surescripts

ePA Accelerator

surescripts

CONTENTS

surescripts

1. Electronic Prior Authorization Accelerator Overview

Product Overview

Electronic Prior Authorization (ePA) integrates directly with provider vendor systems, enabling healthcare professionals to easily obtain prior authorizations at the point of care. The prior authorization messages are designed to support the exchange of information obtained from the electronic health record system. The messages allow for the use of coded references, increasing the opportunity for interoperability. It is important for both provider vendors and PBM/payers to implement ePA using the most automated system possible to ensure timely responses to each of the messages.

About this Guide

Surescripts created this document to assist provider vendors in implementing the Electronic Prior Authorization Accelerator, which is based on the NCPDP SCRIPT standard for ePA.

The audience for this document includes any customer responsible for integrating the Accelerator within an existing provider vendor system.

Notes:

- The "Electronic Prior Authorization Accelerator" may also be referred to as "the Accelerator" throughout the rest of this guide.
- The terms PBM/payer or processor, who acts on behalf of the PBM/payer, are referred to as PBM/payer throughout this guide.
- The term third-party processor is used to describe the following:
  A third-party that processes prior authorizations (PA) on behalf of the PBM/payer or health plan.
- A health plan that processes their own PAs rather than delegating PA processing to their PBM/payer.

Document References

Please reference the following documents when reading this guide:

Surescripts provided materials
Connectivity and Authentication Implementation Guide
Surescripts Directory 6.2 Implementation Guide
Error Guidance Supplement - Provides common errors and specific recommendations related to SCRIPT TransactionErrorCodes(s), DescriptionCode(s), and other errors experienced on the network.
Electronic Prior Authorization Translation Matrix - Provides details on translation between different versions of the Surescripts implementation of the NCPDP SCRIPT standard.

External materials to be obtained by the customer (available at www.NCPDP.org)
NCPDP SCRIPT Standard 2023011 files (please note that the schema files version may change and will not be considered final until published in the Generally Available guide): • XML files (schemas) 20230114 • NCPDP SCRIPT Implementation Guide V2023011 (March 2024 republication) • XML Standard Guide V2022071 **Note:** The XML Standard only updates when the guide changes. The latest version is 2022071. NCPDP does not retroactively update XML versions with each new SCRIPT/Specialized Standard. This Standards Matrix shows the XML Guide version for each Standard. • V2023011 SCRIPT Standards Examples Guide V10
NCPDP SCRIPT Implementation Recommendations (latest version)
NCPDP External Code List 202301 (NCPDP online tool)

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

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.

> 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 certification testing is complete and the contract is approved by the Surescripts Certification Review Board, the customer is ready to move into production. Surescripts will configure the production connection and ensure successful operations with the customer.

Prior to transition into production, Surescripts Account Management will work with the customer and internal Surescripts teams to review the following items:

- **Production support contacts**

To be captured in an Escalation Matrix—a tool that specifies how Surescripts should contact customers for support needs, including case routing and escalation paths.

- **Support process and training**
- **Support hours**

> The Escalation Matrix for use by Surescripts customers can be found in the *Surescripts Network Operations Guide*.

Surescripts Terminology Usage

For terminology usage throughout this guide, consider the following:

Term	Term Usage
must	Requirements that are enforced as part of the production code or by Surescripts business rules. **Note:** Some business rules reside in the element details section of the 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.
PA.###	Designates a Surescripts Electronic Prior Authorization ACR.
S.###	Designates an ACR that is shared with other Surescripts products.

> Application certification requirements are found throughout the guide and summarized in the **Application Certification Requirements**.

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 includes:

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

Unprintable characters, such as control characters, are not used within the field sets. Defined unprintable characters are used as delimiters.

UTF-8 is the required character encoding for XML. Other encoding formats are not supported by Surescripts.

Note: It is recommended that customers declare their UTF-8 formatting in the message header.

XML Escape Character

Most symbols should be 'safe' within most alphanumeric fields; however, XML validation has issues with some characters, such as 'less than' (<) and 'ampersand' (&); therefore, these symbols must be properly escaped or unescaped. The quote and apostrophe only need to be escaped when used in an attribute. The most common escape encodings are:

ASCII Characters	XML Escape
quote (")	&quot;
apostrophe (')	&apos;
ampersand (&)	&amp;
less than (<)	&lt;
greater than (>)	&gt;

> String lengths are calculated based on the un-escaped value. For example, ">98 degrees" should be encoded as ">98 degrees" but it would only be counted as a string length of 11, not 14. The quote and apostrophe only need to be escaped if they are used within an attribute.

Timeouts

For timeouts, consider the following:

- When sending a message to Surescripts, the initiator should set the HyperText Transport Protocol (HTTP) timeout to no less than 30 seconds.
    - A receiving system must reply with a valid response within 24 seconds.

Valid responses include:

- Status message (Code 010 – Synchronous Response Indicating Successful Receipt)
- Error message
- Status message (Code 000 – Synchronous Acknowledgement – Further Response to Follow)

- When a Status (Code 000) is sent as a reply, a receiving system must follow-up with a Verify or Error within 24 hours, as Surescripts may return a timeout error.

> Refer to the Acknowledgement Messages in **Messages Overview** for more information on using Status, Error, and Verify messages.

Compliance

The goal of Surescripts 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. Messages Overview

Message Descriptions

The prior authorization messages are not dependent on a particular workflow. There are two possible prior authorization message flows: solicited and unsolicited. In the solicited model, the provider vendor notifies the PBM/payer that they wish to start the prior authorization process to determine if an authorization is needed for the patient and desired medication. In the unsolicited model, the provider vendor presumes that an authorization is needed and submits the information they anticipate the PBM/payer needs. Provider vendors and PBM/payers are required to support all message types outlined below.

PAInitiationRequest

In the solicited model, this message is used by the provider vendor to initiate the prior authorization process by notifying the PBM/payer of the patient and the medication for which prior authorization is being requested, along with the provider vendor's information and other related details. Provider vendors are encouraged to include any available relevant information when populating the PAInitiationRequest to receive the timeliest decision possible. The more detail provided up front, the less the PBM/payer must come back and ask for later.

PAInitiationResponse

In the solicited model, the PBM/payer returns a PAInitiationResponse after receiving the PAInitiationRequest. This message could contain either a set of questions established by the PBM/payer for the prior authorization or a request for information sent in lieu of the question set. If no PA is required, the PBM/payer sends a closed PAInitiationResponse without a question set. A PAInitiationResponse should be returned for each PAInitiationRequest.

> For Prior Authorization Renewal Requests (Unsolicited PAInitiationResponse), the PBM/payer does not receive a PAInitiationRequest before sending the PAInitiationResponse. The PBM/payer sends the PAInitiationResponse to Electronic Prior Authorization Initiate (EPAINI) and Surescripts performs a provider lookup to identify the receiving Surescripts Provider Identification (SPI).

The PAInitiationResponse from the payer should identify all information required to complete the PA determination. This allows a determination to be made in one exchange of the PARequest and response messages.

The PAInitiationResponse is for the medication (name, strength, and dosage form) indicated in the PAInitiationRequest. When responding, the PBM/payer should not alter the values indicated in the PAInitiationRequest (e.g., replying with a generic product equivalent to brand product).

With the availability of the Prior Authorization Renewal Request (also known as the Unsolicited PAInitiationResponse), this allows PBM/payers to send a question set for an expiring PA case. If the EHR is not certified to receive this, Surescripts will attempt to deliver the question set to a PA Gateway Worklist. If this option is also not available, Surescripts will send a PDF notification of the renewal via Clinical Direct Message or Fax.

PARequest

The provider vendor presents questions for the provider to answer and/or extracts information from the patient's electronic medical record, then sends the information to the PBM/payer in the PARequest message.

PAResponse

The PBM/payer determines whether authorization can be granted and provides the determination to the provider vendor in the PAResponse message. In some cases, the PAResponse message may indicate that the PBM/payer needs additional information in order to make a determination. The payer will also indicate whether it supports electronic appeals for this case. Provider vendors implementing an ePA solution should ensure that they have determined how they will support all possible response statuses.

PAAppealRequest

The PAAppealRequest message enables the provider vendor to obtain the information required to submit an appeal. The PAAppealRequest message also enables the provider vendor to submit the appeal information for a prior authorization determination.

> The PAAppealRequest button will only be available when the IsEAppealSupported flag is set to "1" in the PAResponse.

PAAppealResponse

The PAAppealResponse message provides information from the PBM/payer to the provider vendor on what is needed for an appeal. The PAAppealResponse message is also used by the PBM/payer to indicate the outcome of an appeal. In some cases, the PAAppealResponse message may indicate that the PBM/payer needs additional information in order to make a determination.

PACancelRequest

The provider vendor sends the PACancelRequest to a PBM/payer to indicate that they are no longer pursuing their most recent request. This can be sent any time after the PAInitiationResponse with the PACaseID is received by the provider vendor. The provider vendor should ensure that, if changes are permitted to a prescription with an associated PA, the prescription and PA remain in sync. If a prescription changes, the original PA should be cancelled. A determination should be made as to whether a new PA is needed based on the new prescription details.

> PACancelRequest messages should only be sent for PA cases that are still considered open and valid at the PBM/payer. If a closed PAInitiationResponse, PAResponse, or PAAppealResponse determination has been returned to the provider vendor, a PACancelRequest should not be sent to the PBM/payer.

PACancelResponse

The PBM/payer sends the PACancelResponse to a provider vendor to indicate whether the PARequest was canceled. This is only sent if a PACancelRequest is sent by the provider vendor to the PBM/payer.

Acknowledgement Messages

> The terms "free-standing" and "asynchronous" are interchangeable.

Status (Required)

A Status message is used to indicate to the sender that the message was accepted for processing. This is always a synchronous response. Status (Code 010) indicates that message processing is complete and was successful. Status (Code 000) indicates that the final response is pending and the process is not complete. A subsequent acknowledgment message (Error or Verify) should be expected.

Error (Required)

This NCPDP SCRIPT message transmits that an error has occurred, indicating the request was terminated. An error can be generated when there is a communication problem or when the message actually had an error (e.g. a formatting problem).

Verify (Required)

This NCPDP SCRIPT message is returned only as an asynchronous response as a follow-up to Status (Code 000). The Verify message confirms that the receiver successfully received and accepted the message. The process is complete and a subsequent acknowledgment message should not be expected.

Message Type	Code	Meaning(s)	As Sender	As Receiver
Status	000	The transmission has been accepted for processing but has not yet reached its final status. There should be a final Verify (010) or Error forthcoming.	Follow with an asynchronous (free-standing) Verify or Error to complete the message flow.	While an asynchronous Verify or Error is expected to follow a Status 000, if it is not received, no assumption should be made as to whether the original message has been received successfully or not. The recommendation is to wait 24 hours before following a manual process or sending a duplicate message.
Status	010	Confirms successful delivery and acceptance of a message to its final destination. The final acknowledgment message to close the transaction workflow. No further acknowledgment message should follow.	Synchronous response indicates that message processing is complete.	Subsequent acknowledgment message should not be expected.
Verify	010	Asynchronous message that confirms successful delivery and acceptance of a message to its final destination. The final acknowledgment message to close the transaction workflow. No further acknowledgment message should follow.	Send only if the original message was responded to with a Status 000.	Subsequent acknowledgment message should not be expected. Do not respond to a free-standing Error or Verify with a new free-standing Error or Verify.
Error	For a full list of NCPDP Error codes and their descriptions, see the NCPDP External Code List or schema.	Can be generated when there is a communication problem or when an error occurs during the processing of the message. Closes the transaction workflow. Subsequent acknowledgment messages should not be expected. Could be synchronous or asynchronous. A synchronous error is sent in response to a message.	Send to acknowledge that the transaction failed. Do not send if a transaction workflow has already been closed (Status/Verify Code 010). Send as soon as possible after a Status (Code 000) has been sent.	After receiving an Error, do not expect subsequent acknowledgment messages. Do not respond to a free-standing Error or Verify with a new free-standing Error or Verify.

Message Type	Code	Meaning(s)	As Sender	As Receiver
		An asynchronous error (also called Free-Standing Error) is initiated as a new message after the Status (Code 000) is sent.	Customers may create and return unique error descriptions which may further indicate why failure has occurred.	

Detailed Electronic Prior Authorization Message Scenarios

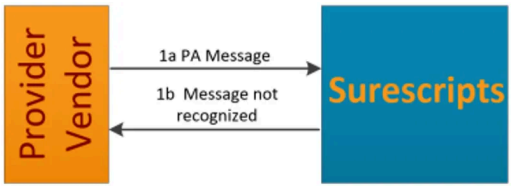
The following diagrams depict various scenarios where Verify, Error, and Status messages are sent in response to a Prior Authorization message.

> This applies to any of the ePA messages for all scenarios.

Surescripts Detailed Electronic Prior Authorization Message Flow Scenarios

> In the following scenarios, if Surescripts routes the PAInitiationRequest to a third-party processor, an error returned to the provider vendor will be sent from a different party than where the request was originally addressed. For example, a PAInitiationRequest sent to EPAINI could result in an error from the PBM/payer.

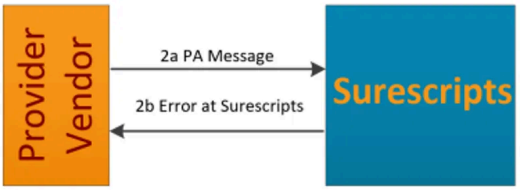
Scenario 1 - Surescripts Cannot Recognize Message:



1a) A PA message is sent to Surescripts.

1b) Surescripts cannot recognize the message and sends error text back to the provider vendor.

Scenario 2 - Surescripts Recognized Message Format but Errors Found:



2a) A PA message is sent to Surescripts.

2b) Surescripts recognizes the format as NCPDP but finds errors in the message. Surescripts returns an Error message.

4. Best Practices

The following best practices apply to all ePA message types.

DrugDescription

Customers should work with their drug database vendors to obtain the preferred drug description. The description should align with industry best practice standards as well as Surescripts requirements. It should be noted that medications that are sent with the same drug name and form may appear slightly different among customers depending on the system's drug compendia.

Examples:

Prevacid 30 MG Oral Delayed-Release Capsule

Prevacid 30 MG DR Capsule

Handling PARequests When a Health Plan Changes ePA Processing Vendor

In instances where the PBM/payer set the <DeadlineForReply> date occurring after their transition from processing the specific ePA, the PBM/payer is still to normally process any ePA message received through that date. For example, if the PBM/payer sends a question set on 12/30/2019 with a <DeadlineForReply> date of 1/5/2020 and the transition date is 1/1/2020, the ePA messages are still to be normally processed by that PBM/payer through 1/5/2020.

Prescriber Order Number

If the Prescriber Order Number (PON) is available, it is strongly recommended to be sent.

Utilization for Non-PA Coverage Rules

Provider vendors are strongly encouraged to initiate an ePA if the target medication has an indication that prior authorization may be required.

Additionally, coverage factors can help to determine when an ePA may be required.

Example:

Coverage Limit	Example
Gender Limit	The Gender Limit indicates coverage for females only, but the doctor wants to prescribe for a male patient.
Age Limit	The Age Limit indicates a cap at 30 years of age, but the doctor wants to prescribe for a 43-year old patient.
Quantity Limit	The Quantity Limit coverage list indicates a coverage limit of 15 days for Imitrex, a migraine treatment, but the doctor wants to prescribe a 30-day treatment.
Step Medication	The Step Medication list indicates that before Bextra (a Cox-2 inhibitor) is covered, at least one medication from the Cox-1 inhibitor category must be tried (e.g. Ibuprofen), but the doctor wants to prescribe Bextra without trying a Cox-1 inhibitor first.
Step Therapy	The Step Therapy coverage list identifies medications that must be tried before the patient can receive coverage for Retin-A, a brand of dermatologic preparation used to treat Acne, but the patient has already failed treatment with generic topical tretinoin.

Sending Prior Authorization Initiation Requests

A PAInitiationRequest should only be sent when it is reasonable to expect that the selected medication requires a prior authorization. A PAInitiationRequest should not be sent for every medication. Provider vendors should use the details from the Real-time Prescription Benefit solution as the primary source of information regarding the need for prior authorization. If this is unavailable, the provider vendor should use formulary as back-up alternative.

Unlike other messages, Surescripts does not require provider vendors to determine ePA support by the PBM/payer. Provider vendors can initiate a PAInitiationRequest, no matter who the patient's PBM/payer is, whenever the provider has reason to believe that the medication requires a PA. All PAInitiationRequests should be addressed to EPAINI.

Provider vendors shall send information (Payer ID, IIN (previously BIN), PCN, etc.) from the Eligibility Response in all PAInitiationRequests. See ACR PA.206 in the **Application Certification Requirements** section for more information.

At a minimum, the PayerID (the Surescripts PBM Participant ID) is needed for Surescripts to be able to determine if the PBM/payer is enabled for ePA. If the provider vendor does not send a PayerID, Surescripts will return an error. In order to maximize the potential for a positive ePA response, the provider vendor needs to send all available fields referenced in the table below.

Identifier	271	PAInitiationRequest
IIN	2110C1/REF/02 REF/01 = “N6” Qualifier	BenefitsCoordination/PayerIdentification/IINNumber
PCN	2110C1/REF/03 REF/01 = “N6” Qualifier	BenefitsCoordination/PayerIdentification/ProcessorIdentification Number
Interchange Control Number	ISA/13	BenefitsCoordination/PayerIdentification/MutuallyDefined
Group Number	2110C1/REF/02 REF/01 = “6P” Qualifier	BenefitsCoordination/GroupID
PBM Member ID	2100C/NM1/09	BenefitsCoordination/PBMMemberID
Cardholder ID	2100C/REF/02 REF/01 = ”HJ” Qualifier	BenefitsCoordination/CardholderID
PBM Participant ID	2100A/NM1/09	BenefitsCoordination/PayerIdentification/PayerID

PAInitiationResponse Reason Codes for PBM/Payers

It is recommended to use all reason codes available and not a subset of reason codes. See **Message Business Flow Response Summary** for more information.

> The reason code "BY – Other" should only be used when the reason for the closed response is not covered under another reason code.

Electronic Appeals

When a PBM/payer chooses to accept electronic appeals for a PA determination, they should make sure to provide as much detail as possible regarding the requirements that apply. This could include any of the following within the Appeal element:

- A descriptive PANote indicating specific information that may be needed from the prescriber in order to return an approved response
- Contact information for the PBM/payer that may indicate where a prescriber can reach out for information on the PA or appeal process
- A discrete ExpirationDate that indicates the date after which an appeal will no longer be considered for the PA

Automatically Initiating ePA for RxChange Messages

RxChange messages can be sent from the pharmacy to request changes to a received prescription. For RxChange messages with a code of “P” (Prior Authorization), it is recommended to automatically initiate an electronic prior authorization.

- If eligibility information is less than 72 hours old, provider vendors should include all benefit information from the original Eligibility Response (271) that is listed in the **Sending Prior Authorization Initiation Requests** section.
- If eligibility information is more than 72 hours old, provider should send a new Eligibility Request (270) for purposes of populating the BenefitsCoordination segment of the PAInitiationRequest.

Re-Submitting and Re-Initiating PAInitiationRequests

It is recommended that provider vendors allow their users the ability to re-submit or re-initiate ePAs in the event of a PBM/payer processing error, or if the deadline for reply has passed. If the deadline has passed, the open question set should be cancelled with a PACancelRequest and the original PAInitiationRequest should be resubmitted with a unique PAReferenceID.

NewRx

To realize the full value of an ePA process, it is strongly recommended that prescriptions requiring a prior authorization are placed on hold and are released to the pharmacy only after a final determination is received from the PBM/payer. When the prior authorization is missing, some PBM/payers will deny the claim, which will require manual follow-up by the pharmacy.

- If the prescription must be sent prior to the ePA process being completed, the provider vendor should indicate it is in process using PriorAuthorizationStatus with the value of "R" (requested). This will let the pharmacy know that they should not initiate the prior authorization process.
- If the prior authorization has been completed and the PBM/payer indicates prior authorization is not required, the provider vendor should indicate that prior authorization is not required by returning a PriorAuthorizationStatus value of “N” (Not Required) in the NewRx.

PA Has Been Approved

To aid in claims processing at the pharmacy and eliminate duplicate prior auth workflows, the electronic prescription (NewRx) should contain the prior authorization approval number (if provided) and approved status. In the ePA response message, the approval number is found in the AuthorizationNumber element. In the NewRx, prior authorization information is found in MedicationPrescribed/PriorAuthorizationNumber and MedicationPrescribed/PriorAuthorizationStatus.

PA Has Been Denied

In situations where the prior authorization has been denied by the PBM/payer, the NewRx should indicate that the prior authorization was denied in MedicationPrescribed/PriorAuthorizationStatus..

PA Has Been Partially Denied

When providers receive a PartiallyDenied PAResponse, the request is closed. There are two options that may be utilized when a PartiallyDenied response is received. The provider may modify the existing NewRx and start a new PA reflecting the revised NewRx. The other option would be to modify the NewRx already on hold to reflect the details approved by the PBM/payer and release the prescription to the pharmacy. If releasing the NewRx with modified details, the status should reflect “A” (Accepted) and, if provided by the PBM/payer, include the AuthorizationNumber. This is not applicable for specialty pharmacies. In this type of instance, the specialty pharmacy should notify the prescriber of the response by the PBM/payer and let the prescriber know what they can change if they want to send a new prescription.

PAReferenceID and NewRx MessageID

Provider vendors may find it easier to track and associate ePA messages with NewRx messages by sharing a common identifier. Provider vendors can utilize a shared value for NewRx MessageID and PAReferenceID of PAInitiationRequest messages in order to provide efficiencies for troubleshooting and other tasks that may require knowledge of which NewRx had association with which ePA.

Attachments

Surescripts Accelerator supports the sending of PDF attachments. It is recommended that provider vendors enable this functionality to allow additional documentation to be included with prior authorization requests.

Only send an attachment if the question set specifically asks for it. If you do not receive a request for an attachment, do not send one. Sending unnecessary attachments may result in manual review and delays. Message size (including attachments) is limited to a maximum of 20MB.

PBM/payers and provider vendors should allow users to view and send PDFs, at a minimum, to keep the prior authorization workflow electronic. However, PBM/payers should populate the QuestionSet/Header/QuestionSetContactCommunicationNumber/Fax/Number field in ePA response messages so provider vendors know where to fax any requested attachments they are unable to send electronically.

If an attachment must be sent through fax, the fax should include the PACaseID from the PAInitiationResponse to aid in linking it to the electronic messages. The provider vendor should also include in the QuestionSetComment field of the PARequest that they are faxing documentation.

NDC Codes

Provider vendors should send the Representative National Drug Code (NDC) of the medication, if available, in the PAInitiationRequest. The NDC is not required by the NCPDP standard, but PBM/payers may do additional validation on this field or may return errors if not included. Sending the NDC can help ensure that the medication can be easily identified in the PBM/payer system.

A Representative NDC is an 11-digit code that depicts a category of medication regardless of package size and manufacturer/labeler. Representative NDCs should be used in the PA process to maximize the opportunity that the selected value exists among the various medication files. A Representative NDC should not be a repackaged, obsolete, private label, or unit dose NDC unless it is the only available identifier.

Note: The drug database compendia should be current so the most applicable NDC available for the requested medication can be sent.

PBM/payers should not expect that provider vendors are always populating the NDC in the PAInitiationRequest and should consider additional medication matching logic in cases when the NDC is not populated. Surescripts may verify NDCs included in PAInitiationRequest messages. In the case of an obsolete NDC being discovered upon verification, Surescripts may replace the original NDC with a non-obsolete Representative NDC. When Surescripts replaces NDCs, Surescripts will provide notice of the change to the prescriber via the PANote in the corresponding PAInitiationResponse.

Empty Tags

Do not send empty elements within XML messaging. Sending empty tags may cause a message to be rejected or error out.

Controlled Substances

Provider vendors are able to use Electronic Prior Authorization for controlled substances even if they are currently not enabled for EPCS on the Surescripts network. There are no prohibitions in the NCPDP SCRIPT standard regarding the use of ePA for controlled substances. However, if the NewRx has been placed on hold, the release of such prescriptions should not be automated. Any EPCS prescriptions must continue to adhere to DEA audit requirements including the support of two-factor authentication and ensuring that only providers with a DEA license are permitted to issue such prescriptions.

5. Element Details

Requirement Designation

Element Attributes:

Code	Description
mandatory	The element must be used per the specification (e.g., XML schema validation).
conditional	The element is to be used per the conditions specified.
business rule	If sent, the element must be used per the Surescripts business rule. **Note**: Not all business rules reside in this table.
recommended	Surescripts recommends sending the element as a best practice.
optional	Some fields do not have specific conditions. Data should be sent if available.
not used	Not used by Surescripts.

> This guide includes data elements only where Surescripts has specific requirements or further explains the field usage. Refer to the schema for a complete list of fields.

Element Tables

Message Header Elements

Element	Code	Comments
To	mandatory	The identification of the intended receiver of the message. This field contains the identification number of the recipient. For all PAInitiationRequests, this field contains EPAINI. For all subsequent Request messages, it will contain the Participant ID of the PBM/payer/third-party processor. For Response messages, this field contains the SPI of the provider vendor.
@Qualifier	conditional	business rule If the To field is sent, then the Qualifier must also be sent. Values: **D** = Prescriber ID (Used for SPI) **PY** = Payer (PBM/payer or third-party processor Participant ID) **ZZZ** = Mutually Defined (used when sending to EPAINI)
From	mandatory	The identification of the sender of the message. This field contains the identification number of the sender. This field contains the SPI (Surescripts Provider ID) of the provider vendor, the Participant ID of the PBM/payer or third-party processor, or EPAINI.
@Qualifier	conditional	business rule If the From field is sent, then the Qualifier must also be sent. Values: **D** = Prescriber ID (Used for SPI) **PY** = Payer (PBM/payer or third-party processor Participant ID) **ZZZ** = Mutually Defined (only used when Surescripts is responding from EPAINI)
MessageID	mandatory	The sender should ensure the combination of the From and MessageID elements is unique for at least 18 months. Surescripts recommends an unformatted Globally Unique Identifier (GUID).
RelatesToMessageID	conditional	Relates to Message ID is required by Surescripts. This field is the MessageID of the most recent and relevant message in the workflow, including outside of only ePA conversational messaging. This is required for PAInitiationRequest, PAInitiationResponse, PARequest, PAResponse, PAAppealRequest, PAAppealResponse, PACancelRequest, PACancelResponse, Error, Verify, and Status.
SentTime	mandatory	Date of the transmission. Please see **UTC Time Format** section.
SenderSoftware	mandatory	Identifies the entity responsible that generated the message. Contains the software vendor name, product, and version/release. These fields were previously blocked by Surescripts when passing the message on to the receiver. They will now be passed on to the receiver.
SenderSoftwareDeveloper	mandatory	Identifies the entity responsible for the software that generated the message. The developer may be a software vendor or, if the software was developed "in-house", the developer is the entity actually sending the message (e.g., a chain). The value transmitted is determined by the Sender Software Developer.
SenderSoftwarePro	mandator	The text field that identifies the software developer's product, as determined by the software developer.

Element	Code	Comments
duct	y	
SoftwareVersionRelease	mandatory	The text field that identifies the version and release of the software product, as determined by the software developer.
TestMessage	conditional	When sent, allowable values must match platform that message is being sent to. Values: **1** = Test (Required in staging environment) All other values = Live (Production) Surescripts does not recommend sending this field within the production environment.
Message Indicator Flag	conditional	Not applicable to prior authorization transactions.
ISA13	conditional	This is the Interchange Control Number ISA13 value from the Eligibility response X12 271 transaction.

PAInitiationRequest Elements

> Refer to the schema for a complete list of fields, including details for PAInitiationResponse elements.

Element	Code	Comments
PAInitiationRequest **Note:** It is recommended to populate as many conditional elements as possible to maximize the prior authorization experience.		
PAReferenceID	mandatory	Assigned by the prescribing system on the initial message and is used as a tracking identifier on all request and response prior authorization messages to tieback related prior authorization messages. The identifier must be unique per prescribing system for at least 18 months. **Note:** PAReferenceID shall be established with the PARequest when responding to a Prior Authorization renewal request (unsolicited PAInitiationResponse).
PAPriorityIndicator	recommended	Indicates the requested PA processing priority. If PAPriorityIndicator is not populated, standard priority will be assumed. Values: **S** - Standard Priority **X** - Urgent/Expedited Priority **Note:** This value should be echoed on subsequent transactions. The priority may also be changed on the PARequest if the provider feels that the patient's health is at risk.
BenefitsCoordination Note: The fields below may be returned in the Eligibility Response (271). If a PBM/payer returns active coverage, these fields should be populated with the returned value. See Benefits Coordination table in **Sending Prior Authorization Initiation Requests** for details on how the 271 response should be mapped to BenefitsCoordination.		
PayerIdentification	conditional	
PayerID	conditional	Surescripts Participant ID assigned to the PBM/payer.
ProcessorIdentificationNumber	conditional	The PCN associated to the patient's benefit.

Element	Code	Comments
IINNumber	conditional	The IIN associated to the patient's benefit.
MutuallyDefined	conditional	The 271 message's ISA13 control number.
PayerName	conditional	PBM/payer name.
CardholderID	conditional	Assigned cardholder ID for the patient. **Note:** PBM/payers may validate patients based on cardholder ID.
CardHolderName	conditional	Cardholder name. **Note:** PBM/payers may validate patients based on cardholder name.
GroupID	conditional	ID number identifying the health plan group.
GroupName	conditional	Name of the health plan group.
PBMMemberID	conditional	PBM/payer-assigned member ID for the patient.
Patient **Note:** Refer to the schema for a complete list of fields.		
Identification	mandatory	Usage requirements: • If patient identification data exists, include the Identification element and populate its child elements. • If no patient identification data is available, include the Identification element but leave it empty (no child elements).

Element	Code	Comments
SocialSecurity	conditional	Patient's Social Security Number. Not recommended to send unless explicitly required by a specific business or regulatory process.
GenderAndSex	mandatory	Values: • **M** - Male • **F** - Female • **U** - Not Specified or Unknown • **N** - Non-binary - An umbrella term for people with gender identities that fall somewhere outside of the traditional conceptions of strictly either female or male.
PregnancyIndicator	conditional	Code indicating the patient as **pregnant** or **non-pregnant**.
Address	mandatory	Parent element for patient address information.
AddressLine1	mandatory	Patient's primary address line.
CIty	mandatory	Patient's city.
StateProvince	conditional	Changed from **Mandatory** in v2017071 to **Conditional**. Recommended for optimal EPA experience.
PostalCode	conditional	Changed from **Mandatory** in v2017071 to **Conditional**. Recommended for optimal EPA experience.
Prescriber **Note:** The prescriber relates to the person requesting the prior authorization.		
Identification	manda	Some PBM/payers may validate prescriber data against NPPES or

Element	Code	Comments
	tory	other prescriber directories using the fields under Identification.
NPI	mandatory	NPI of the prescriber.
Name	mandatory	Surescripts may update Prescriber/Name/FirstName and Prescriber/Name/LastName with data from the Surescripts Directory in PAInitiationRequest messages prior to sending them to PBM/payers.
Specialty	mandatory	Surescripts may update Prescriber/Specialty with data from the Surescripts Directory in PAInitiationRequest messages prior to sending them to PBM/payers.
Address	mandatory	Parent element for prescriber address information.
AddressLine1	mandatory	Prescriber's primary address line.
CIty	mandatory	Prescriber's city.
StateProvince	conditional	Changed from **Mandatory** in v2017071 to **Conditional**. Recommended for optimal EPA experience.
PostalCode	conditional	Changed from **Mandatory** in v2017071 to **Conditional**. Recommended for optimal EPA experience.
MedicationPrescribed		
DrugDescription	mandatory	Drug name (full drug name, strength, and form). For a compound, the names of the ingredients in the compound or the prescriber's preferred name should be communicated here. See DrugDescription Best Practices section for more information.

Element	Code	Comments
DrugCoded	conditional	
NDC	mandatory	The NDC must be a valid 11-digit code and should meet the qualifications for a Representative NDC.
ProductCode	conditional	Product Identifier. The Representative NDC is recommended unless the message contains Quantity Code List Qualifier “CF” or the CompoundInformation element is present. See **NDC Codes** for more information.
Strength	conditional	
StrengthForm/Code	conditional	Must be a valid FMT code. See **Codified Field Usage for Federal Medication Terminologies (FMT) Codes** for details.
StrengthUnitOfMeasure/Code	conditional	Must be a valid FMT code. See **Codified Field Usage for Federal Medication Terminologies (FMT) Codes** for details.
Quantity	mandatory	
QuantityUnitOfMeasure	mandatory	Must be a valid FMT code. See **Codified Field Usage for Federal Medication Terminologies (FMT) Codes** for details.
PADaysSupplyDosePerDay	conditional	MedicationPrescribed/DaysSupply or PADaysSupply and PADosesPerDay must be a whole number. Just one of these fields need to be submitted. **Note:** In order to allow PBM/payers to provide the most accurate response, provider vendors should include DaysSupply for the requested medication within PAInitiationRequests. PBM/payers use days supply in order to determine if prescribers are requesting a quantity outside of the plan limit. Failure to provide days supply

Element	Code	Comments
		could result in unnecessary question sets being returned to prescribers or additional work. Surescripts may parse the SIG text field and use results to populate the DaysSupply field when it is not sent in a PAInitiationRequest message prior to sending to the PBM/payer.
Diagnosis	conditional	Provider vendors should send the valid version of diagnosis codes in the PAInitiationRequest. Sending diagnosis codes may speed up the process. **Note:** If sending ICD-10 codes (Qualifier ABF), sending systems should not include decimal codes for message/submission.
ClinicalInformationQualifier	mandatory	Qualifier for how the diagnosis was obtained. Values: **1** - Prescriber/Prescriber Supplied **2** - Pharmacy Inferred **3** - Patient **4** - Other Source
Qualifier	mandatory	Defines the code list used for the diagnosis code.
Value	mandatory	Code for the diagnosis.
Description	mandatory	Description of the diagnosis code.
Sig	conditional	
SigText	mandatory	SigText will be used to convey patient instructions for use or administration and should be constructed and transmitted in a patient-friendly manner. Sigs should not be truncated because important information can be lost.
Provider **Note:** The provider loop can offer additional contacts beyond the prescriber if there were follow up questions to the PAInitiationRequest.		

Element	Code	Comments
Identification	conditional	
NPI	mandatory	NPI of the initiating party.
Name	mandatory	Initiating party's name. See ACR PA.204 in the **Application Certification Requirements** section for more information.
CommunicationNumbers	mandatory	
PrimaryTelephone	mandatory	Initiating party's phone number. See ACR PA.204 in the **Application Certification Requirements** section for more information.
Fax	conditional	Initiating party's fax number.
Address	mandatory	Parent element for provider address information.
AddressLine1	mandatory	Provider's primary address line.
CIty	mandatory	Provider's city.
StateProvince	conditional	Changed from **Mandatory** in v2017071 to **Conditional**. Recommended for optimal EPA experience.

Element	Code	Comments
PostalCode	conditional	Changed from **Mandatory** in v2017071 to **Conditional**. Recommended for optimal EPA experience.

Codified Field Usage for Federal Medication Terminologies (FMT) Codes

For a list of FMT codes, go to: **http://evs.nci.nih.gov/ftp1/NCPDP/About.html**. Use the valid value from the appropriate row of the downloaded data. This list should be updated on a monthly basis.

The table below includes some example fields.

Codified Field/Field Code	Comment
StrengthForm/Code	Use the NCPDP StrengthForm Terminology rows from the downloaded data. This list should be updated on a monthly basis.
StrengthUnitOfMeasure/Code	Use the NCPDP StrengthUnitOfMeasure Terminology rows from the downloaded data.
QuantityUnitOfMeasure/Code	Use the NCPDP QuantityUnitOfMeasure Terminology rows from the downloaded data.
DoseUnitOfMeasure/Code	Use the NCPDP DoseUnitOfMeasure Terminology rows from the downloaded data.
DEASchedule	Use the NCPDP DEASchedule Terminology rows from the downloaded data.

6. Accelerator Integration Approach

About Electronic Prior Authorization Accelerator

The Accelerator delivers a method for provider vendors to quickly offer customers core ePA features by embedding an iframe (or using a web browser control) in the provider vendor system that will manage the process of viewing and responding to an ePA.

The Accelerator is comprised of two APIs:

- The Web Application API provides a customizable user interface that is displayed in an iframe (or a web browser control) in the provider vendor system
- The Web Service API gives programmatic read-only access to the current state of workflow processes and tasks

The Web Application API interface interacts with the Accelerator and its managed workflow. To use the Web Application API, the provider vendor system must first call the Web Service API to request and receive an authorization token.

> Authorization tokens are issued for 60 minutes and contain the expiration date and time. A new authorization token is not required to relaunch the Accelerator if still active or before the expiration period has elapsed.

When a user sends a PAInitiationRequest via the native Electronic Prior Authorization route, the Accelerator workflow engine creates a workflow process. The workflow process is given an InitiatorsProcessID, which is the PAReferenceID from the PAInitiationRequest. The workflow engine tracks all subsequent messages related to the PAInitiationRequest and distributes workflow tasks related to the process.

Users can interact with a worklist, which is the collection of active tasks that require action from the current user. They are also able to receive read-only information regarding the status of individual tasks or processes using Web Service APIs.

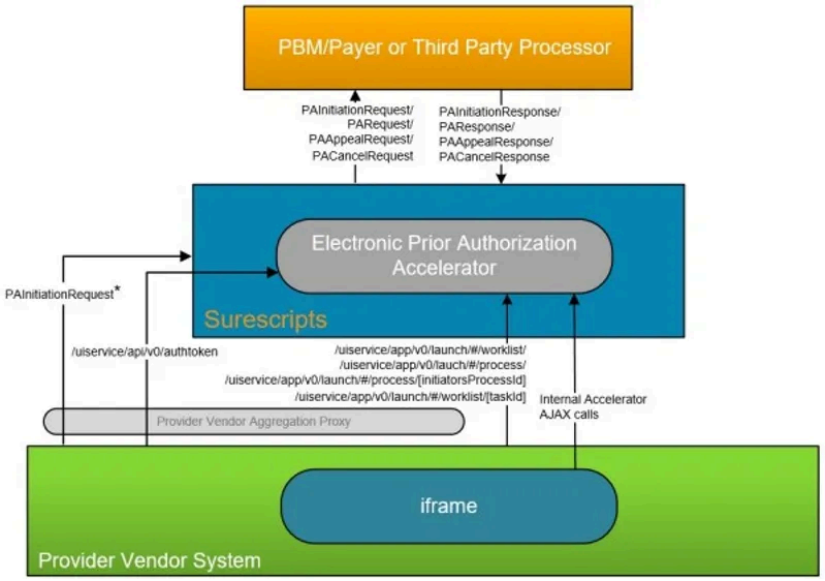
When an Accelerator process reaches a point where the user needs to act on a task (e.g., complete a question set) the user can complete the task via the user interface provided by the Web Application API. Additionally, the Accelerator provides a historic view of in-process and completed ePA activities.

Workflow Integration Approach

Accelerator supports the ePA portions of a provider vendor's workflows. The provider vendor remains responsible for related e-prescribing processes (e.g., creating a NewRx, communicating with the pharmacy, and managing users) and must also determine how best to integrate the Accelerator features into their application.

High-Level Accelerator Process Flow

The following diagram displays the high-level process flow for the Accelerator.



> The asterisk (*) in the process flow above marks the start of the process.

1. The provider vendor sends a PAInitiationRequest to EPAINI (the Surescripts identifier for ePA).
2. The PBM/payer is identified per the details provided in the BenefitsCoordination segment. If the PBM/payer's directory entry indicates that they support ePA, the transaction is sent to them. If the PBM/payer is not enabled for ePA, Surescripts will search for a third-party processor who will make the PA determination on behalf of the PBM/payer.
3. The PBM/payer or a third-party processor returns the PAInitiationResponse indicating:
    - Open status (a question set), or
    - Closed status

See Closed Reason Code information in the **Message Business Flow Response Summary**.

In cases where the payer indicates they are not the PA processor, Surescripts will initiate a search to find a different ePA processor. If an ePA processor is not found, Surescripts will either:

- Create a worklist task for the provider from the most recently received PAInitiationResponse indicating that no processor could be found, and a manual process should be followed, or
- Create a workflow task with a URL in the PANote field that the user can select to launch a fillable PA form via a web-based PA forms provider.

Notes:

Once the user fills out the PA form provided by the web-based PA forms provider, the form is faxed to the PBM/payer and there will be no further electronic communication to the EMR by either the web-based PA forms provider or Surescripts. The PBM/payer will fax the provider vendor back the approval or denial.

The provider vendor will receive one task based on the most recent response. Please note that the task name may differ depending on the response(s) received.

1. The provider vendor application shows the Accelerator worklist.
2. The provider selects a task in the worklist and completes the question set.
3. The Accelerator builds a PARequest from the completed question set and sends it to the PBM/payer on behalf of the provider.
4. The PBM/payer processes the PARequest and sends a PAResponse to the provider.
5. Surescripts intercepts the PAResponse and creates a task for the provider to acknowledge the response.
6. The provider reviews the result of the PAResponse in the Accelerator.

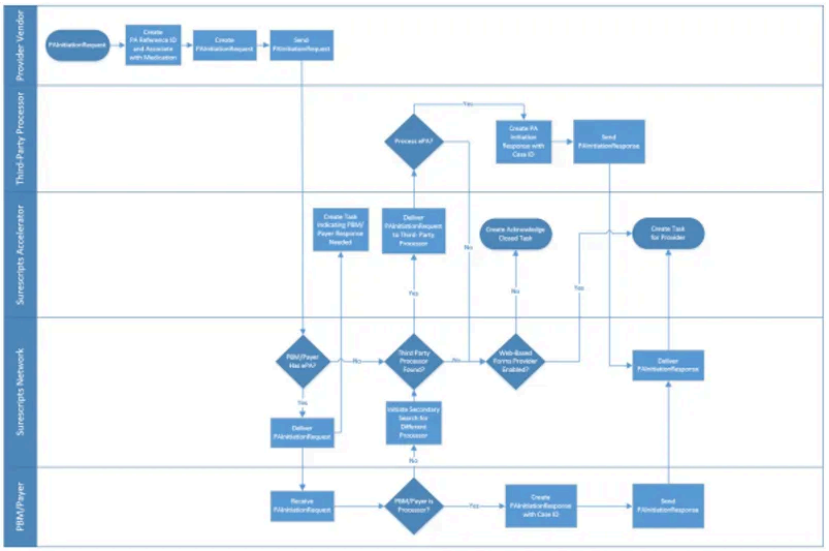
Detailed Accelerator Process Flow

The following process flows display how the provider vendor, PBM/payer, Surescripts, and the Accelerator interact.

Customers may choose to automatically acknowledge/suppress PBM/payer responses that indicate a prior authorization is not necessary. Utilizing this setting allows responses to be auto-acknowledged, cutting the overhead required by the users for the processing of such responses.

To realize the full benefit of this option, it is recommended that API calls are utilized to recognize the receipt of such a response and allow the release of non-EPCS prescriptions that are on hold when such a response is returned. See **Auto-Acknowledge "PA Not Needed" Responses** for more information. This is turned on by default; to not include this option in the implementation, please contact a Surescripts resource.

PAInitiationRequest Process Flow



Accelerator Features

The Accelerator capabilities are highlighted below:

- Enables users to view, answer, and submit the ePA Question Set — supporting all question logic.
- Automatically populates provider details, patient demographics, and medication information by mimicking what the provider vendor sent in the PAInitiationRequest.
- Enables users to view process outcomes, including authorization details.
- Provides worklist tracking to allow users to manage tasks requiring attention.
- Enables users to review completed ePA cases.

> Completed ePA cases are available indefinitely.

- Provides a mechanism for canceling processes (via user interface or API).
- Supports the ePA electronic appeal process.
- Provides role-based access controls to manage delegation of responsibilities among providers (prescribers and their staff).
- Provides ability to view PDF attachments.
- Provides ability to automatically search for an alternative PA processor, if the PBM/payer is not the processor.
- Allows for web-based PA forms provider integration with the ability to provide a PA form URL.
- Provides ability to aggregate notes from multiple processors.

Prerequisites for Using the Electronic Prior Authorization Accelerator

Use of the Accelerator requires that the prescribing system also supports:

- Eligibility request/response (X12 270 and 271 messages) for purposes of identifying PBM/payer and member eligibility.
- Formulary information in the e-prescribing process, including utilization of the PA coverage list (NCPDP Formulary & Benefits Standard) for purposes of prospectively identifying medications that may require prior authorization as well as potentially providing alternatives that the prescriber may select to avoid the need for a PA.
- In cases where the Eligibility Response (271) does not indicate coverage, the provider may utilize the Directory to initiate a prior authorization with a specific PBM/payer. For such instances, the actual payer ID must be used instead of “EPAINI”.
- Beginning the ePA process by sending a PAInitiationRequest to Surescripts.
- The ability to proxy HTTP requests for all Web Service API requests through a single aggregation server, including the auth token request. An auth token is required to use the Web Application API.
- The ability to utilize the download prompts and links to view PBM/payer attachments and forms.

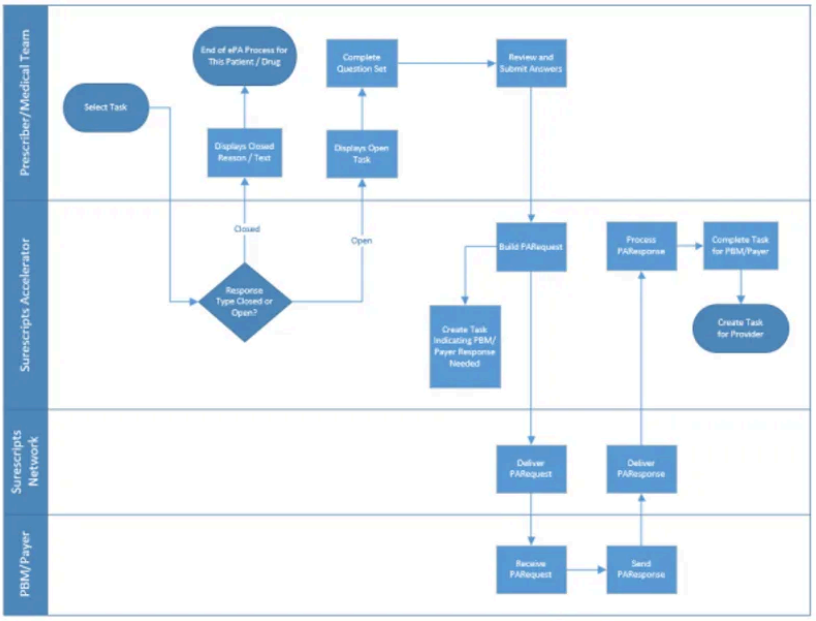
Browser Requirements

The Accelerator requires a web browser with JavaScript and session cookies enabled.

Surescripts expects, but does not guarantee, performance on supported and up-to-date web browser or client/server web browser control. To avoid delays in development, it is recommended to ensure the web browser being utilized is the most current version available.

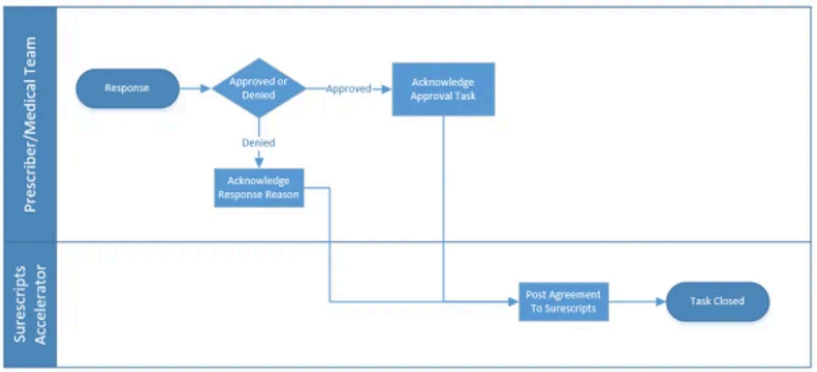
Please be aware that individual browser security settings may prevent the Accelerator from properly launching and should be evaluated based on each provider vendors' implementation. The website surescripts.net may also need to be added as a "trusted site" for proper functioning.

Open / Closed PAInitiationResponse Process Flow



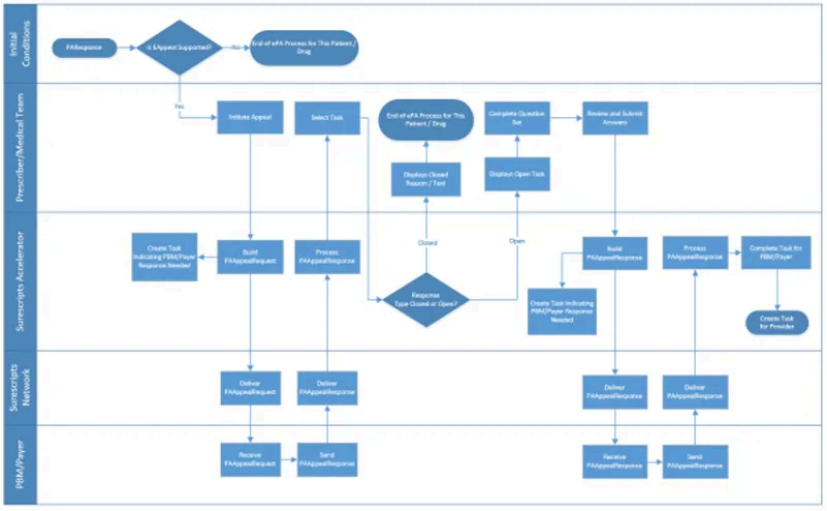
The process flow above reflects the submitter role.

Approved or Denied PAResponse Process Flow

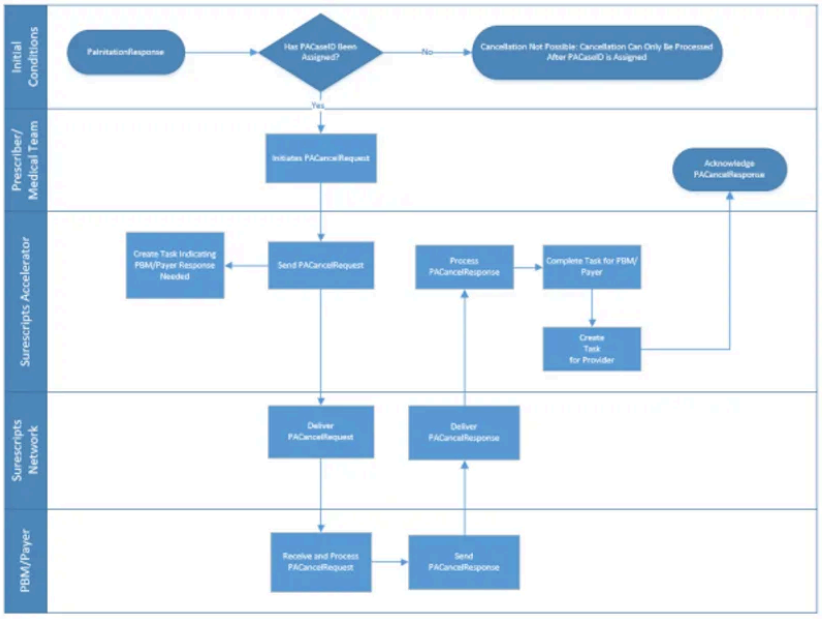


In the process flow above, both approved and denied responses are appealable. For the appeal process flow, see **Approved or Denied PAResponse Appeal Process Flow**.

Approved or Denied PAResponse Appeal Process Flow



Cancelled PAInitiationRequest Process Flow



7. Launching Accelerator with Web Application API

Accelerator Launch Process

Provider vendor systems launch the Accelerator using the following steps:

1. Provider vendor system sends an Auth Token Request through a single vendor aggregation server. The request includes the System User ID, System User Qualifier, End User ID, and roles. Please note that:
    - The user's roles are PAPreparer, PAReviewer and PASubmitter. See **Roles** for more information.
    - The user could be the prescriber or someone working on behalf of the prescriber.
2. Surescripts returns a response that includes the token.
3. The provider vendor system launches the Accelerator using the Token. The launch includes styling parameters as well as launch options. See **Accelerator Launch Options** for more information on worklists and task history.
4. The provider vendor system closes the session.

User Security

The provider vendor is responsible for authenticating users and determining what data and activities each user will have access to. This is done by specifying the End User ID and permissions in the Auth Token Request. In response, the Accelerator generates a token that specifies the user's access information and activities based on the permissions specified in the Auth Token Request.

At a high level, the provider vendor and the Accelerator share responsibilities for maintaining and enforcing user authentication and access controls for product functions. Responsibilities are described below.

The provider vendor:

- Maintains its users, their roles, their relationships (e.g., delegation of authority from providers to staff), and their authentication credentials, etc.
- Includes pertinent user information when initiating the Accelerator's user interface and task list components, including:
    - The End User ID (as defined by the provider vendor system).

    > The provider vendor must ensure the End User ID is unique across their customer base. The End User ID is viewable on the user interface.

    - An assertion that the user has been authenticated by the provider vendor system.
    - The specific roles the user possesses.
    - The providers on whose behalf the user can act.

The Accelerator:

- Logs the End User ID (as defined by the provider vendor system) and activity.
- Enforces the user's role(s) assigned by the provider vendor system.
- Enforces visibility of data according to the delegation information supplied by the provider vendor system.

Authentication and Authorization

Users are authenticated by the provider vendor system. The Accelerator tracks the End User ID of the individual signed into the provider vendor system.

The provider vendor system authorizes a user to perform a task by assigning them to a role. The types of roles are:

- PA Reviewer: The PA Reviewer has read-only permissions.
- PA Preparer: The PA Preparer has create, read, and edit permissions.
- PA Submitter: The PA Submitter has create, read, edit, and submit permissions.

The user may have different roles depending on their relationship to the prescriber (e.g., a user may have the PA Submitter role when working on behalf of Dr. Smith and the PA Reviewer role when working on behalf of Dr. Jones).

The users and their roles are not stored by Surescripts. The provider vendor system must securely communicate the authenticated End User ID and roles to Surescripts.

The table below shows the role(s) authorized to perform each task.

Role/Permission Map

Activity	Description/Notes	PAReviewer	PAPreparer	PASubmitter
View Worklist	View and select tasks that require action. **Note**: Tasks may also be filtered by the provider or patient included in the auth token.	X	X	X
View Response Details	Read-only access to all response data including answers to questions for an Open response.	X	X	X
Prepare Open Response	Answer questions on Open response.		X	X
Submit PARequest	Take the answers created from 'Prepare Open Response' and submit the PARequest message to PBM/payer.			X
Acknowledge Closed	Ends the workflow process for the associated ePA case.		X	X
Acknowledge Denial	Ends the workflow process for the associated ePA case.		X	X
Acknowledge Approval	User indicates agreement with approval restrictions (if any). Ends the workflow process for the associated ePA case.		X	X
Create Appeal	Add information required to submit initial PAAppealRequest and submit.		X	X
Submit Appeal	Take the answers created from 'Prepare Open Response' and submit the PAAppealRequest message to PBM/payer.		X	X
Cancel Process	Submit PACancelRequest request to the PBM/payer.		X	X
Acknowledge Cancel	Ends the workflow process for the associated PA case.		X	X

Auth Token Request

The Web Service API provides an endpoint for requesting an auth token that is used when launching the Accelerator. The auth token identifies user and their role(s), and must be used when the Accelerator is launched. The token may be re-used for multiple launches until it expires.

> The request type is "POST", the Content-Type must be set to "application/json", and the request parameters are JSON encoded as part of the request body. The token follows the JWT pattern and requires that the roles for a particular session be included in the request.

The JSON fields in the following example are defined per the table below:

JSON Fields	Notes
systemUserId	The Data Provider ID (identification of the sender of the message). Use the Network Participant ID provided in your Surescripts Activation Form (for example, ABC123).
systemUserQualifier	The user qualifier assigned to the provider vendor system. The systemUserQualifier = "DP"
endUserId	ID of the user logged into the provider vendor system. **Note**: This value is presented to authorized viewers in Task History details for ePA processes. Provider vendors should use appropriate values to assist in tracking and troubleshooting tasks and processes. endUserID must be less than 250 characters and unique across provider vendor customers.
roles	A list of roles assigned to this provider. **Note**: When Common Roles are included in the request, Roles assigned for Providers will be ignored and should not be sent.
products	Required if customer needs a specific product. If not sent, all products will be returned. Example: epa
id	Product Id. At least one product Id is required if 'products' is used. **Value**: epa
processTypes	Required if process types filter is needed. **Note**: Process types are not currently used for Electronic Prior Authorization. **Value**: null

Auth Token Request Example

POST https://[host]/uiservice/api/v0/authtoken HTTP/1.1

Host: localhost

Connection: keep-alive

Content-Length: 85

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36

Authorization: Basic ZGVtb0Jhc2IjQXV0aElkOmRlbW9CYXNpY0F1dGhQdw==

Content-Type: application/json

Accept: */*

Accept-Encoding: gzip,deflate,sdch

Accept-Language: en-US,en;q=0.8

Body:

```
{
   "systemUserId": "EPA123",
   "systemUserQualifier": "DP",
   "endUserId": "fred",
   "roles": {
      "commonRoles": [
         "pasubmitter"
      ],
      "providers": [
         {
            "id": "6487851746001",
            "qualifier": "D"
         {
      ]
   }
}
```

Note: The basic authentication credentials are detailed in your Surescripts Activation Form. The Body of the request utilizes the application/json format as detailed in the table above..

Roles

A list of roles is required in order to generate an auth token. The provider vendor system assigns a role to the user, which authorizes them to perform certain tasks within the Accelerator. The provider vendor system requests an auth token, which specifies role requests for each launch of the Accelerator UI. The Accelerator encodes the requested roles into the auth token. See **Authentication and Authorization** for role(s) authorized to perform specific tasks.

Roles can be specified per the following:

- A common role can be specified across all providers assigned to that user.
- Roles can be specified on a role-per-provider basis or on a list of providers basis.

> The list of providers the user can access is managed by the provider vendor system.

- Roles can be specified on a per-patient basis or on a list of patients basis.

> The list of patients is managed by the provider vendor system.

If an attempt to create an auth token violates one of the following rules, the Accelerator will return a validation error. The rules include:

- If a role is not specified at any level (Common, Provider, or Patient)
- If a provider is specified more than once.
- If a patient is specified more than once.

Roles Definitions

Roles	Notes
Common Roles	A list of Roles that apply to all submitted providers or patients. When Common Roles are included in the request, Roles assigned for Providers and Patient will be ignored and should not be sent.
Providers	A list of Provider objects, each with a list of roles for that provider, an id, and a qualifier.
Patient	An object that describes a patient with an id, an organization, and a list of roles that apply.

Role Assignment Examples

The JSON field in the following examples is defined per the table below:

JSON Fields	Notes
roles	A list of roles assigned to this provider. **Note**: When Common Roles are included in the request, Roles assigned for Providers will be ignored and should not be sent.

Common Role Assignment

The following example shows how a single role can be assigned for all providers assigned to that user as part of an Auth Token Request.

In addition to the table at the beginning of this section, the table below details additional JSON fields specific to this example.

JSON Fields	Notes
commonRoles	In this example the common role is the pasubmitter.
id	The provider's SPI.
qualifier	The qualifier for the provider. **Qualifier**: D

Common Role Assignment Example

{

```
  "systemUserId": "EPA123",
  "systemUserQualifier": "DP",
  "endUserId": "user1",
  "roles": {
    "commonRoles": [
      "pasubmitter"
    ],
    "providers": [
      {
        "id": ""<ProviderSPI1>",
        "qualifier": "D"
      },
      {
        "id": ""<ProviderSPI2>",
        "qualifier": "D"
      }
    ]
  }
}
```


Role per Provider Assignment

The following example shows how roles can be assigned on a per provider basis.

In addition to the table at the beginning of this section, the table below details additional JSON fields specific to this example.

JSON Fields	Notes
id	The provider's SPI.
qualifier	The qualifier for the provider. **Qualifier**: D

Role per Provider Assignment Example


```
{
  "systemUserId": "EPA123",
  "systemUserQualifier": "DP",
  "endUserId": "user1",
```

```
  "roles": {
    "providers": [
      {
        "roles": [
pasubmitter
        ],
        "id": "<ProviderSPI1>",
        "qualifier": "D"
      },
      {
        "roles": [
pareviewer
        ],
        "id": "<ProviderSPI2>",
        "qualifier": "D"
      }
    ]
  }
}
```

Role per Patient Assignment

The following example shows how a specific role can be assigned for a specific patient.

In addition to the table at the beginning of this section, the table below details additional JSON fields specific to this example.

JSON Fields	Notes
id	Matches the MedicalRecordIdentificationNumberEHR field (...Patient/Identification/MedicalRecordIdentificationNumberEHR) specified in the PAInitiationRequest. This field is optional in the schema, but has to be included in the original PAInitiationRequest in order for the resulting process to be visible when launching the Accelerator for a specific patient.
organization	The value sent in the MutuallyDefined field (...Patient/Identification/MutuallyDefined) of the PAInitiationRequest. Used to identify a patient with the same ID across different locations/facilities. **Note**: If this optional field is not included in the PAInitiationRequest, it should not be included in Auth Token Request for the specified patient. The worklist items will show when patient id equals MedicalRecordIdentificationNumberEHR and organization equals MutuallyDefined. For example, if organization is null in the auth token request but MutuallyDefined is “test clinic” in the PAInitiationRequest, the task for that patient will not be displayed. ID and organization must both match what is on the PAInitiationRequest.

Role per Patient Assignment Example


```
{
"systemUserId":"EPA123",
"systemUserQualifier":"DP",
"endUserId":"fred",
"roles":{
   "patient":{
              "id":"elizabeth",
              "organization":"clinic2",
              "roles":[
               "pasubmitter"
             ]
            }
     }
}
```


Auth Token Response

The Accelerator returns the Auth Token Response in the "application/json" format and may include a success or error message. If the response is successful, the payload of the response will contain an encrypted auth token, which is used to further access the Accelerator.

Auth Token Response Examples

Successful Response Example

HTTP/1.1 200 OK

Cache-Control: no-cache

Pragma: no-cache

Content-Type: application/json; charset=utf-8

Expires: -1

Date: Thu, 06 Mar 2014 16:42:17 GMT

Content-Length: 547

{

"Token"

"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJzZWxmIiwiYXVkIjoiaHR0cDov

L3d3dy5zdXJlc2NyaXB0cy5jb20iLCJuYmYiOjE0MDcxODAyNzMsImV4cCI6MTQwNzIxNjI3My

wibmFtZWlkIjoibWIsZXMiLCJyb2xlIjoie1wiQ29tbW9uUm9sZXNcIjpbXCJwYXN1Ym1pdHRl

clwiXSxcIlByb3ZpZGVyc1wiOlt7XCJSb2xlc1wiOm51bGwsXCJJZFwiOlwiNjQ4Nzg1MTc0Nj

AwMVwiLFwiT3JnYW5pemF0aW9uXCI6bnVsbCxcIlF1YWxpZmllclwiOlwiRFwifV0sXCJQYXRp

ZW50XCI6bnVsbCxcIk9yZ2FuaXphdGlvblwiOm51bGx9Iiwic3VyZXNjcmlwdHMuY29tL3Bvcn

RhbCI6IjExNDYifQ.kKT1EEsJedbaxjKERCUsgLXxtZ7fGe8BJwTb8W3iA7E",

"UtcExpiration": "2014-08-05T05:24:33.9298548Z"

}

Error Response Example

HTTP/1.1 400 Bad Request

Cache-Control: no-cache

Pragma: no-cache

Content-Type: application/json; charset=utf-8

Expires: -1

Date: Mon, 04 Aug 2014 19:28:02 GMT

Content-Length: 148

{

"message": "The request was invalid.",

```
    "modelState":{
     "": [
       "One or more invalid common role has been specified."
     ]
    }
  }
```


Accelerator Launch Overview

If the hosting application is web-based, the Accelerator is embedded in an iframe.

For non-web-based vendor applications, the Accelerator may be embedded in web browser controls that meet the minimum browser requirements.

Embedding the Web Application in an iframe

Launch the Accelerator user interface by posting a form that targets an embedded iframe.

> The following is an example and is for illustration only.

```
<form action="https://[host_name]/uiservice/app/v0/launch/#/worklist" method="post" target="worklist" id="worklistformid">
    <input type="hidden" name="authToken" value="[auth_token]" />
    <input type="hidden" name="appPrimaryColor" value="#0099FF" />
    <input type="submit" value="Work List" />
</form>
<iframe name="worklist" id="worklist"
  style="overflow-y:hidden;width:810px;height:410px">Worklist</iframe>
```

Parameters

Parameter	Description
[host_name]	The address of the Accelerator server being used. Refer to your Surescripts Activation Form and the URL used to launch Accelerator. The route following the hostname specifies which view the Accelerator will initially display (in the case above, the worklist).
[auth_token]	The “Token” retrieved from the auth token web service.
styling parameters – e.g., appPrimaryColor in the example above	See **Styling** below for available options.

Styling

> For more information about color and font styling, refer to the associated Accelerator user documentation for your product.

Launch the iframe (or a web browser control) with a CSS similar to the following:

iframe { overflow-y: hidden;width: 810px;height: 410px; scrolling: no; border: 0;}

The following are the styles that can be set with their default values from parameters on the initial post:

- appFontFamily: "Helvetica Neue", Helvetica, Arial, sans-serif
- appFontSize: 13px
- appFontColor: #333333
- appBorderRadius: 4px
- appBackgroundColor: #FFFFFF
- appPrimaryColor: #3484d6
- appWarningColor: #D66934
- appSuccessColor: #4FD634
- appNavbar: “normal”

Navigation Bar Style Parameter

The navigation bar style parameter is named appNavbar.

> The default is “normal” which is set to 50px.

Value	Result
none	Navigation bar is not displayed
large	70px
normal	50px
small	30px

Example – Turning the navigation bar off:

<input type="hidden" name="appNavbar" value="none" />

Styling Example

POST https://<host>/uiservice/app/v0/launch/#/worklist HTTP/1.1

Body: appPrimaryColor=%23FF0000 &authToken=eyJWZ...

Connection: keep-alive

Content-Length: 419

Cache-Control: max-age=0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8

Origin: https://sur6104.surescripts.local

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/33.0.1750.146 Safari/537.36

Content-Type: application/x-www-form-urlencoded

Accept-Encoding: gzip,deflate,sdch

Accept-Language: en-US,en;q=0.8

Accelerator Launch Options

POST /uiservice/app/v0/launch/#/worklist

The worklist contains the user's actionable tasks. The user may select a task and trigger the Accelerator to go to the appropriate view for that task. Tasks related to ePA are assigned to the recipient of the ePA Message, e.g. the SPI of the provider sent in the PAInitiationRequest.

The user viewing and working on a task does not have to be a prescriber, but the user must have the permissions for the task as described in **Authentication and Authorization**.

POST /uiservice/app/v0/launch/#/process

When posting 'process', this is the task history and shows the status of the ePA and its associated tasks. This endpoint launches a view showing a list of workflow processes matching the search filter criteria. The user may drill down into each process to see the state of all associated tasks.

POST /uiservice/app/v0/launch/#/process/[InitiatorsProcessId]

When sending an InitiatorsProcessID, ensure that the ID equates to the PAReferenceID in the PAInitiationRequest. This value is assigned by the provider vendor system and is associated with all subsequent messages. This endpoint launches a view that shows the state of the workflow process including the status of the process and context specific details.

POST /uiservice/app/v0/launch/#/worklist/[taskId]

The taskId is a Surescripts defined value that is assigned to a specific step within an ePA process. A new taskId is generated when the status of a task is updated or a new task type (represented by taskName) is created within an ePA process. This endpoint will launch a user to the task-specific view within the process.

If a task's most recent status is 'Complete', the user should not be allowed to launch to that specific taskId, as launching to a completed task will result in an error.

Notes:

- Launching by taskId should only be done when the status of a task is awaiting action from the user who initiated the PA (or their proxy). TaskNames that may require action on part of the prescriber (or their proxy) include: "Complete PA Form" and "Acknowledge" tasks (for example, "Acknowledge Closed PA").
  The task status is relative to the user (i.e., a prescriber).
- To obtain taskId for purposes of launching Accelerator by taskId, use "Retrieve Process Detail" API endpoints. See **Retrieve Process Detail** for more details.

Accelerator Configuration Parameter

In addition to the basic styling and launch options currently available for Accelerator customers, Surescripts offers additional ePA-specific configuration settings that may be enabled at a global level (applicable for all customers of the Accelerator-enabled provider vendor). Please work with a Surescripts resource for additional details on which settings are available, how they function in practice, and how to enable them.

Auto-Acknowledge "PA Not Needed" Responses

By default, Surescripts automatically acknowledges a PAInititiationResponse with a ReasonCode of “CC” (PA not required for patient and/or medication). Please note that the auto-acknowledge option is enabled by default and can be turned off by a Surescripts resource. These responses would still be available for review via the Task History page, or via the data that is returned from Accelerator API requests, but would no longer appear on a user’s Worklist for acknowledgment. Provider vendors should ensure that they are communicating this process clearly to their end-users to prevent confusion for tasks that may appear to never receive a response.

> If the provider vendor is holding prescriptions from being sent to the pharmacy, and ‘auto-acknowledge’ is enabled, they should make sure that those prescriptions are being released to the pharmacy after they have been acknowledged. It should also be made clear to the end-user that the prescription was released, and no further action is required on their part.

8. Using Accelerator Web Service API

The Web Service API allows access to selected Accelerator functionality via RESTful web services for customers who prefer to develop their own user interface.

The Web Service API does not enable all Accelerator functionality. Provider vendors must implement the Web Application API to allow users to act on a task.

> **Note:** To reduce unnecessary strain on the Surescripts network, it is recommended that provider vendors implement controls on how often ePA tasks or processes are polled for status and/or details. Possible controls include (but are not limited to):
>
> - Only querying APIs upon user access to worklists that contain ePA references
> Querying APIs for info only when a user requests a specific ePA task
> - Please consult your Surescripts resource for questions regarding appropriate API usage conventions.

General Conventions

The Web Service API follows Representational State Transfer (REST) architecture principles using HTTP requests with appropriate HTTP verbs and status codes.

The message body for requests and responses are of type application/json.

Versioning and URLs

URLs are of the form https://[host]/uiservice/api/[version]/[path], where:

- The host depends on the environment, and is provided as part of the Surescripts integration processes.
- The version is the requested version of the endpoint, and is provided as part of the Surescripts integration processes.
- The path points to the specific service.

The API version number ("v0" in the above example) is incremented when a breaking change is introduced. A breaking change prevents an existing implementation from communicating with a particular version. Adding additional fields is not considered a breaking change. An API client application needs to support additional fields being added to an existing version.

For example: https://uiservice-api-staging.surescripts.net/uiservice/api/v0/processes

Examples of breaking changes includes, but are not limited to:

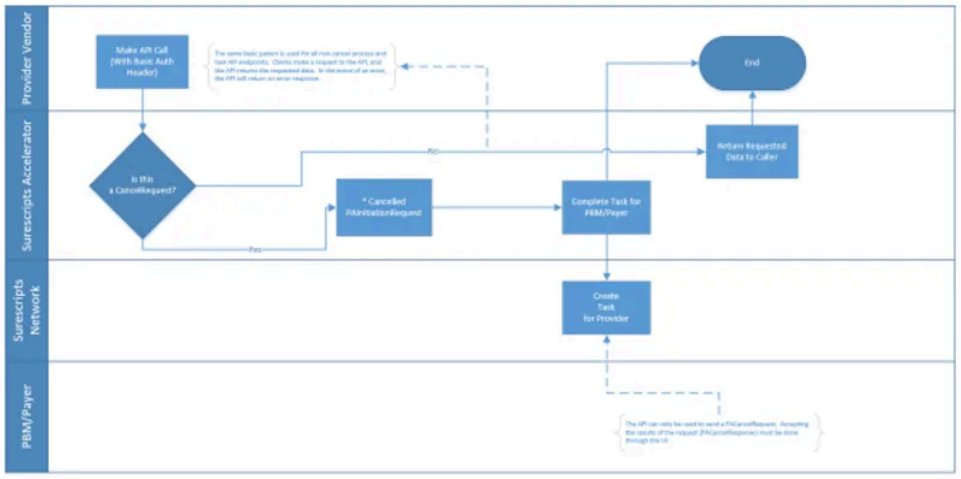
- Changing type or attributes of an existing field
- Changing an endpoint path
- Changing the semantic meaning of an existing field

Summary of Web Service API Endpoints

The following provides a summary of the Web Service API endpoints and the data that they provide.

API Endpoint	Description
[GET] /processes?patientId={{patientId}}&facility={{facility}}&createdFromDate={{createdFromDate}}&createdThruDate={{createdThruDate}}&pageIndex={{pageIndex}}&pageSize={{pageSize}}	Retrieves a paged list of processes for a given patient. The process details include the relevant data from the ePA messages for the current state in the workflow.
[POST] /processes	Retrieves a paged list of processes for specified initiator(s).
[POST] /initiators/processes	Retrieves a paged list of processes for the given list of initiatorsProcessId(s). **Note**: The initiatorsProcessId is the PAReferenceID as defined by the provider vendor system.
[GET] /processes/{{processId}}	Retrieve the process details for a specified processId.
[GET] /initiators/processes/{{initiatorsProcessId}}	Retrieve the process details for a specified initiatorsProcessId.
[GET] /worklist?patientId={{patientId}}&facility={{facility}}&user={{user}}&pageIndex={{pageIndex}}&pageSize={{pageSize}}	Retrieves a paged list of the active tasks for a given patient.
[GET] uiservice/api/v0/worklist/summary?patientId={patientId]&facility={{facility}}	Retrieves the count of worklist tasks for the patient.
[PUT] /cancel/processes/{{processId}}	Initiates cancel for a process based on the processesId.
[PUT] /cancel/initiators/processes/{{initiatorsProcessId}}	Initiates cancel for a process based on the initiatorsProcessId.
[POST] /uiservice/api/v0/complete/tasks	Completes a list of tasks for the given task ids. Returns list of task ids and any errors.
[POST] /uiservice/api/v0/complete/recipients/tasks	Completes all possible tasks for the given task recipient. **Note**: Only tasks that have been configured for this option will complete. Returns a list of task ids that have been completed.

API Auth Token Request and API End Point Call Process Flow



Notes:

- For the Cancelled PAInitiationRequest Process Flow, see **Cancelled PAInitiationRequest Process Flow**.
- API calls, with the exception of the Cancel API call, are read-only queries for processes and tasks. The Cancel API call sends a PACancelRequest message to PBM/payer(s).

User Security

Web Service API messages are sent using HTTPS and routed through an allowed vendor aggregation server.

API Note: Calls require the server IP addresses to be on the allowed list.

A call to a Web Service API requires the following headers:

- A standard basic authentication header using the credentials assigned during the integration process.

For example: Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==

- A custom X-SENDER-UID header that contains a Surescripts identifier assigned during the integration process. The format for the header value is qualifier-id.

For example: X-SENDER-UID: DP-EPA123

Request Headers

Header	Code	Description
Authorization	mandatory	A standard basic authentication header using the credentials assigned during the integration process. Example: Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ==
X-SENDER-UID	mandatory	A custom X-SENDER-UID header that contains a Surescripts identifier assigned during the integration process. The format for the header value is qualifier-id. Format for a system user id is: DP-[SystemUserId] Example: DP-EPA123

Security Headers in an API Call Example

> Security headers for a sample API call are shown in blue below.

```
POST //uiservice/api/v0/worklist HTTP/1.1
Host: localhost
Authorization: Basic dXNlcjE6cGFzc3dvcmQx
X-SENDER-UID: DP-EPA123DP-ABC123NP-ABC123DP-SPO123
Content-Type: application/json
Cache-Control: no-cache
Postman-Token: f920dfc3-bc53-db39-fc4d-b365f8cfb44c
{
 User: "rob",
  "recipients": [
    {
      "id": "6689026196001",
      "qualifier": "DZZ",
      "name": "Mouse, Micky"
    }
  ]
}
```

List Processes

The following endpoints are available to retrieve a list of processes based on parameters and search criteria. Each item in the list is a summary of a process. The responses are the same for all List Processes endpoints.

Created From Date and Created Thru Date Details

- If createdFromDate and createdThruDate range is within 18 months, all dates are returned.
- If createdFromDate and createdThruDate range exceeds 18 months, thru date less 18 months are returned.
- If createdFromDate and createdThruDate = null, current date less 18 months are returned.
- If createdFromDate is populated and createdThruDate = null, current date less 18 months are returned.
- If createdFromDate = null and createdThruDate date is populated, current date less 18 months are returned.

List Processes for PatientId

List Processes for patientId retrieves a paged list of process summaries for a given patient ID. If the optional parameter of Facility is not provided in the request, then only the processes that are not affiliated with a facility will be included in the results.

Request Template

[GET] {{host}}/api/v0/processes?patientId={{patientId}}&facility={{facility}} &fromDate={{fromDate}}&createdThruDate={{createdThruDate}}&pageIndex={{pageIndex}}&pageSize={{pageSize}}

Request Parameters

Parameter	Data Type	Code	Description
patientId	an..35	mandatory	Matches the Patient ID specified in the PAInitiationRequest/Body/Patient/Identification/MedicalRecordIdentificationNumberEHR.
facility	an..35	conditional	Qualifier the vendor may use to make their Patient ID unique across their installations. Facility is only used in conjunction with patientId. If Facility is not specified, only records that have no Facility associated with them will be returned. This field matches the Facility specified in the PAInitiationRequest/Body/Patient/Identification/MutuallyDefined in the NCPDP XML Schema.
createdFromDate	DateTime	conditional	Processes created on or after this date will be returned. See **Created From Date and Created Thru Date Details** for additional information on which dates will be returned. **Note:** DateTime format is CCYY-MM-DDTHH:MM:SSZ
createdThruDate	DateTime	conditional	Processes created on or prior to this date will be returned. See **Created From Date and Created Thru Date Details** for additional information on which dates will be returned. **Note:** DateTime format is CCYY-MM-DDTHH:MM:SSZ
pageSize	Positive Int	conditional	Integer representing the number of records returned in each request. If a pageSize is not specified, the default value of 25 is applied. This value is truncated to 100 if larger.
pageIndex	Positive Int	conditional	Integer representing the page index of data to be returned. If a pageIndex is not specified, the default value of 0 is applied.

Request Example

[GET] {{host}}/uiservice/api/v0/processes?patientId=george123

List Processes for Initiator(s)

List Processes for Initiator(s) retrieves a paged list of process summaries for the specified initiator(s). An initiator is an individual or physician that requested the ePA.

Request Template

[POST] {{host}}/processes

Request Parameters

Parameter	Data Type	Code	Description
initiators	Complex Type	mandatory	A collection of initiators. An initiator is a list containing the initiators identifying data.
id	an13	mandatory	SPI of the initiator. (Limit of 85 SPIs per request.)
qualifier	an1	mandatory	SPI's corresponding qualifier. Value = D
createdFromDate	DateTime	conditional	Processes created on or after this date will be returned. See **Created From Date and Created Thru Date Details** for additional information on which dates will be returned. **Note:** DateTime format is CCYY-MM-DDTHH:MM:SSZ
createdThruDate	DateTime	conditional	Processes created on or prior to this date will be returned. See **Created From Date and Created Thru Date Details** for additional information on which dates will be returned. **Note:** DateTime format is CCYY-MM-DDTHH:MM:SSZ
pageSize	Positive Int	conditional	Integer representing the number of records returned in each request. If a pageSize is not specified, the default value of 25 is applied. This value is truncated to 100 if larger.
pageIndex	Positive Int	conditional	Integer representing the page index of data to be returned. If a pageIndex is not specified, the default value of 0 is applied.

Request Example

[POST] http://[host]/uiservice/api/v0/processes

```
{
  "initiators": [
    {
      "id": "6223033492001",
      "qualifier": "D"
    }
  ],
  "createdFromDate": "2014-12-05T15🕔42.578706-06:00",
  "createdThruDate": "2014-12-05T15🕔42.578706-06:00",
  "pageIndex": 2,
  "pageSize": 3
}
```

List Processes for InitiatorsProcessId(s)

List Processes for initiatorsProcessId(s) retrieves a paged list of processes for the specified initiatorsProcessId(s). InitiatorsProcessId is the PAReferenceID as defined by the provider vendor system.

Request Template

[POST] {{host}}/processes

Request Parameters

Parameter	Data Type	Code	Description
initiatorProcessIds	Complex Type	mandatory	A collection of initiators process IDs.
createdFromDate	DateTime	conditional	Processes created on or after this date will be returned. See **Created From Date and Created Thru Date Details** for additional information on which dates will be returned. **Note:** DateTime format is CCYY-MM-DDTHH:MM:SSZ
createdThruDate	DateTime	conditional	Processes created on or thru this date will be returned. See **Created From Date and Created Thru Date Details** for additional information on which dates will be returned. **Note:** DateTime format is CCYY-MM-DDTHH:MM:SSZ
pageSize	Positive Int	conditional	Integer representing the number of records returned in each request. If a pageSize is not specified, the default value of 25 is applied. This value is truncated to 100 if larger.
pageIndex	Positive Int	conditional	Integer representing the page index of data to be returned. If a pageIndex is not specified, the default value of 0 is applied.

Request Example

[POST] http://[host]/uiservice/api/v0/intiators/processes

```
{
  "initiatorProcessIds": [
    "f050fc8601444af1be3c5ffeb4c1d703",
    "3fb345ccf3814adc835f756e752c851a"
  ]
}
```

List Processes Response

Response HttpStatus Codes

HttpStatus	Description
200	Successful response along with the processList and the corresponding process response fields
400	Bad Request

Response Field Description – List Processes

Field	Description
pageIndex	Integer representing the returned page index
pageSize	Integer representing the page size
totalRows	Integer representing the total number of rows matching the search criteria
items	Collection of process summary objects. See **Response Field Description – Process Summary**

Response Field Description – Process Summary

Field	Description
processId	Unique Surescripts ID which represents the process instance.
processName	Name of the process.
initiatorsProcessId	The initiatorsProcessId is the PAReferenceID as defined by the provider vendor system.
initiator	List containing the recipient's identifying data. Field(s) include: id qualifier name **Note:** Supported qualifiers are D and PY. If the qualifier is D, then the ID will be the SPI. If the qualifier is PY, the ID will be the PBM/payer ID.
patientId	Vendor supplied Patient ID.
facility	Optional vendor supplied value to create uniqueness across vendor installations.
patientFirstName	Patient's first name.
patientLastName	Patient's last name.
patientDob	Patient's date of birth.
createdDate	Date/time process was created.
completedDate	Date/time process was completed.
status	Integer representing the status of the process, possible values are: ~~1: Draft~~ 2: In Progress 3: Complete 4: Cancelled 5: Errored **Note**: Values with a strikethrough are not used for ePA.
statusName	Textual name representing the status of the process.
description	Description of the process instance.

Response Example


```
{
 "pageIndex": 0,
 "pageSize": 25,
 "totalRows": 1,
 "items": [
  {
   "processId": 122908,
```

```
    "processName": "ePA Case",
    "initiatorsProcessId": "06056ea227e144cd820cfc0f6e860d1f",
    "initiator": {
      "id": "6590604289001",
      "qualifier": "D",
      "name": "Mouse, Mickey"
    },
    "patientId": "george",
    "facility": "clinic1",
    "patientFirstName": "George",
    "patientLastName": "Jetson",
    "patientDob": "1962-11-23T00:00:00",
    "createdDate": "2014-12-05T17:36:38.476",
    "completedDate": "2014-12-05T17:37:45.757",
    "status": 4,
    "statusName": "Cancelled",
    "description": "Botox (onabotuilinumatoxina) 20mg vial"
  }
 ]
}
```


Retrieve Process Detail

To retrieve process detail, use the Surescripts processId or the initiatorsProcessId per the information below.

Retrieve Process Detail Using the Surescripts ProcessId

Request Template

[GET] {{host}}/api/v0/processes/{{processId}}

Request Parameters

Parameter	Data Type	Code	Description
processId	Long	mandatory	Unique Surescripts ID which represents the process instance.

Request Example

[GET] {{host}}/uiservice/api/v0/processes/123

Retrieve Process by InitiatorsProcessId

Request Template

[GET] {{host}}/api/v0/initiators/processes/{{initiatorsProcessId}}

Request Parameters

Parameter	Data Type	Code	Description
initiatorsProcessId	String Max 35	mandatory	The initiatorsProcessId is the PAReferenceID as defined by the provider vendor system.

Request Example

[GET] {{host}}/uiservice/api/v0/processes/123p

Retrieve Process Response

> All Process Detail endpoints return the same response.

Response HttpStatus Codes

HttpStatus	Description
200	Successful response along with the Process response fields
400	Bad Request

Response Field Description – Process Detail

Process Summary fields are included in the Process Detail response. See **Response Field Description – Process Summary** table for more information.

Field	Description
tasks	Collection that contains details of each task in the requested process. See **Response Field Description – Task Detail for Process** table for more information.
metadata	List containing the response data corresponding to the state of the process. Field(s) include: PACaseId Response ReferenceNumber Approved Denied Open Closed See the NCPDP SCRIPT Standard Implementation Guide version 2023011 for field definitions. See tables below for possible status response details: **Response Status – "Approved" Response Field Description** **Response Status – "Denied" Response Field Description** **Response Status – "Open" Response Field Description** **Response Status – "Closed" Response Field Description**

Response Field Description – Task Detail for Process

Field	Description
taskId	Unique Surescripts ID which represents the task instance
taskName	Name of the task
recipient	List containing the recipient's identifying data. Field(s) include: id qualifier name **Note:** Supported qualifiers are D and PY. If the qualifier is D, then the ID will be the SPI. If the qualifier is PY, the ID will be the PBM/payer ID.
acceptedBy	Dataset containing the identifying data for the user who acted upon the task. Field(s) include: • id • qualifier • name **Note**: Supported qualifiers are D, PY, and Userid. If the qualifier is D, then the ID will be the SPI. If the qualifier is PY, the ID will be the PBM/payer ID. If the qualifier is Userid, the ID will be the End User ID assigned by the provider vendor system.
dueDate	Date/time task is required to be completed by.
acceptedDate	Date/time task was accepted to be worked on.
responseDate	Date/time task was responded to.
status	Integer representing the status of the task: Task values are: 1: Draft 2: Pending 3: Assigned 4: Distributed 5: Accepted 6: Released 7: Complete 8: Cancelled 9: Errored **Note**: Values with a strikethrough are not used for ePA
statusName	Textual name representing the status of the task.

Response Status – "Approved" Response Field Description

Field	Description
paCaseId	The unique case ID assigned by the PBM/payer in the PAInitiationResponse.
authorizationNumber	This optional element contains a payer-assigned authorization number.
authorizationPeriod	This optional list provides the authorization start and end dates. Field(s) include: EffectiveDate ExpirationDate See the NCPDP SCRIPT Standard Implementation Guide version 2023011 for field definitions.
appeal	This mandatory list indicates whether this determination can be further appealed. Field(s) include: IsEAppealSupported ExpirationDate PANote See the NCPDP SCRIPT Standard Implementation Guide version 2023011 for field definitions.
paNote	Free text note provides additional information regarding the response status.

Response Status – "Denied" Response Field Description

Field	Description
paCaseId	The unique case ID assigned by the PBM/payer in the PAInitiationResponse.
appeal	The Appeal list details corresponding to the corresponding process. Field(s) include: IsEAppealSupported ExpirationDate DateTime PANote **Note**: DateTime format is CCYY-MM-DDTHH:MM:SSZ See the NCPDP SCRIPT Standard Implementation Guide version 2023011 for field definitions.
paNote	Free text note provides additional information regarding the response status.

Response Status – "Open" Response Field Description

Field	Description
paCaseId	The unique case ID assigned by the PBM/payer in the PAInitationResponse.
deadlineForReply	List containing the deadline for reply criteria. Field(s) include: DateTime **Note**: DateTime format is CCYY-MM-DDTHH:MM:SSZ See the NCPDP SCRIPT Standard Implementation Guide version 2023011 for field definitions.
paNote	Free text note provides additional information regarding the response status.
appealCaseId	Only applies to an appeal. Unique appeal case ID attached to the open response of which the appeal it is associated.

Response Status – "Closed" Response Field Description

Field	Description
paCaseId	The unique case ID assigned by the PBM/payer in the PAInitiationResponse.
reasons	List of reasons the ePA has been closed. Field(s) include: Code Description See the NCPDP SCRIPT Standard Implementation Guide version 2023011 for field definitions.
authorizationNumber	This optional element contains a payer-assigned authorization number.
authorizationDetail	This optional list contains authorization details of the approval granted. Either authorizationDetail or authorizationPeriod will be sent. All elements in the composite are optional and used if they apply to the case. Field(s) include: PharmacyType Quantity Value CodeListQualifier QuantityUnitOfMeasure Code DaysSupply NumberOfCycles NumberOfRefills PANote See the NCPDP SCRIPT Standard Implementation Guide version 2023011 for field definitions.
authorizationPeriod	This optional list provides the authorization start and end dates. Either authorizationDetail or authorizationPeriod will be sent. Field(s) include: EffectiveDate ExpirationDate See the NCPDP SCRIPT Standard Implementation Guide version 2023011 for field definitions.
appealCaseId	Only applies to an appeal. If a unique appeal case ID was provided in the initial open PAAppealResponse, it will be provided in every subsequent appeal message – including the closed response.
paNote	Free text note provides additional information regarding the response status.

Response Examples

Example Response – Last Task is Associated to a Pending PBM/Payer Response

JSON


```
{
  "tasks": [
    {
      "taskId": 673672,
      "taskName": "Initiate PA",
      "status": 4,
      "statusName": "Distributed",
      "recipient": {
        "id": "T00000090999022",
        "qualifier": "PY",
        "name": "accl_epa_dev_payer_shawn"
      }
    }
  ],
  "processId": 123188,
  "processName": "ePA Case",
  "initiatorsProcessId": "f63c307e139b49f3b7d7b4c2cd22b14f",
  "initiator": {
    "id": "6223033492001",
    "qualifier": "D",
    "name": "Mouse, Mickey"
  },
  "patientId": "george",
  "facility": "clinic1",
  "patientFirstName": "George",
  "patientLastName": "Jetson",
  "patientDob": "1962-11-23T00:00:00",
  "createdDate": "2014-12-09T20:28:12.417",
  "completedDate": "0001-01-01T00:00:00",
  "status": 2,
  "statusName": "In Progress",
  "description": "Botox (onabotuilinumatoxina) 20mg vial"
}
```


Example Open Response

JSON

```
{
  "metadata": {
    "paCaseId": "665616",
    "response": {
      "referenceNumber": "AN35xxxxx",
      "open": {
        "paCaseId": "665616",
        "deadlineForReply": {
          "dateTime": "2013-08-13T09:30:47Z"
        },
        "paNote": "PANote - the generic medicine must be tried before the name brand is approved."
      }
    }
  },
  "tasks": [
    {
      "taskId": 673714,
      "taskName": "Initiate PA",
      "status": 4,
      "statusName": "Distributed",
      "recipient": {
        "id": "T00000090999022",
        "qualifier": "PY",
        "name": "accl_epa_dev_payer_shawn"
      }
    },
    {
      "taskId": 673716,
      "taskName": "Initiate PA",
      "acceptedDate": "2014-12-09T20:30:28.6",
      "responseDate": "2014-12-09T20:30:28",
      "status": 7,
      "statusName": "Complete",
      "acceptedBy": {
        "id": "T00000090999022"
        "qualifier": "PY"
      },
      "recipient": {
        "id": "T00000090999022",
        "qualifier": "PY",
        "name": "accl_epa_dev_payer_shawn"
      }
    },
    {
      "taskId": 673718,
      "taskName": "Complete PA Form",
      "dueDate": "2013-08-13T09:30:47",
      "status": 4,
      "statusName": "Distributed",
      "recipient": {
        "id": "6223033492001",
        "qualifier": "D",
        "name": "Mouse, Mickey"
      }
    }
  ],
```

```
  "processId": 123190,
  "processName": "ePA Case",
  "initiatorsProcessId": "66561645fcaa4dd88c40cc1c3df9d3e3",
  "initiator": {
    "id": "6223033492001",
    "qualifier": "D",
    "name": "Mouse, Mickey"
  },
  "patientId": "george",
  "facility": "clinic1",
  "patientFirstName": "George",
  "patientLastName": "Jetson",
  "patientDob": "1962-11-23T00:00:00",
  "createdDate": "2014-12-09T20:30:25.58",
  "completedDate": "0001-01-01T00:00:00",
  "status": 2,
  "statusName": "In Progress",
  "description": "Botox (onabotuilinumatoxina) 20mg vial"
}
```


Example Closed Response

JSON

```
{
  "metadata": {
    "paCaseId": "ed717e",
    "response": {
      "referenceNumber": "AN35xxxxx",
      "closed": {
        "paCaseId": "ed717e",
        "reasons": [
          {
            "code": "CC",
            "description": "Prior Authorization not required for patient/medication"
          }
        ],
        "authorizationDetail": [
          {
            "pharmacyType": "retail",
            "daysSupply": "12",
            "numberOfRefills": "3"
          }
        ],
        "authorizationPeriod": {
          "effectiveDate": {
            "dateTime": "2013-10-02T00:00:00"
          },
          "expirationDate": {
            "dateTime": "2013-10-22T00:00:00"
          }
        },
        "paNote": "This medication does not require a PA"
      }
    }
  },
  "tasks": [
    {
      "taskId": 673720,
      "taskName": "Initiate PA",
      "status": 4,
      "statusName": "Distributed",
      "recipient": {
        "id": "T00000090999022",
        "qualifier": "PY",
        "name": "accl_epa_dev_payer_shawn"
      }
    },
    {
      "taskId": 673722,
      "taskName": "Initiate PA",
      "acceptedDate": "2014-12-09T20:32:46.841",
      "responseDate": "2014-12-09T20:32:46",
      "status": 7,
      "statusName": "Complete",
      "acceptedBy": {
        "id": "T00000090999022"
        "qualifier": "PY"
      },
      "recipient": {
```

```
          "id": "T00000090999022",
          "qualifier": "PY",
          "name": "accl_epa_dev_payer_shawn"
        }
      },
      {
        "taskId": 673724,
        "taskName": "Acknowledge PA Not Needed",
        "status": 4,
        "statusName": "Distributed",
        "recipient": {
          "id": "6223033492001",
          "qualifier": "D",
          "name": "Mouse, Mickey"
        }
      }
    ],
    "processId": 123192,
    "processName": "ePA Case",
    "initiatorsProcessId": "ed717eea2cdc4ec483f6af8811864265",
    "initiator": {
      "id": "6223033492001",
      "qualifier": "D",
      "name": "Mouse, Mickey"
    },
    "patientId": "george",
    "facility": "clinic1",
    "patientFirstName": "George",
    "patientLastName": "Jetson",
    "patientDob": "1962-11-23T00:00:00",
    "createdDate": "2014-12-09T20:32:45.496",
    "completedDate": "0001-01-01T00:00:00",
    "status": 2,
    "statusName": "In Progress",
    "description": "Botox (onabotuilinumatoxina) 20mg vial"
  }
```


Example Approved Response

JSON

```
{
  "metadata": {
    "paCaseId": "0f038d",
    "response": {
      "approved": {
        "paCaseId": "0f038d",
        "authorizationNumber": "111222333",
        "authorizationPeriod": [
          {
            "effectiveDate": {
              "dateTime": "2014-12-09T00:00:00"
            },
            "expirationDate": {
              "dateTime": "2015-12-09T00:00:00"
            }
          }
        ],
        "appeal": {
          "isEAppealSupported": "Y",
          "expirationDate": {
            "dateTime": "2014-12-14T00:00:00"
          }
        },
        "paNote": "This is an approved PA Note."
      }
    }
  },
  "tasks": [
    {
      "taskId": 673778,
      "taskName": "Initiate PA",
      "status": 4,
      "statusName": "Distributed",
      "recipient": {
        "id": "T00000090999022",
        "qualifier": "PY",
        "name": "accl_epa_dev_payer_shawn"
      }
    },
    {
      "taskId": 673780,
      "taskName": "Initiate PA",
      "acceptedDate": "2014-12-09T21:14:26.985",
      "responseDate": "2014-12-09T21:14:26",
      "status": 7,
      "statusName": "Complete",
      "acceptedBy": {
        "id": "T00000090999022",
        "qualifier": "PY"
      },
      "recipient": {
        "id": "T00000090999022",
        "qualifier": "PY",
        "name": "accl_epa_dev_payer_shawn"
      }
    },
```

```
    {
      "taskId": 673782,
      "taskName": "Complete PA Form",
      "dueDate": "2013-08-13T09:30:47",
      "status": 4,
      "statusName": "Distributed",
      "recipient": {
        "id": "6223033492001",
        "qualifier": "D",
        "name": "Mouse, Mickey"
      }
    },
    {
      "taskId": 673784,
      "taskName": "Complete PA Form",
      "dueDate": "2013-08-13T09:30:47",
      "acceptedDate": "2014-12-09T21:14:35.563",
      "status": 5,
      "statusName": "Accepted",
      "acceptedBy": {
        "id": "shawn",
        "qualifier": "Userid"
      },
      "recipient": {
        "id": "6223033492001",
        "qualifier": "D",
        "name": "Mouse, Mickey"
      }
    },
    {
      "taskId": 673786,
      "taskName": "Complete PA Form",
      "dueDate": "2013-08-13T09:30:47",
      "acceptedDate": "2014-12-09T21:14:35.563",
      "responseDate": "2014-12-09T21:14:43",
      "status": 7,
      "statusName": "Complete",
      "acceptedBy": {
        "qualifier": "Userid",
        "id": "shawn"
      },
      "recipient": {
        "id": "6223033492001",
        "qualifier": "D",
        "name": "Mouse, Mickey"
      }
    },
    {
      "taskId": 673788,
      "taskName": "Send PA Request",
      "status": 4,
      "statusName": "Distributed",
      "recipient": {
        "id": "System Generated",
        "qualifier": "Userid",
        "name": "System Generated"
      }
    },
    {
```

```
      "taskId": 673790,
      "taskName": "Send PA Request",
      "acceptedDate": "2014-12-09T21:14:43.629",
      "responseDate": "2014-12-09T21:14:43",
      "status": 7,
      "statusName": "Complete",
      "acceptedBy": {
        "id": "6223033492001",
        "qualifier": "D"
      },
      "recipient": {
        "id": "System Generated",
        "qualifier": "Userid",
        "name": "System Generated"
      }
    },
    {
      "taskId": 673792,
      "taskName": "PBM Response",
      "status": 4,
      "statusName": "Distributed",
      "recipient": {
        "id": "T00000090999022",
        "qualifier": "PY",
        "name": "accl_epa_dev_payer_shawn"
      }
    },
    {
      "taskId": 673794,
      "taskName": "PBM Response",
      "acceptedDate": "2014-12-09T21:14:46.626",
      "responseDate": "2014-12-09T21:14:46",
      "status": 7,
      "statusName": "Complete",
      "acceptedBy": {
        "id": "T00000090999022",
        "qualifier": "PY"
      },
      "recipient": {
        "id": "T00000090999022",
        "qualifier": "PY",
        "name": "accl_epa_dev_payer_shawn"
      }
    },
    {
      "taskId": 673796,
      "taskName": "Acknowledge Approval",
      "status": 4,
      "statusName": "Distributed",
      "recipient": {
        "id": "6223033492001",
        "qualifier": "D",
        "name": "Mouse, Mickey"
      }
    }
  ],
  "processId": 123200,
  "processName": "ePA Case",
  "initiatorsProcessId": "0f038d30ca2b404d873c180543041da1",
```

```
  "initiator": {
    "id": "6223033492001",
    "qualifier": "D",
    "name": "Mouse, Mickey"
  },
  "patientId": "george",
  "facility": "clinic1",
  "patientFirstName": "George",
  "patientLastName": "Jetson",
  "patientDob": "1962-11-23T00:00:00",
  "createdDate": "2014-12-09T21:14:26.144",
  "completedDate": "0001-01-01T00:00:00",
  "status": 2,
  "statusName": "In Progress",
  "description": "Botox (onabotuilinumatoxina) 20mg vial"
```


Example Denied Response

JSON

```
{
  "metadata": {
    "paCaseId": "2b2d9b",
    "response": {
      "denied": {
        "paCaseId": "2b2d9b",
        "appeal": {
          "iseAppealSupported": "Y",
          "expirationDate": {
            "dateTime": "2014-12-14T00:00:00"
          },
          "paNote": "Appeal Note"
        },
        "paNote": "This is an denied PA Note."
      }
    }
  },
  "tasks": [
    {
      "taskId": 673798,
      "taskName": "Initiate PA",
      "status": 4,
      "statusName": "Distributed",
      "recipient": {
        "id": "T00000090999022",
        "qualifier": "PY",
        "name": "accl_epa_dev_payer_shawn"
      }
    },
    {
      "taskId": 673800,
      "taskName": "Initiate PA",
      "acceptedDate": "2014-12-09T21:15:35.956",
      "responseDate": "2014-12-09T21:15:35",
      "status": 7,
      "statusName": "Complete",
      "acceptedBy": {
        "id": "T00000090999022"
        "qualifier": "PY",
      },
      "recipient": {
        "id": "T00000090999022",
        "qualifier": "PY",
        "name": "accl_epa_dev_payer_shawn"
      }
    },
    {
      "taskId": 673802,
      "taskName": "Complete PA Form",
      "dueDate": "2013-08-13T09:30:47",
      "status": 4,
      "statusName": "Distributed",
      "recipient": {
        "id": "6223033492001",
        "qualifier": "D",
        "name": "Mouse, Mickey"
```

```
    }
  },
  {
    "taskId": 673804,
    "taskName": "Complete PA Form",
    "dueDate": "2013-08-13T09:30:47",
    "acceptedDate": "2014-12-09T21:15:45.707",
    "status": 5,
    "statusName": "Accepted",
    "acceptedBy": {
      "id": "shawn"
      "qualifier": "Userid"
    },
    "recipient": {
      "id": "6223033492001",
      "qualifier": "D",
      "name": "Mouse, Mickey"
    }
  },
  {
    "taskId": 673806,
    "taskName": "Complete PA Form",
    "dueDate": "2013-08-13T09:30:47",
    "acceptedDate": "2014-12-09T21:15:45.707",
    "responseDate": "2014-12-09T21:15:52",
    "status": 7,
    "statusName": "Complete",
    "acceptedBy": {
      "id": "shawn"
      "qualifier": "Userid",
    },
    "recipient": {
      "id": "6223033492001",
      "qualifier": "D",
      "name": "Mouse, Mickey"
    }
  },
  {
    "taskId": 673808,
    "taskName": "Send PA Request",
    "status": 4,
    "statusName": "Distributed",
    "recipient": {
      "id": "System Generated",
      "qualifier": "Userid",
      "name": "System Generated"
    }
  },
  {
    "taskId": 673810,
    "taskName": "Send PA Request",
    "acceptedDate": "2014-12-09T21:15:54.07",
    "responseDate": "2014-12-09T21:15:54",
    "status": 7,
    "statusName": "Complete",
    "acceptedBy": {
      "id": "6223033492001"
      "qualifier": "D"
    },
```

```
        "recipient": {
          "id": "System Generated",
          "qualifier": "Userid",
          "name": "System Generated"
        }
      },
      {
        "taskId": 673812,
        "taskName": "PBM Response",
        "status": 4,
        "statusName": "Distributed",
        "recipient": {
          "id": "T00000090999022",
          "qualifier": "PY",
          "name": "accl_epa_dev_payer_shawn"
        }
      },
      {
        "taskId": 673814,
        "taskName": "PBM Response",
        "acceptedDate": "2014-12-09T21:15:56.298",
        "responseDate": "2014-12-09T21:15:56",
        "status": 7,
        "statusName": "Complete",
        "acceptedBy": {
          "id": "T00000090999022"
          "qualifier": "PY"
        },
        "recipient": {
          "id": "T00000090999022",
          "qualifier": "PY",
          "name": "accl_epa_dev_payer_shawn"
        }
      },
      {
        "taskId": 673816,
        "taskName": "Acknowledge Denied PA",
        "status": 4,
        "statusName": "Distributed",
        "recipient": {
          "id": "6223033492001",
          "qualifier": "D",
          "name": "Mouse, Mickey"
        }
      }
    ],
    "processId": 123202,
    "processName": "ePA Case",
    "initiatorsProcessId": "2b2d9ba639874df08474b38af8fdc9e9",
    "initiator": {
      "id": "6223033492001",
      "qualifier": "D",
      "name": "Mouse, Mickey"
    },
    "patientId": "george",
    "facility": "clinic1",
    "patientFirstName": "George",
    "patientLastName": "Jetson",
    "patientDob": "1962-11-23T00:00:00",
```

```
    "createdDate": "2014-12-09T21:15:33.966",
    "completedDate": "0001-01-01T00:00:00",
    "status": 2,
    "statusName": "In Progress",
    "description": "Botox (onabotuilinumatoxina) 20mg vial"
```


Retrieve Worklist

Retrieve worklist returns the list of active tasks awaiting action based on the given filter criteria. The items returned by a worklist request either have the acceptedBy as the user in the worklist request or the acceptedBy is null.

Retrieve Worklist by Patient

Retrieving a worklist by patient retrieves a paged list of the active tasks for a given patient.

Request Template Example

[GET] {{host}}/worklist?patientId={{patientId}}&facility={{facility}}&user={{user}}&pageIndex={{pageIndex}}&pageSize={{pageSize}}

Request Parameters

Parameter	Data Type	Code	Description
patientId	an..35	mandatory	Matches the patientId specified in the PAInitiationRequest/Body/Patient/Identification/MedicalRecordIdentificationNumberEHR.
user	an..250	mandatory	Unique provider vendor end user ID representing the user working the task list. This is the end user ID and does not have to be the user ID of the prescriber associated with the tasks.
facility	an..35	conditional	Qualifier the vendor may use to make their Patient ID unique across their installations. Facility is only used in conjunction with patientId. If Facility is not specified, only records that have no Facility associated with them will be returned. This field matches the Facility specified in the PAInitiationRequest/Body/Patient/Identification/MutuallyDefined in the NCPDP XML Schema.
pageSize	Positive Int	conditional	Integer representing the number of records returned in each request. If a pageSize is not specified, the default value of 25 is applied.
pageIndex	Positive Int	conditional	Integer representing the page index of data to be returned. If a pageIndex is not specified, the default value of 0 is applied.

[GET] {{host}}/uiservice/api/v0/worklist?patientId=george&facility=clinic1&user=jake

[GET] {{host}}/uiservice/api/v0/worklist?patientId=george&user=jake

Get Worklist by Recipients

Retrieves a paged list of the active tasks for a given collection of recipients (providers).

Request Template Example

[POST] {{host}}/worklist

Request JSON Fields

JSON Field	Data Type	Code	Description
user	an..250	mandatory	Unique provider vendor End User ID representing the user working the task list. This is the End User ID and does not have to be the user ID of the prescriber associated with the tasks.
recipients	List	mandatory	Collection of recipients for which Accelerator based tasks will be retrieved.
id	an13	mandatory	SPI of the initiator.
qualifier	an1	mandatory	SPI's corresponding qualifier. Qualifier = D
pageSize	Positive Int	conditional	Integer representing the number of records returned in each request. If a pageSize is not specified, the default value of 25 is applied. This value is truncated to 100 if larger.
pageIndex	Positive Int	conditional	Integer representing the page index of data to be returned. If a pageIndex is not specified, the default value of 0 is applied.

Request Example

[POST] https://[host]/uiservice/api/v0/worklist

```
{
  "user": "jake",
  "recipients": [
    {
      "id": "6417473720001"
      "qualifier": "DZZ",
    }
  ]
}
```

Retrieve Worklist Response

Response Codes and Fields

HttpStatus Code	Description
200	Successful response along with the processList and the corresponding process response fields
400	Invalid request

Description Message Body JSON Fields

Field	Description
items	Collection of task details matching the corresponding process See Description Message Body - Items JSON Fields table below.
pageIndex	Integer representing the returned page index
pageSize	Integer representing the page size
totalRows	Integer representing the total number of rows matching the search criteria

Description Message Body - Items JSON Fields

Field	Description
processId	Unique Surescripts ID which represents the process instance
processName	Name of the process
initiatorsProcessId	The initiatorsProcessId is the PAReferenceID as defined by the provider vendor system
taskId	Unique Surescripts ID which represents the task instance
taskName	Name of the task
patientId	Vendor supplied patient ID
facility	Vendor supplied value to create uniqueness across vendor installations
patientFirstName	Patient's first name
patientLastName	Patient's last name
patientDob	Patient's date of birth
initiator	List containing the recipients identifying data. Field(s) include: • ID • Qualifier • Name See the NCPDP SCRIPT Standard Implementation Guide version 2023011 for field definitions.
recipient	List containing the recipient's identifying data. Field(s) include: • ID • Qualifier • Name **Note:** Supported qualifiers are D and PY. If the qualifier is D, then the ID will be the SPI. If the qualifier is PY, the ID will be the PBM/payer ID. See the NCPDP SCRIPT Standard Implementation Guide version 2023011 for field definitions.
acceptedBy	Dataset containing the accepted by identifying data. Field(s) include: • ID • Qualifier • Name **Note:** Supported qualifiers are D, PY, and Userid. If the qualifier is D, then the ID will be the SPI. If the qualifier is PY, the ID will be the PBM/payer ID. Userid is the End User ID assigned by the provider vendor system. See the NCPDP SCRIPT Standard Implementation Guide version 2023011 for field definitions.

Field	Description
createdDate	Date/time task was created.
dueDate	Date/time task is required to be completed by.
acceptedDate	Date/time task was accepted to be worked on.
responseDate	Date/time task was responded to.
status	Integer representing the status of the task: Task values are: 1. Draft 2. Pending 3. Assigned 4. Distributed 5. Accepted 6. Released 7. Complete 8. Cancelled 9. Errored **Note**: Values with a strikethrough are not used for ePA
statusName	Textual name representing the status of the task.

Worklist Response Example


```
{
 "pageIndex": 0,
 "pageSize": 25,
 "totalRows": 2,
 "items": [
  {
   "processId": 123268,
   "processName": "ePA Case",
   "initiatorsProcessId": "a27332aae4a04340a512be900bf09ca0",
   "patientId": "elizabeth",
   "facility": "clinic2",
```

```
    "patientFirstName": "Elizabeth",
    "patientLastName": "Johanson-Anderson",
    "patientDob": "1967-08-13T00:00:00",
    "initiator": {
      "id": "6223033492001",
      "qualifier": "D",
      "name": "Mouse, Mickey"
    },
    "createdDate": "2014-12-10T21:47:59.582",
    "taskId": 674376,
    "taskName": "Complete PA Form",
    "dueDate": "2013-08-13T09:30:47",
    "status": 4,
    "statusName": "Distributed",
    "recipient": {
      "id": "6223033492001",
      "qualifier": "D",
      "name": "Mouse, Mickey"
    }
  },
  {
    "processId": 123266,
    "processName": "ePA Case",
    "initiatorsProcessId": "abd2dc86d5ab4ad09fe7ffa203411ee2",
    "patientId": "george",
    "facility": "clinic1",
    "patientFirstName": "George",
    "patientLastName": "Jetson",
    "patientDob": "1962-11-23T00:00:00",
    "initiator": {
      "id": "6223033492001",
```

```
      "qualifier": "D",
      "name": "Mouse, Mickey"
    },
    "createdDate": "2014-12-10T21:47:50.798",
    "taskId": 674370,
    "taskName": "Complete PA Form",
    "dueDate": "2013-08-13T09:30:47",
    "status": 4,
    "statusName": "Distributed",
    "recipient": {
      "id": "6223033492001",
      "qualifier": "D",
      "name": "Mouse, Mickey"
    }
  }
 ]
}
```


Cancel Process

Cancelling a process sends a PACancelRequest to the PBM/payer, which begins the process of cancelling an ePA case.

Cancel Process Using the ProcessId

Request Template

[PUT] {{host}}/cancel/processes/{{processId}}

Request Parameters

Parameter	Data Type	Description
processId	Long	mandatory Unique Surescripts ID which represents the process instance.

Request JSON Fields

JSON Field	Data Type	Code	Description
user	an..250	mandatory	ID of the end user logged into the provider vendor system.
cancelCode	an2	conditional	NCPDP code for the reason ePA is being cancelled. Values: AC = Prescriber is no longer responsible for patient's care BW = Change to a different medication BY = Other
isPatientNotified	an1	conditional	Boolean value signifying if the patient was notified. Values: Y N
note	an..2000	conditional	Free text note to PBM/payer regard the cancellation of an ePA.

Request Example

[PUT] https://[host]/uiservice/api/v0/cancel/processes/1234

```
{
   User: “Jake”,
   cancelCode: "BW",
   isPatientNotified: “N”,
   note: “Sample note”
}
```

Cancel Process Using InitiatorsProcessId

Use the InitiatorsProcessId to cancel a process.

Request Template

[PUT] {{host}}/cancel/initiators/processes/{{initiatorsProcessId}}

Request Parameters

Parameter	Data Type	Code	Description
initiatorsProcessId	an..255	mandatory	The initiatorsProcessId is the PAReferenceID as defined by the provider vendor.

Description of JSON Fields

JSON Field	Data Type	Code	Description
user	an..250	mandatory	ID of the end user logged into the provider vendor system.
cancelCode	an2	conditional	NCPDP code for the reason ePA is being cancelled. Values: AC = Prescriber is no longer responsible for patient's care BW = Change to a different medication BY = Other
isPatientNotified	an1	conditional	Boolean value signifying if the patient was notified. Values: Y N
note	an..2000	conditional	Free text note to PBM/payer regard the cancellation of an ePA.

Request Example

[PUT] https://[host]/uiservice/api/v0/cancel/initiators/processes/123p

```
{
   user: "Jake",
   cancelCode: "BW",
   isPatientNotified: "N",
   note: "Sample note"
}
```

Cancel a Process Response

Response Codes and Fields

HttpStatus Code	Description
200	Successful response
400	Invalid request

Worklist Summary

The api/v0/worklist/summary provides summary information of task counts for a provider or a patient. The API takes a list of recipients or a patient and a products object, which can include all or just one product. If product is not sent, the customer will get back the task counts for all products implemented.

Patient Request Template Example

[GET] {{host}}/uiservice/api/v0/worklist/summary?patientId={patientId]
&facility={{facility}}

Patient Request Parameters

Parameter	Data Type	Code	Description
patientId	an..35	mandatory	Matches the patientId specified in the PAInitiationRequest/Body/Patient/Identification/MedicalRecordIdentificationNumberEHR.
facility	an..35	conditional	Qualifier the vendor may use to make their Patient ID unique across their installations. Facility is only used in conjunction with patientId. If Facility is not specified, only records that have no Facility associated with them will be returned. This field matches the Facility specified in the PAInitiationRequest/Body/Patient/Identification/MutuallyDefined in the NCPDP XML Schema.
product	an..5	conditional	**Value**: epa Optional qualifier the vendor may use to filter the active tasks to only ePA when using the patientId.

Patient Request Example

[GET] {{host}}/uiservice/api/v0/worklist/summary?patientId=george&facility=clinic1

Provider Worklist Summary

[POST] {{host}}/uiservice/api/v0/worklist/summary

Provider Request JSON Fields

JSON Field	Data Type	Code	Description
recipients	List	mandatory	Collection of recipients for which Accelerator based tasks will be retrieved.
qualifier	an..1	mandatory	SPI's corresponding qualifier. **Value**: D
id	an..13	mandatory	SPI of the provider.
products	List	conditional	Required if customer needs a specific product. If not sent, all products will be returned. Example: epa
id	an..5	conditional	Product Id. At least one product Id is required if 'products' is used. **Value**: epa
processTypes	an..12	conditional	Required if process types filter is needed. Note: Process types are not currently used for Electronic Prior Authorization. **Value**: null

Request Example


```
{
  "recipients": [
    {
      "qualifier": "D",
      "id": "6598607334001"
```

```
      }
   ],
   "products": [
{
         "id": "epa",
          "processTypes": null
      }
   ]
}
```


Worklist Summary Response

HTTP Status Codes

Http Status Code	Description
200	Successful response
400	Invalid request

Description Message Body JSON Fields

Field	Description
product	Name of the product. **Value**: epa
low	Integer representing the count of active tasks with a low priority. **Note**: Currently ePA does not use priority types for messages.
medium	Integer representing the count of active tasks with a medium priority. **Note**: Currently ePA does not use priority types for messages.
high	Integer representing the count of active tasks with a high priority. **Note**: Currently ePA does not use priority types for messages.
urgent	Integer representing the count of active tasks with an urgent priority. **Note**: Currently ePA does not use priority types for messages.
alert	Integer representing the count of active tasks with an alert priority. **Note**: Currently ePA does not use priority types for messages.
total	Integer representing the total count of active tasks.

JSON Response Example


```
[
   {
```

```
    "product": "epa",
    "low": 0,
    "medium": 0,
    "high": 0,
    "urgent": 0,
    "alert": 0,
    "total": 6
  }
]
```


Complete Tasks

Completing a task sets the status of the task to complete and moves the process on to the next task, if there is one. In many cases this could mean also completing the process. Not all task types are allowed to be completed via this API because user interaction may be required to complete the task. The results of the complete API call will indicate which tasks were allowed to be completed via API.

Complete Tasks by TaskIds

Request Template

[POST] {{host}}/uiservice/api/v0/complete/tasks

Request JSON Fields

JSON Field	Data Type	Code	Description
taskIds	List<long>	mandatory	List the IDs of the tasks for completion.

Request Example

[POST] https://[host]/uiservice/api/v0/complete/tasks

[83756, 83757, 83759, 83786]

Complete Tasks by TaskIds Response

Response Codes and Fields

HttpStatus Code	Description
200	Successful response
400	Invalid request

Description Message Body JSON Fields

Collection of the following JSON items

Field	Description
TaskId	The taskId requested to be completed.
Error	Null if successful. Reason for non-completion if not successful.

Complete Response Example


```
{
  "taskId": 83756,
  "error": "Task 83756 has already been completed"
}
{
  "taskId": 83757,
  "error": "Task 83757 is not configured to be completed via an external API call."
}
{
  "taskId": 83759
}
{
  "taskId": 83786
}
```


Complete Tasks by RecipientId

Attempts to complete the tasks associated with the given user.

Request Template

[POST] {{host}}/uiservice/api/v0/complete/recipients/tasks

Description of JSON Fields

JSON Field	Data Type	Code	Description
recipients	List	mandatory	Collection of recipients for which Accelerator based tasks will attempt to be complete.
qualifier	an..1	mandatory	SPI's corresponding qualifier. Value: D
id	an..13	mandatory	SPI of the provider.
products	List	conditional	Required if request should be restricted to specific products. If not sent, Surescripts will attempt to complete tasks for all eligible products Example: epa
id	an..5	conditional	Product Id. At least one product Id is required if 'products' is used. Value: epa
processTypes	an..12	conditional	Required if process types filter is needed. **Note:** Process types are not currently used for Electronic Prior Authorization. Value: null

Request Example

[POST] https://[host]/uiservice/api/v0/complete/recipients/tasks

```
{
  "recipients": [
     {
        "qualifier": "D",
         "id": "6223034442001"
     }
     {
        "qualifier": "D",
         "id": "6223044592002"
     }
     ],
  "products": [
        {"id":"epa"}
        ]
}
```

Complete Tasks by Recipient Response

Response Codes and Fields

HttpStatus Code	Description
200	Successful response
400	Invalid request

Description Message Body JSON Fields

Field	Description
List<long>	The list of task ids that were successfully completed by this call.

9. Application Certification Requirements

> For customers using Electronic Prior Authorization Accelerator, the items marked with * are certified by Surescripts on your behalf.

Provider Vendor Requirements

PA.200: The application shall make available for display the ResponseStatus and each occurrence of a PANote from all response transactions.

> Multiple responses of the same type may be received.*

PA.201: When the PBM/payer provides a question set in the PAInitiationResponse, the application shall display*:

- QuestionSetContactCommunicationNumber
- DeadlineForReply
- AttachmentNotes (when attachment required)

PA.202: The application shall require all specified questions to be answered (not blank) and an order, prior to sending the PARequest. *

PA.203: In the event that a PA has been canceled by the initiator or closed by the PBM, and the prior authorization is still desired, a new PAInitiationRequest shall be sent. *

PA.204: When the prescriber and main prior authorization contact vary, the application shall utilize the Provider composite to supply the prior authorization contact name and PrimaryTelephone.

PA.205: Provider vendors shall not send a PAAppealRequest if the value of the IsEAppealSupported field is "false" in the PAResponse.*

PA.206: Prescribing vendors who are utilizing the Surescripts Eligibility 270/271 messages shall populate the first BenefitsCoordination element with the associated information returned in a Surescripts 271 message. If more than one active coverage is returned, only the single coverage information utilized to write the prescription is sent. The 271 message's ISA13 control number shall be populated in the BenefitsCoordination/InterchangeControlNumber field in all instances including scenarios where there is no active coverage.

PA.207: Either MedicationPrescribed/DaysSuppy or MedicationPrescribed/PADaysSupplyDosesPerDay shall be populated in the PAInitiationRequest when available.

PA.208: If the PAInitiationRequest includes a PAPriorityIndicator flag, the priority must be displayed to the end user. The value of "S" (Standard Priority) should be the default if the data element is sent in the transaction.

PA.209: The application shall make available for display the ExpirationDate to the prescriber for a Prior Authorization renewal request (unsolicited PAInitiationResponse).*

PA.210: If a PBM/payer requires an attachment in an answered question, the application shall not allow a request to be sent without the attachment composite being populated.*

PA.211: When both a brand and generic NDC exist for a medication, in the PAInitiationRequest provider vendors shall submit the drug description and NDC of the form of the medication they want to prescribe for the patient.

PA.214: The application shall support the ePA question set conditional logic as outlined in the NCPDP ePA standard.*

PA.219: The drug name shall be an item that is currently available for dispensing. If there are active NDCs available, an inactive/obsolete NDC shall not be used. Customers are expected to work with their drug compendia vendor to satisfy this requirement.

S.203: Sender demographic information, when included in the message, shall match what was registered by the customer in the Surescripts Directory. When a prescribing system utilizes the Surescripts Learning Directory service, this requirement is satisfied for learned locations.

S.208: Codified values within a message shall not conflict with the related textual concept.

Example 1:

Field Name	Proper Use	Improper Use
DrugDescription	Primariea 10mg/10ml liquid	
DrugCoded/Strength/StrengthValue	10	<empty>
DrugCoded/Strength/StrengthUnitOfMeasure/Code	C42576 (mg/ml)	
DrugCoded/Strength/StrengthForm/Code	C42953	

Example 2:

Field Name	Proper Use	Improper Use
DrugDescription	Amoxicillin 500 mg oral capsule	
DrugCoded/Strength/StrengthValue	500	250
DrugCoded/Strength/StrengthUnitOfMeasure/Code	C28253 (Milligram)	C28254 (Milliliter)
DrugCoded/Strength/StrengthForm/Code	C25158 (Capsule)	C42998 (Tablet)

Provider Vendor and PBM/Payer Requirements

PA.212: Payer and submitter systems shall be able to send and receive PDF attachments at a minimum.*

PA.213: Participant applications shall not allow users/systems to respond multiple times to a message the application has already sent. The only exceptions are: the INProcess PAResponse followed by an Approved/Denied PAResponse message; and PARequest/PAResponses when additional information is requested.

PA.220: Customers shall support the scenarios detailed in the Message Flow section of this guide.

S.200: Customers shall be able to receive syntactically valid maximum and minimum populated messages. Optional data elements (and values therein) shall not cause message failure.

S.201: Customers shall ensure all messages are syntactically correct before transmission to Surescripts.

S.205: The sender shall ensure the combination of the From and MessageID elements, for all messages including Error, is unique for at least 18 months. Surescripts recommends an unformatted Globally Unique Identifier (GUID).

S.206: System shall support user entry and display of up to three digits to the right of the decimal place, without the system rounding or truncating. The decimal point is represented by a period and shall be used as follows:

- Only when there are significant digits to the right of the decimal

- When there is a digit before and after the decimal point
- Not with whole numbers

For example, consider the following possible values:

Correct Format Examples	Incorrect Format Examples
2.5	.27
0.15	3.00
21	14.

S.209: The customer shall implement and maintain a Surescripts approved drug compendia that is updated at least monthly.

PBM/Payer Requirements

PA.215: If the PBM/health plan receives a PAInitiationRequest or PARequest that contains a PAPriorityIndicator of "X" (expedited), the PBM/health plan shall process the request and respond in an expedited manner. If the PAInitiationRequest or PARequest needs to be manually reviewed for priority, then the PBM/health plans shall make the priority indicator available for display.

PA.216: If the provider vendor populates the Provider composite, the PBM system shall make the Provider composite contents available to be viewed by the user.

PA.217: The PBM/payer system shall make available for display the QuestionSetComment from the PARequest and the PAAppealRequest if the prior authorization request would otherwise be denied or approved with modifications.

PA.218: The PBM/payer system shall only send one PAResponse with a status of 'InProcess' and include the ExpectedResponseDate and/or PANote. This PAResponse is to be followed by a subsequent PAResponse with the final determination or request for additional information.

10. Message Example

PA Submitted by a Person Other than the Prescriber

In this example, the initiator of the prior authorization process is a staff member in the provider vendor's organization who can be reached at a different phone number than the prescriber. To provide this additional contact information, the ePA standard enables the initiating person to provide their separate contact information in the Provider element of the prior authorization message (see Provider Elements in the NCPDP SCRIPT Implementation Guide for more information). If the PBM/payer needs to call or fax the initiator during processing, it uses the contact information contained in this Provider element rather than the phone or fax in the Prescriber segment.

The requested medication is Lovenox (Lovenox 40 MG Per 0.4 ML Prefilled Syringe; RxNorm SBD 854236; NDC 58016487201).

> This example does not include the Status messages or any other part of the normal message flow.

PAInitiationRequest (from Initiating Party)

XML

PAInitiationRequest (from Initiating Party)

```
<?xml version="1.0" encoding="utf-8"?>
<Message DatatypesVersion="20230115" TransportVersion="20230115" TransactionDomain="SCRIPT"
TransactionVersion="20230115" StructuresVersion="20230115" ECLVersion="20230115">
    <Header>
        <To Qualifier="ZZZ">EPAINI</To>
        <From Qualifier="D">6842917100001</From>
        <MessageID>4e55ea3f923b4e14a436a3f08554f488</MessageID>
        <SentTime>2015-09-04T17:14.1Z</SentTime>
        <SenderSoftware>
            <SenderSoftwareDeveloper>MDLITE</SenderSoftwareDeveloper>
            <SenderSoftwareProduct>443</SenderSoftwareProduct>
            <SenderSoftwareVersionRelease>2.1</SenderSoftwareVersionRelease>
        </SenderSoftware>
        <TestMessage>false</TestMessage>
    </Header>
    <Body>
        <PAInitiationRequest>
            <PAReferenceID>c7eb89cc8a3c43c8818e456627a91c2d</PAReferenceID>
            <BenefitsCoordination>
                <PayerIdentification>
                    <PayerID>P00000000001192</PayerID>
                    <ProcessorIdentificationNumber>PCN9110</ProcessorIdentificationNumber>
                    <IINNumber>994336</IINNumber>
                </PayerIdentification>
                <PayerName>PBMA</PayerName>
                <CardholderID>209837423</CardholderID>
                <CardHolderName>
                    <LastName>JOHNSON</LastName>
                    <FirstName>RONALD</FirstName>
                    <MiddleName>C</MiddleName>
                </CardHolderName>
                <GroupID>0998723</GroupID>
                <GroupName>9110/3TIER/25</GroupName>
                <PBMMemberID>0398473070389475</PBMMemberID>
            </BenefitsCoordination>
            <Patient>
                <HumanPatient>
                    <Identification>
                        <MedicareBeneficiaryID>333445556A</MedicareBeneficiaryID>
                    </Identification>
                    <Names>
                        <Name>
                            <LastName>SMITH</LastName>
                            <FirstName>MAUREEN</FirstName>
                        </Name>
                    </Names>
                    <GenderAndSex>
                        <AdministrativeGender>F</AdministrativeGender>
                    </GenderAndSex>
                    <DateOfBirth>
                        <Date>1934-11-27</Date>
                    </DateOfBirth>
                    <Address>
                        <AddressLine1>201 GOLDEN LANE</AddressLine1>
                        <AddressLine2>APT 204</AddressLine2>
                        <City>CINCINNATI</City>
```

```
                <StateProvince>OH</StateProvince>
                <PostalCode>45233</PostalCode>
                <CountryCode>US</CountryCode>
            </Address>
        </HumanPatient>
    </Patient>
    <Pharmacy>
        <Identification>
            <NCPDPID>7701631</NCPDPID>
            <NPI>7878787879</NPI>
        </Identification>
        <BusinessName>CINCINNATI PHARMACY</BusinessName>
        <Address>
            <AddressLine1>100 OMNIBUS LANE</AddressLine1>
            <City>CINCINNATI</City>
            <StateProvince>OH</StateProvince>
            <PostalCode>45202</PostalCode>
            <CountryCode>US</CountryCode>
        </Address>
        <CommunicationNumbers>
            <PrimaryTelephone>
                <Number>8595559000</Number>
            </PrimaryTelephone>
        </CommunicationNumbers>
    </Pharmacy>
    <Prescriber>
        <NonVeterinarian>
            <Identification>
                <NPI>1992091797</NPI>
            </Identification>
            <Names>
                <Name>
                    <LastName>ePA-MessageManager30</LastName>
                    <FirstName>Provider</FirstName>
                </Name>
            </Names>
            <Address>
                <AddressLine1>123 Main st</AddressLine1>
                <City>Minneapolis</City>
                <StateProvince>MN</StateProvince>
                <PostalCode>55123</PostalCode>
                <CountryCode>US</CountryCode>
            </Address>
            <CommunicationNumbers>
                <PrimaryTelephone>
                    <Number>6518553000</Number>
                </PrimaryTelephone>
                <Fax>
                    <Number>6518553001</Number>
                </Fax>
            </CommunicationNumbers>
        </NonVeterinarian>
    </Prescriber>
    <MedicationPrescribed>
        <DrugDescription>LOVENOX 40 MG PER 0.4 ML PREFILLED SYRINGE</DrugDescription>
        <Product>
            <DrugCoded>
                <NDC>58016487201</NDC>
                <Strength>
```

```
                        <StrengthValue>100</StrengthValue>
                        <StrengthForm>
                            <Code>C42945</Code>
                        </StrengthForm>
                        <StrengthUnitOfMeasure>
                            <Code>C42576</Code>
                        </StrengthUnitOfMeasure>
                    </Strength>
                </DrugCoded>
            </Product>
            <Quantity>
                <Value>30</Value>
                <CodeListQualifier>38</CodeListQualifier>
                <QuantityUnitOfMeasure>
                    <Code>C48540</Code>
                </QuantityUnitOfMeasure>
            </Quantity>
            <DaysSupply>30</DaysSupply>
            <Diagnosis>
                <ClinicalInformationQualifier>1</ClinicalInformationQualifier>
                <Qualifier>ABF</Qualifier>
                <Value>E11.9</Value>
                <Description>DIABETES MELLITUS</Description>
            </Diagnosis>
            <Sig>
                <SigText>INJECT 40 MG SUBCUTANEOUSLY ONCE DAILY</SigText>
            </Sig>
        </MedicationPrescribed>
        <Provider>
            <Identification>
                <NPI>7777777777</NPI>
            </Identification>
            <Names>
                <Name>
                    <LastName>WEISS</LastName>
                    <FirstName>TERRY</FirstName>
                    <Suffix>RN</Suffix>
                </Name>
            </Names>
            <CommunicationNumbers>
                <PrimaryTelephone>
                    <Number>8595559880</Number>
                </PrimaryTelephone>
                <Fax>
                    <Number>8595559770</Number>
                </Fax>
            </CommunicationNumbers>
        </Provider>
    </PAInitiationRequest>
</Body>
</Message>
```

11. Message Processing Summary

This section explains the message processing that takes place for messages exchanged between a message sender and a message receiver. These messages are initiated in the form of a request and an acknowledgement response message is returned. The sender stays connected until the receiver sends an acknowledgment response. Depending on connectivity, the actual message vehicle may vary.

The system (Surescripts) stores round-trip messages until the receiver acknowledges the message or until the specified time limit has elapsed. If the timeout elapses before the message is processed, Surescripts will return an error message to the sender as the reply (explained below).

> **Note:** Errors are not necessarily processed in the order as shown below. The row number is there to help the reader indicate which error code they may be discussing.

Row	Event	Process	Type of SCRIPT message	SCRIPT Code	SCRIPT Description Code	Message Text
1.0	Surescripts receives a message.					
1.1		Surescripts fails to recognize the sender OR the message is malformed.	NA (HTTP 400 Bad Request Error is returned rather than a SCRIPT message)	NA	NA	Bad request.
1.2		Sender security credentials are missing.	Error	900 – Message rejected	Blank	Missing authentication info in header. [Additional message-specific error specifics]
1.3		Sender security credentials are invalid.	Error	900 – Message rejected	Blank	Authentication failed. Invalid username/password. [Additional message-specific error specifics]
1.4		The incoming message exceeds Surescripts' size limit of 20MB.	NA (HTTP 500 Internal Server Error is returned rather than a SCRIPT message)	NA	NA	Internal server error.
1.5		Any other system-related error which makes it impossible for Surescripts to accept the message.	NA (HTTP 500 Internal Server Error is returned rather than a SCRIPT message)	NA	NA	Internal server error.
2.0	Surescripts processe	No acknowledgement is sent to the sender at this stage of processing.				

Row	Event	Process	Type of SCRIPT message	SCRIPT Code	SCRIPT Description Code	Message Text
	s the message.					
2.1		Surescripts determines that the sending system is not certified to send this type of message.	Error	700 – Configuration error	4030 – Sender not allowed to send this message type.	Sender portal does not support this message type.
2.2		Surescripts determines that the receiving system is not certified to receive this type of message.	Error	700 – Configuration error	4040 – Receiver does not support receiving this message type.	Receiver portal does not support this message type.
2.3		Sender portal configuration error at Surescripts.	Error	700 – Configuration error	4020 – Intermediary system error.	Surescripts system error. Sender portal configuration error.
2.4		Receiver portal configuration error at Surescripts.	Error	700 – Configuration error	4020 – Intermediary system error.	Surescripts system error. Receiver portal configuration error.
2.5		Surescripts determines that the sender's provider vendor SPI or PBM/payer Unique ID was not found in the Surescripts directory.	Error	700 – Configuration error	4030 – Sender not allowed to send this message type.	Sender ID not in directory.
2.6		Surescripts determines that the receiver's SPI or PBM/payer Unique Identifier is not valid.	Error	700 – Configuration error	4040 – Receiver does not support receiving this message type.	Receiver ID not in directory.
2.7		Surescripts determines that the sending provider vendor SPI has not been authorized to send this type of message. (The sender's SPI has not been assigned the ePA service level.)	Error	700 – Configuration error	4030 – Sender not allowed to send this message type.	Sender not allowed to send this type of message.

Row	Event	Process	Type of SCRIPT message	SCRIPT Code	SCRIPT Description Code	Message Text
2.8		Surescripts determines that the receiving provider vendor SPI or PBM/payer unique identifier does not support this type of message. (The sender's SPI or PBM/payer unique identifier has not been assigned the ePA service level.)	Error	700 – Configuration error	4040 – Receiver does not support receiving this message type.	Receiver does not support this message type.
2.9		Surescripts fails to recognize the type of message.	Error	900 – Message rejected		Unable to recognize incoming data as a supported message type.
2.10		Based on the message type, Surescripts determines that a syntax error is present.	Error	900 – Message rejected	3000 – Does not follow NCPDP standard or implementation guide rules.	Invalid syntax. Message fails schema validation. [Additional message-specific error specifics]
2.11		Message fails Surescripts business rule validations.	Error	900 – Message rejected	3000 – Does not follow NCPDP standard or implementation guide rules.	Message fails Surescripts validations. The NDC must be a valid 11-digit code and should meet the qualifications for a Representative NDC. Namespace / version validation: · DatatypesVersion is invalid [field that is in error] doesn't match expected value 2023011 · ECLVersion is invalid [field that is in error] doesn't match expected value 2023011 · PA-StructuresVersion is invalid [field that is in error] doesn't match expected value 2023011 · StructuresVersion is invalid [field that is in error] doesn't match expected value 2023011

Row	Event	Process	Type of SCRIPT message	SCRIPT Code	SCRIPT Description Code	Message Text
						· TransactionVersion is invalid [field that is in error] doesn't match expected value 2023011 · TransportVersion is invalid [field that is in error] doesn't match expected value 2023011 · TransactionDomain is invalid [field that is in error] doesn't match expected value SCRIPT NPI LUHN check: · (Prescriber, Provider, Supervisor, Pharmacy, Facility) Identification/NPI is invalid. Fails LUHN check. PayerID in PAInitiationRequest: · PAInitiationRequests must include PayerID in BenefitsCoordination element PAInitiationRequests addressed to EPAINI: · All PAInitiationRequests must be addressed to EPAINI DaysSupply, PADaysSupply, or PADosesPerDay: · MedicationPrescribed/DaysSupply or PADaysSupply and PADosesPerDay must be a whole number Prescriber Fax: · Prescribers fax number is required PO Box in Address Line 1: · A PO Box must not be sent in Address Line 1 for the Pharmacy, Prescriber, Facility, Supervisor, or Provider elements

Row	Event	Process	Type of SCRIPT message	SCRIPT Code	SCRIPT Description Code	Message Text
						Unsolicited PAInitiationResponse: · Unsolicted PAInitiationResponse must be addressed to EPAINI Renewals for expiring PA: · Renewals for expiring PA must include the expire PACaseID and ExpirationDate AuthorizationDetail or AuthorizationPeriod in Closed – CF PAInitiationResponse: · Closed Prior Authorization with a duplicate/approved Reason code require either an Authorization Detail or Authorization Period Unsolicited PARequest: · Unsolicited model is not supported. AuthorizationDetail or AuthorizationPeriod in Approved PAResponse: · Appoved PA responses require either an Authorization Detail or Period AuthorizationDetail and AuthorizationPeriod in PartiallyDenied PAResponse: · Partially denied PA response require both an Authorization Detail and Period AuthorizationDetail or AuthorizationPeriod in Closed – CF PAResponse: · Closed Prior Authorization with a duplicate/approved Reason code require either

Row	Event	Process	Type of SCRIPT message	SCRIPT Code	SCRIPT Description Code	Message Text
						Authorization Detail or Authorization Period · The application shall make the AuthorizationDetail and AuthorizationPeriod available for display when receiving a PAResponse AuthorizationDetail or AuthorizationPeriod in PAAppealResponse: · Appoved PAAppeal responses require either an Authorization Detail or Period Improper use of QS quantity code: · script:MedicationPrescribed/structures: Quantity is invalid if Value is >0, Code List Qualifier must be 38 but was 'QS' or · script:MedicationPrescribed/structures: Quantity is invalid if Value is 0, Code List Qualifier must be QS but was '38' Invalid Days Supply value: · DaysSupply is invalid Value is [field that is in error], Days Supply must be an integer with 3 or fewer digits · Invalid DescriptionCode in Status message: · DescriptionCode: [field that is in error] is invalid for Status messages · NextQuestionID/Default NextQuestionID Validation: · QuestionSet/Question/DefaultNextQuestionID is invalid for QuestionID [Question #].

Row	Event	Process	Type of SCRIPT message	SCRIPT Code	SCRIPT Description Code	Message Text
						DefaultNextQuestionID must be a QuestionID or END · QuestionSet/Question/QuestionID: Q# is invalid. Question ID must be unique.
2.12		Surescripts experiences a system failure during processing.	Error	900 – Message rejected		Surescripts system error.
3.0	Surescripts passes the message on to the recipient.					
3.1		Surescripts is unable to deliver the message to the receiver (e.g. could not establish communication session).	Error (For Sender)	600 – Communication problem – try again later	4000 – Intermediary is unable to deliver message to the recipient.	Communication problem. Try again later. Unable to deliver message to receiver.
3.2		The message times out before a response is given.	Error (For Sender)	600 – Communication problem – try again later	008 – Request timed out before response could be received.	Connection with receiver timed out before response was returned. Status of message at receiver is unknown.
3.3		Recipient is unable to recognize the type of message or does not have enough info to create an NCPDP Error message. Recipient sends Surescripts HTTP 400 response and text message that indicates “Message cannot be identified nor processed”.	HTTP 400 (from receiver) Error (For Sender)	602 – Receiver System Error	007 – Internal processing Error has occurred	From receiver: “Message cannot be identified nor processed”.
4.0	Recipient responds to the message.					

Row	Event	Process	Type of SCRIPT message	SCRIPT Code	SCRIPT Description Code	Message Text
4.1		Surescripts is unable to process the response from the recipient.	Error (to the sender of the original message)	For Sender: 600 – Communication problem – try again later	For Sender – 4010 – Intermediary is unable to process response from recipient	Unable to process response from receiver. Message delivered but status of message at receiver is unknown.
4.2		The recipient returns an Error message.	Pass through Error to initiating party	Surescripts passes Error message through	[as populated by the recipient]	[as populated by the recipient]
4.3		The recipient returns a valid Status message.	Status (to initiating party)	010 or 000 (As populated by the recipient)		Note: If the recipient is the end point and the return message is a Status, the Status code is 010. If the recipient is not an end point, the Status code is 000.

12. Message Linkage

Message linkage requirements and best practices are described in the table below. Four key fields are used to link messages together, as defined below. This linkage allows a software system to tie one message to another. For example, on a PAInitiationResponse the PBM/payer system needs to find the PAInitiationRequest to which the provider vendor is referring. This linkage should be done in an automated manner.

> NCPDP refers to message linkage as trace number.

MessageID – The unique number assigned to each message. This ID initiates the electronic linking of subsequent message types.

RelatesToMessageID – The MessageID of the most recent and relevant message in the workflow.

PAReferenceID – Assigned by the prescribing system on the initial message (or on PARequest message for Prior Authorization renewal requests) and is used by the prescribing system as a tracking identifier on all request and response prior authorization messages. The identifier must be unique per prescribing system for at least 18 months.

PACaseID – PBM/payer-assigned identifier for this request and used by the PBM/payer as a tracking identifier on all request and response prior authorization messages.

PAInitiationRequest and Response Fields

PAInitiationRequest	PAInitiationResponse	PARequest	PAResponse
MessageID = 123	MessageID = ABC	MessageID = 456	MessageID = DEF
RelatesToMessageID: If Eligibility is used, RelatesToMessageID contains the Interchange Control Number (ISA13 of Eligibility Response (271)). If RTPB is used, then RelatesToMessageID contains the RTPB Response MessageID. If RxChange is used, RelatesToMessageID contains the RxChangeRequest MessageID. **Note**: If multiple products are used, use the latest MessageID that was used as a trigger for the PA.	RelatesToMessageID = 123	RelatesToMessageID = ABC	RelatesToMessageID = 456
PAReferenceID = XYZ	PAReferenceID = XYZ	PAReferenceID = XYZ	PAReferenceID = XYZ
PACaseID = N/A	PACaseID = 999	PACaseID = 999	PACaseID = 999
Status	Status	Status	Status
MessageID = 111	MessageID = 222	MessageID = 333	MessageID = 444
RelatesToMessageID = 123	RelatesToMessageID = ABC	RelatesToMessageID = 456	RelatesToMessageID = DEF

PA Appeal and Cancel Request and Response

PA Appeal Request	PA Appeal Response	PA Cancel Request	PA Cancel Response
MessageID = 789	MessageID = GHI	MessageID = 012	MessageID = JKL
RelatesToMessageID = DEF	RelatesToMessageID = 789	RelatesToMessageID relates to the most recent message received from your trading partner. See definition above. Since PA Cancel Requests can be sent at any time after receiving case ID, this would relate to the most recent message. For example, if this was sent after provider sent the PARequest (and had not yet received a PAResponse from the PBM/payer) the RelatesTo ID would be ABC.	RelatesToMessageID = 012
PAReferenceID = XYZ	PAReferenceID = XYZ	PAReferenceID = XYZ	PAReferenceID = XYZ
PACaseID = 999	PACaseID = 999	PACaseID = 999	PACaseID = 999
Status	Status	Status	Status
MessageID = 555	MessageID = 666	MessageID = 777	MessageID = 888
RelatesToMessageID = 789	RelatesToMessageID = GHI	RelatesToMessageID = 012	RelatesToMessageID = JKL

Third-Party Processor Message Linking

When PBM/payers return a CO or CP reason code, the message becomes marked as closed for third-party processing. If Surescripts' third-party processor search finds a match, Surescripts creates a secondary PAInitiationRequest based upon the initial PAInitiationRequest. The recipient is updated and a secondary MessageID with a prefix of EPAINI is sent to the third-party processor. When the third-party processor responds, the message has a RelatesToMessageID that has a prefix of EPAINI and references the secondary PAInitiationRequest.

> If the PBM/payer referenced is not connected to Surescripts for electronic prior authorization but the third-party processor is, Surescripts forwards the message to the third-party processor as if they were the PBM/payer – no modifications to the MessageID are made.

PAInitiationRequest and Response from Third-Party Processor

PAInitiationRequest (To EPAINI)	PAInitiationResponse (From PBM/Payer - Closed for Third Party)	Second PAInitiationRequest (To Third Party)	Second PAInitiationResponse (From Third Party)**	PARequest (To Third Party)	PAResponse (From Third Party
MessageID = 123	MessageID = ABC	MessageID = EPAINI321	MessageID = DEF	MessageID = 789	MessageID = 010
RelatesToMessageID contains the Interchange Control Number (ISA13 of Eligibility Response (271)).	RelatesToMessageID = 123	RelatesToMessageID = ABC	RelatesToMessageID = EPAINI321	RelatesToMessageID = DEF	RelatesToMessageID = 789
PAReferenceID = XYZ	PAReferenceID = XYZ	PAReferenceID = XYZ	PAReferenceID = XYZ	PAReferenceID = XYZ	PAReferenceID = XYZ
PACaseID = N/A	PACaseID = N/A	PACaseID = N/A	PACaseID = 123456	PACaseID = 123456	PACaseID = 123456
Status	Status	Status	Status	Status	Status
MessageID = 111	MessageID = 222	MessageID = 333	MessageID = 444	MessageID = 555	MessageID = 666
RelatesToMessageID = 123	RelatesToMessageID = ABC	RelatesToMessageID = EPAINI321	RelatesToMessageID = DEF	RelatesToMessageID = 789	RelatesToMessageID = 010

If Surescripts cannot find a third-party processor, Surescripts will return a Closed PAInitiationResponse with EPAINI in the From field and possibly a link to a web-based PA forms provider.

13. 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

14. Document Change Log

2026-09-01

Section Title	Change Description
Using Accelerator Web Service API	Removed the **authorizationDetail** element row from the **Response Status – "Approved" Response Field Description** table. The PAResponse schema does not permit **authorizationDetail** in an **Approved** response; only **authorizationPeriod** is allowed. **Note:** No change was made to the **Response Status – "Closed" Response Field Description** table; the schema permits **authorizationDetail** in a **Closed** response.
Using Accelerator Web Service API	Updated the **Approved Response** code example: replaced the **authorizationDetail** element with **authorizationPeriod**, corrected **iseAppealSupported** to **isEAppealSupported**, and fixed JSON syntax errors.
Using Accelerator Web Service API	Updated the **authorizationPeriod** description in the **Response Status – "Approved" Response Field Description** table to remove the statement "Either authorizationDetail or authorizationPeriod will be sent," as **authorizationPeriod** is the only authorization element permitted in an **Approved** response.
Element Details	Corrected the **Address** child element descriptions under **Prescriber** and **Provider**, which incorrectly referenced patient address information.

2026-08-21

Section Title	Change Description
N/A	N/A - Developer Portal - initial release