HathiTrust Data API

Version 0.9, 10 September 2012 - Updated to reflect OAuth 1.0 signed URL requirements, Key Generation Service and Web Client.

- HathiTrust Data API
  - Introduction
  - Quick Overview
    - Description
    - Uses
    - Access
  - API Details
    - URI scheme
    - HTD Extension Elements, Attributes and Schema
    - Extension Elements
    - Schema
    - Resources and Representations
      - Volume and Rights Metadata (meta)
        - Example URI
      - Single Page Metadata (pagemeta)
        - Example URL
      - Structure (structure)
        - Example URI
      - Aggregate (aggregate)
        - Example URI
      - Single Page Image (pageimage)
        - Example URI
      - Single Page OCR (pageocr)
        - Example URI
      - Single Page Coordinate OCR (pagecoordocr)
        - Example URI
  - Access and Use Details
    - Web Client Access
    - Image of the Portal Page
    - Web Client User Interface
  - Programmatic Access
    - Client Program Development
      - Functional Elements
      - Making a Signed API Request
      - Data API Response Codes
      - Client Implementation Details
      - Signing
      - Sample Client and Server
  - Extended Uses
  - Key Generation Service (KGS)
    - Discussion
    - KGS User Interface
    - Registration
  - Appendices

Appendix A: Data API Sample Client
Appendix B: Items Determined to be in the Public Domain only in the U.S. or only outside the U.S.
Appendix C: Access Categories and Authorization
    Authorization Scheme
Appendix D: Example Volume and Rights Metadata (meta) Response
    RELAX NG Schema - Compact
Appendix E: Example Single Page Metadata (pagemeta) Responseodesnippet]
    Relax NG Schema - Compact
Appendix F: Example structure Response
    RELAX NG Schema - Compact

Introduction

This document describes a RESTful API to provide access to HathiTrust repository data and metadata resources. The HathiTrust Repository Data (HTD) API is referred to simply as API in this document.

Starting in April 2012, the Data API began accepting signed requests. **Beginning October 1st, 2012, all Data API requests must be signed.** See Security - Authentication - Authorization for details.

Quick Overview

Description

The HTD API provides extensible, efficient and secure access to the data and metadata resources of the HathiTrust Repository. The design intent is to support client applications that already have an item identifier and simply need the corresponding data (or metadata). It should make services and uses possible beyond those available through current applications. Examples of current applications are the HathiTrust Collection Builder and Pageturner.

The Data API accepts one ID per request. Applications that need a number of metadata records or data sets must request them one at a time. Another option is to have a dataset created. See http://www.hathitrust.org/datasets for details.

The repository resources consist primarily of digitized print or born-digital volumes composed of page images and OCR text and corresponding structural and administrative metadata. Other APIs and downloadable files provide sources of identifiers and bibliographic metadata. Examples include:

- Files of HathiTrust volume identifiers which can be downloaded.
- OAI at the University of Michigan
- HathiTrust Bibliographic API

The API accepts a request for a resource and returns XML, JSON or binary representations of the resource. The available representations depend on the resource in question.

The resources served by the API are partitioned into classes that have varying access policies.

- **Metadata (meta**, **pagemeta** and **structure)** - Accessible without restriction, subject only to throttling rates.
- **Data/Content (aggregate**, **pageimage**, **pageocr** and **pagecoordocr)** - Access varies:
    - Data unfettered by restrictions of any sort are available to download, subject only to throttling rates. Example: Public domain volumes digitized by Internet Archive.
    - Some forms of data are available for download, subject to throttling rates, and must also be at an IP address approved by HathiTrust for access. Example: Google-digitized public domain volumes in **aggregate** form.
    - In-copyright data is only available through the Data API in extremely rare cases, requiring a special contract.

Uses

Retrieval of volume metadata and content at standard rates (subject to throttling) can serve a variety of general purposes. It is also possible to use the API for extended purposes, that require an agreement with HathiTrust. Some of these extended uses are described in Extended Uses. Please contact us to determine the suitability of the Data API for intended uses. The Data API is meant for burst activities and not large-scale retrieval of content (e.g., for datasets).

Access

Access to the Data API is available either through a web client or programmatically.

API Details

URI scheme

Concrete examples are provided in each section describing a resource below. Square braces indicate an optional parameter. Throughout, variables are UPPERCASE prefixed with a colon, e.g. **`:VAR`**. XPath notation for elements and attributes appears occasionally.

```
http[s]://babel.hathitrust.org/cgi/htd/:RESOURCE/:ID[/:SEQ]
 ?
[v=:N]
[&alt=json[&callback=:CALLBACK]]
```

Access to restricted resources is over SSL using https:// protocol.
The values for the **`:RESOURCE`** variable for version 1 are:

- `meta`
- `structure`
- `aggregate`
- `pageimage`
- `pageocr`
- `pagecoordocr`
- `pagemeta`

The **`:ID`** variable ranges over the all namespace-qualified barcodes or other logical identifiers for repository objects. Examples of namespaces are `mdp, miun, wu`.
The **`:SEQ`** variable is an integer starting at 1 and ranges up to the number of page images in the object.
The version query parameter (`v`) supports backward compatibility as the API evolves. Clients can explicitly specify the version of the requests they are issuing and of the response that they

want. By default, when no version parameter is supplied, responses are in the format of the latest version of the API.
Where `application/xml` responses apply, `alt=json` requests the response in JSON format. In that case, an optional javascript callback function name can be supplied. The default response format is `application/xml` .

HTD Extension Elements, Attributes and Schema

The schema for the XML responses is based on the Atom Syndication Format in the spirit of the response schema for a volume from the Google Book Search Data API as shown in the Data API: Reference Guide.
XML responses are formatted as **`atom:entry`** elements in the default **`atom`** namespace. The required **`atom:id, atom:title, atom:updated`** elements are present. The HTD API schema extends the Atom schema by defining and using the **`htd`** namespace.
Note that the use of the **`atom:entry`** element is adopted in the context of access to data and not necessarily of access to a feed.
The HTD API is a *data* API with accompanying structural and administrative metadata. It is not a bibliographic metadata API. The **`atom:title`** element contains text that describes the entry and is *not* the title of the book. For example,

```
HathiTrust Repository Data API - single page metadata.
```

The schema employs a URI-based scheme for additional values of the **`atom:link[@rel]`** attribute. For resource identifiers we have:

- http://schemas.hathitrust.org/htd/2009#meta
- http://schemas.hathitrust.org/htd/2009#pagemeta
- http://schemas.hathitrust.org/htd/2009#structure
- http://schemas.hathitrust.org/htd/2009#aggregate
- http://schemas.hathitrust.org/htd/2009#pageimage
- http://schemas.hathitrust.org/htd/2009#pageocr
- http://schemas.hathitrust.org/htd/2009#pagecoordocr

The optional element **`atom:link`** appears with the **`rel=alternate`** and **`rel=self`** attributes.

- **`link[@rel='alternate']`** - Generally taken to mean the permalink to the content pointed to by the entry. Currently this includes a link to the HathiTrust pageturner which is quasi-permanent and a link to the Handle Server. For example,

  ```
  http://babel.hathitrust.org/cgi/pt?id=:ID[&seq=:SEQ]
  ```

  and

  ```
  http://hdl.handle.net/2027/:ID
  ```

- **`link[@rel='self']`** - This is the preferred URI for retrieving the entry itself. This value is important in scenarios where only the entry is available and not the location from which the entry was retrieved.

Extension Elements

Extension elements are in the **`htd`** namespace and vary with response. Refer to example abstract responses below.

- `htd:version` - the version number of the API generating the response

`htd:selected_seq` - the page sequence number requested. (`pagemeta` resource only.)

- `htd:numpages` - the number of pages in the volume
- `htd:access_use_statement` – the full text of the Access and Use statement stating the permitted uses and rights to access this item as determined by the `htd:rights/htd:attr` and `htd:rights/htd:source` values.
- `htd:access_use` - a URI equivalent to the htd:access_use_statement with one of the following values. Please refer to the Access and Use page for explanations of these values.
    - `htd:access[@resource]` - asserts whether downloading the page images, OCR and zipped data is available. Metadata access does not require authentication and authorization. Restricted or limited access does not imply restricted viewability in certain contexts or via other HathiTrust web applications (e.g. PageTurner).
      Attribute values:`http://schemas.hathitrust.org/htd/2009#pd`
    - `http://schemas.hathitrust.org/htd/2009#pd-google`
    - `http://schemas.hathitrust.org/htd/2009#pd-us`
    - `http://schemas.hathitrust.org/htd/2009#pd-us-google`
    - `http://schemas.hathitrust.org/htd/2009#oa`
    - `http://schemas.hathitrust.org/htd/2009#oa-google`
    - `http://schemas.hathitrust.org/htd/2009#section108`
    - `http://schemas.hathitrust.org/htd/2009#ic`
    - `http://schemas.hathitrust.org/htd/2009#cc-by`
    - `http://schemas.hathitrust.org/htd/2009#cc-by-nd`
    - `http://schemas.hathitrust.org/htd/2009#cc-by-nc-nd`
    - `http://schemas.hathitrust.org/htd/2009#cc-by-nc`
    - `http://schemas.hathitrust.org/htd/2009#cc-by-nc-sa`
    - `http://schemas.hathitrust.org/htd/2009#cc-by-sa`
    - `http://schemas.hathitrust.org/htd/2009#cc-zero`
    - `http://schemas.hathitrust.org/htd/2009#und-world`
    - `http://schemas.hathitrust.org/htd/2009#open`
    - `http://schemas.hathitrust.org/htd/2009#limited`
    - `http://schemas.hathitrust.org/htd/2009#restricted`
- `htd:rights` - container element for rights metadata:
    - `htd:namespace` - the namespace of the **:ID** (dotted concatenation of `htd:namespace` and `htd:id`
    - `htd:id` - the volume barcode
    - `htd:attr` - See HathiTrust rights database document
    - `htd:reason` - See HathiTrust rights database document
    - `htd:source` - See HathiTrust rights database document
    - `htd:user` - See HathiTrust rights database document
    - `htd:time` - See HathiTrust rights database document
    - `htd:note` - See HathiTrust rights database document
- `htd:pgmap` - container element for page number to page sequence number map
    - `htd:pg[@pgnum]` - the mapping element. attribute is page number, content is page sequence number. one for each page number.
- `htd:seqmap` - container element for map of page sequence number to page number, feature, format.

- `htd:seq[@pseq]` - attribute is the sequence number of the page, content is the page number
- `htd:pnum` - the page number either printed or implicit (if available)
- `htd:imgfmt` - format of the page image: `tiff` or `jp2` or `jpg`
- `htd:pfeat` - the page feature key (if available):
  - `CHAPTER_START`
  - `COPYRIGHT`
  - `FIRST_CONTENT_CHAPTER_START`
  - `FRONT_COVER`
  - `INDEX`
  - `REFERENCES`
  - `TABLE_OF_CONTENTS`
  - `TITLE`

Schema

Refer to the appendix link in each Resource section for schemas of example responses.

Resources and Representations

The API provides access to the following resources. The MIME types of the available representations are shown in the table below. An example URI is provided. An example abstract response is shown for resources with `application/xml` representations.

Resource	Representation(s)/MIME type(s)
Volume and Rights Metadata (meta)	`application/atom+xml & application/json`
METS (structure)	`application/xml & application/json`
zip file (aggregate)	`application/zip`
Single Page Metadata (pagemeta)	`application/atom+xml & application/json`
Single Page Image (pageimage)	`image/jp2 | image/tiff | image/jpg`
Single Page OCR (pageocr)	`text/plain`
Single Page Coordinate OCR (pagecoordocr)	`text/html` or `application/xml`

Volume and Rights Metadata (`meta`)

This resource consists of:

- API version number
- access values
- count of page image / OCR text pairs

- a row of the rights database consisting of the data from following fields: `id, namespace, attr, reason, source, user, time, note` as described in the database layout document
- a map of page sequence number to:
    - page number, either explicitly on the printed page or algorithmically derived during digitization
    - page feature tags as defined by the label attribute of the `METS:structMap/ METS:div` element of the Structure (METS document) resource. See Extension Elements and METS schema.
    - page image file format, one of `tiff` or `jp2` or `jpg`
- a map of page number to page sequence number

Note: Page feature and page number metadata is not available for some instances of this resource.
Compare with Single Page Metadata

Example URI

Resource request for the volume and rights metadata for a public domain, Google-digitized book in response format of `application/atom+xml`. The URI must be signed as described in the security section.

`http://babel.hathitrust.org/cgi/htd/meta/mdp.39015070515765`

Please refer to this Appendix D for an example of the response to this URI or invoke the web client at `http://babel.hathitrust.org/cgi/htdc`.

Single Page Metadata (`pagemeta`)

This resource consists of a partial reiteration of the volume and rights metadata for the book together with the page feature metadata available for the given sequential page.

Example URL

Resource request for the metadata for the 11th sequential page of an in-copyright, Google-digitized book in response format of `application/json`. Compare with Volume and Rights Metadata. The URI must be signed as described in the security section.

`http://babel.hathitrust.org/cgi/htd/pagemeta/mdp.39015005102796/11?alt=json`

Please refer to this Appendix E for an example of the response to this URI or invoke the web client at `http://babel.hathitrust.org/cgi/htdc`.

Structure (`structure`)

This resource is a METS document representing the volume. The `application/xml` representation for the METS portion of this resource is described by the METS schema. This resource gives the client application the most detailed picture of the aggregate repository object.

Example URI

Resource request for the METS document for a public domain (US), Google-digitized book in response format of `application/xml` where the version of the API is explicitly requested. The URI must be signed as described in the security section.

http://babel.hathitrust.org/cgi/htd/structure/mdp.39015064570875?v=1

Please refer to this Appendix F for an example of the response to this URIor invoke the web client at http://babel.hathitrust.org/cgi/htdc.

Aggregate (aggregate)

This resource is a zip file sent as `application/zip`. Currently this resource has only one structure:

- for each page in the resource:
  - the page image
  - corresponding UTF-8 encoded OCR plain text
  - the Source METS

N.B. The Source METS included in the zip file is not the active HathiTrust METS in use in the repository. The Source METS contains information about the object prior to ingest (the differences between the Source and HathiTrust files are discussed at HathiTrust Digital Object Specifications). HathiTrust References to the contents of the zip file are maintained in the **HathiTrust METS.** Clients of the Data API should rely on and retrieve the HathiTrust METS rather than the Source METS. The HathiTrust METS is available as the **structure** resource. Please refer to the structure section of this document for more information about the structure resource.

Example URI

Resource request for the zip file for a in-copyright, Google-digitized book in response format of `application/zip`. The URI must be signed as described in the security section. In addition to a signature, a contract with HathiTrust is required to access in-copyright materials.

http://babel.hathitrust.org/cgi/htd/aggregate/miun.abr0732.0001.001

Invoke the web client at http://babel.hathitrust.org/cgi/htdc to obtain an example response for this resource.

Single Page Image (pageimage)

Example URI

Resource request for the 12th sequential page image from an public domain, DLPS-digitized public-domain book. The URI must be signed as described in the security section.
Depending on how the page was scanned the response format is one of the following.

- `image/tiff`
- `image/jp2`
- `image/jpg`

http://babel.hathitrust.org/cgi/htd/pageimage/miun.abr0732.0001.001/12

Invoke the web client at http://babel.hathitrust.org/cgi/htdc to obtain an example response for this resource.

Single Page OCR (pageocr)

This resource is the UTF-8 encoded OCR plain text of a given page image.

Example URI

Resource request for the OCR text of the 30th sequential page image from a public-domain, Google-digitized book. The response format is `text/plain`. The URI must be signed as described in the security section.

`http://babel.hathitrust.org/cgi/htd/pageocr/mdp.39015000000128/12`

Invoke the web client at `http://babel.hathitrust.org/cgi/htdc` to obtain an example response for this resource.

Single Page Coordinate OCR (`pagecoordocr`)

This resource is the UTF-8 encoded XML for OCR with coordinate information for a given page image.

Example URI

Resource request for the OCR text of the 24th sequential page image from an public domain, Internet Archive digitized book. The URI must be signed as described in the security section. The response format is `text/html` or `application/xml`

`http://babel.hathitrust.org/cgi/htd/pagecoordocr/uc2.ark:/13960/t0dv1g69b/24`

Invoke the web client at `http://babel.hathitrust.org/cgi/htdc` to obtain an example response for this resource.

Access and Use Details

Web Client Access

Data API users who previously invoked the API directly from their browsers will not have that option after the latest security enhancements. However, a web client is available for these users. To use the web client, the user first logs in to the portal at `http://babel.hathitrust.org/cgi/kgs/portal`. "Friend" accounts are available for users not affiliated with a HathiTrust institution. Instructions for setting up a friend account are available on the login page (to access the login page go to http://babel.hathitrust.org/cgi/mb?a=listcs;colltype=pub#all and click login in the top right corner). Upon login a user is automatically authorized to use the web client to request resources at the default level.

Invoke the web client at `http://babel.hathitrust.org/cgi/htdc`

Image of the Portal Page

Following is a screenshot of the Data API portal web page.

HATHI TRUST Digital Library

HathiTrust Data API Access Portal

The HathiTrust Data API provides access to data and metadata resources. Continue on below to get instructions and access as one of these two types of users.

As a web browser user ...

Login to register yourself to use the Data API web client in your browser.

As a program developer ...

Write a **program** to invoke the Data API using 2-legged OAuth 1.0 signed URLs. Visit the HathiTrust Data API Key Service to register and receive OAuth keys to plug into your program to create signed URLs.

Web Client User Interface

Following is a screenshot of the Data API client web page.

HathiTrust Data API request

- An example ID is `mdp.39015011716761`
- Use the ***back*** button to return to this page after your data arrives.

Request parameters

required field *

id *

resource *

- METS file
- entire object
- object metadata
- page metadata
- page image
- page OCR
- page coordinate OCR

Submit

Programmatic Access

Client Program Development

This section describes a 2-legged OAuth mechanism to support secure access to the HathiTrust Data API (API). The mechanism provides API clients with credentials the API can use to authenticate the client's identity and authorize it to access repository resources. The Key Generation Service (KGS) provides users with a registration point and access to keys.

Functional Elements

The Data API security implementation consists of these elements:

- A specification that describes exactly how a client should generate a signed API request URI.

- Generation and transmission of a public **`oauth_consumer_key`** (client ID) and a **`oauth_consumer_secret`** shared between the API and its client. The Key Generation Service (KGS) supports the transmission.
- An authentication database that stores the `oauth_consumer_key` and `oauth_consumer_secret` and assorted client data.
- An authorization database that associates `oauth_consumer_key` with authorization to API resources.
- Logging, monitoring and reporting

Making a Signed API Request

The API client is required to sign the API request URI for all metadata and data resources. Possessing a registered access and oauth_consumer_secret do not automatically give greater privileges. Greater privileges require a special contract to be negotiated with HathiTrust.

A signed API request involves the construction and signing of the request URI and its authentication and authorization by the API.

The client program constructs a request URI from the basic request URI, adding the plain-text `oauth_consumer_key`, `oauth_signature`, `oauth_nonce`, `oauth_timestamp`, `oauth_version` and `oauth_signature_method` to the query parameters.

An example of a signed Data API resource request URI that obeys the OAuth 1.0 specification is:

```
http://babel.hathitrust.org/cgi/htd/pagemeta/mdp.39015000000128/12?
oauth_consumer_key=23f9457e2&oauth_nonce=192ed4d53e27e5d2dcd1&oauth_
signature=HIiQ13Vm0WuZeXKl6qxzgLqxmtI%3D&oauth_signature_method=HMAC-
SHA1&oauth_timestamp=1338838461&oauth_version=1.0
```

The good specification for constructing an OAuth 1.0 signed URL can be found at `http://oauth.googlecode.com/svn/spec/ext/consumer_request/1.0/drafts/1/spec.html`

The dialogue between client program and Data API proceeds as follows.

1. Client signs the constructed URI with the `oauth_consumer_secret` and adds the `oauth_signature` to the query parameters.
2. API receives the signed request.
3. API looks up the client's `oauth_consumer_secret` in the authorization database and uses it to sign the plain-text URI as the `oauth_signature`.
4. API compares its `oauth_signature` with the `oauth_signature` carried by the client's API request URI.
    a. If signatures match, API looks up the privileges associated with the oauth_consumer_key in the authorization database.
        i. If there is sufficient privilege for the requested resource it is delivered in the API response.

ii. If insufficient privilege exists, API responds with authorization failure status.

b. If signatures do not match, API responds with authentication failure status.

Data API Response Codes

Code	Explanation
`200 OK`	No error. The request to retrieve the resource was successful.
`400 BAD REQUEST`	Invalid request URI or HTTP header, or unsupported parameter.
`303 SEE OTHER`	This redirect will be issued when a resource is restricted and the request protocol is not HTTPS. The redirect URL to repeat the request is returned in the HTTP header. It will not contain the required OAuth parameters. These must be regenerated on the client side based on the redirect URL.
`401 UNAUTHORIZED`	This code will be returned when the OAuth signature, timestamp or nonce are invalid or when one or more required OAuth parameters are missing.
`403 FORBIDDEN`	This code will be returned when access key (`oauth_consumer_key`) is not associated with an authorization level sufficient to grant access to the requested resource.
`404 NOT FOUND`	Resource identified by `:ID` or `:ID/:SEQ` not found.
`500 INTERNAL SERVER ERROR`	Internal error. This is the default code that is used for all unrecognized errors.
`503 SERVICE UNAVAILABLE`	Quota exceeded.

Client Implementation Details

All resource request URLs must be signed. This is mandatory as of 1 October, 2012 and is optional but supported currently.

Following registration with KGS, the developer's access key carries default authorization allowing access to resources categorized by **`open`** and **`limited`** access types. All metadata resources are categorized as **`open`**.

The value of the **`htd:access[@resource="aggregate|pageocr|pagecoordocr|pageimage"]`** element in the **`meta`** and **`pagemeta`** resources for a given resource, indicates the restrictions on that resource. Refer to Appendix C for details regarding these access values. Values vary from item to item, and across resource types, most often depending on the digitization source and limitations imposed on the distribution of data by the source. A baseline level of throttling is employed by the Data API in all cases to ensure the system performs consistently. Throttling is used more extensively, for certain forms of data, to ensure the limitations mentioned above are met. Example: Google-digitized volumes in the **aggregate** form are always "restricted," requiring special authorization. However, in individual **pageimage** form, they are "limited," requiring more restrictive throttling rates, but not special authorization.

`htd:access` URN value	**Explanation**
`http://schemas.hathitrust.org/htd/2009#restricted`	Higher level of authorization is required, established through an out-of-band contract negotiation.
`http://schemas.hathitrust.org/htd/2009#limited`	Open, subject to more restrictive throttling rates.
`http://schemas.hathitrust.org/htd/2009#open`	Open, subject to default throttling rates.

URLs requesting resources categorized as `restricted` must use **`https`** protocol. API clients that request restricted resources over **`http`** must be prepared to handle a `303` response code and regenerate a signed URL from the URL returned in the `Location` field of the HTTP header.

Signing

The signing code is based on Perl `OAuth::Lite::Consumer` available from CPAN at `http://search.cpan.org/dist/OAuth-Lite/`. URI signing uses the Hash-based Message Authentication Code HMAC-SHA1 algorithm. One source of code libraries in other languages that implement OAuth using HMAC-SHA1 to sign URLs can be found at `http://code.google.com/p/oauth/`. The key generation code is based on `OAuth::Lite::Util` also available from CPAN as above.

Sample Client and Server

See Appendix A for code that implements a sample Perl CGI client that will invoke our test server. It will run out of the box after you install `OAuth::Lite::Consumer` and `OAuth::Lite::AuthMethod`.

The sample client can be invoked on our servers as:
http://babel.hathitrust.org/cgi/htdc/dapiclient

A sample server that the `dapiclient` talks to by default can also be invoked by your own client at this address:
http://babel.hathitrust.org/cgi/htdc/dapiserver

Extended Uses

The Data API currently offers a wide range of retrieval options for HathiTrust content. However, partners have expressed the desire, and HathiTrust wishes to support, additional options that facilitate extended activities. Some examples of these activities include the following:

Preservation uses of in-copyright content:

1. A partner retrieves single pages of an in-copyright volume to insert into a physical volume
2. A partner retrieves a whole volume in order to make a print replacement copy

Validation of public domain and in-copyright content

1. A partner performs external validation of in-copyright and public domain archival packages (the full package for some, e.g., Google-digitized, content is not currently available through the API)

Development and some data retrieval purposes for publicly available content
Use of Data API without rate limits (throttling) facilitates activities for partners and non-partners alike such as

1. Building and testing new interfaces to content
2. Identifying materials from a particular scanning source
3. Enabling "crawling" or other means of indexing the full-text of all or a large subset of public domain materials

Key Generation Service (KGS)

The Key Generation Service provides developers of API clients with a single point of access to get an oauth_consumer_key and oauth_consumer_secret (key set). Following is a description of the secure mechanism used to communicate the key set to the developer.

The KGS can be invoked at http://babel.hathitrust.org/cgi/kgs/request

Discussion

We must ensure that the recipient of the key set is the same as the person who made the request. To this end, we require a valid email address. The email address serves several purposes:

- To email a KGS-signed, one-time URL link that, when followed by the developer and received and authenticated by KGS, presents a web page that displays the `oauth_consumer_key` and `oauth_consumer_secret`.

- To inform the developer that we may disable their `oauth_consumer_key` if abuse is detected.
- To optionally report daily usage to the developer by email, in order to:
    - alert the developer of illegitimate activity in the event their key set is compromised
    - delete the key set if the email address is no longer valid

The KGS is accessed over https:// to secure the request's form data and the KGS reply that contains the one-time URL link that displays the key set.

KGS User Interface

Following is a screenshot of the HathiTrust Key Service web page.

HATHI TRUST Digital Library

HathiTrust Data API key request

The HathiTrust Data API provides access to data and metadata resources. URLs must include an access key and be signed with a secret key. An access key and signature are required to request any resource.

Consult the HathiTrust Data API document to see an example program to send a basic signed request to our sample server. Copy and edit it to use the keys you request here as a starting point to access the Data API using signed URLs.

Access to restricted data resources may be requested. Some use cases we envision can also be found in the HathiTrust Data API document. Those interested should contact feedback@issues.hathitrust.org with a brief but detailed description of the intended use.

Please submit the following information to request keys

required field *

name *

organization *

e-mail *

Submit

Instructions

- When we receive your information, we'll send email containing a link to the registration page. You may need to allow **feedback@issues.hathitrust.org** to pass your spam filter.
- Follow the registration link in the email to view your keys. Copy them to a secure location and treat them the same as passwords.
- The registration link is **time-sensitive** and **one-time-only**. It expires after the first visit and absolutely after 24 hours.
- A given email address is limited to 10 unconfirmed requests and up to 5 confirmed requests.
- Please contact us at feedback@issues.hathitrust.org if you need assistance.

Registration

The KGS presents a web interface where a developer of an API client enters a user name, institution name, and an email address. The following sequence of operations constitutes the registration process.

1. User enters user name, institution name and email address in a web form and submits them over SSL to KGS.
2. KGS generates an `oauth_consumer_key` and a `oauth_consumer_secret` and stores them and the form-data in the authentication database.
3. KGS generates and emails a confirmation URL from the form data and oauth_consumer_key. The URL endpoint is the KGS. The URL consists of KGS host and path info and an `oauth_consumer_key` parameter in plain-text and a `oauth_signature` that is generated using the `oauth_consumer_secret` to encrypt the host, path info, `oauth_consumer_key` and form-data:
    - **Example URLin email:** `https://babel.hathitrust.org/cgi/kgs/confirm?email=smith%40some.edu&name=Smith&oauth_consumer_key=fea256f552&oauth_nonce=adde9747a65ccaf073b0&oauth_signature=c%2F6KCYVM%2FysRy%2B5c2BR9QoF9syY%3D&oauth_signature_method=HMAC-SHA1&oauth_timestamp=1331924673&oauth_version=1.0&org=Some%20University`
4. User receives email and clicks on URL
5. URL invokes the KGS over SSL. KGS retrieves `oauth_consumer_secret` and email address web form data by `oauth_consumer_key`. KGS performs identical encryption performed by client to sign the request and tests that signature matches and displays `oauth_consumer_key` and the shared `oauth_consumer_secret` on the KGS confirmation web page.

The key-set request is timestamped in the authentication database and deleted if the developer does not activate the key-set within 24 hours. The page displaying the keys cannot be reloaded or re-visited if the browser is closed.

Appendices

Appendix A: Data API Sample Client

This is a sample client written in Perl that uses the `OAuth::Lite` `OAuth::Lite` packages from CPAN. It can be run on our servers as `http://babel.hathitrust.org/cgi/htdc/dapiclient`
or on your own server in a Perl environment with `OAuth::Lite` installed.

```
#!/usr/bin/env perl

=head1 NAME

dapiclient

=head1 DESCRIPTION

This is an example perl 2-legged oauth client that shares the secret
key "PUBLIC_OAUTH_CONSUMER_SECRET" with dapiserver.  It is intended to
aid development of a fully function Data API client in Perl or other
languages that implement HMAC_SHA1 OAuth libraries.

=head1 SYNOPSIS
```

```
For example:

http://yourhost/path_to_client/dapiclient

=head1 OUTPUT

[CLIENT] sent this URL to server:

http://babel.hathitrust.org/cgi/htd/dapiserver?
hello=world&oauth_consumer_key=PUBLIC_OAUTH_CONSUMER_KEY&oauth_nonce=47b8186be439110b4
f98&oauth_signature=2cQYAM%2BYek%2BiOexZKMObM%2B3B2w4%3D&oauth_signature_method=HMAC-
SHA1&oauth_timestamp=1332184191&oauth_version=1.0

[CLIENT] received this HTTP response from server:
  200 OK

[CLIENT] received this content response from server:

        [SERVER] received client request. echoing request parameters:

        hello
        world
        oauth_consumer_key
        PUBLIC_OAUTH_CONSUMER_KEY
        oauth_nonce
        47b8186be439110b4f98
        oauth_signature
        2cQYAM+Yek+iOexZKMObM+3B2w4=
        oauth_signature_method
        HMAC-SHA1
        oauth_timestamp
        1332184191
        oauth_version
        1.0

=cut

use strict;
use warnings;

use CGI;
use OAuth::Lite::Consumer;
use OAuth::Lite::AuthMethod;


my $access_key = 'PUBLIC_OAUTH_CONSUMER_KEY';
my $secret_key = 'PUBLIC_OAUTH_CONSUMER_SECRET';

my $request_url = 'http://babel.hathitrust.org/cgi/htd/dapiserver';

my $consumer = OAuth::Lite::Consumer->new
 (
  consumer_key => $access_key,
  consumer_secret  => $secret_key,
  auth_method  => OAuth::Lite::AuthMethod::URL_QUERY,
 );

my $response = $consumer->request
 (
  method  => 'GET',
  url   => $request_url,
  params  => {
               'hello' => 'world',
               },
 );

print CGI::header();

print "<p><b>[CLIENT] sent this URL to server:</b><br/>";
print $consumer->oauth_request->uri;
```

```
print "<p><b>[CLIENT] received this HTTP response from server:</b><br/>";
print $response->status_line;
if ($response->is_success) {
       print "<br/><b>[CLIENT] received this content response from server:</b><blockquote>" .
$response->content . "</blockquote>";
}

exit 0;
```

Appendix B: Items Determined to be in the Public Domain only in the U.S. or only outside the U.S.

Some data resources may be categorized as "`open`" because the item is in the public domain only in the U.S. or only outside the U.S. The Data API determines access rights in such cases based on the IP address of the requesting client. All clients should check the HTTP header for the `X-HathiTrust-Notice` to be informed, and inform third-party users, of obligations with regard to these items under their local copyright law.

The text of the notice in the HTTP header in each of these cases is available at the following links.

- Access and use policy statement for **"Public Domain only in the U.S."**
- Access and use policy statement for **"Public Domain only outside the U.S."**

Appendix C: Access Categories and Authorization

The Data API provides access to repository data resources and metadata resources derived from the METS document describing the object. Access is categorized along several dimensions. Values along the *first* dimension are a function of **rights attribute** (copyright, license):

- **free**
  - **Data/Content (aggregate**, **pageimage**, **pageocr** and **pagecoordocr)** - with rights in public-domain, world, public-domain-us, Creative Commons license, etc.
  - **Metadata (meta**, **pagemeta** and **structure)**
- **non-free**
  - **Data/Content (aggregate**, **pageimage**, **pageocr** and **pagecoordocr)** in-copyright or otherwise not allowed.

Within the **free** value, some data may, nonetheless, have various restrictions. Example: Google-digitized volumes in the **aggregate** form are always "restricted," requiring special authorization. However, in individual **pageimage** form, they are "limited," requiring more restrictive throttling rates, but not special authorization. Or, there may be restrictions on the rate at which the data may be provided (to prevent bulk downloading). These restrictions are modeled along a *second* dimension of values (refer also to Appendix B):

- **open** - available
  - subject to default HathiTrust rate limits
- **limited** - available
  - subject to more restrictive throttling rate limits
- **restricted** - restricted
  - requires a special contract with HathiTrust in order to obtain access

Authorization Scheme

For access, two authorization levels are defined: **registered** and **trusted**. Both authorization levels require registration and signed URIs for all resource requests (metadata and data). The higher, trusted level, in addition, requires a contract with HathiTrust. Requests are authenticated, logged and monitored for abuse.

- **registered** - authorized to receive **open** and **limited** data and all metadata by registration with KGS and use of signed URLs, including through the web client
    - Negotiated higher data rates are available if required, under contract; otherwise, default rates apply.
- **trusted** - authorized to receive **open**, **limited** and **restricted** data and all metadata by registration with KGS and use of signed URLs, including through the web client
    - Contract required with HathiTrust for restricted data.
    - Negotiated higher data rates are available if required, under contract; otherwise, default rates apply.

Appendix D: Example Volume and Rights Metadata (`meta`) Response

(Edited for brevity)

```
<entry
 xmlns="http://www.w3.org/2005/Atom"
 xmlns:htd="http://schemas.hathitrust.org/htd/2009">
 <link
   rel="alternate"
   href="http://hdl.handle.net/2027/mdp.39015070515765"
   type="text/html"/>
 <link
   rel="self"
   href="http://babel.hathitrust.org/cgi/htd/meta/mdp.39015070515765"
   type="application/atom+xml"/>
 <link
   rel="http://schemas.hathitrust.org/htd/2009#aggregate"
href="https://babel.hathitrust.org/cgi/htd/aggregate/
mdp.39015070515765"
   type="application/zip"/>
 <link
   rel="http://schemas.hathitrust.org/htd/2009#structure"
href="http://babel.hathitrust.org/cgi/htd/structure/
mdp.39015070515765"
   type="application/xml"/>
 <htd:version>1</htd:version>
 <htd:numpages>668</htd:numpages>
 <htd:seqmap>
   <htd:seq pseq="1">
     <htd:pnum></htd:pnum>
     <htd:pfeat>FRONT_COVER</htd:pfeat>
     <htd:pfeat>IMAGE_ON_PAGE</htd:pfeat>
     <htd:pfeat>UNTYPICAL_PAGE</htd:pfeat>
     <htd:pfeat>IMPLICIT_PAGE_NUMBER</htd:pfeat>
     <htd:imgfmt>image/jp2</htd:imgfmt>
   </htd:seq>
   <htd:seq pseq="2">
     <htd:pnum>i</htd:pnum>
```

```
      <htd:pfeat>UNTYPICAL_PAGE</htd:pfeat>
      <htd:pfeat>IMPLICIT_PAGE_NUMBER</htd:pfeat>
      <htd:imgfmt>image/jp2</htd:imgfmt>
    </htd:seq>
    <htd:seq pseq="3">
      <htd:pnum>ii</htd:pnum>
      <htd:pfeat>IMPLICIT_PAGE_NUMBER</htd:pfeat>
      <htd:imgfmt>image/jp2</htd:imgfmt>
    </htd:seq>
  </htd:seqmap>
  <htd:pgmap>
    <htd:pg pgnum="i">2</htd:pg>
    <htd:pg pgnum="ii">3</htd:pg>
  </htd:pgmap>
  <id>http://babel.hathitrust.org/cgi/htd/meta/mdp.39015070515765</id>
  <title>HathiTrust Repository Data API - metadata</title>
  <updated>2009-03-11T17:29:58.602-0400</updated>
  <htd:access resource="pageocr">
    http://schemas.hathitrust.org/htd/2009#limited</htd:access>
  <htd:access resource="pageimage">
    http://schemas.hathitrust.org/htd/2009#limited</htd:access>
  <htd:access resource="aggregate">
    http://schemas.hathitrust.org/htd/2009#restricted</htd:access>
  <htd:rights>
    <htd:note/>
    <htd:user>jhovater</htd:user>
    <htd:time>2008-07-09T00:30:11</htd:time>
    <htd:namespace>mdp</htd:namespace>
    <htd:source>1</htd:source>
    <htd:attr>1</htd:attr>
    <htd:id>39015070515765</htd:id>
    <htd:reason>1</htd:reason>
  </htd:rights>
</entry>
```

RELAX NG Schema - Compact

```
default namespace = "http://www.w3.org/2005/Atom"
namespace htd = "http://schemas.hathitrust.org/htd/2009"
start =
  element entry {
    element link {
      attribute href { xsd:anyURI },
      attribute rel { xsd:anyURI },
      attribute type { text }
    }+,
    element htd:version { xsd:integer },
    element htd:numpages { xsd:integer },
    element htd:seqmap {
      element htd:seq {
        attribute pseq { xsd:integer },
```

```
      element htd:pnum { text },
      element htd:pfeat { xsd:NCName }+,
      element htd:imgfmt { text }
    }+
  },
  element htd:pgmap {
    element htd:pg {
      attribute pgnum { xsd:NCName },
      xsd:integer
    }+
  },
  element id { xsd:anyURI },
  element title { text },
  element updated { xsd:NMTOKEN },
  element htd:access {
    attribute resource { xsd:NCName },
    xsd:anyURI
  }+,
  element htd:rights {
    element htd:note { empty },
    element htd:user { xsd:NCName },
    element htd:time { xsd:NMTOKEN },
    element htd:namespace { xsd:NCName },
    element htd:source { xsd:integer },
    element htd:attr { xsd:integer },
    element htd:id { xsd:integer },
    element htd:reason { xsd:integer }
  }
}
```

Appendix E: Example Single Page Metadata (`pagemeta`) Response

```
{
  "xmlns": "http://www.w3.org/2005/Atom",
  "xmlns:htd": "http://schemas.hathitrust.org/htd/2009",
 "id": "http://babel.hathitrust.org/cgi/htd/pagemeta/
mdp.39015005102796/11",
  "title":"HathiTrust Repository Data API - single page metadata",
  "updated": "2009-03-12T09:48:43.885-0400",
  "link": [
      {
          "rel": "alternate",
          "href": "http://hdl.handle.net/2027/mdp.39015005102796",
          "type": "text/html"
      },
      {
          "rel": "self",
          "href": "http://babel.hathitrust.org/cgi/htd/pagemeta/
mdp.39015005102796/11",
          "type": "application/atom+xml"
      },
      {
```

```
          "rel": "http://schemas.hathitrust.org/htd/2009#pageimage",
          "href": "https://babel.hathitrust.org/cgi/htd/pageimage/
mdp.39015005102796/11",
          "type": "image/tiff"
      },
      {
          "rel": "http://schemas.hathitrust.org/htd/2009#pageocr",
          "href": "https://babel.hathitrust.org/cgi/htd/pageocr/
mdp.39015005102796/11",
          "type": "text/plain"
      },
      {
          "rel": "http://schemas.hathitrust.org/htd/2009#aggregate",
          "href": "https://babel.hathitrust.org/cgi/htd/aggregate/
mdp.39015005102796",
          "type": "application/zip"
      },
      {
          "rel": "http://schemas.hathitrust.org/htd/2009#structure",
          "href": "http://babel.hathitrust.org/cgi/htd/structure/
mdp.39015005102796",
          "type": "application/xml"
      },
      {
          "rel": "http://schemas.hathitrust.org/htd/2009#meta",
          "href": "http://babel.hathitrust.org/cgi/htd/meta/
mdp.39015005102796",
          "type": "application/atom+xml"
      }
  ],
  "htd:version": "1",
  "htd:selected_seq": "11",
  "htd:numpages": "262",
  "htd:seqmap": [
      {
          "htd:seq": {
              "htd:imgfmt": "image/tiff",
              "htd:pnum": "7",
              "htd:pfeat": [
                  "FIRST_CONTENT_CHAPTER_START",
                  "IMPLICIT_PAGE_NUMBER"
              ],
              "pseq": "11"
          }
      }
  ],
  "htd:access": [
      {
 "content": "http://schemas.hathitrust.org/htd/2009#restricted",
          "resource": "pageimage"
      },
      {
```

```
 "content": "http://schemas.hathitrust.org/htd/2009#restricted",
          "resource": "pageocr"
      },
      {
 "content": "http://schemas.hathitrust.org/htd/2009#restricted",
          "resource": "aggregate"
      }
  ],
  "htd:rights": {
      "htd:user": "jhovater",
      "htd:note": {},
      "htd:time": "2008-08-14T22:30:23",
      "htd:namespace": "mdp",
      "htd:source": "1",
      "htd:attr": "2",
      "htd:id": "39015005102796",
      "htd:reason": "1"
  },
  "htd:pgmap": [
      {
          "htd:pg": {
              "content": "11",
              "pgnum": "7"
          }
      }
  ],
}
```

Relax NG Schema - Compact

```
default namespace = "http://www.w3.org/2005/Atom"
namespace htd = "http://schemas.hathitrust.org/htd/2009"
start =
 element entry {
   element link {
     attribute href { xsd:anyURI },
     attribute rel { xsd:anyURI },
     attribute type { text }
   }+,
   element htd:version { xsd:integer },
   element htd:selected_seq { xsd:integer },
   element htd:numpages { xsd:integer },
   element htd:seqmap {
     element htd:seq {
       attribute pseq { xsd:integer },
       element htd:pnum { xsd:NCName },
       element htd:pfeat { xsd:NCName }+,
       element htd:imgfmt { text }
     }
   },
   element htd:pgmap {
```

```
    element htd:pg {
      attribute pgnum { xsd:NCName },
      xsd:integer
    }
  },
  element id { xsd:anyURI },
  element title { text },
  element updated { xsd:NMTOKEN },
  element htd:access {
    attribute resource { xsd:NCName },
    xsd:anyURI
  }+,
  element htd:rights {
    element htd:note { empty },
    element htd:user { xsd:NCName },
    element htd:time { xsd:NMTOKEN },
    element htd:namespace { xsd:NCName },
    element htd:source { xsd:integer },
    element htd:attr { xsd:integer },
    element htd:id { xsd:integer },
    element htd:reason { xsd:integer }
  }
}
```

Appendix F: Example `structure` Response

(Edited for brevity)

```
<entry
xmlns="http://www.w3.org/2005/Atom"
xmlns:htd="http://schemas.hathitrust.org/htd/2009">
<link
  rel="alternate"
  href="http://hdl.handle.net/2027/mdp.39015064570875"
  type="text/html"/>
<link
  rel="self"
href="http://babel.hathitrust.org/cgi/htd/structure/
mdp.39015064570875"
  type="application/xml"/>
<link
  rel="http://schemas.hathitrust.org/htd/2009#aggregate"
href="https://babel.hathitrust.org/cgi/htd/aggregate/
mdp.39015064570875"
  type="application/zip"/>
<link
  rel="http://schemas.hathitrust.org/htd/2009#meta"
  href="http://babel.hathitrust.org/cgi/htd/meta/mdp.39015064570875"
  type="application/xml"/>
<htd:version>1</htd:version>
<id>http://babel.hathitrust.org/cgi/htd/structure/mdp.39015064570875</
id>
```

```
<title>HathiTrust Repository Data API - METS</title>
<updated>2009-03-12T13:33:59.933-0400</updated>
<htd:access resource="pageocr">http://schemas.hathitrust.org/htd/
2009#limited
</htd:access>
<htd:access resource="pageimage">http://schemas.hathitrust.org/htd/
2009#limited
</htd:access>
<htd:access resource="aggregate">http://schemas.hathitrust.org/htd/
2009#restricted
</htd:access>
<htd:rights>
  <htd:note/>
  <htd:user>jhovater</htd:user>
  <htd:time>2007-09-10T09:30:04</htd:time>
  <htd:namespace>mdp</htd:namespace>
  <htd:source>1</htd:source>
  <htd:attr>9</htd:attr>
  <htd:id>39015064570875</htd:id>
  <htd:reason>1</htd:reason>
</htd:rights>
<METS:mets
  xmlns:METS="http://www.loc.gov/METS/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:PREMIS="http://www.loc.gov/standards/premis"
  xsi:schemaLocation="http://www.loc.gov/METS/
  http://www.loc.gov/standards/mets/mets.xsd
  http://purl.org/dc/elements/1.1/"
  OBJID="mdp.39015064570875"
  xml:base="/sdr1/obj/mdp/pairtree_root/39/01/50/64/57/08/75/
39015064570875/39015064570875.mets.xml">
  <METS:metsHdr ID="mdp.39015064570875" CREATEDATE="2008-06-
05T16:06:23" RECORDSTATUS="NEW">
    <METS:agent ROLE="CREATOR" TYPE="ORGANIZATION">
      <METS:name>DLPS</METS:name>
    </METS:agent>
  </METS:metsHdr>
  <METS:dmdSec ID="DMD1">
    <METS:mdRef
      MDTYPE="MARC"
      LOCTYPE="OTHER"
      OTHERLOCTYPE="Item ID stored as second call number in item
record"
      XPTR="mdp.39015064570875"/>
  </METS:dmdSec>
  <METS:amdSec>
    <METS:techMD ID="TMD1">
      <METS:mdRef
        LOCTYPE="OTHER"
        OTHERLOCTYPE="SYSTEM"
        MDTYPE="OTHER"
```

```
        OTHERMDTYPE="text"
        LABEL="production notes"
        xlink:href="notes.txt"/>
    </METS:techMD>
    <METS:techMD ID="TMD2">
      <METS:mdRef
        LOCTYPE="OTHER"
        OTHERLOCTYPE="SYSTEM"
        MDTYPE="OTHER"
        OTHERMDTYPE="text"
        LABEL="page metadata"
        xlink:href="pagedata.txt"/>
    </METS:techMD>
    <METS:techMD ID="premisobject1">
      <METS:mdWrap MDTYPE="PREMIS">
        <METS:xmlData>
          <PREMIS:object>
            <PREMIS:preservationLevel>1</PREMIS:preservationLevel>
          </PREMIS:object>
        </METS:xmlData>
      </METS:mdWrap>
    </METS:techMD>
    <METS:digiprovMD ID="premisevent1">
      <METS:mdWrap MDTYPE="PREMIS">
        <METS:xmlData>
          <PREMIS:event>
            <PREMIS:eventIdentifier>
<PREMIS:eventIdentifierValue>capture1</PREMIS:eventIdentifierValue>
            </PREMIS:eventIdentifier>
            <PREMIS:eventType>capture</PREMIS:eventType>
<PREMIS:eventDateTime>2007-06-18T00:00:00</PREMIS:eventDateTime>
            <PREMIS:linkingAgentIdentifier>
<PREMIS:linkingAgentIdentifierType>AgentID</
PREMIS:linkingAgentIdentifierType>
              <PREMIS:linkingAgentIdentifierValue>Google, Inc.</
PREMIS:linkingAgentIdentifierValue>
            </PREMIS:linkingAgentIdentifier>
          </PREMIS:event>
          <PREMIS:event>
            <PREMIS:eventIdentifier>
<PREMIS:eventIdentifierValue>compression1</
PREMIS:eventIdentifierValue>
            </PREMIS:eventIdentifier>
            <PREMIS:eventType>compression</PREMIS:eventType>
<PREMIS:eventDateTime>2007-09-05T10:09:00</PREMIS:eventDateTime>
            <PREMIS:linkingAgentIdentifier>
<PREMIS:linkingAgentIdentifierType>AgentID</
PREMIS:linkingAgentIdentifierType>
              <PREMIS:linkingAgentIdentifierValue>Google, Inc.</
PREMIS:linkingAgentIdentifierValue>
            </PREMIS:linkingAgentIdentifier>
          </PREMIS:event>
```

```
          <PREMIS:event>
            <PREMIS:eventIdentifier>
<PREMIS:eventIdentifierValue>decryption1</PREMIS:eventIdentifierValue>
            </PREMIS:eventIdentifier>
            <PREMIS:eventType>decryption</PREMIS:eventType>
<PREMIS:eventDateTime>2007-09-08T02:57:45</PREMIS:eventDateTime>
            <PREMIS:linkingAgentIdentifier>
<PREMIS:linkingAgentIdentifierType>AgentID</
PREMIS:linkingAgentIdentifierType>
<PREMIS:linkingAgentIdentifierValue>UM</
PREMIS:linkingAgentIdentifierValue>
            </PREMIS:linkingAgentIdentifier>
          </PREMIS:event>
          <PREMIS:event>
            <PREMIS:eventIdentifier>
<PREMIS:eventIdentifierValue>fixity check1</
PREMIS:eventIdentifierValue>
            </PREMIS:eventIdentifier>
            <PREMIS:eventType>fixity check</PREMIS:eventType>
<PREMIS:eventDateTime>2007-09-08T02:57:45</PREMIS:eventDateTime>
            <PREMIS:eventOutcomeInformation>
<PREMIS:eventOutcomeDetail>pass</PREMIS:eventOutcomeDetail>
            </PREMIS:eventOutcomeInformation>
            <PREMIS:linkingAgentIdentifier>
<PREMIS:linkingAgentIdentifierType>AgentID</
PREMIS:linkingAgentIdentifierType>
<PREMIS:linkingAgentIdentifierValue>UM</
PREMIS:linkingAgentIdentifierValue>
            </PREMIS:linkingAgentIdentifier>
          </PREMIS:event>
          <PREMIS:event>
            <PREMIS:eventIdentifier>
<PREMIS:eventIdentifierValue>ingestion1</PREMIS:eventIdentifierValue>
            </PREMIS:eventIdentifier>
            <PREMIS:eventType>ingestion</PREMIS:eventType>
<PREMIS:eventDateTime>2007-09-08T02:57:45</PREMIS:eventDateTime>
            <PREMIS:linkingAgentIdentifier>
<PREMIS:linkingAgentIdentifierType>AgentID</
PREMIS:linkingAgentIdentifierType>
<PREMIS:linkingAgentIdentifierValue>UM</
PREMIS:linkingAgentIdentifierValue>
            </PREMIS:linkingAgentIdentifier>
          </PREMIS:event>
          <PREMIS:event>
            <PREMIS:eventIdentifier>
<PREMIS:eventIdentifierValue>message digest calculation1</
PREMIS:eventIdentifierValue>
            </PREMIS:eventIdentifier>
<PREMIS:eventType>message digest calculation</PREMIS:eventType>
<PREMIS:eventDateTime>2007-09-05T10:09:00</PREMIS:eventDateTime>
            <PREMIS:eventDetail>jhove1_1e</PREMIS:eventDetail>
            <PREMIS:linkingAgentIdentifier>
```

```
<PREMIS:linkingAgentIdentifierType>AgentID</
PREMIS:linkingAgentIdentifierType>
              <PREMIS:linkingAgentIdentifierValue>Google, Inc.</
PREMIS:linkingAgentIdentifierValue>
            </PREMIS:linkingAgentIdentifier>
          </PREMIS:event>
          <PREMIS:event>
            <PREMIS:eventIdentifier>
<PREMIS:eventIdentifierValue>validation1</PREMIS:eventIdentifierValue>
            </PREMIS:eventIdentifier>
            <PREMIS:eventType>validation</PREMIS:eventType>
<PREMIS:eventDateTime>2007-09-08T02:57:45</PREMIS:eventDateTime>
            <PREMIS:linkingAgentIdentifier>
<PREMIS:linkingAgentIdentifierType>AgentID</
PREMIS:linkingAgentIdentifierType>
<PREMIS:linkingAgentIdentifierValue>UM</
PREMIS:linkingAgentIdentifierValue>
            </PREMIS:linkingAgentIdentifier>
          </PREMIS:event>
        </METS:xmlData>
      </METS:mdWrap>
    </METS:digiprovMD>
  </METS:amdSec>
  <METS:fileSec>
    <METS:fileGrp ID="FG1" USE="zip archive">
      <METS:file ID="ZIP00000001" MIMETYPE="application/zip"
SEQ="00000001" CREATED="2008-06-05T16:06:23" SIZE="7065774"
CHECKSUM="b26aff19a616a83eefd0ffcb43be10b0" CHECKSUMTYPE="MD5">
<METS:FLocat LOCTYPE="OTHER" OTHERLOCTYPE="SYSTEM"
xlink:href="39015064570875.zip"/>
      </METS:file>
    </METS:fileGrp>
    <METS:fileGrp ID="FG2" USE="image">
      <METS:file ID="IMG00000001" MIMETYPE="image/tiff" SEQ="00000001"
CREATED="2007-09-05T13:07:51" SIZE="1498"
CHECKSUM="ad1491cf5c381b7752e5b53f3b50621c" CHECKSUMTYPE="MD5">
<METS:FLocat LOCTYPE="OTHER" OTHERLOCTYPE="SYSTEM"
xlink:href="00000001.tif"/>
      </METS:file>
      <METS:file ID="IMG00000002" MIMETYPE="image/jp2" SEQ="00000002"
CREATED="2007-09-05T13:07:52" SIZE="37689"
CHECKSUM="3c6300639e57f0e9305f6130565aab51" CHECKSUMTYPE="MD5">
<METS:FLocat LOCTYPE="OTHER" OTHERLOCTYPE="SYSTEM"
xlink:href="00000002.jp2"/>
      </METS:file>
      <METS:file ID="IMG00000003" MIMETYPE="image/tiff" SEQ="00000003"
CREATED="2007-09-05T13:07:52" SIZE="1972"
CHECKSUM="b097fa8086b852c07f169b63c3064cc8" CHECKSUMTYPE="MD5">
<METS:FLocat LOCTYPE="OTHER" OTHERLOCTYPE="SYSTEM"
xlink:href="00000003.tif"/>
      </METS:file>
```

```
      <METS:file ID="IMG00000004" MIMETYPE="image/tiff" SEQ="00000004"
CREATED="2007-09-05T13:07:52" SIZE="1970"
CHECKSUM="8d82918c9d351b07f92d84d0d2c98897" CHECKSUMTYPE="MD5">
<METS:FLocat LOCTYPE="OTHER" OTHERLOCTYPE="SYSTEM"
xlink:href="00000004.tif"/>
      </METS:file>
    </METS:fileGrp>
    <METS:fileGrp ID="FG3" USE="ocr">
      <METS:file ID="TXT00000001" MIMETYPE="text/plain" SEQ="00000001"
CREATED="2007-09-05T13:07:51" SIZE="0"
CHECKSUM="d41d8cd98f00b204e9800998ecf8427e" CHECKSUMTYPE="MD5">
<METS:FLocat LOCTYPE="OTHER" OTHERLOCTYPE="SYSTEM"
xlink:href="00000001.txt"/>
      </METS:file>
      <METS:file ID="TXT00000002" MIMETYPE="text/plain" SEQ="00000002"
CREATED="2007-09-05T13:07:52" SIZE="52"
CHECKSUM="226bdcccdb2089e12129a77c137a0eef" CHECKSUMTYPE="MD5">
<METS:FLocat LOCTYPE="OTHER" OTHERLOCTYPE="SYSTEM"
xlink:href="00000002.txt"/>
      </METS:file>
      <METS:file ID="TXT00000003" MIMETYPE="text/plain" SEQ="00000003"
CREATED="2007-09-05T13:07:52" SIZE="0"
CHECKSUM="d41d8cd98f00b204e9800998ecf8427e" CHECKSUMTYPE="MD5">
<METS:FLocat LOCTYPE="OTHER" OTHERLOCTYPE="SYSTEM"
xlink:href="00000003.txt"/>
      </METS:file>
      <METS:file ID="TXT00000004" MIMETYPE="text/plain" SEQ="00000004"
CREATED="2007-09-05T13:07:52" SIZE="0"
CHECKSUM="d41d8cd98f00b204e9800998ecf8427e" CHECKSUMTYPE="MD5">
<METS:FLocat LOCTYPE="OTHER" OTHERLOCTYPE="SYSTEM"
xlink:href="00000004.txt"/>
      </METS:file>
    </METS:fileGrp>
  </METS:fileSec>
  <METS:structMap ID="SM1" TYPE="physical">
    <METS:div TYPE="volume">
<METS:div ORDER="1" TYPE="page" LABEL="FRONT_COVER,
IMPLICIT_PAGE_NUMBER, MISSING_PAGE">
        <METS:fptr FILEID="IMG00000001"/>
        <METS:fptr FILEID="TXT00000001"/>
      </METS:div>
      <METS:div ORDER="2" TYPE="page" LABEL="IMAGE_ON_PAGE,
UNTYPICAL_PAGE, IMPLICIT_PAGE_NUMBER">
        <METS:fptr FILEID="IMG00000002"/>
        <METS:fptr FILEID="TXT00000002"/>
      </METS:div>
<METS:div ORDER="3" TYPE="page" LABEL="BLANK, IMPLICIT_PAGE_NUMBER">
        <METS:fptr FILEID="IMG00000003"/>
        <METS:fptr FILEID="TXT00000003"/>
      </METS:div>
<METS:div ORDER="4" TYPE="page" LABEL="BLANK, IMPLICIT_PAGE_NUMBER">
        <METS:fptr FILEID="IMG00000004"/>
```

```
        <METS:fptr FILEID="TXT00000004"/>
      </METS:div>
    </METS:div>
  </METS:structMap>
</METS:mets>
</entry>
```

RELAX NG Schema - Compact

```
default namespace = "http://www.w3.org/2005/Atom"
namespace METS = "http://www.loc.gov/METS/"
namespace PREMIS = "http://www.loc.gov/standards/premis"
namespace htd = "http://schemas.hathitrust.org/htd/2009"
namespace xlink = "http://www.w3.org/1999/xlink"
namespace xsi = "http://www.w3.org/2001/XMLSchema-instance"
start =
element entry {
 element link {
   attribute href { xsd:anyURI },
   attribute rel { xsd:anyURI },
   attribute type { text }
 }+,
 element htd:version { xsd:integer },
 element id { xsd:anyURI },
 element title { text },
 element updated { xsd:NMTOKEN },
 element htd:access {
   attribute resource { xsd:NCName },
   xsd:anyURI
 }+,
 element htd:rights {
   element htd:note { empty },
   element htd:user { xsd:NCName },
   element htd:time { xsd:NMTOKEN },
   element htd:namespace { xsd:NCName },
   element htd:source { xsd:integer },
   element htd:attr { xsd:integer },
   element htd:id { xsd:integer },
   element htd:reason { xsd:integer }
 },
 element METS:mets {
   attribute OBJID { xsd:NCName },
   attribute xsi:schemaLocation { text },
   attribute xml:base { text },
   element METS:metsHdr {
     attribute CREATEDATE { xsd:NMTOKEN },
     attribute ID { xsd:NCName },
     attribute RECORDSTATUS { xsd:NCName },
     element METS:agent {
       attribute ROLE { xsd:NCName },
       attribute TYPE { xsd:NCName },
```

```
      element METS:name { xsd:NCName }
    }
  },
  element METS:dmdSec {
    attribute ID { xsd:NCName },
    mdRef
  },
  element METS:amdSec {
    element METS:techMD {
      attribute ID { xsd:NCName },
      (mdRef | mdWrap)
    }+,
    element METS:digiprovMD {
      attribute ID { xsd:NCName },
      mdWrap
    }
  },
  element METS:fileSec {
    element METS:fileGrp {
      attribute ID { xsd:NCName },
      attribute USE { text },
      element METS:file {
        attribute CHECKSUM { text },
        attribute CHECKSUMTYPE { xsd:NCName },
        attribute CREATED { xsd:NMTOKEN },
        attribute ID { xsd:NCName },
        attribute MIMETYPE { text },
        attribute SEQ { xsd:integer },
        attribute SIZE { xsd:integer },
        element METS:FLocat {
          attribute LOCTYPE { xsd:NCName },
          attribute OTHERLOCTYPE { xsd:NCName },
          attribute xlink:href { xsd:NMTOKEN }
        }
      }+
    }+
  },
  element METS:structMap {
    attribute ID { xsd:NCName },
    attribute TYPE { xsd:NCName },
    \div
  }
 }
}
mdRef =
element METS:mdRef {
 attribute LABEL { text }?,
 attribute LOCTYPE { xsd:NCName },
 attribute MDTYPE { xsd:NCName },
 attribute OTHERLOCTYPE { text },
 attribute OTHERMDTYPE { xsd:NCName }?,
 attribute XPTR { xsd:NCName }?,
```

```
 attribute xlink:href { xsd:NCName }?
}
mdWrap =
element METS:mdWrap {
 attribute MDTYPE { xsd:NCName },
 element METS:xmlData {
   element PREMIS:object {
     element PREMIS:preservationLevel { xsd:integer }
   }
   | element PREMIS:event {
       element PREMIS:eventIdentifier {
         element PREMIS:eventIdentifierValue { text }
       },
       element PREMIS:eventType { text },
       element PREMIS:eventDateTime { xsd:NMTOKEN },
       (element PREMIS:eventDetail { xsd:NCName }
        | element PREMIS:eventOutcomeInformation {
            element PREMIS:eventOutcomeDetail { xsd:NCName }
          })?,
       element PREMIS:linkingAgentIdentifier {
         element PREMIS:linkingAgentIdentifierType { xsd:NCName },
         element PREMIS:linkingAgentIdentifierValue { text }
       }
     }+
 }
}
\div =
element METS:div {
 attribute LABEL { text }?,
 attribute ORDER { xsd:integer }?,
 attribute TYPE { xsd:NCName },
 (\div,
  element METS:fptr {
    attribute FILEID { xsd:NCName }
  }+)?
}
```