Indiana Health Coverage Programs

EDI Healthcare Transaction Connectivity Guide

Version Number: 3.4
Revision Date: August 2019

Table of Contents

1 INTRODUCTION

This document contains technical information about the Indiana Health Coverage Programs (IHCP) Electronic Data Interchange (EDI) connectivity options for exchanging HIPAA compliant transactions and is intended to assist vendors with developing applications to exchange EDI HIPAA compliant transactions with the IHCP.

1.1 SCOPE

The guide provides technical information regarding initiating and maintaining connectivity, sending and receiving files and other related information for exchanging EDI transactions. This includes information about data transmission protocols, envelope methods and security standards that the IHCP supports.

2 SUBMITTER AUTHENTICATION AND SECURITY

The IHCP requires authentication using a User ID and Password. The Trading Partner must be authorized to exchange the transactions they are attempting to send and receive. This involves using the Trading Partner ID / Submitter ID to ensure the Trading Partner has been authorized to exchange the transactions.

For 835 retrieval requests, the Trading Partner must ensure that the provider has delegated the Trading Partner to download the HIPAA ERA/835 on their behalf.

The IHCP requires Trading Partner's to adhere to the password requirements including changing passwords every 90 days. The system will prompt the Trading Partner to change the password beginning 5-days before expiration at every log-in attempt. The Trading Partner will not be able to log-in until the password is changed. This may cause any automated connection scripts to fail. When the password is manually changed, the same change must be applied to any automated scripts to ensure uninterrupted service. The account will be deactivated at 90-days if the password is not changed. The Trading Partner will need to contact Trading Partner support at INXIXTradingPartner@dxc.com or 1-800-457-4584, option 3, then option 2 to reactivate the account. Passwords are reset on the SFTP server at https://sftp.indianamedicaid.com

3 SUPPORTED TRANSACTIONS

The IHCP supports the following transactions using the above connectivity methods.

Transaction	Web Services Batch and Interactive	File Exchange Batch
837I Health Care Claim Institutional		x
837P Health Care Claim Professional		x
837D Health Care Claim Dental		x
835 Remittance Advice (ERA)	x	x
270/271 Eligibility Benefit Inquiry and Response	x	x
276/277 Claim Status Request and Response	x	x
278 Prior Authorization (PA) Request for Review and Response		x

834 Managed Care Member Enrollment Roster		x
820 Managed Care Capitation Payment Reporting		x

4 INTERFACE OVERVIEW

The IHCP connectivity interfaces support the most commonly used channels of communication, giving clients a variety of interfaces to develop robust interchange solutions.

The IHCP EDI interfaces support the following connectivity methods:

- SFTP - MOVEit Server (File Exchange)
- Web Services
  - HTTP MIME Multipart
  - SOAP + WSDL

4.1 SFTP MOVEIT SERVER (FILE EXCHANGE)

The SFTP MOVEit Server (File Exchange) is an interface provided by the IHCP for secure batch transaction file exchanges. File Exchange is designed to safely and securely collect, store, manage and distribute sensitive information between IHCP and Trading Partners. Web browsers and no- or low-cost secure FTP clients, who are required to transfer files can securely exchange files using File Exchange over encrypted connections using the FTP over SSL (FTPS), FTP over SSH (SFTP) and HTTP over SSL (HTTPS) protocols. Typically, Trading Partners that wish to interact systematically with File Exchange via a batch script will choose one of the two FTP methods listed above.

All HIPAA compliant transaction batch files must be uploaded to the /Distribution/HIPAA Transactions directory. All filenames must adhere to the file naming convention policy listed below. Files that do not adhere to the policy will receive an error message during the upload process to File Exchange and will not successfully upload until corrected. In addition to accepting normal text files, File Exchange can also accept compressed files submitted in a zip format. The file name that is submitted must have a .zip at the end of the file name. Any data contained within the zipped file must contain a valid three character file extension. Zipped files must not contain directory folders or structures and must contain only one (1) file. Zipped files that contain multiple files will not process correctly. Only one file will process and the others will be ignored.

Inbound File Naming Convention Policy:

1. All inbound filenames must have an extension. For example: <filename>.txt or <filename>.X12
2. All inbound filenames must not contain invalid characters from the list below:
   / / " ' < > | : ? * , { } [ ] ~ $ @ ( ) # & ^ ! % = + ; `
3. All inbound filenames must not contain any spaces

All outbound files available for download are created individually with the following naming conventions:

999	**<AckFileID>_<TransmissionID>_<OriginalX12FileName>_<TrackFileID>.999.txt**
TA1	**<AckFileID>_<TransmissionID>_<OriginalX12FileName>_<TrackFileID>.TA1.txt**
271	**<RespFileID>_<TransmissionID>_<OriginalX12FileName>_<TrackFileID>.271X12BATCH.dat**
277	**<RespFileID>_<TransmissionID>_<OriginalX12FileName>_<TrackFileID>.277X12BATCH.dat**
277U	**<RespFileID>_<TransmissionID>_<TradingPartner ID>_277UX12BATCH_CCYYMMDD.dat**
278	**<RespFileID>_<TransmissionID>_<OriginalX12FileName>_<TrackFileID>.278OX12BATCH.dat**
835	**<TradingPartner ID>.835.X.HHMMSS.JulianDate.dat**

Outbound files are not available in a compressed or zip format. All outbound files will be placed in the Trading Partner's home directory. These files will remain available for retrieval for 30 days after they first become available unless they are explicitly deleted from File Exchange by the Trading Partner.

4.1.2 FTP OVER SSL

The following information is provided to assist developers with customizing secure FTP clients using FTP over SSL to enable connectivity to File Exchange. File Exchange fully supports a large number of secure FTP clients using FTP over SSL including the following:

- MOVEit Freely (free command-line)
- MOVEit Automation (Central) 2017 (w/Admin)
- WS_FTP Professional and WS_FTP Home (GUI, version 7 and higher, Windows) (version 12 and higher)
- SmartFTP (GUI, version 1.6 and higher, Windows)
- SmartFTP (free GUI, version 1.0 and higher, Windows)
- Cute FTP Pro (GUI, version 1.0 and higher, Windows)
- BitKinex (GUI, version 2.5 and higher, Windows)
- Glub FTP (GUI, Java 2.0 and higher)
- FlashFXP (GUI, version 3.0 and higher)
- IP*Works SSL (API, Windows, version 5.0)
- LFTP (free command-line, Linux, Unix, Solaris, AIX, etc.)
- NetKit (command-line, Linux, Unix, Solaris, etc.)
- SurgeFTP (command-line, FreeBSD, Linux, Macintosh, Windows, Solaris)
- C-Kermit (command-line; v8.0+, AIX, VMS, Linux, Unix, Solaris)
- AS/400 native FTPS client (OS/400 minicomputer)
- z/OS Secure Sockets FTP client (z/OS mainframe)
- TrailBlaxer ZMOD (OS/400 minicomputer)
- NetFinder (GUI, Apple)
- Sterling Commerce (batch, various)
- Tumbleweed SecureTransport (4.2+ on Windows, batch, various)
- Cleo Lexicom (batch, various)
- bTrade TDAccess (batch, AIX, AS/400, HP-UX, Linux, MVS, Solaris, Windows)
- cURL (command-line, AIX, HP-UX, Linux, QNX, Windows, AmigaOS, BeOS, Solaris, BSD and more)
- South River Technologies "WebDrive" (Windows "drive letter" - requires "passive, implicit and 'PROT P'" options)
- Stairways Software Pty Ltd. "Interarchy" (Mac "local drive" and GUI)

Configurations that are needed no matter which FTP over SSL client is used are listed below. Consult the documentation for your specific FTP client to determine how to configure these settings. If the machine that is initiating the FTP connection resides behind a firewall, the firewall must be configured to allow outbound traffic on any of the ports listed below.

- Host – https://sftp.indianamedicaid.com

- Control Connection Ports
  - 990 if using implicit encryption (recommended)
  - 21 if using explicit encryption

- Data Connection Ports – 3000-3100 (any port in this range)
- Transfer mode – Passive (Active mode transfers will not be accepted)

4.1.2 FTP OVER SSH

The following information is provided to assist developers with customizing secure FTP clients using FTP over SSH to enable connectivity to File Exchange. File Exchange fully supports a large number of secure FTP clients using FTP over SSH including the following:

OpenSSH SFTP (command-line, Unix)
OpenSSH for Windows (command-line, Windows)
PSFTP/PSCP (command-line, Windows)

Configurations that are needed no matter which FTP over SSL client is used are listed below. Consult the documentation for your specific FTP client to determine how to configure these settings. If the machine that is initiating the FTP connection resides behind a firewall, the firewall must be configured to allow outbound traffic on any of the ports listed below.

- Host – https://sftp.indianamedicaid.com
- Port-22 (this is the default SSH port)

4.1.3 MANUAL CONNECTIVITY

If a Trading Partner intends to interact with File Exchange in a manual, or ad-hoc manner, the HTTPS method using the internet is available. The IHCP secure File Exchange website is located at https://sftp.indianamedicaid.com. All Trading Partners can log on to File Exchange using the same ID and password that is used to access File Exchange in the FTP methods listed above. Accessing File Exchange via the internet allows each trading partner to pick up or drop off files outside of an automated script.

4.2 WEB SERVICES

The IHCP Web Service Connection is built around CAQH CORE Operating Rules found at http://caqh.org/CORE_operat_rules.php.

The IHCP Web Services are capable of exchanging the 270/271 and 276/277 batch and interactive transactions and returning the 835 remittance advice transaction using either of the two HTTP/S envelope standards: MIME-Multipart form data and SOAP + WSDL. Each envelope must conform to the CAQH CORE requirements as listed in the CORE Phase I and Phase II rules referenced above for envelope version 2.2.0.

4.2.1 CONNECTING TO THE SERVER

Trading Partners can connect to the web services using a network connection that provides access to the public internet.

Production URLs:

- SOAP:
  https://coresvc.indianamedicaid.com/HP.Core/CoreTransactions.svc
- MIME-Multipart:
  https://coresvc.indianamedicaid.com/HP.Core.Mime/CoreTransactions.aspx

4.2.2 PAYLOAD TYPES

Interactive Payload Types

Request Payload Types (sent by trading partner)
•X12_270_Request_005010X279A1
•X12_276_Request_005010X212

Response Payload Types (returned by IHCP)
•X12_271_Response_005010X279A1
•X12_277_Response_005010X212

Batch Payload Types

Batch Upload Request Payload Types (sent by trading partner)
•X12_270_Request_005010X279A1
•X12_276_Request_005010X212
•X12_999_SubmissionRequest_005010X231A1
This payload type only logs that the submitter is acknowledging receipt of a requested download

Batch Upload Response Types (returned by IHCP)
•X12_Response_ConfirmedReceiptReceived
•For the submission of an X12_999_SubmissionRequest_005010X231A1
•X12_BatchReceiptConfirmation
•For the submission of an X12_270_Request_005010X279A1 and X12_276_Request_005010X212

Batch Download Request Payload Types (sent by trading partner)
•X12_005010_Request_Batch_Results_271
•X12_005010_Request_Batch_Results_277
•X12_005010_Request_Batch_Results_835
•X12_999_RetreivalRequest_005010X231A1

Batch Download Response Payload Types (returned by IHCP)
•X12_271_Response_005010X279A1
•X12_277_Response_005010X212
•X12_835_Response_005010X221A1
•X12_999_RetreivalRequest_005010X231A1
•X12_005010_Response_NoBatchResultsFile

4.2.3 TRANSACTION FORMAT AND CONTENT

This section provides information on the content and format used by each of the envelope standards supported by the IHCP.

4.2.3.1 HTTP MIME MULTI-PART

The HTTP MIME Multi-part consist of two sections. The first section is the standard HTTP Header and the second section is the MIME multi-part data. The Multi-part data includes the certain required data elements to authenticate the user and the actual v5010 X12 real-time or batch transactions.

HTTP MIME Inbound Header

The HTTP Header contains the information to identify the type of protocol being employed. The HTTP Header is included on every transaction submitted to the IHCP. An example of this header shown below.

```
POST /core/eligibility HTTP/1.1
Host: server_host:server_port
```

```
Content-Length: 2408
Content-Type: multipart/form-data;
boundary=XbCY
```

The contents of this header inform the IHCP as to the intent of the attached transaction data. The following is a description of the data contained in the header.

- Line 1 – POST command: To send data to the transaction on the server (/core/eligibility) using the HTTP 1.1 protocol.
- Line 2 – HOST definition: Specifies the URL and Port to use when sending the command.
- Line 3 – Content Length: The length in bytes of the entire transaction (including HTTP information and payload).
- Line 4 – Content Type: Specifies that this transaction is a multi-part message containing form data. Also identifies the boundary characters that delineate a new field of form (XbCY in the example)

HTTP MIME Multi-part Data

Following the HTTP Header is a section that contains the data associated with the transaction. This includes identifying information for the submitter and receiver, as well as the actual transaction data. An example of this section is shown below. The shaded area is the HTTP header data previously described.

```
POST /core/eligibility HTTP/1.1
Host: server_host:server_port
Content-Length: 2408
Content-Type: multipart/form-data;
boundary=XbCY
--XbCY
Content-Disposition: form-data;
name=“PayloadType“
X12_270_Request_005010X279A1
--XbCY
Content-Disposition: form-data;
name=“ProcessingMode"
RealTime
--XbCY
Content-Disposition: form-data;
name=“PayloadID"
e51d4fae-7dec-11d0-a765-00a0c91e6da6
--XbCY
Content-Disposition: form-data;
name=“TimeStamp"
2018-01-01T10:20:34Z
--XbCY
Content-Disposition: form-data;
name=“UserName"
hospa
--XbCY
Content-Disposition: form-data; name=“Password"
8y6dt3dd2
--XbCY
Content-Disposition: form-data; name=“SenderID"
HospitalA
--XbCY
Content-Disposition: form-data; name=“ReceiverID"
IHCP
--XbCY
Content-Disposition: form-data;
name=“CORERuleVersion"
2.2.0
--XbCY
Content-Disposition: form-data; name=“Payload"
<contents of file go here -- 1674 bytes long as
specified above>
--XbCY--
```

The above example defines multiple data elements that would be used when processing the transaction to authenticate the submitter and ensure the attached transaction should be processed.
The Web Service processing this transaction will extract these data elements to use for processing.
Data element “Payload Type” indicates the type of transaction that is attached to this MIME message.

4.2.3.2 SOAP + WSDL MESSAGE FORMAT

In addition to the HTTP MIME formats specified in the preceding sections the IHCP also supports those same transactions submitted using the SOAP + WSDL format. The XSD and WSDL formats are available at the following locations.

- XSD - https://www.caqh.org/sites/default/files/core/wsdl/CORERule2.2.0.xsd?token=4oHOkzHO
- WSDL - http://www.caqh.org/SOAP/WSDL/CORERule2.2.0.wsdl

Like their MIME counterparts, the WSDL transactions will be transmitted using the HTTPS protocol. The HTTP header format will be slightly different to specify the SOAP version 1.2 transaction protocol instead of the MIME Multi-part protocol.

The notable difference with MIME is the “Envelope Processing Error” message. Under SOAP, the envelope errors are returned as a SOAP error and thus do not have a defined message or transaction.

SOAP + WSDL Header

The following is an example HTTP header for WSDL transactions.

```
POST /core/eligibility HTTP/1.1
Host: server_host:server_port
Content-Type: application/soap+xml;charset=UTF-
8;action="http://tempuri.org/CoreTransactions/RealTimeTransactionReq
uest"
```

The contents of this header informs the IHCP as to the intent of the attached transaction data. The following is a description of the data contained in the header.

- Line 1 – POST command: To send data to the transaction on the server (/core/eligibility) using the HTTP 1.1 protocol.
- Line 2 – HOST definition: Specifies the URL and Port to use when sending the command.
- Line 3 – Content Type: Specifies that this transaction is a SOAP XML message using the UTF-8 character set. It also specifies the action/process that needs to be invoked (RealTimeTransactionRequest in this case).

SOAP + WSDL Data Elements

The Real-Time Request message is used to submit a single request for a single individual over a single period of time.

The following is a sample of a Real-Time transaction. The gray shaded area represents the standard HTTP header, the green section represents the SOAP+WSDL envelope data and the blue shaded section contains the actual transaction and payload data.

```
POST /core/eligibility HTTP/1.1
Host: server_host:server_port
Content-Type: application/soap+xml;charset=UTF-
8;action="http://tempuri.org/CoreTransactions/RealTimeTransactionRequest"
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:cor="http://www.caqh.org/SOAP/WSDL/CORERule2.2.0.xsd">
    <soap:Header>
       <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-
       200401-wss-wssecurity-secext-1.0.xsd">
           <wsse:UsernameToken wsu:Id="UsernameToken-1"
           xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
           wssecurity-utility-1.0.xsd">
              <wsse:Username>USER ID</wsse:Username>
              <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-
              200401-wss-username-token-profile-
              1.0#PasswordText">PASSWORD</wsse:Password>
           </wsse:UsernameToken>
       </wsse:Security>
    </soap:Header>
    <soap:Body>
       <cor:COREEnvelopeRealTimeRequest>
           <PayloadType>X12_270_Request_005010X279A1</PayloadType>
           <ProcessingMode>RealTime</ProcessingMode>
           <PayloadID>PAYLOAD ID</PayloadID>
           <TimeStamp>DATE/TIMESTAMP</TimeStamp>
           <SenderID>TRADING PARTNER ID</SenderID>
           <ReceiverID>IHCP</ReceiverID>
           <CORERuleVersion>2.2.0</CORERuleVersion>
           <Payload>PAYLOAD (Transaction)</Payload>
       </cor:COREEnvelopeRealTimeRequest>
    </soap:Body>
</soap:Envelope>
```

The above example defines multiple data elements that would be used when processing the transaction to authenticate the submitter and ensure the attached transaction should be processed.

4.2.3.3 REQUEST AND RESPONSE HANDLING

Real-Time Transactions

A real-time transaction is a single submission or inquiry. It receives a single message as a response. That message is an HTTP Error, an Envelope Error or an invalid v5010 X12 response. The X12 response could be a 999 or TA1 if compliance errors occur, or it could be a 271 or 277 response.

Batch Transactions

A batch transaction includes multiple requests or inquiries. Batch submissions will receive a single response. That response could be an HTTP Error, an Envelope Error or a submission successful response.

Batch Response Transactions

A batch response transaction is a request to retrieve a response file from a previously submitted batch transaction.

4.2.4 HTTP ERROR CODES

The following are common (non-exhaustive) HTTP error codes.

Code	Description
200 OK	Success
202 Accepted	Batch File Submission Has Been Accepted (but not necessarily processed)
400 Bad Request	HTTP Header format errors
403 Forbidden	Sender ID and/or Password was invalid, not on file, or not a match
500 Internal Server Error	Processing error during parsing of the HTTP message or SOAP error encountered during processing of SOAP envelope
501 Not Implemented	Requested Service is not available
502 Bad Gateway	There was an invalid response received from different components of the Healthcare System that are needed to fulfill the transaction
504 Service Unavailable	System is down for maintenance
504 Gateway Timeout	There was a communication error between different components of the Healthcare System that are needed to fulfill the transaction
505 HTTP Version Not Supported	Version of the HTTP used in the incoming message is not supported by the transaction server

5 CONTACT INFORMATION

5.1 DXC EDI TECHNICAL ASSISTANCE

PHONE: 1-800-457-4584, option 3, and then option 2
FAX: (317) 488-5185
EMAIL: INXIXTradingPartner@dxc.com

5.2 APPLICABLE WEBSITES/E-MAIL

Indiana Medicaid for Providers Website: https://www.in.gov/medicaid/providers/index.html

The Trading Partner web page can be found under the Business Transactions, Electronic Data Interchange (EDI) Solutions section of the Indiana Medicaid for Providers Website: https://www.in.gov/medicaid/providers/697.htm

All other contact information is listed under the Contact Information section of the Indiana Medicaid for Providers Website: https://www.in.gov/medicaid/providers/975.htm

6 APPENDICES

6.1 IMPLEMENTATION CHECKLIST

See Trading Partner Information in the Electronic Data Interchange (EDI) Solutions section on the Indiana Medicaid for Providers Website https://www.in.gov/medicaid/providers/697.htm

6.2 CHANGE SUMMAR

This section describes the differences between the current Companion Guide and previous guide(s).

*Core*MMIS Change Summary

Version	CO	CO Name	Revision Date	Revision Status	Revision Page Numbers / Change / Update Details	Completed by
2.0			March 2013	New	CAQH CORE	Systems
2.1		2260	Dec 2013	Implemented	CAQH CORE Phase III	Systems
3.0			Sept 2016	Implemented	Implementation of IN *Core*MMIS System Updates	Systems
3.1			Jan 2017	Implemented	Pg. 6 – Updates to Web Services URLs	Systems
3.2			April 2017	Implemented	Updated throughout document Hewlett Packard Enterprise (HPE) to DXC Technology	Systems
3.3			June 2018	Implemented	CAQH CORE Companion Guide Re-Format Updates	Systems
3.4	58053	File Exchange Domain Name Change	August 2018	Implemented	Updated all links for File Exchange. Pg. 5 – updated list of secure FTP clients. Pg. 12 – updated links to the Indiana Medicaid for Providers website	Systems