



**SERFF State API Version 4.1
Web Services
Developer Guide**
SERFF 7.7
Last Updated 4/08/2016

TABLE OF CONTENTS

INTRODUCTION.....	3
OVERVIEW.....	4
USAGE LIMITATIONS.....	6
SECURITY AND AUTHENTICATION.....	7
SHARED OBJECT DETAIL.....	8
EXCEPTIONS	9
WEB SERVICES.....	11
Data Request Services.....	11
Correspondence Services	14
Update Filing Services	22
Aggregate PDF Services	26
Handling Attachments Services	29
Download Attachment Handling Services.....	32
APPENDIX A: QUERY CONDITION	35
Dates	36
PPACA Field Not Fully Supported	36
HFAI Public Comments	36
APPENDIX B: CLIENT EXCEPTION CODES.....	37
APPENDIX C: SERVER EXCEPTION CODES	38
APPENDIX D: DOCUMENT REVISIONS LIST	39
APPENDIX E: ACRONYMS	40

INTRODUCTION

The State Application Programming Interface (API) enables states to integrate their systems with the System for Electronic Rate and Form Filing (SERFF) application data.

The State API version 4.1 delivers enhancements to the 4.0 version, including new update services for correspondence, to facilitate post-submission data changes allowed by the business requirements.

The State API has usage limitations to regulate system load.

Please send technical questions regarding the API to wsreq@naic.org.

OVERVIEW

The State API web services are grouped into six categories:

1. Data Request Services.
2. Correspondence Services.
3. Update Filing Services.
4. Aggregate Adobe Acrobat Portable Document Format (PDF) Services.
5. Upload Attachment Handling Services.
6. Download Attachment Handling Services.

Data Request Services:

Queries for information and attachments on a Filing and Plan Binder. This grouping has two web services:

1. queryFiling.
2. queryBinder.

Correspondence Services:

Submits an Objection Letter, Note To Filer, or a Disposition. This grouping has eight web services:

1. submitObjectionLetter.
2. finalizeObjectionLetter.
3. updateFinalizedObjection.
4. submitNoteToFiler.
5. finalizeNoteToFiler.
6. submitDisposition.
7. finalizeDisposition.
8. updateFinalizedDisposition.

Update Filing Services:

Updates four data values on a Filing: Reviewer Name, State Tracking Number, State Status and Deemer Date. This grouping has four web services:

1. setReviewerName.
2. setStateTrackingNumber.
3. setStateStatus.
4. setDeemerDate.

Aggregate PDF Services:

Creates a single PDF file of the entire filing and all compatible attachments. This grouping has one web service: Pdf_PdfRequest.

Upload Attachment Handling Services:

Manages the upload of file attachments to SERFF Filing or Aggregate PDF. This service is designed to provide large file flexibility.

Download Attachment Handling Services:

Manages the download of file attachments from a SERFF Filing or Aggregate PDF. This service is designed to provide large file flexibility.

USAGE LIMITATIONS

The State API limits web service calls to regulate system load. These limits are imposed per minute and per hour.

When an initial web service call is made, two time stamps are stored on the system:

- One to provide a reference for calls per minute and
- One for calls per hour

Each call within a minute of the minute timestamp is counted against the maximum allowed web service calls per minute AND against the maximum allowed web service calls per hour.

The maximum allowed web service call per minute is 60. The maximum allowed web service call per hour is 1000.

If either call limit is exceeded, a Simple Object Access Protocol (SOAP) fault is returned containing two elements:

- A message describing the call limit exceeded.
- The number of milliseconds until the next allowable call.

Once a call limit is exceeded, no further web service calls are processed and accepted, until the call limit expires.

Once a call limit expires, a new time stamp for a new call limit is generated and stored, when a web service call is made.

SECURITY AND AUTHENTICATION

Security is managed by using HTTP authentication, a user name and password.

User: The user name and password are assigned (in advance by NAIC), to the SERFF State API Web Service user. This is done to uniquely identify the user.

The SERFF State API Web Service user provides the assigned user name and password , when calling the SERFF State API Web Service.

Passwords are encrypted in the database.

Passwords expire every 90 days.

The SERFF application enforces data access by using defined privileges for a user. The NAIC sets these privileges. A SERFF State API Web Service user is not allowed to modify the user privileges.

Connection: The SERFF State API Web Service is accessible over the public Internet, through the Transport Layer Security (TLS).

The SERFF State API Web Service User software solution must be able to manage TLS service requests and responses.

SHARED OBJECT DETAIL

Shared objects are used by more than one web service.

attachmentIdentifier

Contains two string values: *attachmentName* and *attachmentId*. These values are used by several web services in the SERFF API.

pendingReport

Contains one or more attachmentIdentifier shared objects and one string value: *temporaryReportId*, which is used by *finalizeReport*.

EXCEPTIONS

There are four types of Exceptions discussed in this section:

1. ClientException
2. ServerException
3. CallRateExceededException
4. QueryRowLimitException

1. ClientException:

A ClientException indicates there is an issue in the client's application, which needs to be addressed by the client (e.g., missing data, invalid combinations of data).

A ClientException contains a *ClientException* object with two string elements:

Code: A pre-defined code indicating the error.

Message: A descriptive message of the error.

Note: HFAI Public Comments

If the user is enabled as an HFAI state with Public Comments enabled, it is possible that a public user entered comments on the public filings through the HFAI web application.

A table of exception codes is found in Appendix B: Client Exception Codes.

2. ServerException:

A ServerException indicates an issue with the web services application.

A ServerException contains a *ServerException* object with two string elements:

Code: A pre-defined code indicating the error.

Message: A descriptive message of the error.

A table of exception codes is found in Appendix C: Server Exception Codes.

3. CallRateExceededException:

A CallRateExceededException indicates that the client has reached the queryFiling or queryBinder web service Usage Quota.

The State API limits the usage of queryFiling and queryBinder to 60 calls per minute and 1000 calls per hour. If the user exceeds this limit, a CallRateExceededException is received.

A `CallRateExceededException` contains a *CallRateExceededException* object with two string elements:

- `millisecondsUntilNextAllowedCall`—the number of milliseconds until next allowed web service call.
- `Message`—a descriptive message of the error.

4. QueryRowLimitException:

If the query condition that is submitted to a `queryFiling` or `queryBinder` web service produces more than 600 results, a `QueryRowLimitException` is received. This exception is designed to prevent long-running queries from consuming system resources.

If the user receives this exception, the condition criteria must be refined to reduce the expected number of query results.

WEB SERVICES

Data Request Services	
QueryFiling	
Input	A queryFiling object.
Returns	QueryFilingResponse object with ResultTableType with 0 to many Rows.
Conditions	The queryFiling object condition string value must be valid. Note: PPACA field is not fully supported as a Condition option (see page 36).
Results	Information is returned from SERFF; no changes are made.
Object Detail	field Contains one string value, which represents a field name value that must be returned.
	fields Contains one or more field objects.
	queryFiling Contains one fields object and three string values: <ol style="list-style-type: none"> 1. form Indicates the name of the object type that must be returned (e.g., Filing). 2. condition See Appendix A: Query Condition (for information on writing a condition). 3. attachmentNamePattern <ol style="list-style-type: none"> a. Leave the value blank (null). No attachmentIdentifiers are returned. b. Use an asterisk (*). An <i>attachmentIdentifier</i> is returned for each attachment on the filing documents found by <i>condition</i>. c. Use a string representing a partial or entire file name. An <i>attachmentIdentifier</i> is returned for each attachment with a file name containing that string. d. Only attachments on the filing documents found by <i>condition</i> are evaluated.

Data Request Services (cont'd)	
QueryFiling (cont'd)	
QueryFilingResponse	Contains one ResultTableType object.
ResultTableType	Contains 1 to 600 Row objects. All queries are limited to 600 results. If a query condition results in more than 600 rows returned, a QueryRowLimitException is received. This exception is designed to prevent long-running queries from consuming system resources. If the user receives this exception, the search criteria must be refined by adding more conditions to limit the selection.
	Row Contains one or more Value objects, and, an array of <i>attachmentIdentifier</i> objects.
	Value Contains an array of String, which represents the values of the Fields requested in the query.
QueryBinder	
Input	A queryFiling object.
Returns	QueryFilingResponse object with ResultTableType 0 to many Rows.
Conditions	The queryFilingResponse object condition string value must be valid. Note: PPACA field is not fully supported as a Condition option (see page 36).
Results	Information is returned from SERFF; no changes are made.

Data Request Services (cont'd)	
QueryBinder (cont'd)	
Object Detail	field Contains one string value which represents a field name value that must be returned.
	fields Contains one or more field objects.
	queryBinder Contains one fields object and three string values: <ol style="list-style-type: none"> 1. form: Indicates the name of the object type that must be returned (e.g., Filing). 2. condition: See Appendix A: Query Condition (for information on writing a condition). 3. attachmentNamePattern: <ol style="list-style-type: none"> a. Leave the value blank (null). No attachmentIdentifiers are returned. b. Use an asterisk (*). An <i>attachmentIdentifier</i> is returned for each attachment on the filing documents found by <i>condition</i>. c. Use a string representing a partial or entire file name. An <i>attachmentIdentifier</i> is returned for each attachment with a file name containing that string. Only attachments on the filing documents found by <i>condition</i> are evaluated.
	QueryBinderResponse Contains one ResultTableType object.
	ResultTableType Contains 1 to 600 Row objects. All queries are limited to 600 results. If the query condition results in more than 600 rows returned, a QueryRowLimitException is received. This exception is designed to prevent long-running queries from consuming system resources. If the user receives this exception, the search condition must be refined by adding more conditions to limit the selection.
	Row Contains one or more Value objects, and, an array of <i>attachmentIdentifier</i> objects.
	Value Contains an array of String, which represents the values of the Fields requested in a query.

Correspondence Services	
Shared Objects	
Objection Letter	<p>Contains values as follows: No values are required in the Web Services Description Language (WSDL) except for comments on Objection. The service enforces the data required for that call.</p> <ul style="list-style-type: none"> • Objections are optional. If one is included, the comment field is required. • objectionLetterStatus (string). • effectiveDate (date). • introductionText (string). • conclusionText (string). • respondByDate (date). • objections (ObjectionGroup). • attachments (string array).
ObjectionGroup	Contains one or more Objection.
Objection	<ul style="list-style-type: none"> • comments • array of scheduleItemId
Disposition	<p>Contains values as follows: No values are required in the WSDL. The service enforces the data required for that call.</p> <ul style="list-style-type: none"> • dispositionDate (date). • effectiveOrImplementationDate (date). • effectiveRenewalDate (date). • status (string) (required). • comments (string) (required). • sumOverallRateImpact (number). • sumOverallRateIndicated (number). • CompanyRateGroup object. • Attachments.
CompanyRateGroup	Contains one or more CompanyRate objects.
CompanyRate	<p>Contains values as follows:</p> <ul style="list-style-type: none"> • cocode (string) (required only if multiple companies). • overallRateImpact (number). • premiumRateChange (number). • affectedPolicyHolders (number). • premium. • maxChange (number). • minChange (number).

Correspondence Services (cont'd)	
submitObjectionLetter	Submit an Objection Letter to SERFF.
Input	A submitObjectionLetterRequest object.
Returns	A submitObjectionLetterResponse object.
Conditions	<ul style="list-style-type: none"> • <i>SerffTrackingNumber</i> string must reference a unique Filing. • The target filing must not have a closed status.
Results	<ul style="list-style-type: none"> • The Objection Letter is created on the Filing. • When the first Objection Letter is created on a Filing, the SERFF Status changes to "Pending Industry Response". • SERFF log is updated when an Objection Letter is submitted.
Attachments	<ul style="list-style-type: none"> • Attachments specified in the submitObjectionLetter object are not included in the Objection Letter, until the Upload Attachment Handling Services is called. • The Objection Letter is not submitted to Industry in SERFF, until all specified attachments are uploaded and finalizeObjectionLetter is called.
Required Data	<ul style="list-style-type: none"> • Filing Id • Introduction Text • Objection Letter Status • Effective Date
Object Detail	submitObjectionLetter Contains values as follows: <ul style="list-style-type: none"> • filingId (string) (required) (actual SERFF Tracking Number). • objectionLetter object (required).
	ObjectionLetterResponse <ul style="list-style-type: none"> • attachmentIdentifier

Correspondence Services (cont'd)	
finalizeObjectionLetter	If attachments are submitted in an Objection Letter, the finalizeObjectionLetter service must be called at the end of the process.
Input	A finalizeObjectionLetterRequest object.
Returns	A finalizeObjectionLetterResponse object.
Conditions	<ul style="list-style-type: none"> objectionLetterId must reference a pending submitObjectionLetter. All attachments must be uploaded successfully, by using the Upload Attachment Handling Services. All attachments must be completed successfully, by using endUpload
Results	<ul style="list-style-type: none"> The Objection Letter indicated by <i>objectionLetterId</i> is created and submitted in SERFF. If attachments are included, they appear on the Objection Letter. This occurs only if the attachments are uploaded successfully, by utilizing the Upload Attachment Handling Services.
Object Detail	finalizeObjectionLetter Contains values as follows: <ul style="list-style-type: none"> objectionLetterId (string) (required).
	finalizeObjectionLetterResponse Contains one string value which represents the status of the Objection Letter: "SUCCESS".
updateFinalizedObjectionLetter	This service enables updating of a finalized ObjectionLetter.
Input	An objectionLetterFinalizedUpdateRequest object.
Returns	An objectionLetterFinalizedUpdateResponse object.
Conditions	<i>objectionLetterId</i> must reference a finalized Objection Letter.
Results	The Objection Letter indicated by <i>objectionLetterId</i> is updated in SERFF.
Required Data	<ul style="list-style-type: none"> Objection Letter Id Respond By Date

Correspondence Services (cont'd)	
updateFinalizedObjectionLetter (cont'd)	
Object Detail	ObjectionLetterFinalizedUpdateRequest Contains values as follows: <ul style="list-style-type: none"> objectionLetterId (string) (required). objectionLetter object (required).
	ObjectionLetterFinalizedUpdateResponse Contains one string value which represents the status of an Objection Letter update: "SUCCESS".
submitNoteToFiler	
Input	A submitNoteToFilerRequest object.
Returns	A submitNoteToFilerResponse object.
Conditions	A ClientException is generated if the following conditions are not met: <ul style="list-style-type: none"> The SerffTrackingNumber string must reference a unique Filing.
Results	<ul style="list-style-type: none"> A Note to Filer is created on the Filing in SERFF. The SERFF Log document for the Filing is updated.
Attachments	<ul style="list-style-type: none"> Attachments specified in the submitNoteToFiler object are not included in the Note to Filer, until the Upload Attachment Handling Services is called. The Note To Filer is not created in SERFF, until all specified attachments are uploaded and finalizeReport is called.
Object Detail	submitNoteToFiler Contains values as follows: <ul style="list-style-type: none"> SerffTrackingNumber (string) (required). subject (string) (required). comments (string) (required). attachments (string array).
	NoteToFilerResponse <ul style="list-style-type: none"> <i>attachmentIdentifier</i>

Correspondence Services (cont'd)	
finalizeNoteToFiler	If attachments are submitted on a Note To Filer, <i>finalizeNoteToFiler</i> must be called at end of the process.
Input	A finalizeNoteToFilerRequest object.
Returns	A finalizeNoteToFilerResponse object.
Conditions	<p><i>noteToFilerId</i> must reference a pending submitNoteToFiler.</p> <ul style="list-style-type: none"> All attachments must be uploaded successfully, by using the Upload Attachment Handling Services. All attachments must be completed successfully, by using endUpload.
Results	<p>The Note To Filer indicated by <i>noteToFilerId</i> is created and submitted in SERFF.</p> <ul style="list-style-type: none"> If attachments are included, they appear on the Note To Filer. This occurs only if the attachments are uploaded successfully, by utilizing the Upload Attachment Handling Services.
Object Detail	<p>finalizeNoteToFiler Contains values as follows:</p> <ul style="list-style-type: none"> <i>noteToFilerId</i> (string) (required).
	<p>FinalizeNoteToFilerResponse Contains one string value which represents the status of Note To Filer: "SUCCESS".</p>
submitDisposition	
Input	A submitDispositionRequest object.
Returns	A submitDispositionResponse object.

Correspondence Services (cont'd)	
submitDisposition (cont'd)	
Conditions	<p>A ClientException is generated, if the following conditions are not met:</p> <ul style="list-style-type: none"> • The SerffTrackingNumber string must reference a unique Filing. • The target Filing must not have a closed status. • The SERFF Status is updated when the Disposition is submitted. • The SERFF log is updated when the Disposition is submitted.
Results	<p>A Disposition is created in SERFF as a response to the Filing (indicated by the SerffTrackingNumber). The SERFF Log for the Filing is updated.</p>
Required Data	<ul style="list-style-type: none"> • Filing Id • Status • Comments
Attachments	<ul style="list-style-type: none"> • Attachments specified in the submitDispositionReport object are not included in the Disposition, until Upload Attachment Handling Services is called. • The Disposition is not created in SERFF until all specified attachments are uploaded and finalizeDisposition is called.
Object Detail	<p>DispositionRequest Contains values as follows:</p> <ul style="list-style-type: none"> • filingId (string) (required). • disposition object (required).
	<p>DispositionResponse</p> <ul style="list-style-type: none"> • <i>attachmentIdentifier</i>

Correspondence Services (cont'd)	
finalizeDisposition	If attachments are submitted on a Disposition, <i>finalizeDisposition</i> must be called at end of the process.
Input	A finalizeDispositionRequest object.
Returns	A finalizeDispositionResponse object.
Conditions	<ul style="list-style-type: none"> DispositionId must reference a pending submitDisposition. All attachments must be uploaded successfully, by using Upload Attachment Handling Services. All attachments must be completed successfully, by using endUpload.
Results	<ul style="list-style-type: none"> The Disposition indicated by DispositionId <i>is</i> created and submitted in SERFF. When attachments are included, they appear on the Disposition. This occurs only if the attachments are uploaded successfully, by utilizing the Upload Attachment Handling Services.
Object Detail	finalizeDisposition Contains values as follows: <ul style="list-style-type: none"> dispositionId (string) (required).
	finalizeDispositionResponse Contains one string value which represents status of the Disposition: "SUCCESS".
updateFinalizedDisposition	Enables the updating of a finalized Disposition.
Input	A DispositionFinalizedUpdateRequest object.
Returns	A DispositionFinalizedUpdateResponse object.
Conditions	<i>dispositionId</i> must reference a finalized Disposition.
Results	The Disposition indicated <i>by dispositionId</i> is updated in SERFF.
Required Data	<ul style="list-style-type: none"> Disposition Id. Effective/Implementation Date.

Correspondence Services (cont'd)	
updateFinalizedDisposition (cont'd)	
Object Detail	DispositionFinalizedUpdateRequest Contains values as follows: <ul style="list-style-type: none"> • dispositionId (string) (required). • disposition object (required).
	DispositionFinalizedUpdateResponse Contains one string value which represents the Disposition status: "SUCCESS".
processPostSubmissionUpdate	Enables the processing of Post-Submission updates.
Input	A ProcessPostSubmissionUpdate object.
Returns	A ProcessPostSubmissionUpdateResponse object.
Conditions	<i>postSubmissionUpdateId</i> must reference a submitted, un-processed, Post-submission update.
Results	The Post-submission update indicated by <i>postSubmissionUpdateId</i> is approved or disapproved by SERFF (as indicated in the request object). Comments are recorded.
Required Data	<ul style="list-style-type: none"> • Post-Submission Update ID. • Approved Flag. • Additional comments.
Object Detail	ProcessPostSubmissionUpdate Contains values as follows: <ul style="list-style-type: none"> • postSubmissionUpdateId (string) (required). • approved (Boolean) (required). • comment (String) (optional).
	ProcessPostSubmissionUpdateResponse Contains one ProcessPostSubmissionUpdateResponseCode element that can either be SUCCESS or ALREADY_PROCESSED.

Update Filing Services	
setReviewerName	Adds/changes the Reviewer name on a Filing in SERFF.
Input	A setReviewerNameRequest object.
Returns	A setReviewerNameResponse object.
Conditions	<p>A ClientException is generated if the following conditions are not met:</p> <ul style="list-style-type: none"> • The SerffTrackingNumber string must reference a unique Filing. • The Reviewer Name must be a valid user for that state instance.
Results	<ul style="list-style-type: none"> • Name of State Reviewer on the Filing equals input value for Reviewer Name. • The SERFF status is set to "Assigned". • The SERFF Log document for the Filing is updated.
Object Detail	<p>setReviewerName Contains values as follows:</p> <ul style="list-style-type: none"> • serffTrackingNumber (string) (required). • primaryReviewerName (string) (required). • reviewerName (string).
	<p>ReviewerNameResponse Contains one string value: SUCCESS.</p>
setStateTrackingNumber	Adds/changes the State Tracking Number on a Filing in SERFF.
Input	A setStateTrackingNumberRequest object.
Returns	A setStateTrackingNumberResponse object.
Condition	<p>A ClientException is generated, if the following conditions are not met:</p> <ul style="list-style-type: none"> • The SerffTrackingNumber string must reference a unique Filing. • The target filing must not have a closed status.
Results	<ul style="list-style-type: none"> • State tracking number is set to the input value. • SERFF Log document for the Filing is updated.

Update Filing Services (cont'd)	
setStateTrackingNumber (cont'd)	
Object Detail	setStateTrackingNumber Contains values as follows: <ul style="list-style-type: none"> • serffTrackingNumber (string) (required). • stateTrackingNumber (string) (required).
	StateTrackingNumberResponse Contains one string value: SUCCESS.
setStateStatus	Change the State Status on a Filing in SERFF.
Input	A setStateStatusRequest object.
Returns	A setStateStatusResponse object.
Conditions	A ClientException is generated if the following conditions are not met: <ul style="list-style-type: none"> • SerffTrackingNumber string must reference a unique Filing.
Results	<ul style="list-style-type: none"> • The State Status is set to the input value. • SERFF Log document for a Filing is updated.
Object Detail	setStateStatus Contains values as follows: <ul style="list-style-type: none"> • serffTrackingNumber (string) (required). • stateStatus (string) (required).
	StateStatusResponse Contains one string value: "SUCCESS".
setStateScheduleItemStatus	Change the State Status of Schedule Items on a Filing in SERFF.
Input	A setStateScheduleItemStatus object.
Returns	A stateScheduleItemStatusResponse object.

Update Filing Services (cont'd)	
setStateScheduleItemStatus (cont'd)	
Conditions	<p>A ClientException is generated if the following conditions are not met:</p> <ul style="list-style-type: none"> The SerffTrackingNumber string must reference a unique Filing. The <i>ScheduleItemId</i> is associated with the referenced SerffTrackingNumber.
Results	<ul style="list-style-type: none"> The State Status for the Schedule Item is set to the input value. The SERFF Log document for the filing is updated.
Object Detail	<p>setStateScheduleItemStatus Contains values as follows:</p> <ul style="list-style-type: none"> serffTrackingNumber (string) (required). scheduleItemStateStatusGroup (Object) (required).
	<p>ScheduleItemStateStatusGroup Contains values as follows:</p> <ul style="list-style-type: none"> ScheduleItemStateStatus (object) (required).
	<p>ScheduleItemStateStatus Contains values as follows:</p> <ul style="list-style-type: none"> scheduleItemId (string) (required). stateStatus (string) (required).
	<p>StateScheduleItemStatusResponse Contains values as follows:</p> <ul style="list-style-type: none"> ScheduleItemStateStatusChangeResult.
	<p>ScheduleItemStateStatusChangeResult Contains values as follows:</p> <ul style="list-style-type: none"> scheduleItemId (string). result (string) — current value of state status.

Update Filing Services (cont'd)	
setDeemerDate	Change the Deemer Date on the Filing in SERFF.
Input	A setDeemerDateRequest object.
Returns	A setDeemerDateResponse object.
Conditions	<p>A ClientException is generated if the following conditions are not met:</p> <ul style="list-style-type: none"> • SerffTrackingNumber string must reference a unique Filing.
Results	<ul style="list-style-type: none"> • Deemer Date is set to the input value. • The SERFF Log document for a Filing is updated.
Object Detail	<p>setDeemerDate Contains values as follows:</p> <ul style="list-style-type: none"> • serffTrackingNumber (string) (required). • deemerDate (dateTime) (required).
	<p>StateStatusResponse Contains one string value: "SUCCESS".</p>

Aggregate PDF Services	
RequestPdf	Creates a single Pdf file of the entire Filing and all compatible attachments.
Input	A requestPdf object.
Returns	A PdfResponse object.
Conditions	<p>A ClientException is generated if the following conditions are not met:</p> <ul style="list-style-type: none"> • The SerffTrackingNumber string must reference a unique Filing. • The target Filing must not have a draft status. • Attachments must be in PDF format and must not be larger than 3mb. If an attachment is not in PDF format and is included in the Filing, the service creates a placeholder which includes the attachment name.
Results	An attachment_id is returned. It is used as a key to retrieve the result PDF file, by utilizing the Download Attachment Handling Services (different from what was sent out/later modified).
Object Detail	RequestPdf Contains the value as follows: <ul style="list-style-type: none"> • serffTrackingNumber (string) (required).
	PdfResponse Contains AttachmentIdentifier.
RequestCustomPdf	Creates a single Pdf file of the Filing and all compatible attachments based on the requested exclusions. If no exclusions are requested, the entire Filing is generated.
Input	A requestCustomPdf object.
Returns	A PdfResponse object.

Aggregate PDF Services (cont'd)	
RequestCustomPdf (cont'd)	
Conditions	<p>A ClientException is generated if the following conditions are not met:</p> <ul style="list-style-type: none"> • The SerffTrackingNumber string must reference a unique Filing. • The target Filing must not have a draft status. • Attachments must be in PDF format and must not be larger than 3mb. If an attachment is not in PDF format and is included in the Filing, the service creates a placeholder which includes the attachment name. • The requested exclusions must match the available exclusions.
Available Exclusions	<ul style="list-style-type: none"> • <i>ExcludeNonPublicAccessItems</i> excludes all non-public access sections of the Filing from being generated into the Pdf. • <i>ExcludeDraftItems</i> excludes all draft sections of the Filing from being generated into the Pdf. • <i>ExcludeReviewerNoteItems</i> excludes all Filing reviewer notes from being generated into the Pdf.
Results	<p>An attachment_id is returned. This is used as a key to retrieve the result PDF file with Download Attachment Handling Services.</p>
Object Detail	<p>RequestCustomPdf Contains the value as follows:</p> <ul style="list-style-type: none"> • serffTrackingNumber (string) (required). • An ExclusionRules object (optional). <ul style="list-style-type: none"> • An array of exclusionRules (string) (optional).
	<p>PdfResponse Contains AttachmentIdentifier.</p>

Aggregate PDF Services (cont'd)	
FinalizePdf	This service is exempt from the Throttle.
Input	A FinalizePdfRequest object.
Returns	A FinalizePdfResponse object.
Conditions	AttachmentId must reference an attachment.
Results	When the PDF is downloaded, a temporary file is created, then destroyed.
	FinalizePdfRequest Contains values as follows: <ul style="list-style-type: none">• attachmentId (string) (required).
	FinalizePdfResponse Contains one string value which represents the status: "SUCCESS".

Handling Attachments Services	
<p>This API sends and receives attachments. To handle large attachments, the client calls a series of services.</p> <p>The following three API functions allow a client to send an attachment:</p> <ol style="list-style-type: none"> 1. beginUpload 2. writeBlock 3. endUpload 	
beginUpload	The <i>beginUpload</i> service must be called prior to uploading file data using <i>writeBlock</i> .
Input	A beginUploadRequest object.
Returns	A maxBlockSize is returned, which must be used by writeBlock.
Conditions	<p>A ClientException is generated if the following conditions are not met:</p> <ul style="list-style-type: none"> • To obtain an <i>attachmentId</i>, attachments must be uploaded by utilizing one of the web services in the SERFF State API. • <i>attachmentId</i> must reference a valid attachment id. • An <i>attachmentId</i> is obtained through the results of one of the web services, in the SERFF State API.
Results	<ul style="list-style-type: none"> • The system creates a temporary file to hold an attachment. • The system is prepared to receive files a block-at-a-time through writeBlock.
Object Detail	beginUploadRequest Contains values as follows: <ul style="list-style-type: none"> • attachmentId (string) (required). • size (integer) (required) (byte size of file).
	beginUploadResponse Contains one integer value: maxBlockSize, which represents the maximum block size, in bytes, sent by writeBlock.
writeBlock	This service supports overwriting data; the same block can be sent again, should a failure occur. <i>writeBlock</i> is available until endUpload is used on the attachment.
Input	A writeBlockRequest object.

Handling Attachments Services (cont'd)	
writeBlock (cont'd)	
Returns	A writeBlockResponse object.
Conditions	<p>ClientException is generated, if the following conditions are <i>Not</i> met:</p> <ul style="list-style-type: none"> • <i>attachmentId</i> must reference a valid attachment id • beginUpload must be called, prior to writeBlock. • The size of the base 64 decoded data must be less than the <i>maxBlockSize</i> obtained from beginUploadResponse. • The sum of the file offset and the size of the decoded data must not be greater than the file size specified in beginUpload.
Results	The system stores the data in a temporary file at the offset given by the input parameter. The uploaded block MD5 checksum is returned.
Object Detail	<p>writeBlock Contains values as follows:</p> <ul style="list-style-type: none"> • attachmentId (string) (required). • offset (integer) (bytes) (required). • data (hexBinary) (required).
	<p>writeBlockResponse Contains one string value, which indicates the MD5 checksum of uploaded block.</p>
endUpload	<p>This service indicates that the upload of an attachment, as shown by the <i>attachmentId</i> is complete.</p> <p>Until finalizeReport is called, the Upload Attachment Handling Services can be called again, since the report is still pending.</p> <p>Once endUpload is called, individual blocks are no longer available. All blocks must be re-uploaded.</p>
Input	An endUploadRequest object.
Returns	An endUploadResponse object.

Handling Attachments Services (cont'd)	
endUpload (cont'd)	
Conditions	A Client Exception is generated, if the following conditions are not met: <ul style="list-style-type: none">• attachmentId must reference a valid attachment id.• writeBlock must be called at least once.
Results	<ul style="list-style-type: none">• The attachment is created.• An MD5 checksum of the file is returned.
Object Detail	endUpload <ul style="list-style-type: none">• attachmentId (string) (required).
	endUploadResponse <ul style="list-style-type: none">• Contains one string value, <i>md5</i>, which indicates MD5 checksum of the attachment.

Download Attachment Handling Services	
The Download Attachment Handling web service facilitates downloads with large file attachments.	
beginDownload	<p><i>beginDownload</i> must be called prior to downloading file data using <i>readBlock</i>.</p> <p>This service enables the user to:</p> <ul style="list-style-type: none"> Download a temporary file one block-at-a-time. Re-read those portions that fail the <i>MD5 checksum</i>.
Input	A <i>beginDownloadRequest</i> object.
Returns	A <i>beginDownloadResponse</i> object.
Conditions	<p>A <i>ClientException</i> is generated, if the following conditions are not met:</p> <ul style="list-style-type: none"> Attachments must be downloaded, by using one of the web services in the SERFF State API. <i>attachmentId</i> must be obtained through the results of one of the web services, in the SERFF State API. <i>attachmentId</i> must reference a valid attachment id. A <i>maxBlockSize</i> must be specified. This number must indicate the maximum block size that is acceptable for download. A value of zero indicates use of the server's default block size.
Results	<ul style="list-style-type: none"> The requested resource is located and prepared for download. A <i>blockSize</i> is returned; this indicates the maximum block size the server sends when using <i>readBlock</i>. A <i>fileSize</i> is returned; this indicates the total attachment file size to be downloaded.
Object Detail	beginDownload <ul style="list-style-type: none"> <i>attachmentId</i> (string) (required). <i>maxBlockSize</i> (integer) (required) (bytes).
	beginDownloadResponse <ul style="list-style-type: none"> Contains two integer values: <i>filesize</i> and <i>blockSize</i>, both in bytes.

Download Attachment Handling Services (cont'd)	
readBlock	The <i>readBlock</i> service supports re-reading the data. This ensures the same block can be read again, should a failure occur. <i>readBlock</i> continues to be available until <i>endDownload</i> is used on the attachment.
Input	A readBlockRequest object.
Returns	A readBlockResponse object
Conditions	A ClientException is generated, if the following conditions are not met: <ul style="list-style-type: none"> • attachmentId must reference a valid attachment id. • BeginDownload must be called prior to readBlock. • The size of the base 64 decoded data must be less than the maxBlockSize obtained from beginDownloadResponse. • The sum of the file offset and the size of the decoded data must not be greater than the file size specified in beginDownloadResponse.
Object Detail	readBlock Contains values as follows: <ul style="list-style-type: none"> • attachmentId (string) (required). • offset (integer) (required) (bytes). • blockSize (integer).
	readBlockResponse Contains values as follows: <ul style="list-style-type: none"> • data (hexBinary). • md5 (string).
endDownload	This service removes all temporary resources required for the download.
Input	An endDownloadRequest object.
Returns	An endDownloadResponse object.
Conditions	A ClientException is generated, if the following conditions are not met: <ul style="list-style-type: none"> • <i>attachmentId</i> must reference a valid attachment id. • <i>beginDownload</i> must be called, prior to <i>endDownload</i>.
Results	The system removes all temporary resources created by <i>beginDownload</i> .

Download Attachment Handling Services (cont'd)	
endDownload (cont'd)	
Object Detail	endDownload <ul style="list-style-type: none">attachmentId (string) (required).
	endDownloadResponse <ul style="list-style-type: none">Contains one boolean value which indicates a successful call.

APPENDIX A: QUERY CONDITION

Building a Condition:

A condition is analogous to a single, simple SQL WHERE clause. A condition primarily uses SQL syntax.

Condition Rules:

- Conditions do not support selects.
- Conditions do not support joins.
- Conditions do not support nesting.
- Conditions do not support math expressions.
- String literals must be enclosed in apostrophes (')
- String literals are case sensitive.
- If a string literal contains an apostrophe ('), it is escaped by preceding it with another apostrophe ('). For example: 'Gilligan''s Island'.
- If a LIKE or NOT LIKE clause contains a literal with one of the supported SQL wildcard characters (% or _), or the wildcard escape character (~), it must be escaped by preceding it with the wildcard escape character (~). For example: LIKE 'Average ~% Of Sales'. Without escaping these characters, they will be treated as wild cards and not literals.
- Multiple values can be separated with a comma (,).
- Date operators are not SQL-compatible and use a custom format.

Table of Supported Operators	
=	TEXTFIELD = 'FieldValue' NUMBERFIELD = 4
<>	TEXTFIELD <> 'Not This FieldValue' NUMBERFIELD <> 3
<	NUMBERFIELD < 10
>	NUMBERFIELD > 2
<=	NUMBERFIELD <= 8
>=	NUMBERFIELD >= 3
% (for LIKE and NOT LIKE clauses only)	LIKE 'FieldV%e'
— (for LIKE and NOT LIKE clauses only)	NOT LIKE 'FieldVal_e'
AND	TEXTFIELD = 'FieldValue' AND NUMBERFIELD <> 4
BETWEEN	NUMBERFIELD BETWEEN 3 AND 5
IN	TEXTFIELD IN (3, 4, 5)
LIKE	TEXTFIELD LIKE 'FieldV%e'
NOT BETWEEN	NUMBERFIELD NOT BETWEEN 2 AND 3
NOT IN	TEXTFIELD NOT IN (2, 9)
NOT LIKE	TEXTFIELD NOT LIKE 'FieldV_e'

Dates

Dates comprise a formatted string literal preceeded by the DATE keyword. The string literal requires a date, but a time component is optional.

Formats are:

```
YYYY-MM-DD  
YYYY-MM-DD HH:MM:SS
```

For example, July 20, 1969 in the condition would be:

```
DATE '1969-07-20'
```

July 20, 1969 at 3:17:40 p.m. CDT would be:

```
DATE '1969-07-20 15:17:40'
```

Note: All times are in 24-hour format. Time zone cannot be specified. All dates and times are compared without adjusting for time zone.

PPACA Field Not Fully Supported

Due to implementation details beyond the scope of this document, the State API (SAPI) Web Services do not support using the PPACA field in the criteria of a query filing condition (e.g., PPACA = "Not PPACA Related).

The State API Web Services personnel do not believe this is an issue for SAPI users, but it is something to be aware of—when making decisions about a SAPI implementation.

Please send all questions regarding this limitation to: wsreq@naic.org.

HFAI Public Comments

If the user is enabled as an HFAI state with Public Comments enabled, it is possible that a public user entered comments on the public filings through the HFAI web application.

Appendix B: Client Exception Codes illustrates the HFAI Comments tab displayed in SERFF, for states with the above settings (as noted in the HFAI Public Comments).

APPENDIX B: CLIENT EXCEPTION CODES

Code	Description
1000	Undefined.
1001	Required fields missing.
1002	Invalid parameter.
1003	The query exceeded the maximum number of rows.
1004	No objects were found.
1005	The fields contain no data.
1006	Call rate limit exceeded.
1007	Parsing error.
1008	Invalid field for query.
1009	Fields to return must be provided.
1010	Filing not found for SERFF Tracking Number.
1011	Company represented by cocode not found for SERFF Tracking Number.
1012	Schedule Item represented by Reference ID not found for SERFF Tracking Number.
1013	Cannot create a Disposition for the following reason:
1014	State API User and Reviewer must be a STATE User Account Type.
1015	Unable to obtain the SAPI User or Instance not found for SAPI User or Reviewer.
1016	Unable to obtain the new Primary Reviewer Account.
1017	Reviewer Name does not exist.
1018	The Reviewer name provided is not associated with the filing instance.
1019	Correspondence item not found for id: _____.
1020	Cannot create an Objection Letter for the following reason:
1021	Cannot create a Note to Filer for the following reason:
1022	Cannot submit a Disposition for the following reason:
1023	Cannot submit an Objection Letter for the following reason:
1024	Cannot submit a Note to Filer for the following reason:
1025	Cannot update the reviewers for the following reason:
1026	Cannot update correspondence item for the following reason:

APPENDIX C: SERVER EXCEPTION CODES

Code	Description
2000	Undefined.
2001	Internal database error.
2002	Internal application error.
2003	Missing resource.
2004	Malformed attachment id.
2005	Filing not found.
2006	General error encountered.
2007	IO error occurred.
2008	Client key is null.
2009	Null pointer exception.
2011	Remote PDF Conversion Failure.

APPENDIX D: DOCUMENT REVISIONS LIST

02/24/2011 – Updated Copyright Information.
08/09/2011 – Clarified meaning of QueryRowLimitException.
12/08/2014 – Added information regarding Objections on Objection Letters
04/13/2015 – Updated Copyright Information.

APPENDIX E: ACRONYMS

Acronyms	Descriptions
API	Application Programming Interface
NAIC	National Association of Insurance Commissioners
PDF	Portable Document Format
PPACA	Patient Protection and Affordable Care Act
SAPI	State API
SERFF	System for Electronic Rate and Form Filing
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
TLS	Transport Layer Security