iATS Payments

ProcessLink Web Services Version 2

Document Version 2.0

Document Control

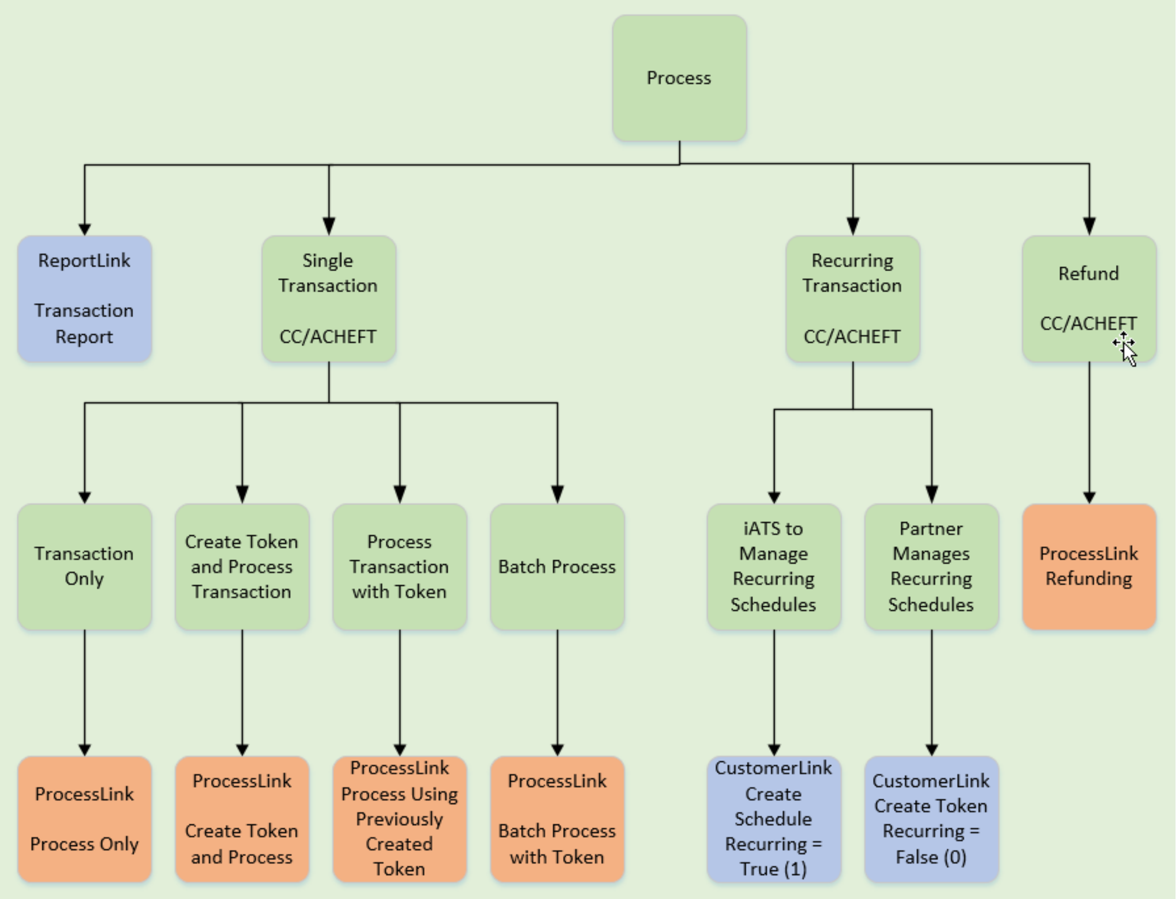
Document ID	Version	Author/Editor	Date
Updated for Batch Processing Changes	2.0	Dan Gill	2/27/2018

Table of Contents

1 Introduction

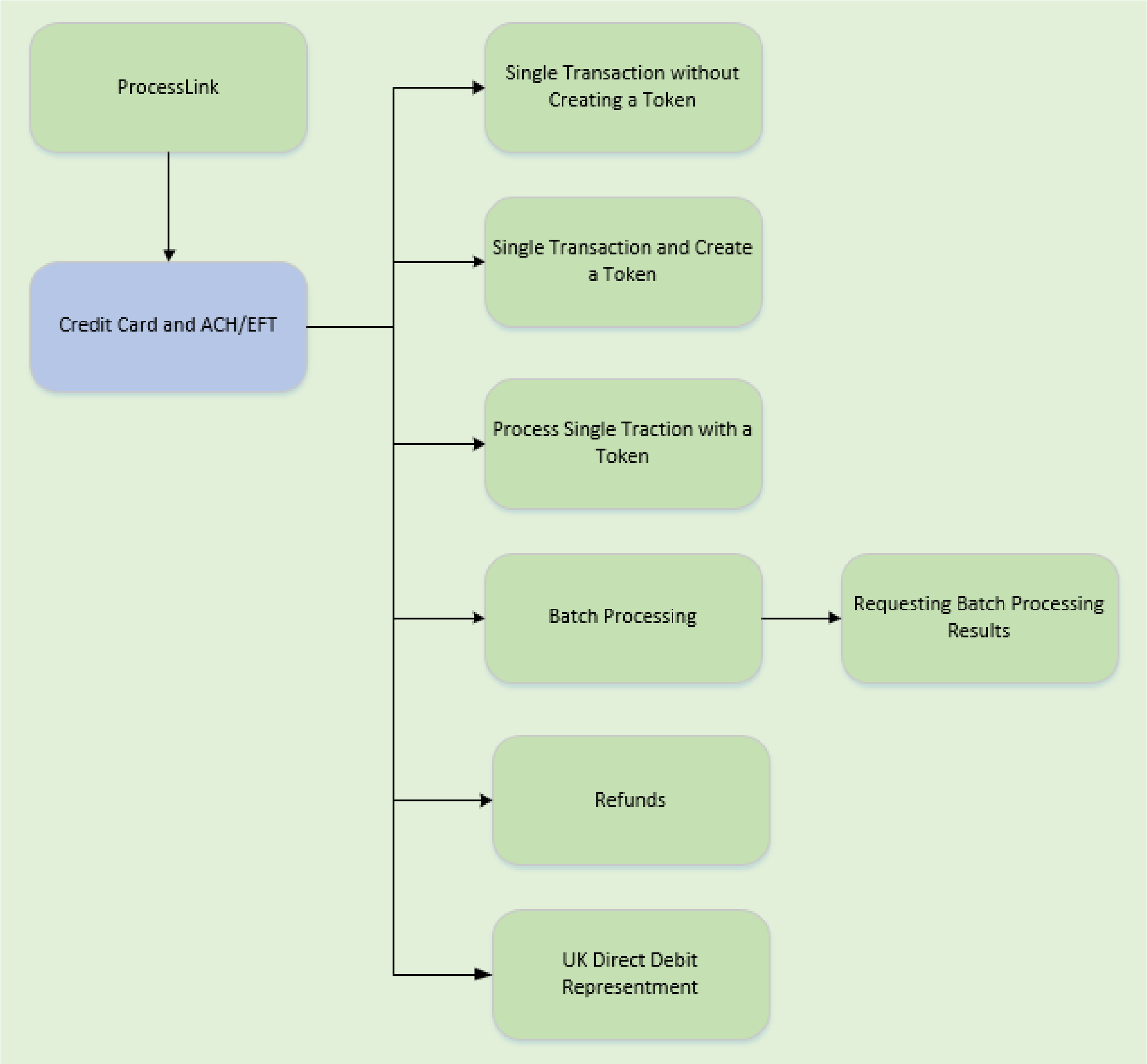
This guide provides an explanation of all the available iATS ProcessLink Web services, how they work, and how to set them up.

Below is a diagram on how ProcessLink fits with other web services from iATS (orange boxes)


Process
ReportLink Transaction Report
Single Transaction CC/ACHEFT
Recurring Transaction CC/ACHEFT
Refund CC/ACHEFT
Transaction Only
Create Token and Process Transaction
Process Transaction with Token
Batch Process
iATS to Manage Recurring Schedules
Partner Manages Recurring Schedules
ProcessLink Refunding
ProcessLink Process Only
ProcessLink Create Token and Process
ProcessLink Process Using Previously Created Token
ProcessLink Batch Process with Token
CustomerLink Create Schedule Recurring = True (1)
CustomerLink Create Token Recurring = False (0)

Please contact iATS directly for information on how to setup UK Direct Debit processing at webservice@iatspayments.com. This method of payment is highly regulated by BACS and has a number of best practises.

Functions available in ProcessLink that are explained in this guide:


ProcessLink
Credit Card and ACH/EFT
Single Transaction without Creating a Token
Single Transaction and Create a Token
Process Single Traction with a Token
Batch Processing
Requesting Batch Processing Results
Refunds
UK Direct Debit Representment

Available actions include:

- Processing ACH or Credit Card transactions
- Processing ACH or Credit Card transactions and tokenizing the payment data
- Processing ACH or Credit Card refund transactions
- Processing Batches of ACH or Credit Card transactions
- Processing UK Direct Debit re-presentments

North American API Directory: https://www.iatspayments.com/netgate/ProcessLinkv2.asmx

UK/Int'l API Directory: https://www.uk.iatspayments.com/netgate/ProcessLinkv2.asmx

2 1 How Do iATS Web Services Work?

2.1 How iATS Web Services Work

When clients sign up with iATS they are provided with a unique merchant account. With this account and once you set up the Web services you require, you will be able to process transactions directly from your donor database, website, or online fundraising solution (without having to negotiate with individual credit card companies and payment processing vendors).

Our Web services are standard SOAP messages that are supported by a server-side programming language. If your Web server is Microsoft Internet Information Server (MS IIS), you can use C# to write the server side code to use the Web service. If your server is PHP-based, you can write PHP code to use the Web service.

A SOAP message is an XML file that contains the following elements:

- An envelope element that identifies the XML document as a SOAP message
- A header element that contains header information
- A body element that contains call and response information
- A fault element containing errors and status information

For a full explanation of SOAP messages, please see https://www.w3schools.com/xml/xml_soap.asp.

A C# sample of an iATS consumer Web service is available for download here. In order to run the sample code, you will need Microsoft Visual Studio 2008 or later installed. The sample calls "GetCreditCardJournalCSVV1" method from our ReportLink.asmx Web service to get details (a journal) of a successful credit card transaction in csv type.

A PHP wrapper is available on our Github repository here. The base SOAP request and response structure for the specific services are explained below. To implement the services, you need to complete the code with your information and then integrate the code into your site or software platform. (Please note that all code will be hosted on your servers.) Once the calls are tested, you are good to go!

We have created a sample client code that allows you to test our payment systems. Please see, " Appendix A: Testing iATS Payments Systems” on page 66 for details on how to use it.

For technical assistance, please email webservice@iatspayments.com.

ProcessLink services are usually used in conjunction with our CustomerLink services (which are explained in the CustomerLink Web Services Guide).

For example, CustomerLink is primarily used for creating, updating, and deleting credit card and ACHEFT (ACH or direct debit) Tokens (Customer Codes) to setup recurring transactions. **ProcessLink, meanwhile, is for processing single and batch transactions and handling refunds. It is also possible to use it to create a Token (Customer Code) at the same time as doing a transaction if required.**

Creating Tokens (Customer Codes) allows future transactions to be carried out using the same credentials without the need to store sensitive credit card or ACH information on local servers.

For managing recurring transactions, you have the option of managing them or having iATS manage them for you:

1. **If iATS manages the recurring schedule:**
   a. Use CustomerLink to set up the recurring details (credit card, start/end date, etc.) with the **recurring** tag set to **true (1)**.
2. **If you manage the recurring schedule:**
   a. Use CustomerLink to set up the individual Tokens (Customer Codes) with the **recurring** tag set to **false (0)**.

   b. You then send a batch file containing the Tokens (Customer Codes) using ProcessLink;
   Or
   Pprocess a single transaction using either of the following ProcessLink services (depending on whether the transaction is using a credit card or ACHEFT): "ProcessCreditCardWithCustomerCodeV1" or "ProcessACHEFTWithCustomerCodeV1."

Please note that iATS operates two server systems, one based in North America (NA) and one in the United Kingdom (UK). Clients in the United States and Canada will use the NA servers, while all other clients use our UK servers. The explanation of the specific services that follows will provide the links to each server address as available.

During single transaction processing calls, if a Customer Code Token is not being used, the default placeholder value is "Quick Client". Please do not remove this default value during processing. When needed, this value should be replaced with the Customer Code token storing the payment details.

Encrypted Swipe credit card functionality is available. For more information or a copy of the installation guide, please email webservice@iatspayments.com.

2.2 Guide to the ProcessLink Web Services

The guide that follows includes a complete overview of the requests available using our ProcessLink Web services. Each request overview includes the following:

- Explanation of the specific Web service.
- NA and UK server addresses for the Web service.
- Example of the SOAP request and response structure.
- Overview of the parameters included in the service. Please note:
  - ➢ String type parameters in bold should have actual values.
  - ➢ Though some string type parameters are not required, it is best practice to include all the parameters in the same sequence when sending their request to iATS web service server. Leave the value blank if it is not required (Example: <customerIPAddress></customerIPAddress>).
  - ➢ Parameters with Boolean or DateTime as data types should always have the actual values and never leave these blank (Example: give 0 as default value of Boolean parameters if not required).
  - ➢ The XML element value will be "string" (that is, letters or numerals) unless otherwise noted.

- The tag name in the XML/SOAP message format is case sensitive. For example, agentCode will work, but agentcode will not.
- Character limits for string values will be noted where applicable.
- Characters values are all case sensitive and should be entered as shown in the call unless otherwise specified.
- For setting the time, our servers are in the following time zones:
  - North America server: PST
  - UK server: GMT
- For setting the date, both the North America and UK WebServices utilize standard xml Date Time type as the parameter. Please use the ISO 8601 format, for example, 2008-10-31T15:07:38.6875000 ("yyyy-MM-dd'T'HH:mm:ss.fffffffZ") wherever the date should be included in your code.
- **iATS utilizes maintenance/update windows of 00:00 to 01:00 in both Pacific and GMT each day.** While downtime, if any, is minimal, we highly discourage sending batch processing transactions during this time.
- Daily cut-off times are aligned with those of our processing banks and are as follows:
  - 2am PT/5am ET for US Merchants
  - 11pm PT/2am ET for Canadian Merchants
  - 12 Midnight GMT for all UK and International Merchants
- Values requiring a True/False response should be entered as 1/0 within the code. Note that if left blank or NULL is used, we will default to True (1).
- Where possible, it's recommended to prevent the use of comma's (,) within the dollar amount fields. iATS reads a comma as a period in this field to compensate for different countries use of the comma within currency. For example: $1,000 should be not allowed.

- Explanation of return values to expect in typical responses. Please note:
  - ACH transactions are not processed in real time. When an ACH transaction is submitted, iATS provides a general response of OK:555555, which indicates iATS has received the transaction attempt and that the request is pending approval.

3 Calls to Process Credit Card Transactions

3.1 Process a Single Credit Card Transaction

This service is for when you want to process a onetime credit card transaction.

- ProcessCreditCard

This call is available on either our NA or UK system.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- UK/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of “ProcessCreditCard”.

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., “string”, “dateTime,” etc.) need to be replaced with actual values.

Request (NA server; please use link above for UK server)

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessCreditCard xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <invoiceNum>string</invoiceNum>
      <creditCardNum>string</creditCardNum>
      <creditCardExpiry>string</creditCardExpiry>
      <cvv2>string</cvv2>
      <mop>string</mop>
      <firstName>string</firstName>
      <lastName>string</lastName>
      <address>string</address>
      <city>string</city>
      <state>string</state>
      <zipCode>string</zipCode>
      <total>string</total>
      <comment>string</comment>
      <title>string</title>
      <phone>string</phone>
      <phone2>string</phone2>
      <fax>string</fax>
      <email>string</email>
      <country>string</country>
      <item1>string</item1>
      <item2>string</item2>
      <item3>string</item3>
      <item4>string</item4>
      <item5>string</item5>
      <item6>string</item6>
    </ProcessCreditCard>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessCreditCardResponse xmlns="https://www.iatspayments.com/NetGate/">
      <ProcessCreditCardResult>xml</ProcessCreditCardResult>
    </ProcessCreditCardResponse>
  </soap12:Body>
</soap12:Envelope>
```

3.1.1 Overview of Request Parameters

Here is an overview of the request parameters for the “ProcessCreditCard,” above. Parameters in **bold** are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	This is the IP address of the donor’s computer. Please send to fully utilise the fraud tools available from iATS. IPv4 only.	N/A
invoiceNum		40
creditCardNum		30 digits
creditCardExpiry	“MM/YY”	5
cvv2	**Optional but highly recommended**	10
mop	NA: VISA, MC, AMX, DSC UK: VISA, MC, AMX, MAESTR	N/A
firstName		30
lastName	**Optional but highly recommended**	30
address	**Optional but highly recommended**	40
city	**Optional but highly recommended**	30
state	State or province (NA only). **Optional but highly recommended**	30
zipCode	**Optional but highly recommended**	19
total	Prevent the use of commas (,) within the dollar amount fields if possible. iATS reads a comma as a period in this field to compensate for different countries use of the comma within currency. For example: $1,000	N/A
comment		110
title		
phone		
phone2	Mobile	
fax		
email		
country		
item1		
item2		
item3		
item4		
item5		
item6		

Sample Reply Message Format

```
<IATSRESPONSE>
    <STATUS>StatusType</STATUS>
    <ERRORS>ErrorCodeType</ERRORS>
    <PROCESSRESULT>
            <AUTHORIZATIONRESULT>AuthorizationResultType
            </AUTHORIZATIONRESULT>
            <CUSTOMERCODE>CustomerCode</CUSTOMERCODE>
            <SETTLEMENTBATCHDATE></SETTLEMENTBATCHDATE>
            <SETTLEMENTDATE></SETTLEMENTDATE>
            <TRANSACTIONID>TransactionId</TRANSACTIONID>
    </PROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values

Return Value	Occurrence	Explanation
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc. *Reject 3: Customer Code does not exist* *Or Account num is too long* *Reject 40: Invalid card number*
CUSTOMERCODE	Never	No Token (Customer Code) is created for this process.
TRANSACTIONID	Conditional	Return information when Status = Success; and the transaction has been processed; value is transactionID.

3.2 Create a Token (Customer Code) and Charge Credit Card

This service is for when you want to create a credit card Token (Customer Code) for payment storage purposes only and to process a onetime credit card transaction at the same time.

- **CreateCustomerCodeAndProcessCreditCard**

This call is available on either our NA or UK system.

The Token (Customer Code) can only be used to process single transactions in the future unless you update the recurring field values separately using the Update API calls from CustomerLink. Encrypted Swipe credit card functionality is available. For more information or a copy of the installation guide, please email webservice@iatspayments.com.

- Please refer to the CustomerLink Guide or reach out to webservice@iatspayments.com for more details on best practises for creating a new Recurring Customer Code token.
- If you will be using the new token for a recurring schedule in the future, please avoid using any sub-codes ending in 90-99 (for example ABDC99) as these will affect future scheduled transactions. We recommend using an 80 level code (for example, ABDC80) for online webpage initiated schedules and a 01 level code (for example, ABCD01) for offline software initiated schedules.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- UK/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of “CreateCustomerCodeAndProcessCreditCard”.

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., “string”, “dateTime,” etc.) need to be replaced with actual values.

Request (NA server, please use UK link above for UK server)

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <CreateCustomerCodeAndProcessCreditCard xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <invoiceNum>string</invoiceNum>
      <ccNum>string</ccNum>
      <ccExp>string</ccExp>
      <firstName>string</firstName>
      <lastName>string</lastName>
      <address>string</address>
      <city>string</city>
      <state>string</state>
      <zipCode>string</zipCode>
      <cvv2>string</cvv2>
      <total>string</total>
      <comment>string</comment>
      <title>string</title>
      <phone>string</phone>
      <phone2>string</phone2>
      <fax>string</fax>
      <email>string</email>
      <country>string</country>
      <item1>string</item1>
      <item2>string</item2>
      <item3>string</item3>
      <item4>string</item4>
      <item5>string</item5>
      <item6>string</item6>
    </CreateCustomerCodeAndProcessCreditCard>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
   <CreateCustomerCodeAndProcessCreditCardResponse
xmlns="https://www.iatspayments.com/NetGate/">

<CreateCustomerCodeAndProcessCreditCardResult>xml</CreateCustomerCodeAndProcessCreditCardResult>
   </CreateCustomerCodeAndProcessCreditCardResponse>
  </soap12:Body>
```

```
</soap12:Envelope>
```

3.2.1 Overview of Request Parameters

Here is an overview of the request parameters for "CreateCustomerCodeAndProcessCreditCard," above. Parameters in **bold** are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	This is the IP address of the donor's computer. Please send to fully utilise the fraud tools available from iATS. IPv4 only.	N/A
invoiceNum		40
ccNum		32 digits
ccExp	MM/YY	10
firstName		100
lastName	**Optional but highly recommended**	100
address	**Optional but highly recommended**	100
city	**Optional but highly recommended**	40
state	**Optional but highly recommended.** State or province (North America only)	40
zipCode	**Optional but highly recommended**	40
cvv2		10
total		N/A
comment		
title		
phone		
phone2	Mobile	
fax		
email		
country		
item1		
item2		
item3		
item4		
item5		
item6		

Sample Reply Message Format

```
<IATSRESPONSE>
    <STATUS>StatusType</STATUS>
    <ERRORS>ErrorCodeType</ERRORS>
    <PROCESSRESULT>
            <AUTHORIZATIONRESULT>AuthorizationResultType
            </AUTHORIZATIONRESULT>
            <CUSTOMERCODE>CustomerCode</CUSTOMERCODE>
        <TRANSACTIONID>TransactionId</TRANSACTIONID>
    </PROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values		
Return Value	**Occurrence**	**Explanation**
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc. *Reject 3: Customer Code does not exist* *Or Account num is too long* *Reject 40: Invalid card number*
CUSTOMERCODE	Conditional	Return information when Status = Success and create Token (Customer Code) is successful; value is new customer code.
TRANSACTIONID	Conditional	Return information when Status = Success; and the transaction has been processed; value is transactionID.

3.3 Process a Credit Card Transaction Using an Existing Token (Customer Code)

This service is for when you want to charge a credit card transaction using a Customer Code Token that already exists.

- ProcessCreditCardWithCustomerCode

This call is available on either our NA or UK system.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- UK/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of "ProcessCreditCardWithCustomerCode".

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., "string", "dateTime," etc.) need to be replaced with actual values.

Request (NA server; please use UK link above for UK server)

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessCreditCardWithCustomerCode xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <customerCode>string</customerCode>
      <invoiceNum>string</invoiceNum>
      <cvv2>string</cvv2>
      <total>string</total>
      <comment>string</comment>
      <item1>string</item1>
      <item2>string</item2>
      <item3>string</item3>
      <item4>string</item4>
      <item5>string</item5>
      <item6>string</item6>
    </ProcessCreditCardWithCustomerCode>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessCreditCardWithCustomerCodeResponse
xmlns="https://www.iatspayments.com/NetGate/">

<ProcessCreditCardWithCustomerCodeResult>xml</ProcessCreditCardWithCustomerCodeResult>
    </ProcessCreditCardWithCustomerCodeResponse>
  </soap12:Body>
</soap12:Envelope>
```

3.3.1 Overview of Request Parameters

Here is an overview of the request parameters for “ProcessCreditCardWithCustomerCode,” above. Parameters in **bold** are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	This is the IP address of the donor’s computer. Please send to fully utilise the fraud tools available from iATS. IPv4 only.	N/A
customerCode	Existing Token (Customer Code)	40
invoiceNum		40
cvv2		10
total	Prevent the use of commas (,) within the dollar amount fields if possible. iATS reads a comma as a period in this field to compensate for different countries use of the comma within currency. For example: $1,000	N/A
comment		100
item1		
item2		
item3		
item4		
item5		
item6		

Sample Reply Message Format

```
<IATSRESPONSE>
    <STATUS>StatusType</STATUS>
    <ERRORS>ErrorCodeType</ERRORS>
    <PROCESSRESULT>
            <AUTHORIZATIONRESULT>AuthorizationResultType
            </AUTHORIZATIONRESULT>
            <CUSTOMERCODE>CustomerCode</CUSTOMERCODE>
            <TRANSACTIONID>TransactionId</TRANSACTIONID>
    </PROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values		
Return Value	**Occurrence**	**Explanation**
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc. *Reject 3: Customer Code does not exist* *Or Account num is too long* *Reject 40: Invalid card number*
CUSTOMERCODE	Never	Return information when Status = Success; value is inputted Token (Customer Code).
TRANSACTIONID	Conditional	Return information when Status = Success; and the transaction has been processed; value is transactionID.

3.4 Process Numerous Credit Card Transactions in a Batch

This service is for when you wish to process multiple credit card transactions (made on multiple credit cards) in one request. Your request will include a range of details for each credit card, as is described in detail below in the sections. Once uploaded, use the GetBatchProcessResultFile call details located in the section “Obtain the Results of a Credit Card Batch File” on page 23.

- ProcessCreditCardBatch

This call is available on either our NA or UK system.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- UK/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of “ProcessCreditCardBatch”.

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., “string”, “dateTime,” etc.) need to be replaced with actual values.

Request for NA system; for UK see link above:

```
POST /netgate/ProcessLinkv3.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
<?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>
<ProcessCreditCardBatchFile xmlns="https://www.iatspayments.com/NetGate/">
<agentCode>string</agentCode>
<password>string</password>
<customerIPAddress>string</customerIPAddress>
<batchFile>base64Binary</batchFile>
</ ProcessCreditCardBatchFile>
</soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
<?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>
<ProcessCreditCardBatchFileResponse xmlns="https://www.iatspayments.com/NetGate/">
<ProcessCreditCardBatchFileResult>xml</ProcessCreditCardBatchFileResult>
```

```
</ProcessCreditCardBatchFileResponse>
</soap12:Body>
</soap12:Envelope>
```

3.4.1 Overview of Request Parameters

Here is an overview of the request parameters for “ProcessCreditCardBatch,” above. Parameters in **bold** are mandatory.

Explanation of Sample Reply Message Format Values

Return Value	Occurrence	Explanation
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc. *Reject 3: Customer Code does not exist* *Or Account num is too long* *Reject 40: Invalid card number*
BATCHID	Conditional	Return information when Status = Success and batch file has been processed.
BATCHPROCESSRESULTFILE	Conditional	Return information when Status = Success and batch file processing has completed. Value will be base64Binary.

3.4.2 Explanation of base64Binary Format

Any binary text (“friendly” readable text) is converted or encoded into **Base64** format so that the batch file can be processed by the web service.

Web service does not understand binary data so it needs to convert the friendly language to something that web service can understand. We encode the binary data into characters, and Base64 is one of these types of encodings.

Example Base64 format:

```
MTAvMjgvMTEsIDAwMDAxLEpvZSxTbWl0aCwxMjMgQW55IFN0LEFueXRvd24sU1QsMTAwMjMsMj
UuMDAsQU1YLDMwMDAwMDAwMDAwMDAwMSwwNzA4DQoxMC8yOC8xMSwgMDAwMDEsSm9l
LFNtaXRoLDEyMyBBbnkgU3QsQW55dG93bixTVCwxMDAyMywyNS4wMCxBTVgsMzAwMDAwMDAw
MDAwMDAxLDA3MDgNCjEwLzI4LzExLCAwMDAwMSxKb2UsU21pdGgsMTIzIEFueSBTdCxBbnl0b3duL
FNULDEwMDIzLDI1LjAwLEFNWCwzMDAwMDAwMDAwMDAwMDEsMDcwOA0KDQo=
```

To produce these results, you will need to make the xml doc a binary object of type base64. Depending on the programming language you are using, you’ll use a Convert method to encode binary to base64 format. As shown in result above.

In C#, you can convert to base64 string using this:

```
Convert.ToBase64String(byteArray)
```

In XML message, put the base64 string between <batchFile> and </batchFile>

Example including formatting from **Request** section above--this is what the request should look like:

```
<?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>
<ProcessCreditCardBatchV1 xmlns="https://www.iatspayments.com/NetGate/">
<agentCode>string</agentCode>
<password>string</password>
<customerIPAddress>string</customerIPAddress>
<batchFile>
MTAvMjgvMTEsIDAwMDAxLEpvZSxTbWl0aCwxMjMgQW55IFN0LEFueXRvd24sU1QsMTAwMjMsMj
UuMDAsQU1YLDMwMDAwMDAwMDAwMDAwMSwwNzA4DQoxMC8yOC8xMSwgMDAwMDEsSm9l
LFNtaXRoLDEyMyBBbnkgU3QsQW55dG93bixTVCwxMDAyMywyNS4wMCxBTVgsMzAwMDAwMDAw
MDAwMDAxLDA3MDgNCjEwLzI4LzExLCAwMDAwMSxKb2UsU21pdGgsMTIzIEFueSBTdCxBbnl0b3duL
FNULDEwMDIzLDI1LjAwLEFNWCwzMDAwMDAwMDAwMDAwMDEsMDcwOA0KDQo=
</batchFile>
</ProcessCreditCardBatchV1>
</soap:Body>
</soap:Envelope>
```

3.4.3 Explanation of Batch File Types

- **Batch file format for U.S. and U.K credit cards**

Format:
Date, Invoice, First Name, Last Name, Street, City, State, Zip Code, Amount, MOP, Credit Card #, Expiry

Example:
10/28/11, 00001,Joe,Smith,123 Any St,Anytown,ST,10023,25.00,AMX,300000000000001,0708

- **Batch file format for Canadian credit cards**

Format:
Date, Invoice, Full Name, Amount, MOP, Credit Card #, Expiry

Example:
10/28/11, 00001,Joe Smith,25.00,AMX,300000000000001,0708

Notes:

- If any fields must be omitted (the invoice number for example), the comma separators must still be included. For example, 10/28/11,,Joe,Smith,123 Any St,Anytown,ST,10023,25.00,AMX,300000000000001,0708
- Credit card abbreviations must all be capitalized and in these formats:
- **NA**: VISA, MC, AMX, DSC
- **UK**: VISA, MC, AMX, MAESTR

- **Batch file format using Tokens (Customer Codes)**

Format:
Invoice Number, Token (Customer Code), Amount, Comment

Example:
a123467,A168786,1.10,comment1

Notes:

- This is for all currencies.
- If any fields must be omitted (the invoice number for example), the comma separators must still be included. For example. ,,A168786,1.10,comment1

3.4.4 Obtain the Results of a Credit Card Batch File

This service is for when you wish to look at the results (rejects and approvals) processing multiple credit card transactions (made on multiple credit cards) in one request from section “Process Numerous Credit Card Transactions in a Batch” on page 19.

- GetBatchProcessResultFile

This call is available on either our NA or UK system.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- UK/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of "GetBatchProcessResultFile".

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., "string", "dateTime," etc.) need to be replaced with actual values.

Request for NA system; for UK see link above:

```
POST /netgate/ProcessLinkv3.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
<?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>
<GetBatchFileProcessResultFile xmlns="https://www.iatspayments.com/NetGate/">
<agentCode>string</agentCode>
<password>string</password>
<customerIPAddress>string</customerIPAddress>
<batchId>string</batchId>
</GetBatchFileProcessResultFile>
</soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
<?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>
<GetBatchProcessResultFileResponse xmlns="https://www.iatspayments.com/NetGate/">
<GetBatchProcessResultFileResult>xml</GetBatchProcessResultFileResult>
</GetBatchProcessResultFileResponse>
```

```
</soap12:Body>
</soap12:Envelope>
```

3.4.5 Overview of Request Parameters

Here is an overview of the request parameters for “GetBatchProcessResultFile,” above. Parameters in **bold** are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	This is the IP address where the batch is sent from. IPv4 only.	N/A
batchId		

Sample Reply Message Format

```
<IATSRESPONSE>
<STATUS>StatusType</STATUS>
<ERRORS>ErrorCodeType</ERRORS>
<BATCHPROCESSRESULT >
<AUTHORIZATIONRESULT>AuthorizationResultType
</AUTHORIZATIONRESULT>
<BATCHID>BatchId</BATCHID>
<BATCHPROCESSRESULTFILE>BatchFile</BATCHPROCESSRESULTFILE>
</BATCHPROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values		
Return Value	**Occurrence**	**Explanation**
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc. *Reject 3: Customer Code does not exist* *Or Account num is too long* *Reject 40: Invalid card number*
BATCHID	Conditional	Return information when Status = Success and batch file has been processed.
BATCHPROCESSRESULTFILE	Conditional	Return information when Status = Success and batch file processing has completed. Value will be base64Binary.

Sample Reply Message Format

```
<IATSRESPONSE>
    <STATUS>StatusType</STATUS>
    <ERRORS>ErrorCodeType</ERRORS>
    <PROCESSRESULT>
            <AUTHORIZATIONRESULT>AuthorizationResultType
            </AUTHORIZATIONRESULT>
            <CUSTOMERCODE>CustomerCode</CUSTOMERCODE>
            <TRANSACTIONID>TransactionId</TRANSACTIONID>
    </PROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values		
Return Value	**Occurrence**	**Explanation**
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc. *Reject 3: Customer Code does not exist* *Or Account num is too long* *Reject 40: Invalid card number*
BATCHID	Conditional	Return information when Status = Success and batch file has been processed.
BATCHPROCESSRESULTFILE	Conditional	Return information when Status = Success and batch file processing has completed. Value will be base64Binary.

3.4.6 Explanation of Processed Batch File

Batch file results:

These messages are appended to the end of the request batch file:

- OK: + 6 digits
- REJECT: TIMEOUT
- WRONG FORMAT
- Duplicated
- …

Example request:

Example:
a123467,A168786,1.10,comment1
a0M28000007iG94EAE,A597362,1000,0062800000IatbEAAR

Example results:

a123467,A168786,1.10,comment1, OK:555555
a0M28000007iG94EAE,A597362,1000,0062800000IatbEAAR,OK:55555

4 Calls for Processing Bank Debit Transactions

Explanation of ACH Processing Stages

As ACH/EFT Transaction processing is not processed in “Real Time” like credit card transactions, there can be processing delays while the data is moving between iATS, our Bank and the donors Bank.

When ACH/EFT transactions are submitted to iATS, we will submit them to our processing bank as per the following schedule:

- Monday and Tuesday: File sent twice per day (3am and 3pm)

- Wednesday to Friday: File sent once per day (3pm)

When ACH/EFT transactions are sent to iATS for processing, there are three stages which each transaction will go through, even though you receive a successful ACH Transaction ID immediately:

1. Immediately following the submission, each transaction will be listed as “Tobesent” within iATS systems. This stage indicates the transaction is waiting to be sent to iATS’ bank as per above schedule.
2. Once the transactions have been sent to the bank, but before final approval/rejection has been received, each transaction will be listed as “Pending”. A response should be received within 1 business day.
3. Once a response has been received, each transaction will be listed as EITHER OK:BankAccept, or OK:Reject. These results can be pulled using our WebService ReportLink options if desired (Journal for approvals, ACH Reject for rejections)

All transactions with the result “OK:Bank Accept” will be sent to the Donor’s bank for final approval. At this point, the Donor or the Donor’s bank may refuse the debit, for example because of an account closure or insufficient funds. If this occurs, a separate iATS Return record is added to the iATS Journal reports (ACH Returns) and a notification email is sent to the client. These returns can be pulled using our WebService ReportLink options if desired.

4.1 Process a Single Bank Debit Transaction

This service is for when you want to process a onetime ACH Bank Debit transaction for USD, CDN, or AUD currency. This call is for our North American and AUD portion of the UK International system only. For UK Direct Debit or SEPA transactions, see the CustomerLink Guide.

- ProcessACHEFT

This call is only available on our NA, and AUD portion of our UK system.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- AUD/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of “ProcessACHEFT”.

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., “string”, “dateTime,” etc.) need to be replaced with actual values.

Request (NA server; for the UK server see link above)

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessACHEFT xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <invoiceNum>string</invoiceNum>
      <accountNum>string</accountNum>
      <accountType>string</accountType>
      <firstName>string</firstName>
      <lastName>string</lastName>
      <address>string</address>
      <city>string</city>
      <state>string</state>
      <zipCode>string</zipCode>
      <total>string</total>
      <comment>string</comment>
      <title>string</title>
      <phone>string</phone>
      <phone2>string</phone2>
      <fax>string</fax>
      <email>string</email>
      <country>string</country>
      <item1>string</item1>
      <item2>string</item2>
      <item3>string</item3>
      <item4>string</item4>
      <item5>string</item5>
      <item6>string</item6>
  </ProcessACHEFT>
 </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
  <ProcessACHEFTResponse xmlns="https://www.iatspayments.com/NetGate/">
   <ProcessACHEFTResult>xml</ProcessACHEFTResult>
  </ProcessACHEFTResponse>
 </soap12:Body>
</soap12:Envelope>
```

4.1.1 Overview of Request Parameters

Here is an overview of the request parameters for “ProcessACHEFT,” above. Parameters in **bold** are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	This is the IP address of the donor’s computer. Please send to fully utilise the fraud tools available from iATS. IPv4 only.	N/A
invoiceNum		40
accountNum	USD: Routing no. (9 digits) + account no. (# of digits varies) CAD: Bank no. (3 digits) + transit no. (5 digits) + account no. (# of digits varies) AUD: BIC (Bank ID) + IBAN (Account) * Order must be followed with NO spaces/dashes. Ie: 12312345123456789 *	30 digits
accountType	Options: CHECKING, SAVING (*Applies to NA only). If left blank we will use Checking.	N/A
firstName	**Highly Recommended**. Transactions will reject if left blank.	30
lastName	**Highly Recommended**. Transactions will reject if left blank.	30

Parameter	Notes/Options	Char. Limit
address		40
city		30
state	State or province (NA only)	30
zipCode		19
total	Prevent the use of commas (,) within the dollar amount fields if possible. iATS reads a comma as a period in this field to compensate for different countries use of the comma within currency. For example: $1,000	N/A
comment		110
title		
phone		
phone2		
fax		
email		
country		
item1		
item2		
item3		
item4		
item5		
item6		

Sample Reply Message Format

```
<IATSRESPONSE>
    <STATUS>StatusType</STATUS>
    <ERRORS>ErrorCodeType</ERRORS>
    <PROCESSRESULT>
            <AUTHORIZATIONRESULT>AuthorizationResultType
            </AUTHORIZATIONRESULT>
            <CUSTOMERCODE>CustomerCode</CUSTOMERCODE>
            <TRANSACTIONID>TransactionId</TRANSACTIONID>
    </PROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values		
Return Value	**Occurrence**	**Explanation**
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc. *Reject 3: Customer Code does not exist* *Or Account num is too long* *Reject 40: Invalid card number*
CUSTOMERCODE	Never	No Token (Customer Code) is created for this process.
TRANSACTIONID	Conditional	Return information when Status = Success and the transaction has been processed; value is transactionID.

4.2 Create a Token (Bank Debit) and Charge Account

This service is for when you want to create a Bank Debit Token (Customer Code) for payment storage purposes only and to process a onetime Bank Debit transaction at the same time.

- CreateCustomerCodeAndProcessACHEFT

This call is only available on our NA, and AUD portion of our UK system.

The Token (Customer Code) can only be used to process single transactions in the future unless you update the recurring field values separately using the Update API calls from CustomerLink.

- Please refer to the CustomerLink Guide or reach out to webservice@iatspayments.com for more details on best practises for creating a new Recurring Customer Code token.
- If you will be using the new token for a recurring schedule in the future, please avoid using any sub-codes ending in 90-99 (for example ABDC99) as these will affect future scheduled transactions. We recommend using an 80 level code (for example, ABDC80) for online webpage initiated schedules and a 01 level code (for example, ABCD01) for offline software initiated schedules.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- AUD/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of “CreateCustomerCodeAndProcessACHEFT”.

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., “string”, “dateTime,” etc.) need to be replaced with actual values.

Request

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <CreateCustomerCodeAndProcessACHEFT xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <firstName>string</firstName>
      <lastName>string</lastName>
      <address>string</address>
      <city>string</city>
      <state>string</state>
      <zipCode>string</zipCode>
      <accountNum>string</accountNum>
      <accountType>string</accountType>
      <invoiceNum>string</invoiceNum>
      <total>string</total>
      <comment>string</comment>
      <title>string</title>
      <phone>string</phone>
      <phone2>string</phone2>
      <fax>string</fax>
      <email>string</email>
      <country>string</country>
      <item1>string</item1>
      <item2>string</item2>
      <item3>string</item3>
      <item4>string</item4>
      <item5>string</item5>
      <item6>string</item6>
    </CreateCustomerCodeAndProcessACHEFT>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <CreateCustomerCodeAndProcessACHEFTResponse
xmlns="https://www.iatspayments.com/NetGate/">

<CreateCustomerCodeAndProcessACHEFTResult>xml</CreateCustomerCodeAndProcessACHEFTResult>
    </CreateCustomerCodeAndProcessACHEFTResponse>
  </soap12:Body>
</soap12:Envelope>
```

4.2.1 Overview of Request Parameters

Here is an overview of the request parameters for “CreateCustomerCodeAndProcessACHEFT” above. Parameters in **bold** are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	This is the IP address of the donor’s computer. Please send to fully utilise the fraud tools available from iATS. IPv4 only.	N/A
firstName	**Highly Recommended.** Transactions will reject if left blank.	30
lastName	**Highly Recommended.** Transactions will reject if left blank.	30
address	**Optional but highly recommended**	40
city	**Optional but highly recommended**	30
state	State or province (NA). **Optional but highly recommended**	30
zipCode	**Optional but highly recommended**	19
accountNum	• USD: Routing no. (9 digits) + account no. (# of digits varies) • CAD: Bank no. (3 digits) + transit no. (5 digits) + account no. (# of digits varies) • AUD: BIC (Bank ID) + IBAN (Account) * Order must be followed with NO spaces/dashes. Ie: 12312345123456789 *	30 digits
accountType	Optional options: CHECKING, SAVING (*Applies to North America only). If left blank we will use Checking.	N/A
invoiceNum		40
total	Prevent the use of commas (,) within the dollar amount fields if possible. iATS reads a comma as a period in this field to compensate for different countries use of the comma within currency. For example: $1,000	N/A
comment		110
title		
phone		
phone2		
fax		
email		
country		
item1		
item2		
item3		

Parameter	Notes/Options	Char. Limit
item4		
item5		
item6		

Sample Reply Message Format

```
<IATSRESPONSE>
    <STATUS>StatusType</STATUS>
    <ERRORS>ErrorCodeType</ERRORS>
    <PROCESSRESULT>
            <AUTHORIZATIONRESULT>AuthorizationResultType
            </AUTHORIZATIONRESULT>
            <CUSTOMERCODE>CustomerCode</CUSTOMERCODE>
            <TRANSACTIONID>TransactionId</TRANSACTIONID>
    </PROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values		
Return Value	**Occurrence**	**Explanation**
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc.
CUSTOMERCODE	Conditional	Return information when Status = Success and create Token (Customer Code) successfully; value is new Token (Customer Code).
TRANSACTIONID	Conditional	Return information when Status = Success and the transaction has been processed; value is transactionID.

4.3 Process a Bank Debit Transaction Using an Existing Token (Customer Code)

This service is for when you want to charge a credit card transaction using a Customer Code Token that already exists.

- ProcessACHEFTWithCustomerCode

This call is only available on our NA, and AUD portion of our UK system.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- AUD/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of “ProcessACHEFTWithCustomerCode”.

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., “string”, “dateTime,” etc.) need to be replaced with actual values.

Request (NA; for AUD International see link above)

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessACHEFTWithCustomerCode xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <customerCode>string</customerCode>
      <invoiceNum>string</invoiceNum>
      <total>string</total>
      <comment>string</comment>
      <item1>string</item1>
      <item2>string</item2>
      <item3>string</item3>
      <item4>string</item4>
      <item5>string</item5>
      <item6>string</item6>
    </ProcessACHEFTWithCustomerCode>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessACHEFTWithCustomerCodeResponse
xmlns="https://www.iatspayments.com/NetGate/">
      <ProcessACHEFTWithCustomerCodeResult>xml</ProcessACHEFTWithCustomerCodeResult>
    </ProcessACHEFTWithCustomerCodeResponse>
  </soap12:Body>
</soap12:Envelope>
```

4.3.1 Overview of Request Parameters

Here is an overview of the request parameters for “ProcessACHEFTWithCustomerCode,” above. Parameters in red are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	This is the IP address of the donor’s computer. Please send to fully utilise the fraud tools available from iATS. IPv4 only.	N/A
customerCode	Existing Token (Customer Code)	40
invoiceNum		40
total	Prevent the use of commas (,) within the dollar amount fields if possible. iATS reads a comma as a period in this field to compensate for different countries use of the comma within currency. For example: $1,000	N/A
comment		100
item1		
item2		
item3		
item4		
item5		
item6		

Sample Reply Message Format

```
<IATSRESPONSE>
    <STATUS>StatusType</STATUS>
    <ERRORS>ErrorCodeType</ERRORS>
    <PROCESSRESULT>
            <AUTHORIZATIONRESULT>AuthorizationResultType
            </AUTHORIZATIONRESULT>
            <CUSTOMERCODE>CustomerCode</CUSTOMERCODE>
            <TRANSACTIONID>TransactionId</TRANSACTIONID>
    </PROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values

Return Value	Occurrence	Explanation
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc.
CUSTOMERCODE	Conditional	Return information when Status = Success and create Token (Customer Code) successfully; value is new Token (Customer Code).
TRANSACTIONID	Conditional	Return information when Status = Success and the transaction has been processed; value is transactionID.

4.4 Process Multiple Bank Debit Transactions in a Batch

This service is for when you wish to process multiple Bank Debit transactions (made on multiple bank accounts) in one request. Your request will include a range of details for each bank account, as is described in detail below in the sections. Once uploaded, use the GetBatchProcessResultFile call details located in section “Obtain the Results of a Bank Debit Batch File” on page 46.

- • ProcessACHEFTChargeBatch

This call is only available on our NA, and AUD portion of our UK system.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- UK/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of “ProcessACHEFTChargeBatch”.

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., “string”, “dateTime,” etc.) need to be replaced with actual values.

Request for NA system, for AUD International see link above:

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessACHEFTChargeBatch xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <batchFile>base64Binary</batchFile>
    </ProcessACHEFTChargeBatch>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessACHEFTChargeBatchResponse xmlns="https://www.iatspayments.com/NetGate/">
      <ProcessACHEFTChargeBatchResult>xml</ProcessACHEFTChargeBatchResult>
    </ProcessACHEFTChargeBatchResponse>
  </soap12:Body>
</soap12:Envelope>
```

4.4.1 Overview of Request Parameters

Here is an overview of the request parameters for “ProcessACHEFTChargeBatch,” above. Parameters in **bold** are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	The IP address where the batch is sent from. IPv4 only.	N/A
batchFile	Value will be base64Binary. Explanation and further details below.	No Limit

Sample Reply Message Format

```
<IATSRESPONSE>
    <STATUS>StatusType</STATUS>
    <ERRORS>ErrorCodeType</ERRORS>
    <BATCHPROCESSRESULT >
            <AUTHORIZATIONRESULT>AuthorizationResultType
            </AUTHORIZATIONRESULT>
            <BATCHID>BatchId</BATCHID>
    <BATCHPROCESSRESULTFILE>BatchFile</BATCHPROCESSRESULTFILE>
    </BATCHPROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values		
Return Value	**Occurrence**	**Explanation**
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc. *Reject 3: Customer Code does not exist* *Or Account num is too long* *Reject 40: Invalid card number*
BATCHID	Conditional	Return information when Status = Success and batch file has been processed.
BATCHPROCESSRESULTFILE	Conditional	Return information when Status = Success and batch file processing has completed. Value will be base64Binary.

4.4.2 Explanation of base64Binary Format

Any binary text ("friendly" readable text) is converted or encoded into Base64 format so that the batch file can be processed by the web service.

Web service does not understand binary data so it needs to convert the friendly language to something that web service can understand. We encode the binary data into characters, and Base64 is one of these types of encodings.

Example Base64 format:

MTAvMjgvMTEsIDAwMDAxLEpvZSxTbWl0aCwxMjMgQW55IFN0LEFueXRvd24sU1QsMTAwMjMsMj
UuMDAsQU1YLDMwMDAwMDAwMDAwMDAwMSwwNzA4DQoxMC8yOC8xMSwgMDAwMDEsSm9l
LFNtaXRoLDEyMyBBbnkgU3QsQW55dG93bixTVCwxMDAyMywyNS4wMCxBTVgsMzAwMDAwMDAw
MDAwMDAxLDA3MDgNCjEwLzI4LzExLCAwMDAwMSxKb2UsU21pdGgsMTIzIEFueSBTdCxBbnl0b3duL
FNULDEwMDIzLDI1LjAwLEFNWCwzMDAwMDAwMDAwMDAwMDEsMDcwOA0KDQo=

To produce these results, you will need to make the xml doc a binary object of type base64. Depending on the programming language you are using, you'll use a Convert method to encode binary to base64 format. As shown in result above.

In C#, you can convert to base64 string using this:

```
Convert.ToBase64String(byteArray)
```

In the XML message, put the base64 string between <batchFile> and </batchFile>

Example including formatting from Request section above--this is what the request should look like:

```
<?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>
    <ProcessACHEFTChargeBatch xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <batchFile>
MTAvMjgvMTEsIDAwMDAxLEpvZSxTbWl0aCwxMjMgQW55IFN0LEFueXRvd24sU1QsMTAwMjMsMj
UuMDAsQU1YLDMwMDAwMDAwMDAwMDAwMSwwNzA4DQoxMC8yOC8xMSwgMDAwMDEsSm9l
LFNtaXRoLDEyMyBBbnkgU3QsQW55dG93bixTVCwxMDAyMywyNS4wMCxBTVgsMzAwMDAwMDAw
MDAwMDAxLDA3MDgNCjEwLzI4LzExLCAwMDAwMSxKb2UsU21pdGgsMTIzIEFueSBTdCxBbnl0b3duL
FNULDEwMDIzLDI1LjAwLEFNWCwzMDAwMDAwMDAwMDAwMDEsMDcwOA0KDQo=
</batchFile>
    </ProcessACHEFTChargeBatch>
  </soap12:Body>
</soap12:Envelope>
```

4.4.3 Explanation of Batch File Types

Format:
Invoice, First Name, Last Name, Account Type, Account # (no spaces or dashes), Amount (no $), Comment

Example:
a1234,Joe,Smith,CHECKING,00567890567890,1.00,comment1

- USD: Routing no. (9 digits) + account no. (# of digits varies)
- CAD: Bank no. (3 digits) + transit no. (5 digits) + account no. (# of digits varies)
- AUD: BIC (Bank ID) + IBAN (Account)

* Order must be followed with NO spaces/dashes. Ie: 12312345123456789 *

4.4.4 Batch file format using Tokens (Customer Codes)

Format:
Invoice Number, Token (Customer Code), Amount, Comment

Example:
a123467,A168786,1.10,comment1

Notes:

- Applies to all currencies.
- If any fields must be omitted (the invoice number for example), the comma separators must still be included. For example,
  ,,Joe,Smith,CHECKING,00567890567890,1.00,comment1
  ,,A168786,1.10,comment1

4.4.5 Obtain the Results of a Bank Debit Batch File

This service is for when you wish to look at the results (rejects and approvals) processing multiple bank debit transactions (made on multiple credit cards) in one request from Section 3.4.

- GetBatchProcessResultFile

This call is available on either our NA or UK system.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- UK/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of “GetBatchProcessResultFile”.

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., “string”, “dateTime,” etc.) need to be replaced with actual values.

Request for NA system; for UK see link above:

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <GetBatchProcessResultFile xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <batchId>string</batchId>
    </GetBatchProcessResultFile>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <GetBatchProcessResultFileResponse xmlns="https://www.iatspayments.com/NetGate/">
      <GetBatchProcessResultFileResult>xml</GetBatchProcessResultFileResult>
    </GetBatchProcessResultFileResponse>
  </soap12:Body>
</soap12:Envelope>
```

4.4.6 Overview of Request Parameters

Here is an overview of the request parameters for “GetBatchProcessResultFile,” above. Parameters in red are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	The IP address where the batch is sent from. IPv4 only.	N/A
batchId		

Sample Reply Message Format

```
<IATSRESPONSE>
    <STATUS>StatusType</STATUS>
    <ERRORS>ErrorCodeType</ERRORS>
    <BATCHPROCESSRESULT >
            <AUTHORIZATIONRESULT>AuthorizationResultType
            </AUTHORIZATIONRESULT>
            <BATCHID>BatchId</BATCHID>
    <BATCHPROCESSRESULTFILE>BatchFile</BATCHPROCESSRESULTFILE>
    </BATCHPROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values		
Return Value	**Occurrence**	**Explanation**
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc. *Reject 3: Customer Code does not exist* *Or Account num is too long* *Reject 40: Invalid card number*
BATCHID	Conditional	Return information when Status = Success and batch file has been processed.
BATCHPROCESSRESULTFILE	Conditional	Return information when Status = Success and batch file processing has completed. Value will be base64Binary.

4.4.7 Explanation of Processed Batch File

Batch file results:

These messages are appended to the end of the request batch file:

- OK: + 6 digits
- REJECT: TIMEOUT
- WRONG FORMAT
- Duplicated
- …

Example request:

```
a1234,Joe,Smith,CHECKING,00567890567890,-1.00,comment1
a2224,Tony,,CHECKING,00567890560000,-20.00,comment2
```

Example results:

```
a1234,Joe,Smith,CHECKING,00567890567890,-1.00,comment1, REJECT:TIMEOUT
a2224,Tony,,CHECKING,00567890560000,-20.00,comment2, OK:555555
```

5 Calls to Refund Credit Card and Bank Debit Transactions

5.1 Refunding a Credit Card Transaction

This service is for when you want to refund a credit card transaction using its Transaction ID.

- ProcessCreditCardRefundWithTransactionId

This call is available on either our NA or UK system.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- UK/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of “ProcessCreditCardRefundWithTransactionId”.

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., “string”, “dateTime,” etc.) need to be replaced with actual values.

Request (NA server; please use UK link above for UK server)

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessCreditCardRefundWithTransactionId xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <transactionId>string</transactionId>
      <total>string</total>
      <comment>string</comment>
    </ProcessCreditCardRefundWithTransactionId>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessCreditCardRefundWithTransactionIdResponse
xmlns="https://www.iatspayments.com/NetGate/">

<ProcessCreditCardRefundWithTransactionIdResult>xml</ProcessCreditCardRefundWithTransaction
IdResult>
    </ProcessCreditCardRefundWithTransactionIdResponse>
  </soap12:Body>
</soap12:Envelope>
```

5.1.1 Overview of Request Parameters

Here is an overview of the request parameters for “ProcessCreditCardRefundWithTransactionId” above. Parameters in **bold** are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	The IP address where the batch is sent from. IPv4 only.	N/A
transactionId		30
total	Must be negative number; e.g., “-10” Prevent the use of commas (,) within the dollar amount fields if possible. iATS reads a comma as a period in this field to compensate for different countries use of the comma within currency. For example: $1,000	N/A
comment		110

Sample Reply Message Format

```
<IATSRESPONSE>
    <STATUS>StatusType</STATUS>
    <ERRORS>ErrorCodeType</ERRORS>
    <PROCESSRESULT>
            <AUTHORIZATIONRESULT>AuthorizationResultType
            </AUTHORIZATIONRESULT>
            <CUSTOMERCODE>CustomerCode</CUSTOMERCODE>
            <TRANSACTIONID>TransactionId</TRANSACTIONID>
    </PROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values		
Return Value	**Occurrence**	**Explanation**
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc. *Reject 3: Customer Code does not exist* *Or Account num is too long* *Reject 40: Invalid card number*
CUSTOMERCODE	Never	No Token (Customer Code) is created for this process.
TRANSACTIONID	Conditional	Return information when Status = Success and the transaction has been processed; value is transactionID.

5.2 Refunding a Bank Debit Transaction – Not for UK Direct Debit Use

This service is for when you want to refund a bank debit transaction using its Transaction ID.

- ProcessACHEFTRefundWithTransactionId

This call is available on either our NA or UK system, however it cannot be used for UK Direct Debit.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- UK/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of “ProcessACHEFTRefundWithTransactionId”.

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., “string”, “dateTime,” etc.) need to be replaced with actual values.

Request (NA; for International see link above)

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessACHEFTRefundWithTransactionId xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <transactionId>string</transactionId>
      <total>string</total>
      <comment>string</comment>
    </ProcessACHEFTRefundWithTransactionId>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessACHEFTRefundWithTransactionIdV1Response
xmlns="https://www.iatspayments.com/NetGate/">

<ProcessACHEFTRefundWithTransactionIdV1Result>xml</ProcessACHEFTRefundWithTransactionIdV
1Result>
    </ProcessACHEFTRefundWithTransactionIdV1Response>
  </soap12:Body>
</soap12:Envelope>
```

5.2.1 Overview of Request Parameters

Here is an overview of the request parameters for “ProcessACHEFTRefundWithTransactionId,” above. Parameters in **bold** are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	The IP address where the batch is sent from. IPv4 only.	N/A
transactionId		30
total	Must be negative number; e.g., “-10” Prevent the use of commas (,) within the dollar amount fields if possible. iATS reads a comma as a period in this field to compensate for different countries use of the comma within currency. For example: $1,000	N/A
comment		110

Sample Reply Message Format

```
<IATSRESPONSE>
    <STATUS>StatusType</STATUS>
    <ERRORS>ErrorCodeType</ERRORS>
    <PROCESSRESULT>
            <AUTHORIZATIONRESULT>AuthorizationResultType
            </AUTHORIZATIONRESULT>
            <CUSTOMERCODE>CustomerCode</CUSTOMERCODE>
            <TRANSACTIONID>TransactionId</TRANSACTIONID>
    </PROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values

Return Value	Occurrence	Explanation
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc. *Reject 3: Customer Code does not exist* *Or Account num is too long* *Reject 40: Invalid card number*
CUSTOMERCODE	Never	No Token (Customer Code) is created for this process.
TRANSACTIONID	Conditional	Return information when Status = Success and the transaction has been processed; value is transactionID.

5.3 Refund Multiple Bank Debit Transactions (Not for UK Direct Debit)

This service is for when you want to refund a multiple Bank Debit transactions in a batch using its Transaction ID.

- ProcessACHEFTRefundBatch

This call is available on either our NA or UK system, however it cannot be used for UK Direct Debit.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- UK/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of “ProcessACHEFTRefundBatch”.

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., “string”, “dateTime,” etc.) need to be replaced with actual values.

Request (NA; for International see link above)

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessACHEFTRefundBatch xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <batchFile>base64Binary</batchFile>
    </ProcessACHEFTRefundBatch>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessACHEFTRefundBatchResponse xmlns="https://www.iatspayments.com/NetGate/">
      <ProcessACHEFTRefundBatchResult>xml</ProcessACHEFTRefundBatchResult>
    </ProcessACHEFTRefundBatchResponse>
  </soap12:Body>
</soap12:Envelope>
```

5.3.1 Overview of Request Parameters

Here is an overview of the request parameters for “ProcessACHEFTRefundBatch,” above. Parameters in **bold** are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	The IP address where the batch is sent from. IPv4 only. Please send to fully utilise the fraud tools available from iATS. IPv4 only.	N/A
batchFile	Value will be base64Binary. See the explanation and format below.	No Limit

5.3.2 Explanation of base64Binary Format

Any binary text (“friendly” readable text) is converted or encoded into **Base64** format so that the batch file can be processed by the web service.

Web service does not understand binary data so it needs to convert the friendly language to something that web service can understand. We encode the binary data into characters, and Base64 is one of these types of encodings.

Example Base64 format:

```
MTAvMjgvMTEsIDAwMDAxLEpvZSxTbWl0aCwxMjMgQW55IFN0LEFueXRvd24sU1QsMTAwMjMsMj
UuMDAsQU1YLDMwMDAwMDAwMDAwMDAwMSwwNzA4DQoxMC8yOC8xMSwgMDAwMDEsSm9l
LFNtaXRoLDEyMyBBbnkgU3QsQW55dG93bixTVCwxMDAyMywyNS4wMCxBTVgsMzAwMDAwMDAw
MDAwMDAxLDA3MDgNCjEwLzI4LzExLCAwMDAwMSxKb2UsU21pdGgsMTIzIEFueSBTdCxBbnl0b3duL
FNULDEwMDIzLDI1LjAwLEFNWCwzMDAwMDAwMDAwMDAwMDEsMDcwOA0KDQo=
```

To produce these results, you will need to make the xml doc a binary object of type base64. Depending on the programming language you are using, you'll use a Convert method to encode binary to base64 format. As shown in result above.

In C#, you can convert to base64 string using this:

```
Convert.ToBase64String(byteArray)
```

In the XML message, put the base64 string between <batchFile> and </batchFile>

Example including formatting from the **Request** section above--this is what the request should look like:

```
<?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>
    <ProcessACHEFTChargeBatchV1 xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <batchFile>
MTAvMjgvMTEsIDAwMDAxLEpvZSxTbWl0aCwxMjMgQW55IFN0LEFueXRvd24sU1QsMTAwMjMsMj
UuMDAsQU1YLDMwMDAwMDAwMDAwMDAwMSwwNzA4DQoxMC8yOC8xMSwgMDAwMDEsSm9l
LFNtaXRoLDEyMyBBbnkgU3QsQW55dG93bixTVCwxMDAyMywyNS4wMCxBTVgsMzAwMDAwMDAw
MDAwMDAxLDA3MDgNCjEwLzI4LzExLCAwMDAwMSxKb2UsU21pdGgsMTIzIEFueSBTdCxBbnl0b3duL
FNULDEwMDIzLDI1LjAwLEFNWCwzMDAwMDAwMDAwMDAwMDEsMDcwOA0KDQo=</batchFile>
    </ProcessACHEFTChargeBatchV1>
  </soap12:Body>
</soap12:Envelope>
```

Explanation of Batch file types:

Format:
Invoice, First Name, Last Name, Account Type, Account # (no spaces or dashes), Amount (no $), Comment

Example:
a1234,Joe,Smith,CHECKING,00567890567890,1.00,comment1

- USD: Routing no. (9 digits) + account no. (# of digits varies)
- CAD: Bank no. (3 digits) + transit no. (5 digits) + account no. (# of digits varies)
- AUD: BIC (Bank ID) + IBAN (Account)
  * Order must be followed with NO spaces/dashes. Ie: 12312345123456789 *

5.3.3 Obtain the Results of a Credit Card Batch File

This service is for when you wish to look at the results (rejects and approvals) processing multiple bank debit transactions (made on multiple credit cards) in one request from Section 3.4.

- GetBatchProcessResultFile

This call is available on either our NA or UK system.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- North America
- UK/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the North American version of "GetBatchProcessResultFile".

Notes:

- This code corresponds to SOAP 1.2 shown at the NA server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., "string", "dateTime," etc.) need to be replaced with actual values.

Request for NA system; for UK see link above:

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <GetBatchProcessResultFile xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <batchId>string</batchId>
    </GetBatchProcessResultFile>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <GetBatchProcessResultFileResponse xmlns="https://www.iatspayments.com/NetGate/">
      <GetBatchProcessResultFileResult>xml</GetBatchProcessResultFileResult>
    </GetBatchProcessResultFileResponse>
  </soap12:Body>
</soap12:Envelope>
```

5.3.4 Overview of Request Parameters

Here is an overview of the request parameters for “GetBatchProcessResultFile,” above. Parameters in **bold** are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	The IP address where the batch is sent from. IPv4 only.	N/A
batchId		

Sample Reply Message Format:

```
<IATSRESPONSE>
    <STATUS>StatusType</STATUS>
    <ERRORS>ErrorCodeType</ERRORS>
    <BATCHPROCESSRESULT >
            <AUTHORIZATIONRESULT>AuthorizationResultType
            </AUTHORIZATIONRESULT>
            <BATCHID>BatchId</BATCHID>
    <BATCHPROCESSRESULTFILE>BatchFile</BATCHPROCESSRESULTFILE>
    </BATCHPROCESSRESULT>
</IATSRESPONSE >
```

The table below describes what can be expected for each of the return values:

Explanation of Sample Reply Message Format Values		
Return Value	**Occurrence**	**Explanation**
STATUS	Always	The system-level acknowledgement code that indicates the iATS system status: • **Success** - iATS system processing succeeded. • **Failure** - An error has occurred on the iATS system side, such as a database or server down.
ERRORS	Conditional	Return information when Status = Failure. Return value will depend on the error.
AUTHORIZATIONRESULT	Conditional	Return information when Status = Success. This value represents the application-level result that iATS processed the request: • **OK** - Request has been approved • **Error** - Request not approved, an error has occurred—possibly due to problems in delivery, attempt by end-user to submit invalid or missing data, etc. iATS does not currently have a list of possible error messages as they can be sent due to different types of processing and from different components, etc. *Reject 3: Customer Code does not exist* *Or Account num is too long* *Reject 40: Invalid card number*
BATCHID	Conditional	Return information when Status = Success and batch file has been processed.
BATCHPROCESSRESULTFILE	Conditional	Return information when Status = Success and batch file processing has completed. Value will be base64Binary.

Explanation of Processed Batch file:

Batch file results:

These messages are appended to the end of the request batch file:

- OK: + 6 digits
- REJECT: TIMEOUT
- WRONG FORMAT
- Duplicated
- …

Example request:

```
a1234,Joe,Smith,CHECKING,00567890567890,-1.00,comment1
a2224,Tony,,CHECKING,00567890560000,-20.00,comment2
```

Example results:

```
a1234,Joe,Smith,CHECKING,00567890567890,-1.00,comment1, REJECT:TIMEOUT
a2224,Tony,,CHECKING,00567890560000,-20.00,comment2, OK:555555
```

6 Process a UK Direct Debit Representment

Debits that appear in a BACS ARUDD report (see ReportLink for call to pull this from iATS) contain debits that were uncollectable. **Only those with reason code 0 (Refer to Payer)** may be represented again for collection provided the collection is made within **one calendar month** of the original failed collection.

Bacs guidelines suggest that the payer should be contacted before the debit is represented to ensure the payer has sufficient funds in their account, but this is not mandatory.

- ProcessDirectDebitRepresentment

This call is available on our UK system.

SOAP Request and Response Server Addresses:

Depending on which server you require, you can find the SOAP request and response structures at the following URLs:

- UK/International

SOAP Request and Response Structure:

The code below shows the SOAP request and response structure for the UK version of “ProcessDirectDebitRepresentment”.

Notes:

- This code corresponds to SOAP 1.2 shown at the UK server address above.
- The addresses above also contain the code for SOAP 1.1 if you require it.
- The request parameter placeholders shown (e.g., “string”, “dateTime,” etc.) need to be replaced with actual values.

Request (UK Only)

```
POST /netgate/ProcessLinkv2.asmx HTTP/1.1
Host: www.uk.iatspayments.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessDirectDebitRepresentment xmlns="https://www.iatspayments.com/NetGate/">
      <agentCode>string</agentCode>
      <password>string</password>
      <customerIPAddress>string</customerIPAddress>
      <ACHEFTReferenceNum>string</ACHEFTReferenceNum>
      <originalDebitDate>dateTime</originalDebitDate>
      <originalDebitAmount>string</originalDebitAmount>
    </ProcessDirectDebitRepresentment>
  </soap12:Body>
</soap12:Envelope>
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?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>
    <ProcessDirectDebitRepresentmentResponse
xmlns="https://www.iatspayments.com/NetGate/">
      <ProcessDirectDebitRepresentmentResult>xml</ProcessDirectDebitRepresentmentResult>
    </ProcessDirectDebitRepresentmentResponse>
  </soap12:Body>
</soap12:Envelope>
```

6.1.1 Overview of Request Parameters

Here is an overview of the request parameters for “ProcessDirectDebitRepresentment,” above. Parameters in **bold** are mandatory.

Parameter	Notes/Options	Char. Limit
agentCode		10
password		10
customerIPAddress	This is the IP address of the donor’s computer. Not needed for this call	N/A
ACHEFTReferenceNum	Reference Number for the Donor	
originalDebitDate	Original Date of the rejected charge.	
originalDebitAmount	Amount of the original Debit to the donor.	

7 Appendix A: Testing iATS Payments Systems

The TEST88 and AURA88 iATS Client Codes have been created to allow for testing iATS Payments systems, now available for both Credit Card and dynamic North American ACH/EFT processing.

These accounts are not live merchant accounts, therefore Authorization and Rejection results are not real.

Please note that this test information is provided to many clients, therefore **please do not modify** or delete any pre-existing Aura Event's or change the password of this code.

User ID = TEST88

Password = TEST88

OR

User ID = AURA88

Password = AURA88

URL: NA = www.iatspayments.com

UK = www.uk.iatspayments.com

7.1 Credit Card Testing

Notes on Credit Card processing:

- To test the Authorization and Rejection responses related to a Charge, use
  - Visa number 4111111111111111
  - MC number 5111111111111118
  - AMX number 371111111111114
  - DSC number 6011111111111117
- To test the Auth responses related to both Charges and Refunds, use credit card number 4222222222222220
- If needed, use any 3 digit CVV2 for Visa, MC and DSC. Use any 4 digit CVV2 for AMX.

Amount	Results when using Visa (4111111111111111), MC, AMX or DSC #'s above
1.00	OK: 678594
2.00	REJ: 15
3.00	OK: 678594
4.00	REJ: 15
5.00	REJ: 15
6.00	OK: 678594:X
7.00	OK: 678594:y
8.00	OK: 678594:A
9.00	OK: 678594:Z
10.00	OK: 678594:N
15.00	If CVV2 = 1234, the response is OK: 678594:Y. If no CVV2 is entered, the response is REJ: 19
16.00	REJ: 2
17.00	REJ: 22

Amount	Results when using Visa (4111111111111111), MC, AMX or DSC #'s above
192.00 (NA Only)	If sent via API (iATSLink or WebService) a) **IP address** is invalid format: Rej: 5. b) **IP address** is valid format: OK: 678594.
All Other Amounts	REJ: 15
All Refund Amounts	REJ: 15
Amount	**Results when using 4222222222222220 only**
Any Sale Amount	OK: 678594
Any Refund Amount	OK: 678594

Please note there is a transaction limit of $2000.00 (£2000.00) per charge. Amounts above will be Rej:39

- For Direct Post Method, the NA and UK TEST88 Process Key credentials are: **PA0940D765F2BD67BD97B82EFAA4D72BE9**

7.2 North American ACH/EFT Debit Testing

Notes on ACH Processing:

- ACH/EFT Transaction processing is not processed in "Real Time" like credit card transactions and there can be processing delays while the data is moving between iATS, our Bank and the donors Bank.
- Live and Test ACH/EFT processing is sent to the bank as per the following schedule:
  - ➢ Monday and Tuesday: File sent twice per day (3am and 3pm)
  - ➢ Wednesday to Friday: File sent once per day (3pm)
  - ➢ No files are sent over the weekend or on US bank holidays.
- When ACH/EFT transactions are sent to iATS for processing, there are three stages which each transaction will go through:
  - ➢ "**Tobesent**" - The transaction is waiting to be sent to iATS' bank as per above schedule.
  - ➢ "**Pending**" - The transactions have been sent to the bank, but final approval/rejection has not been received from the bank (1 Business day in general).
  - ➢ "**OK:BankAccept**", or "**OK:Reject**" – The final results from the bank indicating approval or decline. Check the Details of the Reject Journal on the iATS Portal for more information.
  - ➢ "Return" - A separate Return record may be recorded on the iATS Journal report. These may be posted at any future date, and has been issued by the bank or donor. Check the Details of the Return report on the iATS Portal for more information.
- For North American ACH testing, any account details can be used however we recommend:

Canada	**USD**
Bank: 123 Transit: 00000 Account: 123456	Routing: 111111111 Account: 12345678

ACH Dynamic Testing Parameters:

Desired Final Status	Amount Value to Use	Results and Timing
Approved	$1.00	Final status received within 1 Business day • Status will be dynamically changed from **ToBeSent** (initial value) to **Pending** as per schedule in notes above. • Will be changed to **Approved** approx. 1 hour after sent to bank.
Rejected	Any value except $1.00 and $3.00	Final status received within 1 Business day • Status will be dynamically changed from **ToBeSent** (initial value) to **Pending** as per schedule in notes above. • Will be changed to **Rejected** approx. 1 hour after sent to bank.
Returned	$3.00	Final status received within 1 Business day • Status will be dynamically changed from **ToBeSent** (initial value) to **Pending** as per schedule in notes above. • Will be changed to **Approved** approx. 1 hour after sent to bank. • A new **Return** record will be created and posted the next business day.

Please note there is a transaction limit of $2000.00 per charge. Amounts above will be Rej:39

8 Appendix B: Definition of XML Element Abbreviations

The elements in the below table may or may not apply directly to the ProcessLink WebServices. All possible element options which are currently available within the different CustomerLink, ProcessLink, AuraLink, and ReportLink API calls are listed below.

For more information or if you have any questions, please reach out to iATS at webservice@iatspayments.com.

Element	Definition	Association
AC1, AC2	Account1, Account2,	Start element
ACH	ACHEFT	Start element
ACN	Account number	ACHEFT
ACTYP	Account type	ACHEFT
ADD	Address	Customer
AGT	Agent	Transaction
AMT	Amount	Transaction/recurring
ANM	Anonymous	Transaction
BD	Begin date	Recurring
CC	Credit card	Start element
CCN	Credit card number	Credit card
CM	Comment	Transaction/customer
CNT	Country	Customer
CO	Company	Customer
CST	Customer	Start element
CSTC	Token (Customer Code)	Customer
CSTN	Customer name	Credit card/ACHEFT
CTY	City	Customer
DTM	Date and time	Transaction
ED	End date	Recurring
EM	Email	Customer
EXP	Expiry	Credit card
FLN	Full name	Customer
FN	First name	Customer
FX	Fax	Customer
INV	Invoice	Transaction
IT1, IT2, …	Item1, item2, etc.	Transaction
LN	Last name	Customer
MB	Mobile	Customer
MP	Method of payment	Credit card
PH	Phone	Customer
RCR	Recurring	Start element
RE	Received email	Transaction
RST	Result	Transaction
SCHD	Schedule date	Recurring
SCHTYP	Schedule type	Recurring
ST	State	Customer

Element	Definition	Association
TN	Transaction	Start element
TNID	Transaction ID	Transaction
TNTYP	Transaction type	Transaction
ZC	Zip code	Customer