surescripts

Formulary Download

surescripts

CONTENTS

surescripts

1. Formulary Download Overview

About Formulary File Download

Surescripts is introducing a new way for technology vendors to retrieve NCPDP Formulary & Benefit files—using a REST API.

The API supports both the current NCPDP Formulary & Benefit version 3.0 and version 60, and enables the vendor system to:

- retrieve available formulary sources and list types
- search for formulary list files based on source PBM, list type and load and modified dates
- download located files

About This Overview

The Formulary Download Overview provides detailed instructions for retrieving and managing formulary files from Surescripts. It covers download processes, file structure, handling requirements, and best practices to ensure accurate integration and compliance. Use this guide to streamline formulary data access and maintain consistency across your systems.

Document References

> **Note:** This guide is meant to support, integrate, and further clarify the NCPDP standard, implementation guides and best practices as used through the Surescripts network. It does not reproduce the base standard in its entirety. Requirements beyond those in the NCPDP documentation are called out in this document and are also enforceable. Customers should read and comprehend the associated standards prior to reading this document.

Please reference the following documents when reading this guide:

Surescripts-provided materials
Connectivity and Authentication Implementation Guide

External materials to be obtained by the customer (available at www.NCPDP.org)
NCPDP Formulary and Benefit Standard Implementation Guide (v60 Republished May 2024)
NCPDP Formulary and Benefit Implementation Recommendations Document
NCPDP External Code List (Version 202404)
V60 Formulary and Benefit Standard Examples Guide Version 10
NCPDP Formulary and Benefit Standard Version 60 Transition Guidance Version 10

NCPDP SCRIPT

This guide utilizes the NCPDP (National Council for Prescription Drug Programs) messaging standard Prescriber/Pharmacist Interface SCRIPT version 2023011 as a baseline. To become an NCPDP member, please go to **https://www.ncpdp.org/Membership.aspx**.

The NCPDP SCRIPT Standard 2023011 package contains a complete list of fields and additional requirements. Customers should read and comprehend the associated standards prior to reading this guide.

NCPDP is an American National Standards Institute (ANSI) accredited Standard Development Organization. The NCPDP "SCRIPT" standard is a copyrighted document and may be obtained by contacting:

NCPDP
9240 E. Raintree Drive
Scottsdale, AZ 85260-7518
Phone: (480) 477-1000
Fax: (480) 767-1042
https://www.ncpdp.org

2. Integration & Production

The following section defines related technical and support elements needed to achieve certification for moving into production, along with our Integration, Compliance, and Certification processes.

Integration Process

When a customer begins integrating a Surescripts product into their application(s), they will be provided access to all the Surescripts documentation pertaining to the product being implemented. In addition, customers will be provided access to the staging environment to design, develop and test the product integration within their application.

> **Note:** The time frame of the project can vary depending on the customer's resource allocation for the project.

Meeting Requirements

During Integration, customers will ensure their system has met all product requirements. The Certification Testing will focus on message format, application workflow and display in accordance with Surescripts documentation and the associated Application Certification Requirements (ACRs). Upon successful completion of certification and, when applicable, other pre-production network requirements (e.g., Identity Proofing, Attestations, DEA audit), customers will be enabled in production.

For requirements, consider the following:

- Surescripts ACRs are required to be met to achieve production status on the Surescripts network and will be enforced as part of certification.
- To ensure high-quality transactions, Surescripts applies additional business rules above and beyond the NCPDP schema defined requirements that will cause a message to be successful or rejected.
- Surescripts test cases do not cover all possible scenarios in production. Customers are responsible for testing any other applicable scenarios specific to their production environment.
- In accordance with the customer's legal agreement with Surescripts, each customer is responsible for ensuring compliance with all applicable laws including, but not limited to, local and state laws/regulations where doing business.

Transition to Production

Once and the contract are complete and approved by the Surescripts Certification Review Board, the customer is ready to move into production. Surescripts will configure the production connection and validate successful operations with the customer. Prior to the transition to production, Surescripts Account Management will work with the customer and internal Surescripts teams to discuss the following:

- Production support contacts (Escalation Matrix)
- Support process and training
- Support hours

Surescripts Terminology Usage

The following table outlines terminology usage for this guide.

Term	Term Usage
must	Requirements that are enforced as part of the production code or by Surescripts business rules. Some business rules reside in the Element Details section of this guide.
shall	The requirements customers are required to meet in order to be certified on the Surescripts network. These requirements will be enforced as part of certification, not through business rules.
should	Used for guidance and best practices to produce superior results and enhance electronic messaging. Best practices can also be found in the Best Practice sections. Customers are encouraged, but not required, to meet best practices in order to be certified on the Surescripts network.

Communication Rules

Please refer to the Connectivity and Authentication Guide for details regarding communication and security protocol requirements as well as references to all links and IP addresses associated with Surescripts services. For the network to be reliable, there are communication rules to which all customers must adhere.

UTC Time Format

By using Coordinated Universal Time (UTC), the receiver of a message will know the time regardless of their time zone. For example: If a message was sent from Boston at 5:30PM EDT (Eastern Daylight Time), the time would be sent as 21:30 UTC time. If this message was received in Chicago CDT (Central Daylight Time), the 21:30 UTC could be converted to the local CDT time of 4:30PM. Refer to **http://en.wikipedia.org/wiki/Coordinated_Universal_Time**, or **http://www.w3.org/XML/** for more information. UTC time must be synchronized with NIST (National Institute of Standards and Technology) and the difference must be less than one minute. Drift of no more than one minute will be acceptable.

When sending only a Date (not Date and Time), it should be sent in your local time zone, and the receiver should interpret it in their local time. Neither party should attempt to convert the Date to UTC.

All standard programming languages should have a function for generating a date in the UTC time zone or displaying a date in the local time zone. The format of the date/time fields in the XML schema must use the xsd:dateTime format. Examples of that format are: CCYY-MM-DDTHH:MM:SS.FZ, where the UTC time zone may be specified as Z, + 00:00, or - 00:00. For example, 2013-01-01T16:09:04.5Z, or 2013-01-01T16:09:04.5-00:00, or 2013-01-01T16:09:04.5+00:00, where 16:09:04.5Z would be 16 hours, 09 minutes, 04 seconds, 5 fractional seconds. UTC time is denoted by either the “Z” in the first example or the “-00:00” in the second example, or the “+00:00” in the third example. The fractional seconds is not required. Refer to xsd:dateTime for more information.

For simple date fields that do not include the time portion, the format is: CCYY-MM-DD.

Character Set

The character set contains ASCII values 32 – 126, which include:

Character Type	Characters
Symbols	! " # $ % & ' ( ) * + , - . / : ; < = > ? @ [\ ] ^ _ ` { | } ~
Numerals	Ø to 9
Letters, upper and lower case	A to Z, a to z

UTF-8 is the required character encoding for XML. Other encoding formats are not supported by Surescripts. It is recommended that customers declare their UTF-8 formatting in the message header.

XML uses certain characters to describe the structure of the document. When an element value needs to include one of the special characters, the character must be replaced its predefined entity representation when writing the XML document. Most XML processor libraries automatically handle the character replacement. For more information see **https://en.wikipedia.org/wiki/List_of_XML_and_HTML_character_entity_references#Predefined_entities_in_XML or https://www.w3.org/TR/REC-xml/#sec-predefined-ent**

Character	Character Name	Received Value
>	Greater than	>
<	Less than	<
"	Quote	"
&	Ampersand	&
'	Apostrophe	'

Compliance

Surescripts goal is efficiency and consistency across the network so all customers can meet the highest measures of patient safety, end-to-end reliability, and quality. To ensure that customers comply with and adhere to the approved certification requirements, Surescripts:

- Monitors customers in production to ensure all network customers remain in compliance with certification requirements and contractual terms.
- Initiates a remediation process for identified compliance issues.

To ensure network and patient safety, customer agrees to notify Surescripts of any modifications through use of the Surescripts recertification form. Upon receipt of this form, changes will be reviewed, and a determination made as to what (if any) level of certification may be required before being placed into production.

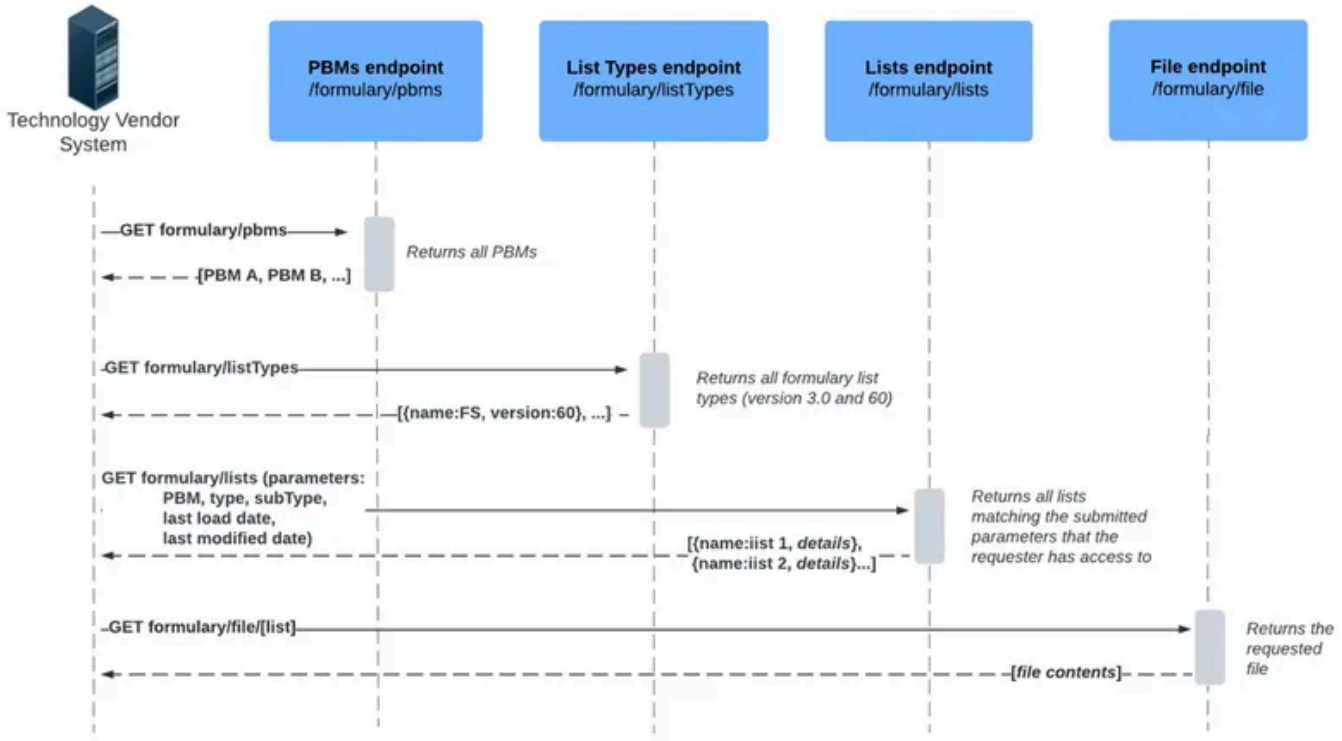
When changes are made to a customer's implementation, the customer should advise their account manager. The customer will be given a recertification guide to provide details regarding the changes made. Upon receipt of the completed document, the details will be reviewed, and a determination will be made as to what (if any) level of recertification is necessary.

As a reminder, Surescripts conducts certification with customers to ensure the application adheres to network requirements. Surescripts will enforce mandatory fields as required by the Standards body and Surescripts guide requirements. To maximize interoperability, customers are recommended to support optional fields that have been created to address gaps in discrete data needs and the many solutions that are in place for the benefit of the receiver. Surescripts encourages, but does not guarantee, the use of optional discrete fields to support end user workflows.

This guide is intended for certification on our network only and is not intended to ensure compliance with state and federal law. In accordance with the customer's legal agreement with Surescripts, each customer is responsible for conducting its own due diligence to ensure compliance with all applicable laws and requirements, including, but not limited to, local and state laws and regulations in which the customer's application is deployed and used.

3. Process Overview

The API provides four REST endpoints that the technology vendor uses to locate and download formulary lists as illustrated below:



4. Security and Data Access

The Formulary Download API restricts content based on the account of the user accessing the system—identified by the MTLS client certificate submitted in the request.

In addition to MTLS certificate validation, all requests to the Formulary Download API must include the HTTP header x-participant-id. This header must contain the Participant ID associated with the requester. Surescripts uses this value in conjunction with the MTLS client certificate to validate that the requester is authorized to access the requested data.

Surescripts validates the submitted certificate and determines the associated Account ID and Participant ID. These are used to determine:

- the NCPDP Formulary & Benefit version(s) available to the requester • the requester's preferred drug database for rolled-up formulary data
- any permission restrictions for accessing formulary data.

5. API Details

This section details the four Download API endpoints.

> **Note on sourceVersions behavior:**
>
> The sourceVersions value indicates the formulary data publication version originally supplied by the PBM.
>
> Possible values include:
>
> - 30 – PBM publishes formulary data using the Version 3.0 format
> - 60 – PBM publishes formulary data using the Version 60 format
>
> Surescripts will always publish the corresponding version of the formulary data, in addition to the following:
>
> - If a PBM publishes formulary data in Version 3.0 format, Surescripts will up-translate and publish the data in Version 60 format.
> - If a PBM publishes formulary data in Version 60 format, Surescripts will down-translate and publish the data in Version 3.0 format.

Get PBMs

/formulary/pbms

The pbms endpoint returns an array of all PBMs that supply formulary data through Surescripts. All PBMs are returned.

Note that the response returns all PBMs, which could contain one or more PBMs the client does not have access to. A subsequent Get Lists request for a PBM the client cannot access will return an empty result.

Request Details

Headers

- x-participant-id (required): Participant ID for the requester.

Parameters

The request contains no parameters.

Response Details

The response consists of three properties for each returned PBM:

- pbmName: string
- pbmId: string containing the PBM's Participant ID
- sourceVersions: array of enumerated versions (30, 60) indicating the versions supported by the PBM (integer)
- lastPublishedDate: string (format: ISO date-time) specifies the last date a file was published by the PBM

The HTTP response codes accompanying a response are:

- 200: The request was processed successfully
- 400: The request contained an error
- 401: Access denied

- 403: The requester does not have access to the endpoint
- 500: A system error occurred

Response Example

Bash


```
[
    {
    "pbmName": "SOUTHWEST_MEDICAID",
    "pbmId": "D00000049124989",
    "sourceVersions": [
      "60"
    ]
  },
  {
    "pbmName": "US_MED_MANAGEMENT",
    "pbmId": "D00000020106965",
    "sourceVersions": [
      "30"
    ]
  },
  {
    "pbmName": “COMMERCIAL_BENEFITS",
    "pbmId": "D00000055125361",
    "sourceVersions": [
      "30",
      "60"
    ]
  },...
```


Get List Types

/formulary/listTypes

The listTypes endpoint returns the full set of list types (in v3.0 and +v60) and sub-types (in v3.0). The set of list types returned, however, are restricted to the formulary version(s) that the client has access to.

Request Details

Headers

- x-participant-id (required): Participant ID for the requester.

Parameters

The request contains no parameters.

Response Details

The response consists of three properties for each returned type:

- name: enumerated string representing the list type (see below)
- subTypes: enumerated string representing the list sub-type (v3.0 only. See below.)
- version: enumerated list version (30, 60) (integer)

Below are the list types and sub-types returned in the response's name and subType properties.

Version 30 (NCPDP F&B version 3.0) name:

- ALT
- COP
  - subType: DS, SL
- COV
  - subType: AL, DE, GL, PA, QL, RD, RS, SM, ST, TM
- FSL

Version +60 (NCPDP F&B version 60) name: FS, DT, AT, AG, CS, PS, PE, AL, GL, PA, QL, SP, ST, SM, PC, PN, GM, ME

The HTTP response codes accompanying a response are:

- 200: The request was processed successfully
- 400: The request contained an error
- 401: Access denied
- 403: The requester does not have access to the endpoint
- 500: A system error occurred

Response Example

Bash


```
[
  {
    "name": "AT",
    "version": 60
  },
  {
    "name": "GL",
    "version": 60
  },
  {
    "name": "COV",
    "version": 30,
    "subTypes": [
      "AL",
      "DE",
      "GL",
      "PA",
      "QL",
      "RD",
      "RS",
      "SM",
      "ST",
      "TM"
    ]
  },
  {
    "name": "ST",
    "version": 60
  },…
```


Get Lists

/formulary/lists?pbmId=<pbm>&type=<type>&subtype=<subtype> &version=<version>&publishedAfter=<publishedAfter> &detailChangeAfter=<detailChangeAfter>&page=<page>&pageSize=<pageSize>

The lists endpoint enables clients to find formulary files for download. It returns an array of objects representing available formulary lists matching the parameters submitted in the request.

This endpoint will be used for two types of processing: Full Formulary Download and Delta File Download.

- Interacting with the API is the same in both modes
- But in the Delta File Download process, the additional publishedAfter parameter is included in the request--to return changes since the last time the formulary file load was processed.

The API returns a lastDetailChangeDate date and a lastPublishedDate value. The lastDetailChangeDate value indicates the last time the list detail records changed. The lastPublishedDate is the last time the file was received from the PBM.

PBM file submissions may or may not include updated detail records from a previous file submission. If a new list file is received, but the detail records have not been updated:

- the lastPublishedDate will be updated to reflect the new date the file was received
- but the lastDetailChangeDate will remain unchanged.

Using the lastDetailChangeDate, list lastPublishedDate, and PBM lastPublishedDate response values, you can reduce the number of file downloads that are needed for processing.

- PBM list searches can be skipped when the PBM lastPublishedDate is older than the last formulary load execution.
- If the lastDetailChangeDate value is newer than the lastDetailChangeDate in your system, then download and process the updated file.
- If the lastDetailChangeDate is the same as the lastDetailChangeDate in your system, but the lastPublishedDate is newer than the lastPublishedDate in your system, then the effective date and expiration date for the list might have changed. You can use the effective date and expiration date returned by the lists API to update your system.
- If the lastDetailChangeDate and lastPublishedDate match the values in your system for the list then the file does not need to be downloaded and processed.

Timing best practices:

- Timing: Full Formulary Downloads should be performed weekly or when drug database changes occur. Delta File Download should be performed nightly.
- For clients using the non-rolled up version of the formulary files, then all files should be downloaded and processed after each drug database update in the client system.

Request Details

Headers

- x-participant-id (required): Participant ID for the requester.

Parameters

The request consists of multiple search parameters that can be used in combination to locate lists available for download.

Only the page parameter is required; all other parameters are optional. All parameters are case-sensitive. They can be submitted in any order.

- pbmId: string
- type: enumerated string, all v3.0 and +v60 list types:
  - "ALT", "CLS", "CRS", "COP", "COV", "FSL", "AG", "AL", "AT", "CS", "FS", "GL", "GM", "ME", "PA", "PC", "PE", "PN", "PS", "QL", "SM", "SP", "ST"
- subtype: enumerated string, only applies to v3.0 list sub-types:
  - "DS", "SL", "AL", "DE", "GL", "PA", "QL", "RD", "RS", "SM", "ST", "TM"
- publishedAfter: string (format: ISO date-time)
  - The last time the file was received from a PBM. Expected format: YYYY-MM-DDTHH:MM:SS.SSSZ (including time)
- detailChangeAfter: string (format: ISO date-time)
  - When the last change to detail records occurred. Expected format: YYYY-MM-DDTHH:MM:SS.SSSZ (including time)
- version: string (30 or 60)
  - If not included, the response defaults to the client's preferred version.
  - If set to a version not available to the client will return an empty result set
- page: required numeric. 1 should be used for the first page of data.
- pageSize: numeric value with default of 10000

Notes:

- A query may include a combination of type and subtype; however, invalid combinations of types and subtypes will return an empty result set
- By default, files of the client's preferred version will be returned based on the Surescripts configuration. The version parameter is available for clients with access to both v30 and +v60

Response Details

The response includes a lists array which consists of nine properties for each returned formulary list, current page, total pages, and page size:

- lists
  - name: string
  - pbmId: string
  - listType: enumerated string, all v3.0 and +v60 list types:
- "ALT", "CLS", "CRS", "COP", "COV", "FSL", "AG", "AL", "AT", "CS", "FS", "GL", "GM", "ME", "PA", "PC", "PE", "PN", "PS", "QL", "SM", "SP", "ST"
  - subType: enumerated string, all v3.0 list sub-types:
- "DS", "SL", "AL", "DE", "GL", "PA", "QL", "RD", "RS", "SM", "ST", "TM" o Only populated for v3.0 coverage and copay list types
  - size: integer
  - lastPublishedDate: string (format: ISO date-time)
  - lastDetailChangeDate: string (format: ISO date-time)
  - effectiveDate: string (format: date without dashes, from original file)
  - expirationDate: string (format: date without dashes, from original file)
- Not always present in the response, as it is an optional list element
  - fileLink: string, contains the url to download the file
- totalPages: Total number of pages based on the current page size.
- pageSize: Page size for the request.
- currentPage: The current page number for the request.

Note: Get Lists requests for PBMs the client doesn't have access to will return an empty result.

The HTTP response codes accompanying a response are:

- 200: The request was processed successfully
- 400: The request contained an error
- 401: Access denied
- 403: The requester does not have access to the endpoint
- 404: No files matching the request were found
- 500: A system error occurred

Response Example (Version 60 list)

Bash


```
{
    "lists": [
        {
            "name": "FILE-FS-1001",
            "pbmId": "T00000000021633",
            "listType": "FS",
            "size": 281,
            "lastPublishedDate": "2024-07-05T21:24:36.708Z",
            "lastDetailChangeDate": "2024-07-05T21:24:36.708Z",
            "effectiveDate": "20230410",
      "expirationDate": "20991231",
  "fileLink":  "https://formulary.surescripts.net/formulary-download/formulary/pbms/T00000000021633/types/FS/lists/FILE-FS-1001/fi
        },     …
] ,
"totalPages": 1,
 "currentPage": 1,
 “pageSize": 10000
}
```


Response Example (Version 30 list including the subtype property)

Bash


```
{
    "lists": [
        {
            "name": "AL73046",
            "pbmId": "T0000000003444",
            "listType": "COV",
            "subType": "AL",
            "size": 277,
            "lastPublishedDate": "2024-07-05T21:24:36.259Z",
            "lastDetailChangeDate": "2024-07-05T21:24:36.259Z",
            "effectiveDate": "20230413",
            "expirationDate": null,
"fileLink":  "https://formulary.surescripts.net/formulary-download/formulary/pbms/T00000000003444/types/COV/subtypes/AL/lists/AL73
        },
     …
] ,
"totalPages": 1,
 "currentPage": 1,
 “pageSize": 10000
}
```


Get File

This endpoint returns the actual formulary file.

Request Details

Headers

- x-participant-id (required): Participant ID for the requester.

Parameters

The request URI is structured as follows for version 3.0 formulary files and version +60 files. All path parameters are case sensitive.

For +v60 & v3.0 FSL & ALT files:

/formulary/pbms/<pbmId>/types/<type>/lists/<list>/file

e.g., /formulary/pbms/T00000000021633/types/FS/lists/FILE-FS-1001/file

For v3.0 COV & COP files:

/formulary/pbms/<pbmId>/types/<type>/subtypes/<subtype>/lists/<list>/file

e.g., /formulary/pbms/T0000000003444/types/COP/subtypes/AL/lists/AL73046/file

Response Details

The endpoint's response is the requested file. Note that, in the event that multiple files match the request (e.g., a current file and one whose effective date is in the future) the endpoint will return the file with the most recent effective date that is either today or in the past.

The HTTP response codes accompanying a response are:

- 200: The request was processed successfully
- 400: The request contained an error
- 401: Access denied
- 403: The requester does not have access to the requested file
- 404: The requested file was not found
- 500: A system error occurred

Response

[streamed text file]

Bash

```
HDR|30|D00000000119037|
|S00000000000001|TestLoad1|2207190021|20220719|08273100|FRM|U|20220714|T
GHD|T3APA00940|PA|D|20220715
GTR|0
TRL|2
```

Error Responses and Handling

Error responses from all formulary download APIs are returned in the same manner, with a JSON structured body accompanying the HTTP response status code.

For example, if a request is submitted to the Get Lists with the invalid version parameter value of 99, the response will be:

HTTP response status code: 400 (Bad Request) - Response body:

Bash

```
{
    "code": "400-01",
    "message": "Invalid version"
}
```

The code value in the response body consists of the HTTP response status code followed by a dash and an additional 2-digit code providing a more specific indication of the issue encountered, when possible.

The table below details the error responses produced by each of the APIs and the appropriate client actions to be taken in each scenario.

API	HTTP status	Response body code/message		Client Action
Get PBMs				
	401	401-01	Missing certificate details	
	403	403-01	Permission denied for the selected version.	
	403	403-02	Could not find participant	
	500	500-01	Internal error.	
Get List Types				
	401	401-01	Missing certificate details	Correct the version parameter and resubmit
	403	403-01	Permission denied for the selected version.	
	403	403-02	Could not find participant	
	500	500-01	Internal error.	
Get Lists				
	400	400-01	Invalid version. Only version 30 or 60 is allowed.	Correct the version parameter and resubmit
	400	400-02	Participant configuration was not found.	Contact Surescripts support to confirm your configuration.
	400	400-03	Invalid PBM Name parameter	Valid PBM values are returned from the PBMs endpoint in the pbmId field.
	400	400-04	Invalid Type parameter	Valid List Type values are returned from the List Type endpoint in the name fields. Confirm you are using the correct list types for the configured formulary version.

API	HTTP status	Response body code/message		Client Action
	400	400-05	Invalid subType parameter.	Valid List Type values are returned from the List Type endpoint in the subTypes fields. Confirm you are using the correct list types for the configured formulary version.
	401	401-01	Missing certificate details	Contact Surescripts support to confirm your configuration.
	403	403-01	Permission denied for the selected version.	Contact Surescripts support to confirm your configuration.
	403	403-02	Could not find participant	Contact Surescripts support to confirm your configuration.
	500	500-01	Internal error.	
Get File				
	400	400-01	Invalid version. Only version 30 or 60 is allowed.	Correct the version parameter and resubmit
	400	400-02	Participant configuration was not found.	Contact Surescripts support to confirm your configuration.
	401	401-01	Missing certificate details	Contact Surescripts support to confirm your configuration.
	403	403-01	Permission denied for the selected version.	Contact Surescripts support to confirm your configuration.
	403	403-02	Could not find participant	Contact Surescripts support to confirm your configuration.
	403	403-03	Permission denied for selected file.	Participant configuration does not allow access to this file.
	404	404-01	File not found	Could not find the selected file. This file may have been removed from the formulary list.
	500	500-01	Internal error.	

6. Disclaimers

Guide Disclaimer

In the event that the Participant chooses to make any software changes based upon recommendations in this guide, the Participant acknowledges and agrees that Surescripts Network shall bear no responsibility or liability for the Participant's changes or any effects thereof. The Participant shall also be required to transition to the new guide at such time as said guide is published, which may involve different or additional parameters than are published in this guide.

Examples Disclaimer

Examples provided throughout this guide are not intended to be all-inclusive. This pertains to example workflows, element-specific (field) examples, or message examples. Participants should not restrict coding to the examples used herein.

When developing, this guide should be used along with additional documentation referenced and applicable customary coding standards.

Copyright

7. Document Change Log

July 24, 2026

Sec. Title	Change Description
N/A	Developer portal - initial release