Authorization Gateway

Implementation Guide

07/02/2012

Version 1.4

Introduction:

This document serves as an implementation guide for interacting with the Authorization Gateway. Its purpose is to provide software developers with a standardized workflow for integrating the Authorization Gateway into their host system, as well as the information necessary to interpret the various Authorization Gateway responses so that the host system can take the appropriate action. The workflow contained within this document was designed to streamline the integration effort and ensure that all parties can accurately track the progress of the integration effort. The information contained within the “Beginning Authorization Gateway Development” section is included to assist your integration team with understanding the business logic and processes that need occur in order to successfully process transactions with the Authorization Gateway.

Overview:

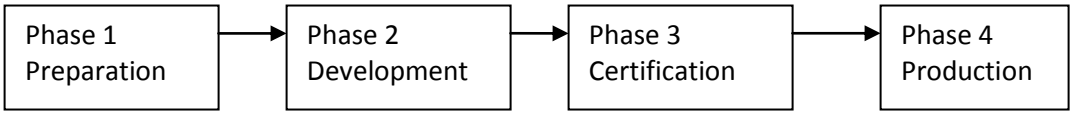
This document will outline a full lifecycle workflow for integrating the Authorization Gateway into your host system. This includes defining what steps will need to be taken for preparing for the integration effort, what requirements need to be met for integration, a required structured certification process, and ultimately configuring the host system for production.

Contents

Workflow Overview:

As mentioned in the overview this document will provide a full lifecycle workflow for integrating the authorization gateway into your host system. This workflow was designed to make the gateway integration process as simple as possible and will be used to guide your team throughout the integration effort and to keep the project on track.

The workflow consists of four distinct phases, with each phase culminating in a major milestone. The four distinct phases of the workflow are Preparation, Development, Certification, and Production. The linear graphic below illustrates how each of these phases leads to the next.



These four distinct phases identify the critical planning, assessment, and coordination of activities between the integration team and GETI's software development team.

Each phase is marked with a single major milestone that represents the successful culmination of all the activities of the phase. In addition to this major event, each phase may also have intermediate milestones leading up to the major milestone. These events mark the self-regulation points of the process. They are review and synchronization points rather than freeze points. They represent points in time when all team members synchronize the integration effort, and members of the project team agree that they have achieved the objectives of that particular phase. Milestones allow the team to assess the status of the integration effort as well as to make any necessary adjustments in the project scope to accommodate any changes that have developed during the course of the integration effort.

The individual phases are outlined below and include a table that defines the milestones required to complete each phase. The tables can also be used by your team to chart the progress of the integration effort.

Phase 1: Preparation

The Preparation Phase is the initial phase of the integration effort. During this phase the integration team will be responsible for reviewing and understanding the Authorization Gateway Specification as well as completing the milestones listed below. The completion of this phase marks the opportunity

for all to evaluate the integration effort, identify any remaining issues, and begin the Development Phase.

Phase	Milestones	Completed
Preparation	Review the Authorization Gateway Specification	
	Obtain a User Name and Password for Certification	
	Determine your SEC Code(s)	
	Determine your XML Template(s)	
	Determine your XML Schema(s)	
	Determine your Certification Terminal ID(s)	
	Establish Connectivity	
	Request Certification Terminal Settings	

Preparation Phase Milestones:

- **Review the Authorization Gateway Specification:** There is a separate document called the Authorization Gateway Specification that you should have received along with this Implementation guide. The Authorization Gateway Specification serves as a technical guide for transmitting check transactions to GETI and details the communication method, authorization request specifications, and response specifications. If you have not received a copy of this document please contact us.
- **Obtain a User Name and Password for Certification:** In order to connect to the Authorization Gateway you must contact us to obtain a user name and password that will allow you to connect to the gateway to begin the integration effort. This user name and password will be unique to your team and will only allow you to invoke web methods used for certification. Once you have reached the Production Phase you will be given another user name and password that will allow you to invoke production web methods.
- **Determine your SEC Code(s):** The SEC Codes are defined later in this document and are the main factor in determining what XML Template and Schema to use.
- **Determine your XML Template(s):** Once you have determined your SEC Code you can determine which XML Template to use. There is a section in the Specification document called "How to determine which XML Template to Use" that explains the purpose of the XML templates and will assist you in determining which Template(s) to use.
- **Determine your XML Schema(s):** Once you have determined your SEC Code you can also determine which XSD will be used to validate your data packet submission. There is a section in the Specification document called "How to determine which XSD to Use" that explains the purpose of the XSDs and will assist you in determining which XSD(s) to use.
- **Determine your Certification Terminal IDs:** Once you have determined your XSD(s) using the Specification document you can easily find the corresponding Certification Terminal ID listed in the same row as the XSD URL.
- **Establish Connectivity:** Create a web reference to the URL defined in the "Connection Method" section of the Specification document. This URL is only good for Testing and Certification a Production URL will be provided during the final phase of the integration effort.

- **Request Certification Terminal Settings:** Successfully invoke the GetCertificationTerminalSettings web method for each Certification Terminal ID previously identified.

Phase 2: Development

The Development Phase is the second phase of the integration effort. During this phase the integration team will be responsible for ensuring the host application can properly handle Authorizations, Declines, Represented Checks, Voids, and Manager Overrides. The section below entitled "Beginning Authorization Gateway Development" details the business logic necessary to complete each milestone in this phase. The completion of this phase marks the opportunity to begin the Certification Phase.

Phase	Milestones	Completed
Development	Validation Handling	
	Process Single Certification Check – Authorization	
	Process Single Certification Check – Check Limit Exceeded	
	Process Single Certification Check – Decline	
	Process Single Certification Check – Manager Needed	
	Process Single Certification Check – Represented Check	
	Process Single Certification Check – Void a previously Authorized Check	
	Exception Handling	
	Request a Certification Date	

Development Phase Milestones:

- **Validation Handling:** Successfully validate a request Data Packet against your published XSD(s) and have the host system be able to handle failed validation messages.
- **Process Single Certification Check – Authorization:** Successfully invoke the ProcessSingleCertificationCheck web method and send a data packet with the necessary information to generate an Authorization response.
- **Process Single Certification Check – Check Limit Exceeded:** Successfully invoke the ProcessSingleCertificationCheck web method and send a data packet with the necessary information to generate a Check Limit Exceed response.
- **Process Single Certification Check – Decline:** Successfully invoke the ProcessSingleCertificationCheck web method and send a data packet with the necessary information to generate a Decline response.
- **Process Single Certification Check – Manager Needed:** Successfully invoke the ProcessSingleCertificationCheck web method and send a data packet with the necessary information to generate a Manager Needed response, and successfully perform an override.

- **Process Single Certification Check –Represented Check:** Successfully invoke the ProcessSingleCertificationCheck web method and send a data packet with the necessary information to generate a Represented Check Response, and successfully perform an override.
- **Process Single Certification Check – Void:** Successfully invoke the ProcessSingleCertificationCheck web method and send a data packet with the necessary information to generate a Void response for a previously authorized check.
- **Exception Handling:** Include exception handling in the host system.
- **Request a Certification Date:** Upon successful completion of the above milestones you can request a date to undergo certification.

Phase 3: Certification

The Certification Phase is the third phase of the integration effort. During this phase the integration team will be responsible for sequentially completing the objectives outlined in the "Certification Script" section of this document. Our software team will closely monitor each transaction to ensure it is valid, and that the host system is properly configured to handle the various responses. The completion of this phase marks the opportunity to begin the Production Phase. However, if it is determined that the host system needs further refinements, it may be necessary to revert back to the Development Phase to make the necessary changes.

Phase	Milestones	Completed
Certification	Complete the Certification Script	

Certification Phase Milestones:

- **Complete the Certification Script:** Successfully complete each objective defined in the certification script included in this document.

Phase 4: Production

The Production Phase is the final phase of the integration effort. During this phase the integration team will be responsible for configuring the host application for production. This includes obtaining the production URL and authorization credentials as well as completing the milestones listed below. The Production Phase culminates in requesting a "Go Live" date.

Phase	Milestones	Completed
Production	Request a User Name and Password for Production	
	Request the Production URL	
	Redirect the host application to use the Production Web Methods using the Production User Name, Password, and Terminal ID.	
	Request a "Go Live" Date	

Production Phase Milestones:

- **Request a User Name and Password for Production:** Obtain a new unique user name and password that is authorized to invoke the production web methods.
- **Request the Production URL:** Obtain the URL that will be used to reference the production web methods.
- **Request a Production Terminal ID:** Obtain a Terminal ID for use in production.
- **Redirect the Host Application:** Redirect the host application to use the production URL and web methods with the provided production User Name, Password, and Terminal ID.
- **Process Single Check:** Process a single real check for testing purposes. Our software team will monitor the transaction to ensure it is properly sent to the Federal Reserve and then onto the merchant.
- **Request a "Go Live" Date:** Upon completion of the successfully submission of a single check a "Go Live" date for the host system can be established.

Preparing for Authorization Gateway Development (Phase 1)

Where Do I Start?

There are several milestones in the Preparation Phase that need to be completed prior to beginning development. These milestones were detailed in the section above, but your first steps will be to review the Authorization Gateway Specification document and obtain a user name and password.

The Authorization Gateway Specification document will assist your implementation team on determining the information for several of the other milestones in the Preparation Phase including the XML Template(s), XSD(s), and Certification Terminal ID(s). Defining this information will then allow you to move onto beginning work on the next milestone within the Preparation Phase which is connecting to the web service URL. It is assumed that you are familiar with working with XML, consuming web services, and with adding SOAP headers. Sample code is provided at the end of the Authorization Gateway Specification document, and the Submission section at the beginning of the Authorization Gateway Specification document defines the SOAP header. If you have any questions or need any guidance please feel free to contact us at the number provided in the Contact Information section.

Once you have successfully connected to the Authorization Gateway and are comfortable with adding the SOAP header, the Preparation Phase culminates with the major milestone of invoking the GetCertificationTerminalSettings web method.

The GetCertificationTerminalSettings web method is defined in the Authorization Gateway Specification document in the "Terminal Settings – XML Specification" section, but essentially this web method can be invoked to request information about a specific certification terminal. This web method does not need to be invoked on a continuous basis, but can be invoked if your implementation team determines that the host system needs to acquire information about the

Authorization Gateway Terminal. The invocation of this web method is made part of the Preparation Phase because it is the simplest web method and requires no input parameters.

It is important to note that the GetCertificationTerminalSettings has a sister web method called GetTerminalSettings that performs the same function for production terminals, but we will go into this in more detail during the Production Phase.

What Are The Different Standard Entry Class (SEC) Codes?

The Authorization Gateway uses the Standard Entry Class (SEC) codes to determine what information is required to be sent in the submission. The National Automated Clearing House Association (NACHA) requires the use of SEC Codes for each transaction settled through the Automated Clearing House (ACH). Each code identifies what type of transaction occurred. In addition, the SEC_CODE element in the response XML Data Packet form the GetCertificationTerminalSettings web method will include the SEC code used from the terminal ID provided. A definition of each of the SEC codes used by the Authorization Gateway can be found below.

- **Point-of-Purchase Entry (POP):** The Point-of-Purchase method of payment is for purchases made for goods or services in person by the consumer. These are non-recurring debit entries. A check reading device must be used to capture the routing number, account number, and check number from the source document (check). The source document cannot be previously used for any prior POP entry, and the source document must be voided and returned to the customer at the point-of-purchase. In addition a signed receipt must be obtained at the point-of-purchase and retained for 2 years from the settlement date. The "Authorization Requirements" section in the Authorization Gateway Specification document contains additional information on the receipt requirements.

- **Internet Initiated Entry (WEB):** An internet initiated entry is a method of payment for goods or services made via the internet.

- **Accounts Receivable Entry (ARC):** An accounts receivable entry is a check received in the U.S. Mail. A check reading device must be used to capture the routing number, account number, and check number from the source document (Check).

- **Back Office Conversion Entry (BOC):** A back office conversion entry is a payment for goods or services made at the point-of-purchase or a manned bill-payment location where the electronic check conversion occurs during back-office processing and not in the presence of the consumer. A check reading device must be used to capture the routing number, account number, and check number.

- **Prearranged Payment and Deposit Entry (PPD):** A prearranged payment and deposit entry is either a standing or single entry authorization where the funds are transferred to or from a consumers account.

- **Telephone Initiated Entry (TEL):** A telephone initiated entry is a payment for goods or services made with a single entry debit with oral authorization obtained from the consumer via the telephone.

- **Check 21 (C21):** Although not an SEC Code C21 is used to denote Check 21 transactions. Check 21 requires a check reading device capture the routing number, account number, and check number from the source document (Check) as well as capture images of both the front and back of the source document.

Beginning Authorization Gateway Development (Phase 2)

I'm Ready To Begin Development. Where Do I Start?

The best place to start is with determining your application architecture for interfacing with the Authorization Gateway. At this point you have already determined which published XSD(s) your XML data packets will be validated against, and you also know the URL for the corresponding XML template(s) for your schema(s). This leaves you with the following possibilities for creating your XML data packets that are sent to the Authorization Gateway:

1. You can load the XML template into an XML document object and use XPath to populate the elements and attributes.
2. You can use an XML Schema Definition Tool (such as Xsd.exe for .NET) to generate a class based on the published XSD, populate the class properties, and then serialize the object.
3. You can build your own XML document and use XPath to populate the elements and attributes.

We recommend that you leverage the published XSDs and XML templates and use either the first or second options when creating the data packets to be sent. The first option is the quickest option to implement, but the second option will provide you with a more object-orientated approach and potentially more flexibility within your host system.
We have also provided example request XML Data Packets to assist your integration team with getting started. A link to these examples can be found at the end of the "How to determine which XML Template to Use" section of the Authorization Gateway Specification document.

What Does It All Mean?

Now that you have determined the best way to create the XML data packets within your host system, we will begin to look at what each element and attribute means, and when to apply a given value. The Authorization Gateway Specification document provides a detailed description of each element and includes a text description of the regular expressions, data types, or enumerations that control the allowed data formats for each element as well as links to the published data type XSDs. Therefore in this document we will focus on what each value means and when to use it.

How Do I Identify My Data?

The specification for the Authorization Gateway XML Data Packet allows you to optionally identify your data in two distinct ways. Again, these settings are completely optional, but have been built in so that your host system can match a response from the Authorization Gateway with the original request. Since these identifiers are generated by the host system, the Authorization Gateway does not ensure that any optional identifiers contained in the XML Data Packet are unique. Rather the Authorization Gateway leaves the responsibility of determining if an identifier is unique to the host system. It is not required that optional identifiers are unique, but it is recommended. If an identifier is not unique it may become difficult for your host system to match responses or retrieve archived responses. In the examples provided in the Authorization Gateway Specification document, GUIDs have been used as optional identifiers. The use of GUIDs ensures uniqueness, but any value can be used as an identifier, including database identity column values. It is also important to note that if the implementation team determines an identifier needs to be unique, that it only needs to be unique for a specific terminal ID, but it can be unique across all terminal IDs for a given user.

As previously mentioned there are two distinct ways you can optionally identify the XML Data Packet. These are the REQUEST_ID attribute contained within the AUTH_GATEWAY element and the TRANSACTION_ID element.

The REQUEST_ID attribute is a unique identifier that is used to identify the overall data packet. When your data packet is received by the Authorization Gateway it is not only processed, it is also asynchronously stored along with the response. This was done so that if necessary the host system can invoke the GetArchivedResponse web method to request a previous response. The GetArchivedResponse web method accepts the REQUEST_ID as an input parameter and will return the corresponding response. It is important to note that the GetArchivedResponse is a production only web method, and can only be effectively used if the host system keeps track of and submits values in the REQUEST_ID attribute. The value in the REQUEST_ID attribute of the request data packet is also returned in the response data packet in the REQUEST_ID attribute of the RESPONSE element.

The TRANSACTION_ID element is a unique identifier that is used to identify a specific transaction. The value contained in the TRANSACTION_ID element is recorded by the Authorization Gateway, but is not used internally and cannot be used to request a specific transaction. The value in the TRANSACTION_ID element is however returned in the response data packet in the TRANSACTION_ID element within the parent AUTHORIZATION_MESSAGE element. This was done so that your host system can match the response for a specific transaction to an internal record in the host system.

Again, setting values for both the REQUEST_ID attribute and the TRANSACTION_ID element are optional, but if the implementation team determines there is a business need to match records

internal to the host system with Authorization Gateway responses or request archived responses; then these identifiers can be utilized to implement that functionality.

What Do The Different Identifiers Mean?

Each request XML Data Packet must contain a valid identifier for its schema. The identifier you use will change depending on the context of the transaction being sent. Your integration team will become more familiar with the different identifiers as you begin to work on each milestone. However, a list of all the valid identifiers can be found below, and the Authorization Gateway Specification document contains a definition and an example of the IDENTIFER element in the request XML Data Packet as well as a link to the XSD that defines the identifier data types.

- **Authorize (A):** This is used in schemas for ARC, BOC, POP, TEL, WEB, and Check 21 to indicate that an authorization is requested for the XML Data Packet being sent.

- **Void (V):** This is used in schemas for ARC, BOC, POP, TEL, WEB, and Check 21 to void a previously authorized transaction. However, it should be noted that transactions can only be voided on the same calendar day they were authorized.

- **Override (O):** This is used in schemas for ARC, BOC, POP, TEL, WEB, and Check 21 when the host system receives a manager needed message to void the previous transaction and input a new transaction in its place.

- **Payroll (P):** This is used in schemas for ARC, BOC, POP, and Check 21 for business and payroll checks. What this does is NOT link the driver's license to the routing/ account numbers since the person writing/cashing the check is usually not the business.

- **Recurring (R):** This is used in schemas for PPD to indicate reoccurring transactions.

What Does Verification Only Mean?

If the gateway terminal is setup as verification only or the VERIFICATION_ONLY element is set to true, then the transaction will be processed as verification only. This means that an authorization will be run, but that the check will not undergo Electronic Check Conversion (ECC) and that the check will have to be taken to the bank for deposit. In addition, depending on the merchants program, the funds may or may not be guaranteed.

What Do I Need to Provide In The Account Section?

All PPD, TEL, WEB, and Check 21 (C21) schemas define what ACCOUNT child elements must contain values and what ACCOUNT child elements can be left empty. However, all of the child elements within the ACCOUNT element, except the ACCOUNT_TYPE, for the ARC, BOC, and POP schemas define the data as optional. This is because for these SEC codes you can either provide the swiped MICR data or provide the routing, account, and check numbers. If the MICR_DATA, ROUTING_NUMBER, ACCOUNT_NUMBER, and CHECK_NUMBER are all left empty in the request data packet then the transaction cannot be processed. Either the MICR_DATA or the ROUTING_NUMBER, ACCOUNT_NUMBER, and CHECK_NUMBER elements must contain values.
It is important to note that if the swiped MICR data in the MICR_DATA element is missing, but the ROUTING_NUMBER, ACCOUNT_NUMBER, and CHECK_NUMBER elements contain values then the transaction will be processed as verification only; even if the CONTROL_CHAR indicates that the information was retrieved from a check reader. In addition, if the MICR_DATA, ROUTING_NUMBER, ACCOUNT_NUMBER, and CHECK_NUMBER elements all contain values, then the Authorization Gateway will only use the information in the MICR_DATA element and will parse it out overwriting any values sent in the ROUTING_NUMBER, ACCOUNT_NUMBER, and CHECK_NUMBER elements.

When Do I Need To Include Identity Information?

Identity information needs to be included when the terminal is setup to do identity verification. There are schemas that will handle the validation for terminals that are setup to do identity verification, and the GetCertificationTerminalSettings web method will return a response of “true” in the RUN_IDENTITY_VERIFICATION element. If a terminal is setup to do identity verification then the host system is required to send either the last 4 of the check writers social security number or their birth year.

Development Phase Milestones

There are several milestones that need to be completed in the Development Phase prior to beginning the Certification Phase. These milestones are designed to help you with integrating the Authorization Gateway into your host system and with understanding the business processes for handling the returned responses. For development and certification there are set of routing numbers that will generate fixed responses. This will allow your integration team to continuously refine the development process with guaranteed reproducible results from the Authorization Gateway for each milestone.

To begin development for processing a single check you will need to invoke the AuthGatewayCertification web method. This web method will accept an XML Data Packet and will return an XML response. Again, the specifications for request and response Data Packets are defined in the Authorization Gateway Specification document. As you proceed with the Development Phase of the integration effort please reference the Authorization Gateway Specification document to view

example response XML Data Packets and to gain a more detailed definition of each element and attribute within the response.

Validation Handling

When the AuthGatewayCertification web method receives a request it will first validate your request XML Data Packet against the published XSD for your terminal. Each returned response will include a VALIDATION_MESSAGE element. If the request XML Data Packet successfully passes validation the RESULT child element of the VALIDATION_MESSAGE element will contain a value of "Passed", but if the validation failed, the RESULT element will contain a value of "Failed". These values can be coded into your host system for determining if a request passed or failed validation. The VALIDATION_MESSAGE element will also contain a SCHEMA_FILE_PATH element. The SCHEMA_FILE_PATH element will be present regardless of if the request XML Data Packet passed or failed validation and will include the full URI for the XSD that was used for validating the request XML Data Packet. In addition, if the RESULT element contains "Passed" then only the RESULT and SCHEMA_FILE_PATH elements will be present as child elements of the VALIDATION_MESSAGE. However, if the request XML Data Packet fails validation, and the RESULT element contains a value of "Failed", then the VALIDATION_MESSAGE will contain one or more VALIDATION_ERROR elements. The VALIDATION_ERROR element will contain SEVERITY and MESSAGE elements that will detail exactly what failed in the request XML Data Packet as well as LINE_NUMBER and LINE_POSITION attributes that will define exactly where the validation error occurred.

The host system should always check each response to make sure the RESULT child element of the VALIDATION_MESSAGE is set to "Passed". If it is not then there are validation errors and the transaction was not processed. The host system will have to correct any validation errors outlined in the VALIDATION_ERROR element(s) and then resubmit the request XML Data Packet.

Process Single Certification Check – Authorization

Now that your integration team has been able to successfully create and send a valid request XML Data Packet we can move forward and begin to process single certification checks. To do this you will need to invoke the ProcessSingleCertificationCheck web method. This web method accepts the same request XML Data Packet as an input parameter as the AuthGatewayCertification web method did. Only this time after your request XML Data Packet passes validation, it will be processed.

However before we begin it is important to take a look at the response XML Data Packet and to understand what each element means and what information your host system should use to interpret the response from the Authorization Gateway. The Authorization Message Response section of the Authorization Gateway Specification document provides an example and a definition for each element, so in this document we are going to focus on how to use the values contained in each element.

The RESULT_CODE element contains a numeric bit that indicates one or many result messages. The host system should conduct a bit comparison of the RESULT_CODE to determine exactly how the transaction was processed and from there can determine exactly what action to take if any.

The RESPONSE_TYPE contains an identifier that will give your host system a general overview of the processed transaction, and the RESPONSE_TYPE_TEXT element will contain the full text description of the identifier contained in the RESPONSE_TYPE element. The RESULT_CODE should be the primary driver for determining how the host system should act in response to various responses from the Authorization Gateway, but the values in the RESPONSE_TYPE can be used to determine additional information for processing by the host system. This includes determining if a transaction was processed as Verification Only.

The TYPE_CODE element contains a numeric bit that indicates one or many type messages. The host system should conduct a bit comparison of the TYPE_CODE element to determine additional detailed information about the transaction which can be provided to the user.

The CODE element contains a text message with the Authorization Number if the transaction was approved or other additional information if the transaction was not approved. The CODE element should be used by the host system to display and record the any authorization numbers or additional information.

The MESSAGE element contains additional text, and should be used by the host system to display and record any additional information about the transaction.

The TOKEN element contains the return Token used for the transaction. The Token is used in place of Account Type, Routing Number, and Account Number. NOTE: This is only available when using a Token or when requesting a Token.

Now that we have covered how each element should be used, we can begin with processing a single certification check with an authorization response. It is important to note that as your integration team works through each of the following milestones that you will be coding your host system to react to fixed responses. Again, this was done so that you can continuously refine the integration of the Authorization Gateway with your host system with reproducible results, but once you move into the production, any response may be returned to the host system.

When processing a single certification check for Authorization you will need to invoke the ProcessSingleCertificationCheck web method and set the routing number to 490000018 in the ROUTING_NUMBER element of the request XML Data Packet. You will also have to set the value of the IDENTIFER element to “R” if you are using a PPD schema or “A” for all other schemas. If the request XML Data Packet is valid then this routing number will trigger the Authorization Gateway to return a response with the following information to the host system:

- **RESPONSE_TYPE:** A
- **RESPONSE_TYPE_TEXT:** APPROVED
- **RESULT_CODE:** 0
- **TYPE_CODE:** 4096
- **CODE:** AUTH NUM 272-172
- **MESSAGE:** APPROVAL

Again, the host system should first check to make sure the RESULT child element of the VALIDATION_MESSAGE is set to "Passed". If the request XML Data Packet passed validation it was successfully processed and the above elements will be present as child elements of the AUTHORIZATION_MESSAGE element. The host system should store all of the returned data and at a minimum conduct a bit comparison of the value in the RESULT_CODE element. If the value in the RESULT_CODE is 0 then the transaction has been approved. The response Data Packet also contains a value of 4096 for the TYPE_CODE element which indicates an Internal Override which in this instance means that Authorization Gateway returned a predetermined fixed response. A link to the XSD that defines all of the Authorization Gateway Response Types can be found in the Data Types section of the Authorization Gateway Specification document.

In addition to displaying and recording the response the host system is also required to generate a receipt for point-of-purchase (POP) entries. The Authorization Requirements section of the Authorization Gateway Specification document details these requirements for POP terminals.

Process Single Certification Check – Check Limit Exceeded

The check limit for processing single certification checks is $25. If a check amount in excess of $25 is sent to the Authorization Gateway during development or certification phases then the Authorization Gateway will return "Check Limit Exceeded - Decline".

When processing a single certification check for Check Limit Exceeded you will need to invoke the ProcessSingleCertificationCheck web method and set the routing number to 490000018 in the ROUTING_NUMBER element of the request XML Data Packet. You will also have to set the value of the of the IDENTIFER element to "R" if you are using a PPD schema or "A" for all other schemas. Finally you will need to include a check amount larger than $25 in the CHECK_AMOUNT element. If the request XML Data Packet is valid then this will trigger the Authorization Gateway to return a response with the following information to the host system:

- **RESPONSE_TYPE:** D
- **RESPONSE_TYPE_TEXT:** DECLINED
- **RESULT_CODE:** 136
- **TYPE_CODE:** 256
- **CODE:** DECLINE CHECK LIMIT EXCEEDED

- **MESSAGE:** DECLINE CHECK LIMIT EXCEEDED

If a transaction is declined the Authorization Gateway will return an 8 in the RESULT_CODE element which indicates a Decline Message. In the returned response above the RESULT_CODE element has a value of 136. If the host system is setup to do a bit comparison of the value in the RESULT_CODE you will discover that the 136 is made of an 8 indicating a Decline Message and 128 which indicates that the transaction limit has been exceeded. The fixed decline response also contains a value of 256 in the TYPE_CODE element. This indicates that the host system was unable to determine if this was the first time the check was presented or if it is a representment.

Process Single Certification Check – Decline

When processing a single certification check for Decline you will need to invoke the ProcessSingleCertificationCheck web method and set the routing number to 490000034 in the ROUTING_NUMBER element of the request XML Data Packet. You will also have to set the value of the IDENTIFER element to "R" if you are using a PPD schema or "A" for all other schemas. If the request XML Data Packet is valid then this routing number will trigger the Authorization Gateway to return a response with the following information to the host system:

- **RESPONSE_TYPE:** D
- **RESPONSE_TYPE_TEXT:** DECLINED
- **RESULT_CODE:** 520
- **TYPE_CODE:** 4100
- **CODE:** DECLINE CHECK 5 UNPAID (ALL) AMT=$5478 GLOBAL eTELECOM 888-481-0757
- **MESSAGE:** DECLINE CHECK 5 UNPAID (ALL) AMT=$5478 GLOBAL eTELECOM 888-481-0757

If a transaction is declined the Authorization Gateway will return an 8 in the RESULT_CODE element which indicates a Decline Message. In the returned response above the RESULT_CODE element has a value of 520. If the host system is setup to do a bit comparison of the value in the RESULT_CODE you will discover that the 520 is made of an 8 indicating a Decline Message and 512 which indicates that the unpaid check Limit has been exceeded. The fixed decline response also contains a value of 4100 in the TYPE_CODE element. This again indicates an internal override was done because the Authorization Gateway is returning a predetermined fixed response. However, if the host system is setup to conduct a bit comparison on the value of the TYPE_CODE element it can also be determined that TYPE_CODE also contains a 4, which indicates a Business Check was sent for processing.

Process Single Certification Check – Manager Needed

When processing a single certification check for Manager Needed you will need to invoke the ProcessSingleCertificationCheck web method and set the routing number to 490000021 in the ROUTING_NUMBER element of the request XML Data Packet. You will also have to set the value of the IDENTIFER element to "R" if you are using a PPD schema or "A" for all other schemas. If the

request XML Data Packet is valid then this routing number will trigger the Authorization Gateway to return a response with the following information to the host system:

- **RESPONSE_TYPE:** W
- **RESPONSE_TYPE_TEXT:** Warning
- **RESULT_CODE:** 132
- **TYPE_CODE:** 4100
- **CODE:** MANAGER NEEDED CHECK TOO LARGE
- **MESSAGE:** MANAGER NEEDED CHECK TOO LARGE

If a transaction is returned with a warning message the RESULT_CODE element will contain a 4. In the above response message a bit comparison will show that the RESULT_CODE element also contains a 128 which indicates that the transaction limit has been exceeded. The host system should be setup to recognize warning messages and any other additional information contained within the RESULT_CODE element. In this case the MESSAGE element indicates that the Manager is needed because the check is too large. The TYPE_CODE again shows that 4096 was returned indicating that there was an internal override due to a predetermined fixed response being returned as well as a 4 indicating a Business Check was sent for processing.

If the host system receives a warning message back and indicates “MANAGER NEEDED” and you are not doing PPD, then you have the option of sending an override request packet back to the Authorization Gateway. The override request is created by sending the same request XML Data Packet back to the Authorization Gateway. However, you must change the value of the IDENTIFIER element to “O”. You also have the option of changing the values of the TRANSACTION_ID element and REQUEST_ID attribute so that your host system can record and track the override request as a separate transaction. In this instance you also have the option of changing the value in the CHECK_AMOUNT element. Again, identifying a request XML Data Packet as an override will void the previous transaction and input a new transaction in its place, and your host system will receive an authorization in return.

Process Single Certification Check – Represented Check

When processing a single certification check for Re-Presented Check you will need to invoke the ProcessSingleCertificationCheck web method and set the routing number to 490000047 in the ROUTING_NUMBER element of the request XML Data Packet. You will also have to set the value of the IDENTIFER element to “R” if you are using a PPD schema or “A” for all other schemas. If the request XML Data Packet is valid then this routing number will trigger the Authorization Gateway to return a response with the following information to the host system:

- **RESPONSE_TYPE:** W
- **RESPONSE_TYPE_TEXT:** Warning
- **RESULT_CODE:** 4

- **TYPE_CODE:** 4228
- **CODE:** MANAGER NEEDED REPRESENTED CHECK
- **MESSAGE:** MANAGER NEEDED REPRESENTED CHECK

If a transaction is returned with a warning message the RESULT_CODE element will contain a 4. In this case the MESSAGE element indicates that the Manager is needed because this is a represented check. The TYPE_CODE in this response contains a lot of information. A bit comparison will show that the value of 4228 in the TYPE_CODE element contains 128 which indicates a represented check as well as 4096 indicating that there was an internal override due to a predetermined fixed response being returned, and a 4 indicating a Business Check was sent for processing. The host system should be able to recognize that a warning message was received from the Authorization Gateway and that it was a represented check.

If the host system receives a warning message back and indicates "MANAGER NEEDED" and you are not doing PPD, then you have the option of sending an override request packet back to the Authorization Gateway. The override request is created by sending the same request XML Data Packet back to the Authorization Gateway. However, you must change the value of the IDENTIFIER element to "O". You also have the option of changing the values of the TRANSACTION_ID element and REQUEST_ID attribute so that your host system can record and track the override request as a separate transaction. Again, identifying a request XML Data Packet as an override will void the previous transaction and input a new transaction in its place, and your host system will receive an authorization in return.

Process Single Certification Check – Void

When voiding a previously approved single certification check you will need to invoke the ProcessSingleCertificationCheck web method and set the routing number to 490000018, 490000021, or 490000047 in the ROUTING_NUMBER element of the request XML Data Packet. You will also have to set the value of the IDENTIFER element to "V". This milestone has been built into the development phase so that you can incorporate this functionality into your host system. If the request XML Data Packet is valid then a Void identifier for previously approved transaction with the routing numbers noted above will trigger the Authorization Gateway to return a response with the following information to the host system:

- **RESPONSE_TYPE:** A
- **RESPONSE_TYPE_TEXT:** APPROVED
- **RESULT_CODE:** 0
- **TYPE_CODE:** 5120
- **CODE:** VOID ACCEPTED
- **MESSAGE:** VOID ACCEPTED

You should note that the returned information for a voided transaction contains a RESULT_CODE of 0. This indicates that the void was approved, but it also illustrates the importance of examining the

information contained with the TYPE_CODE. If the host systems interface with the Authorization Gateway was only set to interpret the RESULT_CODE the full meaning of the overall response would be lost. In this case the TYPE_CODE returned in the response XML Data Packet contains 5120. A bit comparison of this value indicates that the value contains 1024 indicating a voided check, and 4096 indicating that there was an internal override due to a predetermined fixed response being returned.

Exception Handling

Incorporating exception handling into your host system for the Authorization Gateway interface is important so that the host system can check to ensure that nothing unexpected occurred during processing. The "Exceptions" section of the Authorization Gateway Specification has examples and defines the EXCEPTION element, but there are basically two reasons the Authorization Gateway may return an exception in the response XML Data Packet.

Although the Authorization Gateway has been rigorously tested it is possible that an internal error may occur. If this happens the Authorization Gateway will return an EXCEPTION element with a message of "An internal error occurred. The Transaction was NOT processed". If this exception is return to the host system our software team will be immediately notified with detailed information about the problem, and will work to correct the issue. We work hard to ensure these types of exceptions do not occur, but your integration team should understand what internal errors mean, how they are handled, and configure the host system to take appropriate action.

The Authorization Gateway may also return exception messages if there are authentication, authorization, or data related errors. The message for these types of exceptions will vary, but the host system will receive a detailed message within the EXCEPTION element that outlines exactly what the problem was. These types of exceptions are built into the Authorization Gateway by design and the Authorization Gateway relies on the host system to resolve the issue.

We expect that your integration team has included at least a minimal level of exception handling into your host system prior to beginning the Certification Phase.

Requesting a Certification Date

Requesting a certification date is the major milestone of the Development Phase. It signifies that your integration team has completed the integration effort and alerts our software team that the host system is ready to undergo certification. It is important that your integration team contact us to request a certification date. If a certification date is not requested, but you begin the Certification Phase, our software team will not be able to properly certify your host system and your team will have to rerun the certification script prior to moving to the Production Phase.

Beginning Certification (Phase 3)

During the certification phase your integration team will be responsible for sequentially completing the objectives in the certification script below. Your team should now be intimately familiar with the

Authorization Gateway and the host system should now be able to handle the completion of these objectives without any problems. During the certification phase our software team will closely monitor each transaction to ensure it is valid, and that the host system is properly configured. As each objective is run we will alert you to the status of the transaction in our system, and advise you if there are any modifications that need to be made to the host system. The successful completion of each objective outlined in the certification script signifies the completion of the major milestone for the Certification Phase and marks the opportunity to begin the Production Phase.

Certification Script

Objective	Routing Number	Completed
Process Single Certification Check: Authorization	490000018	
Process Single Certification Check: Decline	490000034	
Process Single Certification Check: Manager Needed	490000021	
Process Single Certification Check: Send Override for Manager Needed	490000021	
Process Single Certification Check: Represented Check	490000047	
Process Single Certification Check: Send Override for Represented Check	490000047	
Process Single Certification Check: Void previously approved transaction	490000018	

Migrating to Production (Phase 4)

The Production Phase is the final phase of the integration effort. During this phase you will have to make some minimal changes to the host system in order to use the Authorization Gateway in a production environment. This includes the following:

- You will need to request a user name and password for production. This user name and password will be different from the user name and password provided for certification and will only be valid for production. The host system will then need to be modified to include this user name and password in the authentication header when invoking a production web method.
- You will need to request the production URL for the Authorization Gateway. This URL will be different then the URL used for certification, however it will contain identical web methods. The host system will need to be modified to invoke web methods on the production URL.
- You will also need to change the certification web methods listed below to their sister production web methods.

Certification Web method	Production Web Method
GetCertificationTerminalSettings	GetTerminalSettings
ProcessSingleCertificationCheck	ProcessSingleCheck

- Once the host system has been modified to include these changes for processing transactions in a production environment you will need to request a “Go Live” date. Requesting a “Go Live” date signifies completion of the last major milestone of the integration effort and indicates to our software team that the host system is ready for production.

Contact Information

For questions or to receive certification and live username/passwords and URLs please contact:

Integration Department
integration@globaletelecom.com

Document History

Version Number	Modification Date	Modification
1.0	08/01/2008	Created
1.1	04/08/2010	Corrected verbiage
1.2	12/14/2011	Updated Contact Information
1.3	04/17/2012	Added token information
1.4	07/02/2012	Removed outdated verbiage for PPD voids

877-454-3835
www.GlobaleTelecom.com
www.CheckTraining.com
Sales@GlobaleTelecom.com