#Optional
AccountNumber	A unique account number at the Merchant associated with individual that is transacting.	String (30)	Risk	Mandatory
UserId	This is a Merchant configurable data field typically used to identify a customer uniquely within the system, regardless of the amount of accounts that the customer has.	String (36)	Risk	Optional
UserMachineId	This is an ID used to identify the computer (actual hardware) uniquely within the system regardless of the amount of customers or accounts that's using the computer. Examples of the User Machine ID are Motherboard ID, Hard drive ID, CPU ID, etc.	String (25)	Risk	Optional
UserProfile1	This is a customizable field that may be used by the merchant to supplement fraud screening processes according to a risk classification set by the merchant (e.g. VIP flag, indication whether the transaction is part of a promotion, a confidence level around the identity of the customer, etc).	String (20)	Risk	Optional

UserProfile2	This is a customizable field that may be used by the merchant to supplement fraud screening processes according to a risk classification set by the merchant.	String (20)	Risk	Optional
UserProfile3	This is a customizable field that may be used by the merchant to supplement fraud screening processes according to a risk classification set by the merchant.	String (20)	Risk	Optional
LineItems	Retail merchants can provide detail on the purchased items. The items are passed as Line Items.	See Section 0	Risk	Mandatory for Retail Risk Screening

The Risk Authentication Fields (MerchantId, AccountName and AccountPassword) are by default setup on the APG Merchant Profile. However the default can be overridden. To override the default, values for all the authentication fields (MerchantId, AccountName and AccountPassword) are required on the RiskDetails element of the payment request.

Example XML for RiskDetails complex elements

```

<RiskDetails>
  <MerchantId>123</MerchantId>
  <AccountName>TestAccount</AccountName>
  <AccountPassword>TestPwd</AccountPassword>
  <AccountNumber></AccountNumber>
  <UserId>CUST2468</UserId>
  <UserMachineId>DESKTOP13579</UserMachineId>
  <UserProfile1>is_vip_customer</UserProfile1>
  <UserProfile2>is_promotion</UserProfile2>
  <UserProfile3>is_good_customer</UserProfile3>
  <LineItems/>
</RiskDetails>

```

3.3.2.1.6 Line Item

This is a set of specific details that are required by either the Risk Line Item or the Purchase Detail Line Item.

Element Name	LineItem
Type	LineItem
Position	Purchase.purchaseDetails.RiskDetails.LineItems Purchase.purchaseDetails.LineItems
Children	Yes

Attributes	No
------------	----

3.3.2.1.6.1 Position: Purchase.purchaseDetails.RiskDetails.LineItems

Elements of a LineItem Structure				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
LineItemId	Unique line item ID	Int	Risk	#Mandatory
ProductCode	Line item product code	String(50)	Risk	#Mandatory
ProductDescription	Line item description	String(50)	Risk	#Mandatory
ProductCategory	Type of item as classified by the merchant	String(50)	Risk	#Mandatory
ProductRisk	Indication of how susceptible to product is to fraud. Possible values: High Medium Low	String(50)	Risk	#Mandatory
OrderQuantity	Number of items ordered	Int	Risk	#Mandatory
UnitPrice	The amount of the transaction in the smallest unit of the currency	Int	Risk	#Mandatory

– Line Items are only Mandatory for **Retail** Risk Screening.

Example XML for LineItem complex elements

```

<LineItem>
  <LineItemId>1</LineItemId>
  <ProductCode>Sample</ProductCode>
  <ProductDescription>Sample Description</ProductDescription>
  <ProductCategory>Food</ProductCategory>
  <ProductRisk>Low</ProductRisk>
  <OrderQuantity>1</OrderQuantity>
  <UnitPrice>100</UnitPrice>
</LineItem>

```

3.3.2.1.6.2 Position: Purchase.purchaseDetails.LineItems

Elements of a LineItem Structure

Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
LineItemId	Unique line item ID	Int	AM	Mandatory
ProductCode	Line item product code	String(50)	AM	Mandatory
ProductDescription	Line item description	String(50)	AM	Mandatory
ProductCategory	Type of item as classified by the merchant	String(50)	AM	Mandatory
ProductRisk	Indication of how susceptible to product is to fraud. Possible values: High Medium Low	String(50)	ALL	Not Applicable
OrderQuantity	Number of items ordered	Int	AM	Mandatory
UnitPrice	The amount of the transaction in the smallest unit of the currency	Int	AM	Mandatory

Example XML for LineItem complex elements

```
<LineItem>
  <LineItemId>1</LineItemId>
  <ProductCode>Sample</ProductCode>
  <ProductDescription>Sample Description</ProductDescription>
  <ProductCategory>Food</ProductCategory>
  <OrderQuantity>1</OrderQuantity>
  <UnitPrice>100</UnitPrice>
</LineItem>
```

3.3.2.1.7 MerchantCustomDetails

This is a set of specific details that are required by a CustomDetail Item.

Element Name	CustomDetail
Type	CustomDetail
Position	Purchase.purchaseDetails.CustomMerchantDetails
Children	Yes
Attributes	No

Elements of CustomDetail

Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
FieldValue	Custom field name	String(20)	All	Optional
FieldName	Value associated with the custom field	String(100)	All	Optional

Example XML for CustomDetail complex elements

```
<MerchantCustomDetails>
  <CustomDetail>
    <FieldName>PaymentId</FieldName>
    <FieldValue>1234</FieldValue>
  </CustomDetail>
</MerchantCustomDetails>
```

Example XML Result for CustomDetail complex elements

```
<MerchantCustomDetails>
  <PaymentId>1234</PaymentId>
</MerchantCustomDetails>
```

3.3.2.1.8 PersonalDetails

This is a set of consumer specific details that are required by the Processors.

Element Name	PersonalDetails
Type	PersonalItem
Position	Purchase.purchaseDetails Payout.payoutDetails
Children	Yes
Attributes	No

Elements of Personal Details

Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
FirstName	Consumer's First Name.	String (32)	BU	Recommended

		It must be at least 2 characters long.	EV	Optional - refer to note under PersonalDetails.stName PersonalDetails.tName
			IN	Optional
			MB	Recommended
			NS	Mandatory
			NT	Optional
			O3	Mandatory
			OC	Optional
			Risk	Recommended: The combined length of the FirstName and LastName fields may not exceed 31 characters
LastName	Consumer's Last Name.	String (32) It must be at least 2 characters long.	BU	Recommended
			EP	Optional
			EV	Optional - refer to note under 3.3.2.1.1
			IN	Optional
			MB	Recommended
			NS	Mandatory
			NT	Optional
			O3	Mandatory
			OC	Optional
			Risk	Recommended: The combined length of the FirstName and LastName fields may not exceed 31 characters
Telephone1_CountryCode	Consumer's Telephone Country Code for Telephone1. The country code is the prefix to be used when dialling to a particular country from another country.	Short. This value must not be prefixed with a plus (+) e.g. 1 for United States.	BU	Optional
			EP	Optional
			EV	Optional
			NS	Mandatory
			O3	Optional
			Risk	Recommended
Telephone1_AreaCode	Consumer's Telephone Area Code for Telephone1. This is the 3-digit area code for a particular region.	Short; US: 3 digit number (e.g. 619 for California). CA: 3 digit	BU	Optional
			EV	Optional
			IN	Optional
			NS	Mandatory
			O3	Optional

		number (e.g. 867 for Northwest Territories).	Risk	Recommended
Telephone1	Consumer's Local Home Telephone Number.	String (20); US: The 7 digit home telephone number must be supplied in this field. CA: The 7 digit home telephone number must be supplied in this field.	BU EP EV IN NS O3	Optional Optional Optional Optional Mandatory Optional
Telephone2_CountryCode	Consumer's Telephone Country Code for Telephone2. The country code is the prefix to be used when dialling to a particular country from another country.	Short; This value must not be prefixed with a plus (+) e.g. 1 for United States.	NS O3	Mandatory Optional
Telephone2_AreaCode	Consumer's Telephone Area Code for Telephone2. This is the 3-digit area code for a particular region.	Short; US: 3 digit number (e.g. 619 for California). CA: 3 digit number (e.g. 867 for Northwest Territories).	NS O3	Mandatory Optional
Telephone2	Consumer's Secondary Telephone Number	String (20)	NS O3 Risk	Mandatory Optional Optional

DateOfBirth	Consumer's date of birth.	String (10); Format "yyyy-mm-dd". If a value is supplied, it must be a valid date larger than "1899-12-31".	BU	Optional
			Risk	Recommended
Nationality	Consumer's nationality.	String (3)	BU	Optional
			O3	Optional
			Risk	Optional
Email	Consumer's Email Address.	String (64); The email address must contain an @ and a dot (.)	BU	Recommended
			EN	Recommended
			EP	Mandatory
			EV	Optional
			NS	Mandatory
			NT	Optional
			O3	Mandatory
			OC	Mandatory
			Risk	Recommended
IDNumber	The ID number required differs by country. Please see Appendix 11 for the list of countries and required ids.	String (40)	EP	Optional for Payouts
			Risk	Recommended
TypeOfID	This field indicates the type of ID sent through in the IDNumber field. The associated numeric code for the ID Type must be sent through e.g. If the consumer chose to enter in his passport registry number, then the value "1" must be submitted. If the IDNumber is not supplied, this field should be left blank. Appendix 12 lists the possible ID Types and their numeric codes.	Enum	Risk	Recommended
SSN	If the consumer's country code is US (United States), the last 4 digits of the Social Security Number (SSN) must be provided, else if the consumer's country code is CA (Canada), then the full 9 digit Social Insurance Number (SIN) must be provided in this field.	String (9)	Risk	Recommended

CPF_CNPJ	This is required for Brazilian transactions. <ul style="list-style-type: none"> • CPF – Brazilian individual taxpayer ID number. • CNPJ – ID Number of Brazilian companies. 	String (20)	All	Not Applicable
Gender	Consumer's Gender (M = male, F = Female)	String(1)	EV	Recommended

Example XML for PersonalDetails with all complex elements

```

<PersonalDetails>
    <FirstName>Jerry</FirstName>
    <LastName>Anderson</LastName>
    <Telephone1>6149135</Telephone1>
    <Telephone1_CountryCode>44</Telephone1_CountryCode>
    <Telephone1_AreaCode>105</Telephone1_AreaCode>
    <Telephone2/>
    <Telephone2_CountryCode/>
    <Telephone2_AreaCode/>
    <DateOfBirth>1982-09-14</DateOfBirth>
    <Email>customer@email.com</Email>
    <IDNumber>123456789</IDNumber>
    <TypeofID>PASSPORT</TypeofID>
    <SSN/>
    <CPF_CNPJ/>
    <Nationality>GB</Nationality>
    <Gender>M</Gender>
</PersonalDetails>

```

3.3.2.1.9 NPSItem

These are the fields or structures contained within the NPSItem structure.

Element Name	NPSItem
Type	NPSItem
Position	Purchase.methodDetails
Children	No
Attributes	No

Structure is to be sent blank.

Example XML for NPSItem elements (note that it is to be sent blank, as shown in this small portion of the Purchase request)

```
</purchaseDetails>
<methodDetails xsi:type="NPSItem" />
</Purchase>
```

3.3.2.1.10 NetellerItem

These are the fields or structures contained within the NetellerItem structure.

Element Name	NetellerItem
Type	NetellerItem
Position	Purchase.methodDetails
Children	Yes
Attributes	No

Elements of NetellerItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AccountNumber	Consumer's NETELLER Account ID	String (50)	NT	Mandatory
AccountPIN	Consumer's secure ID that was assigned, when they registered with NETELLER.	String (50)	NT	Mandatory
PaymentInstrumentNumber	This field contains an additional account number relating to the client's account at the financial institution (it is the consumer's last 4 digits of the bank account registered with NETELLER.) This needs to be provided if you would like to make use of NETELLER's instaCASH option.	String (50) This is only available for consumers with a US or Canadian bank account and the transaction currency (i.e. CurrencyCode) must be 840 or 124	NT	Optional

Example XML for NetellerItem complex elements

```
<AccountNumber>TestAccount</AccountNumber>
<AccountPIN>TestPIN</AccountPIN>
<PaymentInstrumentNumber>1234</PaymentInstrumentNumber>
```

3.3.2.1.11 Click2PayItem

These are the fields or structures contained within the Click2PayItem structure.

Element Name	Click2PayItem
Type	Click2PayItem
Position	Purchase.methodDetails
Children	Yes
Attributes	No

Elements of Click2PayItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AccountNumber	Consumer's Click2Pay User Name (in the form of an email address).	String (50)	CL	Mandatory
AccountPIN	Consumer's Click2Pay Personal Account Number (PAN) that was assigned, when they registered with Click2Pay.	String (50)	CL	Mandatory

Example XML for Click2PayDetails complex elements

```
<AccountNumber>test.account@gmail.com</AccountNumber>
<AccountPIN>TestPIN</AccountPIN>
```

3.3.2.1.12 MoneybookersItem

These are the fields or structures contained within the MoneybookersItem structure.

Element Name	MoneybookersItem
Type	MoneybookersItem
Position	Purchase.methodDetails
Children	Yes
Attributes	No

Elements of MoneybookersItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AccountNumber	Consumer's MoneyBookers User Name (in the form of an email address).	String (50)	MB	Mandatory

PaymentMethodType	Payment method for Moneybookers. For more information, please refer to Appendix 16.	Enum	MB	Optional
-------------------	---	------	----	----------

Example XML for MoneybookersItem complex elements

```
<AccountNumber>test.account@gmail.com</AccountNumber>
<PaymentMethodType>NONE</PaymentMethodType>
```

3.3.2.1.13 PaySafeCardItem

These are the fields or structures contained within the PaySafeCardItem structure.

Element Name	PaySafeCardItem
Type	PaySafeCardItem
Position	Purchase.methodDetails
Children	No
Attributes	No

Structure is to be sent blank.

Example XML for PaySafeCardItem elements (note that it is to be sent blank, as shown in this small portion of the Purchase request)

```
</purchaseDetails>
<methodDetails xsi:type="PaySafeCardItem" />
</Purchase>
```

3.3.2.1.14 ClickAndBuyItem

These are the fields or structures contained within the ClickAndBuyItem structure.

Element Name	ClickAndBuyItem
Type	ClickAndBuyItem
Position	Purchase.methodDetails
Children	Yes
Attributes	No

Elements of ClickAndBuyItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AccountNumber	Consumer's ClickAndBuy account number.	String (50)	BU	Mandatory

Example XML for ClickAndBuyItem complex elements

```
<AccountNumber>123456789</AccountNumber>
```

3.3.2.1.15 UKashItem

These are the fields or structures contained within the UKashItem structure.

Element Name	UKashItem
Type	UKashItem
Position	Purchase.methodDetails
Children	Yes
Attributes	No

Elements of UKashItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AccountNumber	Consumer's Ukash Voucher or Account number.	String (19)	UK	Mandatory
AccountPIN	Consumer's Ukash Account number PIN.	String (4)	UK	Optional. This is mandatory if the AccountNumber is 16 digits long.
AccountValue	Ukash voucher or account value.	Int	UK	Mandatory
TypeOfRedemption	This identifies the type of purchase transaction. Example: If the consumer wants to make a cash withdrawal then the value "CASH_WITHDRAWAL" must be passed. Refer to Appendix 15.	Enum	UK	Mandatory

AmountReference	This is the id returned from the 'RetrieveSettlementValue' web method.	String (50)	UK	Optional
-----------------	--	-------------	----	----------

Example XML for UKashItem complex elements

```
<AccountNumber>1234567891234567891</AccountNumber>
<AccountValue>1000</AccountValue>
<TypeOfRedemption>CASH_WITHDRAWAL</TypeOfRedemption>
<AmountReference />
```

3.3.2.1.16 eNetsItem

These are the fields or structures contained within the eNetsItem structure.

Element Name	eNetsItem
Type	eNetsItem
Position	Purchase.methodDetails
Children	No
Attributes	No

Structure is to be sent blank.

Example XML for eNetsItem complex elements (note that it is to be sent blank, as shown in this small portion of the Purchase request)

```
</purchaseDetails>
<methodDetails xsi:type="eNetsItem" />
</Purchase>
```

3.3.2.1.17 OneClickItem

These are the fields or structures contained within the OneClickItem structure.

Element Name	OneClickItem
Type	OneClickItem
Position	Purchase.methodDetails
Children	Yes
Attributes	No

Elements of OneClickItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AccountNumber	N/A	String (50)	OC	N/A
ServiceType	This indicates the type of banking service available from Envoy OneClick. Refer to Appendix 17.	Enum	OC	Mandatory

Example XML for OneClickItem complex elements

```
<ServiceType>IDEAL</ServiceType>
```

3.3.2.1.18 InstaDebitItem

These are the fields or structures contained within the InstaDebitItem structure.

Element Name	InstaDebitItem
Type	InstaDebitItem
Position	Purchase.methodDetails
Children	No
Attributes	No

Structure is to be sent blank.

Example XML for InstaDebitItem elements (note that it is to be sent blank, as shown in this small portion of the Purchase request)

```
</purchaseDetails>
<methodDetails xsi:type="InstaDebitItem" />
</Purchase>
```

3.3.2.1.19 ExperCashItem

These are the fields or structures contained within the ExperCashItem structure.

Element Name	ExperCashItem
Type	ExperCashItem
Position	Purchase.methodDetails
Children	Yes
Attributes	No

Elements of ExperCashItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
SortCode	This is the Customer's bank sort code.	String (8)	EV	ELV: Mandatory. Either the Combination of (SortCode and AccountNumber), or IBAN or PaymentReference must be supplied, but not all values can be empty.
AccountNumber	This is the Customer's bank account number	String (10)	EV	
IBAN	This is the International Bank Account Number. Standard for identifying bank accounts across national borders..	String (34)	EV	
PaymentReference	This is a Merchant configurable data field typically used to identify a customer uniquelywithin the system, regardless of the amount of accounts that the customer has.	String (15)	EV	GiroPay: SortCode is mandatory. Sofort: None of these fields are required.
TypeOfAction	This field indicates the type of purchase Action that the ExperCash gateway needs to perform. Please refer to Appendix 19 for a list of Actions that can be performed.	Enum	EV	Mandatory
DelayDays	This is the number of days on which the direct debit must be actioned. This only applies to 'BUY' type transactions.	Short	EV	Mandatory. If TypeOfAction is 'BUY', then either DelayDays or DebitDate is mandatory, but both

DebitDate	This is the date on which the direct debit must be actioned. Format "yyyy-mm-dd".	String (10)	EV	values must not be supplied.
Profile	The respective profile controls which payment and transaction methods can be selected.	Int	EV	Mandatory

Example XML for ExperCashItem complex elements

```

<SortCode>12345678</SortCode>
<AccountNumber>1234567890</AccountNumber>
<IBAN />
<PaymentReference />
<TypeOfAction>BUY</TypeOfAction>
<DelayDays>5</DelayDays>
<DebitDate />
<Profile>1</Profile>

```

3.3.2.1.20 ElectronicCheckItem

These are the fields or structures contained within the ElectronicCheckItem structure.

Element Name	ElectronicCheckItem
Type	ElectronicCheckItem
Position	Purchase.methodDetails
Children	Yes
Attributes	No

Elements of ElectronicCheckItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
FinancialAccount Number	Customer's bank routing number	String (8)	EL	Mandatory
AccountNumber	This is the Customer's bank account number	String (8)	EL	Mandatory

AccountType	This is the type of the Customer's account. Please refer to Appendix 21 for a list of eCheck account types	Enum	EL	Mandatory
TypeOfAction	This is the type of the transaction. Please refer to Appendix 22 for a list of eCheck action types	Enum	EL	Mandatory
Recurring	This is to indicate that the transaction will be recurring	Boolea	EL	Optional
StartDate	Start date for recurring transactions. Date format is 'DD/MM/YYYY'.	Date	EL	For recurring transactions it is Mandatory. For non recurring transactions it is Optional.
EndDate	End date for recurring transactions. Date format is 'DD/MM/YYYY'.	Date	EL	Optional
Frequency	Recurrence pattern for the date on which clients will be billed every month. The frequency pattern uses a subset of a standard CRON expression, comprising 3 fields separated by a white space. Please refer to Appendix 20 for frequency pattern fields.	String(12)	EL	For recurring transactions it is Mandatory. For non recurring transactions it is Optional.

Example XML for ElectronicCheckItem complex elements

```

<FinancialAccountNumber>061000104</FinancialAccountNumber>
<AccountNumber>1432456</AccountNumber>
<AccountType>CONSUMER_SAVING</AccountType>
<TypeOfAction>AUTH_AND_CAPTURE</TypeOfAction>
<Recurring>true</Recurring>
<StartDate>05/13/2010</StartDate>
<EndDate>05/13/2011</EndDate>
<Frequency>? 3 WED</Frequency>
```

3.3.2.1.21 AmazonItem

These are the fields or structures contained within the AmazonItem structure.

Element Name	AmazonItem
Type	AmazonItem
Position	Purchase.methodDetails
Children	Yes
Attributes	No

Elements of AmazonItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AmazonContractID	Amazon ContractID	String (255)	AM	Mandatory

Example XML for AmazonItem complex elements

```
<AmazonContractID>amzn1.contract.1.1.1.8a34d7a8db9a4e8be2b25d9e2e67ffff1</
AmazonContractID>
```

3.3.2.1.22 iDealItem

These are the fields or structures contained within the iDealItem structure.

Element Name	iDealItem
Type	iDealItem
Position	Purchase.methodDetails
Children	Yes

Elements of iDealItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
IssuerId	IssuerId the consumer has selected from the drop down on the merchant site.	Int	I3	Mandatory

Example XML for iDealItem complex elements

```
<iDealItem>
  <IssuerId>123</IssuerId>
</iDealItem>
```

3.3.2.1.23 iDealBankListItem

These are the fields or structures contained within the iDealItem structure.

Element Name	iDealBankListItem
Type	iDealBankListItem
Position	GetInformation.methodDetails
Children	Yes
Attributes	No

Elements of iDealItem

Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
None	-	-	-	-

Example XML for iDealBankListItem complex element

```
<iDealBankListItem>
</iDealBankListItem>
```

3.3.2.1.24 WebMoney

These are the fields or structures contained within the WebMoney structure.

Element Name	WebMoneyItem
Type	WebMoneyItem
Position	Purchase.methodDetails

Children	Yes
Attributes	No

Elements of WebMoneyItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
None	-	-	-	-

Example XML for WebMoneyItem complex elements

```
<methodDetails xsi:type="WebMoneyItem">
</methodDetails>
```

3.3.2.1.25 OBTItem

These are the fields or structures contained within the OBTItem structure.

Element Name	OBTItem
Type	OBTItem
Position	Purchase.methodDetails
Children	Yes
Attributes	No

Elements of OBTItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
ServiceType	This indicates the type of banking service available. Refer to Appendix 27.	Enum	O3	Mandatory

Example XML for OBTItem complex elements

```
<methodDetails xsi:type="OBTItem">
  <ServiceType>LA_CAIXA</ServiceType>
</methodDetails>
```

3.3.2.1.26 methodDetails (PaymentMethodItem)

This is a parent structure from which all the individual specific payment methods are inherited.

Element Name	methodDetails
Type	PaymentMethodItem
Position	Purchase
Children	Yes
Attributes	Yes

Elements of methodDetails			
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details
Please refer to the following sections for details relevant to each individual payment method:			
• AM	See section 3.3.2.1.21		
• BU	See section 3.3.2.1.14		
• CL	See section 3.3.2.1.11		
• EL	See section 3.3.2.1.20		
• EN	See section 3.3.2.1.16		
• EV	See section 3.3.2.1.19		
• I3	See section 3.3.2.1.22		
• IN	See section 3.3.2.1.18		
• MB	See section 3.3.2.1.12		
• NS	See section 3.3.2.1.9		
• NT	See section 3.3.2.1.10		
• O3	See section 3.3.2.1.25		
• OC	See section 3.3.2.1.17		
• PF	See section 3.3.2.1.13		
• UK	See section 3.3.2.1.15		
• W3	See section 3.3.2.1.24		

Attributes of methodDetails				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
xsi:type	This indicates the individual payment method involved in this transaction. Refer to Appendix 7 for the Payment Method Type options.	Enum	ALL	Mandatory

Example XML for methodDetails complex elements (no individual payment method structure involved e.g. eNets)

```
<methodDetails xsi:type="eNetsItem" />
```

Example XML for methodDetails complex elements (individual payment method structure involved e.g. Ukash)

```
<methodDetails xsi:type="UKashItem">
    <AccountNumber>1234567891234567891</AccountNumber>
    <AccountPIN />
    <AccountValue>1000</AccountValue>
    <TypeOfRedemption>CASH_WITHDRAWAL</TypeOfRedemption>
    <AmountReference />
</methodDetails>
```

3.3.2.2 Examples of XML for Purchase Requests

3.3.2.2.1 Ukash input values

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">
            <purchaseDetails>
                <TransactionId>A99_080708134612</TransactionId>
                <Description></Description>
                <TransactionDateTime>2008-07-08 13:46:12</TransactionDateTime>
                <UserIP>10.0.0.1</UserIP>
                <Language>en</Language>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails/>
                <BillingDetails>
                    <Name>Home</Name>
                    <AddressDetails>
                        <StreetLine1>Koningensplein 113</StreetLine1>
                        <StreetLine2></StreetLine2>
                        <City>Amsterdam</City>
                        <PostalCode>1011 TG</PostalCode>
                        <StateProvince>NH</StateProvince>
                        <Country>NL</Country>
                    </AddressDetails>
                    <AmountDetails>
                        <Amount>0</Amount>
                        <Exponent>2</Exponent>
                        <CurrencyCode>840</CurrencyCode>
                    </AmountDetails>
                </BillingDetails>
            </purchaseDetails>
        </Purchase>
    </soap:Body>
</soap:Envelope>
```

```

</BillingDetails>
<MerchantCustomDetails>
    <CustomDetail>
        <FieldName>PaymentId</FieldName>
        <FieldValue>1234</FieldValue>
    </CustomDetail>
</MerchantCustomDetails>
<LineItems/>
</purchaseDetails>
<methodDetails xsi:type="UKashItem">
    <AccountNumber>633718009 xxxxxx 5236</AccountNumber>
    <AccountPIN></AccountPIN>
    <AccountValue>1000</AccountValue>
    <TypeOfRedemption>CASH_WITHDRAWAL</TypeOfRedemption>
    <AmountReference/>
</methodDetails>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.2 ClickAndBuy input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">
            <purchaseDetails>
                <TransactionId>A99_080715150056</TransactionId>
                <Description />
                <TransactionDateTime>2008-07-15 15:00:56</TransactionDateTime>
                <SuccessURL>MerchantSuccessURL</SuccessURL>
                <FailureURL>MerchantFailureURL</FailureURL>
                <UserIP>10.0.0.1</UserIP>
                <Language>en</Language>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>Bridgette</FirstName>
                    <LastName>Robins</LastName>
                    <Telephone1>65012781</Telephone1>
                    <Telephone1_CountryCode>44</Telephone1_CountryCode>
                    <Telephone1_AreaCode>20</Telephone1_AreaCode>
                    <Telephone2 />
                    <Telephone2_CountryCode>0</Telephone2_CountryCode>
                    <Telephone2_AreaCode>0</Telephone2_AreaCode>
                    <DateOfBirth>1975-11-23T00:00:00</DateOfBirth>
                    <Email>customer@email.com</Email>
                    <TypeOfID>PASSPORT</TypeOfID>
                    <SSN>0</SSN>
                    <Nationality>GB</Nationality>
                </PersonalDetails>

```

```

<BillingDetails>
    <Name>Home</Name>
    <AddressDetails>
        <StreetLine1>51 Rocky Road</StreetLine1>
        <StreetLine2/>
        <City>London</City>
        <PostalCode>SW1 1HM</PostalCode>
        <StateProvince>EN</StateProvince>
        <Country>GB</Country>
    </AddressDetails>
    <AmountDetails>
        <Amount>100</Amount>
        <Exponent>2</Exponent>
        <CurrencyCode>826</CurrencyCode>
    </AmountDetails>
</BillingDetails>
<RiskDetails>
    <MerchantId>1234</MerchantId>
    <AccountName>TestRiskAccount</AccountName>
    <AccountPassword>TestRiskPwd</AccountPassword>
    <AccountNumber>TestRiskAccountNum</AccountNumber>
    <UserId/>
    <UserMachineId/>
    <UserProfile1/>
    <UserProfile2/>
    <UserProfile3/>
    <LineItems/>
</RiskDetails>
<MerchantCustomDetails>
    <CustomDetail>
        <FieldName>PaymentId</FieldName>
        <FieldValue>1234</FieldValue>
    </CustomDetail>
</MerchantCustomDetails>
<LineItems/>
</purchaseDetails>
<methodDetails xsi:type="ClickAndBuyItem">
    <AccountNumber>123456789</AccountNumber>
</methodDetails>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.3 Click2Pay input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">
            <purchaseDetails>
                <TransactionId>A99_080717124604</TransactionId>
                <Description/>
                <TransactionDateTime>2008-07-17 12:46:04</TransactionDateTime>
                <SuccessURL>MerchantSuccessURL</SuccessURL>

```

```
<FailureURL>MerchantFailureURL</FailureURL>
<UserIP>10.0.0.1</UserIP>
<Language>en</Language>
<AccountDetails>
    <AccountName>TestAccount</AccountName>
    <AccountPIN>TestPIN</AccountPIN>
    <CompanyId>A99</CompanyId>
</AccountDetails>
<PersonalDetails>
    <FirstName>Jerry</FirstName>
    <LastName>Anderson</LastName>
    <Telephone1>6149135</Telephone1>
    <Telephone1_CountryCode>44</Telephone1_CountryCode>
    <Telephone1_AreaCode>105</Telephone1_AreaCode>
    <Telephone2/>
    <Telephone2_CountryCode>0</Telephone2_CountryCode>
    <Telephone2_AreaCode>0</Telephone2_AreaCode>
    <DateOfBirth>1982-09-14T00:00:00</DateOfBirth>
    <Email>customer@email.com</Email>
    <TypeOfID>PASSPORT</TypeOfID>
    <SSN>0</SSN>
    <Nationality>GB</Nationality>
</PersonalDetails>
<BillingDetails>
    <Name>Home</Name>
    <AddressDetails>
        <StreetLine1>123 Roman Street</StreetLine1>
        <StreetLine2/>
        <City>London</City>
        <PostalCode>W4 1BT</PostalCode>
        <StateProvince>EN</StateProvince>
        <Country>GB</Country>
    </AddressDetails>
    <AmountDetails>
        <Amount>725</Amount>
        <Exponent>2</Exponent>
        <CurrencyCode>826</CurrencyCode>
    </AmountDetails>
</BillingDetails>
<RiskDetails>
    <MerchantId>1234</MerchantId>
    <AccountName>TestRiskAccount</AccountName>
    <AccountPassword>TestRiskPwd</AccountPassword>
    <AccountNumber>TestRiskAccountNum</AccountNumber>
    <UserId/>
    <UserMachineId/>
    <UserProfile1/>
    <UserProfile2/>
    <UserProfile3/>
    <LineItems/>
</RiskDetails>
<MerchantCustomDetails>
    <CustomDetail>
        <FieldName>PaymentId</FieldName>
        <FieldValue>1234</FieldValue>
    </CustomDetail>
</MerchantCustomDetails>
```

```

        <LineItems/>
    </purchaseDetails>
    <methodDetails xsi:type="Click2PayItem">
        <AccountNumber>test.account@gmail.com</AccountNumber>
        <AccountPIN>TestMethodPIN</AccountPIN>
    </methodDetails>
    </Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.4 Neteller input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">
            <purchaseDetails>
                <TransactionId>A99_080716134142</TransactionId>
                <Description/>
                <TransactionDateTime>2008-07-16 13:41:42</TransactionDateTime>
                <SuccessURL>MerchantSuccessURL</SuccessURL>
                <FailureURL>MerchantFailureURL</FailureURL>
                <UserIP>10.0.0.1</UserIP>
                <Language>en</Language>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>Caylin</FirstName>
                    <LastName>Williams</LastName>
                    <Telephone1>6149135</Telephone1>
                    <Telephone1_CountryCode>44</Telephone1_CountryCode>
                    <Telephone1_AreaCode>105</Telephone1_AreaCode>
                    <Telephone2/>
                    <Telephone2_CountryCode>0</Telephone2_CountryCode>
                    <Telephone2_AreaCode>0</Telephone2_AreaCode>
                    <DateOfBirth>1982-09-14T00:00:00</DateOfBirth>
                    <Email>customer@email.com</Email>
                    <TypeOfID>PASSPORT</TypeOfID>
                    <SSN>0</SSN>
                    <Nationality>GB</Nationality>
                </PersonalDetails>
                <BillingDetails>
                    <Name>Home</Name>
                    <AddressDetails>
                        <StreetLine1>78 Praed Street</StreetLine1>
                        <StreetLine2/>
                        <City>London</City>
                        <PostalCode>W2 1NT</PostalCode>
                        <StateProvince>EN</StateProvince>
                        <Country>GB</Country>
                    </AddressDetails>

```

```

<AmountDetails>
    <Amount>725</Amount>
    <Exponent>2</Exponent>
    <CurrencyCode>826</CurrencyCode>
</AmountDetails>
</BillingDetails>
<RiskDetails>
    <MerchantId>1234</MerchantId>
    <AccountName>TestRiskAccount</AccountName>
    <AccountPassword>TestRiskPwd</AccountPassword>
    <AccountNumber>TestRiskAccountNum</AccountNumber>
    <UserId/>
    <UserMachineId/>
    <UserProfile1/>
    <UserProfile2/>
    <UserProfile3/>
    <LineItems/>
</RiskDetails>
<MerchantCustomDetails>
    <CustomDetail>
        <FieldName>PaymentId</FieldName>
        <FieldValue>1234</FieldValue>
    </CustomDetail>
</MerchantCustomDetails>
<LineItems/>
</purchaseDetails>
<methodDetails xsi:type="NetellerItem">
    <AccountNumber>TestMethodAccount</AccountNumber>
    <AccountPIN>TestMethodPIN</AccountPIN>
    <PaymentInstrumentNumber>6789</PaymentInstrumentNumber>
</methodDetails>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.5 MoneyBookers input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">
            <purchaseDetails>
                <TransactionId>A99_080717124013</TransactionId>
                <Description/>
                <TransactionDateTime>2008-07-17 12:40:13</TransactionDateTime>
                <SuccessURL>MerchantSuccessURL</SuccessURL>
                <FailureURL>MerchantFailureURL</FailureURL>
                <UserIP>10.0.0.1</UserIP>
                <Language>nl</Language>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
            </purchaseDetails>
        </Purchase>
    </soap:Body>
</soap:Envelope>

```

```

<PersonalDetails>
    <FirstName>Mary</FirstName>
    <LastName>Bakkers</LastName>
    <Telephone1>4512834</Telephone1>
    <Telephone1_CountryCode>31</Telephone1_CountryCode>
    <Telephone1_AreaCode>20</Telephone1_AreaCode>
    <Telephone2/>
    <Telephone2_CountryCode>0</Telephone2_CountryCode>
    <Telephone2_AreaCode>0</Telephone2_AreaCode>
    <DateOfBirth>1957-04-03T00:00:00</DateOfBirth>
    <Email>customer@email.com</Email>
    <TypeOfID>PASSPORT</TypeOfID>
    <SSN>0</SSN>
    <Nationality>NL</Nationality>
</PersonalDetails>
<BillingDetails>
    <Name>Home</Name>
    <AddressDetails>
        <StreetLine1>Koningensplein 113</StreetLine1>
        <StreetLine2/>
        <City>Amsterdam</City>
        <PostalCode>1011 TG</PostalCode>
        <StateProvince>NH</StateProvince>
        <Country>NL</Country>
    </AddressDetails>
    <AmountDetails>
        <Amount>150</Amount>
        <Exponent>2</Exponent>
        <CurrencyCode>978</CurrencyCode>
    </AmountDetails>
</BillingDetails>
<RiskDetails>
    <MerchantId>1234</MerchantId>
    <AccountName>TestRiskAccount</AccountName>
    <AccountPassword>TestRiskPwd</AccountPassword>
    <AccountNumber>TestRiskAccountNum</AccountNumber>
    <UserId/>
    <UserMachineId/>
    <UserProfile1/>
    <UserProfile2/>
    <UserProfile3/>
    <LineItems/>
</RiskDetails>
<MerchantCustomDetails>
    <CustomDetail>
        <FieldName>PaymentId</FieldName>
        <FieldValue>1234</FieldValue>
    </CustomDetail>
</MerchantCustomDetails>
<LineItems/>
</purchaseDetails>
<methodDetails xsi:type="MoneybookersItem">
    <AccountNumber>test.account@gmail.com</AccountNumber>
    <PaymentMethodType>NONE</PaymentMethodType>
</methodDetails>
</Purchase>
</soap:Body>

```

```
</soap:Envelope>
```

3.3.2.2.6 PaySafeCard input values

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">
            <purchaseDetails>
                <TransactionId>A99_080715151418</TransactionId>
                <Description/>
                <TransactionDateTime>2008-07-15 15:14:18</TransactionDateTime>
                <SuccessURL>MerchantSuccessURL</SuccessURL>
                <FailureURL>MerchantFailureURL</FailureURL>
                <UserIP>10.0.0.1</UserIP>
                <Language>nl</Language>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>Joost</FirstName>
                    <LastName>Blom</LastName>
                    <Telephone1>3129011</Telephone1>
                    <Telephone1_CountryCode>31</Telephone1_CountryCode>
                    <Telephone1_AreaCode>30</Telephone1_AreaCode>
                    <Telephone2/>
                    <Telephone2_CountryCode>0</Telephone2_CountryCode>
                    <Telephone2_AreaCode>0</Telephone2_AreaCode>
                    <DateOfBirth>1967-07-13T00:00:00</DateOfBirth>
                    <Email>customer@email.com</Email>
                    <TypeOfID>PASSPORT</TypeOfID>
                    <SSN>0</SSN>
                    <Nationality>NL</Nationality>
                </PersonalDetails>
                <BillingDetails>
                    <Name>Tuis</Name>
                    <AddressDetails>
                        <StreetLine1>Kerkstraat 78</StreetLine1>
                        <StreetLine2/>
                        <City>Nieuwegein</City>
                        <PostalCode>3278 HB</PostalCode>
                        <StateProvince>ZH</StateProvince>
                        <Country>NL</Country>
                    </AddressDetails>
                    <AmountDetails>
                        <Amount>1</Amount>
                        <Exponent>2</Exponent>
                        <CurrencyCode>840</CurrencyCode>
                    </AmountDetails>
                </BillingDetails>
                <RiskDetails>
                    <MerchantId>1234</MerchantId>
                </RiskDetails>
            </purchaseDetails>
        </Purchase>
    </soap:Body>
</soap:Envelope>
```

```

<AccountName>TestRiskAccount</AccountName>
<AccountPassword>TestRiskPwd</AccountPassword>
<AccountNumber>TestRiskAccountNum</AccountNumber>
<UserId/>
<UserMachineId/>
<UserProfile1/>
<UserProfile2/>
<UserProfile3/>
<LineItems/>
</RiskDetails>
<MerchantCustomDetails>
  <CustomDetail>
    <FieldName>PaymentId</FieldName>
    <FieldValue>1234</FieldValue>
  </CustomDetail>
</MerchantCustomDetails>
<LineItems/>
</purchaseDetails>
<methodDetails xsi:type="PaySafeCardItem"/>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.7 NPS input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <Purchase xmlns="http://DPG-CPT.datacash.com/">
      <purchaseDetails>
        <TransactionId>A99_080715162705</TransactionId>
        <Description/>
        <TransactionDateTime>2008-07-15 16:27:05</TransactionDateTime>
        <SuccessURL>MerchantSuccessURL</SuccessURL>
        <FailureURL>MerchantFailureURL</FailureURL>
        <UserIP>10.0.0.1</UserIP>
        <Language>en</Language>
        <AccountDetails>
          <AccountName>TestAccount</AccountName>
          <AccountPIN>TestPIN</AccountPIN>
          <CompanyId>A99</CompanyId>
        </AccountDetails>
        <PersonalDetails>
          <FirstName>William</FirstName>
          <LastName>Dickson</LastName>
          <Telephone1>81272781</Telephone1>
          <Telephone1_CountryCode>1</Telephone1_CountryCode>
          <Telephone1_AreaCode>205</Telephone1_AreaCode>
          <Telephone2>91272791</Telephone2>
          <Telephone2_CountryCode>2</Telephone2_CountryCode>
          <Telephone2_AreaCode>305</Telephone2_AreaCode>
          <DateOfBirth>1985-12-10T00:00:00</DateOfBirth>
          <Email>customer@email.com</Email>
          <TypeOfID>PASSPORT</TypeOfID>
        </PersonalDetails>
      </purchaseDetails>
    </Purchase>
  </soap:Body>
</soap:Envelope>

```

```

<SSN>0</SSN>
<Nationality>US</Nationality>
</PersonalDetails>
<BillingDetails>
  <Name>Home</Name>
  <AddressDetails>
    <StreetLine1>51 Rocky Road</StreetLine1>
    <StreetLine2/>
    <City>San Diego</City>
    <PostalCode>20091</PostalCode>
    <StateProvince>EN</StateProvince>
    <Country>US</Country>
  </AddressDetails>
  <AmountDetails>
    <Amount>125</Amount>
    <Exponent>2</Exponent>
    <CurrencyCode>156</CurrencyCode>
  </AmountDetails>
</BillingDetails>
<ShippingDetails>
  <Name>Holiday House</Name>
  <AddressDetails>
    <StreetLine1>Greenway Terraces</StreetLine1>
    <StreetLine2>210 Bayside Road</StreetLine2>
    <City>New York</City>
    <PostalCode>10028</PostalCode>
    <StateProvince>NY</StateProvince>
    <Country>US</Country>
  </AddressDetails>
  <AmountDetails>
    <Amount>185</Amount>
    <Exponent>2</Exponent>
    <CurrencyCode>156</CurrencyCode>
  </AmountDetails>
</ShippingDetails>
<RiskDetails>
  <MerchantId>1234</MerchantId>
  <AccountName>TestRiskAccount</AccountName>
  <AccountPassword>TestRiskPwd</AccountPassword>
  <AccountNumber>TestRiskAccountNum</AccountNumber>
  <UserId/>
  <UserMachineId/>
  <UserProfile1/>
  <UserProfile2/>
  <UserProfile3/>
  <LineItems/>
</RiskDetails>
<MerchantCustomDetails>
  <CustomDetail>
    <FieldName>PaymentId</FieldName>
    <FieldValue>1234</FieldValue>
  </CustomDetail>
</MerchantCustomDetails>
<LineItems/>
</purchaseDetails>
<methodDetails xsi:type="NPSItem"/>
</Purchase>

```

```

    </soap:Body>
</soap:Envelope>
```

3.3.2.2.8 eNETS input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">
            <purchaseDetails>
                <TransactionId>A99_080618153446</TransactionId>
                <Description/>
                <TransactionDateTime>2008-06-18 15:34:46 </TransactionDateTime>
                <SuccessURL>MerchantSuccessURL</SuccessURL>
                <FailureURL>MerchantFailureURL</FailureURL>
                <UserIP>10.0.0.1</UserIP>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>Betty</FirstName>
                    <LastName>Kheng Sing Peh</LastName>
                    <Telephone1>5130771</Telephone1>
                    <Telephone1_CountryCode>65</Telephone1_CountryCode>
                    <Telephone1_AreaCode>6</Telephone1_AreaCode>
                    <Telephone2/>
                    <Telephone2_CountryCode>0</Telephone2_CountryCode>
                    <Telephone2_AreaCode>0</Telephone2_AreaCode>
                    <DateOfBirth>1957-04-03T00:00:00</DateOfBirth>
                    <Email>customer@email.com</Email>
                    <TypeOfID>PASSPORT</TypeOfID>
                    <SSN>0</SSN>
                    <Nationality>SG</Nationality>
                </PersonalDetails>
                <BillingDetails>
                    <Name>Home</Name>
                    <AddressDetails>
                        <StreetLine1>1094 Lower Delta Road</StreetLine1>
                        <StreetLine2>#07-01 Motorway Building</StreetLine2>
                        <City>Singapore</City>
                        <PostalCode>169205</PostalCode>
                        <StateProvince/>
                        <Country>SG</Country>
                    </AddressDetails>
                    <AmountDetails>
                        <Amount>100</Amount>
                        <Exponent>2</Exponent>
                        <CurrencyCode>702</CurrencyCode>
                    </AmountDetails>
                </BillingDetails>
                <ShippingDetails>
                    <Name>Work</Name>
```

```

<AddressDetails>
    <StreetLine1>Waterview Park</StreetLine1>
    <StreetLine2>Century City</StreetLine2>
    <City>Cape Town</City>
    <PostalCode/>
    <StateProvince>WC</StateProvince>
    <Country>ZA</Country>
</AddressDetails>
<AmountDetails>
    <Amount>100</Amount>
    <Exponent>2</Exponent>
    <CurrencyCode>702</CurrencyCode>
</AmountDetails>
</ShippingDetails>
<Language>EN</Language>
<RiskDetails>
    <MerchantId>1234</MerchantId>
    <AccountName>TestRiskAccount</AccountName>
    <AccountPassword>TestRiskPwd</AccountPassword>
    <AccountNumber>TestRiskAccountNum</AccountNumber>
    <UserId/>
    <UserMachineId/>
    <UserProfile1/>
    <UserProfile2/>
    <UserProfile3/>
    <LineItems/>
</RiskDetails>
<MerchantCustomDetails>
    <CustomDetail>
        <FieldName>PaymentId</FieldName>
        <FieldValue>1234</FieldValue>
    </CustomDetail>
</MerchantCustomDetails>
    <LineItems/>
</purchaseDetails>
<methodDetails xsi:type="eNetsItem"/>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.9 OneClick input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">
            <purchaseDetails>
                <TransactionId>A99_080618153446</TransactionId>
                <Description/>
                <TransactionDateTime>2008-06-18 15:34:46 </TransactionDateTime>
                <SuccessURL>MerchantSuccessURL</SuccessURL>
                <FailureURL>MerchantFailureURL</FailureURL>
                <UserIP>10.0.0.1</UserIP>
                <AccountDetails>

```

```

<AccountName>TestAccount</AccountName>
<AccountPIN>TestPIN</AccountPIN>
<CompanyId>A99</CompanyId>
</AccountDetails>
<PersonalDetails>
    <FirstName>Betty</FirstName>
    <LastName>Kheng Sing Peh</LastName>
    <Telephone1>5130771</Telephone1>
    <Telephone1_CountryCode>65</Telephone1_CountryCode>
    <Telephone1_AreaCode>6</Telephone1_AreaCode>
    <Telephone2/>
    <Telephone2_CountryCode>0</Telephone2_CountryCode>
    <Telephone2_AreaCode>0</Telephone2_AreaCode>
    <DateOfBirth>1957-04-03T00:00:00</DateOfBirth>
    <Email>customer@email.com</Email>
    <TypeOfID>PASSPORT</TypeOfID>
    <SSN>0</SSN>
    <Nationality>SG</Nationality>
</PersonalDetails>
<BillingDetails>
    <Name>Home</Name>
    <AddressDetails>
        <StreetLine1>1094 Lower Delta Road</StreetLine1>
        <StreetLine2>#07-01 Motorway Building</StreetLine2>
        <City>Singapore</City>
        <PostalCode>169205</PostalCode>
        <StateProvince/>
        <Country>NL</Country>
    </AddressDetails>
    <AmountDetails>
        <Amount>100</Amount>
        <Exponent>2</Exponent>
        <CurrencyCode>702</CurrencyCode>
    </AmountDetails>
</BillingDetails>
<ShippingDetails>
    <Name>Work</Name>
    <AddressDetails>
        <StreetLine1>Waterview Park</StreetLine1>
        <StreetLine2>Century City</StreetLine2>
        <City>Cape Town</City>
        <PostalCode/>
        <StateProvince>WC</StateProvince>
        <Country>ZA</Country>
    </AddressDetails>
    <AmountDetails>
        <Amount>100</Amount>
        <Exponent>2</Exponent>
        <CurrencyCode>702</CurrencyCode>
    </AmountDetails>
</ShippingDetails>
<Language>EN</Language>
<RiskDetails>
    <MerchantId>1234</MerchantId>
    <AccountName>TestRiskAccount</AccountName>
    <AccountPassword>TestRiskPwd</AccountPassword>
    <AccountNumber>TestRiskAccountNum</AccountNumber>

```

```

<UserId/>
<UserMachineId/>
<UserProfile1/>
<UserProfile2/>
<UserProfile3/>
<LineItems/>
</RiskDetails>
<MerchantCustomDetails>
  <CustomDetail>
    <FieldName>PaymentId</FieldName>
    <FieldValue>1234</FieldValue>
  </CustomDetail>
</MerchantCustomDetails>
<LineItems/>
</purchaseDetails>
<methodDetails xsi:type="OneClickItem">
  <ServiceType>IDEAL</ServiceType>
</methodDetails>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.10 iDeal input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <Purchase xmlns="http://DPG-CPT.datacash.com/">
      <purchaseDetails>
        <TransactionId>A99_091203142822</TransactionId>
        <TransactionDateTime>2009-12-03 14:28:22</TransactionDateTime>
        <UserIP>10.0.0.1</UserIP>
        <SuccessURL>MerchantSuccessURL</SuccessURL>
        <FailureURL>MerchantFailureURL</FailureURL>
        <Language>en</Language>
        <AccountDetails>
          <AccountName>TestAccount</AccountName>
          <AccountPIN>TestPIN</AccountPIN>
          <CompanyId>A99</CompanyId>
        </AccountDetails>
        <PersonalDetails>
          <FirstName>Wolfgang</FirstName>
          <LastName>Middendorp</LastName>
          <Telephone1>6149135</Telephone1>
          <Telephone1_CountryCode>49</Telephone1_CountryCode>
          <Telephone1_AreaCode>10</Telephone1_AreaCode>
          <Email>customer@email.com</Email>
          <Gender>M</Gender>
        </PersonalDetails>
        <BillingDetails>
          <Name>Home</Name>
          <AddressDetails>
            <StreetLine1>Berliner Straße 213</StreetLine1>
            <StreetLine2/>
            <City>Berlin</City>

```

```

<PostalCode>10825</PostalCode>
<StateProvince/>
<Country>DE</Country>
</AddressDetails>
<AmountDetails>
<Amount>5240</Amount>
<Exponent>2</Exponent>
<CurrencyCode>978</CurrencyCode>
</AmountDetails>
</BillingDetails>
<RiskDetails>
<MerchantId>1234</MerchantId>
<AccountName>TestRiskAccount</AccountName>
<AccountPassword>TestRiskPwd</AccountPassword>
<AccountNumber>TestRiskAccountNum</AccountNumber>
<UserId/>
<UserMachineId/>
<UserProfile1/>
<UserProfile2/>
<UserProfile3/>
<LineItems/>
</RiskDetails>
<MerchantCustomDetails>
<CustomDetail>
<FieldName>PaymentId</FieldName>
<FieldValue>1234</FieldValue>
</CustomDetail>
</MerchantCustomDetails>
<LineItems/>
</purchaseDetails>
<methodDetails xsi:type="iDealItem">
<IssuerId>123</IssuerId>
</methodDetails>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.11 InstaDebit input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <Purchase xmlns="http://DPG-CPT.datacash.com/">
      <purchaseDetails>
        <TransactionId>A99_091014141919</TransactionId>
        <TransactionDateTime>2009-10-14 14:19:19</TransactionDateTime>
        <UserIP>10.0.0.1</UserIP>
        <CustomerReference>2345</CustomerReference>
        <AccountDetails>
          <AccountName>TestAccount</AccountName>
          <AccountPIN>TestPIN</AccountPIN>
          <CompanyId>A99</CompanyId>
        </AccountDetails>
        <PersonalDetails>
          <FirstName>Wolfgang</FirstName>

```

```

<LastName>Middendorp</LastName>
<Telephone1>6149135</Telephone1>
<Telephone1_CountryCode>49</Telephone1_CountryCode>
<Telephone1_AreaCode>10</Telephone1_AreaCode>
<Email>customer@email.com</Email>
<TypeOfID>NONE</TypeOfID>
<SSN/>
</PersonalDetails>
<BillingDetails>
    <Name>Wolfgang Middendorp</Name>
    <AddressDetails>
        <StreetLine1>Berliner Straße 213</StreetLine1>
        <StreetLine2/>
        <City>Berlin</City>
        <PostalCode>10825</PostalCode>
        <Country>DE</Country>
    </AddressDetails>
    <AmountDetails>
        <Amount>5240</Amount>
        <Exponent>2</Exponent>
        <CurrencyCode>978</CurrencyCode>
    </AmountDetails>
</BillingDetails>
<Language>de</Language>
<RiskDetails>
    <MerchantId>1234</MerchantId>
    <AccountName>TestRiskAccount</AccountName>
    <AccountPassword>TestRiskPwd</AccountPassword>
    <AccountNumber>TestRiskAccountNum</AccountNumber>
    <UserId/>
    <UserMachineId/>
    <UserProfile1/>
    <UserProfile2/>
    <UserProfile3/>
    <LineItems/>
</RiskDetails>
<MerchantCustomDetails>
    <CustomDetail>
        <FieldName>PaymentId</FieldName>
        <FieldValue>1234</FieldValue>
    </CustomDetail>
</MerchantCustomDetails>
</purchaseDetails>
<LineItems/>
<methodDetails xsi:type="InstaDebitItem"/>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.12 ExperCash input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">

```

```

<purchaseDetails>
    <TransactionId>A99_091203142822</TransactionId>
    <TransactionDateTime>2009-12-03 14:28:22</TransactionDateTime>
    <UserIP>10.0.0.1</UserIP>
    <AccountDetails>
        <AccountName>TestAccount</AccountName>
        <AccountPIN>TestPIN</AccountPIN>
        <CompanyId>A99</CompanyId>
    </AccountDetails>
    <PersonalDetails>
        <FirstName>Wolfgang</FirstName>
        <LastName>Middendorp</LastName>
        <Telephone1>6149135</Telephone1>
        <Telephone1_CountryCode>49</Telephone1_CountryCode>
        <Telephone1_AreaCode>10</Telephone1_AreaCode>
        <Email>customer@email.com</Email>
        <Gender>M</Gender>
    </PersonalDetails>
    <BillingDetails>
        <Name>Home</Name>
        <AddressDetails>
            <StreetLine1>Berliner Straße 213</StreetLine1>
            <StreetLine2/>
            <City>Berlin</City>
            <PostalCode>10825</PostalCode>
            <StateProvince/>
            <Country>DE</Country>
        </AddressDetails>
        <AmountDetails>
            <Amount>5240</Amount>
            <Exponent>2</Exponent>
            <CurrencyCode>978</CurrencyCode>
        </AmountDetails>
    </BillingDetails>
    <RiskDetails>
        <MerchantId>1234</MerchantId>
        <AccountName>TestRiskAccount</AccountName>
        <AccountPassword>TestRiskPwd</AccountPassword>
        <AccountNumber>TestRiskAccountNum</AccountNumber>
        <UserId/>
        <UserMachineId/>
        <UserProfile1/>
        <UserProfile2/>
        <UserProfile3/>
        <LineItems/>
    </RiskDetails>
    <MerchantCustomDetails>
        <CustomDetail>
            <FieldName>PaymentId</FieldName>
            <FieldValue>1234</FieldValue>
        </CustomDetail>
    </MerchantCustomDetails>
    <Language>en</Language>
    <LineItems/>
</purchaseDetails>
<methodDetails xsi:type="ExperCashItem">
    <SortCode>12345678</SortCode>

```

```

<AccountNumber>34567890</AccountNumber>
<TypeOfAction>BUY</TypeOfAction>
<DelayDays>5</DelayDays>
<Profile>1</Profile>
</methodDetails>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.13 ElectronicCheck input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <Purchase xmlns="http://DPG-CPT.datacash.com/">
      <purchaseDetails>
        <TransactionId>T10_LENFRUCVQVQ1</TransactionId>
        <Description/>
        <TransactionDateTime>2010-10-13 06:13:20</TransactionDateTime>
        <SuccessURL>MerchantSuccessURL</SuccessURL>
        <FailureURL>MerchantFailureURL</FailureURL>
        <UserIP>10.0.0.1</UserIP>
        <AccountDetails>
          <AccountName>TestAccount</AccountName>
          <AccountPIN>TestPIN</AccountPIN>
          <CompanyId>A99</CompanyId>
        </AccountDetails>
        <PersonalDetails>
          <FirstName>Candy</FirstName>
          <LastName>Daniels</LastName>
          <Telephone1>6525971</Telephone1>
          <Telephone1_CountryCode>1</Telephone1_CountryCode>
          <Telephone1_AreaCode>212</Telephone1_AreaCode>
          <Telephone2/>
          <Telephone2_CountryCode>0</Telephone2_CountryCode>
          <Telephone2_AreaCode>0</Telephone2_AreaCode>
          <DateOfBirth>1987-08-27T00:00:00</DateOfBirth>
          <Email>Candy.daniels@hotmail.com</Email>
          <TypeOfID>NONE</TypeOfID>
          <SSN>3124</SSN>
          <Nationality>US</Nationality>
        </PersonalDetails>
        <BillingDetails>
          <Name>Home</Name>
          <AddressDetails>
            <StreetLine1>88 High Street</StreetLine1>
            <StreetLine2>Faulty Towers</StreetLine2>
            <City>New York</City>
            <PostalCode>10025</PostalCode>
            <StateProvince>NY</StateProvince>
            <Country>US</Country>
          </AddressDetails>
          <AmountDetails>
            <Amount>950</Amount>

```

```

<Exponent>2</Exponent>
<CurrencyCode>840</CurrencyCode>
</AmountDetails>
</BillingDetails>
<Language>en</Language>
<MerchantCustomDetails>
  <CustomDetail>
    <FieldName>PaymentId</FieldName>
    <FieldValue>1234</FieldValue>
  </CustomDetail>
</MerchantCustomDetails>
</purchaseDetails>
<methodDetails xsi:type="ElectronicCheckItem">
  <AccountNumber>1432456</AccountNumber>
  <AccountType>COMMERCIAL_CHECK</AccountType>
  <RoutingNumber>061000104</RoutingNumber>
  <TypeofAction>AUTH_AND_CAPTURE</TypeofAction>
  <Recurring>true</Recurring>
  <StartDate>10/10/2010</StartDate>
  <EndDate>10/12/2010</EndDate>
  <Frequency>L * ?</Frequency>
</methodDetails>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.14 Amazon input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <Purchase xmlns="http://DPG-CPT.datacash.com/">
      <purchaseDetails>
        <TransactionId>T10_111111111111</TransactionId>
        <Description/>
        <TransactionDateTime>2011-04-04 08:30:30</TransactionDateTime>
        <SuccessURL>MerchantSuccessURL</SuccessURL>
        <FailureURL>MerchantFailureURL</FailureURL>
        <UserIP>10.0.0.1</UserIP>
        <AccountDetails>
          <AccountName>Test_DataCash</AccountName>
          <AccountPIN>xxxxxxxxxxxx</AccountPIN>
          <CompanyId>X10</CompanyId>
        </AccountDetails>
        <PersonalDetails>
          <FirstName>Wolfgang</FirstName>
          <LastName>Middendorp</LastName>
          <Telephone1>6149135</Telephone1>
          <Telephone1_CountryCode>49</Telephone1_CountryCode>
          <Telephone1_AreaCode>10</Telephone1_AreaCode>
          <Telephone2/>
          <Telephone2_CountryCode>0</Telephone2_CountryCode>
          <Telephone2_AreaCode>0</Telephone2_AreaCode>
          <DateOfBirth>0001-01-01T00:00:00</DateOfBirth>
          <Email>test@hotmail.com</Email>
          <IDNumber/>
        </PersonalDetails>
      </purchaseDetails>
    </Purchase>
  </soap:Body>
</soap:Envelope>

```

```
<TypeOfID>NONE</TypeOfID>
<SSN>0</SSN>
<Nationality/>
<Gender>M</Gender>
</PersonalDetails>
<BillingDetails>
<Name>Home</Name>
<AddressDetails>
<StreetLine1>Berliner Straße 213</StreetLine1>
<StreetLine2/>
<City>Berlin</City>
<PostalCode>10825</PostalCode>
<StateProvince/>
<Country>DE</Country>
</AddressDetails>
<AmountDetails>
<Amount>5241</Amount>
<Exponent>2</Exponent>
<CurrencyCode>826</CurrencyCode>
</AmountDetails>
</BillingDetails>
<ShippingDetails>
<Name>Work</Name>
<AddressDetails>
<StreetLine1>Blue Moon Terraces</StreetLine1>
<StreetLine2>254 Paddington Road</StreetLine2>
<City>London</City>
<PostalCode>W1 3KB</PostalCode>
<StateProvince>EN</StateProvince>
<Country>GB</Country>
</AddressDetails>
<AmountDetails>
<Amount>5240</Amount>
<Exponent>2</Exponent>
<CurrencyCode>826</CurrencyCode>
</AmountDetails>
</ShippingDetails>
<RiskDetails>
<MerchantId>415</MerchantId>
<AccountName>GamingTest</AccountName>
<AccountPassword>password123</AccountPassword>
<AccountNumber>danger</AccountNumber>
<UserId/>
<UserMachineId/>
<UserProfile1/>
<UserProfile2/>
<UserProfile3/>
</RiskDetails>
<Language>en</Language>
<LineItems>
<LineItem>
<LineItemId>0</LineItemId>
<ProductCode>1</ProductCode>
<ProductDescription>1</ProductDescription>
<ProductCategory>1</ProductCategory>
<ProductRisk/>
<OrderQuantity>1</OrderQuantity>
```

```

        <UnitPrice>1</UnitPrice>
    </LineItem>
</LineItems>
</purchaseDetails>
<methodDetails xsi:type="AmazonItem">

    <AmazonContractID>amzn1.contract.1.1.1.8a34d7a8db9a4e8be2b25d9e2e67ffff1
</AmazonContractID>
    </methodDetails>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.15 WebMoney input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">
            <purchaseDetails>
                <TransactionId>T10_091014141919</TransactionId>
                <Description/>
                <TransactionDateTime>2009-10-14 14:19:19</TransactionDateTime>
                <SuccessURL>MerchantSuccessURL</SuccessURL>
                <FailureURL>MerchantFailureURL</FailureURL>
                <UserIP>10.0.0.1</UserIP>
                <AccountDetails>
                    <AccountName>Test_DataCash</AccountName>
                    <AccountPIN>xxxxxxxxxxxx</AccountPIN>
                    <CompanyId>X10</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>Jerry</FirstName>
                    <LastName>Anderson</LastName>
                    <Telephone1>6149135</Telephone1>
                    <Telephone1_CountryCode>44</Telephone1_CountryCode>
                    <Telephone1_AreaCode>105</Telephone1_AreaCode>
                    <Telephone2/>
                    <Telephone2_CountryCode>0</Telephone2_CountryCode>
                    <Telephone2_AreaCode>0</Telephone2_AreaCode>
                    <DateOfBirth>1982-09-14T00:00:00</DateOfBirth>
                    <Email>candice_williams@hotmail.com</Email>
                    <TypeOfID>PASSPORT</TypeOfID>
                    <SSN>0</SSN>
                    <Nationality>GB</Nationality>
                </PersonalDetails>
                <BillingDetails>
                    <Name>David Williams</Name>
                    <AddressDetails>
                        <StreetLine1>123 Roman Street</StreetLine1>
                        <StreetLine2/>
                        <City>London</City>
                        <PostalCode>W4 1BT</PostalCode>
                        <StateProvince>EN</StateProvince>
                        <Country>GB</Country>

```

```

</AddressDetails>
<AmountDetails>
    <Amount>725</Amount>
    <Exponent>2</Exponent>
    <CurrencyCode>826</CurrencyCode>
</AmountDetails>
</BillingDetails>
<Language>en</Language>
<RiskDetails>
    <MerchantId>1000</MerchantId>
    <AccountName>Test</AccountName>
    <AccountPassword>test123</AccountPassword>
    <AccountNumber>danger</AccountNumber>
    <UserId/>
    <UserMachineId/>
    <UserProfile1/>
    <UserProfile2/>
    <UserProfile3/>
</RiskDetails>
</purchaseDetails>
<methodDetails xsi:type="WebMoneyItem"/>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.2.16 Offline Bank Transfer input values

```

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">
            <purchaseDetails>
                <TransactionId>A99_080618153446</TransactionId>
                <Description/>
                <TransactionDateTime>2008-06-18 15:34:46 </TransactionDateTime>
                <SuccessURL>MerchantSuccessURL</SuccessURL>
                <FailureURL>MerchantFailureURL</FailureURL>
                <UserIP>10.0.0.1</UserIP>
                <CustomerReference>378KN430</CustomerReference >
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>Betty</FirstName>
                    <LastName>Kheng Sing Peh</LastName>
                    <Telephone1>5130771</Telephone1>
                    <Telephone1_CountryCode>65</Telephone1_CountryCode>
                    <Telephone1_AreaCode>6</Telephone1_AreaCode>
                    <Telephone2/>
                    <Telephone2_CountryCode>0</Telephone2_CountryCode>
                    <Telephone2_AreaCode>0</Telephone2_AreaCode>
                </PersonalDetails>
            </purchaseDetails>
        </Purchase>
    </soap:Body>
</soap:Envelope>

```

```

<DateOfBirth>1957-04-03T00:00:00</DateOfBirth>
<Email>customer@email.com</Email>
<TypeOfID>PASSPORT</TypeOfID>
<SSN>0</SSN>
<Nationality>SG</Nationality>
</PersonalDetails>
<BillingDetails>
  <Name>Home</Name>
  <AddressDetails>
    <StreetLine1>1094 Lower Delta Road</StreetLine1>
    <StreetLine2>#07-01 Motorway Building</StreetLine2>
    <City>Singapore</City>
    <PostalCode>169205</PostalCode>
    <StateProvince/>
    <Country>NL</Country>
  </AddressDetails>
  <AmountDetails>
    <Amount>100</Amount>
    <Exponent>2</Exponent>
    <CurrencyCode>702</CurrencyCode>
  </AmountDetails>
</BillingDetails>
<ShippingDetails>
  <Name>Work</Name>
  <AddressDetails>
    <StreetLine1>Waterview Park</StreetLine1>
    <StreetLine2>Century City</StreetLine2>
    <City>Cape Town</City>
    <PostalCode/>
    <StateProvince>WC</StateProvince>
    <Country>ZA</Country>
  </AddressDetails>
  <AmountDetails>
    <Amount>100</Amount>
    <Exponent>2</Exponent>
    <CurrencyCode>702</CurrencyCode>
  </AmountDetails>
</ShippingDetails>
<Language>EN</Language>
<MerchantCustomDetails>
  <CustomDetail>
    <FieldName>PaymentId</FieldName>
    <FieldValue>1234</FieldValue>
  </CustomDetail>
</MerchantCustomDetails>
<LineItems/>
</purchaseDetails>
<methodDetails xsi:type="OBTItem">
  <ServiceType>LA_CAXIA</ServiceType>
</methodDetails>
</Purchase>
</soap:Body>
</soap:Envelope>

```

3.3.2.3 Schema elements for Purchase Response

In this section, the fields that can be presented with Purchase transaction Responses will be presented along with example XML for those fields.

The top-level XML element is the `Response` element. This will be followed by the `Event` element, with the Purchase method's response details inside that, in the Purchase complex structure.

Note that various responses will result in various XML formats being returned - the following list contains some of the response fields, but you should refer to the XML samples below and in Section 3.3.2.5 for further information.

- `Response` - see section 3.3.1.2.3
 - `Event` - see section 3.3.1.2.2
 - `Purchase` - see section 3.3.2.3.4
 - `PaymentMethod` - see section 3.3.2.1.26
 - `PaymentMethod` - *specific fields*
 - `Error` - see section 3.3.1.2.1
 - `RiskResult` - see section 3.3.2.3.1

3.3.2.3.1 RiskResult

The result of Risk Screening will be displayed here.

Element Name:	RiskResult
Position	Response.Event.Purchase
Children	Yes
Attributes	No

Elements of RiskResult		
Element Name	Description	Data type / Limitations
StatusCode	This returns the Risk Status Code.	Short
StatusDescription	This is a text field used to give a short description of the Risk Status Code.	String (255)

Example XML for RiskResult complex elements

```
<RiskResult>
  <StatusCode>0</StatusCode>
  <Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
```

3.3.2.3.2 Redirect transaction specific elements

A variety of elements will appear in various positions, to serve different functions during a redirect transaction. Please refer to the individual "Position" descriptions in the

following sections, for more information on all the Redirect details.

Redirect transaction responses are characterized with "REDIRECT" in the Status field. These Purchase Payment Methods are categorized as Redirect payment methods:

- BU
- EN
- EV (Giropay & Sofort)
- I3
- IN
- MB
- NS
- OC
- PF
- W3

For all non-redirect transactions except O3, these fields are not applicable. O3 makes use of the RecommendedDisplayMessage only.

Element name:	RecommendedDisplayMessage
Position	Response.Event.Purchase.PaymentMethod
See section	3.3.2.3.3

Element name	RedirectURL
Position	Response.Event.Purchase
See section	3.3.2.3.4

Element name	PopupBrowser
Position	Response.Event.Purchase
See section	3.3.2.3.4

Element name	RedirectParameters
Position	Response.Event.Purchase
See section	3.3.2.3.4

Element name	RedirectParameter
Position	Response.Event.Purchase.RedirectParameters
See section	3.3.2.3.4

3.3.2.3.3 PaymentMethod

The details involved in the specific Payment Method of this transaction are displayed here. The top-level element "PaymentMethod" will contain the payment method for the transaction, while the sub elements will contain the rest of the details.

Element Name:	PaymentMethod
Position	Response.Event.Purchase
Children	Yes
Attributes	Yes

Attributes involved in the PaymentMethod element			
Attribute Name	Located in which Element	Description	Data type / Limitations
name	PaymentMethod	Indicates the Payment Method (Processor) involved in this transaction. Refer to Appendix 7 for the possible values.	String
isLeftToRight	RecommendedDisplayMessage	Indicates the direction of the language used in the recommended display message.	Boolean ONLY relevant to Redirect transactions.

Elements of PaymentMethod		
Element Name	Description	Data type / Limitations
ReferenceId	This is a value returned by the processor to uniquely identify the transaction.	String (50) NOT relevant to Redirect transactions.
ResponseCode	This is a return code, which gets returned by the processor.	String (10) NOT relevant to Redirect transactions.
ResponseDescription	This is a text field used to give a short text description of the response code from the processor.	String (255) NOT relevant to Redirect transactions.

RecommendedDisplayMessage	<p>This field contains the recommended display message that can be displayed to the consumer.</p> <p>The RecommendedDisplayMessage will be returned in the language which was passed through in Language field of original transaction.</p> <p>This is dependent on whether the language is supported by APG. If the language is not supported, the message will default to English.</p>	<p>String (1000)</p> <p><i>ONLY relevant to Redirect transactions.</i></p>
AmountReference	<p>This is a value returned by the Ukash processor to uniquely identify the settlement amount transaction.</p>	<p>String (50)</p> <p><i>ONLY relevant to Retrieve Settlement Values transactions (See section 0)</i></p>
<i>Payment-method specific elements</i>	<p>Note: More fields could be returned here, this is dependent on the payment method. Refer to the XML Samples for details (see section 3.3.2.5)</p>	
MerchantCustomDetails	<p>The merchant custom fields and values are returned in the element.</p> <p>The custom fields and values will be returned as follows:</p> <p><FieldName>FieldValue</FieldValue></p>	

Examples of XML for PaymentMethod complex elements (These examples are specifically for a successfully completed PaySafeCard transaction, with the payment method specific details displayed in italics. Example 1 is the redirect XML returned to the Merchant during the initial phase of the transaction, while Example 2 is the final result XML at the end of the transaction.)

1:

```
<PaymentMethod name="PF">
    <RecommendedDisplayMessage isRightToLeft="False">Please wait while
        you are being redirected to the PaySafeCard
        website.</RecommendedDisplayMessage>
</PaymentMethod>
```

2:

```
<PaymentMethod name="PF">
    <ReferenceId>A99_090909123456</ReferenceId>
    <ResponseCode>0</ResponseCode>
    <ResponseDescription>Consumed</ResponseDescription>
    <SerialNumbers>
        <SerialNumber>
            <CardSerialNumber>701130000701130</CardSerialNumber>
            <CardDispositionValue>3.50</CardDispositionValue>
        </SerialNumber>
    </SerialNumbers>
</PaymentMethod>
```

3.3.2.3.4 Purchase Response

The results of a Purchase transaction will be returned in this complex structure. Note that the details depend on the Payment Method involved – this is a very dynamic complex structure that can take on a variety of shapes, depending on the scenario.

Element Name:	Purchase
Position	Response.Event
Children	Yes
Attributes	Yes

Attributes involved in the Purchase element			
Attribute Name	Located in which Element	Description	Data type / Limitations
DPGReferenceId	Purchase	The DPGReferenceId is a unique APG transaction identifier which is generated by the APG for each merchant transaction.	Long
TransactionId	Purchase	This will return the original TransactionId that the Merchant sent through. This is for informational purposes only.	String (16)
CurrencyCode	Amount	This is the transaction currency. Please refer to Appendix 10.	Short
Exponent	Amount	This is the transaction currency exponent. Please refer to Appendix 10.	Short
CurrencyCode	SaleAmount	This is the transaction currency. Please refer to Appendix 10.	Short
Exponent	SaleAmount	This is the transaction currency exponent. Please refer to Appendix 10.	Short
Method	RedirectParameters	Indicates how the redirect parameters should be sent to the Processor. See section 3.3.2.5.	String "GET" or "POST"
contentType	RedirectParameters	Indicates how the redirect parameters should be sent to the Processor. See section 3.3.2.5.	String
Name	RedirectParameter	"Name" field of the specific Redirect parameter name-value pair.	String
Value	RedirectParameter	"Value" field of the specific Redirect parameter name-value pair.	String

Elements of Purchase		
Element Name	Description	Data type / Limitations
Purchase	Top-level XML element indicating a Purchase transaction - the DPGReferenceId and TransactionId for the transaction are also returned as attributes of this field – refer to the Attributes table above for more information.)	
PaymentMethod	See section 3.3.2.3.3	
Status	This is the response, which indicates if the transaction was accepted, or not at the processor. See Appendix 6 for possible response messages.	String (20)
Error	See section 0. This element will appear in the Purchase XML if the Status = "ERROR".	
Amount	<p>This field contains the amount taken from the client's account at the financial institution. The field type is a numeric value in cents. (The currency and exponent for the amount are also returned as attributes of this field – refer to the Attributes table above for more information.)</p> <p>Applicable to all Payment Types</p>	Int
SaleAmount	<p>This field contains the amount credited to the merchant account in the currency provided within the SaleAmount field in the merchant request. If no SaleAmount values were provided within the merchant request, the amount returned will be provided in the currency of the Amount field within the merchant request.</p> <p>Currently available for the following Payment Methods: OC</p>	Int
RiskResult	See section 3.3.2.3.1	

RecommendedClientMessage	<p>This field contains the recommended client message that can be displayed to the consumer. A list of these messages will be detailed in the Product Document. In certain instances, the exact nature of the error will be hidden from the consumer e.g. error relating to merchant account details. The true nature of the error can always be found in the ResponseDescription or Error Description response fields.</p> <p>Note: The RecommendedClientMessage field might contain end-of-line characters. Merchants should design their front-end systems in such a way that the end-of-line characters are correctly displayed to end-users.</p> <p>For example, if an HTML page is used as the display method in the front-end system, the merchant must replace the end-of-line characters in the RecommendedClientMessage field with the appropriate
 tags, otherwise the HTML front-end will not display the appropriate line breaks correctly.</p> <p>Alternatively, the contents of the RecommendedClientMessage can be included in <PRE> tags.</p> <p>The RecommendedClientMessage will be returned in the language which was passed through in Language field of original transaction. This is dependent on whether the language is supported by APG. If the language is not supported, the message will default to English.</p>	String (1000)
--------------------------	--	---------------

RecommendedDisplayMessage	<p>This field contains the recommended display message that can be displayed to the consumer.</p> <p>The RecommendedDisplayMessage will be returned in the language which was passed through in Language field of original transaction.</p> <p>This is dependent on whether the language is supported by APG. If the language is not supported, the message will default to English.</p> <p>Currently available for the following Payment Methods:</p> <ul style="list-style-type: none"> • BU • EN • EV (Only for Giropay and Sofort) • I3 • IN • MB • NS • O3 • OC • PF • W3 	String (1000)
RedirectURL	<p>URL to which the merchant needs to redirect the consumer.</p> <p>This value might also contain redirect parameters. In this scenario, the RedirectParameters element would typically not contain any RedirectParameter child elements.</p>	String (1000) ONLY relevant to Redirect transactions
PopupBrowser	<p>Merchant needs to popup a new browser depending on the value in this field.</p> <p>'true' – Popup a new browser and then redirect to the website specified in the RedirectURL field in the popup browser.</p> <p>'false' – Redirect to the website specified in the RedirectURL in the current browser session.</p>	Boolean ONLY relevant to Redirect transactions

RedirectParameters	A collection of RedirectParameters, which represent all the information necessary to successfully redirect the consumer to the RedirectURL. (The method and contentType are also returned as attributes of this field – refer to the Attributes table above for more information.)	RedirectParameter (see below) <i>ONLY relevant to Redirect transactions</i>
RedirectParameter	This Element contains information in its Attributes, which the merchant needs to transform into form variables and post them to the RedirectURL (please refer to the Attributes table above for more information). If the RedirectURL element already contains the necessary values, then the RedirectParameters element would typically not contain any RedirectParameter child elements.	String name-value pairs (contained as Attributes) <i>ONLY relevant to Redirect transactions</i>

Examples of XML for Purchase complex elements

Please refer to section 0 for examples per Payment Method.

3.3.2.4 Interpreting the Purchase Responses

There are various types of possible responses that can be returned from the APG to the Merchant. These include:

- Successful transaction
- Unsuccessful transaction
- Redirect transaction
- Event notification

Refer to section 0 for examples.

The APG can return multiple responses per transaction, and as a result the merchant needs to analyze these and perform the appropriate action. Appendix 6 details the various responses. The Status field will indicate the status of the transaction. The merchant should only credit the consumer's account when the Status=AUTHORISED, in all other cases the merchant should not credit the consumer's account.

In the case whereby the Status = REDIRECT, then the following needs to happen:

- Merchant to transform the RedirectParameter attributes into form variables
- Use the RedirectURL as the action of the form
- Display the suggested RecommendedDisplayMessage to the consumer
- Depending on the redirect the form variables needs to be posted or otherwise the Query string parameters need to be appended to the url before the consumer is redirected.

Here is an example of the Redirect response and parameters that the APG returns to the merchant:

Example of Redirect Response

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="234" TransactionId="A99_090909141714">
      <PaymentMethod name="OC">
        <RecommendedDisplayMessage isRightToLeft="False">Please wait
          while you are being redirected to the Envoy OneClick
          website.</RecommendedDisplayMessage>
      </PaymentMethod>
      <Status>REDIRECT</Status>
      <RedirectURL><! [CDATA[https://test.envoytransfers.com]]>
      </RedirectURL>
      <PopupBrowser>false</PopupBrowser>
      <RedirectParameters method="POST" contentType="application/x-
        www-form-urlencoded">
        <RedirectParameter name="merchantID" value="123321"/>
        <RedirectParameter name="customerRef"
          value="A99_090909141714"/>
        <RedirectParameter name="country" value="GB"/>
        <RedirectParameter name="amount" value="1.00"/>
        <RedirectParameter name="currency" value="EUR"/>
        <RedirectParameter name="receiveCurrency" value="USD"/>
        <RedirectParameter name="email" value="customer@email.com"/>
        <RedirectParameter name="service" value="IDEAL"/>
        <RedirectParameter name="language" value="EN"/>
      </RedirectParameters>
      <RiskResult>
        <StatusCode>0</StatusCode>
        <Message><! [CDATA[Transaction Approved ]]></Message>
      </RiskResult>
    </Purchase>
  </Event>
</Response>

```

Below is an example of a form the merchant needs to post to the Processor, following from the Response XML example above. The redirect can be done by creating a form and then using java script to submit it.

```

<form method="post" Action=" https://test.envoytransfers.com "
  enctype="application/x-www-form-urlencoded">
  <input type="hidden" name="customerRef" value="A99_090909141714">
  <input type="hidden" name="amount" value="1.00">
  <input type="hidden" name="currency" value="EUR">
  <input type="submit" name="button" value="Submit payment">
</form>

```

The consumer will then authenticate themselves on the Processor's website, and then they will be redirected to the APG Redirect Solution. At this stage the transaction will be processed further in the APG, or be passed onto the processor.

Once processing has completed, the APG will send a notification of the transaction result to the merchant's Event_URL. On signing-up to the APG system, the merchant would have provided DataCash with the Event_URL. Merchant should expect to receive all out-of-process event notifications at this URL. Refer to section 0 for examples of

event notifications (any of the responses, excluding Redirect responses, can be sent as event notifications).

3.3.2.5 Examples of XML for Purchase Responses

3.3.2.5.1 Ukash Return values

3.3.2.5.1.1 AUTHORISED

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="UKASHTESTMERCHANT">
    <Event>
        <Purchase DPGReferenceId="91598" TransactionId="A99_080708134612">
            <PaymentMethod name="UK">
                <ReferenceId>SV_2526_200807081146</ReferenceId>
                <ResponseCode>0</ResponseCode>
                <ResponseDescription>Accepted</ResponseDescription>
                <changeIssueAmount></changeIssueAmount>
                <changeIssueVoucherCurr></changeIssueVoucherCurr>
                <changeIssueVoucherNumber></changeIssueVoucherNumber>
                <changeIssueExpiryDate></changeIssueExpiryDate>
                <currencyConversion>TRUE</currencyConversion>
            </PaymentMethod>
            <Status>AUTHORISED</Status>
            <Amount CurrencyCode="840" Exponent="2">1909</Amount>
            <RiskResult>
                <StatusCode>0</StatusCode>
                <Message>
                    <![CDATA[Transaction Approved ]]>
                </Message>
            </RiskResult>
            <RecommendedClientMessage><![CDATA[Your transaction has been processed successfully.]]></RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>
```

3.3.2.5.1.2 REFUSED

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="UKASHTESTMERCHANT">
    <Event>
        <Purchase DPGReferenceId="112622" TransactionId="A99_090331144404">
            <PaymentMethod name="UK">
                <ReferenceId>SV_1819_200903311244</ReferenceId>
                <ResponseCode>1</ResponseCode>
                <ResponseDescription>Declined</ResponseDescription>
                <changeIssueAmount/>
                <changeIssueVoucherCurr/>
                <changeIssueVoucherNumber/>
                <changeIssueExpiryDate/>
                <currencyConversion/>
            </PaymentMethod>
            <Status>REFUSED</Status>
            <Amount CurrencyCode="840" Exponent="2">150</Amount>
            <RiskResult>
```

```

<StatusCode>0</StatusCode>
<Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.1.3 ERROR

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
<Event>
<Purchase DPGReferenceId="112662" TransactionId="A99_090403102334">
<PaymentMethod name="UK">
<ReferenceId/>
<ResponseCode>99</ResponseCode>
<ResponseDescription>Failed</ResponseDescription>
<changeIssueAmount/>
<changeIssueVoucherCurr/>
<changeIssueVoucherNumber/>
<changeIssueExpiryDate/>
<currencyConversion/>
</PaymentMethod>
<Status>ERROR</Status>
<Error>
<Code>301</Code>
<Description>Invalid Login and/or BrandID</Description>
</Error>
<Amount CurrencyCode="840" Exponent="2">150</Amount>
<RiskResult>
<StatusCode>0</StatusCode>
<Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.2 Click2Pay return values

3.3.2.5.2.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
<Event>
<Purchase DPGReferenceId="91683" TransactionId="A99_080717124604">
<PaymentMethod name="CL">
<ReferenceId>30a0a9b50a01001169f94884ff1efa44</ReferenceId>
<ResponseCode>000</ResponseCode>
<ResponseDescription>Transaction is
Acknowledged</ResponseDescription>

```

```

<PaymentType>FUNDS</PaymentType>
</PaymentMethod>
<Status>AUTHORISED</Status>
<Amount CurrencyCode="978" Exponent="2">1067</Amount>
<RiskResult>
    <StatusCode>0</StatusCode>
    <Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your transaction has been
processed successfully.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.2.2 REFUSED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="112623" TransactionId="A99_090331144718">
            <PaymentMethod name="CL">
                <ReferenceId/>
                <ResponseCode>200</ResponseCode>
                <ResponseDescription>You have provided invalid CLICK2PAY user
data or the account is not active. Please check and try again.
If the problem persists, please contact our helpdesk at
service@click2pay.com.</ResponseDescription>
            </PaymentMethod>
            <Status>REFUSED</Status>
            <Amount CurrencyCode="826" Exponent="2">725</Amount>
            <RiskResult>
                <StatusCode>0</StatusCode>
                <Message><! [CDATA[Transaction Approved ]]></Message>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>

```

3.3.2.5.2.3 ERROR

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="112624" TransactionId="A99_090331144826">
            <PaymentMethod name="CL">
                <ReferenceId/>
                <ResponseCode>101</ResponseCode>
                <ResponseDescription>The supplied CLICK2PAY UserName is wrong.
Please check and try again. If the problem should still exist,
please contact our customer service at
service@click2pay.com.</ResponseDescription>
            </PaymentMethod>
            <Status>ERROR</Status>

```

```

<Error>
  <Code/>
  <Description/>
</Error>
<Amount CurrencyCode="826" Exponent="2">725</Amount>
<RiskResult>
  <StatusCode>0</StatusCode>
  <Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your request could not be completed. Please contact merchant customer service for more information.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.3 ClickAndBuy return values

3.3.2.5.3.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="91671" TransactionId="A99_080715150056">
      <PaymentMethod name="BU">
        <ReferenceId>62018196</ReferenceId>
        <ResponseCode>True</ResponseCode>
        <ResponseDescription>success</ResponseDescription>
        <HTTP_X_USERID>123456789</HTTP_X_USERID>
        <HTTP_X_SESSIONID>710706429</HTTP_X_SESSIONID>
        <HTTP_X_CONTENTID>175761</HTTP_X_CONTENTID>
        <HTTP_X_PRICE>100000</HTTP_X_PRICE>
        <HTTP_X_ACTUALBDRAMOUNT>0</HTTP_X_ACTUALBDRAMOUNT>
        <HTTP_X_CURRENCY>GBP</HTTP_X_CURRENCY>
        <HTTP_X_BILLEDURRENCY>EUR</HTTP_X_BILLEDURRENCY>
        <HTTP_X_BILLEDAMOUNT>122000</HTTP_X_BILLEDAMOUNT>
      </PaymentMethod>
      <Status>AUTHORISED</Status>
      <Amount CurrencyCode="978" Exponent="2">122</Amount>
      <RiskResult>
        <StatusCode>0</StatusCode>
        <Message><! [CDATA[Transaction Approved ]]></Message>
      </RiskResult>
      <RecommendedClientMessage><! [CDATA[Your transaction has been processed successfully.]]></RecommendedClientMessage>
    </Purchase>
  </Event>
</Response>

```

3.3.2.5.3.2 REDIRECT

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="112627" TransactionId="A99_090331152024">
      <PaymentMethod name="BU">
        <RecommendedDisplayMessage isRightToLeft="False">Please wait

```

```

        while you are being redirected to the ClickAndBuy
        website.</RecommendedDisplayMessage>
    </PaymentMethod>
    <Status>REDIRECT</Status>
    <RedirectURL><! [CDATA[https://eu.clickandbuy.com/newauth/showlogou
t/http://premium-
5kz59vuejdgt.eu.clickandbuy.com/$1410$0$/ClickandBuyServerNotification.as
px]]></RedirectURL>
    <PopupBrowser>false</PopupBrowser>
    <RedirectParameters method="GET" contentType="application/x-www-
form-urlencoded">
        <RedirectParameter name="Nation" value="GB"/>
        <RedirectParameter name="price" value="100"/>
        <RedirectParameter name="lang" value="EN"/>
        <RedirectParameter name="Email" value="customer@email.com"/>
        <RedirectParameter name="user" value="customer@email.com"/>
        <RedirectParameter name="FirstName" value="Bridgette"/>
        <RedirectParameter name="LastName" value="Robins"/>
        <RedirectParameter name="Street" value="51 Rocky Road"/>
        <RedirectParameter name="City" value="London"/>
        <RedirectParameter name="ZIP" value="SW1 1HM"/>
        <RedirectParameter name="Phone" value="6465012781"/>
        <RedirectParameter name="cb_currency" value="GBP"/>
        <RedirectParameter name="DateOfBirth" value="11/23/1975"/>
        <RedirectParameter name="externalBDRID"
            value="A99_090331152024"/>
    </RedirectParameters>
    <RiskResult>
        <StatusCode>0</StatusCode>
        <Message><! [CDATA[Transaction Approved ] ]></Message>
    </RiskResult>
</Purchase>
</Event>
</Response>

```

3.3.2.5.4 Neteller return values

3.3.2.5.4.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="1760" TransactionId="A99_080716134142">
            <PaymentMethod name="NT">
                <ReferenceId>733389</ReferenceId>
                <ResponseCode>yes</ResponseCode>
                <ResponseDescription>yes</ResponseDescription>
                <FirstName>Test</FirstName>
                <LastName>Test</LastName>
                <Email>customer@email.com</Email>
                <FxRate>0.4870</FxRate>
            </PaymentMethod>
            <Status>AUTHORISED</Status>
            <Amount CurrencyCode="840" Exponent="2">1488</Amount>
            <RiskResult>
                <StatusCode>0</StatusCode>
                <Message><! [CDATA[Transaction Approved ] ]></Message>
            </RiskResult>

```

```

<RecommendedClientMessage><! [CDATA[Your transaction has been
processed successfully.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.4.2 ERROR

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
<Event>
<Purchase DPGReferenceId="112670" TransactionId="A99_090403103728">
<PaymentMethod name="NT">
<ReferenceId/>
<ResponseCode>no</ResponseCode>
<ResponseDescription>no</ResponseDescription>
</PaymentMethod>
<Status>ERROR</Status>
<Error>
<Code>1007</Code>
<Description>The Account ID or Secure ID you entered is not
valid. Please enter your correct account
information.</Description>
</Error>
<Amount CurrencyCode="826" Exponent="2">725</Amount>
<RiskResult>
<StatusCode>0</StatusCode>
<Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.5 MoneyBookers return values

3.3.2.5.5.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
<Event>
<Purchase DPGReferenceId="91682" TransactionId="A99_080717124013">
<PaymentMethod name="MB">
<ReferenceId>87712395</ReferenceId>
<ResponseCode>2</ResponseCode>
<ResponseDescription>2</ResponseDescription>
<Email>test.account@gmail.com</Email>
<AccountNumber>TestMethodAccount</AccountNumber>
<PaymentType>WLT</PaymentType>
</PaymentMethod>
<Status>AUTHORISED</Status>
<Amount CurrencyCode="710" Exponent="2">1818</Amount>
<RiskResult>
<StatusCode>0</StatusCode>

```

```

<Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Uw transactie is
geslaagd.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>
```

3.3.2.5.5.2 CANCELLED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
<Event>
<Purchase DPGReferenceId="112629" TransactionId="A99_090331152315">
<PaymentMethod name="MB">
<ReferenceId/>
<ResponseCode>-1</ResponseCode>
<ResponseDescription>-1</ResponseDescription>
<Email/>
<AccountNumber/>
<PaymentType/>
</PaymentMethod>
<Status>CANCELLED</Status>
<Amount CurrencyCode="978" Exponent="2">150</Amount>
<RiskResult>
<StatusCode>0</StatusCode>
<Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Uw transactie is op uw verzoek
geannuleerd. Neem a.u.b. contact op met de verkoopklantenservice
voor nadere informatie.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>
```

3.3.2.5.5.3 REDIRECT

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
<Event>
<Purchase DPGReferenceId="112629" TransactionId="A99_090331152315">
<PaymentMethod name="MB">
<RecommendedDisplayMessage isRightToLeft="False">Even geduld
graag. U wordt doorgeleid naar de MoneyBookers
website.</RecommendedDisplayMessage>
</PaymentMethod>
<Status>REDIRECT</Status>
<RedirectURL><! [CDATA[https://www.moneybookers.com/app/payment.pl?
sid=fcce7ddb4f5a162b429e2ed0a206132]]></RedirectURL>
<PopupBrowser>false</PopupBrowser>
<RedirectParameters method="GET" contentType="application/x-www-
form-urlencoded"/>
<RiskResult>
<StatusCode>0</StatusCode>
<Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
```

```

    </Purchase>
  </Event>
</Response>
```

3.3.2.5.5.4 PENDING

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="112636" TransactionId="A99_090331155455">
      <PaymentMethod name="MB">
        <ReferenceId>139561234</ReferenceId>
        <ResponseCode>0</ResponseCode>
        <ResponseDescription>0</ResponseDescription>
        <Email>test.account@gmail.com</Email>
        <AccountNumber>TestMethodAccount</AccountNumber>
        <PaymentType>PBT</PaymentType>
      </PaymentMethod>
      <Status>PENDING</Status>
      <Amount CurrencyCode="710" Exponent="2">18870</Amount>
      <RiskResult>
        <StatusCode>0</StatusCode>
        <Message><! [CDATA[Transaction Approved ]]></Message>
      </RiskResult>
      <RecommendedClientMessage><! [CDATA[Your request could not entirely
be completed. Please check your account later for an updated
status. You may also contact merchant customer service for more
information.]]></RecommendedClientMessage>
    </Purchase>
  </Event>
</Response>
```

3.3.2.5.6 PaySafeCard return values

3.3.2.5.6.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="5990" TransactionId="A99_090519134105">
      <PaymentMethod name="PF">
        <ReferenceId>A99_090519134105</ReferenceId>
        <ResponseCode>0</ResponseCode>
        <ResponseDescription>Consumed</ResponseDescription>
        <SerialNumbers>
          <SerialNumber>
            <CardSerialNumber>1234500000701130</CardSerialNumber>
            <CardDispositionValue>3.50</CardDispositionValue>
          </SerialNumber>
        </SerialNumbers>
      </PaymentMethod>
      <Status>AUTHORISED</Status>
      <Amount CurrencyCode="978" Exponent="2">350</Amount>
      <RiskResult>
        <StatusCode>0</StatusCode>
        <Message><! [CDATA[Transaction Approved ]]></Message>
      </RiskResult>
```

```

<RecommendedClientMessage><! [CDATA[Uw transactie is
geslaagd.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.6.2 REFUSED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="112672" TransactionId="A99_090403104148">
      <PaymentMethod name="PF">
        <ReferenceId>A99_090403104148</ReferenceId>
        <ResponseCode>L</ResponseCode>
        <ResponseDescription>Closed</ResponseDescription>
      </PaymentMethod>
      <Status>REFUSED</Status>
      <Amount CurrencyCode="978" Exponent="2">350</Amount>
      <RiskResult>
        <StatusCode>0</StatusCode>
        <Message><! [CDATA[Transaction Approved ]]></Message>
      </RiskResult>
      <RecommendedClientMessage><! [CDATA[Uw verzoek kan niet afgehandeld
worden. Neem a.u.b. contact op met de verkoopklantenservice voor
nadere informatie.]]></RecommendedClientMessage>
    </Purchase>
  </Event>
</Response>

```

3.3.2.5.6.3 REDIRECT

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="1999" TransactionId="A99_091104101009">
      <PaymentMethod name="PF">
        <RecommendedDisplayMessage isRightToLeft="False">Even geduld
        graag. U wordt doorgeleid naar de PaySafeCard
        website.</RecommendedDisplayMessage>
      </PaymentMethod>
      <Status>REDIRECT</Status>
      <RedirectURL><! [CDATA[https://customer.test.at.paysafecard.com/psc
customer/GetCustomerPanelServlet?cid=0&
currency=EUR&
mid=MerchAccount&
mtid=A99_091104101009&
amount=3.50&
language=nl]]></RedirectURL>
      <PopupBrowser>false</PopupBrowser>
      <RedirectParameters method="GET" contentType="application/x-www-
form-urlencoded"/>
      <RiskResult>
        <StatusCode>0</StatusCode>
        <Message><! [CDATA[Transaction Approved ]]></Message>
      </RiskResult>
    </Purchase>
  </Event>
</Response>

```

```

    </Purchase>
</Event>
</Response>
```

3.3.2.5.7 NPS return values

3.3.2.5.7.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
<Event>
<Purchase DPGReferenceId="91676" TransactionId="A99_080715162705">
<PaymentMethod name="NS">
<ReferenceId>953123</ReferenceId>
<ResponseCode>2</ResponseCode>
<ResponseDescription>2</ResponseDescription>
</PaymentMethod>
<Status>AUTHORISED</Status>
<Amount CurrencyCode="156" Exponent="2">125</Amount>
<RiskResult>
<StatusCode>0</StatusCode>
<Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your transaction has been
processed successfully.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>
```

3.3.2.5.7.2 REDIRECT

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
<Event>
<Purchase DPGReferenceId="112673" TransactionId="A99_090403105139">
<PaymentMethod name="NS">
<RecommendedDisplayMessage isRightToLeft="False">Please wait
while you are being redirected to NPS website. You will be
given opportunity to choose your Chinese banking
institution</RecommendedDisplayMessage>
</PaymentMethod>
<Status>REDIRECT</Status>
<RedirectURL><! [CDATA[https://payment.nps.cn/ReceiveMerchantAction
.do]]></RedirectURL>
<PopupBrowser>false</PopupBrowser>
<RedirectParameters method="GET" contentType="application/x-www-
form-urlencoded">
<RedirectParameter name="OrderMessage"
value="12345645838A13F803F93333137D03C603B1331DC7B6452313910388
F5C8039493640345438A03F833F91333037D53C613B1331D93B6C7231391E28
823C85394C7641345C78A86F847F94733077D37C6A3B1F21DF2B64623079176
88D2C8469456646745E28A36F807F94733E27D46C616B1471D16B6362316913
78886C8E2943664F645D68AF2F8E4F9F633477D96C666B1961D36B616234791
9688F6C8E694F2642545568A46F896F92733567D36C647B1841D16B6E623469
1C68856C827943764F245E48A05F835F9E233167D37C607B1871DC7B61323C7
91758896C8C694C6649645168AD6F802F94433967D36C6B6B1371DF6B6E623C
791538813C802942564F645368AB6F897F90233257DF6C616B1461D02B6C223
```

```

0291358816C8E69402644445968A56F876F9F633C27D02C654B1E41DC2B6022
355913588C7C8239403640345938A13F8C7F92333037D63C683B1131D23B673
232391738883C81394C7647545968AC6F8C6F99633167DD6C6E2B1441D96B63
623B6913788F6C8E69404647645C68AF6F826F91633C67DE2C636B1F61DD6B6
C7238491F688C6C8969446641645978A02F884F9F633577D37C656B1C71D74B
6272356915688E6C877941649745028A45F856F92733277D16C636B1561D37
B602232391138803C8029424641645978A37F896F94633567D02C625B1F61D1
6B64623C2910288E4C8569477640245958AF6F827F9B633C27D02C6E4B1951D
C2B6022355913588C7C8139403640345238A83F8C7F92333037D63C683B1131
D23B673232391738883C81394C7647545968AC6F8C6F99633167DD6C6E2B144
1D96B63623B6913788F6C8E69404647645C68AF6F826F91633C67DE2C636B1F
61DD6B6C7234591138803C8F59403649345038A43F803F93333137D03C653B1
131D33B69323C7910388C7C8239403640345938AD2F803F94333D27D03C633B
1" />
<RedirectParameter name="digest"
    value="123A456B830273A80B52DDC0B"/>
<RedirectParameter name="M_ID" value="MerchAccount"/>
</RedirectParameters>
<RiskResult>
    <StatusCode>0</StatusCode>
    <Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
</Purchase>
</Event>
</Response>

```

3.3.2.5.8 eNETS return values

3.3.2.5.8.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="347856" TransactionId="A99_200808131326">
            <PaymentMethod name="EN">
                <ReferenceId>123456789235517339</ReferenceId>
                <ResponseCode>00008_000000</ResponseCode>
                <ResponseDescription>Success</ResponseDescription>
                <PaymentMode>DD</PaymentMode>
                <Tid>A99</Tid>
                <Mid>12345</Mid>
                <TransactionCurrencyCode>702</TransactionCurrencyCode>
                <TransactionAmount>100</TransactionAmount>
                <BANK_ID>7123</BANK_ID>
                <BANK_Ref_Code>181234563</BANK_Ref_Code>
                <BANK_Remarks>No Remarks</BANK_Remarks>
                <BANK_Status>00000</BANK_Status>
                <BANK_Currency_Code>702</BANK_Currency_Code>
                <BANK_Exchange_Rate>1.0</BANK_Exchange_Rate>
                <BANK_Transaction_Amount>100</BANK_Transaction_Amount>
                <BANK_Time_Zone>+08:00</BANK_Time_Zone>
                <BANK_Txn_Date>06/08/2008</BANK_Txn_Date>
                <BANK_Txn_Time>00:47:29</BANK_Txn_Time>
                <NETSExchangeRate>1.0</NETSExchangeRate>
                <NETSTimeZone>+08:00</NETSTimeZone>
                <NETSTransactionDate>06/08/2008</NETSTransactionDate>
                <NETSTransactionTime>00:48:57</NETSTransactionTime>
                <NETSTransactionDateDateTime>06/08/2008

```

```

00:48:57</NETSTransactionDateTime>
<NETSValueDate>06/08/2008</NETSValueDate>
</PaymentMethod>
<Status>SUCCESS</Status>
<Amount CurrencyCode="702" Exponent="2">100</Amount>
<RiskResult>
    <StatusCode>0</StatusCode>
    <Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your transaction has been
processed successfully.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.8.2 REDIRECT

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="112681" TransactionId="A99 090403135837">
            <PaymentMethod name="EN">
                <RecommendedDisplayMessage isRightToLeft="False">Please wait
                    while you are being redirected to the eNETS
                    website.</RecommendedDisplayMessage>
            </PaymentMethod>
            <Status>REDIRECT</Status>
            <RedirectURL><! [CDATA[https://test.enets.sg/enets2/PaymentListener
.do]]></RedirectURL>
            <PopupBrowser>true</PopupBrowser>
            <RedirectParameters method="POST" contentType="application/x-www-
form-urlencoded">
                <RedirectParameter name="message"
                    value="sdojfawfscRUDJTiBQR1AgTUVTU0FHR0tLS0tDQpWZXJzaW9uOibTzW
N1cmVCbGFja2JveCA1IChQR1BCbGFja2JveCkNCg0Kd2NCT0E5SE1VYTd5Z0Uws
kVBUUF1bU90RGdzS05KbVE3TkI1ZEt6M2ZCb001TVhtbDFUVXVjRnJZRm1KdFlB
SQ0KNWE0QzdJOXNIWX1RQk1xZWVjejRCW1VFa0FFL0JzsksJSNUxuWld0NGtUc1d
ZQmRPNFY5N1o2VGViTUwrUnRuDQ0KNVZ1bmF0bzdtZ1NxQkt4UkdrVkJNETHNRek
FmTkcwUFNxL3pJaGxTZjBnODJIMjZCaWZ3M0RB2RnVVNPSEFNQ0KQVBvUnM2e
XE2bVpnNFhjWTAzWWQ5eVJtaW1EK0UvWW5JREphODVGNNhibX1EcEt3Qk51c3Zh
eEE4TnpabXFhWA0KQWxhd2tuemtxdnZTWZmb1FYz09JSGJid1IxUthVXZ5U1d
RWVp4V2xnTzFYOFNEUz1lbk9kNHNaOHY2SWwrNg0KSHk0VnRoczBRWnc4OFZ4R0
tMznZwaG1PMExqVVFYQTzHRGR2eGRReUNFWkUwc0ovQWVvUUtSbXZUc1QzQ2Jp0
Q0KRzNhT31RQm0yTU0wVit0d2JYWUJDQjBpbzeONU1mZk5BQUxpRnZ6c0VFTVv3
eFVCQ3zsYWJYMUziaGozUmlqRg0KU213Sk5GaWwrWE94YkdGNkxPMUFaR3R5YjZ
MMHMvRURLM2swQTZnMGhzV1RrK0w0R3duVUN5TnVqdGI1Snd2aA0KazV5UWp1UW
xtVvh1dFgyem1WQ2pZS3NTeE5GOVQ4UndmcWFqYTRCL2ZDV1dMWVFRWExRMWYxT
jUxdTVzb0p1Yw0KU3ErRHZ5VzR4cDFwNFRGa0xp2svVGg0NFZ1WmhmbFAYU3pZ
Q1hCZ2VKc1E4dmNJa0tZdE4zM1NseEJJYmsraQ0KOuptSDRhajg3QS93WHVeEI
2WDVMSUNFBnFRS2hPMXZ6YnBySXAzdgFjMU9WOGsvV210VWRtdHg0bHdoY3JOTw
0KcytLLzBTUVc1T0xPUWhYektxQ245V0Rjd21UNVRTY1I3QWxicGxCQ0JzdjMzN
nluM1hyQmtZs1Z4TWhkUmZsRg0KeDQyeU9GNFI5Y1g2QzhmUHVaVnBNOTkyMnFH
Sm80REpoEU0SnZrRW9VUGdFcFFjOXJQUXpGSjFnNnk3cXptaQ0Kd21DVHZ6Tit
RcgJCTU5yREhCMCtoeDdFU0xNTzh1NStHSFNaVEE1QkhMTXM1OxhVMitoL0o3Vz
dCRTVlemlLw0KRmtNUXhqVTQyVE1RWUU4Q0ZVYU13WXhVL2dkMzVaaFB3R3ZDe
1ZVUHdEOGT3U0srNkQ2RX1uQnpsbkJFWXd0Rw0KSW5QRFQyVmVtZXRQWXpDcURz
Q2NnV0JSYTJ0amNWYjhMUDVkcnp1Vkt1eW9CMWtXVjR1U1F6UVZpS1YwMDE5cA0

```

```

KVmpBULZFaVp1SktYQ2xtUFNzbnhadmtycjMxcWo2bm5oUWI0a3NjZ3VxWWs1Sk
ZTSnUwRmJSaXV1T09SeElzSQ0KTU1QMEFpWkd0R0dtVmtdG1oM3B2eUtHeXJOV
2Fl0XVHNkpGUHZqSk1IQ1F1SzJuSC8ydFVpSUJWSndDYzBDTg0KOWdXZXExQ1E4
aGsyRXVOb1B4c3pSdjFWUmp1N1ZQdU91dmNDVGFkM2Jwd2VSvnicW10R0tjVVB
SK1JBcnQ2dQ0KNDhiaXZsSTQ2RE1VVtg1ZncxZWN4b1FVNfY2WnFXbjQ2OEHBMj
dDdE40ZWEyMWfpOU9RV0Q2enN0bkVEallKaQ0KL0NxndKUndoMW1QdXdrcDZtW
VZRL1puZDAxM2g2MG1WUjdQKzJmVmhkTFdrT1loZG9YTnVRRjFXUKu2R2M2cw0K
bGJWQzJyOEZqK1VHUHpkMnhXcXd5Y2VZNzRFZWh5S2IveFI0K2tTdW1JOG1SOVV
zczc4Q1N0dmVvbFNZcm1HUA0KRkRCZHJ1ZXJnM2hKWEpjYTNwUz1JNkNpandtOD
Jhc2lMd2xOa1hrMEJLVmlvdFgxak4vOHN1TEpSbnplUVVHLw0KdFVDZw0KPVR6M
mkNCi0tLS0tRU5EIFBHUCBNRVNTQUdFLS0tLS0="/>
</RedirectParameters>
<RiskResult>
  <StatusCode>0</StatusCode>
  <Message><![CDATA[ Transaction Approved ]]></Message>
</RiskResult>
</Purchase>
</Event>
</Response>

```

3.3.2.5.8.3 ERROR

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="112681" TransactionId="A99_090403135837">
      <PaymentMethod name="EN">
        <ReferenceId>12345678994628167</ReferenceId>
        <ResponseCode>99</ResponseCode>
        <ResponseDescription>Failure</ResponseDescription>
        <PaymentMode/>
        <Tid>A99</Tid>
        <Mid>12345</Mid>
        <TransactionCurrencyCode>702</TransactionCurrencyCode>
        <TransactionAmount>100</TransactionAmount>
        <BANK_ID>7123</BANK_ID>
        <BANK_Ref_Code/>
        <BANK_Remarks>Bank Remarks</BANK_Remarks>
        <BANK_Status>20001</BANK_Status>
        <BANK_Currency_Code/>
        <BANK_Exchange_Rate>1</BANK_Exchange_Rate>
        <BANK_Transaction_Amount>0</BANK_Transaction_Amount>
        <BANK_Time_Zone/>
        <BANK_Txn_Date>03/04/2009</BANK_Txn_Date>
        <BANK_Txn_Time>19:47:29</BANK_Txn_Time>
        <NETSExchangeRate>1</NETSExchangeRate>
        <NETSTimeZone>+08:00</NETSTimeZone>
        <NETSTransactionDate>03/04/2009</NETSTransactionDate>
        <NETSTransactionTime>19:46:28</NETSTransactionTime>
        <NETSTransactionDateTime/>
        <NETSValueDate>03/04/2009</NETSValueDate>
      </PaymentMethod>
      <Status>ERROR</Status>
      <Error>
        <Code>000008_020001</Code>
        <Description/>
      </Error>
    </Purchase>
  </Event>
</Response>

```

```

<Amount CurrencyCode="702" Exponent="2">100</Amount>
<RiskResult>
    <StatusCode>0</StatusCode>
    <Message><! [CDATA[Transaction Approved ]]></Message>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your request could not be completed. Please contact merchant customer service for more information.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.8.4 CANCELLED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="112682" TransactionId="A99_090403140653">
            <PaymentMethod name="EN">
                <ReferenceId>123456789195436168</ReferenceId>
                <ResponseCode>9</ResponseCode>
                <ResponseDescription>Cancel</ResponseDescription>
                <PaymentMode/>
                <Tid>A99</Tid>
                <Mid>12345</Mid>
                <TransactionCurrencyCode>702</TransactionCurrencyCode>
                <TransactionAmount>100</TransactionAmount>
                <BANK_ID/>
                <BANK_Ref_Code/>
                <BANK_Remarks/>
                <BANK_Status/>
                <BANK_Currency_Code/>
                <BANK_Exchange_Rate/>
                <BANK_Transaction_Amount>0</BANK_Transaction_Amount>
                <BANK_Time_Zone/>
                <BANK_Txn_Date/>
                <BANK_Txn_Time/>
                <NETSExchangeRate/>
                <NETSTimeZone>+08:00</NETSTimeZone>
                <NETSTransactionDate/>
                <NETSTransactionTime/>
                <NETSTransactionDateTime/>
                <NETSValueDate/>
            </PaymentMethod>
            <Status>CANCELLED</Status>
            <Amount CurrencyCode="702" Exponent="2">100</Amount>
            <RiskResult>
                <StatusCode>0</StatusCode>
                <Message><! [CDATA[Transaction Approved ]]></Message>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your transaction has been cancelled as per your request. Please contact merchant customer service for more information.]]></RecommendedClientMessage>
            </Purchase>
        </Event>
    </Response>

```

3.3.2.5.9 OneClick return values
 3.3.2.5.9.1 AUTHORISED

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="23423" TransactionId="A99_200808131326">
      <PaymentMethod name="OC">
        <ReferenceId>F1IQS95A-1234-5678-B554-44E97F68F4SL</ReferenceId>
        <ResponseCode>0</ResponseCode>
        <ResponseDescription>Success</ResponseDescription>
        <uniqueReference>ASLUS1234561</uniqueReference>
        <postingDate>2006-05-26T11:35:01.33</postingDate>
        <bankCurrency>978</bankCurrency>
        <bankAmount>143.98</bankAmount>
        <countryCode>NL</countryCode>
        <bankInformation>Henrik Hamberger JSUE12349</bankInformation>
      </PaymentMethod>
      <Status>AUTHORISED</Status>
      <Amount CurrencyCode="978" Exponent="2">4655</Amount>
      <SaleAmount CurrencyCode="840" Exponent="2">6190</Amount>
      <RiskResult>
        <StatusCode>0</StatusCode>
        <Message><![CDATA[Transaction Approved ]]></Message>
      </RiskResult>
      <RecommendedClientMessage>string</RecommendedClientMessage>
    </Purchase>
  </Event>
</Response>
```

3.3.2.5.9.2 CANCELLED

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="112687" TransactionId="A99_090403141004">
      <PaymentMethod name="OC">
        <ReferenceId/>
        <ResponseCode>CANCELLED</ResponseCode>
        <ResponseDescription>CANCELLED</ResponseDescription>
        <uniqueReference>DM2US12345</uniqueReference>
        <language>nl</language>
        <country>NL</country>
        <email>test.account@gmail.com</email>
      </PaymentMethod>
      <Status>CANCELLED</Status>
      <Amount CurrencyCode="978" Exponent="2">100</Amount>
      <RiskResult>
        <StatusCode>0</StatusCode>
        <Message><![CDATA[Transaction Approved ]]></Message>
      </RiskResult>
      <RecommendedClientMessage><![CDATA[Uw transactie is op uw verzoek geannuleerd. Neem a.u.b. contact op met de verkoopklantenservice voor nadere informatie.]]></RecommendedClientMessage>
    </Purchase>
  </Event>
```

```
</Response>
```

3.3.2.5.9.3 PENDING

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="112688" TransactionId="A99_090403141122">
      <PaymentMethod name="OC">
        <ReferenceId/>
        <ResponseCode>OPEN</ResponseCode>
        <ResponseDescription>OPEN</ResponseDescription>
        <uniqueReference>DM2US123456</uniqueReference>
        <language>nl</language>
        <country>NL</country>
        <email>test.account@gmail.com</email>
      </PaymentMethod>
      <Status>PENDING</Status>
      <Amount CurrencyCode="978" Exponent="2">100</Amount>
      <RiskResult>
        <StatusCode>0</StatusCode>
        <Message><! [CDATA[Transaction Approved ]]></Message>
      </RiskResult>
      <RecommendedClientMessage><! [CDATA[Uw verzoek kan niet helemaal
afgerond worden. Raadpleeg uw account voor de laatste informatie.
U kunt ook contact opnemen met de verkoopklantenservice voor
nadere informatie.]]></RecommendedClientMessage>
    </Purchase>
  </Event>
</Response>
```

3.3.2.5.9.4 REFUSED

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="112690" TransactionId="A99_090403141723">
      <PaymentMethod name="OC">
        <ReferenceId/>
        <ResponseCode>FAILURE</ResponseCode>
        <ResponseDescription>FAILURE</ResponseDescription>
        <uniqueReference>DM2US123457</uniqueReference>
        <language>nl</language>
        <country>NL</country>
        <email>test.account@gmail.com</email>
      </PaymentMethod>
      <Status>REFUSED</Status>
      <Amount CurrencyCode="978" Exponent="2">100</Amount>
      <RiskResult>
        <StatusCode>0</StatusCode>
        <Message><! [CDATA[Transaction Approved ]]></Message>
      </RiskResult>
      <RecommendedClientMessage><! [CDATA[Uw verzoek kan niet afgehandeld
worden. Neem a.u.b. contact op met de verkoopklantenservice voor
nadere informatie.]]></RecommendedClientMessage>
    </Purchase>
  </Event>
</Response>
```

```

</Event>
</Response>
```

3.3.2.5.9.5 REDIRECT

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="112690" TransactionId="A99_090403141723">
            <PaymentMethod name="OC">
                <RecommendedDisplayMessage isRightToLeft="False">Even geduld
                    graag. U wordt doorgeleid naar de Envoy OneClick
                    website.</RecommendedDisplayMessage>
            </PaymentMethod>
            <Status>REDIRECT</Status>
            <RedirectURL><! [CDATA[https://test.envoytransfers.com]]>
            </RedirectURL>
            <PopupBrowser>false</PopupBrowser>
            <RedirectParameters method="POST" contentType="application/x-www-
                form-urlencoded">
                <RedirectParameter name="merchantID" value="MerchAccount"/>
                <RedirectParameter name="customerRef"
value="A99_090403141723"/>
                <RedirectParameter name="country" value="NL"/>
                <RedirectParameter name="amount" value="1.00"/>
                <RedirectParameter name="currency" value="EUR"/>
                <RedirectParameter name="receiveCurrency" value="USD"/>
                <RedirectParameter name="email" value="test.account@gmail.com"/>
                <RedirectParameter name="service" value="IDEAL"/>
                <RedirectParameter name="language" value="NL"/>
            </RedirectParameters>
            <RiskResult>
                <StatusCode>0</StatusCode>
                <Message><! [CDATA[Transaction Approved ] ]></Message>
            </RiskResult>
        </Purchase>
    </Event>
</Response>
```

3.3.2.5.9.6 ERROR

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="9985" TransactionId="A99_091209114405">
            <PaymentMethod name="OC">
                <ReferenceId/>
                <ResponseCode>ERROR</ResponseCode>
                <ResponseDescription>Om technische redenen kan de bank uw
                    betaling op dit moment niet verwerken. Probeer u het
                    alstublieft later nog eens.</ResponseDescription>
                <uniqueReference>DC4GB9890955</uniqueReference>
                <language>nl</language>
                <country>NL</country>
                <email>test.account@gmail.com</email>
            </PaymentMethod>
```

```

<Status>ERROR</Status>
<Amount CurrencyCode="978" Exponent="2">100</Amount>
<RiskResult>
    <StatusCode>0</StatusCode>
    <StatusDescription><! [CDATA[Transaction
Approved]]></StatusDescription>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Uw verzoek kan niet afgehandeld
worden. Neem a.u.b. contact op met de verkoopklantenservice voor
nadere informatie.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.10 InstaDebit return values

3.3.2.5.10.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="45253" TransactionId="A99_45723456249">
            <PaymentMethod name="IN">
                <ReferenceId>56235634</ReferenceId>
                <ResponseCode>S</ResponseCode>
                <ResponseDescription>Success</ResponseDescription>
                <user_id>test.user</user_id>
                <txn_type>T</txn_type>
                <merchant_user_id>34563DSFSDF</merchant_user_id>
                <txn_fee>0.10</txn_fee>
                <channel/>
            </PaymentMethod>
            <Status>AUTHORISED</Status>
            <Amount CurrencyCode="978" Exponent="2">100</Amount>
            <RiskResult>
                <StatusCode>0</StatusCode>
                <StatusDescription><! [CDATA[Transaction
Approved]]></StatusDescription>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your transaction has been
processed successfully..]]></RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>

```

3.3.2.5.10.2 ERROR

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="4534" TransactionId="A99_345723494743">
            <PaymentMethod name="IN">
                <ReferenceId/>
                <ResponseCode/>
                <ResponseDescription/>
            </PaymentMethod>
            <Status>ERROR</Status>

```

```

<Error>
  <Code>812</Code>
  <Description>The InstaDebit notification is
invalid</Description>
</Error>
<Amount CurrencyCode="978" Exponent="2">100</Amount>
<RiskResult>
  <StatusCode>0</StatusCode>
  <StatusDescription><![CDATA[Transaction
Approved]]></StatusDescription>
</RiskResult>
<RecommendedClientMessage><![CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.10.3 REDIRECT

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Purchase DPGReferenceId="45253" TransactionId="A99_45723456249">
      <PaymentMethod name="IN">
        <RecommendedDisplayMessage isRightToLeft="False">Even geduld
        graag. U wordt doorgeleid naar de InstaDebit
        website</RecommendedDisplayMessage>
      </PaymentMethod>
      <Status>REDIRECT</Status>

        <RedirectURL>https://staging.instadebit.com/instadebit/merGatew
ay.do </RedirectURL>
      <PopupBrowser>false</PopupBrowser>
      <RedirectParameters method="POST" contentType="application/x-www-
form-urlencoded">
        <RedirectParameter name="merchant_id" value="4SDFS"/>
        <RedirectParameter name="merchant_user_id" value="DFG4523"/>
        <RedirectParameter name="merchant_txn_num"
value="A99_45723456249"/>
        <RedirectParameter name="txn_amount" value="1.00"/>
        <RedirectParameter name="txn_currency" value="EUR"/>
        <RedirectParameter name="first_name" value="John"/>
        <RedirectParameter name="last_name" value="Doe"/>
        <RedirectParameter name="addr_1" value="Unit 11, Forest Edge
Villa"/>
        <RedirectParameter name="addr_2" value="Alpharetta"/>
        <RedirectParameter name="city" value="Atlanta"/>
        <RedirectParameter name="state" value="GA"/>
        <RedirectParameter name="zip" value="56476"/>
        <RedirectParameter name="country" value="US"/>
        <RedirectParameter name="hph_area_code" value="021"/>
        <RedirectParameter name="hph_local_number" value="7496734"/>
        <RedirectParameter name="dob_day" value="30"/>
        <RedirectParameter name="dob_month" value="11"/>
        <RedirectParameter name="dob_year" value="2009"/>
        <RedirectParameter name="return_url" value="https://dpg-

```

```

        cpt.datacash.com/Notification/RedirectHandlers/InstaDebit.aspx"
    />
    <RedirectParameter name="merchant_sub_id" value="123"/>
</RedirectParameters>
<RiskResult>
    <StatusCode>0</StatusCode>
    <StatusDescription><! [CDATA[Transaction
Approved]]></StatusDescription>
</RiskResult>
</Purchase>
</Event>
</Response>

```

3.3.2.5.10.4 REFUSED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
<Event>
    <Purchase DPGReferenceId="112690" TransactionId="A99_090403141723">
        <PaymentMethod name="IN">
            <ReferenceId>DSF57898SDF</ReferenceId>
            <ResponseCode>98</ResponseCode>
            <ResponseDescription>Transaction is terminated. The transaction
is declined (generic error).</ResponseDescription>
            <user_id>test.user</user_id>
            <txn_type>T</txn_type>
            <merchant_user_id>34DSFSDF</merchant_user_id>
            <txn_fee>0.10</txn_fee>
            <channel/>
        </PaymentMethod>
        <Status>REFUSED</Status>
        <Amount CurrencyCode="978" Exponent="2">100</Amount>
        <RiskResult>
            <StatusCode>0</StatusCode>
            <StatusDescription><! [CDATA[Transaction Approved
]]></StatusDescription>
        </RiskResult>
        <RecommendedClientMessage><! [CDATA[Uw verzoek kan niet afgehandeld
worden. Neem a.u.b. contact op met de verkoopklantenservice voor
nadere informatie.]]></RecommendedClientMessage>
    </Purchase>
</Event>
</Response>

```

3.3.2.5.10.5 CANCELLED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
<Event>
    <Purchase DPGReferenceId="112687" TransactionId="A99_090403141004">
        <PaymentMethod name="IN">
            <ReferenceId/>
            <ResponseCode>99</ResponseCode>
            <ResponseDescription>The transaction is cancelled by the
customer</ResponseDescription>
            <user_id>test.user</user_id>

```

```

<txn_type>T</txn_type>
<merchant_user_id>34DSFSDF</merchant_user_id>
<txn_fee>0.10</txn_fee>
<channel/>
</PaymentMethod>
<Status>CANCELLED</Status>
<Amount CurrencyCode="978" Exponent="2">100</Amount>
<RiskResult>
    <StatusCode>0</StatusCode>
    <StatusDescription><! [CDATA[Transaction Approved
    ]]></StatusDescription>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Uw transactie is op uw verzoek
geannuleerd. Neem a.u.b. contact op met de verkoopklantenservice
voor nadere informatie.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.11 ExperCash return values

3.3.2.5.11.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="113835" TransactionId="A99_091127161216">
            <PaymentMethod name="EV">
                <ReferenceId>00000000000040078700</ReferenceId>
                <ResponseCode>100</ResponseCode>
                <ResponseDescription>Transaction
successfull</ResponseDescription>
                <TimeStamp>2009-11-27 15:12:25</TimeStamp>
                <PaymentId>DD0675275361000</PaymentId>
            </PaymentMethod>
            <Status>AUTHORISED</Status>
            <Amount CurrencyCode="978" Exponent="2">5240</Amount>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your transaction has been
processed successfully..]]></RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>

```

3.3.2.5.11.2 PENDING

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="113836" TransactionId="A99_091127163001">
            <PaymentMethod name="EV">
                <ReferenceId>00000000000040079890</ReferenceId>
                <ResponseCode>100</ResponseCode>
                <ResponseDescription>Transaction

```

```

        successfull</ResponseDescription>
        <TimeStamp>2009-11-27 15:30:19</TimeStamp>
    </PaymentMethod>
    <Status>PENDING</Status>
    <Amount CurrencyCode="978" Exponent="2">5240</Amount>
    <RiskResult>
        <StatusCode>100</StatusCode>
        <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
    </RiskResult>
    <RecommendedClientMessage><! [CDATA[Your request could not entirely
be completed. Please check your account later for an updated
status. You may also contact merchant customer service for more
information. ]]></RecommendedClientMessage>
    </Purchase>
</Event>
</Response>
```

3.3.2.5.11.3 ERROR

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
    <Event>
        <Purchase DPGReferenceId="113843" TransactionId="A99091130134345">
            <PaymentMethod name="EV">
                <ReferenceId/>
                <ResponseCode/>
                <ResponseDescription/>
            </PaymentMethod>
            <Status>ERROR</Status>
            <Error>
                <Code>402</Code>
                <Description>Either SortCode and AccountNumber OR IBAN OR
PaymentReference must be supplied - no other combinations
permitted.</Description>
            </Error>
            <Amount CurrencyCode="978" Exponent="2">5240</Amount>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your request could not be
completed. Please contact merchant customer service for more
information. ]]></RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>
```

3.3.2.5.11.4 REFUSED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="113851" TransactionId="A99_091201111145">
            <PaymentMethod name="EV">
                <ReferenceId>00000000000040420664</ReferenceId>
                <ResponseCode>300</ResponseCode>
```

```

<ResponseDescription>REJECTED: possibly wrong bank account
number or bank code</ResponseDescription>
<TimeStamp>2009-12-01 10:12:01</TimeStamp>
</PaymentMethod>
<Status>REFUSED</Status>
<Amount CurrencyCode="978" Exponent="2">5240</Amount>
<RiskResult>
    <StatusCode>100</StatusCode>
    <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>
```

3.3.2.5.11.5 REDIRECT (SOFORT)

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="3832" TransactionId="A99_100423095517">
            <PaymentMethod name="EV">
                <RecommendedDisplayMessage isRightToLeft="False"/>
            </PaymentMethod>
            <Status>REDIRECT</Status>

            <RedirectURL><! [CDATA[https://www.sofortueberweisung.de/payment/start]]>
            </RedirectURL>
            <PopupBrowser>true</PopupBrowser>
            <RedirectParameters method="POST" contentType="application/x-www-
form-urlencoded">
                <RedirectParameter name="user_id" value="123"/>
                <RedirectParameter name="project_id" value="7445"/>
                <RedirectParameter name="amount" value="52.40"/>
                <RedirectParameter name="currency_id" value="EUR"/>
                <RedirectParameter name="reason_1" value="SU1000"/>
                <RedirectParameter name="reason_2" value="3832"/>
                <RedirectParameter name="user_variable_0" value="1237755"/>
                <RedirectParameter name="user_variable_1" value=""/>
                <RedirectParameter name="user_variable_2" value="10.0.0.1"/>
                <RedirectParameter name="hash"
value="f1a51d213fe82b90037d6708035f46704cf6d353"/>
            </RedirectParameters>
            <RiskResult>
                <StatusCode>0</StatusCode>
                <StatusDescription><! [CDATA[Transaction
Approved]]></StatusDescription>
            </RiskResult>
        </Purchase>
    </Event>
</Response>
```

3.3.2.5.11.6 REDIRECT (GIROPAY)

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchantAccount">
    <Event>
```

```

<Purchase DPGReferenceId="16078" TransactionId="A99_100524073622">
    <PaymentMethod name="EV">
        <RecommendedDisplayMessage isRightToLeft="False"/>
    </PaymentMethod>
    <Status>REDIRECT</Status>

    <RedirectURL><! [CDATA[https://giropaytest1.fiducia.de/ShopSystem/bank/BankEntry]]></RedirectURL>
        <PopupBrowser>true</PopupBrowser>
        <RedirectParameters method="POST" contentType="application/x-www-form-urlencoded">
            <RedirectParameter name="redirectData"><! [CDATA[operatorId=002&merchantId=MerchantAccount&txId=F000007SOJ&merchantTxId=56749466&customerBankBLZ=44441111&shopEndSuccessURL=https%3A%2F%2Fdpg-cpt.datacash.com%2FNotification%2FRedirectHandlers%2FExperCash.aspx%3Ftai d%3D56749466&shopEndFailureURL=https%3A%2F%2Fdpg-cpt.datacash.com%2FNotification%2FRedirectHandlers%2FExperCash.aspx%3Faborted%3Dtrue&beneficiaryName1=easyDebit&beneficiaryName2=&beneficiaryAcctNo=1234567890&beneficiaryBankBLZ=44441111&amount=5240&currency=EUR&paymentPurposeLine1=GP0808502474000&paymentPurposeLine2=GIROPAY0028993390F000007SOJ&operatorSignature=kZ7X72%2F88h9pJGx%2BulrJ7gX%2FjEwBOmC8HZ1F%2FiG%2FJYXwtW2fLRe12biJ1vkqB6%2FhPJfJVyTkoiGkcqdTy3lVWttJTvFQYe0YkurrAoOQA2k6wseVI6Kof2EaEy%2F7ofRVN6MziZrx6WJFjbQ8Bz3B%2FFJrFHrlhAmvFekN1SwyZy0%3D] ]></RedirectParameter>
        </RedirectParameters>
        <RiskResult>
            <StatusCode>9</StatusCode>
            <StatusDescription><! [CDATA[Error] ]></StatusDescription>
        </RiskResult>
    </Purchase>
</Event>
</Response>

```

3.3.2.5.12 iDeal return values

3.3.2.5.12.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="113835" TransactionId="A99_091127161216">
            <PaymentMethod name="I3">
                <ReferenceId>00000000000040078700</ReferenceId>
                <ResponseCode>100</ResponseCode>
                <ResponseDescription>Transaction successfull</ResponseDescription>
                <TimeStamp>2009-11-27 15:12:25</TimeStamp>
                <PaymentId>DD0675275361000</PaymentId>

```

```

</PaymentMethod>
<Status>AUTHORISED</Status>
<Amount CurrencyCode="978" Exponent="2">5240</Amount>
<RiskResult>
    <StatusCode>100</StatusCode>
    <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your transaction has been
processed successfully..]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.12.2 PENDING

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
<Event>
    <Purchase DPGReferenceId="113836" TransactionId="A99_091127163001">
        <PaymentMethod name="I3">
            <ReferenceId>00000000000040079890</ReferenceId>
            <ResponseCode>100</ResponseCode>
            <ResponseDescription>Transaction
            successfull</ResponseDescription>
            <TimeStamp>2009-11-27 15:30:19</TimeStamp>
        </PaymentMethod>
        <Status>PENDING</Status>
        <Amount CurrencyCode="978" Exponent="2">5240</Amount>
        <RiskResult>
            <StatusCode>100</StatusCode>
            <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
        </RiskResult>
        <RecommendedClientMessage><! [CDATA[Your request could not entirely
be completed. Please check your account later for an updated
status. You may also contact merchant customer service for more
information. ]]></RecommendedClientMessage>
    </Purchase>
    </Event>
</Response>

```

3.3.2.5.12.3 ERROR

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
<Event>
    <Purchase DPGReferenceId="113843" TransactionId="A99091130134345">
        <PaymentMethod name="I3">
            <ReferenceId/>
            <ResponseCode/>
            <ResponseDescription/>
        </PaymentMethod>
        <Status>ERROR</Status>
        <Error>
            <Code>451</Code>
            <Description>iDealItem.Issuer_Id is invalid.</Description>
        </Error>
</Response>

```

```

<Amount CurrencyCode="978" Exponent="2">5240</Amount>
<RiskResult>
    <StatusCode>100</StatusCode>
    <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your request could not be
completed. Please contact merchant customer service for more
information. ]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.12.4 REFUSED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Purchase DPGReferenceId="113851" TransactionId="A99_091201111145">
            <PaymentMethod name="I3">
                <ReferenceId>00000000000040420664</ReferenceId>
                <ResponseCode>102</ResponseCode>
                <ResponseDescription>REJECTED: possibly wrong bank account
number or bank code</ResponseDescription>
                <TimeStamp>2009-12-01 10:12:01</TimeStamp>
            </PaymentMethod>
            <Status>REFUSED</Status>
            <Amount CurrencyCode="978" Exponent="2">5240</Amount>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your request could not be
completed. Please contact merchant customer service for more
information. ]]></RecommendedClientMessage>
            </Purchase>
        </Event>
    </Response>

```

3.3.2.5.12.5 REDIRECT

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="50764">
    <Event>
        <Purchase DPGReferenceId="130212" TransactionId="T10_110705143745">
            <PaymentMethod name="I3">
                <RecommendedDisplayMessage isRightToLeft="False">Please wait
while you are being redirected to the iDeal
website.</RecommendedDisplayMessage>
            </PaymentMethod>
            <Status>REDIRECT</Status>
            <RedirectURL><! [CDATA[https://some.ideal.url]]></RedirectURL>
            <PopupBrowser>true</PopupBrowser>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
        </Purchase>
    </Response>

```

```

</Event>
</Response>
```

3.3.2.5.13 ElectronicCheck return values

3.3.2.5.13.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchantAccount">
    <Event>
        <Purchase DPGReferenceId="41570" TransactionId="T10_SOCGYIEEMFZT">
            <PaymentMethod name="EL">
                <ReferenceId>4CB551629EFCC4A40CFAB958CF97A544DF9154E0</ReferenceId>
                <ResponseCode>1</ResponseCode>
                <ResponseDescription>Approved</ResponseDescription>
                <TimeStamp>20101013022746</TimeStamp>
                <PaymentId>123456</PaymentId>
            </PaymentMethod>
            <Status>AUTHORISED</Status>
            <Amount CurrencyCode="840" Exponent="2">950</Amount>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your transaction has been
processed successfully.]]></RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>
```

3.3.2.5.13.2 SENT FOR CAPTURE

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchantAccount">
    <Event>
        <Purchase DPGReferenceId="41571" TransactionId="T10_ECSGE1A9JP8J">
            <PaymentMethod name="EL">
                <ReferenceId>4CB551EF4C6B600C6BD174ECBD2527925E2F547C</ReferenceId>
                <ResponseCode>1</ResponseCode>
                <ResponseDescription>Approved</ResponseDescription>
                <TimeStamp>20101013023008</TimeStamp>
                <PaymentId>123456</PaymentId>
            </PaymentMethod>
            <Status>SENT_FOR_CAPTURE</Status>
            <Amount CurrencyCode="840" Exponent="2">950</Amount>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your transaction has been
processed successfully.]]></RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>
```

3.3.2.5.13.3 CAPTURED (Recurring transaction)

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchantAccount">
    <Event>
        <Purchase DPGReferenceId="107369" TransactionId="T10_ZTOC44MTDZQ2">
            <PaymentMethod name="EL">
                <ReferenceId>20238.0426f</ReferenceId>
                <ResponseCode>Deposit</ResponseCode>
                <ResponseDescription>The funds have been captured successfully.</ResponseDescription>
                <TimeStamp>08/31/2010</TimeStamp>
                <Recurring>True</Recurring>
            <OriginalTransactionId>T10_GUSWMVR2PNW</OriginalTransactionId>
            </PaymentMethod>
            <Status>CAPTURED</Status>
            <Amount CurrencyCode="840" Exponent="2">950</Amount>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your transaction has been processed successfully.]]></RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>
```

3.3.2.5.13.4 CAPTURED (Non-recurring transaction)

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchantAccount">
    <Event>
        <Purchase DPGReferenceId="107360" TransactionId="T10_GBCOWVOTNYDH">
            <PaymentMethod name="EL">
                <ReferenceId>20238.0426f</ReferenceId>
                <ResponseCode>Deposit</ResponseCode>
                <ResponseDescription>The funds have been captured successfully.</ResponseDescription>
                <TimeStamp>08/31/2010</TimeStamp>
                <Recurring>False</Recurring>
            </PaymentMethod>
            <Status>CAPTURED</Status>
            <Amount CurrencyCode="840" Exponent="2">120000</Amount>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your transaction has been processed successfully.]]></RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>
```

3.3.2.5.13.5 ERROR

```
<?xml version="1.0" encoding="utf-8"?>
```

```

<Response version="1.0" AccountName="MerchantAccount">
  <Event>
    <Purchase DPGReferenceId="95145" TransactionId="T10_F0MZ9MZUX71R">
      <PaymentMethod name="EL">
        <ReferenceId>4C91D9E75A8B2BD9B4771764B3FB555635855493</ReferenceId>
          <ResponseCode>0</ResponseCode>
          <ResponseDescription>Bad Account Number
Data</ResponseDescription>
          <TimeStamp>20100916044839</TimeStamp>
        </PaymentMethod>
        <Status>ERROR</Status>
        <Amount CurrencyCode="840" Exponent="2">950</Amount>
        <RiskResult>
          <StatusCode>100</StatusCode>
          <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
        </RiskResult>
        <RecommendedClientMessage><! [CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
      </Purchase>
    </Event>
  </Response>

```

3.3.2.5.13.6 REFUSED (Recurring transaction)

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchantAccount">
  <Event>
    <Purchase DPGReferenceId="122110" TransactionId="T10_GDGKAWWWLWAG">
      <PaymentMethod name="EL">
        <ReferenceId>62954.3642f</ReferenceId>
        <ResponseCode>REJECT</ResponseCode>
        <ResponseDescription>Failed a front-end
edit</ResponseDescription>
        <TimeStamp>09/01/2010</TimeStamp>
        <Recurring>True</Recurring>
        <OriginalTransactionId>T10_GUSWMVR2PNW</OriginalTransactionId>
      </PaymentMethod>
      <Status>REFUSED</Status>
      <Amount CurrencyCode="840" Exponent="2">950</Amount>
      <RiskResult>
        <StatusCode>100</StatusCode>
        <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
      </RiskResult>
      <RecommendedClientMessage><! [CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
    </Purchase>
  </Event>
</Response>

```

3.3.2.5.13.7 REFUSED (Non-recurring transaction)

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchantAccount">
  <Event>

```

```

<Purchase DPGReferenceId="122110" TransactionId="T10_GDGKAWWWLWAG">
    <PaymentMethod name="EL">
        <ReferenceId>62954.3642f</ReferenceId>
        <ResponseCode>REJECT</ResponseCode>
        <ResponseDescription>Failed a front-end
edit</ResponseDescription>
        <TimeStamp>09/01/2010</TimeStamp>
        <Recurring>False</Recurring>
    </PaymentMethod>
    <Status>REFUSED</Status>
    <Amount CurrencyCode="840" Exponent="2">950</Amount>
    <RiskResult>
        <StatusCode>100</StatusCode>
        <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
    </RiskResult>
    <RecommendedClientMessage><! [CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
    </Purchase>
</Event>
</Response>

```

3.3.2.5.13.8 REFUNDED (Recurring transaction)

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchantAccount">
    <Event>
        <Purchase DPGReferenceId="122109" TransactionId="T10_FVBMSTLDAIER">
            <PaymentMethod name="EL">
                <ReferenceId>29759745</ReferenceId>
                <ResponseCode>R03</ResponseCode>
                <ResponseDescription>No account/unable to locate
account</ResponseDescription>
                <Recurring>True</Recurring>
                <OriginalTransactionId>T10_GUSWFMVR2PNW</OriginalTransactionId>
            </PaymentMethod>
            <Status>REFUNDED</Status>
            <Amount CurrencyCode="840" Exponent="2">950</Amount>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your transaction has been
refunded. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
            </Purchase>
        </Event>
    </Response>

```

3.3.2.5.13.9 REFUNDED (Non-recurring transaction)

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchantAccount">
    <Event>
        <Purchase DPGReferenceId="122109" TransactionId="T10_FVBMSTLDAIER">
            <PaymentMethod name="EL">

```

```

<ReferenceId>29759745</ReferenceId>
<ResponseCode>R03</ResponseCode>
<ResponseDescription>No account/unable to locate
account</ResponseDescription>
<Recurring>False</Recurring>
</PaymentMethod>
<Status>REFUNDED</Status>
<Amount CurrencyCode="840" Exponent="2">950</Amount>
<RiskResult>
    <StatusCode>100</StatusCode>
    <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your transaction has been
refunded. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
    </Purchase>
</Event>
</Response>

```

3.3.2.5.13.10 CHARGEBACK (Recurring transaction)

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchantAccount">
    <Event>
        <Purchase DPGReferenceId="122108" TransactionId="T10_FONYYOIBFBRL">
            <PaymentMethod name="EL">
                <ReferenceId>29733229</ReferenceId>
                <ResponseCode>83</ResponseCode>
                <ResponseDescription>Fraudulent Transaction: Card Not Present
Environment</ResponseDescription>
                <Recurring>True</Recurring>
                <OriginalTransactionId>T10_GUSWFMVR2PNW</OriginalTransactionId>
            </PaymentMethod>
            <Status>CHARGED_BACK</Status>
            <Amount CurrencyCode="840" Exponent="2">950</Amount>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your transaction has been
charged back. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>

```

3.3.2.5.13.11 CHARGEBACK (Non-recurring transaction)

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchantAccount">
    <Event>
        <Purchase DPGReferenceId="122108" TransactionId="T10_FONYYOIBFBRL">
            <PaymentMethod name="EL">
                <ReferenceId>29733229</ReferenceId>
                <ResponseCode>83</ResponseCode>
                <ResponseDescription>Fraudulent Transaction: Card Not Present

```

```

    Environment</ResponseDescription>
        <Recurring>False</Recurring>
    </PaymentMethod>
    <Status>CHARGED_BACK</Status>
    <Amount CurrencyCode="840" Exponent="2">950</Amount>
    <RiskResult>
        <StatusCode>100</StatusCode>
        <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
    </RiskResult>
    <RecommendedClientMessage><! [CDATA[Your transaction has been
charged back. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
    </Purchase>
</Event>
</Response>

```

3.3.2.5.14 Amazon return values

3.3.2.5.14.1 PENDING

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="A98E8GW3O4IUZ">
    <Event>
        <Purchase DPGReferenceId="32604" TransactionId="T10_110404083030">
            <PaymentMethod name="AM">
                <ReferenceId>202-2262839-7923559</ReferenceId>
                <ResponseCode>100</ResponseCode>
                <ResponseDescription/>
                <TimeStamp>4/4/2011 8:30:57 AM</TimeStamp>
            </PaymentMethod>
            <Status>PENDING</Status>
            <Amount CurrencyCode="826" Exponent="2">5241</Amount>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[Your request could not entirely
be completed. Please check your account later for an updated status. You
may also contact merchant customer service for more
information.]]></RecommendedClientMessage>
            </Purchase>
        </Event>
    </Response>

```

3.3.2.5.14.2 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="A98E8GW3O4IUZ">
    <Event>
        <Purchase DPGReferenceId="32604" TransactionId="T10_110404083030">
            <PaymentMethod name="AM">
                <ReferenceId>202-2262839-7923559</ReferenceId>
                <ResponseCode>ReadyToShip</ResponseCode>
                <ResponseDescription/>
                <TimeStamp>2009-08-31</TimeStamp>
            </PaymentMethod>
            <Status>AUTHORISED</Status>

```

```

<Amount CurrencyCode="826" Exponent="2">5241</Amount>
<RiskResult>
    <StatusCode>100</StatusCode>
    <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
</RiskResult>
<RecommendedClientMessage><! [CDATA[Your transaction has been
processed successfully.]]></RecommendedClientMessage>
</Purchase>
</Event>
</Response>
```

3.3.2.5.14.3 SENT_FOR_CAPTURE

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="A98E8GW3O4IUZ">
    <Event>
        <Modification DPGReferenceId="32604"
TransactionId="T10_110404083030">
            <PaymentMethod name="AM">
                <ReferenceId>202-2262839-7923559</ReferenceId>
                <ResponseCode>200</ResponseCode>
                <ResponseDescription/>
                <TimeStamp>4/4/2011 8:33:36 AM</TimeStamp>
            </PaymentMethod>
            <Status>SENT_FOR_CAPTURE</Status>
            <Amount CurrencyCode="978" Exponent="2">5240</Amount>
        </Modification>
    </Event>
</Response>
```

3.3.2.5.14.4 SENT_FOR_CANCEL

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="A98E8GW3O4IUZ">
    <Event>
        <Modification DPGReferenceId="32604"
TransactionId="T10_110404083030">
            <PaymentMethod name="AM">
                <ReferenceId>202-2262839-7923559</ReferenceId>
                <ResponseCode>200</ResponseCode>
                <ResponseDescription/>
                <TimeStamp>4/4/2011 8:33:36 AM</TimeStamp>
            </PaymentMethod>
            <Status>SENT_FOR_CANCEL</Status>
            <Amount CurrencyCode="978" Exponent="2">5240</Amount>
        </Modification>
    </Event>
</Response>
```

3.3.2.5.14.5 SENT_FOR_REFUND

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="A98E8GW3O4IUZ">
    <Event>
        <Modification DPGReferenceId="32604"
```

```

<TransactionId="T10_110404083030">
    <PaymentMethod name="AM">
        <ReferenceId>202-2262839-7923559</ReferenceId>
        <ResponseCode>200</ResponseCode>
        <ResponseDescription/>
        <TimeStamp>4/4/2011 8:33:36 AM</TimeStamp>
    </PaymentMethod>
    <Status>SENT_FOR_REFUND</Status>
    <Amount CurrencyCode="978" Exponent="2">5240</Amount>
</Modification>
</Event>
</Response>

```

3.3.2.5.14.6 CAPTURED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="A98E8GW3O4IUZ">
    <Event>
        <Purchase DPGReferenceId="32604" TransactionId="T10_110404083030">
            <PaymentMethod name="AM">
                <ReferenceId>202-2262839-7923559</ReferenceId>
                <ResponseCode>OrderFulfilled</ResponseCode>
                <ResponseDescription/>
                <TimeStamp>4/4/2011 8:35:02 AM</TimeStamp>
                <FeedID>1564308162</FeedID>
            </PaymentMethod>
            <Status>CAPTURED</Status>
            <Amount CurrencyCode="826" Exponent="2">5241</Amount>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
            <RecommendedClientMessage><! [CDATA[]]></RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>

```

3.3.2.5.14.7 CANCELLED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="A98E8GW3O4IUZ">
    <Event>
        <Purchase DPGReferenceId="32604" TransactionId="T10_110404083030">
            <PaymentMethod name="AM">
                <ReferenceId>202-2262839-7923559</ReferenceId>
                <ResponseCode>OrderFulfilled</ResponseCode>
                <ResponseDescription/>
                <TimeStamp>4/4/2011 8:35:02 AM</TimeStamp>
                <FeedID>1564308162</FeedID>
            </PaymentMethod>
            <Status>CANCELLED</Status>
            <Amount CurrencyCode="826" Exponent="2">5241</Amount>
            <RiskResult>
                <StatusCode>100</StatusCode>
                <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
            </RiskResult>
        </Purchase>
    </Event>
</Response>

```

```

        <RecommendedClientMessage><! [CDATA[ ]]></RecommendedClientMessage>
    </Purchase>
</Event>
</Response>

```

3.3.2.5.14.8 REFUNDED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="A98E8GW3O4IUZ">
<Event>
    <Purchase DPGReferenceId="32604" TransactionId="T10_110404083030">
        <PaymentMethod name="AM">
            <ReferenceId>202-2262839-7923559</ReferenceId>
            <ResponseCode>OrderFulfilled</ResponseCode>
            <ResponseDescription/>
            <TimeStamp>4/4/2011 8:35:02 AM</TimeStamp>
            <FeedID>1564308162</FeedID>
        </PaymentMethod>
        <Status>REFUNDED</Status>
        <Amount CurrencyCode="826" Exponent="2">5241</Amount>
        <RiskResult>
            <StatusCode>100</StatusCode>
            <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
        </RiskResult>
        <RecommendedClientMessage><! [CDATA[ ]]></RecommendedClientMessage>
    </Purchase>
</Event>
</Response>

```

3.3.2.5.14.9 ERROR

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
<Event>
    <Purchase DPGReferenceId="32608" TransactionId="T10_110404084800">
        <PaymentMethod name="AM">
            <ReferenceId/>
            <ResponseCode>InvalidParameterValue</ResponseCode>
            <ResponseDescription>Contract with id 1111 does not
exist.</ResponseDescription>
        </PaymentMethod>
        <Status>ERROR</Status>
        <Amount CurrencyCode="826" Exponent="2">5241</Amount>
        <RiskResult>
            <StatusCode>100</StatusCode>
            <StatusDescription><! [CDATA[No Risk]]></StatusDescription>
        </RiskResult>
        <RecommendedClientMessage><! [CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
    </Purchase>
</Event>
</Response>

```

3.3.2.5.15 WebMoney Return Values

3.3.2.5.15.1 Redirect Purchase Response

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="string">
    <Event>
        <Purchase DPGReferenceId="long" TransactionId="string">
            <PaymentMethod name="W3">
                <RecommendedDisplayMessage
isRightToLeft="False">string</RecommendedDisplayMessage>
            </PaymentMethod>
            <Status>string</Status>
            <RedirectURL>string</RedirectURL>
            <PopupBrowser>bool</PopupBrowser>
            <RedirectParameters method="string" contentType="string">
                <RedirectParameter name="LMI_PAYMENT_AMOUNT" value="double"/>
                <RedirectParameter name="LMI_PAYEE_PURSE" value="string"/>
                <RedirectParameter name="LMI_PAYMENT_NO" value="string"/>
                <RedirectParameter name="LMI_PAYMENT_DESC" value="string"/>
                <RedirectParameter name="LMI_SIM_MODE" value="string"/>
            </RedirectParameters>
            <RiskResult>
                <StatusCode>string</StatusCode>
                <StatusDescription>string</StatusDescription>
            </RiskResult>
        </Purchase>
    </Event>
</Response>
```

3.3.2.5.15.2 Successful Purchase Response

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="string">
    <Event>
        <Purchase DPGReferenceId="long" TransactionId="string">
            <PaymentMethod name="W3">
                <ReferenceId>string</ReferenceId>
                <ResponseCode>string</ResponseCode>
                <ResponseDescription>string</ResponseDescription>
                <TimeStamp>string</TimeStamp>
                <LMI_PAYEE_PURSE></LMI_PAYEE_PURSE>
                <LMI_SYS_INVS_NO></LMI_SYS_INVS_NO>
                <LMI_SYS_TRANS_NO></LMI_SYS_TRANS_NO>
                <LMI_PAYER_PURSE></LMI_PAYER_PURSE>
                <LMI_PAYER_WM></LMI_PAYER_WM>
                <LMI_SYS_TRANS_DATE></LMI_SYS_TRANS_DATE>
            </PaymentMethod>
            <Status>string</Status>
            <Amount CurrencyCode="short" Exponent="short">int</Amount>
            <RiskResult>
                <StatusCode>string</StatusCode>
                <StatusDescription>string</StatusDescription>
            </RiskResult>
            <RecommendedClientMessage>string</RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>
```

3.3.2.5.15.3 UnSuccessful Purchase Response

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="string">
    <Event>
        <Purchase DPGReferenceId="long" TransactionId="string">
            <PaymentMethod name="W3">
                <ReferenceId/>
                ; <ResponseCode>string</ResponseCode>
                <ResponseDescription>string</ResponseDescription>
                <LMI_PAYEE_PURSE></LMI_PAYEE_PURSE>
                <LMI_SYS_INVS_NO></LMI_SYS_INVS_NO>
                <LMI_SYS_TRANS_NO></LMI_SYS_TRANS_NO>
                <LMI_PAYER_PURSE></LMI_PAYER_PURSE>
                <LMI_PAYER_WM></LMI_PAYER_WM>
                <LMI_SYS_TRANS_DATE></LMI_SYS_TRANS_DATE>
            </PaymentMethod>
            <Status>string</Status>
            <Error>
                <Code>string</Code>
                <Description>string</Description>
            </Error>
            <Amount CurrencyCode="short" Exponent="short">int</Amount>
            <RiskResult>
                <StatusCode>string</StatusCode>
                <StatusDescription>string</StatusDescription>
            </RiskResult>
            <RecommendedClientMessage>string</RecommendedClientMessage>
        </Purchase>
    </Event>
</Response>
```

3.3.2.5.16 Offline Bank Transfer return values

3.3.2.5.16.1 AUTHORISED

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="000291XX">
    <Event>
        <Purchase DPGReferenceId="123124" TransactionId="T10_110630093650">
            <PaymentMethod name="O3">
                <ReferenceId>68dc5897-fca2-e011-b645-0026b979425c</ReferenceId>
                <ResponseCode>0</ResponseCode>
                <ResponseDescription>Success</ResponseDescription>
                <RequestDate>6/29/2011 9:37:38 AM</RequestDate>
                <FirstName>John</FirstName>
                <LastName>Doe</LastName>
                <Email>john.doe@amnesia.co.za</Email>
                <CustomerReference>435NSDVON252J3</CustomerReference>
                <uniqueReference>DC5EU9351351</uniqueReference>
                <postingDate>6/30/2011 9:43:38 AM</postingDate>
                <externalBankAccountNumber>31083899</externalBankAccountNumber>
                <bankName>Barclays London</bankName>
                <bankCurrency>USD</bankCurrency>
                <bankAmount>100.00</bankAmount>
                <appliedCurrency>978</appliedCurrency>
                <appliedAmount>7.25</appliedAmount>
            </PaymentMethod>
        </Purchase>
    </Event>
</Response>
```

```

<countryCode>string</countryCode>
<bankInformation>US</bankInformation>
<merchantAccountNumber>0000330000000000</merchantAccountNumber>
<envoyBankAccountNumber>0000330000000001</envoyBankAccountNumber>
<statementNumber>385</statementNumber>
<statementId>875576d1-5e99-43d2-98ed-faa28d8a7a33</statementId>
<transferType>BANKIN</transferType>
<refererType>BANKIN</refererType>
</PaymentMethod>
<Status>AUTHORISED</Status>
<Amount CurrencyCode="826" Exponent="2">4932</Amount>
<RecommendedClientMessage>Your transaction has been processed
successfully.</RecommendedClientMessage>
</Purchase>
</Event>
</Response>

```

3.3.2.5.16.2 MARKED FOR REFUND

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="000291XX">
<Event>
<Purchase DPGReferenceId="123124" TransactionId="T10_110630093650">
<PaymentMethod name="O3">
<ReferenceId>68dc5897-fca2-e011-b645-0026b979425c</ReferenceId>
<ResponseCode>210</ResponseCode>
<ResponseDescription>Incorrect amount
deposited</ResponseDescription>
<RequestDate>6/29/2011 9:37:38 AM</RequestDate>
<FirstName>John</FirstName>
<LastName>Doe</LastName>
<Email>john.doe@amnesia.co.za</Email>
<CustomerReference>435NSDVON252J3</CustomerReference>
<uniqueReference>DC5EU9351351</uniqueReference>
<postingDate>6/30/2011 9:43:38 AM</postingDate>
<externalBankAccountNumber>31083899</externalBankAccountNumber>
<bankName>Barclays London</bankName>
<bankCurrency>USD</bankCurrency>
<bankAmount>100.00</bankAmount>
<appliedCurrency>978</appliedCurrency>
<appliedAmount>7.25</appliedAmount>
<countryCode>string</countryCode>
<bankInformation>US</bankInformation>
<merchantAccountNumber>0000330000000000</merchantAccountNumber>
<envoyBankAccountNumber>0000330000000001</envoyBankAccountNumber>
<statementNumber>385</statementNumber>
<statementId>875576d1-5e99-43d2-98ed-faa28d8a7a33</statementId>
<transferType>BANKIN</transferType>
<refererType>BANKIN</refererType>
</PaymentMethod>
<Status>MARKED_FOR_REFUND</Status>
<Amount CurrencyCode="826" Exponent="2">4932</Amount>
<RecommendedClientMessage><![CDATA[Your transaction has been marked
for refund. ]]></RecommendedClientMessage>
</Purchase>
</Event>

```

</Response>

3.3.2.5.16.3 MARKED FOR REFUND (Exception)

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="000291XX">
  <Event>
    <Purchase ExceptionId="123124" TransactionId="T10_110630093650">
      <PaymentMethod name="O3">
        <ReferenceId>68dc5897-fca2-e011-b645-0026b979425c</ReferenceId>
        <ResponseCode>220</ResponseCode>
        <ResponseDescription>Duplicate deposit of correct
amount</ResponseDescription>
        <OriginalTransactionId>T13_3453V7SD9F</OriginalTransactionId>
        <RequestDate>6/29/2011 9:37:38 AM</RequestDate>
        <FirstName>John</FirstName>
        <LastName>Doe</LastName>
        <Email>john.doe@amnesia.co.za</Email>
        <CustomerReference>435NSDVON252J3</CustomerReference>
        <uniqueReference>DC5EU9351351</uniqueReference>
        <postingDate>6/30/2011 9:43:38 AM</postingDate>
        <externalBankAccountNumber>31083899</externalBankAccountNumber>
        <bankName>Barclays London</bankName>
        <bankCurrency>USD</bankCurrency>
        <bankAmount>100.00</bankAmount>
        <appliedCurrency>978</appliedCurrency>
        <appliedAmount>7.25</appliedAmount>
        <countryCode>string</countryCode>
        <bankInformation>US</bankInformation>
        <merchantAccountNumber>0000330000000000</merchantAccountNumber>
        <envoyBankAccountNumber>0000330000000001</envoyBankAccountNumber>
        <statementNumber>385</statementNumber>
        <statementId>875576d1-5e99-43d2-98ed-faa28d8a7a33</statementId>
        <transferType>BANKIN</transferType>
        <refererType>BANKIN</refererType>
      </PaymentMethod>
      <Status>MARKED_FOR_REFUND</Status>
      <Amount CurrencyCode="826" Exponent="2">4932</Amount>
      <RecommendedClientMessage><![CDATA[Your transaction has been marked
for refund. ]]></RecommendedClientMessage>
    </Purchase>
  </Event>
</Response>
```

3.3.2.5.16.4 PENDING

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="000291XX">
  <Event>
    <Purchase DPGReferenceId="123124" TransactionId="T10_110630093650">
      <PaymentMethod name="O3">
        <ReferenceId>123US4532546</ReferenceId>
        <ResponseCode>SUCCESS</ResponseCode>
        <ResponseDescription>Bank Transfer Details
retrieved</ResponseDescription>
        <BankTransferDetails>
```

```
<BankDetails>
    <bank>Werther Bank AG</bank>
</BankDetails>
</BankTransferDetails>
</PaymentMethod>
<Status>PENDING</Status>
<Amount CurrencyCode="short" Exponent="short">int</Amount>
<RecommendedDisplayMessage><![CDATA[<p>
<strong>1.</strong>&
nbsp;You can transfer funds from your personal bank account to the
account shown by using any of your banks fund transfer facilities e.g.
online banking, telephone banking or in person over-the-counter.</p>
<p>
<strong>2.</strong>&
nbsp;<b>Important:</b> Quote your reference number when you do your transfer.
<font color="red">Inclusion of your reference with your deposit is
crucial to ensure your payment is processed timeously and is required in
order for your account balance to be updated at the merchant.</font>
</b>
</p>
<p>
<strong>3.</strong>&
nbsp;Once you have made your transfer, because of the local clearing
cycle, there will be a short delay before the funds are received at the
collecting bank. Usually this is in the order of a few days but depends
on the banking territory concerned.</p>
<p>
<strong>4.</strong>&
nbsp;It is recommended that all receipts and references are retained as
proof of payment for any future queries you might
have.</p>]]></RecommendedDisplayMessage>
    </Purchase>
    </Event>
</Response>
```

3.3.3 Payout

A call to the Payout web method of the APG Web Service is the first step in the lifecycle of a payout transaction.

Validation will occur during the call to the web method - see Appendix 6 for a list of possible validation errors.

The result of this request will determine if the transaction has completed processing.

The Payout web method accepts two parameters: `payoutDetails` of type `PayoutItem`, and `methodDetails` of type `PaymentMethodItem`. The two types are custom complex structures. For more information on the composition of these structures, please refer to the WSDL which can be downloaded from the URLs specified in section 0.

3.3.3.1 Schema Elements for Payout Request

In this section, the fields that can be presented with Payout transaction Requests will be presented along with example XML for those fields.

To submit Processor transaction information for a Payout transaction, one should use the relevant `PaymentMethodItem` within the `Payout` schema element:

- `Payout`
 - `PayoutDetails` – see section 3.3.3.1.1
 - `AccountDetails` – see section 3.3.1.1.1
 - `PersonalDetails` – see section 3.3.2.1.8
 - `BillingDetails` – see section 3.3.2.1.3
 - `AddressDetails` – see section 3.3.2.1.2
 - `AmountDetails` – see section 3.3.1.1.2
 - `MethodDetails` – see section 3.3.3.1.2
 - One of the following:
 - `Click2PayItem` – section 3.3.3.1.40
 - `ClickandBuyItem` – section 3.3.3.1.8
 - `EntroPayItem` – section 3.3.3.1.9
 - `InstaDebitItem` – section 3.3.3.1.6
 - `MoneybookersItem` – section 3.3.3.1.5
 - `NetellerItem` – section 3.3.3.1.3
 - `OneClickItem` – section 3.3.3.1.7
 - `Ukash` – section 3.3.2.1.15

The “Required / Payment Method Specific” column lists the following details:

- *Mandatory* - This means that field cannot be empty.
- *Recommended* - The data is highly recommended. The fields are not mandated by APG since not all processors mandate them. Please note that if these fields are not supplied and the transaction is processed through a processor which does mandate them, the transaction will be rejected.
- *Optional* - The data is not required for processing but should be supplied if available.
- *Not Applicable* - The data is not applicable for this payment method and will be discarded if supplied.

If no data is available for Recommended or Optional fields, the field can simply be left out completely. Under no circumstances is the word 'NULL' to be used as a default value for blank fields; merchants should rather leave the field blank.

NOTE: If the value sent for a field should exceed the maximum length then the transaction may be rejected.

SOAP XML Payout Request document:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Payout xmlns="http://DPG-CPT.datacash.com/">
      <payoutDetails>
        <TransactionId>string</TransactionId>
        <Description>string</Description>
        <TransactionDateTime>string</TransactionDateTime>
        <UserIP>string</UserIP>
        <Language>string</Language>
        <UserAgentHeader>string</UserAgentHeader>
        <CustomerReference>string</CustomerReference>
        <AccountDetails>
          <AccountName>string</AccountName>
          <AccountPIN>string</AccountPIN>
          <CompanyId>string</CompanyId>
        </AccountDetails>
        <PersonalDetails>
          <FirstName>string</FirstName>
          <LastName>string</LastName>
          <Telephone1>string</Telephone1>
          <Telephone1_CountryCode>short</Telephone1_CountryCode>
          <Telephone1_AreaCode>short</Telephone1_AreaCode>
          <DateOfBirth>dateTime</DateOfBirth>
          <Email>string</Email>
          <IDNumber>string</IDNumber>
          <TypeOfID>NONE or PASSPORT or PERSONAL_ID or IDENTITY_CARD or
DRIVERS_LICENSE or TRAVEL_DOCUMENT or RESIDENCE_PERMIT or
IDENTITY_CERTIFICATE_NUMBER or
REGISTRO_FEDERAL_DE_CONTRIBUYENTES or
CREDENCIAL_DE_ELECTOR</TypeOfID>
          <SSN>int</SSN>
          <CPF CNPJ>string</CPF CNPJ>
          <Nationality>string</Nationality>
        </PersonalDetails>
        <BillingDetails>
          <Name>string</Name>
          <AddressDetails>
            <StreetLine1>string</StreetLine1>
            <StreetLine2>string</StreetLine2>
            <City>string</City>
            <PostalCode>string</PostalCode>
            <StateProvince>string</StateProvince>
            <Country>string</Country>

```

```

</AddressDetails>
<AmountDetails>
    <Amount>int</Amount>
    <Exponent>short</Exponent>
    <CurrencyCode>short</CurrencyCode>
</AmountDetails>
</BillingDetails>
</payoutDetails>
<methodDetails/>
</Payout>
</soap:Body>
</soap:Envelope>

```

3.3.3.1.1 PayoutDetails

This is a set of fields and structures required for a transaction.

Element Name	PayoutDetails
Type	PayoutItem
Position	Payout
Children	Yes
Attributes	No

Elements of PayoutDetails			
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details
TransactionId	The TransactionId is a unique identifier generated by the integrating (Merchant) system. Usually this is the integrating system's internal payment order reference number.	String (16); We Recommend this value be be prefixed with the CompanyId plus an underscore e.g. ABC_1234567 (where the CompanyId is a value that will be supplied by DataCash).	Mandatory for all Payment methods
Description	Use this field to supply descriptive text for the transaction.	String (255)	BU Optional
			CL Optional
			EP Optional
			IN Optional
			MB Optional
			NT Optional
			OC Optional

TransactionDateTime	The Merchant's time stamp of the transaction.	String (19); Format "yyyy-mm-dd hh:mm:ss"	BU	Optional
UserAgentHeader	This is a value indicating the browser type and operating system details of the user. This information can be retrieved from the HTTP_USER_AGENT header.	String (255)	CL	Optional
Language	This field contains the language in which the client's user experience will take place. Please refer to Appendix 13 for supported languages. This includes the following Payment method's pages: MB If no language is supplied, the value defaults to "en" (English).	String (20)	IN	Optional
UserIP	Consumer's IP address.	String (26); (Format: 255.255.255.255)	MB	Mandatory
CustomerReference	This is the generic identifier of a customer.	String(70)	NT	Optional
AccountDetails	This is a set of the merchant's authentication details which is required in order to access the APG.	See section 3.3.1.1.1		

PersonalDetails	This is a set of consumer specific details that are required by the Processors.	See section 3.3.2.1.8
BillingDetails	This is a set of the consumer's billing details that are required by the Processors.	See section 3.3.2.1.3
MerchantCustomDetails	Optional structure allowing Merchants to pass custom fields and values.	See section 3.3.2.1.7

Example XML for PayoutDetails complex elements

```

<payoutDetails>
    <TransactionId>string</TransactionId>
    <Description>string</Description>
    <TransactionDateTime>string</TransactionDateTime>
    <UserIP>string</UserIP>
    <UserAgentHeader>string</UserAgentHeader>
    <Language>string</Language>
    <CustomerReference>string</CustomerReference>
    <AccountDetails>
        <AccountName>TestAccount</AccountName>
        <AccountPIN>TestPIN</AccountPIN>
        <CompanyId>A99</CompanyId>
    </AccountDetails>
    <PersonalDetails>
        <FirstName>Jerry</FirstName>
        <LastName>Anderson</LastName>
        <Telephone1>6149135</Telephone1>
        <Telephone1_CountryCode>44</Telephone1_CountryCode>
        <Telephone1_AreaCode>105</Telephone1_AreaCode>
        <DateOfBirth>1982-09-14</DateOfBirth>
        <Email>customer@email.com</Email>
        <IDNumber>123456789</IDNumber>
        <TypeofID>PASSPORT</TypeofID>
        <SSN/>
        <CPF_CNPJ/>
        <Nationality>GB</Nationality>
        <Gender/>
    </PersonalDetails>
    <BillingDetails>
        <Name>Home</Name>
        <AddressDetails>
            <StreetLine1>404 Mulberry Court</StreetLine1>
            <StreetLine2>78 Praed Street</StreetLine2>
        </AddressDetails>
    </BillingDetails>
</payoutDetails>

```

```

<City>London</City>
<PostalCode>W2 1NT</PostalCode>
<StateProvince>EN</StateProvince>
<Country>GB</Country>
</AddressDetails>
<AmountDetails>
    <Amount>725</Amount>
    <Exponent>2</Exponent>
    <CurrencyCode>826</CurrencyCode>
</AmountDetails>
</BillingDetails>
<MerchantCustomDetails>
    <CustomDetail>
        <FieldName>PaymentId</FieldName>
        <FieldValue>1234</FieldValue>
    </CustomDetail>
</MerchantCustomDetails>
</payoutDetails>
</payoutDetails>

```

3.3.3.1.2 methodDetails (PaymentMethodItem)

This is a parent structure from which all the individual specific payout methods are inherited.

Element Name	methodDetails
Type	PaymentMethodItem
Position	Payout
Children	Yes
Attributes	Yes

Elements of methodDetails

Please refer to the following sections for details relevant to each individual payout method:

- BU – ClickandBuy see section 3.3.3.1.8
- CL – Click2Pay see section 3.3.3.1.4
- EP – EntroPay see section 3.3.3.1.9
- IN – InstaDebit see section 3.3.3.1.6
- MB – MoneyBookers see section 3.3.3.1.5
- NT – Neteller see section 3.3.3.1.3
- OC – Envoy OneClick see section 3.3.3.1.7
- UK – Ukash see section 3.3.2.1.15

Attributes of methodDetails

Element Name	Description	Data type / Limitations	Required / Payout Method Specific details
xsi:type	This indicates the individual payout method involved in this transaction. Refer to Appendix 7 for the Payout Method Type options.	Enum	Mandatory for all Payment methods

Example XML for methodDetails complex elements (individual payout method structure involved e.g. Click2Pay)

```
<methodDetails xsi:type="Click2PayItem">
  <AccountNumber>string</AccountNumber>
  <AccountPIN>string</AccountPIN>
</methodDetails>
```

3.3.3.1.3 NetellerItem

These are the fields or structures contained within the NetellerItem structure.

Element Name	NetellerItem
Type	NetellerItem
Position	Payout.methodDetails
Children	Yes
Attributes	No

Elements of NetellerItem

Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AccountNumber	Consumer's NETELLER Account ID	String (50)	NT	Mandatory

Example XML for NetellerItem complex elements

```
<AccountNumber>TestAccount</AccountNumber>
```

3.3.3.1.4 Click2PayItem

These are the fields or structures contained within the Click2PayItem structure.

Element Name	Click2PayItem
Type	Click2PayItem
Position	Payout.methodDetails

Children	Yes
Attributes	No

Elements of Click2PayItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AccountNumber	Consumer's Click2Pay User Name (in the form of an email address).	String (50)	CL	Mandatory
AccountPIN	Consumer's Click2Pay Personal Account Number (PAN) that was assigned, when they registered with Click2Pay.	String (50)	CL	Optional

Example XML for Click2PayDetails complex elements

```
<AccountNumber>test.account@gmail.com</AccountNumber>
<AccountPIN>TestPIN</AccountPIN>
```

3.3.3.1.5 MoneybookersItem

These are the fields or structures contained within the MoneybookersItem structure.

Element Name	MoneybookersItem
Type	MoneybookersItem
Position	Payout.methodDetails
Children	Yes
Attributes	No

Elements of MoneybookersItem				
Element Name	Description	Data type / Limitations	Required / Payout Method Specific details	
AccountNumber	Consumer's MoneyBookers User Name (in the form of an email address).	String (50)	MB	Mandatory

PaymentMethodType	Payment method for Moneybookers. For more information, please refer to Appendix 16.	Enum	MB	Optional
-------------------	---	------	----	----------

Example XML for MoneybookersItem complex elements

```
<AccountNumber>test.account@gmail.com</AccountNumber>
```

3.3.3.1.6 InstaDebitItem

These are the fields or structures contained within the InstaDebitItem structure.

Element Name	InstaDebitItem
Type	InstaDebitItem
Position	Payout.methodDetails
Children	Yes
Attributes	No

Elements of InstaDebitItem

Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AccountNumber	Consumer's InstaDebit User ID.	String (30)	IN	Mandatory

Example XML for InstaDebitItem complex elements

```
<AccountNumber>DEMO+CUSTOMER+PL6CDYYL</AccountNumber>
```

3.3.3.1.7 OneClickItem

These are the fields or structures contained within the OneClickItem structure.

Element Name	OneClickItem
Type	OneClickItem
Position	Payout.methodDetails
Children	Yes
Attributes	No

Elements of OneClickItem

Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AccountNumber	Consumer's OneClick account number (in the form of an email address or account number).	String (50)	OC	Mandatory
ServiceType	This indicates the type of banking service available from Envoy OneClick. Refer to Appendix 17.	Enum	OC	Mandatory
UseSaleCurrency	This is to indicate that the sales currency code and exponent must be used.	Boolean	OC	Optional

Example XML for OneClickItem complex elements

```
<AccountNumber>test.account@gmail.com</AccountNumber>
<ServiceType>EWIRE</ServiceType>
```

3.3.3.1.8 ClickAndBuyItem

These are the fields or structures contained within the ClickAndBuyItem structure.

Element Name	ClickAndBuyItem
Type	ClickAndBuyItem
Position	Payout.methodDetails
Children	Yes
Attributes	No

Elements of ClickAndBuyItem

Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AccountNumber	Consumer's ClickAndBuy account number or account email.	String (50)	BU	Mandatory

Example XML for ClickAndBuyItem complex elements

```
<AccountNumber>123456789</AccountNumber>
```

3.3.3.1.9 EntroPayItem

These are the fields or structures contained within the EntroPayItem structure.

Element Name	EntroPayItem
Type	EntroPayItem
Position	Payout.methodDetails
Children	Yes
Attributes	No

Elements of EntroPayItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
AccountReference	Entropay Account Reference	String (7)	EP	Optional
CardNumber	Entropay CardNumber	String (24)	EP	Optional
Email	Entropay Email	String (255)	EP	Optional
ClaimCode	Entropay ClaimCode	String (15)	EP	Optional
PaymentType	This indicates which security details will be used for this transaction. Refer to Appendix 28.	Enum	EP	Mandatory

EntroPay Note: When submitting an EntroPay (EP) transaction, the below combinations are allowed per the PaymentType for the EntroPayItem.

Field	Combo 1	Combo 2	Combo 3
PaymentType	EMAIL	CARD_NUMBER	ACCOUNT_REFERENCE
AccountReference			X
CardNumber		X	
Email	X		
ClaimCode	X		

Example XML for EntroPayItem complex elements

```
<methodDetails xsi:type="EntroPayItem">
    <AccountReference/>
    <CardNumber/>
    <Email>testcustomer@datacash.co.za</Email>
    <ClaimCode>123456789</ClaimCode>
    <PaymentType>EMAIL</PaymentType>
</methodDetails>
```

3.3.3.2 Examples of XML for Payout Requests

3.3.3.2.1 Neteller input values

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Payout xmlns="http://DPG-CPT.datacash.com/">
            <payoutDetails>
                <TransactionId>A99_080716134142</TransactionId>
                <Description/>
                <TransactionDateTime>2008-07-16 13:41:42</TransactionDateTime>
                <UserIP>10.0.0.1</UserIP>
                <Language>en</Language>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>Caylin</FirstName>
                    <LastName>Williams</LastName>
                    <Telephone1>6149135</Telephone1>
                    <Telephone1_CountryCode>44</Telephone1_CountryCode>
                    <Telephone1_AreaCode>105</Telephone1_AreaCode>
                    <DateOfBirth>1982-09-14T00:00:00</DateOfBirth>
                    <Email>customer@email.com</Email>
                    <TypeOfID>PASSPORT</TypeOfID>
                    <SSN>0</SSN>
                    <Nationality>GB</Nationality>
                </PersonalDetails>
                <BillingDetails>
                    <Name>Home</Name>
                    <AddressDetails>
                        <StreetLine1>78 Praed Street</StreetLine1>
                        <StreetLine2/>
                        <City>London</City>
                        <PostalCode>W2 1NT</PostalCode>
                        <StateProvince>EN</StateProvince>
                        <Country>GB</Country>
                    </AddressDetails>
                    <AmountDetails>
```

```

<Amount>725</Amount>
<Exponent>2</Exponent>
<CurrencyCode>826</CurrencyCode>
</AmountDetails>
</BillingDetails>
<MerchantCustomDetails/>
</payoutDetails>
<methodDetails xsi:type="NetellerItem">
    <AccountNumber>TestMethodAccount</AccountNumber>
</methodDetails>
</Payout>
</soap:Body>
</soap:Envelope>

```

3.3.3.2.2 Click2Pay input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Payout xmlns="http://DPG-CPT.datacash.com/">
            <payoutDetails>
                <TransactionId>A99_080717124604</TransactionId>
                <Description/>
                <TransactionDateTime>2008-07-17 12:46:04</TransactionDateTime>
                <UserIP>10.0.0.1</UserIP>
                <Language>en</Language>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>Jerry</FirstName>
                    <LastName>Anderson</LastName>
                    <Telephone1>6149135</Telephone1>
                    <Telephone1_CountryCode>44</Telephone1_CountryCode>
                    <Telephone1_AreaCode>105</Telephone1_AreaCode>
                    <DateOfBirth>1982-09-14T00:00:00</DateOfBirth>
                    <Email>customer@email.com</Email>
                    <TypeOfID>PASSPORT</TypeOfID>
                    <SSN>0</SSN>
                    <Nationality>GB</Nationality>
                </PersonalDetails>
                <BillingDetails>
                    <Name>Home</Name>
                    <AddressDetails>
                        <StreetLine1>123 Roman Street</StreetLine1>
                        <StreetLine2/>
                        <City>London</City>
                        <PostalCode>W4 1BT</PostalCode>
                        <StateProvince>EN</StateProvince>
                        <Country>GB</Country>
                    </AddressDetails>
                <AmountDetails>

```

```

        <Amount>725</Amount>
        <Exponent>2</Exponent>
        <CurrencyCode>826</CurrencyCode>
    </AmountDetails>
</BillingDetails>
<MerchantCustomDetails/>
</payoutDetails>
<methodDetails xsi:type="Click2PayItem">
    <AccountNumber>test.account@gmail.com</AccountNumber>
    <AccountPIN>TestMethodPIN</AccountPIN>
</methodDetails>
</Payout>
</soap:Body>
</soap:Envelope>

```

3.3.3.2.3 MoneyBookers input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Payout xmlns="http://DPG-CPT.datacash.com/">
            <payoutDetails>
                <TransactionId>A99_080717124013</TransactionId>
                <Description />
                <TransactionDateTime>2008-07-17 12:40:13</TransactionDateTime>
                <UserIP>10.0.0.1</UserIP>
                <Language>nl</Language>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>Mary</FirstName>
                    <LastName>Bakkers</LastName>
                    <Telephone1>4512834</Telephone1>
                    <Telephone1_CountryCode>31</Telephone1_CountryCode>
                    <Telephone1_AreaCode>20</Telephone1_AreaCode>
                    <DateOfBirth>1957-04-03T00:00:00</DateOfBirth>
                    <Email>customer@email.com</Email>
                    <TypeOfID>PASSPORT</TypeOfID>
                    <SSN>0</SSN>
                    <Nationality>NL</Nationality>
                </PersonalDetails>
                <BillingDetails>
                    <Name>Home</Name>
                    <AddressDetails>
                        <StreetLine1>Koningensplein 113</StreetLine1>
                        <StreetLine2 />
                        <City>Amsterdam</City>
                        <PostalCode>1011 TG</PostalCode>
                        <StateProvince>NH</StateProvince>
                        <Country>NL</Country>
                    </AddressDetails>

```

```

<AmountDetails>
    <Amount>150</Amount>
    <Exponent>2</Exponent>
    <CurrencyCode>978</CurrencyCode>
</AmountDetails>
</BillingDetails>
<MerchantCustomDetails/>
</payoutDetails>
<methodDetails xsi:type="MoneybookersItem">
    <AccountNumber>test.account@gmail.com</AccountNumber>
    <PaymentMethodType>NONE</PaymentMethodType>
</methodDetails>
</Payout>
</soap:Body>
</soap:Envelope>

```

3.3.3.2.4 InstaDebit input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Payout xmlns="http://DPG-CPT.datacash.com/">
            <payoutDetails>
                <TransactionId>A99_091014141919</TransactionId>
                <TransactionDateTime>2009-10-14 14:19:19</TransactionDateTime>
                <UserIP>10.0.0.1</UserIP>
                <CustomerReference>2345</CustomerReference>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>Wolfgang</FirstName>
                    <LastName>Middendorp</LastName>
                    <Telephone1>6149135</Telephone1>
                    <Telephone1_CountryCode>49</Telephone1_CountryCode>
                    <Telephone1_AreaCode>10</Telephone1_AreaCode>
                    <Email>customer@email.com</Email>
                    <TypeOfID>NONE</TypeOfID>
                    <SSN/>
                </PersonalDetails>
                <BillingDetails>
                    <Name>Wolfgang Middendorp</Name>
                    <AddressDetails>
                        <StreetLine1>Berliner Straße 213</StreetLine1>
                        <StreetLine2/>
                        <City>Berlin</City>
                        <PostalCode>10825</PostalCode>
                        <Country>DE</Country>
                    </AddressDetails>
                    <AmountDetails>
                        <Amount>5240</Amount>
                        <Exponent>2</Exponent>
                        <CurrencyCode>978</CurrencyCode>

```

```

        </AmountDetails>
        </BillingDetails>
        <Language>de</Language>
        <MerchantCustomDetails/>
    </payoutDetails>
    <methodDetails xsi:type="InstaDebitItem">
        <AccountNumber>DEMO+CUSTOMER+PL6CDYYL</AccountNumber>
    </methodDetails>
</Payout>
</soap:Body>
</soap:Envelope>

```

3.3.3.2.5 Envoy OneClick input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Payout xmlns="http://DPG-CPT.datacash.com/">
            <payoutDetails>
                <TransactionId>T99_123456789012</TransactionId>
                <Description/>
                <TransactionDateTime>2010-10-01 01:02:03</TransactionDateTime>
                <UserIP>10.0.0.1</UserIP>
                <CustomerReference>test.user</CustomerReference>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPin</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>DEMO</FirstName>
                    <LastName>CUSTOMER</LastName>
                    <Telephone1_CountryCode>0</Telephone1_CountryCode>
                    <Telephone1_AreaCode>0</Telephone1_AreaCode>
                    <Telephone2_CountryCode>0</Telephone2_CountryCode>
                    <Telephone2_AreaCode>0</Telephone2_AreaCode>
                    <DateOfBirth>0001-01-01T00:00:00</DateOfBirth>
                    <Email>consumer@email.com</Email>
                    <TypeOfID>NONE</TypeOfID>
                    <SSN>0</SSN>
                </PersonalDetails>
                <BillingDetails>
                    <Name>Demo Customer</Name>
                    <AddressDetails>
                        <StreetLine1>Descartes House</StreetLine1>
                        <StreetLine2>Centre-du- Québec-Sud</StreetLine2>
                        <City>Quebec City</City>
                        <PostalCode>HJ6JVG</PostalCode>
                        <StateProvince>QC</StateProvince>
                        <Country>DK</Country>
                    </AddressDetails>
                    <AmountDetails>
                        <Amount>1000</Amount>
                        <Exponent>2</Exponent>
                        <CurrencyCode>208</CurrencyCode>
                    </AmountDetails>
                </BillingDetails>
            </payoutDetails>
        </Payout>
    </soap:Body>
</soap:Envelope>

```

```

        </AmountDetails>
        <BillingDetails>
        <Language>EN</Language>
        <MerchantCustomDetails/>
    </payoutDetails>
    <methodDetails xsi:type="OneClickItem">
        <ServiceType>EWIRE</ServiceType>
        <AccountNumber>consumer@email.com</AccountNumber>
    </methodDetails>
</Payout>
</soap:Body>
</soap:Envelope>

```

3.3.3.2.6 ClickAndBuy input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Payout xmlns="http://DPG-CPT.datacash.com/">
            <payoutDetails>
                <TransactionId>A99_123456789012</TransactionId>
                <Description/>
                <TransactionDateTime>2010-10-01 01:02:03</TransactionDateTime>
                <UserIP>10.0.0.1</UserIP>
                <CustomerReference>test.user</CustomerReference>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPin</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>Bridgette</FirstName>
                    <LastName>Robins</LastName>
                    <Telephone1>65012781</Telephone1>
                    <Telephone1_CountryCode>44</Telephone1_CountryCode>
                    <Telephone1_AreaCode>20</Telephone1_AreaCode>
                    <Telephone2/>
                    <Telephone2_CountryCode>0</Telephone2_CountryCode>
                    <Telephone2_AreaCode>0</Telephone2_AreaCode>
                    <DateOfBirth>1975-11-23T00:00:00</DateOfBirth>
                    <Email>consumer@email.com</Email>
                    <IDNumber/>
                    <TypeOfID>NONE</TypeOfID>
                    <SSN>0</SSN>
                    <Nationality>GB</Nationality>
                </PersonalDetails>
                <BillingDetails>
                    <Name>Home</Name>
                    <StatementNarrative/>
                    <AddressDetails>
                        <StreetLine1>51 Rocky Road</StreetLine1>
                        <StreetLine2/>
                        <City>London</City>
                        <PostalCode>SW1 1HM</PostalCode>
                        <StateProvince>EN</StateProvince>
                    </AddressDetails>
                </BillingDetails>
            </payoutDetails>
        </Payout>
    </soap:Body>
</soap:Envelope>

```

```

<Country>GB</Country>
</AddressDetails>
<AmountDetails>
    <Amount>1</Amount>
    <Exponent>2</Exponent>
    <CurrencyCode>124</CurrencyCode>
</AmountDetails>
</BillingDetails>
<Language>en</Language>
<MerchantCustomDetails/>
</payoutDetails>
<methodDetails xsi:type="ClickAndBuyItem">
    <AccountNumber>100000000</AccountNumber>
</methodDetails>
</Payout>
</soap:Body>
</soap:Envelope>

```

3.3.3.2.7 EntroPay input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Payout xmlns="http://DPG-CPT.datacash.com/">
            <payoutDetails>
                <TransactionId>A99_091014141919</TransactionId>
                <TransactionDateTime>2009-10-14 14:19:19</TransactionDateTime>
                <UserIP>10.0.0.1</UserIP>
                <CustomerReference>2345</CustomerReference>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId>A99</CompanyId>
                </AccountDetails>
                <PersonalDetails>
                    <FirstName>Wolfgang</FirstName>
                    <LastName>Middendorp</LastName>
                    <Telephone1>6149135</Telephone1>
                    <Telephone1_CountryCode>49</Telephone1_CountryCode>
                    <Telephone1_AreaCode>10</Telephone1_AreaCode>
                    <Email>customer@email.com</Email>
                    <TypeOfID>NONE</TypeOfID>
                    <SSN/>
                </PersonalDetails>
                <BillingDetails>
                    <Name>Wolfgang Middendorp</Name>
                    <AddressDetails>
                        <StreetLine1>Berliner Straße 213</StreetLine1>
                        <StreetLine2/>
                        <City>Berlin</City>
                        <PostalCode>10825</PostalCode>
                        <Country>DE</Country>
                    </AddressDetails>
                    <AmountDetails>
                        <Amount>5240</Amount>

```

```

        <Exponent>2</Exponent>
        <CurrencyCode>978</CurrencyCode>
    </AmountDetails>
</BillingDetails>
<Language>en</Language>
<MerchantCustomDetails/>
</payoutDetails>
<methodDetails xsi:type="EntroPayItem">
    <AccountReference/>
    <CardNumber/>
    <Email>testcustomer@datacash.co.za</Email>
    <ClaimCode>123456789</ClaimCode>
    <PaymentType>EMAIL</PaymentType>
</methodDetails>
</Payout>
</soap:Body>
</soap:Envelope>

```

3.3.3.2.8 Ukash input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Purchase xmlns="http://DPG-CPT.datacash.com/">
            <payoutDetails>
                <TransactionId>A99_080708134612</TransactionId>
                <TransactionDateTime>2008-07-08 13:46:12</TransactionDateTime>
                <CustomerReference>Test</CustomerReference>
                <AccountDetails>
                    <AccountName>TestAccount</AccountName>
                    <AccountPIN>TestPIN</AccountPIN>
                    <CompanyId> A99</CompanyId>
                </AccountDetails>
                <PersonalDetails/>
                <BillingDetails>
                    <AddressDetails>
                        <StateProvince>EN</StateProvince>
                        <Country>GB</Country>
                    </AddressDetails>
                    <AmountDetails>
                        <Amount>1000</Amount>
                        <Exponent>2</Exponent>
                        <CurrencyCode>826</CurrencyCode>
                    </AmountDetails>
                </BillingDetails>
                <Language>EN</Language>
            </payoutDetails>
            <methodDetails xsi:type="UKashItem">
                <AccountNumber>633718009</AccountNumber>
                <AccountPIN/>
                <AccountValue>1000</AccountValue>
                <TypeOfRedemption>ISSUE_VOUCHER</TypeOfRedemption>
                <AmountReference/>
            </methodDetails>
        </Purchase>
    </soap:Body>
</soap:Envelope>

```

```
</soap:Body>
</soap:Envelope>
```

3.3.3.3 Schema elements for Payout Response

In this section, the fields that can be presented with Payout transaction Responses will be presented along with example XML for those fields.

The top-level XML element is the `Response` element. This will be followed by the `Event` element, with the Payout method's response details inside that, in the Payout complex structure.

Note that various responses will result in various XML formats being returned - the following list contains some of the response fields, but you should refer to the XML samples below and in Section 0 for further information.

- `Response` – see section 3.3.1.2.3
 - `Event` – see section 3.3.1.2.2
 - `Payout` – see section 3.3.3.3.2
 - `PaymentMethod` – see section 3.3.3.3.1
 - `PaymentMethod` – *specific fields*
 - `Error` – see section 3.3.1.2.1

3.3.3.3.1 PaymentMethod

The details involved in the specific Payment Method of this transaction are displayed here. The top-level element "PaymentMethod" will contain the payment method for the transaction, while the sub elements will contain the rest of the details.

Element Name:	PaymentMethod
Position	Response.Event.Payout
Children	Yes
Attributes	Yes

Attributes involved in the PaymentMethod element			
Attribute Name	Located in which Element	Description	Data type / Limitations
Please refer to section 3.3.2.3.3			

Elements of PaymentMethod		
Element Name	Description	Data type / Limitations
Please refer to section 3.3.2.3.3		

Example of XML for PaymentMethod complex elements (*The example is specifically for a successfully completed Neteller transaction*)

```
<PaymentMethod name="NT">
  <ReferenceId>1308228</ReferenceId>
  <ResponseCode>yes</ResponseCode>
  <ResponseDescription>yes</ResponseDescription>
  <FirstName>Test</FirstName>
  <LastName>Test</LastName>
  <Fee>0.20</Fee>
  <FxRate>0.6391</FxRate>
</PaymentMethod>
```

3.3.3.3.2 Payout

The results of a Payout transaction will be returned in this complex structure. Note that the details depend on the Payment Method involved – this is a very dynamic complex structure that can take on a variety of shapes, depending on the scenario.

Element Name:	Payout
Position	Response.Event
Children	Yes
Attributes	Yes

Attributes involved in the Payout element			
Attribute Name	Located in which Element	Description	Data type / Limitations
DPGReferenceId	Payout	The DPGReferenceId is a unique APG transaction identifier which is generated by the APG for each merchant transaction.	Long
TransactionId	Payout	This will return the original TransactionId that the Merchant sent through. This is for informational purposes only.	String (16)
CurrencyCode	Amount	This is the transaction currency. Please refer to Appendix 10.	Short
Exponent	Amount	This is the transaction currency exponent. Please refer to Appendix 10.	Short

Elements of Payout		
Element Name	Description	Data type / Limitations

Payout	Top-level XML element indicating a Payout transaction - the DPGReferenceId and TransactionId for the transaction are also returned as attributes of this field – refer to the Attributes table above for more information.)	
PaymentMethod	See section 3.3.2.3.3	
Status	This is the response, which indicates if the transaction was accepted, or not at the processor. See Appendix 6 for possible response messages.	String (20)
Error	See section 3.3.1.2.1. This element will appear in the Payout XML if the Status = "ERROR".	
Amount	This field contains the amount payed to the client's account at the financial institution. The field type is a numeric value in cents. (The currency and exponent for the amount are also returned as attributes of this field – refer to the Attributes table above for more information.) Currently available for the following Payout Methods: <ul style="list-style-type: none">• BU• CL• EP• IN• MB• NT• OC• UK	Int

	<p>This field contains the recommended client message that can be displayed to the consumer. A list of these messages will be detailed in the Product Document. In certain instances, the exact nature of the error will be hidden from the consumer e.g. error relating to merchant account details. The true nature of the error can always be found in the ResponseDescription or Error Description response fields.</p> <p>Note: The RecommendedClientMessage field might contain end-of-line characters. Merchants should design their front-end systems in such a way that the end-of-line characters are correctly displayed to end-users.</p> <p>For example, if an HTML page is used as the display method in the front-end system, the merchant must replace the end-of-line characters in the RecommendedClientMessage field with the appropriate
 tags, otherwise the HTML front-end will not display the appropriate line breaks correctly.</p> <p>Alternatively, the contents of the RecommendedClientMessage can be included in <PRE> tags.</p> <p>The RecommendedClientMessage will be returned in the language which was passed through in Language field of original transaction. This is dependent on whether the language is supported by APG. If the language is not supported, the message will default to English.</p>	
RecommendedClientMessage		String (1000)

Examples of XML for Payout complex elements

Please refer to section 0 for examples per Payment Method.

3.3.3.4 Interpreting the Payout Responses

There are various types of possible responses that can be returned from the APG to the Merchant. These include:

- Successful transaction
- Unsuccessful transaction
- Event notification

Refer to section 3.3.2.5 for examples.

The APG can return multiple responses per transaction, and as a result the merchant needs to analyze these and perform the appropriate action. Appendix 6 details the various responses. The Status field will indicate the status of the transaction. The merchant should only credit the consumer's account when the Status=AUTHORISED, in all other cases the merchant should not credit the consumer's account.

3.3.3.5 Examples of XML for Payout Responses

3.3.3.5.1 Neteller return values

3.3.3.5.1.1 AUTHORISED

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Payout DPGReferenceId="1760" TransactionId="A99_080716134142">
      <PaymentMethod name="NT">
        <ReferenceId>733389</ReferenceId>
        <ResponseCode>yes</ResponseCode>
        <ResponseDescription>yes</ResponseDescription>
        <FirstName>Test</FirstName>
        <LastName>Test</LastName>
        <Fee>0.00</Fee>
        <FxRate>0.4870</FxRate>
      </PaymentMethod>
      <Status>AUTHORISED</Status>
      <Amount CurrencyCode="840" Exponent="2">1488</Amount>
      <RecommendedClientMessage><![CDATA[Your transaction has been
      processed successfully.]]></RecommendedClientMessage>
    </Payout>
  </Event>
</Response>
```

3.3.3.5.1.2 ERROR

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Payout DPGReferenceId="112670" TransactionId="A99_090403103728">
      <PaymentMethod name="NT">
        <ReferenceId/>
        <ResponseCode>no</ResponseCode>
        <ResponseDescription>no</ResponseDescription>
      </PaymentMethod>
      <Status>ERROR</Status>
    </Event>
</Response>
```

```

<Error>
  <Code>1007</Code>
  <Description>The Account ID or Secure ID you entered is not
    valid. Please enter your correct account
    information.</Description>
</Error>
<Amount CurrencyCode="826" Exponent="2">725</Amount>
<RecommendedClientMessage><![CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
</Payout>
</Event>
</Response>

```

3.3.3.5.2 Click2Pay return values

3.3.3.5.2.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Payout DPGReferenceId="91683" TransactionId="A99_080717124604">
      <PaymentMethod name="CL">
        <ReferenceId>30a0a9b50a01001169f94884ff1efa44</ReferenceId>
        <ResponseCode>000</ResponseCode>
        <ResponseDescription>Transaction is
          Acknowledged</ResponseDescription>
        <PaymentType>FUNDS</PaymentType>
      </PaymentMethod>
      <Status>AUTHORISED</Status>
      <Amount CurrencyCode="978" Exponent="2">1067</Amount>
      <RecommendedClientMessage><![CDATA[Your transaction has been
processed successfully.]]></RecommendedClientMessage>
    </Payout>
  </Event>
</Response>

```

3.3.3.5.2.2 REFUSED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Payout DPGReferenceId="112623" TransactionId="A99_090331144718">
      <PaymentMethod name="CL">
        <ReferenceId/>
        <ResponseCode>200</ResponseCode>
        <ResponseDescription>You have provided invalid CLICK2PAY user
          data or the account is not active. Please check and try again.
          If the problem persists, please contact our helpdesk at
          service@click2pay.com.</ResponseDescription>
      </PaymentMethod>
      <Status>REFUSED</Status>
      <Amount CurrencyCode="826" Exponent="2">725</Amount>
      <RecommendedClientMessage><![CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
    </Payout>
  </Event>
</Response>

```

```

</Event>
</Response>
```

3.3.3.5.2.3 ERROR

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Payout DPGReferenceId="112624" TransactionId="A99_090331144826">
            <PaymentMethod name="CL">
                <ReferenceId/>
                <ResponseCode>101</ResponseCode>
                <ResponseDescription>The supplied CLICK2PAY UserName is wrong.  
Please check and try again. If the problem should still exist,  
please contact our customer service at  
service@click2pay.com.</ResponseDescription>
            </PaymentMethod>
            <Status>ERROR</Status>
            <Error>
                <Code/>
                <Description/>
            </Error>
            <Amount CurrencyCode="826" Exponent="2">725</Amount>
            <RecommendedClientMessage><![CDATA[Your request could not be  
completed. Please contact merchant customer service for more  
information.]]></RecommendedClientMessage>
        </Payout>
    </Event>
</Response>
```

3.3.3.5.3 MoneyBookers return values

3.3.3.5.3.1 AUTHORISE

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Payout DPGReferenceId="91682" TransactionId="A99_080717124013">
            <PaymentMethod name="MB">
                <ReferenceId>87712395</ReferenceId>
                <ResponseCode>2</ResponseCode>
                <ResponseDescription>processed</ResponseDescription>
            </PaymentMethod>
            <Status>AUTHORISED</Status>
            <Amount CurrencyCode="710" Exponent="2">1818</Amount>
            <RecommendedClientMessage><![CDATA[Uw transactie is  
geslaagd.]]></RecommendedClientMessage>
        </Payout>
    </Event>
</Response>
```

3.3.3.5.3.2 PENDING

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
```

```

<Payout DPGReferenceId="112629" TransactionId="A99_090331152315">
  <PaymentMethod name="MB">
    <ReferenceId>56235634</ReferenceId>
    <ResponseCode>1</ResponseCode>
    <ResponseDescription>scheduled</ResponseDescription>
  </PaymentMethod>
  <Status>PENDING</Status>
  <Amount CurrencyCode="710" Exponent="2">764</Amount>
  <RecommendedClientMessage><![CDATA[Your request could not entirely
be completed. Please check your account later for an updated
status. You may also contact merchant customer service for more
information.]]></RecommendedClientMessage>
</Payout>
</Event>
</Response>

```

3.3.3.5.3.3 ERROR

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Payout DPGReferenceId="112629" TransactionId="A99_090331152315">
      <PaymentMethod name="MB">
        <ReferenceId/>
        <ResponseCode/>
        <ResponseDescription/>
      </PaymentMethod>
      <Status>ERROR</Status>
      <Error>
        <Code>800</Code>
        <Description>AccountNumber must be supplied</Description>
      </Error>
      <Amount CurrencyCode="124" Exponent="2">100</Amount>
      <RecommendedClientMessage><![CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
    </Payout>
  </Event>
</Response>

```

3.3.3.5.4 InstaDebit return values

3.3.3.5.4.1 AUTHORISED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Payout DPGReferenceId="45253" TransactionId="A99_45723456249">
      <PaymentMethod name="IN">
        <ReferenceId>56235634</ReferenceId>
        <ResponseCode>S</ResponseCode>
        <ResponseDescription>Success</ResponseDescription>
        <txn_fee>0.00</txn_fee>
      </PaymentMethod>
      <Status>AUTHORISED</Status>
      <Amount CurrencyCode="124" Exponent="2">100</Amount>
      <RecommendedClientMessage><![CDATA[Your transaction has been

```

```

        processed successfully..]]></RecommendedClientMessage>
    </Payout>
</Event>
</Response>
```

3.3.3.5.4.2 ERROR

```

<?xml version="1.0" encoding="utf-8"?>

<Response version="1.0" AccountName="MerchAccount">
<Event>
<Payout DPGReferenceId="4534" TransactionId="A99_345723494743">
<PaymentMethod name="IN">
<ReferenceId/>
<ResponseCode/>
<ResponseDescription/>
</PaymentMethod>
<Status>ERROR</Status>
<Error>
<Code>812</Code>
<Description>The InstaDebit notification is
invalid</Description>
```

3.3.3.5.4.3 REFUSED

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="UkashTestMerchant">
<Event>
<Payout DPGReferenceId="136921" TransactionId="T10_110906134303">
<PaymentMethod name="UK">
<ReferenceId>SV_0070_201109061143</ReferenceId>
<ResponseCode>1</ResponseCode>
<ResponseDescription>Declined</ResponseDescription>
<IssuedAmount/>
<IssuedVoucherCurr/>
<IssuedVoucherNumber/>
<IssuedExpiryDate/>
<currencyConversion/>
</PaymentMethod>
<Status>REFUSED</Status>
<Amount CurrencyCode="826" Exponent="2">1000</Amount>
<RecommendedClientMessage><![CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
</Payout>
</Event>
</Response>
```

3.3.4 Transaction Querying

The TransactionQuery call to the APG Web Service is used to determine the current status of a transaction which was previously submitted. The result from the request will determine if the transaction has completed processing or not.

The APG exposes two web methods that allows the querying of transactions.

1. TransactionQuery – Allows transaction querying by “DPGReferenceId”.
2. TransactionQueryByTransactionId – Allows transaction querying by “TransactionId”, “TransactionType”, “PaymentMethod”

3.3.4.1 TransactionQuery

The TransactionQuery web method accepts two parameters: accountDetails of type AccountItem, and “DPGReferenceId” of type Long. The AccountItem type is a custom complex structure. For more information on the composition of this structure, please refer to the WSDL which can be downloaded from the URLs specified in section 3.2.1.

3.3.4.1.1 Schema Elements for TransactionQuery Request

In this section, the fields that can be presented with TransactionQuery transaction Requests will be presented along with example XML for those fields.

- TransactionQuery
 - Account Details – see section 3.3.1.1.1
 - DPGReferenceID

SOAP XML TransactionQuery Request document:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <TransactionQuery xmlns="http://DPG-CPT.datacash.com/">
      <accountDetails>
        <AccountName>string</AccountName>
        <AccountPIN>string</AccountPIN>
        <CompanyId>string</CompanyId>
      </accountDetails>
      <DPGReferenceId>long</DPGReferenceId>
    </TransactionQuery>
  </soap:Body>
</soap:Envelope>
```

3.3.4.1.2 TransactionQuery Structure

Element Name	TransactionQuery
Type	TransactionQuery
Position	Soap:Body
Children	Yes

Attributes	Yes
------------	-----

Elements of TransactionQuery		
Element Name	Description	Data type / Limitations
accountDetails	This is a set of the merchant's authentication details which is required in order to access the APG.	AccountItem – see section 3.3.1.1.1
DPGReferenceID	The DPGReferenceId is a unique APG transaction identifier which is generated by the APG for each merchant transaction. This identifier is returned in the response to a Purchase or Payout transaction.	Long This value is mandatory at all times.

See the next section for an XML example.

3.3.4.1.3 Example of XML for TransactionQuery Requests

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
    <soap12:Body>
        <TransactionQuery xmlns="http://DPG-CPT.datacash.com/">
            <accountDetails>
                <AccountName>TestAccount</AccountName>
                <AccountPIN>TestPIN</AccountPIN>
                <CompanyId>A99</CompanyId>
            </accountDetails>
            <DPGReferenceId>12345</DPGReferenceId>
        </TransactionQuery>
    </soap12:Body>
</soap12:Envelope>
```

3.3.4.2 TransactionQueryByTransactionId

The TransactionQueryByTransactionId web method accepts four parameters: accountDetails of type AccountItem, "TransactionId" of type string, "TransactionType" of type enum, and "PaymentMethod" of type string. The AccountItem type is a custom complex structure. For more information on the composition of this structure, please refer to the WSDL which can be downloaded from the URLs specified in section 3.2.1

3.3.4.2.1 Schema Elements for TransactionQueryByTransactionId Request

In this section, the fields that can be presented with TransactionQueryByTransactionId transaction Requests will be presented along with example XML for those fields.

- TransactionQueryByTransactionId
 - Account Details – see section 3.3.1.1.1
 - TransactionId
 - PaymentMethod
 - TransactionType

SOAP XML TransactionQueryByTransactionId Request document:

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <TransactionQueryByTransactionId xmlns="http://DPG-
CPT.datacash.com/">
      <accountDetails>
        <AccountName>TestAccount</AccountName>
        <AccountPIN>TestPIN</AccountPIN>
        <CompanyId>A99</CompanyId>
      </accountDetails>
      <TransactionId>T10_1236547890</TransactionId>
      <TransactionType>PURCHASE</TransactionType>
      <PaymentMethod>MB</PaymentMethod>
    </TransactionQueryByTransactionId>
  </soap12:Body>
</soap12:Envelope>
```

3.3.4.2.2 TransactionQueryByTransactionId Structure

Element Name	TransactionQueryByTransactionId
Type	TransactionQueryByTransactionId
Position	Soap:Body
Children	Yes
Attributes	Yes

Elements of TransactionQuery		
Element Name	Description	Data type / Limitations
accountDetails	This is a set of the merchant's authentication details which is required in order to access the APG.	AccountItem – see section 3.3.1.1.1 Mandatory

TransactionID	The TransactionId is a unique identifier generated by the integrating (Merchant) system. Usually this is the integrating system's internal payment order reference number.	String (16) Mandatory
PaymentMethod	The payment method of the transaction.	String (3) Mandatory
TransactionType	This is a indicator as to what they of transaction should be searched for. Possible Values: <ul style="list-style-type: none">• PURCHASE• PAYOUT	String Enum Mandatory

See the next section for an XML example.

3.3.4.2.3 Example of XML for TransactionQuery Request

```

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
    <soap12:Body>
        <TransactionQueryByTransactionId xmlns="http://DPG-
CPT.datacash.com/">
            <accountDetails>
                <AccountName>TestAccount</AccountName>
                <AccountPIN>TestPIN</AccountPIN>
                <CompanyId>A99</CompanyId>
            </accountDetails>
            <TransactionId>T10_1236547890</TransactionId>
            <TransactionType>PURCHASE</TransactionType>
            <PaymentMethod>MB</PaymentMethod>
        </TransactionQueryByTransactionId>
    </soap12:Body>
</soap12:Envelope>

```

3.3.4.3 Schema Elements and Example XML for TransactionQuery and TransactionQueryByTransactionId Response

The Response that is generated for a TransactionQuery and TransactionQueryByTransactionId Request is similar to the Response that is generated for a Purchase or Payout transaction.

See section 3.3.2.4 for details of the Purchase (and TransactionQuery) Response structure.

See section 3.3.2.5 for examples of these Responses.

See section 3.3.3.2 for details of the Payout (and TransactionQuery) Response structure.

See section 3.3.3.5 for examples of these Responses.

The following is a list of possible results that can be returned by the TransactionQuery or TransactionQueryByTransactionId web method's Status fields:

- AUTHORISED
- CANCELLED
- PENDING
- UNKNOWN
- ERROR
- NOTFOUND
- FRAUD_REJECTED
- REFUNDED
- REFUSED
- SENT_FOR_CAPTURE
- SENT_FOR_REFUND
- CAPTURED
- CHARGED_BACK

3.3.5 *RetrieveSettlementValue*

The *RetrieveSettlementValue* call to the APG Web Service is used to retrieve the settlement amount when the currency of the merchant account and the voucher/user account is different. This service is only available for the following payment method:

- UK

The *RetrieveSettlementValue* web method accepts six parameters: "TransactionId" of type string, "PaymentMethod" of type string, "AccountNumber" of type string, "Language" of type string, AccountDetails of type AccountItem and AmountDetails of type AmountItem. The AccountItem and AmountItem types are custom complex structures. For more information on the composition of this structure, please refer to the WSDL which can be downloaded from the URLs specified in section 0.

3.3.5.1 Schema Elements for *RetrieveSettlementValue* Request

In this section, the fields that can be presented with *RetrieveSettlementValue* transaction Requests will be presented along with example XML for those fields.

- *RetrieveSettlementValue*
 - TransactionID
 - PaymentMethod
 - AccountNumber
 - Language
 - AccountDetails – section 3.3.1.1.1
 - AmountDetails – section 3.3.1.1.2

SOAP XML *RetrieveSettlementValue* Request document:

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <RetrieveSettlementValue xmlns="http://DPG-CPT.datacash.com/">
      <TransactionId>string</TransactionId>
      <PaymentMethod>string</PaymentMethod>
      <AccountNumber>string</AccountNumber>
      <Language>string</Language>
      <AccountDetails>
        <AccountName>string</AccountName>
        <AccountPIN>string</AccountPIN>
        <CompanyId>string</CompanyId>
      </AccountDetails>
      <AmountDetails>
        <Amount>int</Amount>
        <Exponent>short</Exponent>
        <CurrencyCode>short</CurrencyCode>
      </AmountDetails>
    </RetrieveSettlementValue>
  </soap12:Body>
</soap12:Envelope>
```

3.3.5.1.1 RetrieveSettlementValue

Element Name	RetrieveSettlementValue
Type	RetrieveSettlementValue
Position	Soap:Body
Children	Yes
Attributes	Yes

Elements of RetrieveSettlementValue			
Element Name	Description	Data type / Limitations	Required
TransactionID	The TransactionId is a unique identifier generated by the integrating (Merchant) system. Usually this is the integrating system's internal payment order reference number.	String (16) This value must be prefixed with the CompanyId plus an underscore e.g. ABC_1234567 (where the CompanyId is a value that will be supplied by DataCash).	Mandatory
PaymentMethod	The payment method of the transaction.	String (3)	Mandatory
AccountNumber	Digits 7-9 of the consumer's Voucher or Account number.	String (3)	Mandatory
Language	This field contains the language in which the client's user experience will take place.	String (20)	Mandatory
AccountDetails	This is a set of the merchant's authentication details which is required in order to access the APG.	AccountItem – see section 3.3.1.1.1	Mandatory
AmountDetails	This is a set of amount details.	AmountItem – see section 3.3.1.1.2.	Mandatory

3.3.5.2 Example of XML for RetrieveSettlementValue Requests

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <RetrieveSettlementValue xmlns="http://DPG-CPT.datacash.com/">
            <TransactionId>A99_090912112233</TransactionId>
            <PaymentMethod>UK</PaymentMethod>
            <AccountNumber>AccountNum</AccountNumber>
            <Language>EN</Language>
            <AccountDetails>
                <AccountName>TestAccount</AccountName>
                <AccountPIN>TestPIN</AccountPIN>
                <CompanyId>A99</CompanyId>
            </AccountDetails>
            <AmountDetails>
                <Amount>1000</Amount>
                <Exponent>2</Exponent>
                <CurrencyCode>840</CurrencyCode>
            </AmountDetails>
        </RetrieveSettlementValue>
    </soap:Body>
</soap:Envelope>

```

3.3.5.3 Schema Elements for RetrieveSettlementValue Response

The following is a list of possible results that can be returned by the `RetrieveSettlementValue` web method in the `Status` field:

- INFORMATION_SUPPLIED
- ERROR

3.3.5.3.1 Settlement

The results of a `RetrieveSettlementValue` transaction will be returned in this complex structure.

Element Name:	Settlement
Position	Response.Event
Children	Yes
Attributes	Yes

Attributes involved in the Settlement element			
Attribute Name	Located in which Element	Description	Data type / Limitations
TransactionId	Settlement	This will return the original TransactionId that the Merchant sent through. This is for informational purposes only.	String (16)
CurrencyCode	Amount	This is the transaction currency. Please refer to Appendix 10.	Short

Exponent	Amount	This is the transaction currency exponent. Please refer to Appendix 10.	Short
name	PaymentMethod	This string value will be equal to "UK"	

Elements of Settlement		
Element Name	Description	Data type / Limitations
Settlement	Top-level XML element indicating a RetrieveSettlementValue transaction. The TransactionId for the transaction is also returned as an attribute of this field – refer to the Attributes table above for more information.)	
PaymentMethod	See section 3.3.3.3.1. The name for the payment method is also returned as an attribute of this field – refer to the Attributes table above for more information.) Note: This complex structure will contain the AmountReference field for this RetrieveSettlementValue transaction. This is a value returned by the Ukash processor to uniquely identify the settlement amount transaction.	
Status	This is the response to the RetrieveSettlementValue transaction.	String (20) These are the possible values: <ul style="list-style-type: none">• INFORMATION_SUPPLIED• ERROR
Error	See section 3.3.1.2.1. This element will appear in the Settlement XML if the Status = "ERROR".	
Amount	This field contains the amount taken from the client's account at the financial institution. The field type is a numeric value in cents. (The currency and exponent for the amount are also returned as attributes of this field – refer to the Attributes table above for more information.)	Int
RecommendedClientMessage	See section 3.3.2.3.4 for more information.	

Example of XML for Settlement complex elements

```
<Settlement TransactionId="A99_090912112233">
  <PaymentMethod name="UK">
    <ResponseCode>0</ResponseCode>
    <ResponseDescription>Settlement Amount Retrieved</ResponseDescription>
    <amountReference>481236</amountReference>
    <voucherCurrency>826</voucherCurrency>
  </PaymentMethod>
  <Status>INFORMATION_SUPPLIED</Status>
  <Amount CurrencyCode="840" Exponent="2">1339</Amount>
  <RecommendedClientMessage><![CDATA[Your transaction has been processed successfully.]]></RecommendedClientMessage>
</Settlement>
```

3.3.5.4 Example XML for RetrieveSettlementValue Response

3.3.5.4.1 Successful transaction

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Settlement TransactionId="A99_090912112233">
      <PaymentMethod name="UK">
        <ResponseCode>0</ResponseCode>
        <ResponseDescription>Settlement Amount Retrieved</ResponseDescription>
        <amountReference>489123</amountReference>
        <voucherCurrency>826</voucherCurrency>
      </PaymentMethod>
      <Status>INFORMATION_SUPPLIED</Status>
      <Amount CurrencyCode="840" Exponent="2">1339</Amount>
      <RecommendedClientMessage><![CDATA[Your transaction has been processed successfully.]]></RecommendedClientMessage>
    </Settlement>
  </Event>
</Response>
```

3.3.5.4.2 Unsuccessful transaction

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
  <Event>
    <Settlement TransactionId="A99_090912112233">
      <PaymentMethod name="UK">
        <ResponseCode>400</ResponseCode>
        <ResponseDescription>Failed</ResponseDescription>
        <amountReference/>
        <voucherCurrency/>
      </PaymentMethod>
      <Status>ERROR</Status>
      <Error>
        <Code>400</Code>
        <Description>Required Currency Conversion not
```

```

        supported</Description>
    </Error>
    <Amount CurrencyCode="840" Exponent="2">1339</Amount>
    <RecommendedClientMessage><![CDATA[Your request could not be
completed. Please contact merchant customer service for more
information.]]></RecommendedClientMessage>
    </Settlement>
</Event>
</Response>
```

3.3.6 *GetInformation*

The GetInformation call to the APG Web Service retrieves information that can be used for the processing of a transaction. This service is only available for the following payment methods

- AM
- BT
- EV
- I3
- MB

The GetInformation web method accepts 2 parameters: accountDetails of type AccountItem and methodDetails of type PaymentMethodItem. The AccountItem and PaymentMethodItem types are custom complex structures. For more information on the composition of this structure, please refer to the WSDL which can be downloaded from the URLs specified in section 3.2.1.

3.3.6.1 Schema Elements for GetInformation Request

In this section, the fields that can be presented with GetInformation transaction Requests will be presented along with example XML for those fields.

- InformationRequest – see section 3.3.6.1.4
 - AccountDetails – see section 3.3.1.1.1
 - MethodDetails – see section 3.3.2.1.26
 - DBTItem – see section 3.3.6.1.1
 - ExperCashChargeBackItem – see section 3.3.6.1.2
 - iDealBankListItem – see section 3.3.2.1.23
 - MoneybookerItem – see section 3.3.2.1.12

SOAP XML GetInformation Request document:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <InformationRequest xmlns="http://DPG-CPT.datacash.com/">
      <accountDetails>
        <AccountName>string</AccountName>
        <AccountPIN>string</AccountPIN>
        <CompanyId>string</CompanyId>
      </accountDetails>
      <methodDetails xsi:type="DBTItem">
```

```

<ServiceType> COUNTRY_LIST or
BANK_TRANSFER_DETAILS</ServiceType>
<Language>string</Language>
<AccountNumber>string</AccountNumber>
<Country>string</Country>
<CurrencyCode>string</CurrencyCode>
<FirstName>string</FirstName>
<LastName>string</LastName>
</methodDetails>
</InformationRequest>
</soap:Body>
</soap:Envelope>

```

3.3.6.1.1 DBTItem

These are the fields or structures contained within the DBTItem structure.

Element Name	DBTItem
Type	DBTItem
Position	InformationRequest.methodDetails
Children	Yes
Attributes	No

Elements of DBTItem				
Element Name	Description	Data type / Limitations	Required / Payment Method Specific details	
ServiceType	This indicates the type of transaction request for DBT. Refer to Appendix 18.	Enum	BT	Mandatory
Language	This is the ISO language code of the consumer. Refer to Appendix 13.	String (20)	BT	Mandatory
AccountNumber	Unique Account number assigned by the Merchant to identify a customer's account. If a unique number or username is not used, insert the Customer's email address into this field.	String (50)	BT	Optional. Mandatory if ServiceType = "BANK_TRANSFER_DETAILS".

Country	This is the country code of the consumer. Refer to Appendix 9.	String (60)	BT	Optional. Mandatory if ServiceType = "BANK_TRANSFER_DETAILS".
CurrencyCode	This is the ISO currency code of the account of the consumer. Refer to Appendix 10.	Short	BT	Optional. Mandatory if ServiceType = "BANK_TRANSFER_DETAILS".
FirstName	The first name of the merchant's customer	String (100)	BT	Optional. Mandatory if ServiceType = "BANK_TRANSFER_DETAILS".
LastName	The last name of the merchant's customer	String (100)	BT	Optional. Mandatory if ServiceType = "BANK_TRANSFER_DETAILS".

Example XML for DBTItem complex elements

```

<ServiceType>COUNTRY_LIST</ServiceType>
<Language>NL</Language>
<AccountNumber>TestMethodAccount</AccountNumber>
<Country></Country>
<CurrencyCode>826</CurrencyCode>
<FirstName>Joe</FirstName>
<LastName>Jackson</LastName>

```

3.3.6.1.2 ExperCashChargeBackItem

These are the fields or structures contained within the ExperCashChargeBackItem structure.

Element Name	ExperCashChargeBackItem
Type	ExperCashChargeBackItem
Position	InformationRequest.methodDetails
Children	Yes
Attributes	No

Elements of ExperCashChargebackItem				
Element Name	Description	Data type/Limitations	Required / Payment Method Specific details	
StartDate	This indicates the start date and time to retrieve the chargeback	String	EV	Optional

EndDate	This indicates the end date and time to retrieve the chargeback.	String	EV	Optional
---------	--	--------	----	----------

Example XML for ExperCashChargebackItem complex elements

```
<StartDate>2010-01-27 09:54</StartDate>
<EndDate>2010-01-28 09:54</EndDate>
```

3.3.6.1.3 AmazonItem

These are the fields or structures contained within the AmazonItem structure.

Element Name	AmazonItem
Type	AmazonItem
Position	InformationRequest.methodDetails
Children	Yes
Attributes	No

Elements of AmazonItem				
Element Name	Description	Data type/Limitations	Required / Payment Method Specific details	
AmazonContractID	The Amazon Contract ID	String	AM	Mandatory

Example XML for ExperCashChargebackItem complex elements

```
<AmazonContractID>
amzn1.contract.1.1.1.8a34d7a8db9a4e8be2b25d9e2e67fff1</AmazonContractID>
```

3.3.6.1.4 InformationRequest

Element Name	InformationRequest
Type	InformationRequest
Position	Soap:Body
Children	Yes
Attributes	Yes

Elements of InformationRequest		
Element Name	Description	Data type / Limitations

accountDetails	This is a set of the merchant's authentication details which is required in order to access the APG.	AccountItem – see section 3.3.1.1.1
methodDetails	This is a parent structure from which all the individual specific payment methods are inherited. For the InformationRequest structure, only the DBTItem and ExperCashChargeBackItem are involved.	PaymentMethodItem – see section 3.3.3.3.1. For details of the DBTItem, see section 3.3.6.1.1. For details of the ExperCashChargeBackItem, see section 3.3.6.1.2

3.3.6.2 Examples of XML for GetInformation Requests

3.3.6.2.1 "Bank Transfer Details" input values

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetInformation xmlns="http://DPG-CPT.datacash.com/">
      <accountDetails>
        <AccountName>TestAccount</AccountName>
        <AccountPIN>TestPIN</AccountPIN>
        <CompanyId>A99</CompanyId>
      </accountDetails>
      <methodDetails xsi:type="DBTItem">
        <ServiceType>BANK_TRANSFER_DETAILS</ServiceType>
        <Language>NL</Language>
        <AccountNumber>TestMethodAccount</AccountNumber>
        <Country>NL</Country>
        <CurrencyCode>978</CurrencyCode>
        <FirstName>Joe</FirstName>
        <LastName>Lotz</LastName>
      </methodDetails>
    </GetInformation >
  </soap:Body>
</soap:Envelope>
```

3.3.6.2.2 "Country List" input values

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetInformation xmlns="http://DPG-CPT.datacash.com/">
      <accountDetails>
        <AccountName>TestAccount</AccountName>
        <AccountPIN>TestPIN</AccountPIN>
```

```

<CompanyId>A99</CompanyId>
</accountDetails>
<methodDetails xsi:type="DBTItem">
    <ServiceType>COUNTRY_LIST</ServiceType>
    <Language>NL</Language>
    <AccountNumber/>
    <Country/>
    <CurrencyCode/>
    <FirstName/>
    <LastName/>
</methodDetails>
</GetInformation>
</soap:Body>
</soap:Envelope>

```

3.3.6.2.3 "Chargeback" input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <GetInformation xmlns="http://DPG-CPT.datacash.com/">
            <accountDetails>
                <AccountName>TestAccount</AccountName>
                <AccountPIN>TestPIN</AccountPIN>
                <CompanyId>A99</CompanyId>
            </accountDetails>
            <methodDetails xsi:type="ExperCashChargeBackItem">
                <StartDate>2010-01-27 09:54</StartDate>
                <EndDate>2010-01-28 09:54</EndDate>
            </methodDetails>
        </GetInformation>
    </soap:Body>
</soap:Envelope>

```

3.3.6.2.4 MoneyBookers input values

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <GetInformation xmlns="http://DPG-CPT.datacash.com/">
            <accountDetails>
                <AccountName>TestAccount</AccountName>
                <AccountPIN>TestPIN</AccountPIN>
                <CompanyId>A99</CompanyId>
            </accountDetails>
            <methodDetails xsi:type="MoneybookersItem">
                <AccountNumber>test@email.com</AccountNumber>
                <PaymentMethodType>NONE</PaymentMethodType>
            </methodDetails>
        </GetInformation>
    </soap:Body>
</soap:Envelope>

```

3.3.6.2.5 Amazon Contract input values

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <GetInformation xmlns="http://DPG-CPT.datacash.com/">
            <accountDetails>
                <AccountName>TestAccount</AccountName>
                <AccountPIN>TestPIN</AccountPIN>
                <CompanyId>A99</CompanyId>
            </accountDetails>
            <methodDetails xsi:type="AmazonItem">
                <AmazonContractID>
amzn1.contract.1.1.1.8a34d7a8db9a4e8be2b25d9e2e67ffff1AmazonContractID</AmazonContractID>
                </methodDetails>
            </GetInformation>
        </soap:Body>
    </soap:Envelope>
```

3.3.6.2.6 iDeal Bank List input values

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <GetInformation xmlns="http://DPG-CPT.datacash.com/">
            <accountDetails>
                <AccountName>TestAccount</AccountName>
                <AccountPIN>TestPIN</AccountPIN>
                <CompanyId>A99</CompanyId>
            </accountDetails>
            <methodDetails xsi:type="iDealBankListItem">
                </methodDetails>
            </GetInformation>
        </soap:Body>
    </soap:Envelope>
```

3.3.6.3 Schema Elements and Example XML for GetInformation Response

The following is a list of possible results that can be returned by the GetInformation web method in the Status field:

- INFORMATION_SUPPLIED
- ERROR

3.3.6.3.1 Country

Element Name:	Country
Position	Response.Event.Information.PaymentMethod.Countries
Children	Yes

Attributes	No
------------	----

Elements of Country		
Element Name	Description	Data type / Limitations
Code	This is the ISO code of the country.	String (2)
Name	This is the name of the country.	String (30)

Example of XML for Country complex elements

```
<Country>
  <Code>NL</Code>
  <Name>Netherlands</Name>
</Country>
```

3.3.6.3.2 Backcharge

Element Name: backcharge	
Position	Response.Event.Information.PaymentMethod.backcharge
Children	Yes
Attributes	No

Elements of backcharge		
Element Name	Description	Data type / Limitations
epi_payment_id	PaymentID of the chargeback.	String (15)
epi_backcharge_date	Date of the chargeback	DateTime
epi_backcharge_type	Always "DD" for direct debit	String(2)
epi_backcharge_reason	Reason for chargeback	Integer

Example of XML for Backcharge complex elements

```

<backcharge>
  <epi_payment_id>DD0328222535000</epi_payment_id>
  <epi_backcharge_date>20031021</epi_backcharge_date>
  <epi_backcharge_type>DD</epi_backcharge_type>
  <epi_backcharge_reason>100</epi_backcharge_reason>
</backcharge>

```

3.3.6.3.3 PurchaseContract

Element Name:	PurchaseContract
Position	Response.Event.Information.PaymentMethod.PurchaseContract
Children	Yes
Attributes	No

Elements of PurchaseContract

Element Name	Description	Data type / Limitations
Id	PurchaseContractId	String (255)
ExpirationTimeStamp	Expiration Time of PurchaseContract	DateTime
MerchantId	AWS MerchantId	String
MarketplaceId	AWS MarketplaceId	String
State	PurchaseContract State	String
Destinations	Delivery Destinations	Array of Destination Elements. See section 0
ShippingAddress->Name	Shipping Name	String
ShippingAddress->AddressLineOne	Shipping Address Line One	String
ShippingAddress->AddressLineTwo	Shipping Address Line Two	String
ShippingAddress->City	Shipping Address City	String
ShippingAddress->PostalCode	Shipping Address Postal Code	String
ShippingAddress->CountryCode	Shipping Address Country Code	String
ShippingAddress->PhoneNumber	Shipping Address Phone Number	String

3.3.6.3.3.1 Destination

Element Name:	Destination
Position	Response.Event.Information.PaymentMethod.PurchaseContract.Destination
Children	Yes
Attributes	No

Elements of Destination

Element Name	Description	Data type / Limitations
DestinationName	Delivery Destinations	String
DestinationType	Delivery Destinations	String
PhysicalDestinationAttributes	Delivery Destinations	ShippingAddress see section 0

3.3.6.3.3.2 ShippingAddress

Element Name:	Destination
Position	Response.Event.Information.PaymentMethod.PurchaseContract.Destination.PhysicalDestinationAttributes.ShippingAddress
Children	Yes
Attributes	No

Elements of Destination

Element Name	Description	Data type / Limitations
Name	Shipping Name	String
AddressLineOne	Shipping Address Line One	String
AddressLineTwo	Shipping Address Line Two	String
City	Shipping Address City	String
PostalCode	Shipping Address Postal Code	String
CountryCode	Shipping Address Country Code	String
PhoneNumber	Shipping Address Phone Number	String

Example of XML for PurchaseContract complex elements

```

<PurchaseContract>
    <Id xmlns="http://payments.amazon.com/checkout/v2/2010-08-31/">amzn1.contract.1.1.1.a90ea1f23f87f4d395213ce3f5c340ce</Id>
    <ExpirationTimeStamp
        xmlns="http://payments.amazon.com/checkout/v2/2010-08-31/">2011-04-04T17:07:14.285Z</ExpirationTimeStamp>
    <MerchantId
        xmlns="http://payments.amazon.com/checkout/v2/2010-08-31/">A98E8GW3O4IUZ</MerchantId>
    <MarketplaceId
        xmlns="http://payments.amazon.com/checkout/v2/2010-08-31/">A1V3L06WH786V3</MarketplaceId>
    <State
        xmlns="http://payments.amazon.com/checkout/v2/2010-08-31/">ACTIVE</State>
    <Destinations
        xmlns="http://payments.amazon.com/checkout/v2/2010-08-31/">
        <Destination>
            <DestinationName>#default</DestinationName>
            <DestinationType>PHYSICAL</DestinationType>
            <PhysicalDestinationAttributes>
                <ShippingAddress>
                    <Name>Ajmuddien Hoosain</Name>
                    <AddressLineOne>FLAT, HABIB HOUSE 9
13</AddressLineOne>
                    <AddressLineTwo>FULHAM HIGH STREET</AddressLineTwo>
                    <City>LONDON</City>
                    <PostalCode>SW6 3JH</PostalCode>
                    <CountryCode>GB</CountryCode>
                    <PhoneNumber>447999578892</PhoneNumber>
                </ShippingAddress>
            </PhysicalDestinationAttributes>
        </Destination>
    </Destinations>
</PurchaseContract>

```

3.3.6.3.4 iDeal Bank List

Element Name:	banks
Position	Response.Event.Information.PaymentMethod.banks
Children	Yes
Attributes	Yes

Elements of Bank		
Element Name	Description	Data type / Limitations
bank	Complex Element holding the individual bank details.	Complex Element
issuer_id	Banks unique Issuer Id.	int

name	Bank name	String(30)
------	-----------	------------

Attributes of Bank		
Element Name	Description	Data type / Limitations
listType	Indicator for where the bank should be displayed on the merchant web site. Values which can be returned: <ul style="list-style-type: none"> • short • long 	String (20)

Example of XML for Country complex elements

```

<banks>
  <bank listType="short">
    <issuer_id>121</issuer_id>
    <name>Test Issuer</name>
  </bank>
  <bank listType="short">
    <issuer_id>151</issuer_id>
    <name>Test Issuer 2</name>
  </bank>
  <bank listType="long">
    <issuer_id>171</issuer_id>
    <name>Test Issuer 3</name>
  </bank>
</banks>

```

3.3.6.3.5 PaymentMethod

Please note that not all the elements in this complex structure are always returned – please see the XML examples for a further explanation.

Element Name:	PaymentMethod
Position	Response.Event.Information
Children	Yes
Attributes	Yes

Attributes involved in the PaymentMethod element			
Attribute Name	Located in which Element	Description	Data type / Limitations
name	PaymentMethod	This string value will be equal to "AM", "BT", "EV", "I3" or "MB"	

Elements of PaymentMethod

Element Name	Description	Data type / Limitations
ResponseCode	This is a return code, which gets returned by the processor.	String (10)
ResponseDescription	This is a text field used to give a short text description of the response code from the processor.	String (255)
BankName	This is the bank name of the account.	String (50)
AccountHoldingBranch	This is the branch where the account was registered.	String (50)
GiroNumber	Additional information pertaining to the bank account.	String (50)
BranchCode	The code of the branch.	String (20)
BankCode	The code of the bank where the account was registered.	String (20)
BSB	Additional information pertaining to the bank account.	String (20)
AccountNumber	The number of the bank account.	String (20)
AccountName	The name of the bank account.	String (35)
CheckDigit	The check digit of the account number	Short
Sortcode	Additional information pertaining to the bank account.	String (15)
IBAN	Additional information pertaining to the bank account.	String (35)
SwiftBIC	Additional information pertaining to the bank account.	String (15)
DBTVAN	VAN model: The customer's Direct Bank Transfer reference number (Virtual Account Number) Merchant Reference model: beneficiary reference including Merchant Reference prefix.	String(30)
timestamp	Time of the reply (only if there is an error)	DateTime
taid	TransactionID (only if there is an error)	String(20)
Countries	A list of country details, each one contained in a country complex structure – see section 3.3.6.3.1	
Backcharges	A list of backcharges, each one contained in a backcharge complex structure – see section 3.3.6.3.2	
PurchaseContract	Amazon PurchaseContract – see section 3.3.6.3.3	

For examples of XML for PaymentMethod complex elements, see section 0

3.3.6.3.6 Information

Please note that not all the elements in this complex structure are always returned – please see the XML examples for a further explanation.

Element Name:	Information
Position	Response.Event
Children	Yes
Attributes	Yes

Attributes involved in the Information element			
Attribute Name	Located in which Element	Description	Data type
InformationId	Information	This is an ID to identify the information request and response. This field will comprise of the CompanyId and the current date and time, e.g. "A99_090204143312093".	String (19)

Elements of Information		
Element Name	Description	Data type / Limitations
PaymentMethod	See section 3.3.6.3.5	
Status	This is the response to the GetInformation transaction.	String (20) These are the possible values: <ul style="list-style-type: none">• INFORMATION_SUPPLIED• ERROR
RecommendedClientMessage	See section 3.3.2.3.4 for more information.	
Error	See section 3.3.1.2.1. This element will appear in the Information XML if the Status = "ERROR".	

Example of XML for Information complex elements

See the samples in section 3.3.6

3.3.6.4 Examples of XML for GetInformation Responses

3.3.6.4.1 "Bank Transfer Details" response values

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
  <Event>
    <Information InformationId="A99_090204143312093">
      <PaymentMethod name="BT">
        <ResponseCode>0</ResponseCode>
        <ResponseDescription>Bank Transfer Details Retrieved</ResponseDescription>
        <BankName>ABN Amro Bank</BankName>
        <AccountHoldingBranch>Amsterdam</AccountHoldingBranch>
        <GiroNumber/>
        <BranchCode>AMS_639AB</BranchCode>
        <BankCode>AAB</BankCode>
        <BSB/>
        <AccountNumber>12345667804275</AccountNumber>
        <AccountName>TestAccountName</AccountName>
        <CheckDigit/>
        <SortCode/>
        <IBAN>ABC-12345-DE</IBAN>
        <SwiftBIC/>
        <dbtVAN>XYZ_12A12B12C99</dbtVAN>
      </PaymentMethod>
      <Status>INFORMATION_SUPPLIED</Status>
      <RecommendedClientMessage><![CDATA[Requested information has been provided.]]></RecommendedClientMessage>
    </Information>
  </Event>
</Response>
```

3.3.6.4.2 "Country List" response values

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
  <Event>
    <Information InformationId="A99_090204143012197">
      <PaymentMethod name="BT">
        <ResponseCode>0</ResponseCode>
        <ResponseDescription>Countries Retrieved</ResponseDescription>
        <Countries>
          <Country>
            <Code>NL</Code>
            <Name>Netherlands</Name>
          </Country>
        </Countries>
      </PaymentMethod>
      <Status>INFORMATION_SUPPLIED</Status>
      <RecommendedClientMessage><![CDATA[Requested information has been provided.]]></RecommendedClientMessage>
    </Information>
  </Event>
</Response>
```

3.3.6.4.3 "BackCharge" response values

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
  <Event>
    <Information InformationId="A99_4629457823859">
      <PaymentMethod name="EV">
        <ResponseCode>Success</ResponseCode>
        <ResponseDescription>Success</ResponseDescription>
        <BackCharges>
          <epi_payment_id>DD0328222535000</epi_payment_id>
          <epi_backcharge_date>20031021</epi_backcharge_date>
          <epi_backcharge_type>DD</epi_backcharge_type>
          <epi_backcharge_reason>100</epi_backcharge_reason>
        </BackCharges>
      </PaymentMethod>
      <Status>INFORMATION_SUPPLIED</Status>
      <RecommendedClientMessage>string</RecommendedClientMessage>
    </Information>
  </Event>
</Response>

```

3.3.6.4.4 Unsuccessful response values

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
  <Event>
    <Information InformationId="A99_090204144314157">
      <PaymentMethod name="BT">
        <ResponseCode/>
        <ResponseDescription/>
      </PaymentMethod>
      <Status>ERROR</Status>
      <Error>
        <Code>002</Code>
        <Description>AccountDetails must be supplied</Description>
      </Error>
      <RecommendedClientMessage><![CDATA[Your request could not be completed. Please contact the merchant's customer service for more information.]]></RecommendedClientMessage>
    </Information>
  </Event>
</Response>

```

3.3.6.4.5 "PurchaseContract" response values

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
  <Event>
    <Information InformationId="T10_110404140839435">
      <PaymentMethod name="AM">
        <ResponseCode>100</ResponseCode>
        <ResponseDescription/>
        <PurchaseContract>
          <Id xmlns="http://payments.amazon.com/checkout/v2/2010-08-31/">amzn1.contract.1.1.1.a90ea1f23f87f4d395213ce3f5c340ce</Id>
          <ExpirationTimeStamp
            xmlns="http://payments.amazon.com/checkout/v2/2010-08-31/">2011-04-

```

```

04T17:07:14.285Z</ExpirationTimeStamp>
    <MerchantId
xmlns="http://payments.amazon.com/checkout/v2/2010-08-
31/">A98E8GW3O4IUZ</MerchantId>
    <MarketplaceId
xmlns="http://payments.amazon.com/checkout/v2/2010-08-
31/">A1V3L06WH786V3</MarketplaceId>
        <State xmlns="http://payments.amazon.com/checkout/v2/2010-08-
31/">ACTIVE</State>
        <Destinations
xmlns="http://payments.amazon.com/checkout/v2/2010-08-31/">
            <Destination>
                <DestinationName>#default</DestinationName>
                <DestinationType>PHYSICAL</DestinationType>
                <PhysicalDestinationAttributes>
                    <ShippingAddress>
                        <Name>Ajmuddien Hoosain</Name>
                        <AddressLineOne>FLAT, HABIB HOUSE 9
13</AddressLineOne>
                        <AddressLineTwo>FULHAM HIGH STREET</AddressLineTwo>
                        <City>LONDON</City>
                        <PostalCode>SW6 3JH</PostalCode>
                        <CountryCode>GB</CountryCode>
                        <PhoneNumber>447999578892</PhoneNumber>
                    </ShippingAddress>
                </PhysicalDestinationAttributes>
            </Destination>
        </Destinations>
    </PurchaseContract>
</PaymentMethod>
<Status>INFORMATION_SUPPLIED</Status>
<RecommendedClientMessage><![CDATA[Your transaction has been
processed successfully.]]></RecommendedClientMessage>
    </Information>
</Event>
</Response>

```

3.3.6.4.6 MoneyBooker response values

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
    <Event>
        <Information InformationId="A99 090204143312093">
            <PaymentMethod name="MB">
                <ResponseCode>OK</ResponseCode>
                <ResponseDescription>Customer Information
retrieved</ResponseDescription>
                <CustomerID>2900704</CustomerID>
                <ACCOUNTTYPE>T</ACCOUNTTYPE>
                <FNAME>Test</FNAME>
                <LNAME>Test</LNAME>
                <ADDRESS1>No. 1 Test</ADDRESS1>
                <CITY>Test City</CITY>
                <ZIPCODE>1234</ZIPCODE>
                <STATE>Test</STATE>
                <COUNTRY>Test</COUNTRY>
                <DOB>20011980</DOB>

```

```

<LOCKLEVEL>NOT_LOCKED</LOCKLEVEL>
</PaymentMethod>
<Status>INFORMATION_SUPPLIED</Status>
<RecommendedClientMessage><![CDATA[Requested information has been
provided.]]></RecommendedClientMessage>
</Information>
</Event>
</Response>
```

3.3.6.4.7 iDeal Bank List response values

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
    <Event>
        <Information InformationId="A99_090204143012197">
            <PaymentMethod name="I3">
                <ResponseCode>0</ResponseCode>
                <ResponseDescription>Transaction
Successful</ResponseDescription>
                <banks>
                    <bank listType="short">
                        <issuer_id>121</issuer_id>
                        <name>Test Issuer</name>
                    </bank>
                    <bank listType="short">
                        <issuer_id>151</issuer_id>
                        <name>Test Issuer 2</name>
                    </bank>
                    <bank listType="long">
                        <issuer_id>171</issuer_id>
                        <name>Test Issuer 3</name>
                    </bank>
                </banks>
            </PaymentMethod>
            <Status>INFORMATION_SUPPLIED</Status>
            <RecommendedClientMessage><![CDATA[Requested information has been
provided.]]></RecommendedClientMessage>
        </Information>
    </Event>
</Response>
```

3.3.7 Modification

The Modification call to the APG Web Service is used to issue an instruction to update the status of a Purchase transaction which was already submitted to the APG. This service is only available for the following payment methods:

- AM
- EL
- EV
- W3

The following actions can be performed on a Modification transaction:

- CANCEL
- CAPTURE
- REFUND

For more information on these actions, please refer to Appendix 19.

The Modification web method accepts five parameters namely; accountDetails of type AccountItem, DPGReferenceId of type Long, Type of Action of type ActionType, amountDetails of type AmountItem and Note of type String. The AccountItem and AmountItem types are custom complex structures, and the ActionType type is a custom simple structure. For more information on the composition of these structures, please refer to the WSDL which can be downloaded from the URLs specified in section 0.

3.3.7.1 Schema Elements for Modification Request

In this section, the fields that can be presented with Modification transaction Requests will be presented along with example XML for those fields.

- Modification
 - AccountDetails – section 3.3.1.1.1
 - DPGReferenceID
 - TypeOfAction
 - AmountDetails

SOAP XML Modification Request document:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <Modification xmlns="http://DPG-CPT.datacash.com/">
      <accountDetails>
        <AccountName>string</AccountName>
        <AccountPIN>string</AccountPIN>
        <CompanyId>string</CompanyId>
      </accountDetails>
      <DPGReferenceId>long</DPGReferenceId>
      <typeOfAction>CANCEL or CAPTURE or REFUND</typeOfAction>
      <amountDetails>
        <Amount>int</Amount>
        <Exponent>short</Exponent>
        <CurrencyCode>short</CurrencyCode>
      </amountDetails>
    </Modification>
  </soap:Body>
</soap:Envelope>
```

3.3.7.1.1 Modification

Element Name	Modification
Type	Modification
Position	Soap:Body
Children	Yes

Attributes	Yes
------------	-----

Elements of Modification

Element Name	Description	Data type / Limitations
accountDetails	This is a set of the merchant's authentication details which is required in order to access the APG.	AccountItem – see section 3.3.1.1.1
DPGReferenceId	The DPGReferenceId is a unique APG transaction identifier which is generated by the APG for each merchant transaction. This identifier is returned in the response to a Purchase or Payout transaction.	Long This value is mandatory at all times.
typeOfAction	This field indicates the type of Action that the Modification web method needs to perform. Please reference Appendix 19.	Enum This value is mandatory at all times. Supported Action Types: <ul style="list-style-type: none">• CANCEL• CAPTURE• REFUND
amountDetails	This is a set of amount details.	AmountItem – see section 3.3.1.1.2 Optional if action is Refund. Amount cannot be higher than original purchase amount. Currency must be the same as original purchase currency. If AmountDetails details structure is sent to APG, then all its fields must be populated with valid values.
Note	This is a description for the transaction	String (255) Optional

See the next section for an XML example.

3.3.7.2 Example of XML for Modification Requests

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <Modification xmlns="http://DPG-CPT.datacash.com/">
            <accountDetails>
                <AccountName>TestAccount</AccountName>
                <AccountPIN>TestPIN</AccountPIN>
                <CompanyId>A99</CompanyId>
            </accountDetails>
            <DPGReferenceId>12345</DPGReferenceId>
            <typeOfAction>CANCEL</typeOfAction>
            <amountDetails>
                <Amount>550</Amount>
                <Exponent>2</Exponent>
                <CurrencyCode>978</CurrencyCode>
            </amountDetails>
        </Modification>
    </soap:Body>
</soap:Envelope>

```

3.3.7.3 Schema Elements and Example XML for Modification Response

The Response that is generated for a Modification Request is similar to the Response that is generated for a Purchase transaction.

See section 3.3.2.3.4 for details of the Purchase (and Modification) Response structure.

See section 3.3.2.5 for examples of these Responses.

The following is a list of possible results that can be returned by the Modification web method in the Status field:

- AUTHORISED
- CANCELLED
- ERROR
- NOT_FOUND
- REFUNDED
- SENT_FOR_CAPTURE
- SENT_FOR_REFUND

3.3.7.3.1 Example Modification Responses

3.3.7.3.1.1 AUTHORISED

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Modification DPGReferenceId="113836"
                      TransactionId="A99_091127163001">
            <PaymentMethod name="EV">
                <ReferenceId>00000000000040295434</ReferenceId>
                <ResponseCode>100</ResponseCode>
                <ResponseDescription>Transaction
successfull</ResponseDescription>
                <TimeStamp>2009-11-30 09:25:53</TimeStamp>
                <PaymentId>DD0677342915000</PaymentId>
            </PaymentMethod>
            <Status>AUTHORISED</Status>
            <Amount CurrencyCode="978" Exponent="2">5239</Amount>
        </Modification >
    </Event>
</Response>
```

3.3.7.3.1.2 CANCELLED

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Modification DPGReferenceId="113836"
                      TransactionId="A99_091127163001">
            <PaymentMethod name="EV">
                <ReferenceId>00000000000040313557</ReferenceId>
                <ResponseCode>100</ResponseCode>
                <ResponseDescription>Transaction
successfull</ResponseDescription>
                <TimeStamp>2009-11-30 10:51:16</TimeStamp>
            </PaymentMethod>
            <Status>CANCELLED</Status>
        </Modification >
    </Event>
</Response>
```

3.3.7.3.1.3 REFUNDED

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchAccount">
    <Event>
        <Modification DPGReferenceId="113835"
                      TransactionId="A99_091127161216">
            <PaymentMethod name="EV">
                <ReferenceId>00000000000040315286</ReferenceId>
                <ResponseCode>100</ResponseCode>
                <ResponseDescription>Transaction
successfull</ResponseDescription>
                <TimeStamp>2009-11-30 11:03:23</TimeStamp>
            </PaymentMethod>
            <Status>REFUNDED</Status>
        </Modification >
    </Event>
</Response>
```

```

        <Amount CurrencyCode="978" Exponent="2">550</Amount>
    </Modification>
</Event>
</Response>
```

3.3.7.3.1.4 ERROR

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
<Event>
    <Modification DPGReferenceId="113879"
    TransactionId="A99_091202120706">
        <PaymentMethod name="EV">
            <ReferenceId/>
            <ResponseCode/>
            <ResponseDescription/>
        </PaymentMethod>
        <Status>ERROR</Status>
        <Error>
            <Code>020</Code>
            <Description>PurchaseDetails.AccountDetails.AccountPIN value
                invalid</Description>
        </Error>
    </Modification >
</Event>
</Response>
```

3.3.7.3.1.5 NOT_FOUND

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="">
<Event>
    <Modification DPGReferenceId="113879" TransactionId="">
        <PaymentMethod name="EV">
            <ReferenceId/>
            <ResponseCode/>
            <ResponseDescription/>
        </PaymentMethod>
        <Status>NOT_FOUND</Status>
        <Error>
            <Code>035</Code>
            <Description>Transaction not found</Description>
        </Error>
    </Modification >
</Event>
</Response>
```

3.3.7.3.1.6 SENT_FOR_CAPTURE

```

<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="MerchantAccount">
<Event>
    <Modification DPGReferenceId="41179"
    TransactionId="T10_PNTJTNJ4L3IV">
        <PaymentMethod name="EL">
            <ReferenceId>4CB42399C8511E228E17C1DB286ED114B1CB54A1</ReferenceId>
```

```
<ResponseCode>1</ResponseCode>
<ResponseDescription>Approved and
Completed</ResponseDescription>
<TimeStamp>20101012050009</TimeStamp>
</PaymentMethod>
<Status>SENT_FOR_CAPTURE</Status>
<Amount CurrencyCode="840" Exponent="2">950</Amount>
</Modification>
</Event>
</Response>
```

3.3.7.3.1.7 SENT_FOR_REFUND

```
<?xml version="1.0" encoding="utf-8"?>
<Response version="1.0" AccountName="041756">
    <Event>
        <Modification DPGReferenceId="41181"
TransactionId="T10_NAVD1ABSLAUB">
            <PaymentMethod name="EL">
                <ReferenceId>4CB4247590C560C47F53F66E3D7E96EEF5F354C3</ReferenceId>
                <ResponseCode>1</ResponseCode>
                <ResponseDescription/>
                <TimeStamp>20101012050349</TimeStamp>
            </PaymentMethod>
            <Status>SENT_FOR_REFUND</Status>
            <Amount CurrencyCode="840" Exponent="2">950</Amount>
        </Modification>
    </Event>
</Response>
```

[Back to Contents](#)

3.4 APG EVENT NOTIFICATIONS

The APG system has events that take place outside the normal flow of a transaction. When such events take place, notifications will be sent to the merchant's Event_URL. Below is a list of the events that can take place out-of-process:

- AUTHORISED
- CANCELLED
- CAPTURED
- CHARGED_BACK
- MARKED_FOR_REFUND
- PENDING
- REFUNDED
- REFUSED
- SEND_FOR_CAPTURE
- SEND_FOR_REFUND

Example: In a MoneyBookers transaction, the transaction amount exceeds the available balance in the consumer's account. This transaction can then become "Pending" in APG (inside the normal flow of this transaction). The consumer can then either cancel or pay the remainder on the MoneyBookers website during the next two days. MoneyBookers will send this final transaction result to the APG (outside the normal flow of this transaction), and the APG will map this result to one of the responses/events mentioned above, and send the relevant notification to the merchant's Event_URL.

When a notification is sent to the Event_URL, the merchant will need to analyze the event and then take the appropriate action. After receiving the event, the merchant needs to confirm receipt of the notification by sending an OK response back to the APG system. In the event that the APG does not receive a confirmation of receipt, the APG will retry multiple times to send the notification.

Refer to section 0 for examples of event notifications (any of the responses, excluding Redirect responses, can be sent as event notifications).

When a merchant receives this event notification, they then need to respond with the following XML:

```
<Response>OK</Response>
```

[Back to Contents](#)