

Clinical Data Exchange (CDex) API User Guide



Prepared By:
Acentra Health
1600 Tysons Boulevard, Suite 1000
McLean, VA 22102



evoBrixx X v2026-08
September 01, 2026

Revision History

Revision Level	Date	Description	Change Summary
Version 1.0	04/30/2026	Clinical Data Exchange (CDex) API Product Guide	Initial Delivery
Version 2.0	06/23/2026	Changes in Processing of CDex Unsolicited attachments.	The sections 2.1.1 Clinical Data Exchange (CDex) API and 2.1.2 Processing of CDex Unsolicited attachments has been updated to reflect changes related to • Inclusion of Presigned url mechanism. • Handling of maximum attachment payload size and attachment validation • Attachment delivery method configuration Included IG reference of “ESMD FHIR Guide” in Section 2.1.4 CDex API Documentation.
Version 3.0	07/08/2026	Inclusion of Presigned URL Mechanism, configurations and Attachment Validation flow.	1. As part of URL based attachment submission, Presigned URL Mechanism has been included and Configuration, Attachment Validation flow has been updated in section 2.1.2 Processing of CDex Unsolicited attachments. 2. Replaced Screenshot of CDex API in section 2.2.
Version 4.0	08/14/2026	Changes in Processing of CDex unsolicited attachments in Request to support encoded pre-signed URLs for attachments.	CDex \$submit-attachment operation has been enhanced to support encoded pre-signed URLs for attachments and updated the configuration and attachment validation flow in 2.1.1 Clinical Data Exchange (CDex) API and 2.1.2 Processing of CDex Unsolicited attachments.

Table of Contents

Purpose and Scope	1
Document Conventions	1
Audience Requirements	1
Chapter 1 – Overview	2
1.1 Data Flow	3
Chapter 2 – Clinical Data Exchange (CDex) API.....	4
2.1 Overview	4
2.1.1 Clinical Data Exchange (CDex) API	4
2.1.2 Pre-Signed URL Mechanism.....	5
2.1.3 Processing of CDex API Request	7
2.1.4 Process attachment Response	9
2.1.5 CDex API Documentation	9
2.2 FHIR Server Sandbox Environment.....	10
2.3 Accessing the Sandbox Environment	11
2.4 Base URLs	11
2.5 Capability Statement	11
2.6 Supported Scopes.....	12
2.7 Resources	12
2.8 Request Headers	12
Chapter 3 –Clinical Data Exchange (CDex) API.....	13
Chapter 4 – Third-Party App Registration	14
4.1 Acentra Sandbox Environment	14
4.2 Environments	14
Chapter 5 – Glossary	15
Chapter 6 – Acronyms	16

Figures

Figure 1. Acentra Interoperability Solution- CDex API.....	2
Figure 2 Pre-Signed URL Flow	5
Figure 3: Acentra Sandbox Environment – Online Developer Documentation.....	10
Figure 4: Third-Party App Registration Process	14

Tables

Table 1. Clinical Data Exchange (CDex) API Solution Components.....	2
Table 2. Attachment Submission Methods	4
Table 3. Levels of FHIR Validation.....	7
Table 4. Configured Delivery Methods	8
Table 5. Online Documentation - CDex API	9
Table 6. Base URLs for Base URLs for CDex API	11
Table 7. Resources and URLs for and URLs for CDex API.....	12
Table 8. Request Headers	12
Table 9. Glossary.....	15
Table 10. Acronyms.....	16

Preface

Purpose and Scope

The Da Vinci Burden Reduction Implementation Guides (IGs) are a set of HL7 FHIR-based specifications developed under the HL7 Da Vinci Project. These guides support an integrated workflow that enables EHR and Payer Systems to automate the submission of required documentation and Prior Authorization information.

The CDex API supports unsolicited clinical data exchange, enabling EHR/EMR Systems to submit attachments to a Payer without an explicit request. The Payer then associates the submitted attachment with the corresponding Prior Authorization.

The CDex Implementation Guide, based on the HL7 FHIR Release 4 (R4) standard, provides detailed guidance for implementers on using FHIR-based interactions to support specific clinical data exchanges between Providers and Payers. These interactions help reduce administrative burden and improve operational efficiency.

Document Conventions

This document uses the following conventions:

- The term “you” refers to the application owners who use this document.
- Hyperlink names appear in bold fonts.
- A **Note**, presented in the following style:



Provides important information the user needs to observe or act upon.

Audience Requirements

This document is intended for third-party application owners and developers who need to connect their applications to Acentra’s Clinical Data Exchange (CDex) API, as well as Payers who implement the CDex API to provide data to third-party applications.

Chapter 1 – Overview

The Acentra Interoperability Solution supports the business and technical requirements defined in the CMS mandate (CMS Final Rule 0057 F). The solution uses the Da Vinci Clinical Data Exchange (CDex) 2.1.0 – STU 2.1 Implementation Guide, which is based on HL7 FHIR Release 4 (R4). The solution also incorporates data exchange standards, including HL7® Fast Healthcare Interoperability Resources (FHIR®), attachment document standards defined by the LOINC Document Ontology, X12 attachment codes, and CMS-recommended Da Vinci Implementation Guides (IGs).

The figure below provides the overall Acentra Interoperability Solution.

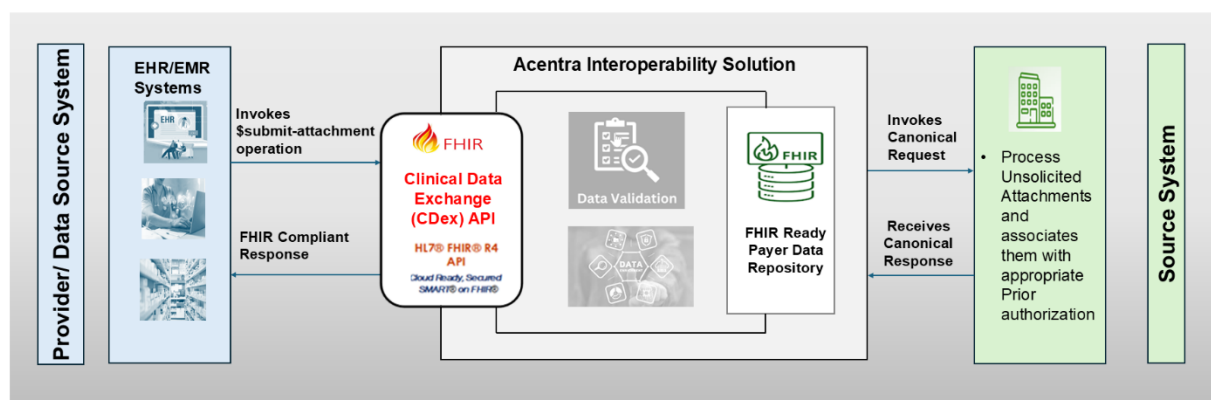


Figure 1. Acentra Interoperability Solution- CDex API

The table below describes each component of the Acentra Interoperability Solution.

Table 1. Clinical Data Exchange (CDex) API Solution Components

Component	Features
FHIR Server	The solution implements a RESTful FHIR API service that supports applicable CDex profiles and related frameworks based on the Da Vinci Clinical Data Exchange Implementation Guide. The FHIR Server performs five levels of validation on the attachment request payload in accordance with the Implementation Guide (IG). Upon successful validation, the FHIR Server stores the validated data in the FHIR-Ready Payer Data Repository (FPDR), which facilitates the transformation of canonical requests and responses to and from the FHIR Server. The FHIR Server then constructs a FHIR-compliant response to enable standardized data exchange and seamless interoperability across integrated Systems.
Data Repository	The System stores the response data obtained from the PAS response bundle during the \$submit operation. The System uses the combination of the PA Tracking Number and Member ID in the ETL tables, along with the corresponding Payer ID, to identify and invoke the appropriate source System endpoint.

Component	Features
FHIR Adapter	The integration layer facilitates data retrieval from the FPDR and enables communication between the FHIR Server and the FPDR.
SMART on FHIR Module	The solution secures the Clinical Data Exchange (CDex) API by implementing the CMS-recommended SMART® on FHIR authorization sequence for standalone launch, based on the OAuth 2.0 framework. EHR/EMR Systems connecting to the FHIR Server must authenticate through a manual process over secure TLS channels. The CDex API integrates with SMART® on FHIR and uses OAuth 2.0 authorization to obtain an access token. The access token enables EHR/EMR Systems to invoke the \$submit-attachment operation to submit additional documentation.

1.1 Data Flow

The EHR/EMR Systems submit unsolicited clinical attachments to support a Prior Authorization based on Payer-defined rules by invoking the CDex \$submit-attachment operation without receiving an explicit request from the Payer. The EHR/EMR Systems securely transmit the attachment payloads to Acentra's Interoperability Solution, which performs five levels of validation in accordance with the CDex API Implementation Guide (IG).

Based on the validated request data, the FHIR Server generates a canonical request and routes it to the appropriate source-specific web service according to the routing configuration. The FHIR Server then receives the canonical response from the Source System, prepares a FHIR-compliant response, and sends the response to the EHR/EMR Systems.

Chapter 2 – Clinical Data Exchange (CDex) API

2.1 Overview

The Clinical Data Exchange (CDex) API enables the EHR/EMR Systems to submit clinical attachments required to support Prior authorization, even when the Payer does not explicitly request the attachments. The FHIR Server facilitates the secure exchange of processed attachment data between Payer Systems (Producers) and authorized EHR/EMR Systems (Consumers) through authorized third-party consumer applications.

Acentra's Interoperability Solution encrypts data both in transit and at rest. To enable authorized access, EHR/EMR Systems connecting to the FHIR Server must register with the Payer Systems through a manual process over TLS. The CDex API integrates with SMART® on FHIR and uses OAuth 2.0 authorization to obtain an access token. Registered and authorized EHR/EMR Systems can then invoke the \$submit-attachment operation to submit unsolicited clinical attachments.

2.1.1 Clinical Data Exchange (CDex) API

The Provider/Data Source System, typically an EHR/EMR System, enables authorized Providers to submit attachments or additional supporting documentation to the Payer System as part of the Prior Authorization process, even when the Payer has not explicitly requested the documentation.

The Provider can submit these unsolicited attachments after submitting the Prior Authorization request. The EHR/EMR System uses the \$submit-attachment operation to submit the attachments.

Supported Attachment Submission Methods

The EHR/EMR System can submit attachments using either of the following methods:

1. **Inline Submission** – The EHR/EMR System includes documents directly in the request as Base64-encoded content.
2. **External Document Reference** – The EHR/EMR System shares documents through a pre-signed URL, enabling the Payer to securely access and retrieve the attachments from an external location.

The EHR/EMR System selects the appropriate submission method based primarily on the attachment size and System configuration.

Table 2. Attachment Submission Methods

Attachment/Request Size	Submission Method	Requirement
Attachment ≤ 10 MB	Inline Base64 or Pre-Signed URL.	Either method may be used, subject to the API request payload limit.

Attachment/Request Size	Submission Method	Requirement
Attachment > 10 MB	Pre-Signed URL	Mandatory



API Gateway Limit: The API supports a maximum total request payload of 10 MB for inline Base64 submissions. If an inline submission causes the total request payload to exceed 10 MB, the API rejects the entire request. In such cases, the EHR System must use the pre-signed URL mechanism to submit the attachments.

2.1.2 Pre-Signed URL Mechanism

The Pre-Signed URL Mechanism enables document externalization by allowing EHR/EMR Systems to upload attachments directly to Amazon S3 storage managed by the FHIR platform using secure, time-limited AWS S3 pre-signed URLs.

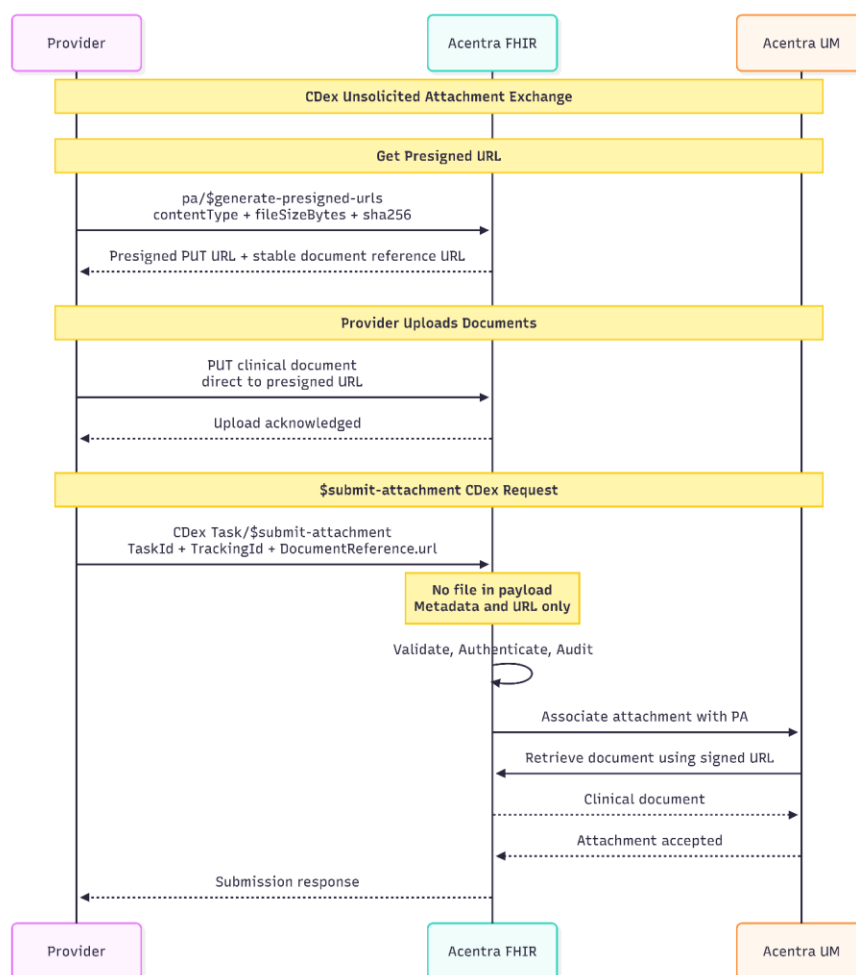


Figure 2 Pre-Signed URL Flow

Instead of embedding the document content within the FHIR request, the EHR/EMR System uploads the document to the designated Amazon S3 location. The FHIR platform then provides a stable, platform-issued FHIR Binary handle, which the EHR/EMR System references in the DocumentReference URL to identify and access the uploaded document.

The functional flow is as follows:

2.1.2.1 Authenticate with the FHIR Platform

The EHR/EMR System authenticates through the existing SMART on FHIR (SoF) authentication process and obtains a bearer token.

2.1.2.2 Request Pre-Signed Upload URLs

For documents that require externalized upload, the EHR/EMR System requests a pre-signed URL by invoking the Binary/\$generate-presigned-urls operation and providing the required document information, including the content type, file size, and document reference identifier.

The FHIR platform validates the request and returns a time-limited, pre-signed Amazon S3 upload URL and a stable FHIR Binary handle for each document.

2.1.2.3 Upload Documents to S3

The EHR/EMR System uploads the document directly to the designated Amazon S3 storage managed by the FHIR platform using the pre-signed URL. The System securely transfers the document over TLS without routing it through the FHIR API Gateway. After the upload is complete, the FHIR platform initiates an asynchronous virus scan of the document.



The FHIR platform performs virus scanning asynchronously, outside the FHIR request and transformation flow, to prevent attachment scanning from affecting FHIR processing performance.

2.1.2.4 Submit the FHIR Bundle

The EHR/EMR System submits the FHIR Bundle using the standard submission operation.

- For documents uploaded through the Pre-Signed URL Mechanism, DocumentReference.attachment.url contains the stable FHIR Binary handle rather than the temporary pre-signed URL.
- For small documents submitted inline based on the configuration, DocumentReference.attachment.data contains either Base64-encoded document content or the Base64-encode the pre-signed URL.



When populating the pre-signed URL as Base64-encoded value, the EHR system must set DocumentReference.attachment.contentType to text/uri-list.

2.1.2.5 Validate FHIR Submission

The FHIR platform validates the submitted Bundle for FHIR structure, applicable profile requirements, and attachment-specific rules. For externally uploaded documents, the FHIR platform also verifies that the Binary reference points to an approved location managed by the FHIR platform.

2.1.2.6 Validate and Process the Uploaded Document

The FHIR platform resolves the FHIR Binary handle to the corresponding document in Amazon S3 and verifies that the document has successfully passed the virus scan. The FHIR platform also validates the actual document type and size against the information provided when the pre-signed URL was generated.

After successful validation, the FHIR platform retrieves the document and prepares it in the format required by the downstream Payer System.

2.1.3 Processing of CDex API Request

When the EHR/EMR System invokes the \$submit-attachment operation, the FHIR System validates the request and associated attachments before sending them to the configured Source System.

2.1.3.1 Request Validation

The FHIR System performs the following levels of validation:

Table 3. Levels of FHIR Validation

Level	Validation	Description
Level 1	Syntax Integrity	Verifies that the request is valid JSON and can be processed as a FHIR request.
Level 2	FHIR Structure	Verifies that the request conforms to FHIR R4 and applicable profile requirements, including required elements, cardinality, and data types.
Level 3	Relational Validation	Verifies relationships and dependencies between elements and referenced resources.
Level 4	Code Set Validation	Verifies that coded elements use the allowed values defined by the applicable Implementation Guide.
Level 5	Attachment Validation	Verifies attachment file type, file size, number of files, and applicable attachment configuration.

The request proceeds only when all required validations are successful.

2.1.3.2 Attachment Processing

After receiving the attachment, the FHIR System processes it according to the delivery method configured for the Source System.

Table 4. Configured Delivery Methods

Attachment Received From EHR	Configured Delivery Method	Processing
Inline Base64	Base64	The FHIR System validates the document and delivers it to the Payer System as a Base64-encoded attachment.
Inline Base64	URL	The FHIR System securely stores the document and delivers it to the Payer System using a pre-signed download URL.
URL	Base64	The FHIR System retrieves the document and delivers it to the Payer System as a Base64-encoded attachment.
URL	URL	The FHIR System retrieves the document and makes it available to the Payer System through a pre-signed download URL.

2.1.3.3 Pre-Signed URL Attachments

For documents uploaded using the FHIR platform's Pre-Signed URL Mechanism, the EHR/EMR System references the uploaded document in the FHIR request using the document reference issued by the FHIR platform.

The FHIR System verifies the reference and confirms that the corresponding document is available in platform-managed storage before processing the document.



The FHIR platform scans all uploaded documents for viruses. Only documents that successfully pass the virus scan are eligible for downstream processing.

2.1.3.4 Validation or Routing Failure

If the request fails validation, contains an invalid attachment, or does not match a configured Source System:

- The FHIR System stops processing the request.
- The FHIR System does not send the attachment request to the Source System.
- The FHIR System returns an **OperationOutcome** containing applicable errors and diagnostic information.



When the payload exceeds the allowed size, the Server returns HTTP 413 Content Too Large client error response status code.

2.1.4 Process attachment Response

After the Source System processes the attachments, it returns a Canonical Response to the FHIR System containing the processing results for the submitted documents. The Canonical Response includes details of the attachments that the Source System accepted and successfully processed. The OperationOutcome resource from the Payer System provides information about invalid or rejected attachments. The FHIR Server then transforms the Canonical Response into a FHIR-compliant response that indicates the processing status of the unsolicited attachments and returns the response to the EHR/EMR Systems.

The FHIR System maintains audit logs for traceability. The audit logs capture details of validated and invalid requests, routing configuration failures, attachment retrieval outcomes, successfully processed attachments, and attachment processing errors returned by the Source System.

2.1.5 CDex API Documentation

The table below provides links to the CDex API online documentation.

Table 5. Online Documentation - CDex API

Function	Online Documentation
IG References	Da Vinci Clinical Data Exchange (CDex) - STU 2.1 - Ballot IG Home - Da Vinci Clinical Data Exchange (CDex) v2.1.0 ESMD FHIR Implementation Guide 1.0.0 - esmd Generate Presigned URL API - ESMD FHIR Implementation Guide v1.0.0
IG Reference for Submit Attachment Operation and Input parameter profiles	\$submit-attachment operation Submit Attachment Operation - Da Vinci Clinical Data Exchange (CDex) v2.1.0 CDex Parameters Submit Attachment Profile CDex Parameters Submit Attachment Profile Document Reference https://hl7.org/fhir/R4/documentreference.html
Error Codes	 FHIR-Error%20Codes _CDex%20API.xlsx

2.2 FHIR Server Sandbox Environment

Acentra's Interoperability Solution Sandbox Environment provides a synthetic and reliable dataset for the applicable CDex API profiles to facilitate API testing. The Sandbox dataset represents realistic data and maintains consistency across the dataset.

For detailed information about Acentra's Interoperability Solution Sandbox Environment, visit <https://sandbox.mhbapp.com/index.html>. The site also provides information about contacting Acentra to register an application.

As shown in Figure 3, the Acentra Sandbox FHIR Server documentation portal provides third-party application developers with the information required to register their applications, connect to the Sandbox FHIR Server, and access the CDex API.

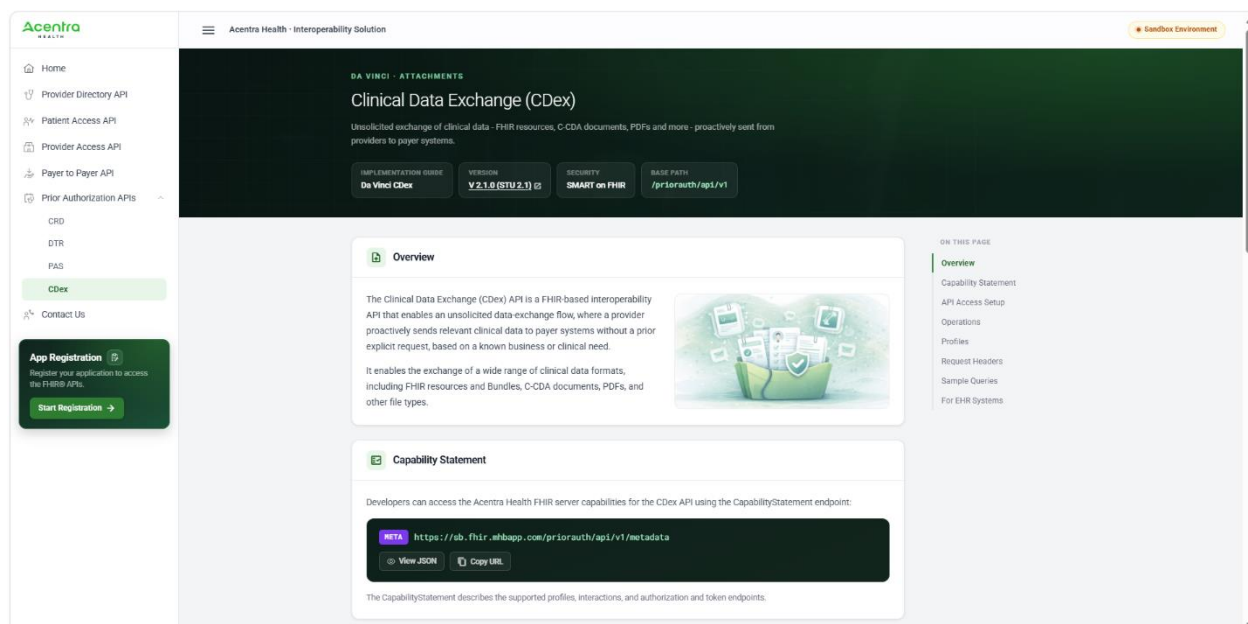


Figure 3: Acentra Sandbox Environment – Online Developer Documentation

2.3 Accessing the Sandbox Environment

Before accessing the CDex API, you must register your application with Acentra's Sandbox Environment.



Refer to the Sandbox FHIR server documentation information at <https://sandbox.mhbapp.com/index.html>

2.4 Base URLs

The following three types of FHIR Server environments are available for the CDex API:

- Acentra Sandbox
- UAT
- Production

The table below provides the FHIR Server base URLs for the CDex API. The base URLs are not accessible through a web browser and must be accessed programmatically.

Table 6. Base URLs for Base URLs for CDex API

Environment	Base URL CDex API
Acentra Sandbox	https://sb.fhir.mhbapp.com/priorauth/api/v1/
UAT	<a href="https://<<tenant-id>>.np.fhir.mhbapp.com/priorauth/api/v1/">https://<<tenant-id>>.np.fhir.mhbapp.com/priorauth/api/v1/
Production	<a href="https://<<tenant-id>>.fhir.mhbapp.com/priorauth/api/v1/">https://<<tenant-id>>.fhir.mhbapp.com/priorauth/api/v1/

2.5 Capability Statement

The Capability Statement describes the capabilities and behaviors supported by the FHIR Server. For example, it identifies the supported profiles and interactions and provides information about the authorization and token endpoints.



The FHIR Server capabilities for the CDex API can be accessed through the Capability Statement endpoint: https://<<base_url_priorauth>>/metadata.

Refer to Table 6 for the CDex API base URLs. For example, the Capability Statement URL for the Sandbox Environment is: <https://sb.fhir.mhbapp.com/priorauth/api/v1/metadata>.

2.6 Supported Scopes

The FHIR Authorization Server supports the following scopes to enable secure backend system-to-system access to the CDex API:

- system/DocumentReference.create
- system/OperationOutcome.read

2.7 Resources

The Acentra Interoperability Solution's Clinical Data Exchange (CDex) API supports the resources listed in the following table:

Table 7. Resources and URLs for and URLs for CDex API

Resources	URL
Submit-attachment	https://{base_url}/\$submit-attachment
Generate Presigned URL	https://{base_url}/\$generate-presigned-urls
Regenerate Presigned URL	https://{base_url}/\$regenerate-presigned-urls

2.8 Request Headers

All requests for CDex API resources must include the headers listed in the following table:

Table 8. Request Headers

Headers	Values
content-type	application/json
accept	application/json
authorization	Bearer<<auth_code>>

Chapter 3 –Clinical Data Exchange (CDex) API

- EHR/EMR Systems must register and receive approval through the App Registration Portal and FHIR Admin Portal.
- The SMART Backend Services Authorization Framework secures system-to-system integration.
- All communications must use TLS to ensure encrypted data transmission.
- The System authenticates EHR/EMR Systems using OAuth 2.0 and JWT-based authentication mechanisms.
- Only authorized applications can access the FHIR APIs based on the defined scopes.
- For detailed onboarding, configuration, and authentication instructions, refer to the following user guides:
 - App Registration User Guide
 - Admin Portal User Guide
- After successful registration and authentication, EHR/EMR Systems can invoke the \$submit-attachment operation to submit additional documentation.

Chapter 4 – Third-Party App Registration

4.1 Acentra Sandbox Environment

The Acentra Sandbox Environment enables third-party application owners to perform connectivity and application testing. Third-party application owners can contact Acentra directly or submit a registration request through the Payer organization. If the Payer organization approves the request, Acentra provides access to the Sandbox Environment. If a third-party application owner contacts Acentra directly, Acentra reviews the request and determines whether to approve or deny access.

4.2 Environments

In the Acentra Interoperability Solution, the intake process collects registration and attestation information from the third-party application owner and sends it to be authorized personnel within the Payer organization (or Tenant). The Payer organization reviews the information provided by Acentra to verify the third-party application.

If the Payer organization approves the registration request, it notifies Acentra. Acentra then issues the required credentials (Client ID) to the third-party application owner for the appropriate Production environment. If the Payer organization does not approve the application registration, it notifies the application owner directly and informs Acentra of the decision. Figure 4 illustrates this process.

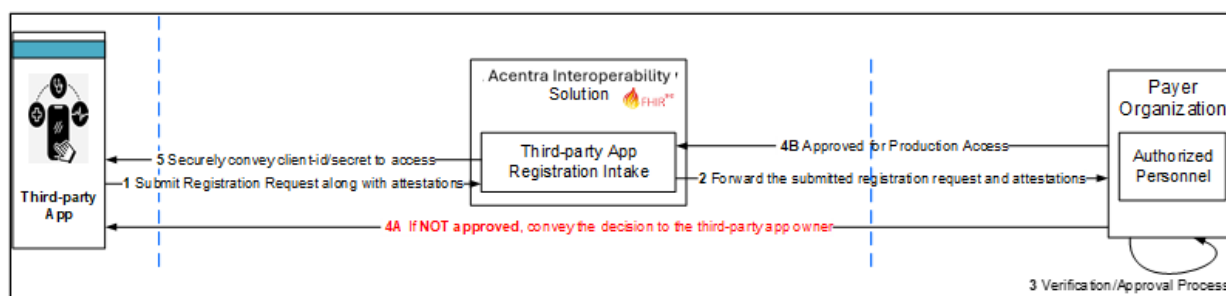


Figure 4: Third-Party App Registration Process

Chapter 5 – Glossary

This section provides definitions for document-specific terms in the following format: Entries are in alphabetical order, and items beginning with numbers will come first.

Table 9. Glossary

Term	Definition
Application Programming Interface	An interface that defines interactions between multiple software applications.
Acentra Interoperability Solution	A solution designed to support the business and technical requirements defined in the CMS mandate, CMS Final Rule 0057-F.
Payer Organization	A business entity that adjudicates and processes payments for healthcare claims, including pharmacy claims. Examples include Medicare, state Medicaid programs, and commercial insurance carriers. In this document, “Tenant” also refers to the Payer Organization associated with the environment.
Sandbox	An environment that includes hardware, instrumentation, simulators, software tools, and other supporting resources. Also known as a “test bed.”
Tenant	See “Payer Organization”.
Third-Party App	An application or program that is neither owned by Acentra nor the Washington State Health Care Authority and that has authorized API access to deliver data to authorized users (EHR/EMR Systems).

Chapter 6 – Acronyms

This section provides definitions for document-specific terms in the following format: Entries are listed in alphabetical order, and items beginning with numbers will come first.

Table 10. Acronyms

Acronym	Definition
API	Application Programming Interface
CMS	Centers for Medicare & Medicaid Services
CDex	Clinical data Exchange
EDI	Electronic Data Interchange
EHR	Electronic Health record
FHIR	Fast Healthcare Interoperability Resources
FPDR	FHIR Ready Payer Data Repository
ID	Identification
IG	Implementation Guide
JSON	JavaScript Object Notation
HIPAA	Health Insurance Portability and Accountability Act
HTTP	Hypertext Transfer Protocol
LOINC	Logical Observation Identifiers Names and Codes
TLS	Transport Layer Security
OIDC	Open ID Connect
OAuth	Open Authorization
PAS	Prior Authorization Support
REST	Representational State Transfer
SMART	Substitutable Medical Applications and Reusable Technologies
STU	Standard for Trial Use
URL	Uniform Resource Locator