Imaging Solutions

Synergy Enterprise Content Management™

Release 2025.1

# Synergy REST API

08/05/2025

Printed in the United States of America.

# Synergy REST APIs

Synergy is an Enterprise Content Management solution used to store and retrieve electronic items such as documents, checks, and reports. Synergy is a Service Gateway Provider for these contracts.

jXchange provides a web-based, services-oriented architecture on a .NET platform that offers open connectivity between Jack Henry Banking core and complementary solutions and third- party products. This contemporary integration utility increases a financial institution's operating flexibility and enables them to use their existing and future technology investments.

> **Note:** Synergy provides two REST APIs with similar features but very different architecture frameworks and support:

- The Enterprise REST API uses the Jack Henry jXchange architecture and should be used by all Jack Henry products and vendor integrations.
- The Internal REST API is designed for on-premise integrations with closed networks where jXchange is not available.

In addition to these two APIs, the Authentication REST API is provided for manual authentication to Synergy. This REST API authenticates a specific user, returning a JWT that can be used for subsequent REST API calls. This API is utilized for the Internal REST API only.

## About this Guide

This guide is designed as supplemental documentation for the Synergy REST APIs. Additional documentation is provided within the Swagger interface for the given REST API, as well as the Synergy documentation guides.

This guide focuses on general concepts not covered in the Swagger documentation, as well as additional detail for certain REST API function call parameters. In some cases, parameters may be explained in a prior section or are relatively self-explanatory by name; in these cases, there may be no additional detail for the parameter listed here.

## Functionality

Synergy is a service provider for the following functionality:

- Enumeration of Storage CTI (used for filing documents), including defined Synergy Institutions, Cabinets, Types, and Indexes
- Enumeration of Retrieval CTI (used for performing searches), including defined Synergy Institutions, Applications/Cabinets, Report/Types, and Indexes
- Running Open Searches, Predefined Searches, and Finding Reports
- Archiving, refiling, and cross-indexing documents
- Retrieving items
- Listing, adding, editing, and deleting text notes

## General Information and Best Practices

This document is intended for application engineers needing to integrate with Synergy.

Direct all integration installation requests to the Jack Henry & Associates IDG - Product Adoption team.

Integration testing must be deemed successful by a member of Jack Henry & Associates before beta testing or rollout.

## Synergy Overview

### Organizations

An Organization is the highest classification level within Synergy. Organizations are normally tie to a given banking or credit union (JH) institution. Synergy users are always defined within the context of an Organization.

Organizations also contain File Rooms. These File Rooms are used to keep data separated for security or usage purposes. File Rooms consist of one or more database connections to the allowed/licensed modules for a system. This section contains specific request information for you to enter in the request message. The data entered in the request determines what data you receive in the response message.

## Synergy Institutions

If further classification is needed within the Organization, use Synergy Institutions. These institutions are used in cases where multiple banks are consolidated in a merger and must be kept separate for a given amount of time.

Institutions are assigned to users, and users can have multiple institutions assigned.

### Synergy Institutions vs JH Institutions

Synergy Institutions are not the same entity as a JH institution (routing number), which is used within the Enterprise REST API and configured in Synergy to ensure requests are routed to the correct organization. JH Institutions are mapped to a Synergy Organization. Synergy Institutions are used within Synergy to classify data further.

## Modules

The following modules exist in Synergy:

### Document Module

The Document Module is used to store multiple different types of documents. These documents can be image files, PDFs, Word files, Excel® files, etc. Documents can also be a compound collection of different file types. Documents can be indexed, searched, and viewed.

### Check Module

The Check Module is used to store check images. Like documents, checks can be indexed, searched, and viewed. However, the process for acquiring checks into Synergy is different. Synergy has check-specific processing options available, whereas some document-specific functions are not available within the Check Module.

### Report Module

The Report Module to store reports in Synergy. Reports are often in text format, although some may be in PDF, or other “legacy” APA formats that allow for an image-based rendering of the text. In all cases, the backing data for the report is text-based.

Reports can be indexed, searched, and viewed. However, reports can be stored without being indexed, in which case a text search can be performed.

# Data Classification

Data is classified as follows:

## Document/Check

### Synergy Institution

Institutions are configured at the Organization level and can be used to restrict access to data. Documents and checks are assigned to an Institution when archived into Synergy. Users are also assigned to Institutions. To access a document or check, the user must also be assigned to the same institution assigned to the document. (Additional security restrictions may apply as well, see below.)

### View Access

When assigning a user to a Synergy Institution, they also receive certain View Access Authorities relatedto that institution. For Documents/Checks, these authorities are a numeric value (1,2,3,4,5,6,7,8,9). These values work like a series of locked doors granting access to a single room. If a user has a key to one or more locks, they can access the room.

#### Example:

A document is assigned to Institution 00, with View Access authorities 1, 4, and 9.

The user is assigned to Institution 00, with View Access authorities 1 and 2.

The user can see the document because they have View Access Authority 1.

### Cabinet and Type

Synergy stores documents using a classification system consisting of Cabinet and Type. A Cabinet is like a physical filing cabinet where documents are stored. A Type is a specific type of document stored in the cabinet.

### Indexes

In addition to Cabinets and Types, documents/check can be assigned Indexes. Indexes are used to classify a document/check. Indexes consist of name/value pairs.

### Name

A document also has a name. Usually, the name is the Type name.

#### Example of Cabinet, Type, Indexes, and Name:

A document is archived with the following:

- Cabinet Name: LOANS – AUTO
- Type Name: APPLICATION
- Document Name: APPLICATION
- Index Name: ACCOUNT NUMBER
- Index Value: 123456789

## Report

Reports have a similar classification system.

### Synergy Institution

Synergy uses institutions to grant access to specific reports. A single institution is associated with a given report instance.

### View Access

For Reports, institutions use a more specific classification system. Reports can be assigned a General, Special, or Privileged view access authority. A given report has either a General or Special designation. The Privileged flag can be assigned to a given report as well, allowing access only to users with the designated view access.

### Application/Report

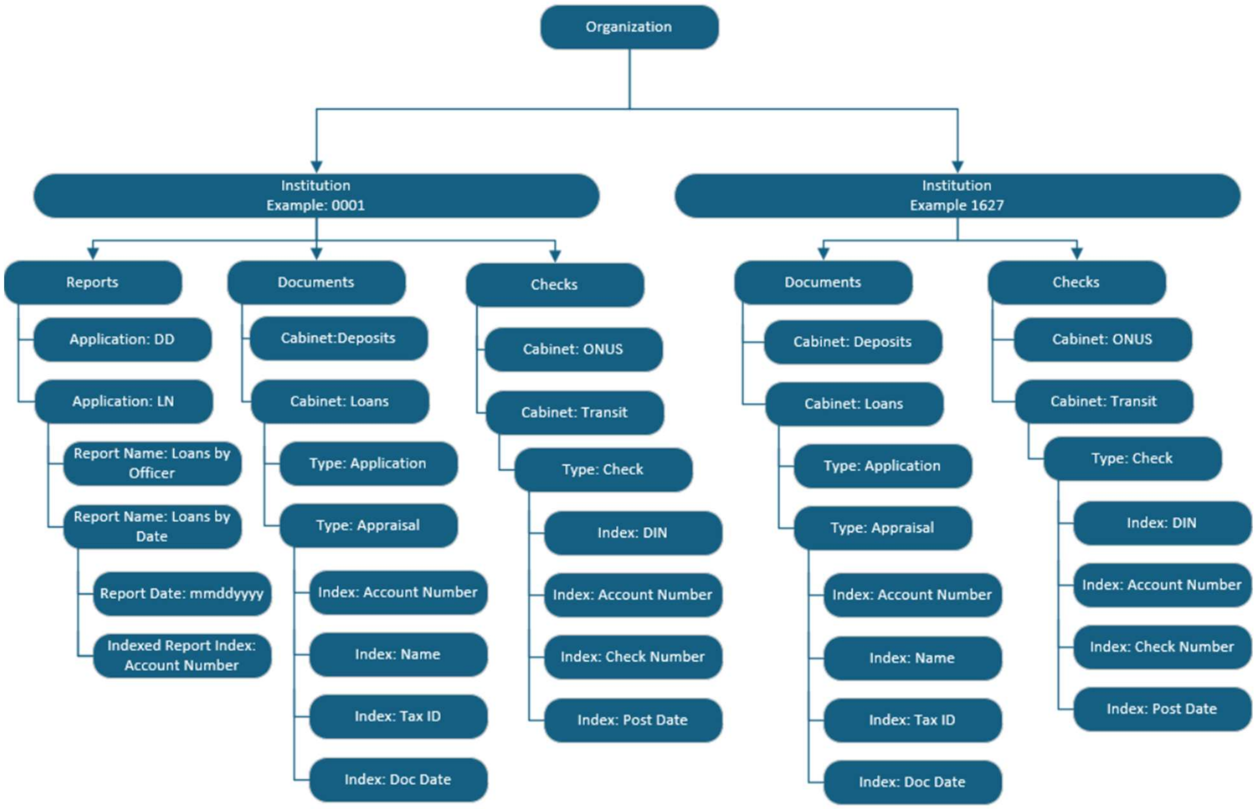
You can define reports further with an Application name and a specific Report name. The Report name may be dynamic or fixed, depending on how the report was loaded.

### Date

A Report also has a Report Date associated with it. This date is normally the date the report was loaded into the system.

### Hierarchy Overview

Visual representation of the Synergy Structure:




# Authentication

To access the Synergy REST APIs, it is necessary to authenticate with Synergy. This authentication is accomplished using different methods. It also varies depending on which REST API is being used.

## jXchange Authentication Framework

Jack Henry jXchange uses OAuth 2.0 and OpenID Connect Solutions for Authentication and information on their framework can be found at https://jackhenry.dev/open-api-docs/authentication-framework.

Users can be authenticated to the Internal REST API via an API key. This API key is defined in Synergy System Administration. API keys expire after one year and are associated with a specific user account. The account cannot be an administrator.

## Synergy Authentication REST API

A user account, password, Organization, and File room can be passed to the Synergy Authentication REST API. This returns as JWT token that can be used in the Internal

REST API for authentication. This token can be used with the Internal REST API only. The token is valid for up to one hour (a smaller time can be specified when initiating the request).

### User Sessions and Licenses

When using REST API functions, a session is obtained in Synergy. This session also consumes a Synergy license (or multiple module licenses, depending on the licensing schema). The license and session are used during the REST API call and are released once the call returns.

### User Authorities

The user account using the REST API must have access to the Synergy data in question, as well as any required authorities for performing given operations.

#### Examples:

To archive a document to Synergy Institution 01, the user must have Synergy Institution 01 assigned.

To cross-index a document, the user must have Cross Index authority.

For more information on user access and assignments, refer to the Synergy Org Administration guide.

## Classification Information for Document Storage

The following details apply to the Cabinet/Type/Index (CTI) information that must be provided for any storage call. This information includes Archive Transactions (single document), Batch Transactions (multiple documents), Refile (document modification), Cross Index (adding additional metadata sets to an existing document).

Provide the following information for archive transactions, batch transactions, and cross-index. For refile, it is valid to modify only a single property listed as follows.

### Cabinet Name

### Type Name

### At least one Index Name/Value Pair

For more information on these parameters, refer to the Data Classification – Document/Check section. For refile, it is not necessary to provide at least one index name/value pair if the index information is not changing.

## Input Parameters:

### Synergy Institution

This parameter can be passed or determined by Synergy and left blank. To determine the institution in Synergy:

1. For the Enterprise REST API, the Synergy Institution must be assigned in jXchange Integration Maintenance.
2. If Filing Preferences are defined in Synergy for the specified Cabinet/Type combination, it is used. Otherwise –
3. If there is only ONE institution assigned in jXchange Integration Maintenance and that institution is also assigned to the user, it is used. Lastly –
4. If there is only ONE institution assigned to the user, it is used.
5. If an institution still cannot be determined, an error is returned. It is for this reason that if the Institution Value is known, pass it with the call. This error typically happens when clients have more than one institution in Synergy and helps prevent future issues when new institutions are added or the setup is modified.

For more on Filing Preferences or jXchange Integration Maintenance settings, refer to the Synergy Org Administration guide and the Synergy Desktop Administration guide, respectively.

### Document Name

Can be passed in or specified via Filing Preferences. If Filing Preferences are not specified, the Type Name is used as the Document Name.

### View Authorities

Can be passed in or specified via Filing Preferences. If Filing Preferences are not configured in Synergy, all authorities (1-9) are assigned.

### Folder Name

Optionally, specify a folder name to archive the document into a folder defined in Synergy. Folders can be used for additional classification but are not required.

### Open Vocabulary Name

A "vocabulary" can be defined in Synergy in case the integrating system has different names for values. Open Vocabulary Mapping can be used to map from the source system to Synergy. For more information on Open Vocabulary Mapping, refer to the Synergy Org Administration guide.

### Use For Smart Indexing

One or more values can be passed to look up additional values that are used for Archive.

### Example:

A user passes in an Account Number, and the system is able to look up additional information such as Name, and CIF Number, etc.

Configure Smart Index in Synergy. Smart Index requires a database connection or integration to the Core for lookup calls. For more information, refer to the Synergy Desktop Administration guide for R.2024.3 and prior, and the Synergy Administration Web Client guide for post R.2024.3 releases.

## Paging for Lists Returned from the REST APIs

Many of the functions that return lists in the REST API support paging. For calls where the entire list is returned on the first call, the NextOffset parameter is not returned.

## The Paging Object

For lists of information, a paging object is returned, even for functions that do not require paging (where the entire list is returned in a single call) to indicate the total number of results returned.

### Next Offset

If more than one page of data is available, this value is provided, indicating that additional hits are available starting with this offset.

#### Results

The number of results returned.

#### Total

The total number of results available.

### Using Paging in Supported Functions

If the Next Offset field is greater than zero, subsequent hits can be returned from the server using the following optional query parameters:

#### Offset

This parameter is the starting record for the request.

#### Count

The number of items to retrieve for the request.

## Return Codes

Return codes are standardized across REST API calls.

### 200

Indicates success and that the call/transaction is complete.

### 201

This return code normally indicates that the call was successful, and a resource was created as part of this call. Depending on the call, this resource may be used in subsequent calls (such as the batch transaction Id when creating a batch transaction).

### 401

This code is returned if the security headers are missing or not valid. This usually occurs if authentication information has not been provided or is expired/not valid.

### 403

The user is not authorized to perform the given action in Synergy.

### 404

The requested item could not be found.

### 500

An error has occurred in the Synergy Server. Additional details can be found in Synergy Central Logging. In general, these errors should occur minimally. If this error is received without additional descriptive information, contact Synergy Support.

## Requesting API Version Information

Sometimes, it is necessary to determine what API versions are available to know if certain functionality is supported. There are two API calls for this – one for Storage calls (such as Archive Transactions, Storage CTI, Batch Transactions) and one for Retrieval calls (Retrieval CTI, document/check/report retrieval, search).

### storage/apiversions (get)

### retrieval/apiversions (get)

Retrieves the versions available for the respective API.

#### Output Parameters

##### Version Tags

List of supported versions for the requested API.

##### Paging

Contains paging information. See the Paging for Lists Returned from the REST APIs section for paging information.

## Archive Transactions

The Archive Transactions REST API calls can be used for direct archival of a single document to Synergy. To archive multiple documents, cross -index, or for documents that require intervention in Capture, use the Batch Transactions API calls.

## archivetransactions (post)

To archive a document, it is necessary to start with the Begin call to initiate the transaction.

> **Note:** The recommended maximum size is no more than 10 MB per chunk.

### Input Parameters (required):

#### CTI Metadata

See the Classification Information for Document Storage section for more information.

#### Files

The File Name, File Type, and File Size are passed in the begin call and the file is pre-registered.

### Output Parameters

#### Document Transaction Id

If the call is successful, a transaction Id is returned that is used during the archive transaction.

#### File Response

If file information was specified in the request, this section includes the assigned File Ids for each file, and the associated file name that was passed in with the file information.

## Adding Files (ad hoc method)

## archivetransactions/{documentTransactionId}/files (put)

> **Note:** This method was superseded by the following call, which uses file pre-registration.

## Adding Files (File Id method)

## archivetransactions/{documentTransactionId}/files/{fileId} (put)

If the file information was specified when initiating the archive transaction, files can be added via File Id. The Document Transaction Id is required.

This method can be used multiple times for the same Document Transaction Id to add multiple files to a given document.

### Input Parameters (required)

#### Document Transaction Id

The Id for this transaction.

#### File Id

The previously assigned File Id for the specific file.

#### Input File

The input file is specified as part of this transaction.

> **Note:** The recommended maximum size is no more than 10 MB per chunk.

### Output Parameters

None

## Setting the State

## archivetransactions/{documentTransactionId}/state (post)

Once the transaction is ready to go or if it must be canceled, the State transaction can be used.

### Input Parameters (required)

#### State

Either Commit or Rollback. Commit initiates the archive. Rollback cancels the transaction and removes all files.

### Output Parameters

#### Unique Item Id

If successfully archived, the Unique Item Id for the new document is provided.

# Batch Transactions

The Batch Transactions REST API calls can be used for more complex archival scenarios. These scenarios may include needing to archive multiple documents at once, cross indexing of documents during archive, or the need for manual intervention via the Synergy Capture Client (such as for manual indexing or OCR).

## Starting a Batch

## batchtransactions (post)

To start a batch, it is necessary to begin a batch transaction.

### Optional Input Parameters:

#### Batch Name

This parameter can be passed or can be left blank and is set based on the user and machine name. Use Batch Names to help discern batches created with the API call and to help make working with and reviewing batches in the capture queue easier. Common practice is to make batch names variables such as using the Account Number or CIF Number or Name to help find the Batch in the queue.

#### Batch Template Name

If not specified, the default batch template is used. To review the default batch template configuration, it is necessary to log in to Synergy Capture and select Configure->Batch Template Setup and review the default template configuration. Avoid using the Default Template and instead create a new Batch Template for each unique API integration. Using a new template helps with identifying batches from their sources as well as restricting who can interact with those batches in Synergy Capture.

#### Batch Routing Step

This parameter is the desired batch routing step to start the batch in. If not specified, the batch is set to the Acquire state. See the following Batch Templates and Routing Steps section for additional information on each of the Batch Routing Steps and when to use each of the steps.

#### Output Parameters:

#### Batch Transaction Id

This parameter is used throughout the batch transaction and uniquely identifies this transaction. Even though a transaction ID has been assigned, the batch does not yet exist nor is it viewable within Capture. Commit the batch to see it in Capture.

## Batch Templates and Routing Steps

Batch Templates are configured in Capture and have a series of routing steps that are followed as the batch progresses. For detailed information about routing steps, refer to the Synergy Capture guide.

In general, within Capture, batches go through an Acquire state, in which files are added either directly from disk or via scanning. Batches can optionally be moved to the Clean state, in which images are cleaned according to the batch template configuration. They can then be moved to either a Manual Index or OCR state, depending on how indexing data is provided. An optional Review step can be provided with additional verification tasks to ensure that the batch is correctly indexed. Last, the batch is set to File to initiate the archival process. If this process fails, the batch is returned in a Correct state so that corrections can be made.

From the REST API, there are certain functions that are not valid. For example, there is no scan option, so files must be added manually. Additionally, a batch template configured for OCR is not set initially to the File step (see the File step in the following step list), since this step skips the OCR step in Capture.

In general, batches are set upon the begin call as follows:

### Acquire

Set the batch to this step if:

- Additional files are going to be added to the batch from outside of the REST API, via import or scan.

- If the batch template has an OCR routing step. The OCR routing step is not set by default and only occurs when the batch template is configured for the OCR routing step.

To set a batch in Acquire status, it is not necessary to provide metadata. However, if metadata is provided:

- If one or more indexes are sent, a Type must be sent.
- If a type is sent, a Cabinet must be sent.
- All provided indexes must exist for the specified Type.
- The provided Type must exist for the specified Cabinet.

### Clean

Set to this state if:

- No additional files will be added to the batch.
- The batch template contains a Clean step. A Clean step is not set by default and only occurs when the batch template is configured for a Clean step.

### Manual Index

Set to this state if:

- No additional files will be added to the batch.
- The batch template does not contain a Clean step.
- Metadata is provided outside of the REST API (in the Capture Client).

### OCR

Set to this state if:

- No additional files will be added to the batch.
- The batch template does not contain a Clean step.
- The batch template contains an OCR step. The OCR routing step is not set by default and only occurs when the batch tempate is configured for the OCR routing step.
- Metadata is provided outside of the REST API (in the Capture Client), if OCR does not provide all required metadata.

### Review

Set to this state if:

- No additional files will be added to the batch.
- The batch template does not contain a Clean step.
- All indexing metadata is provided within the subsequent REST API calls.
- The batch template contains a Review step.The Review step is not set by default and only occurs when the batch template is configured for the Review step.

### File

Set to this state if:

- No additional files will be added to the batch.
- The batch template does not contain a Clean step.
- All indexing metadata is within the subsequent REST API calls.
- The batch template does not contain a Review step.The Review step is not set by default and only occurs when the batch template is configured for the Review step.
- No additional intervention is required within Capture. In this case, the goal is to archive the supplied documents straight from the REST API.

Documents set to the Review or File batch state have stricter metadata requirements than the previous steps. In addition to meeting the previous requirements, the user submitting the batch must have authority to all items via storage profiles, Synergy institution assignments, view authority assignments, etc.

## Adding Documents

## batchtransactions/{batchTransactionId}/documents (post)

One or more documents can be added to a batch once a Transaction Id is established. Likewise, multiple files can be added to a document. To start a document, it is necessary to begin a document transaction.

### Input Parameters:

#### Batch Transaction Id

This parameter is the batch transaction Id assigned previously.

#### CTI Metadata

See the [Classification Information for Document Storage] section.

#### File Information

As with the Archive Transactions calls, files can be pre-registered ahead of time or can be added "ad hoc."

### Output Parameters:

#### Document Transaction Id

This parameter is used throughout the document transaction and uniquely identifies this transaction.

#### File Response

If files were pre-registered, this parameter includes the File Id and File Name for each registered file.

## Adding Files (ad hoc method)

## batchtransactions/{batchTransactionId}/documents/{documentTransactionId}/files (put)

If the file information was not specified up front, files can be added via the Files method. The Batch Transaction Id and Document Transaction Id are required. A 1-based file order can be specified per file. If set to 0 (default), files are added in the order received.

This method can be used multiple times for the same Document Transaction Id to add multiple files to a given document.

### Input Parameters (required)

#### Batch Transaction Id

The Id for this batch transaction.

#### Document Transaction Id

The Id for this document transaction.

#### Input File

The input file is specified as part of this transaction.

### Input Parameters (optional)

#### Attachment Audience

If an attachment is not specified, leave this parameter blank.

#### Filename

If this parameter is provided, it is used to determine the file type.

### Output Parameters

None

## Adding Files (File Id method)

## batchtransactions/{batchTransactionId}/documents/{documentTransactionId}/files/fileId (put)

If the file information was specified up front, files can be added via File Id. The Document Transaction Id is required.

This method can be used multiple times for the same Document Transaction Id to add multiple files to a given document.

### Input Parameters (required)

#### Batch Transaction Id

The Id for this batch transaction.

#### Document Transaction Id

The Id for this document transaction.

#### File Id

The previously assigned File Id for the specific file.

#### Input File

The input file is specified as part of this transaction.

### Output Parameters

None

## Cross Indexing

## batchtransactions/{batchTransactionId}/documents/{documentTransactionId}/crossindexdocuments (post)

As part of the batch process, any document can be Cross Indexed. Cross-indexing is the process of adding additional, unique metadata sets for a given document.

> **Example:** A Loan has two co-signers. For the APPLICATION, which is filed into Synergy, there are two unique sets of metadata. One has an applicant named JUSTIN CASE, and the other has an applicant named ANNIE CASE.

### Input Parameters (required)

#### Batch Transaction Id

The Id for this batch transaction.

#### Document Transaction Id

The Id for this document transaction.

### Request Body (Cabinet Name, Type Name, Synergy Institution, etc…)

The input metadata that describes the document to archive.

For more information on these input parameters and how they are used, refer to the Begin section in the Archive Transactions section as the metadata structure is the same.

### Output Parameter

#### Document Transaction Id

Confirmation of the Document Transaction Id for this document transaction.

# Setting the State

# batchtransactions/{batchTransactionId}/state (post)

Use the State transaction once the batch is ready or if it must be canceled.

## Input Parameters (required)

### Batch Transaction Id

The Id for this batch transaction.

### State

Either Commit or Rollback. Commit initiates the archive. Rollback cancels the transaction and removes all files.

## Output Parameters

None if the batch is rolled back. Otherwise, the Batch ID for the batch is returned, and the batch is now viewable in Capture.

# Documents

Various document operations can be performed, such as retrieving document properties, adding a metadata set (cross-index) or changing the existing metadata (refile).

# Document Properties

# documents/{uniqueItemId} (get)

To retrieve the properties for a document, pass the unique item Id. This call is related to storage-specific metadata specifically. To retrieve more detailed properties for an item stored in the system, use the Items API calls (documented later).

## Input Parameters (required)

### Unique Item Id

The unique item Id for this item.

## Output Parameters

The document, check, or report properties are returned, including:

- Cabinet Name
- Type Name
- Synergy Institution Name
- Document Name
- View Authorities
- Index name/value pairs
- Folder name (for future use. Currently, This parameter returns empty, even if the item is in a folder.)

## Refile a Document

## documents/{uniqueItemId} (patch)

To modify the properties for a document or check, pass the unique item Id.

### Input Parameters (required)

#### Unique Item Id

The unique item Id for this item.

#### CTI Metadata

See the Classification Information for Document Storage section.

For Refile, it is not necessary to provide all parameters. However, changing certain parameters can affect others. For example, changing the cabinet can render the selected type invalid. Likewise, changing the type can impact the indexes.

### Output Parameters

The unique item Id for the refiled item, as confirmation.

## Deleting a Document

## documents/{uniqueItemId} (delete)

To delete a document or check, pass the unique item Id.

### Input Parameters (required)

#### Unique Item Id

The unique item Id for this item.

### Output Parameters

The unique item Id for the deleted item, as confirmation.

## Cross-Indexing a Document

## documents/{uniqueItemId}/crossindexdocuments (post)

To cross-index a document or check, pass the unique item Id and the item metadata.

### Input Parameters (required)

#### Unique Item Id

The unique item Id for the item to be cross indexed.

#### CTI Metadata

See the Classification Information for Document Storage section.

### Output Parameters

The unique item Id for the cross-indexed item.

## Document Attachments

## documents/{uniqueItemId}/attachments (get)

Used to retrieve an attachment that was archived with a document. This API call is primarily used for very specific third-party integrations as these attachments cannot be viewed in Synergy.

### Input Parameters (required)

#### Unique Item Id

The unique item Id for which the attachment exists.

### Input Parameters (optional)

#### Audience

Narrows down the attachment type. Only attachments archived with the specified audience are returned.

### Output Parameters

The unique item Id for the item as confirmation, as well as the attachment details, including the unique identifier, the audience, the data type, data size, and description.

## Document Attachment Data

## documents/{uniqueItemId}/attachments/{attachmentId}/data (get)

Once the unique attachment Id has been retrieved, the attachment data can be retrieved.

### Input Parameters (required)

#### Attachment Id

The Id of the attachment.

### Output Parameters

The attachment data in string format.

## Retrieving Document Status

## documents/{uniqueItemId}/status (get)

A Document Status can be assigned to a document. Retrieve the current status via a Get call.

A Document Status can be set on a document for reference and can also be tied to a Jack Henry Event when the status changes. For more information on Document Status, refer to the Synergy Org Administration guide.

### Input Parameters (required)

#### Unique Item Id

The unique item Id for the item.

### Output Parameters

#### Document Status Name

The current status value.

#### Version Management Information

If Version Management is enabled, this parameter contains the version number, comment, current version management state, date of the last version check-in, and the username for the last user to perform an action on the document.

## Updating Document Status

## documents/{uniqueItemId}/status (post)

The Document Status can be updated.

### Input Parameters (required)

#### Unique Item Id

The unique item Id for the item.

#### Document Status

The new status value.

### Output Parameters

#### Unique Item Id

Confirmation of the unique item Id.

#### Previous Document Status

The prior value (can be empty).

#### Current Document Status

New value.

### Deleting a Document Status

### documents/{uniqueItemId}/status (delete)

The Document Status can be deleted (cleared).

#### Input Parameters (required)

##### Unique Item Id

The unique item Id for the item.

#### Output Parameters

##### Unique Item Id

Confirmation of the unique item Id.

##### Previous Document Status

The prior value (can be empty).

##### Current Document Status

New value (cleared).

## Document Status Definitions

Use Document Status Definitions to get a list of available document status definitions that can be used in the system.

### Get Document Status Definitions

### /documentstatusdefinitions (get)

Retrieves a list of all currently defined document status definitions.

#### Output Parameters

##### Document Status Names

An array of document status names.

##### Paging

Contains paging information. See the [Paging for Lists Returned from the REST APIs] section for more information.

# Storage CTI

The REST APIs offer the ability to enumerate CTI data within the system. There are two different types of formats: Storage CTI and Retrieval CTI. The Storage CTI API calls are used for retrieving data pertinent to Archival only, and returns only document information, whereas the Retrieval CTI (covered later) supports all module types and more complex query scenarios.

## Synergy Institutions

## synergyinstitutions (get)

Retrieves a list of institutions that can be used for archival.

### Output Parameters

#### Cabinet Names

A list of cabinets that the authenticated user can access.

#### Paging

Contains paging information. See the Paging for Lists Returned from the REST APIs section for more information.

## Cabinets

## cabinets (get)

Retrieves a list of cabinets that can be used for archival.

### Input Parameters (optional)

#### Synergy Institution

A single institution can be passed to narrow down the type list. Otherwise, the type list is for all institutions.

### Output Parameters

#### Cabinet Names

A list of types that the authenticated user can access.

### Paging

Contains paging information. See the [Paging for Lists Returned from the REST APIs] section for more information.

## Types

# {cabinetName}/types (get)

Retrieves a list of types that can be used for archival.

### Input Parameters (required)

#### Cabinet

Use a single cabinet name to pass to narrow the type list.

### Input Parameters (optional)

#### Synergy Institution

A single institution can be passed to narrow down the type list. Otherwise, the type list is for all institutions.

### Output Parameters

#### Type Names

A list of types that the authenticated user can access.

#### Paging

Contains paging information. See the Paging for Lists Returned from the REST APIs section for more information.

## Indexes

# {cabinetName}/types/{typeName}/indexes (get)

Retrieves a list of types that can be used for archival.

### Input Parameters (required)

#### Cabinet

A single cabinet name must be passed to narrow down the index list.

### Type

A single type name must be passed to narrow down the index list.

## Input Parameters (optional)

### Synergy Institution

A single institution can be passed to narrow down the index list. Otherwise, the index list will be for all institutions.

## Output Parameters

### Indexes

An array of indexes that the authenticated user can access.

Each entry includes:

**Index Name**
Name of the index.

**Index Type**
Numeric, Date, or String.

**Is Multi Value**
True or false – indicates whether the index supports multiple values for a single index.

**Length**
Length of the value.

**Is Required**
If this parameter is true, the value must be provided when archiving.

**Is Smart Index Lookup**
Indicates that this field can be used for Smart Indexing. The value can be used to "lookup" additional values for the record.

**Has Smart Index Mapping**
If true, this parameter is a value that can get populated as part of Smart Indexing.

#### Paging

Contains paging information. See the [Paging for Lists Returned from the REST APIs]() section for more information.

# Retrieval CTI

Retrieval CTI offers the ability to enumerate metadata from all modules and offers additional options for retrieving metadata.

## Institutions Search

## institutions-search (post)

Retrieves a list of institutions in the system.

### Input Parameters (Optional)

#### Modules

This parameter is standard in all Retrieval CTI API calls. For institutions, it does not impact the results and does not need to be specified.

### Output Parameters

#### Synergy Institution Names

A list of cabinets that the authenticated user can access.

#### Paging

Contains paging information. See the [Paging for Lists Returned from the REST APIs]() section for more information.

## Cabinets Search

## cabinets-search (post)

Retrieves a list of applications and cabinets in the system.

### Input Parameters (required)

#### Modules

Pass in Document, Check, or Report in any combination to retrieve the applications/cabinets for the authenticated user.
Input Parameters (optional)

#### Institution Names

Pass in one or more institution names to refine the results.

### Output Parameters

#### Cabinet Names

A list of applications and cabinets that the authenticated user can access.

#### Paging

Contains paging information. See the Paging for Lists Returned from the REST APIs section for more information.

## Types Search

## types-search (post)

Retrieves a list of reports and types that are in the system.

### Input Parameters (required)

#### Modules

Pass in Document, Check, or Report in any combination to retrieve the reports/types for the authenticated user.

### Input Parameters (optional)

#### Synergy Institution Names

Pass in one or more institution names to refine the results.

#### Cabinet Names

Pass in one or more application/cabinet names to refine the results.

### Output Parameters

#### Type Names

A list of applications and cabinets that the authenticated user can access.

#### Paging

Contains paging information (see above section on paging information).

## Indexes Search

## indexes-search (post)

Retrieves a list of indexes that are in the system.

### Input Parameters (required)

#### Modules

Pass in Document, Check, or Report in any combination to retrieve the indexes for the authenticated user.

### Input Parameters (optional)

#### Synergy Institution Names

Pass in one or more institution names to refine the results.

#### Cabinet Names

Pass in one or more application/cabinet names to refine the results.

#### Type Names

Pass in one or more report/type names to refine the results.

### Output Parameters

#### Indexes

An array of indexes that the authenticated user can access.

Each entry includes:

**Index Name**

Name of the index.

**Index Type**
Numeric, Date, or String.

**Is Multi Value**
True or false – indicates whether the index supports multiple values for a single index.

**Length**
Length of the value.

#### Paging

Contains paging information. See the Paging for Lists Returned from the REST APIs section for more information.

# Item Retrieval

There are multiple methods for retrieving items and item properties via the REST APIs.

## Retrieve Item Properties

## items/{uniqueItemId} (get)

Retrieves the properties for a document, check, or report.

### Input Parameters (Required)

#### Unique Item Id

The unique item Id for the item.

### Output Parameters

#### Cabinet Name

Application or Cabinet, depending on whether it is a document, check, or report.

#### Type Name

Type or Report, depending on whether it is a document, check, or report.

#### Synergy Institution Name

#### Unique Item Id (for confirmation)

#### Module

#### Indexes (Name/Value pairs only, no attributes)

#### Document Date

This parameter is the date that the item was created.

#### View Authority

1-9 for documents or checks, G, S, A (all, meaning both general and special) and/or P for reports. For more info on View authorities, refer to the Synergy Org Administration guide.

#### Has Crop Zone

Boolean flag indicating whether the document has a Crop Zone. Crop Zones are used to return zoomed-in images. For example, on a signature card, only the signature portion could be returned if a crop zone is defined as such. Only valid for documents.

#### Restrictions

Indicates that the item has restrictions applied that can limit who can view the item. Users attempting to view the item have the correct authorization. For more on Restrictions, consult the Synergy Org Administration guide.

#### Document Info

Document/check module-specific information.

**Legacy Document Id**
This parameter is the Doc Id field in Synergy.

**Document Name**
Name of the document.

**Modification Date**
Date the document was last modified.

**Check Return Reason**
If the item is a returned check, this parameter indicates the reason for the return.

### Report Info

Report module-specific information.

**Report Name**
Name of the report.

**Has Statements**
Indicates whether the report was loaded as an entire report or by statement, which can affect how it is viewed or retrieved.

**Statement Page Count**
The unique item Id can refer to a specific statement within a report loaded by statement. In this case, the page count returned is for that current statement. If no page is indicated in the unique item Id, it is for the first statement.

**Date/Group**
The report date. This parameter is either the date the report was loaded, or it can be a custom “Group” name of up to six characters. In most scenarios, it will be the report loaded date.

### Page Count

The number of pages in the document, check, or report.

### Files

**File Id**
The sequential Id for each file. This parameter is only relevant for documents, which can consist of compound files. For non-compound documents and for checks and reports, this parameter is set to 1.

**File Type**
*Subpage Count*
Number of pages in this file.

# Get Item

## items/{uniqueItemId}/pages (get)

Retrieves the entire document, check, or report.

### Input Parameters (Required)

#### Unique Item Id

The unique item Id for the item.

#### Export Format

For this API call, the only supported formats are TXT and PDF. PDF can be used for any file; TXT is only supported for non-APA (text) reports.

#### Render Options

For the PDF export format, the Annotations, Overlays, and Green Bar options may be selected. Green Bar is only supported for non-APA (text) reports.

### Output Parameters

#### Output File

The return for this API call is the requested file.

# Get Pages

## items/{uniqueItemId}/pages/{pageNumber} (get)

Retrieves a page range for a document, check, or report.

### Input Parameters (Required)

#### Unique Item Id

The unique item Id for the item.

#### Page Number

Starting page or page to retrieve.

## Input Parameters (Optional)

### Page Count

If not specified, this parameter defaults to 0, and only the page specified in Page Number is returned.

### Export Format

Format for the output file. If not specified, it returns PDF.

**Note the following restrictions on export format:**

- PDF: Supported for all document types and supports a page range.
- TIFF: Supported for documents only and supports a page range.
- JPEG: Supported for documents only and does not support a page range (single page only).
- PNG: Supported for documents only and does not support a page range (single page only).
- TEXT: Supported for text reports only and supports a page range.

### Render Options

For the PDF export format, the Annotations, Overlays, and Green Bar options may be selected. Green Bar is only supported for non-APA (text) reports.

## Output Parameters

### Output File

The return for this API call is the requested file.

# Retrieve File

# items/{uniqueItemId}/files/{fileNumber} (get)

Retrieves a file from a document or check. This works for any non-report item, if the file number is within range, but is primarily for "compound" documents that are made up of multiple files. An example of this would be documents that contain a mix of PDF and TIFF files saved into one document. The file number is used to retrieve one of the respective files in the document, which is always returned in its native format.

### Input Parameters (Required)

#### Unique Item Id

The unique item Id for the item.

#### File Number

The 1-based sequence number of the file. For items that only contain one file, this will be 1.

### Output Parameters

#### Output File

The return for this API call is the requested file.

## Retrieve Crop Zone

## items/{uniqueItemId}/cropzone (get)

A Crop Zone can be defined for a document or check, which provides a zoomed-in view of a given area. An example of this would be for viewing just the signature portion of a signature card. The crop zone must be pre-defined in Synergy. If iit is not known whether a crop zone exists for a given item, the properties call indicates whether there is a crop zone.

### Input Parameters (Required)

#### Unique Item Id

The unique item Id for the item.

### Input Parameters (Optional)

#### Export Format

Format for the output file. If not specified, it returns PDF.

#### Render Options

For the PDF export format, Annotations and Overlays may be selected.

### Output Parameters

#### Output File

The return for this API call is the requested file.

# Searching for Documents, Checks, and Reports

The REST APIs support the ability to perform a Predefined Search or an open search (for authorized users).

## Searching for Items

## items-search (post)

To begin a search, it is necessary to send search criteria and/or a predefined search name.

### Input Parameters:

Note that in this case, required vs optional are not listed, as this can vary based on the type of search performed.

#### Search Clauses

One or more index name/value pairs OR metadata fields. Metadata fields describe the properties of the item. The metadata fields are specified in the same manner as they would be for a general search within Synergy.

**Supported Metadata Fields:**

- <Created Date>
- <Document ID>
- <Document Name>
- <File Extension>
- <Modified Date>
- <Text Value> (for searching reports only)
- <Unique Item ID>
- <Version Management Comment>
- <Version Management Status>

**Index Name**
Name of the index or metadata field

**Index Value**
Value for the index or metadata field

**Operator**
Type of search to perform for this value.

**Valid operator values:**

- eq (equals)
- neq (not equals)
- lt (less than)
- lte (less than or equal to)
- gt (greater than)
- gte (greater than or equal to)
- in (equal to one of these values, values are separated by a \t separator)
- con (contains)

### Open Vocabulary Name

A "vocabulary" can be defined in Synergy in case the integrating system has different names for values. Open Vocabulary Mapping can be used to map from the source system to Synergy. For more information on Open Vocabulary Mapping, refer to the Synergy Org Administration guide.

### Modules

List of modules to search: Document, Check, Report. If a Predefined Search is specified (see the Predefined Search Name section), This parameter should not be passed.

### Cabinet Names

Optional array of one or more applications/cabinets.

### Type Names

Optional array of one or more reports/types. If a Predefined Search is specified (see the Predefined Search section), This parameter should not be passed.

### Synergy Institution Names

Optional array of one or more institutions. If a Predefined Search is specified (see the Predefined Search section), This parameter should not be passed.

### Sort Index Name

In the results, this parameter is the index/metadata name to sort the data return by.

### Hitlist View Name

Optional hitlist view name. If not specified, the default hitlist view is used.

### Report Dates From/To

For Report Searches, a date range can be specified. If a Predefined Search is specified (see the Predefined Search Name section), this parameter should not be passed.

### Force Async

Searches that return < 100 hits and execute in less than one minute will, by default, return the results synchronously. If either of these thresholds are exceeded, the search is performed asynchronously. Setting this value to true forces the search to be asynchronous, regardless of the time or number of hits.

### Predefined Search Name

Optional parameter specifying which predefined search to perform.

Any Cabinets, if defined, should be a subset of those defined in the Predefined Search (if no applications/cabinets are defined in the Predefined Search, this limitation does not apply. Any Applications/Cabinets may be specified).

The index name/value pairs should, at a minimum, contain any required values to perform the Predefined Search.

## Output Parameters

### Search Id

If the call is successful, a search id is returned that can be used to subsequently retrieve results, if the search was performed asynchronously.

### Is Search Complete

If the entire search was completed and returned synchronously, this value is true.

### Results

If the search was performed synchronously, this will contain the search results, including the following fields per item:

- *Cabinet Name* (Application/Cabinet)
- *Type Name* (Report/Type)
- *Synergy Institution Name*
- *Unique Item Id*
- *Module*
- *Indexes (name/value pairs)*
- *Document Date*
- *View Authority*
- *Has Crop Zone* (true or false)
- *Restrictions* (any restrictions applied on this account by the Core)
- *Paging Information*

Document Specific info, including:

- *Legacy Document Id* (Doc Id in Synergy)
- *Document Name*
- *Modification Date*
- *Check Return Reason* (valid for checks only, code and string value containing a return reason if the check was returned)

Report Info, including:

- *Report Name*

- *Has Statements* (true or false)
- *Statement Page Count* (for the given statement)
- *Date Group*
- *Hit Page*, including page number and report total page count

# Checking Search Status

## items-search/{searchId} (get)

If the search is performed asynchronously, the status can be checked with this call.

### Input Parameters (required)

#### Search Id

The search id returned from the initial search request.

### Output Parameters

#### Search Id

The search id is returned for confirmation.

#### Is Search Complete

True or false

#### Total Result Count

Total number of hits returned.

#### Is Canceled

True or false – if the search was canceled via the REST API call this will be true.

# Canceling a Search

## items-search/{searchId} (delete)

Cancels a running search.

### Input Parameters (required)

#### Search Id

The search id returned from the initial search request.

## Retrieving Search Results

## items-search/{searchId}/results (get)

If the search is performed asynchronously, the results can be retrieved with this call once the search is complete.

### Input Parameters (required)

#### Search Id

The search id returned from the initial search request.

### Output Parameters

#### Search Id, Is Search Complete, Results

The results for this call are identical to that which is found in the synchronous search results. See the (Items-Search (post)) section.

## Finding Reports Without a Search

## reports-search (get)

For documents and checks, in the search API calls above, it is not necessary to provide index information. In this case, all documents and checks are returned for the given parameters. For reports, this is not the case. It is necessary to provide at least one index or text value to search on. To retrieve a list of reports in the system without providing index or text criteria, this API call can be used.

### Input Parameters (optional)

#### Synergy Institution Names

One or more institution names can be provided to narrow the results.

#### Application Names

One or more application names can be provided to narrow the results.

#### Report Names

One or more report names can be provided to narrow the results.

#### Report Dates From/To

An optional date range can be provided.

### Output Parameters

#### Reports

An array of report objects, consisting of:

- *Unique Item Id*
- *Synergy Institution*
- *Application*
- *Report Name*
- *Report Date*
- *Paging Object*

## Retrieve a List of Predefined Searches

## predefinedsearches (get)

To perform a predefined search, it may be necessary to get a list of those available in the system.

### Output Parameters

#### Predefined Searches

An array of predefined searches, consisting of:

**Predefined Search Id**
Used in a subsequent call to look up the predefined search details.

**Predefined Search Name**
Name of the Predefined Search.

**File Room**

File Room for which the Predefined Search has been configured.

## Retrieve Predefined Search

## predefinedsearches/{predefinedSearchId} (get)

This call retrieves additional details about a given search.

### Input Parameters (required)

#### Predefined Search Id

### Output Parameters

#### Predefined Search Name

#### Predefined Search Id

#### Search Groups

An array of search groups, consisting of:

**Group Id**
Zero-based group id

**Modules**
Document, Check, Report

**Cabinet Names**
**Cabinet Name Fragment**
A list of succinct cabinet names or a “fragment” (partial) name can be sent in.

**Indexes**
An array of indexes, consisting of:

- *Index Name*
- *Operator*
- *Is Locked* - If true, the value cannot be changed.

Values are not returned for security reasons.

# Text Notes

Text notes can be retrieved, added, edited, and deleted.

## List Text Notes

items/{uniqueItemId}/texnotes (get)
The list of text notes can be retrieved for a Synergy item.

### Input Parameters (required):

#### Unique Item Id

### Output Parameters:

#### Text Notes

An array of text notes, containing:

- *Text Note Id*
- *Note Name*
- *Is Classified* - True/false depending on whether the note is classified.
- *Note Type* - Indicates whether the note is a page, application, or report note for Reports, or a document, type, or cabinet note for Documents/Checks. Note that as of today, only page/document notes are returned.

## Creating Text Notes

## items/{uniqueItemId}/texnotes (post)

Create a text note. Currently only supports page/document notes.

### Input Parameters (required):

#### Note Name

Name of the note

#### Note Text

Text note content

## Input Parameters (optional)

### Is Classified

If the note is to be a classified text note, set to true. If not specified, the note will not be classified.

### Note Key

If the note is to be password protected, enter a value. If nothing is passed, the note will not have a password.

## Output Parameters

### Text Note Id

The new Id for the text note.

# Get Text Note Properties

# items/{uniqueItemId}/texnotes/{textNoteId} (get)

Get properties of a specific text note.

## Input Parameters (required):

### Unique Item Id

The Synergy item.

### Text Note Id

The Id for the text note to delete.

## Input Parameters (optional):

### Note Key

Password, if the note is password protected.

### Output Parameters:

#### Text Note Id

#### Note Name

#### Is Classified

True/false depending on whether the note is classified.

#### Note Type

Indicates whether the note is a page, application, or report note for Reports, or a document, type, or cabinet note for Documents/Checks. Note that as of today, only page/document notes are returned.

#### Owner

Note creator.

#### Modification Date

#### Page Number

Page number where the note is located (applicable for reports only).

#### IsKeyProtected

Requires a password.

#### Note Text

Note content.

## Deleting Text Notes

## items/{uniqueItemId}/texnotes/{textNoteId} (delete)

Delete a text note.

### Input Parameters (required):

#### Unique Item Id

The Synergy item.

### Text Note Id

The Id for the text note to delete.

### Output Parameters:

### Text Note Id

The Id for the text note, for confirmation.

## Editing Text Notes

## items/{uniqueItemId}/texnotes/{textNoteId} (patch)

Edit a text note. Currently only supports page/document notes.

### Input Parameters (required):

### Note Name

Name of the note.

### Note Text

Text note content.

### Input Parameters (optional):

### Is Classified

If the note is to be a classified text note, set to true. If not specified, the note will not be classified.

### Note Key

If the note is to be password protected, enter a value. If nothing is passed, the note will not have a password.

### Output Parameters:

### Text Note Id

The Id for the text note for confirmation.

# Appendix

Key changes between REST API versions:

**Note:** When starting a new integration, use the latest available version of the REST API unless there is a required reason to use a previous version.

| Version | Change Notes |
|---|---|
| **2.1** | Superseded by Version 2.2 |
| **2.2** | Added Batch Capture functionality, Document Status Support |
| **2.3** | Route format was updated – note this is a breaking change from prior versions.<br>Sample route change from 2.2 to 2.3:<br><br>/v1/institutions/{InstitutionId}/environments/{InstitutionEnvironment}/products/{ProductId}/**storage/v2.2/**batchtransactions<br><br>/v1/institutions/{InstitutionId}/environments/{InstitutionEnvironment}/products/{ProductId}/**v2.3/storage/**batchtransactions<br><br>Note the updated version location in **bold**. |