

## Inter-Entity Payment Protocol (IPP)

The Inter-Entity Payment Protocol (IPP) facilitates adhoc interactions between independent accounting systems on the web. It is a server-to-server protocol.

### 1 Brands, Accountants, and Manifest Declaration

The entities in a transaction are represented as domain names, or more accurately, as [subdomain.]domain[/path]. These entities will be referred to as **brands** in this document. Each entity must be HTTP-accessible and must declare an ipp manifest URL in a metatag in the header section of the HTML page that is found at the domain. Example:

```
<meta name="ipp_manifest" content="http://tyaga.org/manifest.php?brand=tyaga.org" />
```

A brand is represented by one or more Accountants as declared in its manifest.

### 2 Manifest

This serves as a discovery gateway for gathering the necessary information to complete auditable, inter-entity transactions in real time. It is primarily a list of URLs that point to locations of more detailed information resources. For efficiency, some of the information such as public key certificates may be cached but with careful consideration of policies such as those related to expiration dates and revocation lists.

**keyCert** – (*optional*) URLs in this section points to a X.509 or PGP certificate copy, or a keyserver. This information is optional in version 0.1 as an entity may opt to use the simpler query verification method instead of the more complex asymmetric encryption method.

**Accountants** – URLs in this section points to service providers that understand the messaging semantics for processing IPP payments and completing a transaction in real time, most likely on behalf of another entity.

**Reports** – (*optional*) URLs in this section refers to portals for auditors to search and get records.

### 3 Payment States

The main categories of payment state are Offer, Pending, Decided, and Error (see Section 9 for the codes corresponding to each state). From the initial state of an Offer, a payment may transition into a Pending state or one of the Decided states (Approved As-Is, Reset or Voided.) An approved or reset payment should create a reportable transaction between two entities.

If a payment has been approved as-is or reset, the payee may refund the amount paid, in part or in whole, by initiating a Reversal Offer. A reversal offer is indicated by a negative payment amount, which effectively reverses the debiting and crediting of the payment endpoints.

#### 4 Transaction Flow

The main steps in the communication of a payment state between two Accountants are Notification, Verification, Response and Reaction.

- a. Notification refers to the communication of a payment offer or decision (see Section 5.) If an accountant does not have current cached manifest information for a co-transactor, then the IPP manifest must be fetched as described in Sections 1 and 2.
- b. Verification refers to a particular procedure for ascertaining the validity of a payment offer or decision (see Section 6.) Manifest information may have to be fetched for the same reason as in 4a.
- c. Response refers to the Verifier reply to the Notifier. Depending on the results of the Verification step, the Verifier may create a new payment record, transition it to a new state, or leave the payment in its current state (see Section 7.)
- d. Reaction refers to the Notifier's action based on the Verifier's Response. For example, the Notifier should roll-back a processed payment in case of a voidance. An Error status could prompt the Notifier to retry (see Section 8.)

#### 5 Notification

IPP notification is submitted as URL-encoded parameter values. The notification message semantics is between accounting systems and is invisible to the user.

Offer	Approved	Reset
ipp_version=0.1& ipp_code=600& ipp_status=Offer& ipp_relay=102-23-90&  ver_scheme=...& ver_data=...&  xtime=1234567890& amount=100.00& from=buyer.org& to=seller.com& memo=office+supply	ipp_version=0.1& ipp_code=620& ipp_status=Approved+As-Is&  ver_scheme=...& ver_data=...&  xtime=1234567890& amount=100.00& from=buyer.org& to=seller.com& memo=office+supply&	ipp_version=0.1& ipp_code=640& ipp_status=Approved+As-Is&  ver_scheme=...& ver_data=...&  xtime=1234567890& amount=80.00& from=buyer.org& to=seller.com& memo=office+supply&  ref_amount=100.00

### 5.1 IPP Parameters

**ipp\_version** – The only possible value right now is 0.1.

**ipp\_code** – Numeric. See Section 9.

**ipp\_status** – Human-readable text equivalent of the ipp\_code. See Section 9.

**ipp\_relay** – (*optional*) Any information, such as a one-time password, that is submitted by one transactor on behalf of the other transactor. This information is passed on to the accounting system of the “to” endpoint, most likely to be used for determining the particular account to credit and the seller to notify as part of the payment processing.

### 5.2 Verification Parameters

**ver\_scheme** – A URI that points to the verification procedure that was used to process the record. The URI should present human readable description of the verification steps that needs to be taken and any additional parameters that need to be provided as an ipp parameter. Example URI’s are given in Section 6.

**ver\_data** – (*optional*) The output of a ver\_scheme, if any. For example, a private-key encrypted record that would have to be decrypted with a public key to verify its source and integrity.

### 5.3 Record Parameters

**xtime** – unixtime, number of seconds since the Unix Epoch (January 1 1970 00:00:00 GMT).

**from** – the entity that will debit its account, i.e., subtract the amount from one of its accounts.

**to** – the entity that will credit its account, i.e., add the amount to one of its accounts.

**amount** – a positive or negative amount, with the latter representing a reversal of roles between the from and to endpoints.

**unit** – the accounting unit that will be used by both entities for reporting this particular payment

**memo** – optional information related to the payment, such as a tracking number or transactor notes

In addition to the above core record parameters, reference parameters are also needed in case of resets or reverse transactions. (Note that these are not needed in approvals or voidance as these parameters should match the original values exactly.)

**ref\_xtime** – The original record’s xtime; required in a reversal, optional in a reset transaction.

**ref\_amount** – The original record’s amount; required in a reversal or if the amount has been reset.

**ref\_memo** – The original record’s memo; optional in case of a reversal or reset and when the memo is different from the original.

## 6 Verification

IPP does not specify a particular scheme for verifying user credentials and payment authorizations. Instead, an accounting system must understand the implications of the `ver_scheme` parameter in order to properly conduct the verification process. A `ver_scheme` may define and require additional parameters to be submitted in the payment notification message.

Example Verification Schemes:

<http://tyaga.org/ipp/xotaur-sha1-query.html>,

<http://tyaga.org/ipp/xotaur-sha1-rsa.html>

## 7 Response (from Verifier)

An IPP response message has the following elements:

***ipp\_reply*** – wrapper xml tag (not needed in a json formatted response)

***code*** –numeric codes for finer-grained communication of the status, most applicable to errors (see Section 7)

***status*** –optional human-readable text equivalent of the code

***message*** – in case of an error, a description of its cause

***data*** – optional data in case a payment decision has been made immediately approved

### Examples

Pending	Approved	Reset
<pre>{ "code":610, "status":"Pending" }</pre>	<pre>{ "code":620, "status":"Approved As-Is", "ver_scheme": "...", "ver_data": "..." }</pre>	<pre>{ "code":640, "status":"Reset", "amount":80.00, "ver_scheme": "...", "ver_data": "...", "ref_amount": 100.00 }</pre>
Voided	Error	Error
<pre>{ "code":660, "status":"Voided", "ver_scheme": "...", "ver_data": "..." }</pre>	<pre>{ "code": 804, "status": "Verification Failed", "message": "Unable to verify." }</pre>	<pre>{ "code": 802, "status": "Invalid Credentials", "message": "User not authorized." }</pre>

## 8 Reaction (of Notifier)

In general, the notifier should mark its transaction record with the `ipp_response` status. Only mutually approved records should be included in audited currency activity reports. As much as possible, automated retries should be attempted for recoverable errors. For non-recoverable errors, the message sender should be notified of the error.

## 9 Codes

Code	Status	Explanation / Reaction
<i>600 – 699 are non-error codes</i>		
<b>600</b>	Payment Offer	Accountant should verify payment offer
<i>601 to 609</i>	<i>(to be specified)</i>	
<b>610</b>	Pending Decision	Accountant should expect a delayed notification of recipient decision
<i>611 to 619</i>	<i>(to be specified)</i>	
<b>620</b>	Approved As-Is	Accountant should mark payment as approved for reporting
<i>621 to 639</i>	<i>(to be specified)</i>	
<b>640</b>	Approved with Reset	Accountant should revise a payment record to include the modified xtime, amount, and/or memo
<i>641 to 659</i>	<i>(to be specified)</i>	
<b>660</b>	Voided	Accountant should release the funds back to the payer's account
<i>661</i>	Voided/On blocked list	
<i>662</i>	Voided/Per Advisory	
<i>663</i>	Voided/Unexpected Offer	
<i>664-699</i>	<i>(to be specified)</i>	
<i>700 – 799 indicate errors related to the IPP Manifest</i>		
701	Manifest Not Found/Invalid Format	
702	Manifest URL not Found/Invalid Declaration	
703	Ver_scheme dependency not found	
<i>800 – 899 indicate errors by the Notifier</i>		
800	Bad Request	Accountant should correct a malformed notification
801	Invalid Relay/User Not Found	Payer should make sure that the recipient's relay information is accurate, or assume unauthorized User
802	Invalid Credentials	The relay is valid but the user-credentials are invalid
804	Verification Failed	The verifier supports the ver_scheme and has invalidated the Notifier's ver_data

<b>Code</b>	<b>Status</b>	<b>Explanation / Reaction</b>
806	Reset not Acceptable	The decider's reset of the xtime, amount, and/or memo is not acceptable to the offerer
809	Conflict/Status	The verifier's cache of the transaction has a status that does not allow update to the Notifier's code
812	Conflict/Payment Units	The payer's unit does not match the receiving account's ledger units
<i>900 – 999 indicate errors by the Verifier</i>		
900	Server Error	A catch-all error code that should be clarified in a detailed error message
901	Ver_scheme not supported	The Notifier should use a different ver_scheme
907	Insufficient Funds	The verifier cannot complete an inflow transaction because of insufficient debits for cancellation (only applicable for offers)
908	Account not Authorized	The receiving account is not authorized to receive inflow
930	User not Authorized	The receiving account is not authorized to receive inflow