

[](http://www.google.com/url?sa=i&rct=j&q=&esrc=s&frm=1&source=images&cd=&cad=rja&docid=owCr1QpR6RHqlM&tbnid=Nzjz_YVq0miSxM:&ved=0CAUQjRw&url=http://blogs.archives.gov/aotus/?attachment_id=1236&ei=51H6Ub_GHdO64APu34H4Aw&bvm=bv.50165853,d.dmg&psig=AFQjCNEMljwv59HAB11EckpVc6CXxxOUkA&ust=1375445858498456)

***National Archives & Records Administration***

***OPA Public API Reference Guide***

***Version 1.4***

***Contract Reference GS-35F-054IU***

***April 22, 2014***

Contents

[1 Overview 2](#_Toc384766125)

[1.1 OPA Content 2](#_Toc384766126)

[1.2 API Philosophy 3](#_Toc384766127)

[1.3 Identifiers in OPA 4](#_Toc384766128)

[1.3.1 The NAID 4](#_Toc384766129)

[1.3.2 Other Identifiers 4](#_Toc384766131)

[1.4 API Responses 4](#_Toc384766132)

[1.4.1 API Response Formats 4](#_Toc384766133)

[1.4.2 General Response Structure 5](#_Toc384766134)

[1.5 Loggin In with Username/Password 7](#_Toc384766136)

[1.5.1 Login 7](#_Toc384766137)

[1.5.2 Using OPA Credentials 8](#_Toc384766138)

[1.6 Registration 8](#_Toc384766139)

[2 OPA API Parameters 11](#_Toc384766140)

[2.1 Parameters 11](#_Toc384766141)

[2.2 Error Responses 16](#_Toc384766142)

[3 Searching 19](#_Toc384766143)

[3.1 Search Results Entries 19](#_Toc384766144)

[3.2 Searching Public Contributions 20](#_Toc384766145)

[3.3 Search Functions 21](#_Toc384766146)

[3.4 Example Searches 22](#_Toc384766147)

[3.5 Query Expressions 23](#_Toc384766148)

[3.5.1 Keyword Searching 23](#_Toc384766149)

[3.5.2 Date and Date-Time Fields 25](#_Toc384766150)

[3.5.3 Integer Fields 26](#_Toc384766151)

[3.5.4 Machine Tags 27](#_Toc384766152)

[3.6 Search Fields 27](#_Toc384766153)

[3.6.1 Top-Level Fields 27](#_Toc384766154)

[3.6.2 Nested Fields for Sort or Range Searching 29](#_Toc384766155)

[3.7 Search Results Output 30](#_Toc384766156)

[3.7.1 Header Information 31](#_Toc384766157)

[3.7.2 Search Results 32](#_Toc384766158)

[3.7.3 Web Page Results 35](#_Toc384766159)

[3.7.4 Complete Example 36](#_Toc384766160)

[4 Creating or Updating Public Contributions 43](#_Toc384766161)

[4.1 Requires login credentials 43](#_Toc384766162)

[4.2 Selecting What to Annotate 43](#_Toc384766163)

[4.2.1 Selecting Comments to Modify 43](#_Toc384766164)

[4.2.2 Selecting Translations to Modify 43](#_Toc384766165)

[4.3 Specifying the Contribution Modification 43](#_Toc384766166)

[4.3.1 Comments 44](#_Toc384766167)

[4.3.2 Tags 44](#_Toc384766168)

[4.3.3 Transcriptions 45](#_Toc384766169)

[4.3.4 Translations 45](#_Toc384766170)

[5 Exports 46](#_Toc384766171)

[5.1 Examples 46](#_Toc384766172)

[5.2 Response (standard export) 47](#_Toc384766173)

[5.3 Bulk Exports 47](#_Toc384766174)

[5.3.1 Fetching Status on a Bulk Export 48](#_Toc384766175)

[5.3.2 Downloading a Bulk Export File 50](#_Toc384766176)

[6 Objects Metadata 51](#_Toc384766177)

[6.1 Metadata Description 51](#_Toc384766178)

[6.2 Example 53](#_Toc384766179)

[7 Public Contributions Metadata 60](#_Toc384766180)

[7.1 Comments 60](#_Toc384766181)

[7.2 Tags 62](#_Toc384766182)

[7.3 Transcriptions 63](#_Toc384766183)

[7.4 Translations 65](#_Toc384766184)

[7.5 Complete Example 66](#_Toc384766185)

[8 Bulk Import 71](#_Toc384766186)

[8.1 Bulk Import Comments 71](#_Toc384766187)

[8.2 Bulk Import Tags 71](#_Toc384766188)

[8.3 Bulk Import Transcriptions 72](#_Toc384766189)

[8.4 Bulk Import Translations 73](#_Toc384766190)

[9 Appendices 74](#_Toc384766191)

[9.1 Location IDs 74](#_Toc384766192)

[9.2 Technical Metadata 75](#_Toc384766193)

[9.3 Languages 75](#_Toc384766194)

Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Reviewer** | **Summary Description** |
| 0.1 | 2014-03-11 | Paul Nelson | Complete Initial Draft |
| 1.2 | 2014-03-15 | Paul Nelson | Rebrand as the “OP Public API Reference Guide”. Incorporate OPA Prod Design Document feedback received on 3/12. |
| 1.2.1 | 2014-03-22 | Paul Nelson | Corrections to section 4. |
| 1.3 | 2014-04-07 | Paul Nelson | Remove the OPA\_ID from all documents. Change all instances of @id to @commentId, @user, or @replyId to be compatible with JSON-LD in the future. Change instances of @language to @langName to be compatible with JSON-LD in the future. Change instances of @type to @resultType to be compatible with JSON-LD in the future.  Change URLs in section 4 to be RESTful to the Whitehouse API guidelines as specified by the NARA SE team.  Add lists of available range search and sort fields.  Modifications to simplify export parameters (export.what instead of type and options).  Rewrite examples to DAS XML instead of ARC XML.  Added machine tags. |
| 1.4 | 2014-04-22 | Paul Nelson | Answered questions from April 14, 2014 DCRF. Added back search parameters for annotation. Added technical metadata for all types. Added CSV example. Other minor corrections. |

# Overview

This document is the reference guide for the OPA APIs.

Specifically, this document will cover APIs to access records, digital objects, and public contributions from the OPA system. In addition, APIs for updating public contributions are provided.

## OPA Content

Content in OPA is a rich mixture of many types of information:

* **Holdings** – contain descriptive metadata about the holdings plus objects, public contributions, etc.
  + Example metadata includes title, dates covered, creator, author, etc.
  + In OPA, holdings will describe both on-line content as well as physical materials which can only be viewed by visiting one of the Archive’s facilities.
* **Digital Objects** – are digital copies of the on-line holdings which can be downloaded and viewed through OPA.
* **Authority Records** – represent people, places, locations (i.e. geographic subjects), topics (i.e. topical subjects) and organizations.
  + The primary purpose of authority records is to link archival holdings to famous people, organizations, locations, topics, etc. in a uniform fashion.
* **Public Contributions** – are for information added to NARA holdings by users of OPA. This can include:
  + *Transcriptions* – Transcribed text from scanned documents, audio recordings, or video recordings.
  + *Translations* – Translated versions of the holdings.
  + *Comments* – Discussions about the holdings.
  + *Tags* – User generated tags to aid in the search and categorization of the holdings.
* **Web Pages** – OPA will crawl the archives.gov web site and all of the presidential web sites. All web pages crawled by OPA will be included in the OPA search engine index.

The information in OPA falls into several different categories. The point of identifying these categories is to note that not all OPA information is appropriate for all API calls.

* *Searchable Information*: includes all things which are indexed into the OPA search engine and which can be retrieved by searches. This includes:
  + *Holdings*
  + *Authority Records*
  + *Digital Objects* (scanned images, videos, audio files, etc.)
  + *Web Pages*

The content and fields of these “searchables” will be indexed when available.

* *Contributeable Information*: includes everything which can be contributed to by a user.
  + Public contributions consist of tags, comments, translations, and transcriptions.
  + Not everything can have the same public contributions:
    - *Holdings* – Tags and comments
    - *Authority Records* – Tags and comments
    - *Digital Objects* (scanned images, videos, audio files, etc.) – Tags, comments, transcriptions, translations
    - *Web Pages* – Web pages cannot be contributed to via OPA.

## API Philosophy

OPA APIs are parameter-based and support the GET, POST, PUT, and DELETE methods:

* GET for performing searches and retrieving metadata.
* POST is used when creating new things (e.g. creating new public contributions)
* PUT is used when updating something (e.g. saving public contributions)
* DELETE is used when removing things (e.g. deleting public contributions)

The proper HTTP Request method to use for every function will be identified in the documentation.

When Using POST

In the documentation below, this documentation will specify example URLs as follows:

POST https://research.archives.gov/api/v1/id/5278823/objects/1223/transcription?text=George McGovern, U.S. Senate, Washington DC, Mr. President: It was an honor for me to meet you…

Note, however, when using POST, it is expected that all of the content after the “?” will be transmitted as request data (i.e. not in the URL itself, like it would be for a simple GET).

HTTPS Required

All accesses to the API use HTTPs.

## Identifiers in OPA

### The NAID

The primary identifier for OPA will be the “National Archives ID” (NAID), which is used to access all content.

All holdings and authority records will have an NAID, and all content (digital objects, public contributions, data files, etc.) will be grouped under the appropriate NAID.

As an example, the “*Letter from George McGovern to Harry S. Truman, 08/19/1972*” has the NAID 7226539. This means that the descriptive metadata for this item can be retrieved using:

GET https://research.archives.gov/api/v1?naIds=7226539

### Other Identifiers

Most everything in NARA will also have an identifier, which will be used to fetch the content.

* *Authority Record ID* – There are five different types of authority records, “people”, “organizations”, “topical subjects”, “geographic subjects”, and “special records types”:
  + Every authority record which represents a person will have a person ID
  + Every authority record which represents an organization will have an organization ID
* *Object IDs* – For on-line content, identifies individual digital objects within an NAID.
  + Object IDs will be the stable ID from DAS or the ID from the SEIP.
* *Comment IDs* – Every comment and reply will have an ID.
* *Tag* – The tag text will serve as the unique identifier for a tag.

## API Responses

### API Response Formats

API Search results, the content detail page (also known as full result display), information fetched from OPA, and responses are available either as XML or JSON. The parameter format is used to specify which type:

&format=JSON

&format=XML

If not specified, JSON is the default.

Pretty Printing the Response

Add the special parameter:

&pretty=true

To pretty print the output with correct indentation, tags aligned, etc. The default is pretty=false to save network bandwidth on communications.

*Note*: All of the examples in this document show pretty printed output even though the pretty=true parameter is not shown. This is to make this documentation more readable.

### General Response Structure

This section discusses the standard response structure returned by OPA.

OPA will return the standard HTTP response code (e.g. 200, 400, 404, etc.) in the response headers *and* will also return more detail on the response in the response data.

#### Response Codes

All APIs will return the following standard HTTP response codes:

* 200 – OK
* 400 – Bad Request (*bad parameters specified*)
* 401 – Unauthorized (*attempted to perform a function without logging in first, OR attempted to modify someone else’s data*)
* 404 – Not Found (*when the URL path is poorly formed, or when one of the identifiers specified in the URL is not found*)
* 500 – OPA internal server error

#### Response Data

Some API requests will simply return the content directly. For example, GET’ing a movie will simply stream the bytes of the movie back as the API response data.

But API methods which do something (save a transcription, for example) or list things (e.g. search) will return an <opa-response> tag with the following sub-elements:

* **header** – Holds information about the request, including:
  + **@status** – The HTTP status code
  + **@duration** – The time the system spent handling the request (in ms).
  + **request** – Request information including the following sub-elements:
    - **@path** – The path requested
    - Request parameters
      * These will be name/value pairs, one for every parameter included in the request.
      * Note that some parameters will be purposefully removed (such as password), or any parameter which is not a legal XML tag.
  + **error** – Additional error information (if an error occurred), including the following sub-elements
    - **@code** – The OPA error code.
      * This will be a string such as “INVALID\_LIST” which can be used to construct an appropriate error message to the end user.
      * A list of error codes returned by each call is listed in the documentation.
    - Error parameters
      * These will be name/value pairs which provide additional data on the error which occurred.
      * Note: Should any parameters not be legal characters for XML tags, they will be removed from the output.
* Results section [optional]
  + What appears in this section varies based upon the API request. Some requests will not have results, only status.
  + If there was an error, this section will not be present.

Outline

The following is a general outline of the header returned by most OPA API calls:

<opa-response>

<header status="*HTTP Status*" duration="Time in ms">

<request path="URL-path">

<PARAM-NAME1>param-value1</PARAM-NAME1>

<PARAM-NAME2>param-value2</PARAM-NAME2>

<PARAM-NAME3>param-value3</PARAM-NAME3>

</request>

<errors>

<error code="ERROR-CODE">

<PARAM-NAME1>param-value1</PARAM-NAME1>

<PARAM-NAME2>param-value2</PARAM-NAME2>

<PARAM-NAME3>param-value3</PARAM-NAME3>

</error>

</errors>

</header>

.

. Results specific tags as defined later in this document goes here

.

</opa-response>

XML Example

<opa-response>

<header status="200" duration="132">

<request path="/api/v1">

<q>kennedy</q>

<item.title>rose</item.title>

</request>

</header>

.

. The search results would be returned here

.

</opa-response>

JSON Example

{

"header":{

"@status":"200",

"@duration":"132",

"request":{

"@path":"/api/v1",

"q":"kennedy",

"description.title":"rose"

}

},

.

. The search results would be returned here

.

}

## Loggin In with Username/Password

All registered OPA users will be able to log in directly to OPA with a username and password. This will only be required for functions which require updates to public contributions.

### Login

Use the following URL to log in:

POST https://research.archives.gov/api/v1/login?user={user-id}&password={password}

Note:

* The credentials will be returned in the login response (in a nested <credentials> tag).
* For security reasons, the password parameter will not be returned in the response.

The example login responses below only show whether or not the login was successful. A “400” (BAD REQUEST) error will be returned if login failed.

XML Example Login Response

<opa-response>

<header status="200" time="132"/>

<request path="/api/v1/login">

<user>tjefferson</user>

</request>

</header>

<credentials>7F8A29ECAA4326AC81329AFBB26649ACE</credentials>

</opa-response>

JSON Example Login Response

{

"header": {

"@status":"200",

"@duration":"132",

"request": {

"@path":"/api/v1/login",

"user":"tjefferson"

}

},

"credentials":"7F8A29ECAA4326AC81329AFBB26649ACE"

}

### Using OPA Credentials

To use the OPA Credentials, specify the credentials “Authorization:” HTTP header.

For example:

Authorization: OPA:7F8A29ECAA4326AC81329AFBB26649ACE

## Registration

Use the following API call to register a new user:

POST https://research.archives.gov/api/v1/login?action=register&*parameters...*

Note that only users who are logged in (see previous section) are allowed to register new users, and only those users with sufficient rights.

Parameters

The following parameters are available when registering a new user:

| **Parameter** | | **Description** | **Examples** |
| --- | --- | --- | --- |
| For use when action=change | | | |
| **user=** *user-id* | [REQUIRED] Specify the user ID for the new user. | | user=kmartin |
| **password=** *password* | [REQUIRED] Specify the new password. The password must contain a minimum of 8 characters. | | password=wxyz9876 |
| **email=***email* | [REQUIRED] Specify the new e-mail address for the user. | | email=jdoe@ficticious.com |
| **fullName=***full-name* | [OPTIONAL] Specify the new full name for the user. | | fullName=Thomas%20Jefferson |
| **displayFullName=** *true|false* | [OPTIONAL] Set to “true” if the full name for the user should be displayed to the public, “false” otherwise. | | displayFullName=true |
| **returnUrl=** *<Return-URL>* | [OPTIONAL] Specify the return URL | | returnUrl=https://myserver.com/ opa-newly-activated.html |
| **returnText=** *<Return-Text>* | [OPTIONAL] Specifies the link text to be used for the return URL. | | returnText=Click here to return to The Military Records Transcription by Veterans Project. |

Response

Upon successful completion, the user will be sent an E-mail with an OPA URL where they can activate their account. Once the user has clicked on the OPA URL, they will be taken to OPA, will be asked if they agree with various government warnings and notifications, and they can activate their account.

Once the account is activated, the user will be presented with the specified return URL and it will be labeled with the specified return text. The user, once their account is activated, can click on the URL and they will be taken back to your web site.

The response will the standard OPA response (see section 1.4.2 for more information), with an additional message to be displayed to the user in the “message” tag. Note that the password is removed from the registration response.

XML Example Registration Response

POST https://research.archives.gov/api/v1/login?action=register&user=tjefferson&password=abcdef&  
email=tjefferson@ficticious-server.com&returnUrl=http%3A%2F%2Fficticious-server.com%2F  
opa-user‑verified.html&returnText=Return%20to%20Ficticious%20Server%20to%20start%20  
your%20tagging%20project!%0A&format=xml

<opa-response>

<header status="200" time="132">

<request path="/api/v1/login">

<user>tjefferson</user>

<email>tjefferson@ficticious-server.com</email>

<returnUrl>http://ficticious-server.com/opa-user-verified.html</returnUrl>

<returnText>Return to Ficticious Server to start your tagging project!</returnText>

</request>

</header>

<message><![CDATA[An email verification has been sent to the following email address.

- tjefferson@ficticious-server.com. Please check your email to complete the registration process. Didn't get the email.? <a href="http://research.archives.gov/registration/login?action=resend‑email&user=tjefferson">Resend the Verification email.</a>]]>

</message>

</opa-response>

JSON Example Registration Response

POST https://research.archives.gov/api/v1/login?action=register&user=tjefferson&password=abcdef&  
email=tjefferson@ficticious-server.com&returnUrl=http%3A%2F%2Fficticious-server.com%2F  
opa-user-verified.html&returnText=Return%20to%20Ficticious%20Server%20to%20start%20  
your%20tagging%20project!%0A

{

"header":{

"@status":"200",

"@time":"132",

"request":{

"@path":"/api/v1/login",

"user":"tjefferson",

"email":"tjefferson@ficticious-server.com",

"returnUrl":"http://ficticious-server.com/opa-user-verified.html",

"returnText":"Return to Ficticious Server to start your tagging project!"

}

},

"message":"An email verification has been sent to the following email address. - tjefferson@ficticious-server.com. Please check your email to complete the registration process. Didn't get the email.? <a href=\"http://research.archives.gov/registration/login?action=resend‑email&user=tjefferson\">Resend the Verification email.</a>"

}

Error Codes

If the registration parameters are incorrect, then a standard error header will be returned (see section 1.4.2) with one of the following error codes:

|  |  |  |
| --- | --- | --- |
| Error Code | Description | Parameters |
| BAD\_PASSWORD | The password provided does not match the user’s current password. | none |
| MISSING\_PARAM | One of the required parameters is missing. | *param*: The missing parameter. |
| INSUFFICIENT\_RIGHTS | The user attempting to register users has insufficient rights to do so. |  |
| ACCOUNT\_NOT\_YET\_ VALIDATED | The user has attempted to log into the account with a correct password, but the account has not yet been validated via E-mail. |  |

## Rate Limiting

The OPA Public APIs will be rate limited to ensure that sufficient resources are available to all users and to prevent denial of service attacks.

The various rate limits implemented by the OPA Public API reference guide are as follows:

* Total IP accesses per second
  + Limits:
    - Anonymous API users (without login) – 60 requests every 600 seconds
    - Logged-In Users – 600 requests every 600 seconds
  + If the limit is reached for an IP address
    - A “500” HTTP error is returned for all requests
    - After 600 seconds, the block is removed.
* Total contributions per day
  + Contributions are only allowed for logged-in users.
  + Limits:
    - Standard Users – 1,000 contributions per day.
    - “Power” Users – 10,000 contributions per day.
  + If the limit is reached for total contributions per day:
    - Returns a 400 error with a standard error response with the “CONTRIBUTION\_LIMIT\_REACHED” error code.
* Number of simultaneously locked files:
  + Limits:
    - Standard Users – 10 simultaneously locked files
    - “Power” Users – 50 simultaneously locked files
  + If the limit is reached for maximum number of simultaneously locked files:
    - Returns a 400 error with a standard error response with the “LOCK\_LIMIT\_REACHED” error code.

Error Codes

The following are the error codes for total contributions per day and number of simultaneously locked files. These error codes are returned as part of a standard error header (see section 1.4.2):

|  |  |  |
| --- | --- | --- |
| Error Code | Description | Parameters |
| CONTRIBUTION\_LIMIT\_ REACHED | The total number of contributions per day has been reached by the specified user. | *param*: The user name. |
| LOCK\_LIMIT\_REACHED | The total number of simultaneously locked files allowed for the user has been reached. | *param*: The user name. |

# OPA API Parameters

In addition to results, the OPA API can also do the following:

* **Save results for bulk export** – This allows you to bulk export metadata and content from OPA into compressed files (tar.gz) which can be efficiently downloaded.
  + Bulk exports are processed in the background.
* **Add or Modify Public Contributions** – Public contributions can be added, updated, or deleted.

This section provides a complete list of all OPA API parameters and Error Responses.

GET https://research.archives.gov/api/v1?*parameters. . .*

## Parameters

| **Parameter** | **Description** | **Examples** |
| --- | --- | --- |
| High-Level Parameters | | |
| **resultTypes**=*type1, type2,type3, . . .* | Specifies a list of result types to select. Types can be any of:   * holding – All NARA holdings, includes record groups, collections, file units, series, and items. * recordGroup – Record group archival records. * collection – Collection archival records.series – Series archival records. * fileUnit – File unit archival records * item – Item archival records. * object – Digital objects (files) * authority – All authority records, includes person, organization, topicalSubject, geographicSubject, and specialRecordsType * person – People authority records * organization – Organization authority records * topicalSubject – Topical subject authority records * geographicSubject – Geographic subject authority records * specialRecordsType – Special records types authority records * archivesWeb – Web pages from archives.gov. * presidentialWeb – Web pages from presidential libraries.   If no “type” parameter is specified, OPA will select all types. | resultTypes=holding  resultTypes=holding, object,archivesWeb |
| **naIds=***naId1, naId2, naId3 . . .* | Specifies a list of NAIDs to select. | naIds=7238423,536932, 4562889,3552324 |
| **objectIds=***objectId1, objectid2, objectid3* | Specifies an individual object IDs to select. | objectId=183254, 2932423 |
| Search Parameters  *See section 3 below for more details.* | | |
| **q**=*query expression* | Keyword search over the full-text content (and selected full-text fields such as title).  Full query expressions can be used as the value for ‘q’. See section 3.5 for more details.  The ‘q’ parameter is optional. When missing, the search results will be based entirely on the filter parameters below.  The exact list of fields which will be searched by the ‘q’ parameter will be different for each different type of content source, but usually includes embedded text content plus high-quality full text fields such as ‘title’, ‘description’, etc.  See the appropriate Data Model Definition (DMD) document for each source to identify exactly what fields are searched by ‘q’. | q=Navy  q=Kennedy and NASA  q=kennedy missile crisis  q= johnson and space and (nasa or "space agency") |
| ***{field}****= query expression* | Further, any individual field can be used for keyword searches.  The query expression can be any expression specified in section 3.5.1.  Note that “field” in this context can include sub-fields as well (such as “description.title”).  For keyword search fields, you may omit intermediate tags. Only the first and last tags are required. For example:   * description.variant-number=   + Searches all <variant-number> tags in the description. * description.creator.creator-type=   + Searches all <creator-type> tags found within <creator> tags in the description | description=JFK  description.title=campaign and (JFK or LBJ) |
| ***{dateField}***= *date expression* | *dateField* is used to restrict results based on a date or date range. Note that only select fields can be used for date range expressions.  See section 3.5.2 for date expressions available to search date fields. | description.recordHistory. created.dateTime= range(1900,1999) |
| ***{intField}***= *int expression* | *intField* is used to restrict results based on an integer or integer range. Note that only select fields can be used for integer range expressions.  Integer expressions can use operators specified in section 3.5.3. | objects.object.file.technicalMetadata.size= range(1000000, 5000000) |
| ***machineTags=*** *machine tags expression* | Specify a query expression for searching over machine tags.  Machine tag expressions are specified in section 3.5.4. | machineTags=nara:person:"president Lincoln”  machineTags=\*:title:"It’s a wonderful life"  machineTags=nara:\*: (Lincoln or Washington) |
| **resultFields**=*field1, field2,field3, . . .* | List of fields to return in the result set.  If a field is given that cannot be put in the return (such as a field that does not exist), an error is returned.  Nested fields can be individually returned by specifying the tag (or sub-tag or attribute) with “field.tag.tag” notation.  ***Note: If resultFields is missing, all fields in the index entry are returned.*** | resultFields=description  resultFields=description. topicalSubjectArray. topicalSubject. termName  resultFields=description, objects |
| **sort**= *field1 [asc|desc], field2 [asc|desc], . . .* | Sorts results based on field values. Specify the field to sort by followed by “asc” for ascending or “desc” for descending.  Note that only select fields are available for sorting.  If a field name is given that cannot be sorted, an error is returned. If an invalid sort type is given, an error is returned. | sort=description.title desc  sort=description. recordHistory.created. dateTime desc |
| **rows**=*#* | Indicates the number of search results to fetch. The default is 25.  Note that different maximum values for “rows” may apply depending on what is done with the search results.   * For simple searches, “rows” is limited to 200. * For exporting bulk results, “rows” is limited to 10000. | rows=50 |
| **offset**=*#* | The zero-based offset into the search results of the first result to be returned.  This parameter is used in combination with ‘rows’ to paginate the results. If missing, the default offset is 0.  If an offset less than 0 or greater than the maximum is specified, an error is returned. | offset=100 |
| Contribution Parameters  *See section 4 for more information on creating, deleting, or modifying public contributions.* | | |
| **text=<text>** | Specify the translation text, tag text, or comment text. | text=Caro signor Truman: Si prega di accettare le nostre scusato. . . |
| Export Parameters  *See section 5 for more information on exporting.* | | |
| **action=export** | Save the search results with additional information as a bulk export file. | action=export |
| **export.what**= *metadata | thumbnails | tags | comments | transcriptions | translations* | Specifies what to export. Can be any of:   * metadata – Specify that metadata should be exported. * thumbnails *–* Thumbnails. * tags – Export tags attached to the record. * comments – All comments attached to each record. * transcriptions – Transcriptions for each digital object. * translations – Translations for each digital object. | export.what=metadata, thumbnails,tags  export.what=metadata, tags,comments |
| **export.languages=** *list of language codes* | If export.what=translations, then list the IS0 639-3 (three letter) language codes to export. | export.languages=spa,ita, deu |
| **export.languages.all=***true/false* | If export.options=translations, then use this parameter to specify that all languages should be exported. | export.languages.all=true |
| **resultFields=***list of fields…* | If “export.type=metadata”, the list of fields to export is specified with a “resultFields” parameter. See section 3.5.4 for the list of fields which can be exported. | resultFields= description.title, description.naId |
| **export.format=***xml|json|csv|txt| pdf|html* | Specify the format for the search results.   * xml – Export all metadata in XML format. * json – Export all metadata in XML format. * csv – Export metadata in CSV format. All fields specified by resultFields will be exported with one field per CSV column.   + Fields which contain complex metadata will be exported as a single string value, with nexted text content separated by the pipe (|) character. * txt – Export metadata in text format.   + Incompatible with resultFields parameter. Will return an error if specified.   + Incompatible with “export.what=thumbnails” (option will be ignored if present). * pdf – Export a PDF rendition of the results.   + Incompatible with resultFields. Will return an error if specified. * html – Export an HTML rendition of the results.   + Incompatible with resultFields parameter. Will return an error if specified.   If missing, defaults to the same format as specified by the “format” parameter. If the format parameter is also missing, defaults to JSON. | export.results.format=json |
| **export.bulk=** *true/false* | If exporting a very large number of records, records with a large number of objects, or exporting content files, then you will likely need to specify “export.bulk=true”.  When export.bulk=true, a bulk export will be queued for background processing and a bulk export ID will be returned.  See section 5.3.1 for getting the status of the bulk export and getting a download URL when the export file is complete. | export.bulk=true |
| **export.bulk.content=***objects| thumbnails* | When export.bulk=true, additional content files can also be exported. Use this parameter to specify what content files to export.   * objects – Digital object files attached to each result. * thumbnails – Thumbnail files attached to each result. | export.bulk.content= objects,thumbnails |
| **bulkExportId=***#* | Specify the bulk export ID to get information on the status of a bulk export.  *Note: Parameter must be specified alone. Can not be specified with any other parameter.* | bulkExportId=32 |

## Error Responses

The following error codes are currently defined as part of the OPA API response:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Error Code | Description | | Parameters | |
| EMBEDDED\_HTML | The query or filter parameters contain embedded HTML which implies a potential cross-site scripting attack. | | *param*: The parameter where the embedded HTML was detected. | |
| INVALID\_PARAM\_VALUE | The value specified to a parameter was invalid. For example, an incorrect source value, incorrect group type, etc. | | *param*: The parameter which contained the invalid value. | |
| INVALID\_FIELD | A request was made to search over an invalid field.  *Note: embedded fields will not be flagged as invalid. Only top-level fields will be flagged as invalid.* | | *param*: The parameter which contained the invalid field.  field: The invalid field name. | |
| INVALID\_SORT\_FIELD | A request was made to sort on a field which can be used for sorting. | | *param*: The invalid field. | |
| INVALID\_RANGE\_FIELD | A request was made to perform a range search on a field which does not allow for range searching. | | *param*: The invalid field. | |
| QUERY\_PARSE\_ERROR | An error occurred parsing the query.  *Note: This should rarely occur. Default behavior is defined for most invalid expressions.* | | *param*: The parameter which contained the invalid search expression. | |
| EXPORT\_NON\_BULK\_ ROWS\_EXCEEDED | Number of rows requested exceeded the limit for exporting records without performing a bulk export.  *Note: If this error is received, then the query should be resubmitted with export.bulk=true parameter set.* | | *request*: The number of rows requested.  *max*: The maximum number of rows allowed without performing a bulk export. | |
| OFFSET\_LIMIT\_EXCEEDED | The maximum search result request has exceeded the limit. See the “offset” parameter for more details. | | *request*: The result number requested.  *max*: The maximum result number allowed. | |
| ROWS\_LIMIT\_EXCEEDED | The maximum number of rows to be returned with the search results was reached.  See also EXPORT\_NON\_BULK\_ROWS\_EXCEEDED above. | | *request*: The number of rows requested.  *max*: The maximum number of rows allowed. | |
| QUERY\_TIMEOUT | Query execution took too long and was canceled. | | *max*: The maximum number of seconds allowed for a query to execute. | |
| INVALID\_PARAM\_VALUE | A value for a parameter is invalid. | | *param*: The parameter with the invalid value.  *value*: The invalid value. | |
| INCOMPATIBLE\_EXPORT\_ OPTIONS | The options specified are incompatible with the format specified or with the type of export requested. | | *option*: The incompatible option.  *format*: The incompatible format.  *type*: The incompatible type of export. | |
| EXTRANEOUS\_EXPORT\_ PARAM | One of the parameters is extraneous. For example, export.bulk.content when export.bulk is false. Or export.languages when translations are not exported. | | *param*: The parameter which is extraneous. | |
| NOT\_LOGGED\_IN | | Attempted to add or modify a contribution, or to create a bulk export without being logged in. | | none |
| NOT\_OWNER | | Attempted to modify someone else’s contribution. | | *user*: The user ID attempting to modify the list. |

XML Example

GET https://research.archives.gov/api/v1?action=search&q=truman&rows=10000&format=xml

<opa-response>

<header status="400" time="8">

<request path="/api/v1">

<action>search</action>

<q>truman</q>

<rows>10000</rows>

</request>

</header>

<errors>

<error code="ROW\_LIMIT\_EXCEEDED">

<request>10000</request>

<max>200</max>

</error>

</errors>

.

. search results will be returned here

.

</opa-response>

JSON Example

GET https://research.archives.gov/api/v1?action=search&q=truman&rows=10000

{

"header":{

"@status":"400",

"@duration":"8",

"request":{

"@path":"/api/v1",

"action":"search",

"q":"truman",

"rows":"10000"

}

},

"errors":{

"error":{

"@code":"ROW\_LIMIT\_EXCEEDED",

"request":"10000",

"max":"200"

}

},

.

. search results will be returned here

.

}

# Searching

With the OPA API, you specify query terms, filters, and the scope of the search as parameters, and the search engine will return results. Each result will contain metadata fields.

## Search Results Entries

The following can be returned as individual entries in the search results:

* **Holdings** – Descriptive metadata, public contributions and information on digital objects (where appropriate) about every collection, record group, series, file unit, and item available from NARA.
* **Authority Records** – Authoritative information about people, organizations, geographic subjects, topical subjects, and special records types which may be referenced by archival descriptions.
* **Digital Objects** – Files which represent the on-line content (and technical documentation about the on-line content) can be individually retrieved using the OPA API.

How Do I: Search for any type of archival description / holding?

Filter on type = “holding”:

GET https://research.archives.gov/api/v1?q=truman&resultTypes=holding

How Do I: Search for only file units?

Filter on type = “fileUnit”:

GET https://research.archives.gov/api/v1?q=truman&resultTypes=fileUnit

How Do I: Search for Digital Objects?

Filter on type = “objects”:

GET https://research.archives.gov/api/v1?q=truman&resultTypes=object

How Do I: Search for Both?

Filter on type = “holding,object”:

GET https://research.archives.gov/api/v1?q=truman&resultTypes=holding,object

How Do I: Retrieve all Digital Objects for a Description?

Filter on naIds=<ID of the description> and resultTypes=object:

GET https://research.archives.gov/api/v1?resultTypes=object&naIds=7283924

How Do I: Search for People?

Filter on type = “people”:

GET https://research.archives.gov/api/v1?q=truman&resultTypes=person

How Do I: Search for everything?

Have no filter at all:

GET https://research.archives.gov/api/v1?q=truman

## Searching Public Contributions

Individual contributions are not indexed as separate entries in the search engine index, but instead are indexed with the content (holding, authoriy record or an object) to which they are attached.

Since the search engine can return descriptions, digital objects, authority records, and web pages as search results, public contributions will be returned as fields inside each search result.

The OPA API can:

* Locate all holdings, digital objects, or authority records which contain:
  + A particular tag
  + Specified query terms in any tag or tags
  + Any tag created by a specified user ID
  + Specified query terms in any comment or reply
  + Comments or replies by a specific user ID
* Locate all digital objects which contain:
  + Specified query terms in a transcription or translation
  + Transcriptions or translations originated by a particular user ID
  + Transcriptions or translations edited by a particular user ID

The OPA API cannot:

* List all public contributions by a particular user ID
  + However, the OPA API *can* return a list of holdings, digital objects, or authority records which contain a contribution by a particular user ID.
* List all comments or replies which contain specified query words
  + However, the OPA API *can* return a list of holdings, digital objects, or authority records which contain the specified query words in their comments.
* List all tags which contain specified query words
  + However, the OPA API *can* return a list of holdings, digital objects, or authority records which contain the specified query words in their tags.
* List all unique tags
  + Tags can be returned by holding, digital object, or authority record.
  + A single list of all available tags is not anticipated for OPA production (first release)
* List all comments
  + Comments can be returned by holding, digital object, or authority record.
  + A single list of all comments is not anticipated for OPA production

## Search Functions

The OPA API provides a number of capabilities:

* High-quality relevancy ranked keyword text search of content and selected metadata fields
* Keyword text search of any public, indexed, metadata field
* Computing a ‘relevancy score’ for all search results, based on how well the document matches the query
* Boolean queries with AND, OR, and NOT
* Range searches such as locating documents with a specified date or integer range
  + *Note: Range searches will only be available on select fields.*
* Filtering results based on keywords found in metadata or content fields
* Specifying fields to return, in addition to default fields
* Sorting of search results
  + The default sort order is by relevancy in descending order
  + Sorting can be done over multiple fields as well as relevancy with the same API call (e.g., primary sort, secondary sort, etc.)
  + *Note: Sorting will only be available on select fields.*
* Search results list pagination, including returning a specified page or range of results
* Bulk export of search results (requires login)
  + May include digital objects associated with the search results as well

## Example Searches

Simple search:

GET https://research.archives.gov/api/v1?q=truman

Boolean query search:

GET https://research.archives.gov/api/v1?q=(bess or harry) and truman

Boolean query search with phrases:

GET https://research.archives.gov/api/v1?q="president truman" or "harry s truman"

Search over everything from a particular location:

GET https://research.archives.gov/api/v1?q=truman&  
description.referenceUnit="Harry S. Truman Library"

Find all titles with “truman”:

GET https://research.archives.gov/api/v1?description.title=truman

Search for “truman”, filter by inclusive-start date between 1980 and 1990:

GET https://research.archives.gov/api/v1?q=truman&  
description.inclusiveDates.inclusiveStartDate=range(1980,1990)

Search over holdings returning just the archival description XML:

GET https://research.archives.gov/api/v1?q=truman&resultTypes=holding&  
resultFields=description

Search over posters (specificRecordsType from the archival description contains “poster”):

GET https://research.archives.gov/api/v1?q=truman&resultTypes=holding&  
description.specificRecordsType.termName=poster&resultFields=description

Find all holdings tagged with “Kennedy Vietnam”

GET https://research.archives.gov/api/v1?publicContributions.tag="Kennedy Vietnam"&  
resultTypes=holding

Find all digital objects tagged with any tag that contains “kennedy”:

GET https://research.archives.gov/api/v1?publicContributions.tag=kennedy&  
resultTypes=object

Find all digital objects tagged with a “nara:person” machine tag that contains “john kennedy”:

GET https://research.archives.gov/api/v1?machineTags=nara:person:"john kennedy"&  
resultTypes=object

Find documents which contain the key word “truman” and sort by inclusive start date, descending:

GET https://research.archives.gov/api/v1?action=search&q=truman&  
sort=description.inclusiveDates.inclusiveStartDate desc

Return the first 10 results which contain the key word “truman”:

GET https://research.archives.gov/api/v1?q=truman&rows=10

Return the results 11-20 which contain the key word “truman”:

GET https://research.archives.gov/api/v1?q=truman&rows=10&offset=10

## Query Expressions

This section covers the operators which are available to construct query expressions in OPA.

### Keyword Searching

The following operators are available for searching all keyword fields:

|  |  |  |
| --- | --- | --- |
| Query Operator | Description | Examples |
| *implied and* | Any time an operator is missing, it is an *implied and*. Both of the words or sub-expressions specified must exist in the returned results. | Katyn massacre  (john or jack) kennedy |
| and | Return results that have both of the words or sub-expressions specified.  Can be upper or lower case “AND”. | katyn and massacre  (john or jack) AND kennedy |
| or | Return results that have either of the words or sub-expressions specified.  Can be upper or lower case “OR”. | kennedy or Johnson  (john kennedy) OR  (lyndon johnson) |
| not | Return results which do not have the word or sub-expression specified.  Can be upper or lower case “NOT”.  Expressions without a positive clause *are not* accepted in OPA at this time. | kennedy not Jacqueline  johnson NOT "lady bird" |
| ( *expression* ) | Parenthetical expressions are allowed for more carefully specifying the boolean logic desired. | (john kennedy) OR  (lyndon johnson)  (john or jack) AND kennedy |
| "term term term . . ." | Phrases can be expressed with double-quotes. All terms must exist in the specified order before the result is returned. | "lady bird johnson"  "world war ii" |
| *xmlField*:(expression...) | Search on keywords within an XML field.  This search will find the key word inside any tag within the XML data. | description:poster |
| *xmlField*.*tag*: (expression…) | Search on keywords within a specific XML tag.  *Note 1: Any tag at any level of nesting can be specified.*  *Note 2: The entire expression specified by (expression…) must match within the same tag. Search results will not be returned if parts of the expression match in one instance of the tag, and other parts of the expression match elsewhere.*  *Note 3: The nested expression can contain the following operators: “and”, “or”, phrases, parenthesis, and nested field: operators for XML tags nested within the parent tag specified.*  *Note that the NOT operator is* not *supported in XML fields.* | description. creatingOrganization: (emergency and management) |
| *xmlField*.*tag.tag*: (expression…) | Specify multiple tags to require nesting.  For example, with  description.referenceUnit.termName:(park)  only results with the keyword “park” inside the <name> tag inside the <reference-unit> tag will be returned. | description. referenceUnit.termName: (college park) |
| *xmlField.tag.@attribute: (terms…)* | Attributes can be specified using the “@” symbol for the XML tag. | publicContributions. comment.@user:tjefferson |

Tokenization of Keyword Fields

Text in keyword searches are automatically split into separate terms on punctuation and white space. For example, the query:

diesel-electric tank/armored-vehicle

Will be searched as:

diesel AND electric AND tank AND armored AND vehicle

Example of Complex Nested Field: Operators

XML fields support complex nested field: operators. For example, to search only those variant control numbers where the specified variant type is “Search Identifier”, you might use the following expression:

description.variantControlNumber:(type:"Search Identifier" AND   
number:(learning center))

This expression will match *only* descriptions which contain the specified <type> and <number> inside the *same* <variantControlNumber> tag. If the type and number are within two *different* <variantControlNumber> tags, then it will not be matched.

For example, given the above search expression, the following will be matched:

<variantControlNumber>

<number>si **Learning Center**</number>

<note>si Learning Center</note>

<type><termName>Search Identifier</termName></type>

</variantControlNumber>

But the following will *not*:

<variantControlNumber>

<number>si Learning Center</number>

<note>si Learning Center</note>

<type><termName>Learning Center Type</termName></type>

</variantControlNumber>

<variantControlNumber>

<number>si National History Standard Era 8</number>

<note>si National History Standard Era 8</note>

<type><termName>Search Identifier</termName></type>

</variantControlNumber>

### Date and Date-Time Fields

The following operators and options can be used to search over both date and date-time fields.

Note that date and date-time *range* searches will only work over select fields. In particular, the field must not be multi-valued.

| Query Operator | Description | Examples |
| --- | --- | --- |
| *YYYY* | Automatically search on the entire range specified by the year.  This includes all seconds from midnight January 1 to 11:59:59 on December 31, inclusive. | description.recordHistory. created.dateTime:2014 |
| *YYYY-MM* | Automatically search on the entire range specified by the year and month.  This includes all seconds from midnight on the first of the month to 11:59:59pm on the last day of the month, inclusive. | description.recordHistory. created.dateTime:2014-04 |
| *YYYY-MM-DD* | Automatically search on the entire range specified by the day, from 0:00:00 to 23:59:59. | description.recordHistory. created.dateTime: 2014-04-01 |
| *YYYY-MM-DDTHH* | *For date-time fields only*, automatically search on the entire range specified by the hour. | description.recordHistory. created.dateTime: 2014-04-01T10 |
| *YYYY-MM-DDTHH:MM* | *For date-time fields only*, automatically search on all seconds within the specified minute. | description.recordHistory. created.dateTime: 2014-04-01T10:30 |
| *YYYY-MM-DDT HH:MM:SS* | *For date-time fields only*, search just for the specified second. | description.recordHistory. created.dateTime: 2014-04-01T10:30:45 |
| range(*date1,date2*) | *For both date and date-time fields*, specify the range of values (inclusive) over which to search.  If any date value is truncated to just the Year or Year-month, the range will be expanded appropriately to cover the entire inclusive range.  For example, for range(2001, 2002) the search will include all days from January 1, 2001 to December 31, 2002, inclusive.  For range(2001-09, 2001-12), the search will include all days from the September 1, 2001 to December 31, 2001. | description.recordHistory. created.dateTime: range(2014-01-01,  2014-04-30) |
| range(*dateTime1, dateTime2*) | *For date-time fields only*, search over the specified range, inclusive.  If any date value is truncated to just the Year or Year-month, the range will be expanded appropriately to cover the entire inclusive range.  For example, for range(2001, 2002) the search will include all days from January 1, 2001 to December 31, 2002, inclusive.  For range(2001-09, 2001-12), the search will include all days from the September 1, 2001 to December 31, 2001. | description.recordHistory. created.dateTime: range(2014-04-01T05:00:00,  2014-04-01T06:59:59) |

### Integer Fields

Note that NAID and other identifiers are indexed as keyword fields by OPA and not integer fields.

Note that integer *range* searches will only work over select fields. In particular, the field must not be multi-valued.

Integer fields can be searched either for a specific number, or for a range.

| Query Operator | Description | Examples |
| --- | --- | --- |
| range(*#,#*) | Search for all numbers within the specified range (inclusive). | objects.file.technicalMetadata. size:range(0,1048575) |

### Machine Tags

The following operators are available for searching machine tags:

|  |  |  |
| --- | --- | --- |
| Query Operator | Description | Examples |
| {namespace}:{field}:{expr} | Search for a query expression within a specific field and namespace.  Note that any query expression from section 3.5.1 can be put inside of {expr}. | nara:title:"president lincoln"  nara:title:(president or congressman) |
| {namespace}:{field}:\* | Search for anything which has the specified namespace and field in a tag. | nara:title:\*  tc:date:\* |
| \*:{field}:{expr} | Search for a query expression within a specific field for any namespace.  Note that any query expression from section 3.5.1 can be put inside of {expr}. | title:"president lincoln"  title:(president or congressman) |
| \*:{field}:\* | Search for any machine tag with the specified field in any namespace. | \*:title:\*  \*:date:\* |
| {namespace}:\*:{expr} | Search for a query expression within a specific namespace across all fields.  Note that any query expression from section 3.5.1 can be put inside of {expr}. | nara:\*:"president lincoln"  nara:\*:(president or congressman) |
| {namespace}:\*:\* | Search for any instance of the specified namespace across all fields. | nara:\*:\*  tc:\*:\* |
| {expr} | Search for a query expression within any machine tag.  Note that any query expression from section 3.5.1 can be put inside of {expr}. | "president lincoln"  (president or congressman) |

Note that advanced combinations of multiple tags can be included in the same expression:

nara:title:(president or congressman) AND nara:person:(lincoln or washington)

## Search Fields

### Top-Level Fields

There are only four top-level fields defined to be returned in search results for the OPA API.

* **description** – Contains the metadata for the archival description.
  + Only available for holdings and digital objects  
    (i.e., resultTypes=holding, recordGroup, collection, series, fileUnit, item, or object).
  + For digital objects, this will hold the metadata of the archival description to which the object belongs.
  + Note that the digital object information (the <*digitalObjectArray*> tag and its contents) is not included with this XML.
* **person** – Contains the complete XML for the person authority record.
  + Only available for people authority records (i.e., resultTypes=person).
* **organization** – Contains the complete XML for the organization authority record.
  + Only available for organization authority records (i.e., resultTypes=organization).
* **topicalSubject** – Contains the complete XML for the topical subject authority record.
  + Only available for organization authority records (i.e., resultTypes=topicalSubject).
* **geographicSubject** – Contains the complete XML for the geographic subject authority record.
  + Only available for organization authority records (i.e., resultTypes=geographicSubject).
* **specificRecordsType** – Contains the complete XML for the specific records type authority record.
  + Only available for organization authority records (i.e., resultTypes=specificRecordsType).
* **objects** – Holds the complete object XML for all digital objects (if a description) OR just the one <object> tag (for digital objects).
  + See section 6 for a complete description of the contents of this field.
* **publicContributions** – [*xml*] Holds the complete object XML for all digital objects (if a description) OR just the one <object> tag (for digital objects).
  + See section 7 for a complete description of the contents of this field.

### Nested Fields for Sort or Range Searching

Sorting and range searching are only allowed on select fields, as specified below:

#### Within the “description” field

|  |  |
| --- | --- |
| Field | Features |
| description.localIdentifier | sort |
| description.naId | sort |
| description.title | sort |
| description.recordHistory.created.dateTime | sort, range-search |
| description.recordHistory.changed.modification.dateTime | sort, range-search |
| description.productionDateArray.proposableQualifiableDate.logicalDate | sort,range-search |
| description.broadcastDateArray.proposableQualifiableDate.logicalDate | sort,range-search |
| description.releaseDateArray.proposableQualifiableDate.logicalDate | sort,range-search |
| description.coverageDates.coverageStartDate.logicalDate | sort,range-search |
| description.coverageDates.coverageEndDate.logicalDate | sort,range-search |
| description.inclusiveDates.inclusiveStartDate.logicalDate | sort,range-search |
| description.inclusiveDates.inclusiveEndDate.logicalDate | sort,range-search |

#### Within the “person” field

|  |  |
| --- | --- |
| Field | Features |
| person.termName | sort |
| person.naId | sort |
| person.birthDate | sort, range-search |
| person.deathDate | sort, range-search |

#### Within the “organization” field

|  |  |
| --- | --- |
| Field | Features |
| organization.termName | sort |
| organization.naId | sort |
| organization.establishDate | sort, range-search |
| person.abolishDate | sort, range-search |

#### Within the “topicalSubject” field

|  |  |
| --- | --- |
| Field | Features |
| topicalSubject.termName | sort |
| topicalSubject.naId | sort |

#### Within the “geographicSubject” field

|  |  |
| --- | --- |
| Field | Features |
| geographicSubject.termName | sort |
| geographicSubject.naId | sort |

#### Within the “specificRecordsType” field

|  |  |
| --- | --- |
| Field | Features |
| specificRecordsType.termName | sort |
| specificRecordsType.naId | sort |

#### Within the “objects” field

When included inside of a holding (resultTypes=holding), the objects field is multi-valued and therefore there are no object fields which can be used for sorting or numeric or date range searching.

When included inside of a digital object (resultTypes=object), the following fields are available:

|  |  |
| --- | --- |
| Field | Features |
| objects.object.file.technicalMetadata.size | sort, range-search |
| objects.object.@objectId | sort |

#### Within the “publicContributions” field

Since all data elements within the public contributions field are multi-valued, there are no sub-fields within the “publicContributions” field which can be used for sorting or for numeric or date range searching.

## Search Results Output

In response to a search request, the search API will return the following information, either structured as XML or JSON as requested:

* Header information
  + Status of the request
  + Any error messages which occurred
  + The time it took to satisfy the request
  + All of the search parameters specified with the request
* Search results
  + The main list of search results, including the total number of documents which match the query
  + Each result will contain the score (aka relevancy) of the result and a list of all of the fields requested to be included with the result.

### Header Information

The standard headers specified in section 1.4.2 are returned by the search API.

For example:

XML Example

GET https://research.archives.gov/api/v1?q=truman&format=xml

<opa-response>

<header status="200" time="231">

<request path="/api/v1">

<q>truman</q>

<format>xml</format>

</request>

</header>

.

. search results will be returned here

.

</opa-response>

JSON Example

GET https://research.archives.gov/api/v1?q=truman

{

"header":{

"@status":"200",

"@duration":"231",

"request":{

"@path":"/api/v1",

"q":"truman"

}

},

.

. search results will be returned here

.

}

### Search Results

Search results will be returned as a list of <result> tags inside a nested <results> tag.

The metadata returned inside <results> will include:

* **@total** – The total number of results across all of OPA which match the query. This number may be thousands or millions large.
* **@offset** – The offset number of the first result returned.
* **@rows** – The actual number of results returned.

Within each <result> tag will include the following:

* **@num** – The number of this result across all results (never less than @offset)
* **@score** – The relevancy score for this result.
* **@resultType** – The type of the result.
  + One of “description”, “object”, “person”, “organization”, “topicalSubject”, “geographicSubject”, “specificRecordsType”, “archivesWeb" or “presidentialWeb”.
* **@naId** – The NAID for the result, if it is a description or digital object.
* **description** – The complete description metadata, for descriptions and digital objects.
* **person** – The complete person metadata, for people authority records.
* **organization** – The complete organization metadata, for organization authority records.
* **topicalSubject** – The complete topical subject authority record metadata.
* **geographicSubject** – The complete geographic subject authority record metadata.
* **specificRecordsType** – The complete specific records type authority record metadata.
* **publicContributions** – The complete contribution metadata, holding all public contributions and associated metadata.
  + See section 7 for more information on the contributions metadata.
* **objects** – The complete objects metadata, holding all objects and nested metadata (for descriptions) or the single object metadata (for digital objects).
  + See section 6 for more information on the objects metadata.

XML Example

https://research.archives.gov/api/v1?action=search&q=truman&resultTypes=holding&format=xml

<results total="6695" offset="0" rows="25">

<result num="0" score="2.34" naId="7226539" resultType="item">

<description>

.

. complete description goes here

.

</description>

<objects version="OPA-OBJECTS-1.0" count="3" created="2014-03-14T13:44:34">

.

. List of all objects (plus all object technical metadata and contribution metadata) goes here

.

</objects>

<publicContributions>

.

. All description-level public contributions go here

.

</publicContributions>

</result>

<result num="1" score="2.14" naId="7226539" resultType="person">

<person>

.

. complete person metadata goes here

.

</person>

<publicContributions>

.

. All authority person-level public contributions go here

.

</publicContributions>

</result>

.

. MORE RESULTS GO HERE

.

</results>

JSON Example

https://research.archives.gov/api/v1?action=search&q=truman&resultTypes=holding

{

"@total":"6695",

"@offset":"0",

"@rows":"25",

"result":[

{

"@num":"0",

"@score":"2.34",

"@naId":"7226539",

"@resultType":"item",

"description":{

.

. complete description goes here

.

},

"objects":{

.

. List of all objects (plus all object technical metadata and contribution metadata) goes here

.

},

"publicContributions":{

.

. All description-level public contributions go here

.

}

},

{

"@num":"1",

"@score":"2.14",

"@naId":"7226539",

"@resultType":"person",

"person":{

.

. complete person metadata goes here

.

},

"publicContributions":{

.

. All authority person-level public contributions go here

.

}

},

.

. MORE RESULTS GO HERE

.

]

}

#### Search results with portions of XML Fields

In addition to selecting an entire XML field, portions of the XML field can be selected:

XML Example

GET https://research.archives.gov/api/v1?q=truman&resultTypes=holding&  
resultFields=description.localIdentifier,description.topicalSubjectArray.topicalSubject.termName  
&format=xml

<results total="6695" offset="0" rows="25">

<result num="0" score="2.34" resultType="item" naId="65671">

**<description>**

**<localIdentifier>342-USAF-18636</localIdentifier>**

**<topicalSubjectArray>**

**<topicalSubject>**

**<termName>Airmen</termName>**

**</topicalSubject>**

**<topicalSubject>**

**<termName>Atomic bomb</termName>**

**</topicalSubject>**

**<topicalSubject>**

**<termName>Beaches</termName>**

**</topicalSubject>**

**</topicalSubjectArray>**

**</description>**

</result>

.

. MORE RESULTS GO HERE

.

</results>

JSON Example

GET https://research.archives.gov/api/v1?q=truman&resultTypes=holding&  
resultFields=description.local-identifier,description.subject-references.subject-reference

{

"@total":"6695",

"@offset":"0",

"@rows":"25",

"result":[

{

"@num":"0",

"@score":"2.34",

"@resultType":"item",

"@naId":"65671",

**"description":{**

**"localIdentifier":"342-USAF-18636",**

**"topicalSubjectArray":{**

**"topicalSubject":[**

**{**

**"termName":"Airmen"**

**},**

**{**

**"termName":"Atomic bomb"**

**},**

**{**

**"termName":"Beaches"**

**}**

**]**

**}**

**}**

**},**

.

. MORE RESULTS GO HERE

.

]

}

### Web Page Results

Wep page results are returned with just the URL and title.

XML Example

GET https://research.archives.gov/api/v1?q=truman&  
resultTypes=archivesWeb,presidentialWeb&format=xml

<results total="312" offset="0" rows="3">

<result num="0" score="1.54" resultType="presidentialWeb">

<title>Harry S. Truman Library and Museum</title>

<url>https://www.trumanlibrary.org/</url>

</result>

<result num="1" score="1.55" resultType="archivesWeb">

<title>Spring Prologue Magazine Highlights Truman’s 125th Birthday</title>

<url>https://www.archives.gov/press/press-releases/2009/nr09-74.html</url>

</result>

<result num="2" score="1.53" resultType="presidentialWeb">

<title>Truman: HST Biography</title>

<url>https://www.trumanlibrary.org/hst-bio.htm</url>

</result>

</results>

JSON Example

GET https://research.archives.gov/api/v1?q=truman&resultTypes=archivesWeb,presidentialWeb

{

"@total":"312",

"@offset":"0",

"@rows":"3",

"result":[

{

"@num":"0",

"@score":"1.54",

"@resultType":"presidentialWeb",

"title":"Harry S. Truman Library and Museum",

"url":"https://www.trumanlibrary.org/"

},

{

"@num":"1",

"@score":"1.55",

"@resultType":"archivesWeb",

"title":"Spring Prologue Magazine Highlights TrumanÆs 125th Birthday",

"url":"https://www.archives.gov/press/press-releases/2009/nr09-74.html"

},

{

"@num":"2",

"@score":"1.53",

"@resultType":"presidentialWeb",

"title":"Truman: HST Biography",

"url":"https://www.trumanlibrary.org/hst-bio.htm"

}

]

}

### Complete Example

The following is a complete Search API response example.

*Note that the content is for documentation purposes only – the actual content returned by the given search will be different.*

XML Example

GET https://research.archives.gov/api/v1?q=truman&format=xml

<opa-response>

<header status="200" time="231">

<request path="/api/v1">

<q>truman</q>

</request>

</header>

<results total="6695" offset="0" rows="25">

<result num="0" score="2.34" resultType="item" naId="7226539">

<description>

<generalRecordsTypeArray>

<generalRecordsType>

<naId>10031415</naId>

<termName>Textual Records</termName>

</generalRecordsType>

</generalRecordsTypeArray>

<isUnderEdit>false</isUnderEdit>

<localIdentifier>hst-ppp\_93-1\_18</localIdentifier>

<naId>C</naId>

<recordKey>7328414</recordKey>

<title>Letter from George McGovern to Harry S. Truman</title>

.

. More description fields goes here

.

</description>

<objects version="OPA-OBJECTS-1.0" count="3" created="2014-03-14T13:44:34">

<object>

<file path="./content/hst-ppp\_93-1\_18-01.jpg" url="https://research.archives.gov/id/7226539/content/hst-ppp\_93-1\_18-01.jpg"/>

<thumbnail path="./opa-rendition/thumbnails/hst-ppp\_93-1\_18-01-thumb.jpg" url="https://research.archives.gov/id/7226539/opa-rendition/thumbnails/hst-ppp\_93-1\_18-01-thumb.jpg" mime="image/jpeg"/>

<publicContributions>

<transcription lastModified="2014-02-21T16:22:58" version="232">

<users total="3">

<user user="tjefferson" fullName="Thomas Jefferson" isNaraStaff="true" lastModified="2014-02-21T16:22:58" version="13"/>

<user user="mstewart" fullName="Meredith Stewart" isNaraStaff="true" lastModified="2014-02-08T12:59:49" version="12"/>

<user user="kmartin" lastModified="2014-02-02T08:44:12" version="10"/>

</users>

<text>GEORGE McGOVERN

United States Senate

Washington, D.C. 20510

August 19, 1972

The Honorable Harry S. Truman

Independence, Missouri

My dear Mr. President:

It is an honor for me, as our party's

candidate for President of the United States. . .</text>

</transcription>

</publicContributions>

</object>

<object>

<file path="./content/hst-ppp\_93-1\_18-02.jpg" url="https://research.archives.gov/id/7226539/content/hst-ppp\_93-1\_18-02.jpg"/>

<thumbnail path="./opa-rendition/thumbnails/hst-ppp\_93-1\_18-02-thumb.jpg" url="https://research.archives.gov/id/7226539/opa-rendition/thumbnails/hst-ppp\_93-1\_18-02-thumb.jpg" mime="image/jpeg"/>

</object>

<object>

<file path="./content/hst-ppp\_93-1\_18-03.jpg" url="https://research.archives.gov/id/7226539/content/hst-ppp\_93-1\_18-03.jpg"/>

<thumbnail path="./opa-rendition/thumbnails/hst-ppp\_93-1\_18-03-thumb.jpg" url="https://research.archives.gov/id/7226539/opa-rendition/thumbnails/hst-ppp\_93-1\_18-03-thumb.jpg" mime="image/jpeg"/>

</object>

</objects>

<publicContributions>

<comments comments="3" replies="2" total="5">

<comment commentId="19328" user="kmartin" created="2014-02-10T14:36:23"

lastModified="2014-02-10T14:40:12">

<text>One wonders if this letter to Frm Pres. Truman was felt to be obligatory, or if there was real affection between the two.</text>

<replies>

<reply replyId="45239" commentId="19328" user="mstewart" fullName="Meredith Stewart" naraStaff="true" created="2014-02-10T15:00:03" lastModified="2014-02-10T15:00:03">

<text>True, but from different generations.</text>

</reply>

<reply replyId="45422" commentId="19328" user="mkoneni" created="2014-03-08T04:12:52" lastModified="2014-03-08T04:18:02">

<text>I'm sure they ran into each other a bunch, you know at charity functions and golf tournaments...</text>

</reply>

</replies>

</comment>

<comment commentId="19102" user="tjefferson" created="2014-03-08T04:12:52" lastModified="2014-03-08T04:12:52" isDeleted="true"/>

<comment commentId="17024" user="rbarnes" created="2014-03-09T11:14:27" lastModified="2014-03-08T04:12:52" >

<text>Seems rather formulaic to me. I bet a staff member wrote it.</text>

</comment>

</comments>

<tags total="3">

<tag user="jdoe" created="2014-02-21T16:22:58">Truman Post President</tag>

<tag user="agullet" created="2014-02-27T18:23:47">McGovern Letters</tag>

<tag user="mstewart" fullName="Meredith Stewart" naraStaff="true" created="2014-03-19T07:07:42">nara:president=Truman</tag>

</tags>

</publicContributions>

</result>

<result num="1" score="2.14" resultType="item" naId="7283002">

<description>

<generalRecordsTypeArray>

<generalRecordsType>

<naId>10031415</naId>

<termName>Textual Records</termName>

</generalRecordsType>

</generalRecordsTypeArray>

<isUnderEdit>false</isUnderEdit>

<localIdentifier>hst-ppp\_93-1\_18</localIdentifier>

<naId>7283002</naId>

<recordKey>7395123</recordKey>

<title>Spring 1976: 7-69-1: Truman on CIA (Examining President Truman's Role in the Establishment of the Agency), by Thomas F. Troy</title>

.

. More description fields goes here

.

</description>

<objects version="OPA-OBJECTS-1.0" count="1" created="2014-03-01T12:12:02">

.

. More objects go here

.

</objects>

<publicContributions>

.

. More public contributions

.

</publicContributions>

</result>

.

. MORE RESULTS GO HERE

.

</results>

</opa-response>

JSON Example

GET https://research.archives.gov/api/v1?q=truman

{

"header":{

"@status":"200",

"@time":"231",

"request":{

"@path":"/api/v1",

"q":"truman"

}

},

"results":{

"@total":"6695",

"@offset":"0",

"@rows":"25",

"result":[

{

"@num":"0",

"@score":"2.34",

"@resultType":"item",

"@naId":"7226539",

"description":{

"generalRecordsTypeArray":{

"generalRecordsType":{

"naId":"10031415",

"termName":"Textual Records"

}

},

"isUnderEdit":"false",

"localIdentifier":"hst-ppp\_93-1\_18",

"naId":"7226539",

"recordKey":"7328414",

"title":"Letter from George McGovern to Harry S. Truman",

.

. More description fields goes here

.

},

"objects":{

"@version":"OPA-OBJECTS-1.0",

"@count":"3",

"@created":"2014-03-14T13:44:34",

"object":[

{

"file":{

"@path":"./content/hst-ppp\_93-1\_18-01.jpg",

"@url":"https://research.archives.gov/id/7226539/content/hst-ppp\_93-1\_18-01.jpg"

},

"thumbnail":{

"@path":"./opa-rendition/thumbnails/hst-ppp\_93-1\_18-01-thumb.jpg",

"@url":"https://research.archives.gov/id/7226539/opa-rendition/thumbnails/hst-ppp\_93-1\_18-01-thumb.jpg",

"@mime":"image/jpeg"

},

"publicContributions":{

"transcription":{

"@lastModified":"2014-02-21T16:22:58",

"@version":"232",

"users":{

"@total":"3",

"user":[

{

"@user":"tjefferson",

"@fullName":"Thomas Jefferson",

"@isNaraStaff":"true",

"@lastModified":"2014-02-21T16:22:58",

"@version":"13"

},

{

"@user":"mstewart",

"@fullName":"Meredith Stewart",

"@isNaraStaff":"true",

"@lastModified":"2014-02-08T12:59:49",

"@version":"12"

},

{

"@user":"kmartin",

"@lastModified":"2014-02-02T08:44:12",

"@version":"10"

}

]

},

"text":"GEORGE McGOVERN

United States Senate

Washington, D.C. 20510

August 19, 1972

The Honorable Harry S. Truman

Independence, Missouri

My dear Mr. President:

It is an honor for me, as our party's

candidate for President of the United States. . ."

}

}

},

{

"file":{

"@path":"./content/hst-ppp\_93-1\_18-02.jpg",

"@url":"https://research.archives.gov/id/7226539/content/hst-ppp\_93-1\_18-02.jpg"

},

"thumbnail":{

"@path":"./opa-rendition/thumbnails/hst-ppp\_93-1\_18-02-thumb.jpg",

"@url":"https://research.archives.gov/id/7226539/opa-rendition/thumbnails/hst-ppp\_93-1\_18-02-thumb.jpg",

"@mime":"image/jpeg"

}

},

{

"file":{

"@path":"./content/hst-ppp\_93-1\_18-03.jpg",

"@url":"https://research.archives.gov/id/7226539/content/hst-ppp\_93-1\_18-03.jpg"

},

"thumbnail":{

"@path":"./opa-rendition/thumbnails/hst-ppp\_93-1\_18-03-thumb.jpg",

"@url":"https://research.archives.gov/id/7226539/opa-rendition/thumbnails/hst-ppp\_93-1\_18-03-thumb.jpg",

"@mime":"image/jpeg"

}

}

]

},

"publicContributions":{

"comments":{

"@comments":"3",

"@replies":"2",

"@total":"5",

"comment":[

{

"@commentId":"19328",

"@user":"kmartin",

"@created":"2014-02-10T14:36:23",

"@lastModified":"2014-02-10T14:40:12",

"text":"One wonders if this letter to Frm Pres. Truman was felt to be obligatory, or if there was real affection between the two.",

"replies":{

"reply":[

{

"@replyId":"45239",

"@commentId":"19328",

"@user":"mstewart",

"@fullName":"Meredith Stewart",

"@naraStaff":"true",

"@created":"2014-02-10T15:00:03",

"@lastModified":"2014-02-10T15:00:03",

"text":"True, but from different generations."

},

{

"@replyId":"45422",

"@commentId":"19328",

"@user":"mkoneni",

"@created":"2014-03-08T04:12:52",

"@lastModified":"2014-03-08T04:18:02",

"text":"I'm sure they ran into each other a bunch, you know at charity functions and golf tournaments..."

}

]

}

},

{

"@commentId":"19102",

"@user":"tjefferson",

"@created":"2014-03-08T04:12:52",

"@lastModified":"2014-03-08T04:12:52",

"@isDeleted":"true"

},

{

"@commentId":"17024",

"@user":"rbarnes",

"@created":"2014-03-09T11:14:27",

"@lastModified":"2014-03-08T04:12:52",

"text":"Seems rather formulaic to me. I bet a staff member wrote it."

}

]

},

"tags":{

"@total":"3",

"tag":[

{

"@user":"jdoe",

"@created":"2014-02-21T16:22:58","$":"Truman Post President"

},

{

"@user":"agullet",

"@created":"2014-02-27T18:23:47","$":"McGovern Letters"

},

{

"@user":"mstewart",

"@fullName":"Meredith Stewart",

"@naraStaff":"true",

"@created":"2014-03-19T07:07:42","$":"nara:president=Truman"

}

]

}

}

},

{

"@num":"1",

"@score":"2.14",

"@resultType":"item",

"@naId":"7283002",

"description":{

"generalRecordsTypeArray":{

"generalRecordsType":{

"naId":"10031415",

"termName":"Textual Records"

}

},

"isUnderEdit":"false",

"localIdentifier":"hst-ppp\_93-1\_18",

"naId":"7283002",

"recordKey":"7395123",

"title":"Spring 1976: 7-69-1: Truman on CIA (Examining President Truman's Role in the Establishment of the Agency), by Thomas F. Troy",

.

. More description fields goes here

.

},

"objects":{

"@version":"OPA-OBJECTS-1.0",

"@count":"1",

"@created":"2014-03-01T12:12:02",

.

. More objects go here

.

},

"publicContributions":{

.

. More public contributions

.

}

},

.

. MORE RESULTS GO HERE

.

],

}

}

# Creating or Updating Public Contributions

The OPA API provides methods for creating, updating, and deleting public contributions.

Note that modified public contributions will not show up in search results until they have been re-indexed.

## Requires login credentials

Modifying public contributions requires that you log-in with a username/password and obtain the “credentials=” parameter.

See section 1.5 for more information.

## Selecting What to Contribute to – URL Path Version

As with standard RESTful APIs, selecting what to contribute to is specified using the path portion of the URL.

For example, to add a comment to the description with NAID = 72265389:

POST https://research.archives.gov/api/v1/id/7226539/comments?text=<comment-text-here>

For example, to add a comment to object 3 within the description with NAID = 72265389:

POST https://research.archives.gov/api/v1/id/7226539/objects/3/comments?  
text=<comment-text-here>

### Selecting Comments to Modify

Comments are a special case because there can be multiple comments with a description, object, or authority record. Therefore, use the “commentId” to select a specific comment.

For example, to delete a specific comment by comment ID, just do this:

DELETE https://research.archives.gov/api/v1/comments/182423

### Selecting Translations to Modify

Since there can be multiple translations, specify the ISO 639-3 (three letter) language code of the translation to create or change:

PUT https://research.archives.gov/api/v1/id/7226539/objects/314232/translations/ita?  
text=Traduzione testo va qui.

## Selecting What to Annotate – Search Version

You can also use the parameters specified in section 2.1 to search for a record and then contribute to the returned item.

*Note: If you select multiple things to contribute to, only the first will actually be modified.*

For example, to add a comment to the description with NAID = 72265389:

POST https://research.archives.gov/api/v1?type=description&naIds=7226539&  
comment=<comment-text-here>

For example, to add a comment to object 3 within the description with NAID = 72265389:

POST https://research.archives.gov/api/v1?naIds=7226539&objectIds=3&  
comment=<comment-text-here>

Note that “type=object” is not needed above because the “objectIds” parameter will only return objects.

### Selecting Comments to Modify

Comments are a special case because there can be multiple comments with a description, object, or authority record. Therefore, an additional “commentId” parameter is specified to select a specific comment.

For example, to delete a specific comment by comment ID, just do this:

DELETE https://research.archives.gov/api/v1?commentId=182423

### Selecting Translations to Modify

Since there can be multiple translations, specify the ISO 639-3 (three letter) language code of the translation to create or change:

PUT https://research.archives.gov/api/v1?naIds=7226539&objectIds=3&language=ita&  
translation=Traduzione testo va qui.

## Specifying the Contribution Modification – URL Path Version

HTTP methods will be used to specify what to do with the contribution:

* POST – Create a new contribution.
* PUT – Update a contribution
* DELETE – Delete an contribution

Not all public contributions accept all methods.

Use the “text=” parameter to specify the text for the contribution (for the POST or PUT method).

### Comments

This section covers the types of modifications which are allowed on comments.

Creating a new comment:

POST https://research.archives.gov/api/v1/{resource}/comments?text={comment-text-here}

Deleting a comment:

DELETE https://research.archives.gov/api/v1/comments/{comment-id}

Updating a comment:

PUT https://research.archives.gov/api/v1/comments/{comment-id}?text={comment-text-here}

Examples:

Creating a new comment on any NAID:

POST https://research.archives.gov/api/v1/id/88423/comments?text=I went to this conference. It was fabulous!

Deleting a comment from any NAID:

DELETE https://research.archives.gov/api/v1/comments/5566

Updating a comment for any NAID:

PUT https://research.archives.gov/api/v1/comments/5566?comment=I went to this conference. It was super-fabulous!

Creating a new comment on an object:

POST https://research.archives.gov/api/v1/id/5278823/objects/12/comments?text=Clear evidence of a coverup here. Smoking gun.

### Tags

This section covers the types of modifications which are allowed on tags.

Add new tag:

POST https://research.archives.gov/api/v1/{resource}/tags?text={tag-text}

Delete tag:

DELETE https://research.archives.gov/api/v1/{resource}/tags/{tag-text}

Examples:

Add new tag to any NAID (description or authority record):

POST https://research.archives.gov/api/v1/id/88423/tags?text=TrumanResearch

Delete tag from any NAID (description or authority record):

DELETE https://research.archives.gov/api/v1/id/88423/tags/TrumanResearch

Add new tag to an object:

POST https://research.archives.gov/api/v1/id/5278823/objects/12/tags?text=TrumanResearch

Delete tag from an object:

DELETE https://research.archives.gov/api/v1/id/5278823/objects/12/tags/TrumanResearch

### Transcriptions

This section covers the types of modifications which are allowed on transcriptions.

Create or update a transcription:

POST https://research.archives.gov/api/v1/{resource}/transcription?text=<transcription-text>

Examples:

POST https://research.archives.gov/api/v1/id/5278823/objects/12/transcription?text=George McGovern, U.S. Senate, Washington DC, Mr. President: It was an honor for me to meet you…

### Translations

This section covers the types of modifications which are allowed on translations.

Create or update a translation:

POST https://research.archives.gov/api/v1/{resource}/translations/{lang-code}?text=<translation-text>

Examples:

POST https://research.archives.gov/api/v1/id/5278823/objects/121423/translations/ita?  
text=George McGovern, Senato degli Stati Uniti, Washington DC, signor Presidente: E 'stato un onore per me incontrarvi…

## Specifying the Contribution Modification – Search Version

To specify the contribution to be written into OPA, specify one of the appropriate parameters:

* tag= To select or create tags
* comment= To specify text for a new comment
* transcription= To specify text for a new or updated transcription
* translation= To specify text for a new or updated translation

HTTP methods will be used to specify what to do with the contribution:

* POST – Create a new contribution.
* PUT – Update a contribution
* DELETE – Delete a contribution

Not all contributions accept all methods.

### Comments

This section covers the types of modifications which are allowed on comments.

Creating a new comment:

POST https://research.archives.gov/api/v1?{selection-parameters}&comment={comment-text-here}

Deleting a comment:

DELETE https://research.archives.gov/api/v1?commentId={comment-id}

Updating a comment:

PUT https://research.archives.gov/api/v1?commentId={comment-id}&comment={comment-text-here}

Examples:

Creating a new comment on any NAID:

POST https://research.archives.gov/api/v1?naIds=88423&comment=I went to this conference. It was fabulous!

Deleting a comment from any NAID:

DELETE https://research.archives.gov/api/v1?commentId=5566

Updating a comment for any NAID:

PUT https://research.archives.gov/api/v1?commentId=5566&comment=I went to this conference. It was super-fabulous!

Creating a new comment on an object:

POST https://research.archives.gov/api/v1?naIds=5278823&objectIds=12&comment=Clear evidence of a coverup here. Smoking gun.

### Tags

This section covers the types of modifications which are allowed on tags.

Add new tag:

POST https://research.archives.gov/api/v1?{select-parameters}&tag={tag-text}

Delete tag:

DELETE https://research.archives.gov/api/v1?{selection-parameters}&tag={tag-text}

Examples:

Add new tag to any NAID (description or authority record):

POST https://research.archives.gov/api/v1?naIds=88423&tag=TrumanResearch

Delete tag from any NAID (description or authority record):

DELETE https://research.archives.gov/api/v1?naIds=88423&tag=TrumanResearch

Add new tag to an object:

POST https://research.archives.gov/api/v1?naIds=5278823&objectIds=12&tag=TrumanResearch

Delete tag from an object:

DELETE https://research.archives.gov/api/v1?naIds=5278823&objectIds=12&tag=TrumanResearch

### Transcriptions

This section covers the types of modifications which are allowed on transcriptions.

Create or update a transcription:

POST https://research.archives.gov/api/v1?{selection-parameters}&transcription=<transcription-text>

Examples:

POST https://research.archives.gov/api/v1?naIds=5278823&objectIds=12&transcription=George McGovern, U.S. Senate, Washington DC, Mr. President: It was an honor for me to meet you…

### Translations

This section covers the types of modifications which are allowed on translations.

Create or update a translation:

POST https://research.archives.gov/api/v1?{selection-parameters}&language=<code>&translation=<text>

Examples:

POST https://research.archives.gov/api/v1?naIds=5278823&objectIds=12&language=ita&  
translation=George McGovern, Senato degli Stati Uniti, Washington DC, signor Presidente: E 'stato un onore per me incontrarvi…

# Exports

OPA can export search results and content in a variety of different formats with a variety of different content. Output can be viewed, printed, or downloaded for off-line processing.

Exports are implemented by doing the following:

1. Specify what results to export.
   1. This is done with the search parameters specified in section 2.
2. Specify what to export for each result.
   1. This is done with the export parameters specified in section 2.

## Examples

Export the full metadata for results of a simple search as HTML:

GET https://research.archives.gov/api/v1?action=export&q=truman&resultTypes=holding&  
export.what=metadata&export.format=html

Export the full metadata for results of a simple search as HTML, with thumbnails:

GET https://research.archives.gov/api/v1?action=export&q=truman&resultTypes=holding&  
export.what=metadata,thumbnails&export.format=html

Export the full results of a simple search as PDF, with thumbnails, tags, and comments:

GET https://research.archives.gov/api/v1?action=export&q=truman&resultTypes=holding&  
export.what=metadata,thumbnails,tags,comments&export.format=pdf

Export the full results of selected descriptions as text, with transcript:

GET https://research.archives.gov/api/v1?action=export&naIds=72889324,6232422,  
5523412,6872412,7114423&resultTypes=holding&  
export.what=metadata,transcriptions&export.format=txt

Export the full results of a simple search as XML, with everything possible including all language translations:

GET https://research.archives.gov/api/v1?action=export&q=truman&resultTypes=holding&  
export.what=metadata,thumbnails,tags.comments,transcriptions,translations&  
export.format=xml&export.languages.all=true

Export selected fields as XML:

GET https://research.archives.gov/api/v1?action=export&q=truman&resultTypes=holding&  
export.what=metadata&resultFields=description.naId,description.title,  
description.localIdentifier, description.coverageDates&export.format=xml

Export all of the fields above, *plus* the actual content files and thumbnail files as a bulk export:

GET https://research.archives.gov/api/v1?action=export&q=truman&resultTypes=holding&  
export.what=metadata&resultFields=description.naId,description.title,  
description.localIdentifier, description.coverageDates&export.format=xml&export.bulk=true&  
export.bulk.content=objects,thumbnails

## Response (standard export)

When exporting results which are not bulk exports, OPA will directly respond with the content.

For example, if executing the following URL:

GET https://research.archives.gov/api/v1?action=export&q=truman&resultTypes=holding&  
export.what=metadata&export.format=html

The data in the response will be HTML data.

Error Responses

If there is an error in one of the parameters, OPA will respond with an HTTP “400” error (bad request). The content of the error will be a standard <opa-response> XML (see section 1.4.2) or JSON.

Possible error codes are specified above in section 2.2.

### Example CSV Response

As an example, the following is how a CSV response will be returned by the APIs. For the following URL:

GET https://research.archives.gov/api/v1?action=export&q=truman&resultTypes=holding&  
export.what=metadata&resultFields=description.naid&description.localIdentifier&description.title&description.generalRecordsTypeArray&export.format=csv&rows=5

The result will be: (note: titles truncated to increase readability)

description.naid,description.localIdentifier,description.title,description.generalRecordsTypeArray

7328414,hst-ppp\_93-1\_18,Letter from George McGovern to Harry S. Truman,10031415|Textual Records

7283080,,"Spring 1981: 7-89-4: Harry S. Truman on CIA Covert Operati...",10031415|Textual Records

7283002,,"Spring 1976: 7-69-1: Truman on CIA (Examining President Tr...",10031415|Textual Records

7283870,,"Fall 1973: 17-34-2: What did Truman Say About the CIA?, by...",10031415|Textual Records

7542722,,"Correspondence between Harry S. Truman and President Richa...",10031415|Textual Records

## Bulk Exports

Bulk exports from OPA are performed using the following steps:

1. Choose the set of results to export (either using a search query, a list of IDs, or a “My List”, see above).
2. Initiate the bulk-export with the parameters above.
   1. Set export.bulk=true.
3. Save the export ID number returned by OPA, see below.
   1. This number will be used to get status on the bulk export and to download the resulting compressed file when it is ready.
4. Get the status of the bulk download by going to:

GET https://research.archives.gov/api/v1?bulkExportId={export-id}

1. Once the bulk-download is ready, fetch the bulk download file from:

GET https://research.archives.gov/api/v1/bulk-exports/files/nara-bulk-export-{export-id}.tar.gz

and then download the contents.

XML Example

GET https://research.archives.gov/api/v1?action=export&q=truman&  
export.what=metadata,transcriptions&resultFields=description&export.format=xml&  
export.bulk=true&export.bulk.content=objects,thumbnails&format=xml

<opa-response>

<header status="200" time="231">

<request path="/api/v1">

<action>export</action>

<q>truman</q>

<export.what>metadata,transcriptions</export.what>

<resultFields>description</resultFields>

<export.format>xml</export.format>

<export.bulk>true</export.bulk>

<export.bulk.content>objects,thumbnails</export.bulk.content>

</request>

</header>

<bulk-export exportId="42"/>

</opa-response>

JSON Example

GET https://research.archives.gov/api/v1?action=export&q=truman&export.what=metadata&  
resultFields=description&export.format=json&export.bulk=true&  
export.bulk.content=objects,thumbnails,transcriptions

{

"header":{

"@status":"200",

"@duration":"231",

"request":{

"@path":"/api/v1",

"action":"export",

"q":"truman",

"export.what":"metadata,transcriptions",

"resultFields":"description",

"export.format":"json",

"export.bulk":"true",

"export.bulk.content":"objects,thumbnails"

}

},

"bulk-export":{

"@exportId":"42"

}

}

### Fetching Status on a Bulk Export

The status of a prior bulk export initiated can be acquired with the following URL:

GET https://research.archives.gov/api/v1?bulkExportId={export-id}

Note:

* Bulk exports are only available when logged in.
* The logged in user can only fetch the bulk exports for themselves (not for others).
* However, the download file *is* available to anyone.

Response

In the download output, the following attributes are provided for the bulk export:

* **@bulkExportId** – The ID number for the bulk export. This will be the same ID number reported when the bulk export was first created.
* **@summary** – A computer generated description about the bulk export from the query used to generate the results.
* **@isComplete** – “true” if the download is complete and ready to be downloaded.
* **@pctComplete** – The percentage complete. An integer from 1 to 100.
* **@count** – The total count of search results selected to be included in the download.
  + For bulk exports from searches, this will be the number of search results.
* **@format** – The search results format requested for the bulk-export.
* **@requested** – The date-time the request was made and the bulk-export was queued.
* **@completed** – The date-time when the export was completed and made available.
* **@expires** – The date-time when the export file will be removed from the system.
* **@downloadUrl** – The URL where the export file can be downloaded.

XML Example

GET https://research.archives.gov/api/v1?bulkExportId=42&format=xml

<bulkExport bulkExportId="42" summary="Search: truman" isComplete="false" pctComplete="50" count="432" format="XML" requested="2014-02-11T12:41:44" />

JSON Example

GET https://research.archives.gov/api/v1?bulkExportId=42

{

"@bulkExportId":"42",

"@summary":"Search: truman",

"@isComplete":"false",

"@pctComplete":"50",

"@count":"432",

"@format":"XML",

"@requested":"2014-02-11T12:41:44"

}

### Downloading a Bulk Export File

Once the download is complete, the download file will be returned in the bulk exports list. The file name will be based on the download ID:

GET https://research.archives.gov/api/v1/bulk-exports/files/nara-bulk-export-{bulkExportId}.tar.gz

For example:

GET https://research.archives.gov/api/v1/bulk-exports/files/nara-bulk-export-43.tar.gz

# Objects Metadata

This section covers the metadata structure for the “objects” field returned in search results.

## Metadata Description

The fields in the objects field are as follows:

* **@version** – Specifies the type and version of the objects XML.
  + The only version currently defined is “OPA-OBJECTS-1.0”
* **@created** – Date-time this objects.xml file was first created.
  + Note: Must be maintained as the file is updated.
* **object** – specifies a digital object
  + This is a holder for multiple renditions and metadata for the object

Each <object> tag will have the following nested information:

* **@objectId** – The identifier for this object, unique across all objects within the same NAID.
  + Note: This will be the stable DAS object ID for the object, for objects from DAS.
  + For objects from SEIP, this will be a unique number for the object within the SEIP.
* **description** – This is a copy of the object’s description from the archival description.
  + From the LCDRG: “Provides information about the Digital Object that is not apparent from Object Designator or Object Identifier”
* **designator** – This is a copy of the object’s designator from the archival description.
  + From the LCDRG: “An identifier for each digital object when there is more than one digital object associated with an archival description.”
* **file** – Specifies the file which holds a copy or alternative representation of the content of the digital object. Has the following sub-elements:
  + **@purpose** – The purpose of the file (only “primary” supported in 1.0)
  + **@mime** – The MIME type of the file
  + **@path** – The path to the file, relative to the NAID URL.
  + **@url** – The full URL to the file.
  + **technicalMetadata** – Holds extracted metadata for the file

The contents of the technical metadata section will vary based on the type of file. See the appendix (section 9.2) for detailed information on all of the technical metadata which is gathered for each different file type.

* **thumbnail** – Specifies the location of the thumbnail for the object.

Note: This is only used for thumbnails which are an image of the actual content, and it may be missing. If missing, the mime type should be used to show an appropriate icon for the content in place of the thumbnail.

Note that thumbnails are OPA-generated for presentation purposes only and will not have technical metadata.

Has the following sub-elements:

* + **@path** – A path, relative to the NAID URL, for accessing the thumbnail.
  + **@url** – The full URL to the file.
  + **@mime** – The mime type of the thumbnail.
* **imageTiles** – If the content represents an image, <imageTiles> will specify the location of a tiled version of the document which can be accessed by [Open Seadragon](http://openseadragon.github.io/).

Has the following sub-elements:

* + **@path** – The path, relative to the NAID URL, where the image tiles can be found. This is often a path to a directory where the individual tiled files can be accessed.
  + **@url** – The full URL to the file.
  + **@format** – The format of the image tiles. Currently, only “DZI” is supported.
* **seData** – Holds search engine data for the object. This is additional extracted metadata for the file such as from/to/cc/bcc for E-mail files. This metadata will be specified as XML in the NARA SEIP (version 1) SE data format.

Has the following sub-elements:

* + **@path** – Specifies the path to the Search Engine metadata, relative to the NAID URL.
  + **@url** – The full URL to the file.
  + **@mime** – The mime type of the SE data. Always “text/xml”.
  + **all se-data fields** – Note that the <seData> tag will contain the entire metadata from the SE data file for the specified object.
* **publicContributions** – This is the parent tag for all public contributions contained within the object. It contains the following sub-elements:
  + **comments** – Lists all of the comments on the object, in the same format as described in section 7.1.
  + **tags** – Lists the tags on the object, in the same format as described in section 7.2.
  + **transcription** – If the object has a transcription, then the file name for the transcription will be specified, in the same format as described in section 7.3.  
    Contains the following sub-elements:
  + **translations** – Parent tag to hold the object translations, in the same format as described in section 7.4.

## Example

XML Example

Note that the metadata below would typically be embedded inside of a search result returned by the search engine. See section 3.7 for more information.

<objects version="OPA-OBJECTS-1.0" count="3" created="2014-03-14T13:44:34">

<object objectId="1">

<designator>1</designator>

<file purpose="primary" path="./content/hst-ppp\_93-1\_18-01.jpg"  
url="https://research.archives.gov/id/7228539/content/hst-ppp\_93-1\_18-01.jpg"  
mime="image/jpeg">

<technicalMetadata>

<createDate>2012-10-22T03:26:34Z</createDate>

<metadataDate>2013-02-20T02:55:27Z</metadataDate>

<modifyDate>2013-02-20T02:55:27Z</modifyDate>

<size>1142680</size>

<mime>image/jpeg</mime>

<width>2181</width>

<height>2781</height>

<resolution x="300" y="300" units="pixels/inch"/>

<bitsPerSample>8 8 8</bitsPerSample>

<photometricInterpretation>RGB</photometricInterpretation>

<orientation>horizontal</orientation>

<samplesPerPixel>3</samplesPerPixel>

<planarConfiguration>chunky</planarConfiguration>

<colorSpace>sRGB</colorSpace>

<compression>uncompressed</compression>

<software>Adobe Photoshop CS6 (Windows)</software>

</technicalMetadata>

</file>

<thumbnail path="./opa-rendition/thumbnails/hst-ppp\_93-1\_18-01-thumb.jpg"   
url="https://research.archives.gov/id/7228539/opa-rendition/thumbnails/hst-ppp\_93-1\_18-01-thumb.jpg"   
mime="image/jpeg"/>

<imageTiles path="./opa-rendition/image-tiles/hst-ppp\_93-1\_18-01\_jpg"  
url="https://research.archives.gov/id/7228539/opa-rendition/image-tiles/hst-ppp\_93-1\_18-01\_jpg" />

<publicContributions>

<comments comments="3" replies="2" total="5">

<comment commentId="19328" user="kmartin" created="2014-02-10T14:36:23"

lastModified="2014-02-10T14:40:12">

<text>One wonders if this letter to Frm Pres. Truman was felt to be obligatory, or if there was real affection between the two.</text>

<replies>

<reply replyId="45239" commentId="19328" user="mstewart" fullName="Meredith Stewart" naraStaff="true" created="2014-02-10T15:00:03" lastModified="2014-02-10T15:00:03">

<text>True, but from different generations.</text>

</reply>

<reply replyId="45422" commentId="19328" user="mkoneni" created="2014-03-08T04:12:52"  
lastModified="2014-03-08T04:18:02">

<text>I'm sure they ran into each other a bunch, you know at charity functions and golf tournaments...</text>

</reply>

</replies>

</comment>

<comment commentId="19102" user="tjefferson" created="2014-03-08T04:12:52"   
lastModified="2014-03-08T04:12:52" isDeleted="true"/>

<comment commentId="17024" user="rbarnes" created="2014-03-09T11:14:27"  
lastModified="2014-03-09T11:14:27">

<text>Seems rather formulaic to me. I bet a staff member wrote it.</text>

</comment>

</comments>

<tags total="3">

<tag text="Truman Post President" user="jdoe" created="2014-02-21T16:22:58"/>

<tag text="McGovern Letters" user="agullet" created="2014-02-27T18:23:47"/>

<tag text="nara:president=Truman" user="mstewart" fullName="Meredith Stewart" naraStaff="true" created="2014-03-19T07:07:42"/>

</tags>

<transcription lastModified="2014-02-21T16:22:58" isLocked="true" version="232">

<lockedBy user="gwashington" when="2014-02-22T10:08:41"/>

<users total="3">

<user user="tjefferson" fullName="Thomas Jefferson" isNaraStaff="true"

lastModified="2014-02-21T16:22:58" version="13"/>

<user user="mstewart" fullName="Meredith Stewart" isNaraStaff="true"

lastModified="2014-02-08T12:59:49" version="12"/>

<user user="kmartin" lastModified="2014-02-02T08:44:12" version="10"/>

</users>

<text>GEORGE McGOVERN

United States Senate

Washington, D.C. 20510

August 19, 1972

The Honorable Harry S. Truman

Independence, Missouri

My dear Mr. President:

It is an honor for me, as our party's

candidate for President of the United States. . .</text>

</transcription>

<translations total="3">

<translation langCode="ita" langName="Italian"   
lastModified="2014-02-21T16:22:58" isLocked="true" version="232">

<lockedBy user="gwashington" when="2014-02-22T10:08:41"/>

<users total="3">

<user user="tjefferson" fullName="Tomás Jefferson" isNaraStaff="true"

lastModified="2014-02-21T16:22:58" version="3"/>

<user user="gwashington" fullName="Giorgio Washington" isNaraStaff="true"

lastModified="2014-02-08T12:59:49" version="2"/>

</users>

<text>George McGovern

Senato degli Stati Uniti

Washington, D.C. 20510

19 ago 1972

L'Onorevole Harry S. Truman

Independence, Missouri

Mio caro Signor Presidente:

E 'un onore per me, come il nostro partito di

candidato alla Presidenza degli Stati Uniti. .</text>

</translation>

<translation langCode="spa" langName="Spanish">

.

. Full information on the Spanish translation goes here

.

</translation>

<translation langCode="deu" langName="German">

.

. Full information on the German translation goes here

.

</translation>

</translations>

</publicContributions>

</object>

<object objectId="2">

<designator>2</designator>

<file purpose="primary" path="./content/hst-ppp\_93-1\_18-02.jpg"   
url="https://research.archives.gov/id/7228539/content/hst-ppp\_93-1\_18-02.jpg"  
mime="image/jpeg">

<technicalMetadata>

.

. Technical metadata for the primary file for the second object will go here

.

</technicalMetadata>

</file>

<thumbnail path="./opa-rendition/thumbnails/hst-ppp\_93-1\_18-02-thumb.jpg"   
url="https://research.archives.gov/id/7228539/opa-rendition/thumbnails/hst-ppp\_93-1\_18-02-thumb.jpg"  
mime="image/jpeg"/>

<imageTiles path="./opa-rendition/image-tiles/hst-ppp\_93-1\_18-02\_jpg"  
url="https://research.archives.gov/id/7228539/opa-rendition/image-tiles/hst-ppp\_93-1\_18-02\_jpg"/>

</object>

<object objectId="3">

<designator>3</designator>

<file purpose="primary" path="./content/hst-ppp\_93-1\_18-03.jpg"   
url="https://research.archives.gov/id/7228539/content/hst-ppp\_93-1\_18-03.jpg  
mime="image/jpeg">

<technicalMetadata>

.

. Technical metadata for the primary file for the third object will go here

.

</technicalMetadata>

</file>

<thumbnail path="./opa-rendition/thumbnails/hst-ppp\_93-1\_18-03-thumb.jpg"   
url="https://research.archives.gov/id/7228539/opa-rendition/thumbnails/hst-ppp\_93-1\_18-03-thumb.jpg"  
mime="image/jpeg"/>

<imageTiles path="./opa-rendition/image-tiles/hst-ppp\_93-1\_18-03\_jpg"   
url="https://research.archives.gov/id/7228539/opa-rendition/image-tiles/hst-ppp\_93-1\_18-03\_jpg"  
format="DZI"/>

</object>

</objects>

JSON Example

Note that the metadata below would typically be embedded inside of a search result returned by the search engine. See section 3.7 for more information.

{

"@version":"OPA-OBJECTS-1.0",

"@count":"3",

"@created":"2014-03-14T13:44:34",

"object":[

{

"@**objectId**":"1",

"designator":"1",

"file":{

"@purpose":"primary",

"@path":"./content/hst-ppp\_93-1\_18-01.jpg",

"@url":"https://research.archives.gov/id/7228539/content/hst-ppp\_93-1\_18-01.jpg",

"@mime":"image/jpeg",

"technicalMetadata":{

"createDate":"2012-10-22T03:26:34Z",

"metadataDate":"2013-02-20T02:55:27Z",

"modifyDate":"2013-02-20T02:55:27Z",

"size":"1142680",

"mime":"image/jpeg",

"width":"2181",

"height":"2781",

"resolution":{

"@x":"300",

"@y":"300",

"@units":"pixels/inch"

},

"bitsPerSample":"8 8 8",

"photometricInterpretation":"RGB",

"orientation":"horizontal",

"samplesPerPixel":"3",

"planarConfiguration":"chunky",

"colorSpace":"sRGB",

"compression":"uncompressed",

"software":"Adobe Photoshop CS6 (Windows)"

}

},

"thumbnail":{

"@path":"./opa-rendition/thumbnails/hst-ppp\_93-1\_18-01-thumb.jpg",

"@url":"https://research.archives.gov/id/7228539/opa-rendition/thumbnails/hst-ppp\_93-1\_18-01-thumb.jpg",

"@mime":"image/jpeg"

},

"imageTiles":{

"@path":"./opa-rendition/image-tiles/hst-ppp\_93-1\_18-01\_jpg"

"@url":"https://research.archives.gov/id/7228539/opa-rendition/image-tiles/hst-ppp\_93-1\_18-01\_jpg"

},

"publicContributions":{

"comments":{

"@comments":"3",

"@replies":"2",

"@total":"5",

"comment":[

{

"@commentId":"19328",

"@user":"kmartin",

"@created":"2014-02-10T14:36:23",

"@lastModified":"2014-02-10T14:40:12",

"text":"One wonders if this letter to Frm Pres. Truman was felt to be obligatory, or if there was real affection between the two.",

"replies":{

"reply":[

{

"@replyId":"45239",

"@commentId":"19328",

"@user":"mstewart",

"@fullName":"Meredith Stewart",

"@naraStaff":"true",

"@created":"2014-02-10T15:00:03",

"@lastModified":"2014-02-10T15:00:03",

"text":"True, but from different generations."

},

{

"@replyId":"45422",

"@commentId":"19328",

"@user":"mkoneni",

"@created":"2014-03-08T04:12:52",

"@lastModified":"2014-03-08T04:18:02",

"text":"I'm sure they ran into each other a bunch, you know at charity functions and golf tournaments..."

}

]

}

},

{

"@commentId":"19102",

"@user":"tjefferson",

"@created":"2014-03-08T04:12:52",

"@lastModified":"2014-03-08T04:12:52",

"@isDeleted":"true"

},

{

"@commentId":"17024",

"@user":"rbarnes",

"@created":"2014-03-09T11:14:27",

"@lastModified":"2014-03-09T11:14:27",

"text":"Seems rather formulaic to me. I bet a staff member wrote it."

}

]

},

"tags":{

"@total":"3",

"tag":[

{

"@text":"Truman Post President",

"@user":"jdoe",

"@created":"2014-02-21T16:22:58"

},

{

"@text":"McGovern Letters",

"@user":"agullet",

"@created":"2014-02-27T18:23:47"

},

{

"@text":"nara:president=Truman",

"@user":"mstewart",

"@fullName":"Meredith Stewart",

"@naraStaff":"true",

"@created":"2014-03-19T07:07:42"

}

]

},

"transcription":{

"@lastModified":"2014-02-21T16:22:58",

"@isLocked":"true",

"@version":"232",

"lockedBy":{

"@user":"gwashington",

"@when":"2014-02-22T10:08:41"

},

"users":{

"@total":"3",

"user":[

{

"@user":"tjefferson",

"@fullName":"Thomas Jefferson",

"@isNaraStaff":"true",

"@lastModified":"2014-02-21T16:22:58",

"@version":"13"

},

{

"@user":"mstewart",

"@fullName":"Meredith Stewart",

"@isNaraStaff":"true",

"@lastModified":"2014-02-08T12:59:49",

"@version":"12"

},

{

"@user":"kmartin",

"@lastModified":"2014-02-02T08:44:12",

"@version":"10"

}

]

},

"text":"GEORGE McGOVERN\n United States Senate\n Washington, D.C. 20510\n\n August 19, 1972\n\n The Honorable Harry S. Truman\n Independence, Missouri\n\n My dear Mr. President:\n\n It is an honor for me, as our party's\n candidate for President of the United States. . ."

},

"translations":{

"@total":"3",

"translation":[

{

"@langCode":"ita",

"@langName":"Italian",

"@lastModified":"2014-02-21T16:22:58",

"@isLocked":"true",

"@version":"232",

"lockedBy":{

"@user":"gwashington",

"@when":"2014-02-22T10:08:41"

},

"users":{

"@total":"3",

"user":[

{

"@user":"tjefferson",

"@fullName":"Tomás Jefferson",

"@isNaraStaff":"true",

"@lastModified":"2014-02-21T16:22:58",

"@version":"3"

},

{

"@user":"gwashington",

"@fullName":"Giorgio Washington",

"@isNaraStaff":"true",

"@lastModified":"2014-02-08T12:59:49",

"@version":"2"

}

]

},

"text":"George McGovern \n\n Senato degli Stati Uniti \n Washington, D.C. 20510 \n\n 19 ago 1972 \n\n L'Onorevole Harry S. Truman \n Independence,Missouri \n\n Mio caro Signor Presidente: \n\n E 'un onore per me, come il nostro partito di \n candidato alla Presidenza degli Stati Uniti. ."

},

{

"@langCode":"spa",

"@langName":"Spanish",

.

. Full information on the Spanish translation goes here

.

},

{

"@langCode":"deu",

"@langName":"German",

.

. Full information on the Spanish translation goes here

.

}

]

}

}

},

{

"@objectId":"2",

"designator":"2",

"file":{

"@purpose":"primary",

"@path":"./content/hst-ppp\_93-1\_18-02.jpg",

"@url":"https://research.archives.gov/id/7228539/content/hst-ppp\_93-1\_18-02.jpg",

"@mime":"image/jpeg",

"technicalMetadata":{

.

. Technical metadata for the primary file for the second object will go here

.

}

},

"thumbnail":{

"@path":"./opa-rendition/thumbnails/hst-ppp\_93-1\_18-02-thumb.jpg",

"@url":"https://research.archives.gov/id/7228539/opa-rendition/thumbnails/hst-ppp\_93-1\_18-02-thumb.jpg",

"@mime":"image/jpeg"

},

"imageTiles":{

"@path":"./opa-rendition/image-tiles/hst-ppp\_93-1\_18-02\_jpg"

"@url":"https://research.archives.gov/id/7228539/opa-rendition/image-tiles/hst-ppp\_93-1\_18-02\_jpg"

}

},

{

"@objectId":"3",

"designator":"3",

"file":{

"@purpose":"primary",

"@path":"./content/hst-ppp\_93-1\_18-03.jpg",

"@url":"https://research.archives.gov/id/7228539/content/hst-ppp\_93-1\_18-03.jpg",

"@mime":"image/jpeg",

"technicalMetadata":{

.

. Technical metadata for the primary file for the second object will go here

.

}

},

"thumbnail":{

"@path":"./opa-rendition/thumbnails/hst-ppp\_93-1\_18-03-thumb.jpg",

"@url":"https://research.archives.gov/id/7228539/opa-rendition/thumbnails/hst-ppp\_93-1\_18-03-thumb.jpg",

"@mime":"image/jpeg"

},

"imageTiles":{

"@path":"./opa-rendition/image-tiles/hst-ppp\_93-1\_18-03\_jpg",

"@url":"https://research.archives.gov/id/7228539/opa-rendition/image-tiles/hst-ppp\_93-1\_18-03\_jpg",

"@format":"DZI"

}

}

]

}

# Public Contributions Metadata

The “publicContributions” field contains a complete representation of the contribution metadata for the description, digital object, or authority record which is retrieved by the search engine.

Public contributions metadata will be provided in four parts:

* comments – Lists of comments and replies
  + For more information see 7.1 below.
* tags – List of tags
  + For more information see 7.2 below.
* transcription – (only for digital objects) The object transcription
  + For more information see 7.3 below.
* transcriptions – (only for digital objects) All translations on the object
  + For more information see 7.4 below.

For a complete example, see section 7.5 below.

## Comments

The comments embedded in the public contributions metadata, will contain the following information.

* **@comments** – Total count of all comments in the public contributions.
* **@replies** – Total count of all replies in the public contributions.
* **@total** – Total count of all comments and replies in the public contributions.
* **comment** – Parent tag which contains the text and replies for a single comment.   
  Has the following sub-elements:
  + **@commentId** – The unique identifier of the comment.
  + **@user** – The user ID of the person who created the comment.
  + **@fullName** – The full name of the user.
    - The @fullName is only included if the user wishes for the full name to be shown in place of their user ID.
  + **@naraStaff** – “true” if the user is NARA staff. “false” otherwise.
  + **@created** – The date-time when the comment was last edited.
  + **@lastModified** – The date-time when the comment was last edited.
  + **@isDeleted** – “true” if the comment has been deleted. “false” otherwise.
  + **text** – The actual text of the comment.
  + **replies** – Container for the replies of the comment.   
    Has the following sub-elements:
    - **@total** – Total count of the replies for a comment.
    - **reply** – Holds the information for a single reply.   
      Has the following sub-elements:
      * **@replyId** – The unique identifier of the reply.
      * **@commentId** – The unique identifier of the comment to which the reply belongs.
      * **@user** – The user ID of the person who created the reply.
      * **@fullName** – The full name of the replying user.
        + The @fullName is only included if the user wishes for the full name to be shown in place of their user ID.
      * **@naraStaff** – “true” if the replying user is NARA staff. “false” otherwise.
      * **@created** – The date-time when the reply was created.
      * **@lastModified** – The date-time when the reply was last edited.
      * **@isDeleted** – “true” if the comment has been deleted. “false” otherwise.
      * **text** – The actual text of a reply to the comment.

XML Example

<comments comments="3" replies="2" total="5">

<comment commentId="19328" user="kmartin" created="2014-02-10T14:36:23"   
lastModified="2014-02-10T14:40:12">

<text>One wonders if this letter to Frm Pres. Truman was felt to be obligatory, or if there was real affection between the two.</text>

<replies>

<reply replyId="45239" commentId="19328" user="mstewart" fullName="Meredith Stewart" naraStaff="true" created="2014-02-10T15:00:03" lastModified="2014-02-10T15:00:03">

<text>True, but from different generations.</text>

</reply>

<reply replyId="45422" commentId="19328" user="mkoneni" created="2014-03-08T04:12:52"  
lastModified="2014-03-08T04:18:02">

<text>I'm sure they ran into each other a bunch, you know at charity functions and golf tournaments...</text>

</reply>

</replies>

</comment>

<comment commentId="19102" user="tjefferson" created="2014-03-08T04:12:52"   
lastModified="2014-03-08T04:12:52" isDeleted="true"/>

<comment commentId="17024" user="rbarnes" created="2014-03-09T11:14:27"  
lastModified="2014-03-08T04:12:52" >

<text>Seems rather formulaic to me. I bet a staff member wrote it.</text>

</comment>

</comments>

JSON Example

"comments":{

"@comments":"3",

"@replies":"2",

"@total":"5",

"comment":[

{

"@commentId":"19328",

"@user":"kmartin",

"@created":"2014-02-10T14:36:23",

"@lastModified":"2014-02-10T14:40:12",

"text":"One wonders if this letter to Frm Pres. Truman was felt to be obligatory, or if there was real affection between the two.",

"replies":{

"reply":[

{

"@replyId":"45239",

"@commentId":"19328",

"@user":"mstewart",

"@fullName":"Meredith Stewart",

"@naraStaff":"true",

"@created":"2014-02-10T15:00:03",

"@lastModified":"2014-02-10T15:00:03",

"text":"True, but from different generations."

},

{

"@replyId":"45422",

"@commentId":"19328",

"@user":"mkoneni",

"@created":"2014-03-08T04:12:52",

"@lastModified":"2014-03-08T04:18:02",

"text":"I'm sure they ran into each other a bunch, you know at charity functions and golf tournaments..."

}

]

}

},

{

"@commentId":"19102",

"@user":"tjefferson",

"@created":"2014-03-08T04:12:52",

"@lastModified":"2014-03-08T04:12:52",

"@isDeleted":"true"

},

{

"@commentId":"17024",

"@user":"rbarnes",

"@created":"2014-03-09T11:14:27",

"@lastModified":"2014-03-09T11:14:27",

"text":"Seems rather formulaic to me. I bet a staff member wrote it."

}

]

}

## Tags

Within the public contributions, the tags area contains the list of tags.

* **@total** – Total count of all tags in the public contributions metadata.
* **tag** – Detailed information on an individual tag. This contains the following sub-elements:
  + **@text** – The actual text of the tag.
  + **@user** – The user ID of the person who created the tag.
  + **@fullName** – The full name of the user.
    - The @fullName tag is only included if the user wishes for the full name to be shown in place of their user ID.
  + **@isNaraStaff** – “true” if the user is NARA staff. “false” otherwise.
  + **@created** – The date-time when the tag was created.

XML Example

<tags total="3">

<tag text="Truman Post President" user="jdoe" created="2014-02-21T16:22:58"/>

<tag text="McGovern Letters" user="agullet" created="2014-02-27T18:23:47"/>

<tag text="nara:president=Truman" user="mstewart" fullName="Meredith Stewart" naraStaff="true" created="2014-03-19T07:07:42"/>

</tags>

JSON Example

{

"@total":"3",

"tag":[

{

"@text":"Truman Post President",

"@user":"jdoe",

"@created":"2014-02-21T16:22:58"

},

{

"@text":"McGovern Letters",

"@user":"agullet",

"@created":"2014-02-27T18:23:47"

},

{

"@text":"nara:president=Truman",

"@user":"mstewart",

"@fullName":"Meredith Stewart",

"@naraStaff":"true",

"@created":"2014-03-19T07:07:42"

}

]

}

## Transcriptions

The following metadata will be returned within the public contributions for objects with transcriptions.

* **@lastModified** – The date-time when the transcription was last modified.
* **@version** – The version number of the latest transcription.
  + This number will increment for every modification (including deletes and restores) made to the transcription.
* **users** – Parent tag which contains a list of the users who have contributed to this transcription. Contains the following sub-elements:
  + **@total** – Total number of unique users who have contributed to the transcription.
  + **user** – Information on an individual user who helped contribute to the transcription. Contains the following sub-elements:
    - **@user** – The user ID of the user (for example, “kmartin”).
    - **@fullName** – The full name of the user. Attribute is only present if the user wishes to make their full name public.
    - **@isNaraStaff** – “true” if the user is NARA staff. “false” otherwise.
    - **@lastModified** – The date-time when this user last modified the transcription.
    - **@version** – The version number most recently contributed by the user.
* **text** – The actual text of the transcription.

Note that users will be automatically sorted by @lastModified date (most recent modifier first).

XML Example

<transcription lastModified="2014-02-21T16:22:58" version="232">

<users total="3">

<user user="tjefferson" fullName="Thomas Jefferson" isNaraStaff="true"   
lastModified="2014-02-21T16:22:58" version="13"/>

<user user="mstewart" fullName="Meredith Stewart" isNaraStaff="true"   
lastModified="2014-02-08T12:59:49" version="12"/>

<user user="kmartin" lastModified="2014-02-02T08:44:12" version="10"/>

</users>

<text>GEORGE McGOVERN

United States Senate

Washington, D.C. 20510

August 19, 1972

The Honorable Harry S. Truman

Independence, Missouri

My dear Mr. President:

It is an honor for me, as our party's

candidate for President of the United States. . .</text>

</transcription>

JSON Example

{

"@lastModified":"2014-02-21T16:22:58",

"@version":"232",

"users":{

"@total":"3",

"user":[

{

"@user":"tjefferson",

"@fullName":"Thomas Jefferson",

"@isNaraStaff":"true",

"@lastModified":"2014-02-21T16:22:58",

"@version":"13"

},

{

"@user":"mstewart",

"@fullName":"Meredith Stewart",

"@isNaraStaff":"true",

"@lastModified":"2014-02-08T12:59:49"

"@version":"12"

},

{

"@user":"kmartin",

"@lastModified":"2014-02-02T08:44:12"

"@version":"10"

}

]

},

"text":"GEORGE McGOVERN\nUnited States Senate\nWashington, D.C. 20510\n\nAugust 19, 1972\n\nThe Honorable Harry S. Truman\nIndependence, Missouri\n\nMy dear Mr. President:\n\n It is an honor for me, as our party's\ncandidate for President of the United States. . ."

}

## Translations

Translations will be returned for objects within the public contributions. The metadata for each translation will be as follows:

* **@langCode** – The language code of the language.
  + This is the ISO 639-3 (three-letter) code for the language.
  + See section 9.3 for a list of the languages supported by NARA.
* **@langName** – The English name for the language from the table in section 9.3.
* **@lastModified** – The date-time when the translation was last modified.
* **@version** – The version number of the latest translation.
  + This number will increment for every modification (including deletes and restores) made to the translation.
* **users** – Parent tag which contains a list of the users who have contributed to this translation. Contains the following sub-elements:
  + **@total** – Total number of unique users who have contributed to the translation.
  + **user** – Information on an individual user who helped contribute to the translation. Contains the following sub-elements:
    - **@user** – The user ID of the user (for example, “kmartin”).
    - **@fullName** – The full name of the user. Attribute is only present if the user wishes to make their full name public.
    - **@isNaraStaff** – “true” if the user is NARA staff. “false” otherwise.
    - **@lastModified** – The date-time when this user last modified the translation.
* **text** – The actual text of the translation.

Note that users will be automatically sorted by @lastModified date (most recent modifier first).

XML Example

<translation langCode="ita" langName="Italian" lastModified="2014-02-21T16:22:58" version="232">

<users total="3">

<user user="tjefferson" fullName="Tomás Jefferson" isNaraStaff="true"   
lastModified="2014-02-21T16:22:58" version="3"/>

<user user="gwashington" fullName="Giorgio Washington" isNaraStaff="true"  
lastModified="2014-02-08T12:59:49" version="2"/>

</users>

<text>George McGovern

Senato degli Stati Uniti

Washington, D.C. 20510

19 ago 1972

L'Onorevole Harry S. Truman

Independence, Missouri

Mio caro Signor Presidente:

E 'un onore per me, come il nostro partito di

candidato alla Presidenza degli Stati Uniti. .</text>

</translation>

JSON Example

{

"@langCode":"ita",

"@langName":"Italian",

"@lastModified":"2014-02-21T16:22:58",

"@version":"232",

"users":{

"@total":"3",

"user":[

{

"@user":"tjefferson",

"@fullName":"Tomás Jefferson",

"@isNaraStaff":"true",

"@lastModified":"2014-02-21T16:22:58",

"@version":"3"

},

{

"@user":"gwashington",

"@fullName":"Giorgio Washington",

"@isNaraStaff":"true",

"@lastModified":"2014-02-08T12:59:49",

"@version":"2"

}

]

},

"text":"George McGovern \n\nSenato degli Stati Uniti \nWashington, D.C. 20510 \n\n19 ago 1972 \n\nL'Onorevole Harry S. Truman \nIndependence, Missouri \n\nMio caro Signor Presidente: \n\n E 'un onore per me, come il nostro partito di \ncandidato alla Presidenza degli Stati Uniti. ."

}

## Complete Example

XML Example

<publicContributions>

<comments comments="3" replies="2" total="5">

<comment commentId="19328" user="kmartin" created="2014-02-10T14:36:23" lastModified="2014-02-10T14:40:12">

<text>One wonders if this letter to Frm Pres. Truman was felt to be obligatory, or if there was real affection between the two.</text>

<replies>

<reply replyId="45239" commentId="19328" user="mstewart" fullName="Meredith Stewart" naraStaff="true" created="2014-02-10T15:00:03" lastModified="2014-02-10T15:00:03">

<text>True, but from different generations.</text>

</reply>

<reply replyId="45422" commentId="19328" user="mkoneni" created="2014-03-08T04:12:52" lastModified="2014-03-08T04:18:02">

<text>I'm sure they ran into each other a bunch, you know at charity functions and golf tournaments...</text>

</reply>

</replies>

</comment>

<comment commentId="19102" user="tjefferson" created="2014-03-08T04:12:52"   
lastModified="2014-03-08T04:12:52" isDeleted="true"/>

<comment commentId="17024" user="rbarnes" created="2014-03-09T11:14:27"   
lastModified="2014-03-09T11:14:27">

<text>Seems rather formulaic to me. I bet a staff member wrote it.</text>

</comment>

</comments>

<tags total="3">

<tag text="Truman Post President" user="jdoe" created="2014-02-21T16:22:58"/>

<tag text="McGovern Letters" user="agullet" created="2014-02-27T18:23:47"/>

<tag text="nara:president=Truman" user="mstewart" fullName="Meredith Stewart" naraStaff="true" created="2014-03-19T07:07:42"/>

</tags>

<transcription lastModified="2014-02-21T16:22:58" version="232">

<users total="3">

<user user="tjefferson" fullName="Thomas Jefferson" isNaraStaff="true" lastModified="2014-02-21T16:22:58" version="13"/>

<user user="mstewart" fullName="Meredith Stewart" isNaraStaff="true" lastModified="2014-02-08T12:59:49" version="12"/>

<user user="kmartin" lastModified="2014-02-02T08:44:12" version="10"/>

</users>

<text>GEORGE McGOVERN

United States Senate

Washington, D.C. 20510

August 19, 1972

The Honorable Harry S. Truman

Independence, Missouri

My dear Mr. President:

It is an honor for me, as our party's

candidate for President of the United States. . .</text>

</transcription>

<translations total="2">

<translation langCode="ita" langName="Italian" lastModified="2014-02-21T16:22:58"  
version="232">

<users total="2">

<user user="tjefferson" fullName="Tomás Jefferson" isNaraStaff="true" lastModified="2014-02-21T16:22:58" version="3"/>

<user user="gwashington" fullName="Giorgio Washington" isNaraStaff="true" lastModified="2014-02-08T12:59:49" version="2"/>

</users>

<text>George McGovern

Senato degli Stati Uniti

Washington, D.C. 20510

19 ago 1972

L'Onorevole Harry S. Truman

Independence, Missouri

Mio caro Signor Presidente:

E 'un onore per me, come il nostro partito di

candidato alla Presidenza degli Stati Uniti. .</text>

</translation>

<translation langCode="spa" langName="Spanish">

<users total="1">

<user user="cwashington" fullName="Carlos Washington" isNaraStaff="true" lastModified="2014-02-08T12:59:49" version="4"/>

</users>

<text>George McGovern

Senado de Estados Unidos

Washington, DC 20510

19 de agosto 1972

El Honorable Harry S. Truman

Independence, Missouri

Mi querido Señor Presidente:

Es un honor para mí, como nuestro grupo de

candidato a Presidente de los Estados Unidos. . .</text>

</translation>

</translations>

</publicContributions>

JSON Example

{

"comments":{

"@comments":"3",

"@replies":"2",

"@total":"5",

"comment":[

{

"@commentId":"19328",

"@user":"kmartin",

"@created":"2014-02-10T14:36:23",

"@lastModified":"2014-02-10T14:40:12",

"text":"One wonders if this letter to Frm Pres. Truman was felt to be obligatory, or if there was real affection between the two.",

"replies":{

"reply":[

{

"@replyId":"45239",

"@commentId":"19328",

"@user":"mstewart",

"@fullName":"Meredith Stewart",

"@naraStaff":"true",

"@created":"2014-02-10T15:00:03",

"@lastModified":"2014-02-10T15:00:03",

"text":"True, but from different generations."

},

{

"@replyId":"45422",

"@commentId":"19328",

"@user":"mkoneni",

"@created":"2014-03-08T04:12:52",

"@lastModified":"2014-03-08T04:18:02",

"text":"I'm sure they ran into each other a bunch, you know at charity functions and golf tournaments..."

}

]

}

},

{

"@commentId":"19102",

"@user":"tjefferson",

"@created":"2014-03-08T04:12:52",

"@lastModified":"2014-03-08T04:12:52",

"@isDeleted":"true"

},

{

"@commentId":"17024",

"@user":"rbarnes",

"@created":"2014-03-09T11:14:27",

"@lastModified":"2014-03-09T11:14:27",

"text":"Seems rather formulaic to me. I bet a staff member wrote it."

}

]

},

"tags":{

"@total":"3",

"tag":[

{

"@text":"Truman Post President",

"@user":"jdoe",

"@created":"2014-02-21T16:22:58"

},

{

"@text":"McGovern Letters",

"@user":"agullet",

"@created":"2014-02-27T18:23:47"

},

{

"@text":"nara:president=Truman",

"@user":"mstewart",

"@fullName":"Meredith Stewart",

"@naraStaff":"true",

"@created":"2014-03-19T07:07:42"

}

]

},

"transcription":{

"@lastModified":"2014-02-21T16:22:58",

"@version":"232",

"users":{

"@total":"3",

"user":[

{

"@user":"tjefferson",

"@fullName":"Thomas Jefferson",

"@isNaraStaff":"true",

"@lastModified":"2014-02-21T16:22:58",

"@version":"13"

},

{

"@user":"mstewart",

"@fullName":"Meredith Stewart",

"@isNaraStaff":"true",

"@lastModified":"2014-02-08T12:59:49",

"@version":"12"

},

{

"@user":"kmartin",

"@lastModified":"2014-02-02T08:44:12",

"@version":"10"

}

]

},

"text":"GEORGE McGOVERN\n United States Senate\n Washington, D.C. 20510\n\n August 19, 1972\n\n The Honorable Harry S. Truman\n Independence, Missouri\n\n My dear Mr. President:\n\n It is an honor forme, as our party's\n candidate for President of the United States. . ."

},

"translations":{

"@total":"2",

"translation":[

{

"@langCode":"ita",

"@langName":"Italian",

"@lastModified":"2014-02-21T16:22:58",

"@version":"232",

"users":{

"@total":"2",

"user":[

{

"@user":"tjefferson",

"@fullName":"Tomás Jefferson",

"@isNaraStaff":"true",

"@lastModified":"2014-02-21T16:22:58",

"@version":"3"

},

{

"@user":"gwashington",

"@fullName":"Giorgio Washington",

"@isNaraStaff":"true",

"@lastModified":"2014-02-08T12:59:49",

"@version":"2"

}

]

},

"text":"George McGovern \n\n Senato degli Stati Uniti \n Washington, D.C. 20510 \n\n 19 ago 1972 \n\n L'Onorevole Harry S. Truman \n Independence, Missouri \n\n Mio caro Signor Presidente: \n\n E 'un onore per me, come il nostro partito di \n candidato alla Presidenza degli Stati Uniti. ."

},

{

"@langCode":"spa",

"@langName":"Spanish",

"users":{

"@total":"1",

"user":{

"@user":"cwashington",

"@fullName":"Carlos Washington",

"@isNaraStaff":"true",

"@lastModified":"2014-02-08T12:59:49",

"@version":"4"

}

},

"text":"George McGovern \n Senado de Estados Unidos \n Washington, DC 20510 \n\n 19 de agosto 1972 \n\n El Honorable Harry S. Truman \n Independence, Missouri \n\n Mi querido Señor Presidente: \n\n Es un honor para mí, como nuestro grupo de \n candidato a Presidente de los Estados Unidos. . ."

}

]

}

}

# Bulk Import

This section provides information on how to bulk-import public contributions into OPA. The URL for bulk imports will be:

POST https://research.archives.gov/api/v1/bulk-imports

Instead of name=value parameters, bulk imports require JSON data to be POST’ed to the URL. The JSON to be posted is shown in the sub-sections below.

All fields specified below are required unless specified otherwise.

## Bulk Import Comments

When bulk importing comments, the following elements are expected:

* **comments** – Identifies that this is a list of comments.
* **naId** – Specify the NARA ID for the comment.
  + NOTE: No “type” is required (e.g. “person”, “organization”, “description”, etc.).
* **objectId** – [OPTIONAL] Specify the object ID (within the specified NAID) of the object to which the comment is attached.
* **text** – Specifies the text of the comment.

JSON Example:

{

"comments":[

{

"naId":"19328",

"text":"One wonders if this letter to Frm Pres. Truman was felt to be obligatory, or if there was real affection between the two."

},

{

"naId":"17024",

"objectId":"32",

"text":"Seems rather formulaic to me. I bet a staff member wrote it."

}

]

}

## Bulk Import Tags

When bulk importing tags, the following elements are expected:

* **tags** – Identifies that this is a list of tags.
* **naId** – Specify the NARA ID for the tag.
  + NOTE: No “type” is required (e.g. “person”, “organization”, “description”, etc.).
* **objectId** – [OPTIONAL] Specify the object ID (within the specified NAID) of the object to which the tag is attached.
* **text** – Specifies the text of the tag.

JSON Example

{

"tags":[

{

"naId":"19328",

"text":"Truman Post President"

},

{

"naId":"19328",

"text":"McGovern Letters"

},

{

"naId":"17024",

"objectId":"17024",

"text":"nara:president=Truman"

}

]

}

## Bulk Import Transcriptions

When bulk importing transcriptions, the following elements are expected:

* **transcriptions** – Identifies that this is a list of transcriptions.
* **naId** – Specify the NARA ID for the transcription.
  + NOTE: No “type” is required (e.g. “person”, “organization”, “description”, etc.).
* **objectId** – Specify the object ID (within the specified NAID) of the object to which the transcription is attached. Note: For transcriptions, the objectId is required.
* **text** – Specifies the text of the transcription.

JSON Example

{

"transcriptions":[

{

"naId":"19328",

"objectId":"1",

"text":"GEORGE McGOVERN\n United States Senate\n Washington, D.C. 20510\n\n August 19, 1972\n\n The Honorable Harry S. Truman\n Independence, Missouri\n\n My dear Mr. President:\n\n It is an honor forme, as our party's\n candidate for President of the United States. . ."

},

{

"naId":"19328",

"objectId":"2",

"text":"GEORGE McGOVERN\n United States Senate\n Washington, D.C. 20510\n AIRMAIL"

}

]

}

## Bulk Import Translations

When bulk importing translations, the following elements are expected:

* **translations** – Identifies that this is a list of translations.
* **naId** – Specify the NARA ID for the translation.
  + NOTE: No “type” is required (e.g. “person”, “organization”, “description”, etc.).
* **objectId** – Specify the object ID (within the specified NAID) of the object to which the translation is attached. Note: For translations, the objectId is required.
* **langCode** – Specify the ISO-639-3 (three-letter) language code of the language of the translation.
* **text** – Specifies the text of the translation.

JSON Example

{

"translations":[

{

"naId":"19328",

"objectId":"1",

"langCode":"ita",

"text":"George McGovern \n\n Senato degli Stati Uniti \n Washington, D.C. 20510 \n\n 19 ago 1972 \n\n L'Onorevole Harry S. Truman \n Independence, Missouri \n\n Mio caro Signor Presidente: \n\n E 'un onore per me, come il nostro partito di \n candidato alla Presidenza degli Stati Uniti. ."

},

{

"naId":"19328",

"objectId":"1",

"langCode":"spa",

"text":"George McGovern \n Senado de Estados Unidos \n Washington, DC 20510 \n\n 19 de agosto 1972 \n\n El Honorable Harry S. Truman \n Independence, Missouri \n\n Mi querido Señor Presidente: \n\n Es un honor para mí, como nuestro grupo de \n candidato a Presidente de los Estados Unidos. . ."

}

]

}

# Appendices

## Location IDs

The following is the list of all location ID values.

| **Location Value** | **Search Value (ref-id)** |
| --- | --- |
| William J. Clinton Library | 1 |
| Dwight D. Eisenhower Library | 2 |
| Franklin D. Roosevelt Library | 3 |
| George Bush Library | 4 |
| Gerald R. Ford Library | 5 |
| Gerald R. Ford Museum | 6 |
| Herbert Hoover Library | 7 |
| Harry S. Truman Library | 8 |
| Jimmy Carter Library | 9 |
| John F. Kennedy Library | 10 |
| Lyndon Baines Johnson Library | 11 |
| Richard Nixon Library - College Park | 12 |
| Ronald Reagan Library | 13 |
| National Archives at Boston | 14 |
| National Archives at New York | 15 |
| National Archives at Philadelphia | 17 |
| National Archives at Atlanta | 18 |
| National Archives at Chicago | 19 |
| National Archives at Kansas City | 20 |
| National Archives at Fort Worth | 21 |
| National Archives at Denver | 22 |
| National Archives at Riverside | 23 |
| National Archives at San Francisco | 24 |
| National Archives at Anchorage | 25 |
| National Archives at Seattle | 26 |
| National Personnel Records Center - Civilian Personnel Records | 27 |
| National Personnel Records Center - Military Personnel Records | 28 |
| National Archives - Washington, DC – Cartographic | 29 |
| National Archives - Washington, DC - Motion Pictures | 30 |
| National Archives - Washington, DC - Still Pictures | 31 |
| National Archives - Washington, DC - Archives I Textual Reference | 32 |
| National Archives - Washington, DC - Archives II Textual Reference (Civilian) | 33 |
| National Archives - Washington, DC – FOIA | 34 |
| National Archives - Washington, DC - Archives II Textual Reference (Military) | 35 |
| Center for Legislative Archives | 36 |
| National Archives - Washington, DC – Electronic | 37 |
| Library of Congress, Prints and Photographs Division (an affiliated archives) | 38 |
| National Park Service, Yellowstone National Park Archives (an affiliated archives) | 39 |
| New Mexico Commission of Public Records, State Records Center and Archives (an affiliated archives) | 40 |
| Oklahoma Historical Society (an affiliated archives) | 41 |
| Pennsylvania Historical and Museum Commission, State Archives (an affiliated archives) | 42 |
| United States Military Academy Archives (an affiliated archives) | 43 |
| United States Naval Academy, William W. Jeffries Memorial Archives (an affiliated archives) | 44 |
| Presidential Materials Division | 48 |
| National Archives at St. Louis | 50 |
| Richard Nixon Library | 51 |
| George W. Bush Library | 53 |
| U.S. Government Printing Office (an affiliated archives) | 54 |
| University of North Texas Libraries (an affiliated archives) | 57 |

## Technical Metadata

This section identifies metadata fields in the <technicalMetadata> section for each type of file in the objects XML. See section 6 for more information on the Objects XML.

### Fields Common to All Types

* ***createDate*** – The date and time the file was created, in [ISO 8601](http://en.wikipedia.org/wiki/ISO_8601) format.
* ***modifyDate*** – The date and time the file was last modified, in [ISO 8601](http://en.wikipedia.org/wiki/ISO_8601) format.
* ***size*** – The size of the file, in bytes.
* ***mime*** – The IANA mime type for the file.

### Technical Metadata Fields for Each Type

The following are the technical metadata fields for each type of document.

*Note: All metadata fields are optional. Only metadata fields which actually exist within the document will be extracted.*

* application/pdf
  + ***Page-Count*** – The number of pages in the PDF. Note that the page count is the number of PDF pages, including all front matter (title page, table of contents, etc.) and back matter (indexes).
  + ***Creation-Date*** – The date the file was created, in ISO 8601 format, as stamped in the file metadata. For example, "2014-03-07T02:57:00Z".
  + ***Last-Modified*** – The date the file was last modified, as stamped in the file metadata in ISO 8601 format. For example, "2014-03-16T15:25:00Z".
  + ***Title*** – The title of the file, as stamped in the file’s document properties. For example, "Search Technologies Assessment".
* text/plain
  + No additional technical metadata fields for plain text files.
* text/html
  + All of the fields specified in <META> of the HTML file will be included as technical metadata.
  + ***keywords*** – Search keywords
  + ***description*** – Describes the page itself
  + ***language*** – Language of content.
* video/mp4
  + ***General\_Format*** – The overall file format. For example, “MPEG-4”.
  + ***General\_Format\_profile*** – The format profile and version. For example, “Base Media / Version 2”.
  + ***General\_Codec\_ID*** – The overall description of the codec. For example, “mp42”.
  + ***General\_Duration*** – A string representation of the duration. For example, “3mn 52s”.
  + ***General\_Overall\_bit\_rate\_mode*** – Overall bitrate format. For example, “Variable”.
  + ***General\_Overall\_bit\_rate*** – Average bitrate across the video clip as a string. For example, “13.5 Mbps”.
  + ***Video\_Format*** – Overal video format codec. For example, “AVC”.
  + ***Video\_Format\_Info*** – Short description of the overall video code. For example, “Advanced Video Codec”.
  + ***Video\_Format\_profile*** – Profile for the video codec. For example, “Main@L4.2”.
  + ***Video\_Format\_settings\_\_CABAC*** – Is “Context-adaptive binary arithmetic coding” used for the video? For example, “Yes”.
  + ***Video\_Format\_settings\_\_ReFrames*** – Number of reference frames. For example, “3 frames”.
  + ***Video\_Codec\_ID*** – Video codec ID. For example, “avc1”.
  + ***Video\_Codec\_ID\_Info*** – Short description of the video code. For example, “Advanced Video Coding”.
  + ***Video\_Duration*** – Video duration, expressed as a string. For example, “3mn 52s”.
  + ***Video\_Bit\_rate\_mode*** – Bit rate mode. For example, “Variable”.
  + ***Video\_Bit\_rate*** – Average video bit rate, expressed as a string. For example, “13.4 Mbps”.
  + ***Video\_Maximum\_bit\_rate*** – Maximum video bit rate (for variable bit rates). For example, “15.0 Mbps”.
  + ***Video\_Width*** – Width of video in pixels. For example, “1 920 pixels”.
  + ***Video\_Height*** – Height of video in pixels. For example, “1 080 pixels”.
  + ***Video\_Display\_aspect\_ratio*** – Video aspect ratio. For example, “16:9”.
  + ***Video\_Frame\_rate\_mode*** – Video frame rate mode. For example, “Constant”.
  + ***Video\_Frame\_rate*** – Overall video frame rate. For example, “25.000 fps”.
  + ***Video\_Standard*** – Video interlace standard. For example, “PAL” or “NTSC”.
  + ***Video\_Color\_space*** – Video color space model. For example, “YUV”.
  + ***Video\_Chroma\_subsampling*** – Video color subsampling for compression. For example, “4:2:0”.
  + ***Video\_Bit\_depth*** – Video bit dept. For example, “8 bits”.
  + ***Video\_Scan\_type*** – Video line scan method. For example, “Progressive”.
  + ***Video\_Stream\_size*** – Size of the video stream compared to the file overall. For example, “371 MiB (99%)”.
  + ***Video\_Language*** – Language of the video. For example, “English”.
  + ***Audio\_Format*** – Higher level audio format codec. For example, “AAC”.
  + ***Audio\_Format\_Info*** – Higher level audio format, short description. For example, “Advanced Audio Codec”.
  + ***Audio\_Format\_profile*** – Audio format profile. For example, “LC”.
  + ***Audio\_Codec\_ID*** – Detailed audio codec. For example, “40”.
  + ***Audio\_Duration*** – Duration of the audio expressed as a string. For example, “3mn 52s”.
  + ***Audio\_Bit\_rate\_mode*** – Bit rate mode for audio. For example, “Variable”.
  + ***Audio\_Bit\_rate*** – Overall bit rate for audio. For example, “157 Kbps”.
  + ***Audio\_Maximum\_bit\_rate*** – Maximum bit rate for audio. For example, “249 Kbps”.
  + ***Audio\_Channel\_s\_*** – Number of audio channels. For example, “2 channels” for stereo.
  + ***Audio\_Channel\_positions*** – Positions of the audio sources in respect to the audience. For example, “Front: L R”.
  + ***Audio\_Sampling\_rate*** – Audio sampling rate. For example, “48.0 KHz”.
  + ***Audio\_Compression\_mode*** – Audio compression mode. For example, “Lossy”.
  + ***Audio\_Stream\_size*** – Overall size of the audio stream in relation to the rest of the file, expressed as a string. For example, “4.37 MiB (1%)”.
  + ***Audio\_Language*** – Language of the audio. For example, “English”.
* video/x-ms-wmv
  + ***General\_Duration\_String*** – Duration of the video as a string. For example, “2s 6ms”
  + ***General\_OverallBitRate\_Mode\_String*** – Overall bitrate mode. For example, “Constant”
  + ***General\_OverallBitRate\_String*** – Specifies the overall or average bitrate for the video. For example, “6435 Kbps”.
  + **General\_OverallBitRate\_Maximum\_String** – Maximum bitrate. For example, “8110 Kbps”.
  + ***Video\_Format*** – High level video format. For example, “VC-1”.
  + ***Video\_Format\_Profile*** – Profile for the video format. For example, “MP@ML”
  + ***Video\_CodecID*** – Detailed code ID. For example, “WMV3”
  + ***Video\_CodecID\_Info*** – Codec short description. For example, “Windows Media Video 9”
  + ***Video\_CodecID\_Hint*** – Additional Codec hint. Often the same as the codec ID. For example, “WMV3”.
  + ***Video\_CodecID\_Description*** – Codec long description. For example, “Windows Media Video 9 - Professional”
  + ***Video\_Duration\_String*** – Video duration as a string. For example, “2s 0ms”
  + ***Video\_BitRate\_Mode\_String*** – Video bitrate mode. For example, “Constant”
  + ***Video\_BitRate\_String*** – Video bitrate expressed as a string. For example, “8 000 Kbps”
  + ***Video\_Width\_String*** – Width of the video. For example, “720 pixels”
  + ***Video\_Height\_String*** – Height of the video. For example, “576 pixels”
  + ***Video\_DisplayAspectRatio\_String*** – Aspect ratio of the video.  
    For example, “5:4”
  + ***Video\_FrameRate\_String*** – Frames rate for the video. For example, “25.000 fps”.
  + ***Video\_Standard*** – The video interlacing standard. For example, “PAL” or “NTSC”.
  + ***Video\_ColorSpace*** – Video color space algorithm. For example, “YUV”
  + ***Video\_ChromaSubsampling*** – Video color compression method. For example, “4:2:0”
  + ***Video\_BitDepth\_String*** – Bit depth. For example, “8 bits”.
  + ***Video\_ScanType\_String*** – Video line scan type. For example, “Progressive”
  + ***Video\_Compression\_Mode\_String*** – Video compression mode. For example, “Lossy”
  + ***Video\_StreamSize\_String*** – Video stream size, expressed as a string. For example, “1.91 MiB”
  + ***Audio\_Format*** – Audio high-level code. For example, “WMA”
  + ***Audio\_Format\_Version*** – Audio high level code version. For example, “Version 2”
  + ***Audio\_CodecID*** – Audio codec ID. For example, “161”
  + ***Audio\_CodecID\_Info*** – Audio codec short description. For example, “Windows Media Audio”
  + ***Audio\_CodecID\_Description*** – Audio code long description. For example, “Windows Media Audio 9.1 - 48 kbps, 44 kHz, stereo 1-pass CBR”
  + ***Audio\_Duration\_String*** – Audio duration, expressed as a string. For example, “2s 6ms”
  + ***Audio\_BitRate\_Mode\_String*** – Audio bitrate mode. For example, “Constant”
  + ***Audio\_BitRate\_String*** – Audio bitrate, expressed as a string. For example, “48.0 Kbps”
  + ***Audio\_Channel\_s\_\_String*** – Number of audio channels. For example, “2 channels” for stereo.
  + ***Audio\_SamplingRate\_String*** – Audio sampling rate (audio samples per second). For example, “44.1 KHz”
  + ***Audio\_BitDepth\_String*** – Number of bits per audio sample. For example, “16 bits”
  + ***Audio\_StreamSize\_String*** – Audio stream size, expressed as a string. For example, “11.8 KiB (1%)”
* application/mswrite
  + No additional technical metadata fields for Microsoft Write fies.
* application/excel
  + ***Application-Name*** – The name of the application which created the file. For example, "Microsoft Excel"
  + ***Application-Version*** – The version number of the application which created the file. For example, "14.0300".
  + ***Creation-Date*** – The date the file was created, in ISO 8601 format, as stamped in the file metadata. For example, "2014-03-07T02:57:00Z".
  + ***Last-Modified*** – The date the file was last modified, as stamped in the file metadata in ISO 8601 format. For example, "2014-03-16T15:25:00Z".
* application/msword
  + ***Application-Name*** – The name of the application which created the file. For example, "Microsoft Office Word"
  + ***Application-Version*** – The version number of the application which created the file. For example, "14.0000".
  + ***Word-Count*** – The number of words in the file. For example, "55087".
  + ***Paragraph-Count*** – The number of paragraphs in the file. For example, "736".
  + ***Creation-Date*** – The date the file was created, in ISO 8601 format, as stamped in the file metadata. For example, "2014-03-07T02:57:00Z".
  + ***Last-Modified*** – The date the file was last modified, as stamped in the file metadata in ISO 8601 format. For example, "2014-03-16T15:25:00Z".
  + ***Title*** – The title of the file, as stamped in the file’s document properties. For example, "Search Technologies Assessment".
  + ***Character-Count*** – The total character count in the document. For example, "314002".
* application/mspowerpoint
  + ***Slide-Count*** – Count of slides in the presentation. For example, “5”.
  + ***Application-Name*** – The name of the application which created the file. For example, "Microsoft Office PowerPoint".
  + ***Application-Version***– The version number of the application which created the file. For example, "14.0000".
  + ***Word-Count*** – The number of words in the file. For example, "153".
  + ***Creation-Date***– The date the file was created, in ISO 8601 format, as stamped in the file metadata. For example, "2014-04-19T23:29:53Z".
  + ***Paragraph-Count*** – The number of paragraphs in the file. For example, "75".
  + ***Last-Modified*** – The date the file was last modified, as stamped in the file metadata in ISO 8601 format. For example, "2014-04-21T22:13:55Z".
  + ***PresentationFormat*** – The presentation format and aspect ratio. For example, "On-screen Show (4:3)"
  + ***Title*** – The title of the presentation, as stamped in the file’s document properties. For example, “PowerPoint Presentation”.
* Image (BMP); image/bmp
  + ***Data\_BitsPerSample*** – The number of bits for each sample per pixel. Typically three integers, such as “8 8 8”.
  + ***height*** – Height of the image in pixels
  + ***width*** – Width of the image in pixels
  + ***Compression\_CompressionTypeName*** – The type of compression used for the image. Typically “BI\_RGB”.
* Image (GIF); image/gif
  + ***Compression\_Lossless*** – TRUE/FALSE
  + ***width*** – Width of image in pixels
  + ***height*** – Height of image in pixels
  + ***Dimension\_ImageOrientation*** – Image orientation. Typically “Normal”.
  + ***Compression\_CompressionTypeName*** – Type of image compression.   
    Typically “lzw”.
  + ***Dimension\_HorizontalPixelOffset*** - The width of a pixel, in millimeters, as it should be rendered on media.
  + ***Dimension\_VerticalPixelOffset*** – The height of a pixel, in millimeters, as it should be rendered on media.
  + ***Chroma\_NumChannels*** - The number of channels in the raw image, including alpha
  + ***Compression\_NumProgressiveScans*** – The number of progressive scans used in the image encoding
  + ***Chroma\_ColorSpaceType*** – The raw color space of the image.
  + ***Chroma\_BlackIsZero*** – True if smaller values represent darker shades.
* Image (JPG); image/jpeg
  + ***metadataDate*** – The date of the EXIF metadata.
  + ***width*** – Width of the image in pixels
  + ***height*** – Height of the image in pixels
  + ***resolution*** – Resolution of the image
    - ***@x*** – X resolution
    - ***@y*** – Y resolution
    - ***@units*** – The units for the X and Y dimensions (for example, “pixels/inch")
  + ***bitsPerSample*** – The number of bits for each sample
    - Three numbers, bits for each of R, G, B
  + ***photometricInterpretation*** – The pixel composition.
  + ***orientation*** – The orientation of the image. For example, “horizontal”.
  + ***samplesPerPixel*** – The number of components per pixel. Since this standard applies to RGB and YCbCr images, the value set for this tag is 3.
  + ***planarConfiguration*** – Indicates whether pixel components are recorded in a chunky or planar format. For example, “chunky”.
  + ***colorSpace*** – The color space information tag is always recorded as the color space specifier. Normally sRGB is used to define the color space based on the PC monitor conditions and environment.
  + ***compression*** – The compression scheme used for the image data. For example, “uncompressed”
  + ***software*** – The name and version of the software used to post-process the picture. For example, “Adobe Photoshop CS6 (Windows)”.
* Image (JPEG2000); image/jp2
  + ***metadataDate*** – The date of the metadata.
  + ***width*** – Width of the image in pixels
  + ***height*** – Height of the image in pixels
  + ***resolution*** – Resolution of the image
    - ***@x*** – X resolution
    - ***@y*** – Y resolution
    - ***@units*** – The units for the X and Y dimensions (for example, “pixels/inch" or “dots/inch”)
  + ***bitsPerSample*** – The number of bits for each sample. For JPEG 2000, usually a single number, such as “8”.
  + ***Exposure\_Mode*** – The exposure mode of the camera. For example, "Auto exposure".
  + ***Color\_Space*** - The color space, if know. “Unknown” otherwise.
  + ***Compression\_Type*** – The type of compression. For example, “Baseline”.
  + ***Number\_of\_Components*** – Number of color components. For example, “3”.
  + ***Component\_1*** – Detailed information on the first color component. For example, “Y component: Quantization table 0, Sampling factors 2 horiz/2 vert”
  + ***Component\_2*** – Detailed information on the second color component. For example, “Cb component: Quantization table 1, Sampling factors 1 horiz/1 vert”
  + ***Component\_3*** – Detailed information on the third color component. For example, “Cr component: Quantization table 1, Sampling factors 1 horiz/1 vert”
* Image (TIFF); image/tiff
  + ***Rows\_Per\_Strip*** – Number of rows in each individually compressed image strip. For example, “25”.
  + ***Bits\_Per\_Sample*** – Number of bits per sample. For example, “16”.
  + ***Compression*** – Compression algorithm. For example, “Adobe Deflate".
  + ***Planar\_Configuration*** – Indicates whether pixel components are recorded in a chunky or planar format. For example, “Chunky”.
  + ***width*** – Width of the image in pixels.
  + ***height*** – Height of the image in pixels.
  + ***Samples\_Per\_Pixel*** – The number of samples per pixel.
  + ***Photometric\_Interpretation*** – The pixel composition. For example, "BlackIsZero".
* Sound File (MP3)
  + ***title***– The Title of the composition
  + ***composer*** – The composer of the musical work
  + ***audioCompressor*** – The audio compressor used. For example, “MP3”.
  + ***releaseDate*** – Release date of the work. Could be just a year such as “2008”.
  + ***album*** – If the work is from a larger album, this is the name of the album. For example, “Ninja Tuna”.
  + ***artist*** – The artist responsible for creating the recording. For example, "Mr. Scruff"
  + ***genre*** – The genre of the music for the audio file. For example, “Electronic”.
* Sound File (WAV)
  + ***audioSampleType*** – The type of samples. Typically “8Int” or “16Int”
  + ***samplerate*** – The sample rate (samples per second). For example, “44100.0”
  + ***bits*** – Number of bits per sample. For example, “16”.
  + ***encoding*** – The audio encoding algorithm. For example, “PCM\_SIGNED” (signed integer pulse-coded-modulation).
  + ***channels*** – Number of audio channels. For example, “2” for Stereo.

## Languages

The following languages are supported by OPA Production. The number included is the NARA code for the language. To find the 3-letter ISO 639-3 code for the language, please refer to the [official code lists](http://www-01.sil.org/iso639-3/codes.asp).

|  |  |  |  |
| --- | --- | --- | --- |
| 1 Abkhaz | 2 Achinese | 3 Acoli | 4 Adangme |
| 5 Afar | 6 Afrihili (Artificial language) | 7 Afrikaans | 8 Afroasiatic (Other) |
| 9 Akan | 10 Akkadian | 11 Albanian | 12 Aleut |
| 13 Algonquian (Other) | 14 Aljamma | 15 Altaic (Other) | 16 Amharic |
| 17 Apache languages | 18 Arabic | 19 Aramaic | 20 Arapaho |
| 21 Arawak | 22 Armenian | 23 Artificial (Other) | 24 Assamese |
| 25 Athapascan (Other) | 26 Australian languages | 27 Austronesian (Other) | 28 Avaric |
| 29 Avestan | 30 Awadhi | 31 Aymara | 32 Azerbaijani |
| 33 Balinese | 34 Baltic (Other) | 35 Baluchi | 36 Bambara |
| 37 Bamileke languages | 38 Banda | 39 Bantu (Other) | 40 Basa |
| 41 Bashkir | 42 Basque | 43 Batak | 44 Beja |
| 45 Belarusian | 46 Bemba | 47 Bengali | 48 Berber (Other) |
| 49 Bhojpuri | 50 Bihari | 51 Bikol | 52 Bini |
| 53 Bislama | 54 Bosnian | 55 Braj | 56 Breton |
| 57 Bugis | 58 Bulgarian | 59 Buriat | 60 Burmese |
| 61 Caddo | 62 Carib | 63 Catalan | 64 Caucasian (Other) |
| 65 Cebuano | 66 Celtic (Other) | 67 Central American Indian (Other) | 68 Chagatai |
| 69 Chamic languages | 70 Chamorro | 71 Chechen | 72 Cherokee |
| 73 Cheyenne | 74 Chibcha | 75 Chinese | 76 Chinook jargon |
| 77 Chipewyan | 78 Choctaw | 79 Church Slavic | 80 Chuvash |
| 81 Coptic | 82 Cornish | 83 Corsican | 84 Cree |
| 85 Creek | 86 Creoles and Pidgins (Other) | 87 Creoles and Pidgins, English-based (Other) | 88 Creoles and Pidgins, French-based (Other) |
| 89 Creoles and Pidgins, Portuguese-based (Other) | 90 Croatian | 91 Cushitic (Other) | 92 Czech |
| 93 Dakota | 94 Danish | 95 Dayak | 96 Delaware |
|  | 97 Dinka | 98 Divehi | 99 Dogri |
| 100 Dogrib | 101 Dravidian (Other) | 102 Duala | 103 Dutch |
| 104 Dutch, Middle (ca. 1050-1350) | 105 Dyula | 106 Dzongkha | 107 Efik |
| 108 Egyptian | 109 Ekajuk | 110 Elamite | 111 English, Middle (1100-1500) |
| 112 English, Old (ca. 450-1100) | 113 Eskimo languages | 114 Esperanto | 115 Estonian |
| 116 Ethiopic | 117 Ewe | 118 Ewondo | 119 Fang |
| 120 Fanti | 121 Faroese | 122 Fijian | 123 Filipino |
| 124 Finnish | 125 Finno-Ugrian (Other) | 126 Fon | 127 French |
| 128 French, Middle (ca. 1400-1600) | 129 French, Old (ca. 842-1400) | 130 Frisian | 131 Friulian |
| 132 Fula | 133 Galician | 134 Ganda | 135 Gayo |
| 136 Gbaya | 137 Gc | 138 Georgian | 139 [DEU] German |
| 140 German, Middle High (ca. 1050-1500) | 141 German, Old High (ca. 750-1050) | 142 Germanic (Other) | 143 Gilbertese |
| 144 Gondi | 145 Gorontalo | 146 Gothic | 147 Grebo |
| 148 Greek, Ancient (to 1453) | 149 Greek, Modern (1453- ) | 150 Guarani | 151 Gujarati |
| 152 Gwich'in | 153 Haida | 154 Hausa | 155 Hawaiian |
| 156 Hebrew | 157 Herero | 158 Hiligaynon | 159 Himachali |
| 160 Hindi | 161 Hiri Motu | 162 Hittite | 163 Hmong |
| 164 Hungarian | 165 Hupa | 166 Iban | 167 Icelandic |
| 168 Igbo | 169 Ijo | 170 Iloko | 171 Indian |
| 172 Indic (Other) | 173 Indo-European (Other) | 174 Indonesian | 175 Interlingua (International Auxiliary Language Association) |
| 176 Interlingue | 177 Inuktitut | 178 Inupiaq | 179 Iranian (Other) |
| 180 Irish | 181 Irish, Middle (ca. 1100-1550) | 182 Irish, Old (to 1100) | 183 Iroquoian (Other) |
| 184 [ITA] Italian | 185 Japanese | 186 Javanese | 187 Judeo-Arabic |
| 188 Judeo-Persian | 189 Kabyle | 190 Kachin | 191 Kalbtdlisut |
| 192 Kamba | 193 Kannada | 194 Kanuri | 195 Kara-Kalpak |
| 196 Karen | 197 Kashmiri | 198 Kawi | 199 Kazakh |
| 200 Khasi | 201 Khmer | 202 Khoisan (Other) | 203 Khotanese |
| 204 Kikuyu | 205 Kimbundu | 206 Kinyarwanda | 207 Kiswahili |
| 208 Komi | 209 Kongo | 210 Konkani | 211 Korean |
| 212 Kpelle | 213 Kru | 214 Kuanyama | 215 Kumyk |
| 216 Kurdish | 217 Kurukh | 218 Kusaie | 219 Kutenai |
| 220 Kyrgyz | 221 Ladino | 222 Lahnda | 223 Lakota |
| 224 Lamba | 225 Lao | 226 Latin | 227 Latvian |
| 228 Letzeburgesch | 229 Lezgian | 230 Lingala | 231 Lithuanian |
| 232 Lozi | 233 Luba-Katanga | 234 Luba-Lulua | 235 Luiseqo |
| 236 Lunda | 237 Luo (Kenya and Tanzania) | 238 Lushai | 239 Macedonian |
| 240 Madurese | 241 Magahi | 242 Maithili | 243 Makasar |
| 244 Malagasy | 245 Malay | 246 Malayalam | 247 Maltese |
| 248 Manchu | 249 Mandar | 250 Mandingo | 251 Manipuri |
| 252 Manobo languages | 253 Manx | 254 Maori | 255 Mapuche |
| 256 Marathi | 257 Mari | 258 Marshall | 259 Marwari |
| 260 Masai | 261 Mayan languages | 262 Mende | 263 Micmac |
| 264 Minangkabau | 265 Miscellaneous languages | 266 Mohawk | 267 Mohlam |
| 268 Moldavian | 269 Mon-Khmer (Other) | 270 Mongo-Nkundu | 271 Mongolian |
| 272 Moori | 273 Multiple languages | 274 Munda (Other) | 275 Nahuatl |
| 276 Nauru | 277 Navajo | 278 Ndebele (South Africa) | 279 Ndebele (Zimbabwe) |
| 280 Ndonga | 281 Nepali | 282 Newari | 283 Nias |
| 284 Niger-Kordofanian (Other) | 285 Nilo-Saharan (Other) | 286 Niuean | 287 North American Indian (Other) |
| 288 Northern Sami | 289 Northern Sotho | 290 Norwegian | 291 Nubian languages |
| 292 Nyamwezi | 293 Nyanja | 294 Nyankole | 295 Nyoro |
| 296 Nzima | 297 Occitan (post-1500) | 298 Ojibwa | 299 Old Norse |
| 300 Old Persian (ca. 600-400 B.C.) | 301 Oriya | 302 Oromo | 303 Osage |
| 304 Ossetic | 305 Otomian languages | 306 Pahlavi | 307 Palauan |
| 308 Pali | 309 Pampanga | 310 Pangasinan | 311 Panjabi |
| 312 Papiamento | 313 Papuan (Other) | 314 Persian | 315 Philippine (Other) |
| 316 Phoenician | 317 Polish | 318 Ponape | 319 Portuguese |
| 320 Prakrit languages | 321 Provengal (to 1500) | 322 Pushto | 323 Quechua |
| 324 Raeto-Romance | 325 Rajasthani | 326 Rapanui | 327 Rarotongan |
| 328 Romance (Other) | 329 Romanian | 330 Romany | 331 Rundi |
| 332 Russian | 333 Salishan languages | 334 Samaritan Aramaic | 335 Sami |
| 336 Samoan | 337 Sandawe | 338 Sango | 339 Sanskrit |
| 340 Santali | 341 Sardinian | 342 Sasak | 343 Scots |
| 344 Scottish Gaelic | 345 Selkup | 346 Semitic (Other) | 347 Serbian |
| 348 Serer | 349 Shan | 350 Shona | 351 Sidamo |
| 352 Sign languages | 353 Siksika | 354 Sindhi | 355 Sinhalese |
| 356 Sino-Tibetan (Other) | 357 Siouan (Other) | 358 Slave | 359 Slavic (Other) |
| 360 Slovak | 361 Slovenian | 362 Sogdian | 363 Somali |
| 364 Songhai | 365 Soninke | 366 Sorbian languages | 367 Sotho |
| 368 South American Indian (Other) | 369 [SPA] Spanish | 370 Sukuma | 371 Sumerian |
| 372 Sundanese | 373 Susu | 374 Swahili | 375 Swazi |
| 376 Swedish | 377 Syriac | 378 Tagalog | 379 Tahitian |
| 380 Tai (Other) | 381 Tajik | 382 Tamashek | 383 Tamil |
| 384 Tatar | 385 Telugu | 386 Temne | 387 Terena |
| 388 Tetum | 389 Thai | 390 Tibetan | 391 Tigri |
| 392 Tigrinya | 393 Tiv | 394 Tlingit | 395 Tok Pisin |
| 396 Tokelauan | 397 Tonga (Nyasa) | 398 Tongan | 399 Truk |
| 400 Tsimshian | 401 Tsonga | 402 Tswana | 403 Tumbuka |
| 405 Turkish Ottoman | 406 Turkmen | 407 Tuvaluan | 408 Tuvinian |
| 409 Twi | 410 Ugaritic | 411 Uighur | 412 Ukrainian |
| 413 Umbundu | 414 Undetermined | 415 Urdu | 416 Uzbek |
| 417 Vai | 418 Venda | 419 Vietnamese | 420 Volapjk |
| 421 Votic | 422 Wakashan languages | 423 Walamo | 424 Waray |
| 425 Washo | 426 Welsh | 427 Wolof | 428 Xhosa |
| 429 Yakut | 430 Yao | 431 Yapese | 432 Yiddish |
| 433 Yoruba | 434 Yupik languages | 435 Zande | 436 Zapotec |
| 437 Zenaga | 438 Zhuang | 439 Zulu | 440 Zuni |