

# **Transparent Redirect**

Credit & Debit Card Processing

v3.0.0

## Table of Contents

Introduction .....	4
Intended Audience .....	4
Simplifying the Integration Process.....	4
Transaction Data Flow .....	5
Important Notes.....	7
Gateway URLs.....	7
Hashing Explained .....	7
Simple Hashing Example .....	7
Creating The Hash Digest .....	8
Form Variables .....	10
Table Definitions.....	10
Form Variable Definitions.....	11
Initial Request .....	17
Input Variables .....	17
3D Secure Authentication Required - Fingerprint .....	20
Output Variables .....	20
3D Secure - Environment Request .....	21
Input Variables .....	21
3D Secure Authentication Required - Challenge .....	22
Output Variables .....	22
3D Secure - Authentication Request.....	23
Input Variables .....	23
Payment Complete.....	24
Output Variables .....	24
Appendix 1 - Transaction Status Codes.....	26
Appendix 2 - ACS Simulator .....	27
Appendix 3 - Example Form Data.....	28
Initial Request.....	28
3D Secure Authentication Required – Fingerprint .....	29
3D Secure - Environment Request .....	29
3D Secure Authentication Required – Challenge .....	30

3D Secure - Authentication Request .....	30
Payment Complete .....	31
Appendix 4 - Override Policy Codes & Explanations.....	33
OverrideAVSPolicy Codes .....	33
Character 1 .....	33
Character2.....	33
Character 3 .....	33
Character 4 .....	33
Examples .....	34
Notes .....	34
OverrideCV2Policy Codes .....	35
Character 1 .....	35
Character2.....	35
Examples .....	35
Notes .....	35
Appendix 5 – Abbreviations .....	36
Appendix 6 – Country Codes (ISO 3166-1).....	37
Appendix 7 - Currency Codes (ISO 4217) .....	44

## Introduction

### Intended Audience

This document is technical in nature and should be used by your company's developers to integrate our payment gateway into your systems. It assumes that the reader has knowledge and understanding of basic HTML concepts such as form post.

### Simplifying the Integration Process

There are many complexities when dealing with card transactions. If you try and tackle them all at once the task of integrating will seem complicated. The best way to do the integration is to follow a simple step by step approach and break the process down into manageable sections, each adding functionality as you go along.

Adhering to good coding practices will also greatly simplify your task.

## Transaction Data Flow

The Transparent Redirect API allows the merchant to integrate our payment system into their website without having to worry about the implications of handling sensitive credit/debit card information. This simplifies the requirements for PCI DSS compliance.

A Transparent Redirect transaction follows the following steps.

1. The cardholder visits the merchant's website and enters their card details on the merchant's payment form. This is the merchant's payment form and is hosted on the merchant's system.
2. When the form is submitted it is NOT sent back to the merchant's server but is instead sent (re-directed) to the Transparent Redirect URL (the "Initial Request"). All the transaction information is sent, including the customer's information and the card details. The customer is unaware that the form has been re-directed as nothing changes on their screen whilst processing takes place. The Transparent Redirect API validates the data it receives.
  - If any errors are found the transaction will not proceed further. The error details are passed back to the merchant's server (to the CallbackURL) - continue with step 16.
3. The Transparent Redirect API sends the transaction data in a CardDetailsTransaction request to the Gateway for processing.
4. The Gateway checks the status of the card's enrolment for 3D Secure.
  - If the card is enrolled and a MethodURL is available, the Gateway returns both the MethodURL and the associated MethodData to the merchant's server via the Transparent Redirect API and the customer's browser (a "3D Secure Authentication Required - Fingerprint" response) - continue with step 5.
  - If the card is enrolled but a MethodURL is not available, the Gateway automatically proceeds to the authentication stage - continue with step 9.
  - If the card is not enrolled for 3D Secure processing the Gateway automatically proceeds to the authorisation stage - continue with step 15.
5. The merchant's system validates the data it receives and re-posts the MethodData to the MethodURL (the ACS) via the customer's browser.
6. The ACS interacts with the customer's browser to analyse ("fingerprint") the customer's environment. Gathering this data reduces the chances of a subsequent challenge. It then returns the MethodData (slightly modified) to the FingerprintNotificationURL via the customer's browser. The customer will be unaware of any of this processing.
7. The merchant's system re-posts the MethodData to the Transparent Redirect API (a "3D Secure - Environment Request") via the customer's browser.

8. The Transparent Redirect API validates the data it receives and sends the MethodData in a ThreeDSecureEnvironment request to the Gateway.
9. The Gateway sends an Authentication Request (AReq) to the ACS.
  - If the response from the ACS (ARes) indicates that a challenge is required (a Challenge transaction), the Gateway returns a Challenge Request (CReq) and the ACSURL to the merchant's server via the Transparent Redirect API and the customer's browser (a "3D Secure Authentication Required - Challenge" response) - continue with step 10.
  - If the response from the ACS (ARes) indicates that no challenge is required (a Frictionless transaction), the Gateway automatically proceeds to the authorisation stage - continue with step 15.
10. The merchant's server validates the data it receives and re-posts the CReq to the ACSURL (the ACS) via the customer's browser.
11. The ACS displays a "challenge" window in the customer's browser (similar to the one shown in Appendix 2), asking the customer to provide some information - for example a One-Time Password (OTP).
12. The ACS sends the result of the challenge directly to the Gateway, but also sends the Challenge Response (CRes) back to the ChallengeNotificationURL via the customer's browser.
13. The merchant's system re-posts the CRes back to the Transparent Redirect API (a "3D Secure - Authentication Request") via the customer's browser.
14. The Transparent Redirect API validates the data it receives and sends the CRes in a ThreeDSecureAuthentication request to the Gateway.
15. The Gateway completes the 3D Secure process and submits the transaction to the Acquirer for Authorisation. The result of the authorisation is returned to the merchant's server (to the CallbackURL) via the Transparent Redirect API and the customer's browser.
16. The merchant's system should display the result of the completed transaction to the customer. The result will usually be an Authorised or Declined message, but if an error has occurred during processing it may contain error information. The exact message displayed is up to the merchant.



## IMPORTANT

### Important Notes

#### Gateway URLs

In this document, payment gateway specific URLs have "thepaymentgateway.net" as the domain (for example <https://mms.thepaymentgateway.net/>). When using these URLs in the integration, "thepaymentgateway.net" must be replaced by with the name of the payment gateway provider.

The generalised full URL to use in your posts to the Transparent Redirect method is:

<https://mms.paymentprocessor.net/Pages/PublicPages/TransparentRedirect.aspx>

#### Hashing Explained

The transaction data must be protected as it is being passed to and from the Transparent Redirect page via the customer's browser. The data is protected using hashing. Hashing is used to produce a unique "signature" for the data being passed (it is generated using the data being transmitted together with secret data that is not transmitted, so it is impossible to recreate the hash digest using just the data that is passed via the browser). The hash signature is re-calculated by our system on receipt of the transmitted data and, if it does not match the hash signature that was transmitted with the data, then the data has been tampered with and the transaction will stop with an error message. The same process (in reverse) should be carried out by your site on receipt of the transaction results.

Examples of this type of tampering could be lowering the transaction price (say from £100.00 down to £1.00) or making a failed transaction look like an authorised one. This is known as a "man-in-the-middle" attack.

#### Simple Hashing Example

Here is an example of some transaction variables:

**MerchantID:** YourCo-1234567

**Amount:** 100.00

**CurrencyCode:** 826

**OrderID:** 12345

These variables would be concatenated (in a specific order) and combined with data known only to your system and ours (the account password and PreSharedKey) which is NOT transmitted with the transaction request. This produces the following string:

PreSharedKey=ASecretKey&Password=MyPassword&

MerchantID=YourCo-1234567&Amount=10000&CurrencyCode=826&OrderID=12345

A simple hash method would output the following hash digest (or "Signature") when this string is passed into a hashing function – the hash digest is also transmitted with the transaction variables:

5c6b9c913b2301e9aa6ff488b06e09273cded2a5

If the amount was altered from £100.00 to £1.00:

**MerchantID:** YourCo-1234567

**Amount:** 1.00

**CurrencyCode:** 826

**OrderID:** 12345

Our system would receive these values and build the following string:

PreSharedKey=ASecretKey&Password=MyPassword&

MerchantID=YourCo-1234567&Amount=100&CurrencyCode=826&OrderID=12345

When passed into the same hashing function this would produce a different hash digest (or "Signature"):

4ba1164acbec732c18cd6e5f632adcdd4b440237

This demonstrates that changing any of these variables, even just a single character, results in a different hash digest and makes the process of detecting variable tampering very easy.

## Creating The Hash Digest

A hash digest is created by hashing a string containing secret information together with the form variables specified.

The first variable in the string must be the PreSharedKey, followed by the Password (the password for the merchant's account in the Gateway), followed by the variables marked as "Include In Hash" in the relevant table.

Important:

- The PreSharedKey should ONLY be included if the chosen hash method is standard MD5 or SHA1 (i.e. not HMAC). If the chosen hash method is either HMACMD5 or HMACSHA1, then the pre shared key is used as part of the hash generation so should be ENTIRELY omitted from the string to be hashed - if it is present in these cases (even as an empty string), then an error will be thrown.
- Variables must be added in the order listed in the tables.
- The variable names and values are case-sensitive and should be added EXACTLY as in the form data (not URL encoded).
- Variables only need to be included if they are being sent in the set of posted form variables.
- Special characters in text fields must be properly escaped, otherwise the hash digest will not match.



- All the variables should be added to the string in standard URL format – in name/value pairs, with each pair separated by an ampersand character.
- Do not include any line breaks in your string.
- The resulting hash digest should be sent in the HashDigest form field.

Example (showing only the first four variables):

PreSharedKey=value&Password=value&MerchantID=value&TransactionDateTime=value&...

If line items are included, each set of three fields (quantity, amount, description) must be kept together, and each set added sequentially incrementing the index number for each line item. For example:

LineItem0Quantity=4&LineItem0Amount=200&LineItem0Description=First example item&  
LineItem1Quantity=1&LineItem1Amount=1299&LineItem1Description=Second example item

The HashDigest in responses from the Transparent Redirect API will only contain variables if they are included in the set of returned form variables.

## Form Variables

Communication between the merchant's system/customer's browser to the gateway is via POSTed HTML form variables.

An example HTML form variable which would get POSTed to the gateway, or returned to the merchant's system:

```
<input type="hidden" name="FieldName" value="Field Value" />
```

## Table Definitions

### Field Name

The value in this field is **case sensitive** and should be sent exactly as is stated here. Failure to do so will result in errors, specifically relating to missing variables or hash digest mismatches.

### Data Type

All variables will be converted to a string when the HTML is rendered anyway, but this is to stipulate what the expected value should be readable as. For example, a Boolean value should only be sent as "TRUE" or "FALSE". Some scripting languages like PHP only state "1" or "0" for Boolean, which would be deemed invalid by the gateway, so all values must be converted first.

Data Type	Description
N	Numeric – only numbers allowed.
A	Alpha – any printable character is allowed.
B	Boolean – only TRUE or FALSE are allowed.
DT	Date/Time – Details of the format may be provided in the Comments section.
-	Special types – these variables only allow a specific set of values. Details of the allowed values are given in the Comments section.

### Max Length

This is the maximum length that the gateway will allow for the variable. Fields marked as "-" are variable in length – the Comments will usually contain further details of permitted values. If a value is specified, then a variable longer than this will result in an error.

### Comments

This field should be read thoroughly to determine if there is additional information relevant to the integration you are performing.

### Form Inclusion

Form Inclusion	Description
M	Mandatory – the field must be provided (in a response, it will be included in the data returned).
O	Optional – the field should only be provided if it has a value.
C	Conditional – the field should be provided depending on certain conditions (refer to any notes provided).

## Form Variable Definitions

The table below provides details of all the form variables used in the various requests and responses between the merchant's system and the Transparent Redirect API. They are listed in alphabetical order.

Field Name	Data Type	Max Length	Comments
ACSURL	A	256	The URL to which the MethodData must be sent.
Address1	A	100	Customer's billing address line 1.
Address2	A	50	Customer's billing address line 2.
Address3	A	50	Customer's billing address line 3.
Address4	A	50	Customer's billing address line 4.
AddressNumericCheckResult	A	-	If requested (EchoAVSCheckResult=TRUE in initial request) this gives the result of the address numeric check - will be one of: <ul style="list-style-type: none"> <li>• PASSED</li> <li>• FAILED</li> <li>• PARTIAL</li> <li>• NOT_CHECKED</li> <li>• UNKNOWN</li> </ul>
Amount	N	12	The transaction amount in minor currency. e.g. £10.00 must be submitted as 1000
AVSOverridePolicy	A	4	Sets an override AVS checking policy for this transaction.
BrowserColorDepth	N	2	If JavaScript is enabled, this can be obtained from the browser using the screen.colorDepth property.
BrowserJavaEnabled	B	-	A Boolean value indicating whether Java is enabled in the customer's browser. If JavaScript is enabled, this can be obtained from the browser using the navigator.javaEnabled property.
BrowserJavaScriptEnabled	B	-	A Boolean value indicating whether JavaScript is enabled in the customer's browser.
BrowserLanguage	A	256	The language that the customer's browser is set to use as defined in IETF BCP47 / RFC 5646. If JavaScript is enabled, this can be obtained from the browser using the navigator.language property. e.g. "en-GB"
BrowserScreenHeight	N	6	The height of the device's screen in CSS pixels. If JavaScript is enabled, this can be obtained from the browser using the screen.height property.
BrowserScreenWidth	N	6	The width of the device's screen in CSS pixels. If JavaScript is enabled, this can be obtained from the browser using the screen.width property.

BrowserTimeZone	N	5	The time zone offset in minutes between UTC and the device's local time. If JavaScript is enabled, this can be obtained from the browser using the getTimezoneOffset() function, with an optional sign (+/-). The offset is positive if the local time zone is behind UTC and negative if it is ahead. e.g. "60", "+60", or "-120"
CallbackURL	A	256	The URL of the page on the merchant's website that the results of the transaction will be posted back to.
CardClass	A	-	If requested (EchoCardType=TRUE in initial request) this gives the class of the card used.
CardExpiryDate	A	5	If requested (EchoCardExpiryDate =TRUE in initial request) this gives the expiry date of the card used, in the form "MM/YY". e.g. "02/28"
CardIssuer	A	-	If requested (EchoCardType=TRUE) this gives the issuer of the card used, if known.
CardIssuerCountryCode	N	3	If requested (EchoCardType=TRUE) this gives the three-digit country code of the issuer of the card used, if known. (ISO 3166- 1)
CardName	A	100	The cardholder's name as it appears on the front of the card.
CardNumber	N	20	The full card number as it appears on the front of the card, without spaces.
CardNumberFirstSix	N	6	If requested (EchoCardNumberFirstSix=TRUE in initial request) this gives the first six digits of the card number used.
CardNumberLastFour	N	4	If requested (EchoCardNumberFirstSix=TRUE in initial request) this gives the last four digits of the card number used.
CardType	A	-	If requested (EchoCardType=TRUE in initial request) this gives the type of the card used. e.g. "VISA"
ChallengeNotificationURL	A	256	The full URL on the merchant's system that the customer's browser will be redirected back to after a 3D Secure challenge has been completed.
City	A	50	Customer's billing address city.
ClientReference	A	24	A unique reference for the transaction provided by the Merchant (if not provided one is created automatically).
CReq	A	-	The Challenge Request data, a Base64URL encoded value that contains a set of data that is required by the ACS to perform the authentication Challenge.

CRes	A	-	The Challenge Response data, a Base64URL encoded value that contains a set of data returned from the ACS after completing the authentication Challenge.
CrossReference	A	24	The cross reference, a unique identifier for the transaction returned by the Gateway.
CountryCode	N	3	Customer's billing country code (ISO 3166- 1). e.g. United Kingdom: 826
CurrencyCode	N	3	The currency of the transaction as a three digit currency code (ISO 4217). e.g. 826 (for GBP)
CV2	N	4	The card's CV2 security number (also called CVV, etc.), 3 or 4 digits.
CV2CheckResult	A	-	If requested (EchoCV2CheckResult=TRUE in initial request) this gives the result of the CV2check - will be one of: <ul style="list-style-type: none"> <li>• PASSED</li> <li>• FAILED</li> <li>• NOT_CHECKED</li> <li>• UNKNOWN</li> </ul>
CV2OverridePolicy	A	2	Sets an override CV2 checking policy for this transaction.
DateOfBirth	DT	10	Customer's date of birth, in the form "YYYY-MM-DD". e.g. "1995-08-21"
EchoAVSCheckResult	B	-	Instructs the payment form to include the result of the AVS check in the output variables. Note: defaults to FALSE if not provided.
EchoCardExpiryDate	B	-	Instructs the payment form to include the card expiry date in the output. Note: defaults to FALSE if not provided.
EchoCardNumberFirstSix	B	-	Instructs the payment form to include the first six digits of the card number in the output variables. Note: defaults to FALSE if not provided.
EchoCardNumberLastFour	B	-	Instructs the payment form to include the last four digits of the card number in the output variables. Note: defaults to FALSE if not provided.
EchoCardType	B	-	Instructs the payment form to include the card type in the output variables. Note: defaults to FALSE if not provided.
EchoCV2CheckResult	B	-	Instructs the payment form to include the result of the CV2 check in the output variables. Note: defaults to FALSE if not provided.
EchoFraudProtectionCheckResult	B	-	Instructs the payment form to include the result of the Fraud protection check in the output variables. Note: defaults to FALSE if not provided.

EchoThreeDSecureAuthenticationCheckResult	B	-	Instructs the payment form to include the result of the 3D Secure check in the output variables. Note: defaults to FALSE if not provided.
EmailAddress	A	100	Customer's email address (should be a valid email address).
ExpiryDateMonth	N	2	The first 2 digits from the card's expiry date representing the month part of the expiry date. e.g. 02 if the expiry date is 02/28
ExpiryDateYear	N	2	The last 2 digits from the card's expiry date representing the year part of the expiry date. e.g. 28 if the expiry date is 02/28
FingerprintNotificationURL	A	256	The full URL on the merchant's system that the customer's browser will be redirected back to after the device fingerprinting has been completed.
FraudProtectionCheckResult	A	-	If requested (EchoFraudProtectionCheckResult =TRUE in initial request) this gives the result of the Fraud Protection check - will be one of: <ul style="list-style-type: none"> <li>• PASSED</li> <li>• FAILED</li> <li>• CHALLENGE</li> <li>• ERROR</li> </ul>
HashDigest	A	-	A hashed string that contains all the variables passed, together with some data that is not passed but is known to both sides - the PreSharedKey and the gateway account password.
IssueNumber	N	2	The card's issue number as it appears on the front on the card.
LineItem0Amount	N	15	The monetary amount (cost/price) for the first line item. This MUST be provided if the LineItem Quantity has been provided. Note: Multiple LineItems can be included by adding additional variables and incrementing the index in the variable name (the index must be sequential with no gaps). eg: LineItem1Amount
LineItem0Description	A	100	The description for the first line item. This MUST be provided if the LineItem Quantity has been provided. Note: Multiple LineItems can be included by adding additional variables and incrementing the index in the variable name (the index must be sequential with no gaps). eg: LineItem1Description

LineItem0Quantity	N	15	The quantity for the first line item, either as an integer or a decimal. Note: Multiple LineItems can be included by adding additional variables and incrementing the index in the variable name (the index must be sequential with no gaps). eg: LineItem1Quantity
LineItemSalesTaxAmount	N	15	The total monetary amount of the sales tax for the line items.
LineItemSalesTaxDescription	A	50	The sales tax description for the amount specified in LineItemSalesTaxAmount.
MerchantID	A	15	The merchant ID that corresponds to the gateway account the transaction will be run through. NOTE: If this variable is not present, then the skinning of the payment form will not happen.
Message	A	512	The message returned by the Gateway, giving more information about the status of the transaction.
MethodData	A	512	A Base64URL encoded value that contains a unique identifier for the transaction used by the ACS/DS, and the FingerprintNotificationURL.
MethodURL	A	256	The URL to which the MethodData should be sent.
OrderDescription	A	256	A description for the order. Note: special characters must be properly escaped, otherwise the hash digest will not match.
OrderID	A	50	A merchant side ID for the order - primarily used for determining duplicate transactions. Note: special characters must be properly escaped, otherwise the hash digest will not match.
PhoneNumber	A	30	Customer's phone number.
PostCode	A	50	Customer's billing address post code.
PostCodeCheckResult	A	-	If requested (EchoAVSCheckResult=TRUE in initial request) this gives the result of the post code check - will be one of: <ul style="list-style-type: none"> <li>• PASSED</li> <li>• FAILED</li> <li>• PARTIAL</li> <li>• NOT_CHECKED</li> <li>• UNKNOWN</li> </ul>
PreviousMessage	A	512	If the transaction was deemed to be a duplicate transaction, this gives a more detailed description of the status of the previous transaction.
PreviousStatusCode	N	-	If the transaction was deemed to be a duplicate transaction, this indicates the status of the previous transaction (see Appendix 1).
PrimaryAccountDateOfBirth	DT	10	The date of birth of the primary account holder (MCC 6012 accounts only) in the form "YYYY-MM-DD". e.g. "1995-08-21"

PrimaryAccountName	A	100	The name of the primary account holder (MCC 6012 accounts only).
PrimaryAccountNumber	A	50	The account number of the primary account holder (MCC 6012 accounts only).
PrimaryAccountPostCode	A	50	The post code of the primary account holder (MCC 6012 accounts only).
ShippingAddress1	A	100	Shipping recipient's address line 1.
ShippingAddress2	A	50	Shipping recipient's address line 2.
ShippingAddress3	A	50	Shipping recipient's address line 3.
ShippingAddress4	A	50	Shipping recipient's address line 4.
ShippingCity	A	50	Shipping recipient's address city.
ShippingCountryCode	N	3	Shipping recipient's address country code (ISO 3166- 1). e.g. United Kingdom: 826
ShippingEmailAddress	A	100	Shipping recipient's address email address (should be a valid email address).
ShippingName	A	100	Shipping recipient's name.
ShippingPhoneNumber	A	30	Shipping recipient's address phone number.
ShippingPostCode	A	50	Shipping recipient's address post code.
ShippingState	A	50	Shipping recipient's address state.
State	A	50	Customer's billing address state.
StatusCode	N	-	The status of the transaction returned by the Gateway (see Appendix 1).
ThreeDSecureAuthenticationCheckResult	A	-	If requested (EchoThreeDSecureAuthenticationCheckResult =TRUE in initial request) this gives the result of the 3D Secure check - will be one of: <ul style="list-style-type: none"> <li>• PASSED</li> <li>• FAILED</li> <li>• NOT_CHECKED</li> <li>• UNKNOWN</li> </ul>
TransactionDateTime	DT	26	The date and time (as seen by the merchant's server) of the transaction, in the form "YYYY-MM-DD HH:MM:SS +00:00", with the time in 24-hour format, where 00:00 is the offset from UTC. e.g. "2008-12-01 14:12:00 +01:00"
TransactionType	A	-	Must be one of: <ul style="list-style-type: none"> <li>• SALE</li> <li>• PREAUTH</li> </ul>



## Initial Request

### Input Variables

The following table lists the form variables that should be included in the initial request sent to the Transparent Redirect API.

The initial request may return one of three responses; a "3D Secure Authentication Required - Fingerprint" response with MethodURL and MethodData fields, a "3D Secure Authentication Required - Challenge" response with ACSURL and CReq fields, or a "Payment Complete" response.

Field Name	Include In Form	Include In Hash
HashDigest	M	✗
MerchantID	M	✓
TransactionDateTime	M	✓
ClientReference	O	✓
TransactionType	M	✓
Amount	M	✓
CurrencyCode	M	✓
OrderID	M	✓
OrderDescription	O	✓
LineItemSalesTaxAmount	O	✓
LineItemSalesTaxDescription	O	✓
LineItem0Quantity	O	✓
LineItem0Amount	O	✓
LineItem0Description	O	✓
Address1	O	✗
Address2	O	✗
Address3	O	✗
Address4	O	✗
City	O	✗
State	O	✗
PostCode	O	✗
CountryCode	O	✗
EmailAddress	O	✗
PhoneNumber	O	✗
DateOfBirth	O	✗
ShippingName	O	✗
ShippingAddress1	O	✗
ShippingAddress2	O	✗
ShippingAddress3	O	✗

ShippingAddress4	O	✗
ShippingCity	O	✗
ShippingState	O	✗
ShippingPostCode	O	✗
ShippingCountryCode	O	✗
ShippingEmailAddress	O	✗
ShippingPhoneNumber	O	✗
PrimaryAccountName	O	✓
PrimaryAccountNumber	O	✓
PrimaryAccountDateOfBirth	O	✓
PrimaryAccountPostCode	O	✓
CardName	M	✗
CardNumber	M	✗
ExpiryDateMonth	M	✗
ExpiryDateYear	M	✗
IssueNumber	O	✗
CV2	O	✗
AVSOverridePolicy	O	✓
CV2OverridePolicy	O	✓
FingerprintNotificationURL	C (note 1)	✓
ChallengeNotificationURL	C (note 1)	✓
BrowserJavaScriptEnabled	C (note 1)	✗
BrowserJavaEnabled	C (note 1)	✗
BrowserLanguage	C (note 1)	✗
BrowserScreenHeight	C (note 1)	✗
BrowserScreenWidth	C (note 1)	✗
BrowserColorDepth	C (note 1)	✗
BrowserTimeZone	C (note 1)	✗
EchoCardType	O	✓
EchoCardNumberFirstSix	O	✓
EchoCardNumberLastFour	O	✓
EchoCardExpiryDate	O	✓
EchoAVSCheckResult	O	✓
EchoCV2CheckResult	O	✓
EchoThreeDSecureAuthenticationCheckResult	O	✓
EchoFraudProtectionCheckResult	O	✓
CallbackURL	M	✓

Notes:

1. These fields are mandatory for a transaction that should be authenticated with 3D Secure.

## 3D Secure Authentication Required - Fingerprint

### Output Variables

If the transaction requires 3D Secure Authentication, the first stage is to analyse ("fingerprint") the customer's browser (although in some circumstances this stage may be skipped). If Fingerprinting is required then the returned StatusCode will be 3, the Message will be "Issuer authentication required" and MethodURL and MethodData fields will be returned from the Transparent Redirect API.

The table below gives the full list of variables that will be posted back to the merchant's CallbackURL.

Field Name	Included In Form	Included In Hash
HashDigest	M	✗
MerchantID	M	✓
TransactionDateTime	M	✓
ClientReference	O	✓
CrossReference	M	✓
StatusCode	M	✓
Message	M	✓
OrderID	M	✓
MethodURL	M	✓
MethodData	M	✓

## 3D Secure - Environment Request

### Input Variables

The MethodData returned from the call to the MethodURL should be sent back to the Transparent Redirect API. This call to the Transparent Redirect API should include the variables listed below.

The Environment Request may return one of two responses; a "3D Secure Authentication Required - Challenge" response with ACSURL and CReq fields (a 3D Secure Challenge transaction), or a "Payment Complete" response (a 3D Secure Frictionless transaction).

Field Name	Include In Form	Include In Hash
HashDigest	M	✗
MerchantID	M	✓
TransactionDateTime	M	✓
ClientReference	O	✓
CrossReference	M	✓
MethodData	M	✓
CallbackURL	M	✓

## 3D Secure Authentication Required - Challenge

### Output Variables

If the transaction requires 3D Secure Authentication and the Fingerprinting has been completed (or skipped), the transaction processing may have determined that a Challenge is required to provide further authentication. If a Challenge is required then the returned StatusCode will be 3, the Message will be "Issuer authentication required" and ACSURL and CReq (Challenge Request) fields will be included.

The table below gives the full list of variables that will be posted back to the merchant's CallbackURL.

Field Name	Included In Form	Included In Hash
HashDigest	M	✗
MerchantID	M	✓
TransactionDateTime	M	✓
ClientReference	O	✓
CrossReference	M	✓
StatusCode	M	✓
Message	M	✓
OrderID	M	✓
ACSURL	M	✓
CReq	M	✓

## 3D Secure - Authentication Request

### Input Variables

The CRes returned from the call to the ACSURL should be sent back to the Transparent Redirect API. This call to the Transparent Redirect API should include the variables listed below.

The Authentication Request will return a "Payment Complete" response.

Field Name	Include In Form	Include In Hash
HashDigest	M	✗
MerchantID	M	✓
TransactionDateTime	M	✓
ClientReference	O	✓
CrossReference	M	✓
CRes	M	✓
CallbackURL	M	✓

## Payment Complete

### Output Variables

If the payment transaction is complete, then the returned StatusCode will be 0 and the Message will contain the Authorisation Code.

The table below gives the full list of variables that will be posted back to the merchant's CallbackURL.

Field Name	Included In Form	Included In Hash
HashDigest	M	✗
MerchantID	M	✓
TransactionDateTime	M	✓
ClientReference	O	✓
CrossReference	M	✓
TransactionType	M	✓
StatusCode	M	✓
Message	M	✓
PreviousStatusCode	O	✓
PreviousMessage	O	✓
Amount	M	✓
CurrencyCode	M	✓
OrderID	M	✓
OrderDescription	O	✓
LineItemSalesTaxAmount	O	✓
LineItemSalesTaxDescription	O	✓
LineItem0Quantity	C (note 1)	✓
LineItem0Amount	C (note 1)	✓
LineItem0Description	C (note 1)	✓
Address1	O	✓
Address2	O	✓
Address3	O	✓
Address4	O	✓
City	O	✓
State	O	✓
PostCode	O	✓
CountryCode	O	✓
EmailAddress	O	✓
PhoneNumber	O	✓
DateOfBirth	O	✓
ShippingName	O	✓



ShippingAddress1	0	✓
ShippingAddress2	0	✓
ShippingAddress3	0	✓
ShippingAddress4	0	✓
ShippingCity	0	✓
ShippingState	0	✓
ShippingPostCode	0	✓
ShippingCountryCode	0	✓
ShippingEmailAddress	0	✓
ShippingPhoneNumber	0	✓
PrimaryAccountName	0	✓
PrimaryAccountNumber	0	✓
PrimaryAccountDateOfBirth	0	✓
PrimaryAccountPostCode	0	✓
CardName	0	✓
CardType	0	✓
CardClass	0	✓
CardIssuer	0	✓
CardIssuerCountryCode	0	✓
CardNumberFirstSix	0	✓
CardNumberLastFour	0	✓
CardExpiryDate	0	✓
AddressNumericCheckResult	0	✓
PostCodeCheckResult	0	✓
CV2CheckResult	0	✓
ThreeDSecureAuthenticationCheckResult	0	✓
FraudProtectionCheckResult	0	✓

Notes:

- Line item fields will only be present if line item information was provided in the Initial Request. Each line item comprises a set of the three fields shown – if one is present all three must be. This set of fields will be repeated for each line item with the index number incrementing for each set.

## Appendix 1 - Transaction Status Codes

Status Code	Transaction Result	Description
0	Successful	<b>Transaction Authorised</b> The transaction was successful, and an Authorisation Code will be given as part of the message returned by the gateway.
3	Issuer Authentication Required	<b>Transaction Awaiting 3D Secure Authentication</b> The transaction is now awaiting 3D Secure Authentication. This status has a 2 hour expiry time set by the card scheme, at which point, the transaction will fail and it's status will automatically change to Issuer Authentication Expired.
4	Referred	<b>Transaction Referred</b> The card issuer has parked the transaction awaiting contact with the customer before proceeding to authorise or decline the transaction.
5	Declined	<b>Transaction Failed</b> The transaction was declined by the card issuer or acquiring bank. In the event of the Address or CV2 verification failure, this will also be noted on the message from the gateway (Example, "Card declined: AVS policy + CV2 policy"). If the message given by the gateway only says "Card declined" with no other information, then no other information was given to us from the card issuer or acquiring bank as to the underlying reason why. To find out why the transaction was declined the customer will have to contact their bank directly.
20 21	Duplicate Transaction	<b>Duplicate Transaction</b> The transaction was a duplicate of a transaction that has already been processed. If this is the case, then the original transaction information is also passed back from the gateway so you can determine the result of the original transaction.
30	Failed	<b>Transaction Failed – Error(s) Occurred</b> This is usually an indicator that the integration to the Transparent Redirect API is incomplete and/or not working correctly. There will usually be additional error information returned from the payment gateway to help determine the cause of the error.

## Appendix 2 - ACS Simulator

The test system comes complete with an ACS simulator that allows your developers to simulate the most common responses that might come back from the cardholder's bank's Access Control Server (ACS). Our test cards provide a range of test scenarios including a successful challenge and a challenge resulting in a declined transaction.

**PayVector ACS Simulator**



**Added Protection**  
This ACS simulates the behaviour of a production ACS during authenticated 3D Secure v2 challenge flow

Merchant Name: **IRC Showcase Account**  
Amount: **4.21 GBP**  
Transaction Date/Time (UTC): **19/01/2023 16:48:27**

## Appendix 3 - Example Form Data

Note: Values in square brackets in the following examples are not real data but represent values that have been removed for privacy/security.

### Initial Request

This is a sample transaction, showing the set of form variables from the HTML form that the merchant would submit to the Transparent Redirect API.

```
HashDigest=e1b2f2cadf7cbca2b9cf897f673a44ee7547be75
MerchantID=[YOUR_MERCHANT_ID]
TransactionDateTime=2023-01-20+12%3a05%3a11+%2b00%3a00
ClientReference=CR0379988898098828011687
TransactionType=PREAUTH
Amount=421
CurrencyCode=826
OrderID=TR%3a+20230120120511
OrderDescription=Transparent+Redirect+Showcase
LineItemSalesTaxAmount=20
LineItemSalesTaxDescription=VAT
LineItem0Quantity=1
LineItem0Amount=101
LineItem0Description=Description+of+first+item
LineItem1Quantity=2.75
LineItem1Amount=300
LineItem1Description=Description+of+second+item
Address1=5+Some+Road
City=Woking
State=Surrey
PostCode=GU23+4BH
CountryCode=826
DateOfBirth=1980-12-12
ShippingName=Ship+to+me
ShippingAddress1=Shipping+Address+1
ShippingAddress2=Shipping+Address+2
ShippingAddress3=Shipping+Address+3
ShippingAddress4=Shipping+Address+4
ShippingCity=Shipping+City
ShippingState=Shipping+State
ShippingPostCode=SH1P+2ME
ShippingCountryCode=826
ShippingEmailAddress=email%40shippingrecipient
ShippingPhoneNumber=12345678
PrimaryAccountName=Primary+Account+Name
PrimaryAccountNumber=12345
PrimaryAccountDateOfBirth=1983-11-12
PrimaryAccountPostCode=PR1+4CC
CardName=A+Cardholder
CardNumber=4000005066809071
ExpiryDateMonth=12
ExpiryDateYear=30
IssueNumber=
CV2=345
```

```

AVSOverridePolicy=NPPP
CV2OverridePolicy=FF
FingerprintNotificationURL=[URL_FOR_FINGERPRINT_RESPONSE]
ChallengeNotificationURL=[URL_FOR_CHALLENGE_RESPONSE]
JavaScriptEnabled=True
JavaEnabled=False
Language=en-US
ScreenHeight=1440
ScreenWidth=2560
ColorDepth=24
TimeZone=0
EchoCardType=true
EchoCardNumberFirstSix=True
EchoCardNumberLastFour=True
EchoCardExpiryDate=True
EchoAVSCheckResult=true
EchoCV2CheckResult=true
EchoThreeDSecureAuthenticationCheckResult=true
EchoFraudProtectionCheckResult=True
CallbackURL=[YOUR_CALLBACK_URL]

```

### 3D Secure Authentication Required – Fingerprint

This shows the set of form variables from the HTML form that would be returned to the merchant from the Transparent Redirect API for the fingerprint stage of a 3D Secure transaction.

```

MerchantID=[YOUR_MERCHANT_ID]
TransactionDateTime=2023-01-20+12%3a05%3a11+%2b00%3a00
ClientReference=CR0379988898098828011687
CrossReference=230120120932414100342984
StatusCode=3
Message=Issuer+authentication+required
OrderID=TR%3a+20230120120511
MethodURL=[ACS_URL]
MethodData=eyJ0aHJlZURTU2V0aG9kTm90aWZpY2F0aW9uVGVJMIjoiw1VSTF9GT1JfRk1OR0VSUFJJTlRfUkVTUE9OU0Vd
IiwidGhyZWVEU1NlcnZlclRyYW5zSUQiOiJkMjkzOTFhNC1mYzEwLTQ4YTQtYWY4OS0xOWRhZTRiNzc0ZTMifQ
HashDigest=8ce239ffbe7b0aa925471e4c9d53edd7c37e01f5

```

### 3D Secure - Environment Request

This shows the set of form variables from the HTML form that the merchant would submit to the Transparent Redirect API for the Environment Request.

```

MerchantID=[YOUR_MERCHANT_ID]
TransactionDateTime=2023-01-20+12%3a11%3a33+%2b00%3a00
ClientReference=CR0379988898098828011687
CrossReference=230120120932414100342984
MethodData=eyJ0aHJlZURTU2VydMvYVHJhbnNJRCI6ImQyOTM5MWE0LWZjMTAtNDhhNC1hZjg5LTE5ZGF1NGI3NzRlMyJ9
CallbackURL=[YOUR_CALLBACK_URL]
HashDigest=5df16201eed33e02d31844761095bf1f5a48beb9

```

### 3D Secure Authentication Required – Challenge

This shows the set of form variables from the HTML form that would be returned to the merchant from the Transparent Redirect API for the challenge stage of a 3D Secure transaction.

```
MerchantID=[YOUR_MERCHANT_ID]
TransactionDateTime=2023-01-20+12%3a09%3a32+%2b00%3a00
ClientReference=CR0379988898098828011687
CrossReference=230120120932414100342984
StatusCode=3
Message=Additional+authentication+required
OrderID=TR%3a+20230120120511
ACSURL=[ACS_URL]
CReq=eyJhY3NUcmFuc0lEIjoizGM2M2MyZmItMjVlNC00MzNjLWI2MzktMjU0YTAYMzQwNGRlIiwiaWY2hhdGxlbmd1V2luZG
93U2l6ZSI6IjAxIiwibWVzc2FnZUV4dGVuc2lubiI6W3sibmFtZSI6IktF1dGhvcmlzYXRpb25TdGF0dXMiLCJpZCI6IjE
iLCJjcm10aW9uIj0eUluZG1jYXRvcil6dHJlZSwiZGF0YSI6IktFVVEhPUk1TRUQiFV0sIm1lc3NhZ2VUeXB1IjoizQ1j
cSI6Im1lc3NhZ2VWZXJzaW9uIjoizQ1jYyLjAiLCJ0aHJlZURTU2VydmVyVHJhbnNJRCI6ImQyOTM5MWE0LWZjMTAtNDhh
C1hZjg5LTE5ZGFINGI3NzRlMyJ9
HashDigest=e5499081ed41d2ee80e9bd90548d5c7b22694efe
```

### 3D Secure - Authentication Request

This shows the set of form variables from the HTML form that the merchant would submit to the Transparent Redirect API for the Authentication Request.

```
MerchantID=[YOUR_MERCHANT_ID]
TransactionDateTime=2023-01-20+12%3a13%3a48+%2b00%3a00
ClientReference=CR0379988898098828011687
CrossReference=230120120932414100342984
CRes=eyJjaGFsbGVuZ2VDb21wbGV0aW9uSW5kIjoizQ1jYyLjAiLCJ0cmFuc1N0YXR1cyI6IktFVVEhPUk1TRUQiFV0sIm1lc3NhZ2VWZXJzaW9uIjoizQ1jYyLjAiLCJ0aHJlZURTU2VydmVyVHJhbnNJRCI6ImQyOTM5MWE0LWZjMTAtNDhhNC1hZjg5LTE5ZGFINGI3NzRlMyJ9
CallbackURL=[YOUR_CALLBACK_URL]
HashDigest=6f67c9d515d2b8ee59e7b9f957f1137ebb994549
```

## Payment Complete

This shows the set of form variables from the HTML form that would be returned to the merchant from the Transparent Redirect API when the transaction is complete.

```
MerchantID=[YOUR_MERCHANT_ID]
TransactionDateTime=2023-01-20+12%3a09%3a32+%2b00%3a00
ClientReference=CR0379988898098828011687
CrossReference=230120121442088100189901
TransactionType=PREAUTH
StatusCode=0
Message=AuthCode%3a+992883
Amount=421
CurrencyCode=826
OrderID=TR%3a+20230120120511
OrderDescription=Transparent+Redirect+Showcase
LineItemSalesTaxAmount=20
LineItemSalesTaxDescription=VAT
LineItem0Quantity=1
LineItem0Amount=101
LineItem0Description=Description+of+first+item
LineItem1Quantity=2.75
LineItem1Amount=300
LineItem1Description=Description+of+second+item
Address1=5+Some+Road
City=Woking
State=Surrey
PostCode=GU23+4BH
CountryCode=826
DateOfBirth=1980-12-12
ShippingName=Ship+to+me
ShippingAddress1=Shipping+Address+1
ShippingAddress2=Shipping+Address+2
ShippingAddress3=Shipping+Address+3
ShippingAddress4=Shipping+Address+4
ShippingCity=Shipoping+City
ShippingState=Shipping+State
ShippingPostCode=SH1P+2ME
ShippingCountryCode=826
ShippingEmailAddress=email%40shippingrecipient
ShippingPhoneNumber=12345678
PrimaryAccountName=Primary+Account+Name
PrimaryAccountNumber=12345
PrimaryAccountDateOfBirth=1983-11-12
PrimaryAccountPostCode=PR1+4CC
CardName=A+Cardholder
CardType=VISA
CardClass=UNKNOWN
CardNumberFirstSix=400000
CardNumberLastFour=9071
CardExpiryDate=12%2f30
AddressNumericCheckResult=PASSED
PostCodeCheckResult=PASSED
CV2CheckResult=PASSED
ThreeDSecureAuthenticationCheckResult=PASSED
```

FraudProtectionCheckResult=NOT\_SUBMITTED  
HashDigest=26aa75cfda00a3fa6eb39e7f46ee5aebde422024



## Appendix 4 - Override Policy Codes & Explanations

### OverrideAVSPolicy Codes

The OverrideAVSPolicy is a 4-character code that controls how the gateway handles the AVS (Address Verification Service) check for the transaction.

The first character determines the behaviour when one or both of the address numeric and post code checks is known.

The second and third characters determine the behaviour when dealing with partial matches – this is where either the address numeric check or the post code check returns a partial match.

The fourth character determines the behaviour when neither of the results of the address numeric check or the post code check is known.

#### Character 1

Character Code	Explanation
E	Fail the transaction if either the address numeric check or the post code check fails.
B	Fail the transaction only if both the address numeric check and the post code check fail.
A	Fail the transaction only if the address numeric check fails.
P	Fail the transaction only if the post code check fails.
N	Pass the transaction even if both checks fail.

#### Character 2

Character Code	Explanation
P	Treat a partial address numeric match as a pass.
F	Treat a partial address numeric match as a failure.

#### Character 3

Character Code	Explanation
P	Treat a partial post code match as a pass.
F	Treat a partial post code match as a failure.

#### Character 4

Character Code	Explanation
P	Pass the transaction if both results of the AVS check are unknown.
F	Fail the transaction if both results of the AVS check are unknown.

## Examples

**EEEE** – This is the strongest policy. The transaction will only pass if both the address numeric and post code checks pass. Partial matches are treated as failures.

**EPFP** – This policy means that the transaction will only pass if both the address numeric and post code checks pass, or if the results of both checks are unknown. A partial address numeric match is treated as a pass, but a partial post code match is treated as a failure.

**BPPF** – This policy means that the transaction will fail if both the address numeric and post code checks fail, or if the results of both are unknown. Both address numeric and post code partial results are treated as passes.

**NPPF** – This policy means that the transaction will pass even if both the address numeric and post code checks fail but fail if the results of both checks are unknown - not a recommended policy! Both address numeric and post code partial results are treated as passes.

**NPPP** – This is the weakest policy. The transaction will pass regardless of the results of the address numeric and post code checks. Both address numeric and post code partial results are treated as passes.

## Notes

The following reasons would cause the results of the AVS check to be unknown:

1. The cardholder's address information has not been provided to the Transparent Redirect API. The address numeric check is carried out across the Address1, Address1, Address3, Address4, City, and State fields – if none of them is present the state of the address numeric check will be unknown. Similarly, the post code check is carried out on the PostCode field and if that is not present the state of the post code check will be unknown.
2. If the transaction is a cross reference transaction and the cardholder's address information has not been provided with this transaction, the result will be based on the information in the original (cross-referenced) transaction. This in turn may elicit an unknown result for the AVS check.
3. If there was a problem contacting the provider, or the provider itself had a problem delivering the results of the AVS check (least likely reason).

## OverrideCV2Policy Codes

The OverrideCV2Policy is a 2-character code that controls how the gateway handles the CV2 (CVV) check for the transaction.

The first character determines the behaviour when the result of the CV2 check is known.

The second character determines the behaviour when the result of the CV2 check is unknown.

### Character 1

Character Code	Explanation
P	Pass the transaction if the CV2 check fails.
F	Fail the transaction if the CV2 check fails.

### Character 2

Character Code	Explanation
P	Pass the transaction if the result of the CV2 check is unknown.
F	Fail the transaction if the result of the CV2 check is unknown.

## Examples

**FF** – This is the strongest policy. The transaction will only pass if the CV2 check passes.

**FP** – This policy means that the transaction will pass if the CV2 check passes, or if the result of the CV2 check is unknown.

**PF** – This policy means that the transaction will pass if the CV2 check fails but fail if the result of the CV2 check is unknown - not a recommended policy!

**PP** – This is the weakest policy. The transaction will pass regardless of the result of the CV2 check.

## Notes

The following reasons would cause the result of the CV2 check be unknown:

1. The CV2 was not submitted with the transaction.
2. If the transaction is a cross reference transaction and the CV2 code has not been provided with this transaction (as an override), the result will be based on the information in the original (cross-referenced) transaction. This in turn may elicit an unknown result for the CV2 check.
3. If there was a problem contacting the provider, or the provider itself had a problem delivering the result of the CV2 check (least likely reason).

## Appendix 5 – Abbreviations

Abbreviation	Description
ACS	Access Control Server
API	Application Programming Interface
AReq	Authentication Request
ARes	Authentication Response
AVS	Address Verification Service
CReq	Challenge Request
CRes	Challenge Response
CV2	Card Verification Value (also known as CVV, CVC, CVD, CSC)
DS	Directory Server
HMAC	Hash-based Message Authentication Code
OTP	One-Time Password
PCI DSS	Payment Card Industry Data Security Standard
RReq	Results Request
RRes	Results Response

## Appendix 6 – Country Codes (ISO 3166-1)

ISO Code	Country
826	United Kingdom
840	United States
036	Australia
004	Afghanistan
248	Åland Islands
008	Albania
012	Algeria
016	American Samoa
020	Andorra
024	Angola
660	Anguilla
010	Antarctica
028	Antigua and Barbuda
032	Argentina
051	Armenia
533	Aruba
036	Australia
040	Austria
031	Azerbaijan
044	Bahamas
048	Bahrain
050	Bangladesh
052	Barbados
112	Belarus
056	Belgium
084	Belize
204	Benin
060	Bermuda
064	Bhutan
068	Bolivia
070	Bosnia and Herzegovina
072	Botswana
074	Bouvet Island
076	Brazil
086	British Indian Ocean Territory
096	Brunei Darussalam
100	Bulgaria

854	Burkina Faso
108	Burundi
116	Cambodia
120	Cameroon
124	Canada
132	Cape Verde
136	Cayman Islands
140	Central African Republic
148	Chad
152	Chile
156	China
162	Christmas Island
166	Cocos (Keeling) Islands
170	Colombia
174	Comoros
178	Congo
180	Congo, Democratic Republic of the
184	Cook Islands
188	Costa Rica
384	Côte d'Ivoire
191	Croatia
192	Cuba
196	Cyprus
203	Czech Republic
208	Denmark
262	Djibouti
212	Dominica
214	Dominican Republic
218	Ecuador
818	Egypt
222	El Salvador
226	Equatorial Guinea
232	Eritrea
233	Estonia
231	Ethiopia
238	Falkland Islands (Malvinas)
234	Faroe Islands
242	Fiji
246	Finland
250	France
254	French Guiana

258	French Polynesia
260	French Southern Territories
266	Gabon
270	Gambia
268	Georgia
276	Germany
288	Ghana
292	Gibraltar
300	Greece
304	Greenland
308	Grenada
312	Guadeloupe
316	Guam
320	Guatemala
831	Guernsey
324	Guinea
624	Guinea-Bissau
328	Guyana
332	Haiti
334	Heard Island and McDonald Islands
336	Holy See (Vatican City State)
340	Honduras
344	Hong Kong
348	Hungary
352	Iceland
356	India
360	Indonesia
364	Iran, Islamic Republic of
368	Iraq
372	Ireland
833	Isle of Man
376	Israel
380	Italy
388	Jamaica
392	Japan
832	Jersey
400	Jordan
398	Kazakhstan
404	Kenya
296	Kiribati
408	Korea, Democratic People's Republic of

410	Korea, Republic of
414	Kuwait
417	Kyrgyzstan
418	Lao People's Democratic Republic
428	Latvia
422	Lebanon
426	Lesotho
430	Liberia
434	Libyan Arab Jamahiriya
438	Liechtenstein
440	Lithuania
442	Luxembourg
446	Macao
807	Macedonia, the former Yugoslav Republic of
450	Madagascar
454	Malawi
458	Malaysia
462	Maldives
466	Mali
470	Malta
584	Marshall Islands
474	Martinique
478	Mauritania
480	Mauritius
175	Mayotte
484	Mexico
583	Micronesia, Federated States of
498	Moldova
492	Monaco
496	Mongolia
499	Montenegro
500	Montserrat
504	Morocco
508	Mozambique
104	Myanmar
516	Namibia
520	Nauru
524	Nepal
528	Netherlands
530	Netherlands Antilles
540	New Caledonia



554	New Zealand
558	Nicaragua
562	Niger
566	Nigeria
570	Niue
574	Norfolk Island
580	Northern Mariana Islands
578	Norway
512	Oman
586	Pakistan
585	Palau
275	Palestinian Territory, Occupied
591	Panama
598	Papua New Guinea
600	Paraguay
604	Peru
608	Philippines
612	Pitcairn
616	Poland
620	Portugal
630	Puerto Rico
634	Qatar
638	Reunion Réunion
642	Romania
643	Russian Federation
646	Rwanda
652	Saint Barthélemy
654	Saint Helena
659	Saint Kitts and Nevis
662	Saint Lucia
663	Saint Martin (French part)
666	Saint Pierre and Miquelon
670	Saint Vincent and the Grenadines
882	Samoa
674	San Marino
678	Sao Tome and Principe
682	Saudi Arabia
686	Senegal
688	Serbia
690	Seychelles
694	Sierra Leone

702	Singapore
703	Slovakia
705	Slovenia
90	Solomon Islands
706	Somalia
710	South Africa
239	South Georgia and the South Sandwich Islands
724	Spain
144	Sri Lanka
736	Sudan
740	Suriname
744	Svalbard and Jan Mayen
748	Swaziland
752	Sweden
756	Switzerland
760	Syrian Arab Republic
158	Taiwan, Province of China
762	Tajikistan
834	Tanzania, United Republic of
764	Thailand
626	Timor-Leste
768	Togo
772	Tokelau
776	Tonga
780	Trinidad and Tobago
788	Tunisia
792	Turkey
795	Turkmenistan
796	Turks and Caicos Islands
798	Tuvalu
800	Uganda
804	Ukraine
784	United Arab Emirates
826	United Kingdom
840	United States
581	United States Minor Outlying Islands
858	Uruguay
860	Uzbekistan
548	Vanuatu
862	Venezuela
704	Viet Nam

92	Virgin Islands, British
850	Virgin Islands, U.S.
876	Wallis and Futuna
732	Western Sahara
887	Yemen
894	Zambia
716	Zimbabwe

## Appendix 7 - Currency Codes (ISO 4217)

ISO Code	Currency
826	Pound Sterling
840	US Dollar
978	Euro
971	Afghani
12	Algerian Dinar
32	Argentine Peso
51	Armenian Dram
533	Aruban Guilder
36	Australian Dollar
944	Azerbaijani Manat
44	Bahamian Dollar
48	Bahraini Dinar
764	Baht
590	Balboa
50	Bangladeshi Taka
52	Barbados Dollar
974	Belarusian Ruble
84	Belize Dollar
60	Bermudian Dollar
984	Bolivian Mvdol (Funds code)
68	Boliviano
986	Brazilian Real
96	Brunei Dollar
975	Bulgarian Lev
108	Burundian Franc
124	Canadian Dollar
132	Cape Verde Escudo
136	Cayman Islands Dollar
288	Cedi
952	CFA Franc BCEAO
950	CFA Franc BEAC
953	CFP franc
152	Chilean Peso
963	Code reserved for testing purposes
170	Colombian Peso
174	Comoro Franc
977	Convertible Marks

558	Cordoba Oro
188	Costa Rican Colon
191	Croatian Kuna
192	Cuban Peso
196	Cyprus Pound
203	Czech Koruna
270	Dalasi
208	Danish Krone
807	Denar
262	Djibouti Franc
678	Dobra
214	Dominican Peso
951	East Caribbean Dollar
818	Egyptian Pound
230	Ethiopian Birr
978	Euro
955	European Composite Unit (EURCO)
956	European Monetary Unit
958	European Unit of Account 17 (E.U.A.-17)
957	European Unit of Account 9 (E.U.A.-9)
238	Falkland Islands Pound
242	Fiji Dollar
348	Forint
976	Franc Congolais
292	Gibraltar pound
959	Gold (one Troy ounce)
600	Guarani
324	Guinea Franc
328	Guyana Dollar
332	Haiti Gourde
344	Hong Kong Dollar
980	Hryvnia
352	Iceland Krona
356	Indian Rupee
364	Iranian Rial
368	Iraqi Dinar
388	Jamaican Dollar
392	Japanese yen
400	Jordanian Dinar
404	Kenyan Shilling
598	Kina

418	Kip
233	Kroon
414	Kuwaiti Dinar
894	Kwacha
454	Kwacha
973	Kwanza
104	Kyat
981	Lari
428	Latvian Lats
422	Lebanese Pound
8	Lek
340	Lempira
694	Leone
430	Liberian Dollar
434	Libyan Dinar
748	Lilangeni
440	Lithuanian Litas
426	Loti
969	Malagasy Ariary
458	Malaysian Ringgit
470	Maltese Lira
795	Manat
480	Mauritius Rupee
943	Metical
484	Mexican Peso
979	Mexican Unidad de Inversion (UDI)
498	Moldovan Leu
504	Moroccan Dirham
566	Naira
232	Nakfa
516	Namibian Dollar
524	Nepalese Rupee
532	Netherlands Antillian Guilder
376	New Israeli Shekel
901	New Taiwan Dollar
949	New Turkish Lira
554	New Zealand Dollar
64	Ngultrum
999	No currency
408	North Korean Won
578	Norwegian Krone

604	Nuevo Sol
478	Ouguiya
776	Pa'anga
586	Pakistan Rupee
964	Palladium (one Troy ounce)
446	Pataca
858	Peso Uruguayo
608	Philippine Peso
962	Platinum (one Troy ounce)
826	Pound Sterling
72	Pula
634	Qatari Rial
320	Quetzal
512	Rial Omani
116	Riel
642	Romanian Leu
946	Romanian New Leu
462	Rufiyaa
360	Rupiah
643	Russian Ruble
646	Rwanda Franc
654	Saint Helena Pound
882	Samoan Tala
682	Saudi Riyal
941	Serbian Dinar
690	Seychelles Rupee
961	Silver (one Troy ounce)
702	Singapore Dollar
703	Slovak Koruna
90	Solomon Islands Dollar
417	Som
706	Somali Shilling
972	Somoni
710	South African Rand
410	South Korean Won
960	Special Drawing Rights
144	Sri Lanka Rupee
938	Sudanese Pound
968	Surinam Dollar
752	Swedish Krona
756	Swiss Franc

760	Syrian Pound
834	Tanzanian Shilling
398	Tenge
780	Trinidad and Tobago Dollar
496	Tugrik
788	Tunisian Dinar
800	Uganda Shilling
970	Unidad de Valor Real
990	Unidades de formento
784	United Arab Emirates dirham
840	United States Dollar
860	Uzbekistan Som
548	Vatu
862	Venezuelan bolívar
704	Vietnamese đồng
947	WIR Euro
948	WIR Franc
886	Yemeni Rial
156	Yuan Renminbi
716	Zimbabwe Dollar
985	Zloty
997	No currency
998	No currency