Date submitted: 07/06/2010

Building A Future By Preserving Our Past: The Preservation Infrastructure of HathiTrust Digital Library

Jeremy York
Project Librarian, HathiTrust Digital Library
University of Michigan
Ann Arbor, MI, USA
jjyork@umich.edu

Meeting: **157. ICADS with Information Technology**

WORLD LIBRARY AND INFORMATION CONGRESS: 76TH IFLA GENERAL CONFERENCE AND ASSEMBLY
10-15 August 2010, Gothenburg, Sweden
http://www.ifla.org/en/ifla76

Abstract:

HathiTrust is a growing partnership of research institutions that have pooled resources to digitally preserve and provide access to the vast record of published literature. This paper describes the infrastructure the HathiTrust partners have implemented and the technical strategies they are employing to achieve these goals.

Introduction

In 2008 a group of major U.S. research libraries came together with a common vision to build a new kind of library for the 21st century – a cooperative library founded on principles of commitment, deep resource sharing, and trust, that was so comprehensive in its representation of the published record, so available to users anywhere in the world, and so broad in its impact on the fundamental business of what libraries do, that it could rise to be called a universal library. They named this library HathiTrust.

HathiTrust was formed explicitly as a strategy to preserve and provide access to the published scholarly record, with the knowledge that no such endeavor could succeed without the deep cooperation of co-supporting and co-owning libraries from around the world.[1] Using

[1] HathiTrust invites partnership from institutions around the world.

expertise gained over nearly two decades of experience in digital libraries and employing community standards and best practices in preservation and information management, the founding partners created a robust and scalable infrastructure that would be capable of supporting this grand undertaking.

This paper provides a detailed description of the HathiTrust infrastructure, including design rationales, architecture, functional components and workflow. It elaborates the strategies HathiTrust partners are taking to ensure that our record of scholarship and research is preserved and accessible long into the digital future.

Philosophy and Design

HathiTrust was designed according to the framework for Open Archival Information Systems (OAIS)[2] to fulfill ingest, archival storage, data management, and access functions for large (millions of volumes) amounts of material (see *Figure 1*). This design is realized within the context of community-wide standards and criteria for Trustworthy Digital Repositories.[3]

The logistics of operating a preservation repository at this scale[4] have led to implementation solutions that favor consistency and standardization over variance, simplicity over complexity (in design, not function), and practicality over ideology. The repository functions above all to meet the preservation and access needs of the HathiTrust partners. Although by extension HathiTrust serves a much broader constituency, it is these needs specifically that drive the development of HathiTrust services and capabilities.

This functional, need-driven approach has resulted in a modular and robust architecture where discrete components of the archival environment communicate and interoperate as an integrated whole. Although many repository systems are located on central servers, HathiTrust's modularity, in combination with the use of open standards and open service definitions (APIs), makes it possible for partner institutions to develop services and key pieces of functionality, supporting the long-term sustainability of the operation. Modularity also permits an agile response to problems that may arise (e.g., issues with ingest, storage, or access components of the repository are localized and can be addressed separately).

The architecture and functionality of the repository are described below, preceded by a description of the content in HathiTrust.

Content/Information Package

The initial focus of HathiTrust is on preserving and providing access to the large amounts of digitized book and journal content produced by member libraries through partnerships with entities such as Google and the Internet Archive, as well as local digitization initiatives. By design, and aided tremendously by the homogeneity of Google-digitization outputs[5], content in the repository is largely uniform in its technical properties and characteristics. HathiTrust strives to maintain this uniformity, though there is variation due to the differing practices and specifications involved in different digitization workflows.

---

[2] OAIS. Consultative Committee for Space Data Systems. *The Open Archival Information System Reference Model (OAIS).* Washington, D.C.: CCSDS Secretariat, National Aeronautics and Space Administration, 2002.

[3] TRAC. *Trustworthy Repositories Audit & Certification: Criteria and Checklist.* Center for Research Libraries and OCLC, 2007. http://www.crl.edu/sites/default/files/attachments/pages/trac_0.pdf

[4] As of May 14, HathiTrust contains 5.9 million volumes, more than 1 million of which are in the public domain.

[5] The specifications for Google content and digitization metadata were determined collaboratively by Google library partners.

The repository contains three primary image formats: JPEG, JPEG2000, and ITU G4 (bitonal) TIFF. Derivative formats stored include OCR text and coordinate OCR, which supplements OCR text with information about the location of each word on a page. The typical digital object package consists of image and OCR files representing the original volume, and two METS files that conform to the Metadata Encoding and Transmission Standard[6]: a "source" METS file that contains information about the object from the time of its creation to the time it enters the repository, and a "HathiTrust" METS file that includes a subset of this data, but is primarily a record of the object from the time it enters the repository forward.[7] The former is kept for preservation purposes only. The latter is used for both preservation and access purposes (i.e., in both the archival and dissemination information packages).[8]

Because of this dual use, while the provenance and preservation information recorded in the HathiTrust METS is complete, it is relatively succinct. The overarching approach to technical metadata in the repository is to record the details of systematic actions on and specifications for digital objects in external documentation where possible, with the understanding that deposited items either conform to this documentation, contain small variations that are recorded locally in the items themselves, or do not conform and are not admitted for deposit. While the details of specifications and processes may be recorded externally, at a minimum an indication of the actions or events that occur in relation to an object is recorded in the HathiTrust METS (e.g., transformation, validation, ingest, etc.).

Images, OCR, source METS files, and other associated digital object files are stored in a single directory in the repository as a zip archive. The HathiTrust METS is stored in the same directory but is not compressed in order to facilitate use for access and management purposes.

Ingest

The Trustworthy Repository Audit and Certification (TRAC) documentation describes ingest in the OAIS framework as "the processes that take place from the receipt of the digital objects through the final, preserved form of that object in the repository."[9] HathiTrust distinguishes between two distinct components in the ingest process: one involving ingest of bibliographic metadata into a central bibliographic management system, and one involving ingest of content, including primary and derivative content formats and associated metadata, into repository storage. Within content ingest, HathiTrust distinguishes between ingest proper, and activities that occur prior to ingest to transform content to conform (as closely as possible) to repository standards.

Bibliographic Metadata Ingest

The most common way that bibliographic metadata is received into HathiTrust is through download of prepared files from the source institution. Transfer can also be accomplished via email attachments or on removable media. Ingest of bibliographic data must occur before content is ingested into the repository and must conform to general requirements.[10] Any

[6] METS. http://www.loc.gov/standards/mets/

[7] Preservation information is recorded using PREMIS (Preservation Metadata Implementation Strategies. http://www.loc.gov/standards/premis/). A description of HathiTrust's use of PREMIS is available at http://www.hathitrust.org/digital_object_specificiations.

[8] Examples of source and HathiTrust METS files, as well as a profile for the HathiTrust METS, are available at http://www.hathitrust.org/digital_object_specifications.

[9] TRAC, pg.83.

[10] See http://www.hathitrust.org/ingest.

transformation of the metadata that is needed is performed by partners prior to submission, or, when there are standard systematic differences, information is transferred to match specifications on ingest.

The ingest process uses OCLC numbers to identify incoming bibliographic records that already exist in the repository. When there are existing records, new holdings and corresponding item records are attached to the existing record. Otherwise, a new bibliographic record is created.

Content Ingest

The most common ways that content is received into the repository are through download (e.g., from Google or the Internet Archive) or delivery on removable media. Once content is received, it goes either directly to ingest or proceeds through a series of pre-ingest processes to prepare the content for ingest.

Ingest

The ingest mechanism was designed to accommodate a high volume of materials (more than 500,000 volumes per month). It consists of an array of back-end servers devoted to ingest and other repository administration tasks (such as content backup), and a validation environment called GROOVE: the Google Return Object-oriented Validation Environment. As the name suggests, this environment was created initially to support ingest of partner content digitized by and downloaded from Google's servers, but it has been expanded to accommodate content from other sources.

The main functions of ingest are to 1) ensure that submitted content conforms to required specifications, 2) prevent items with unanticipated problems from entering the repository, and 3) create the working metadata file (HathiTrust METS) that will accompany the content in the repository. Ingest processes do not modify primary or derivative formats or perform metadata transformations beyond the normalizations needed to create the HathiTrust METS file. The simplicity of the ingest function results in very high efficiency and performance. The specific activities GROOVE performs include:

- Barcode Validation – the majority of identifiers in the repository are composed in part of the barcodes of physical volumes held by libraries. Barcodes are validated based on the particular schemes employed (e.g., character length, check digit, etc.).
- Fixity Check – new checksums are calculated for all content and compared with checksums received from the digitization source or produced in pre-ingest processes.
- Consistency Checks – checks are performed to ensure that, for example, there are no missing page sequences and that there is one-to-one correspondence between page images and OCR files.
- Well-formedness Check – Well-formedness and embedded metadata checks are performed on incoming files using JHOVE.[11]
- METS file creation – The HathiTrust METS file is created.
- Permanent URL – HathiTrust uses the Handle system[12] to assign permanent URLs to repository objects.
- Deposit – The HathiTrust METS and a zip archive containing all other files are written to storage.

[11] JHOVE. http://hul.harvard.edu/jhove/
[12] Handle System. http://handle.net/

Once objects have passed through these steps, they await a periodic cycle where they are replicated to each of HathiTrust's storage instances (see *Archival Storage* below). Ingest is not considered complete until this replication has occurred.

Pre-Ingest

Although the majority of volumes in the repository are products of Google digitization, not all are, and there can be significant variation in the characteristics of objects originating from different digitization sources. To preserve the high performance of repository ingest given this variation, HathiTrust developed a separate pre-ingest phase in the flow of digital deposit. In this phase, digital objects are evaluated for compliance with repository standards, including well-formedness of content and metadata requirements. Following this evaluation, HathiTrust 1) determines the standards and specifications that will apply to objects from a particular source (if different from repository norms) and 2) prepares content for ingest by modifying or transforming it to meet those standards and specifications.

Standardization across the repository, to the degree possible, is of utmost concern to HathiTrust. Where variation is required, it is documented either at the submission level (e.g., if content from a specific source varies systematically from validation norms) or at the item level (if individual objects vary from the norm in acceptable ways). Repository consistency must be balanced with the desire and need to preserve the digital assets submitted by partner institutions.

The separation of ingest and pre-ingest functions serves both to maintain the efficiency of the ingest operation (devoting different hardware and resources to each to avoid competition) and to support the modularity of the repository. Although HathiTrust currently is performing pre-ingest transformations, the tools and processes will eventually be made available to partner institutions to prepare their own content for submission to HathiTrust.

Archival Storage

The key requirements of storage in HathiTrust are reliability (maximum ability to ensure integrity of content), redundancy (duplication of data within single and at multiple sites), scalability (including ease of management at scale), and accessibility (availability of content for use in repository services). It is also important that archival storage does not manage objects or dictate storage structure in a way that prevents content from being migrated easily to other platforms. The storage media HathiTrust uses, the distributed storage architecture it implements, and the external management processes it employs are important to fulfilling these requirements and ensuring a robust preservation environment.

Media

HathiTrust uses Isilon Systems[13] storage, whose features and characteristics are optimal for present needs. It is disk-based and has a high degree of internal redundancy, preventing data loss in the event of failure of multiple disks or storage nodes. Its replication software is highly reliable, ensured by checksum validation, and is able to sync content between HathiTrust storage sites securely and efficiently. Continual system checks on data integrity, and detection and repair of corrupted disk sectors provide a high degree of protection against bit rot or other content degradation. Isilon storage is also highly scalable: additional storage nodes can be installed and made operational within a matter of minutes, and a single file system can be extended across a cluster of nodes up to more than five petabytes in size. The features Isilon provides are important to the integrity and maintenance of repository content,

[13] Isilon Systems. http://www.isilon.com/

but the media itself is only one component of HathiTrust's archival storage strategy.

Architecture and Management

HathiTrust maintains two geographically separated instances of repository storage – one in Michigan and one in Indiana. Content is ingested at the Michigan location and replicated to Indiana on a frequent, periodic basis.[14] The sites function with load balancing and failover, so users do not know when they visit HathiTrust if they are accessing volumes stored at the Michigan or the Indiana site. Content is also backed up on magnetic tape, which is stored at a third location in Michigan. HathiTrust replaces storage regularly, every 3-4 years, or as the usable life of storage equipment dictates.

Content is mounted on a single file system in the repository. Each digital object is located in a distinct directory (containing a zip archive and the HathiTrust METS) and objects are aggregated into namespaces that identify different sources of materials (e.g., Indiana University, the University of Minnesota, etc.). If items within an institution have more than one distinct identifier scheme, multiple namespaces are used for that institution. Some examples of object identifiers are:

Google-digitized:
University of Wisconsin: wu.89094366424
University of California: uc1.b3543486
University of Michigan: mdp.39015037375253

Internet-Archive digitized:
University of California: uc2.ark:/13960/t26973133

Locally digitized:
University of Michigan: miun.aaj0523.1950.001

Objects are stored on the file system in a Pairtree directory structure.[15] Pairtree is a hierarchical scheme that maps identifier strings to the paths of the objects they identify, two characters at a time. For instance, the file path for a University of Michigan object would be:

../mdp/pairtree_root/39/01/50/37/37/52/53/39015037375253/

39015037375253.mets.xml
39015037375253.zip

The Pairtree scheme enables content to be imported, understood, and used in a new storage system without the system knowing anything about the nature or contents of the stored objects. It also allows many object operations, such as backup and restore, to be performed with native operating system tools.[16] Organizing objects according to the Pairtree scheme facilitates HathiTrust's ability to recover from disastrous circumstances or migrate to a new storage platform if needed. It also ensures that objects are uniformly accessible to repository systems and access services, such as bibliographic and full text search, and collection-building and page-viewing tools.

[14] Currently daily, and generally not exceeding two days.
[15] CDL Pairtree. https://confluence.ucop.edu/display/Curation/PairTree
[16] J. Kunze, M. Haye, E. Hetzner, M. Reyes, C. Snavely. "Pairtrees for Object Storage". 2008. https://confluence.ucop.edu/download/attachments/14254128/PairtreeSpec.pdf?version=1.

Data Management

The primary data management activities that occur in the repository are bibliographic data management and management of rights information. The systems that support these activities are distinct, but interact in highly integrated ways to provide important inventory and reporting capabilities for the repository and support HathiTrust's access services.

Bibliographic Data Management

HathiTrust is currently using Ex Libris' Aleph library management system for bibliographic management, but the functions of management are not particular to Aleph or tied to the system in specific ways. The primary functions of bibliographic management are to:

1. Provide the official inventory of objects in the repository
2. Accommodate loading of bibliographic records from partner institutions, including updates to records
3. Supply the content ingest mechanism (GROOVE) with the list of objects that can be ingested
4. Detect and collate multiple volumes (and/or copies) associated with a single title
5. Perform automated rights determinations for all ingested volumes
6. Accept and process changes to bibliographic records (e.g. updates to title, author, publication information) and communicate those changes to other systems
7. Export information to build the Solr[17] indexes underlying HathiTrust's VuFind catalog.[18] These indexes are used for
    a. VuFind catalog search and display
    b. Index input for full text search
    c. Support for the HathiTrust Bibliographic API
    d. Display of the source (contributing) institution in the VuFind catalog
8. Support HathiTrust's manual copyright review processes[19]

The bibliographic management system is also the source of information for

1. PageTurner display data
2. Collection Builder display data
3. HathiTrust tab-delimited metadata files
4. OAI output sets
5. Administrative/statistical reporting

Once an object corresponding to a given bibliographic record has entered the repository, a series of management activities are set in motion that include 1) official acceptance of the object into the repository (following replication), indicated by a marker in the bibliographic record; 2) initiation of an automated rights determination process, conducted on every object using bibliographic information such as publisher, publication date, and publication location;

[17] Solr. http://lucene.apache.org/solr/.

[18] VuFind. http://vufind.org/. HathiTrust released a temporary bibliographic catalog in April 2009 based on VuFind and Solr. It is currently developing a permanent catalog in collaboration with OCLC.

[19] Staff at the University of Michigan are currently performing manual copyright review of volumes in HathiTrust and constructing a Copyright Review Management System (CRMS) as part of an IMLS funded grant project. Phase one of the CRMS was completed in April 2010, and phase two, where review will be extended to staff at other HathiTrust partner institutions, is expected to begin in the fall of 2010.

and 3) recording of rights determinations in a rights database (see *Rights Management* below) and in the bibliographic system itself.[20]

Rights Management

All objects that enter the repository have a copyright status, whether that status is known or unknown. In order to display any of the works it is preserving, HathiTrust must take steps to accurately determine the status of each volume. The inputs available for rights determination are bibliographic metadata, which is essential for HathiTrust's automated determination process, and manual review of individual volumes. All rights determinations are stored in a rights database.

Because rights determinations are made through multiple methods (automated and manual), and may occur multiple times (for instance, updates to a bibliographic record or manual review after the initial rights determination may change the status of a volume), the rights database is organized in a system of precedence where some rights determinations are more authoritative than others. The lowest level of precedence is the bibliographic determination that occurs when volumes are ingested. To avoid providing access to copyrighted volumes by mistake, bibliographic determination is relatively straightforward, and conservative.[21]

Manual review has the next, and higher level of precedence. There are three levels of review, or reasons, a volume may be determined to have a particular status. These reasons include, in order of precedence from lowest to highest:

1. No printed copyright notice was found, copyright of a volume was not renewed, or condition review and in-print status research was conducted (possibly making works that are in copyright available for special uses under U.S. Copyright law[22]).
2. Contractual agreements exist with publishers.
3. Access control overrides. These may be made in such cases where the results of copyright research are inconclusive, yet HathiTrust is able assert with high confidence that a particular rights determination is correct.

Eleven reason codes are logged in the rights database currently, and nine corresponding attributes (e.g., public domain, public domain in the U.S., or in copyright).[23]

Access

The importance HathiTrust places on both preservation and access capabilities is expressed in the close relationship between the archival storage, management, and access components of the repository. The access model consists of a layer of services and applications that sit directly on top of storage and management sources, enabling a variety of uses. The primary archival objects themselves are used in these services, accompanied by bibliographic and rights information to make a coherent dissemination package. Use of primary objects in dissemination activities facilitates management of access over time since applications do not

[20] The rights database is the authoritative source for rights determination information, but determinations are included in the bibliographic data to be indexed and made available in access services.
[21] The rationale for copyright determinations is described at http://www.hathitrust.org/rights_management and http://www.hathitrust.org/rights_database.
[22] Under section 108 of U.S. copyright law, U.S. libraries are able to make a digital copy of in copyright works available to users on their library premises if the original work is damaged, deteriorating, lost or stolen, and not available on the market at a reasonable price.
[23] Codes and explanations are listed at http://www.hathitrust.org/rights_database.

need to be written to accommodate a variety of derivative formats, and preservation, since continual usage provides an important check on content integrity.

Access in HathiTrust falls into three main categories: 1) Content Access, in which image, textual content,[24] and associated metadata are delivered to users; 2) Search and Aggregation Access, in which searchable indexes of repository are used to find and link to content itself; and 3) Metadata Access in which descriptive, volume, and rights metadata are made available to users through targeted channels such as feeds and APIs.

Content Access

Mechanisms for content access include the PageTurner application, which supports sequential reading of volumes as images, OCR text, or in Portable Document Format (PDF), and the Data API, which returns entire object packages including METS files, image and text files,[25] and rights metadata.

When a user requests pages of a volume through PageTurner, the application retrieves archival images and OCR text from the repository (literally extracts them from the zip file), and transforms the images on the fly into a derivative format for display (either PNG or PDF, depending on the request; OCR text is displayed without transformation). At the same time, PageTurner retrieves the HathiTrust METS file from the object directory, descriptive metadata from the bibliographic management system, and rights information from the rights database. It uses these data to determine the order of pages for display (from the METS file), provide some contextual information, and allow or deny access to the volume. PageTurner is optimized for accessibility[26] and includes a specialized interface that allows users with print disabilities to view the full text of both public domain and copyrighted volumes in a single browser window.

The Data API is a RESTful[27] interface to the repository, delivering objects or components of objects (e.g., a single image or OCR page, a single METS file), as well as rights information, to users via download or XML response.[28] The Data API was created to enable institutions to develop their own interfaces to access HathiTrust content (such as the PageTurner and Collection Builder applications), and to facilitate other uses such as content validation and auditing.[29]

Search and Aggregation Access

Mechanisms for search and aggregation include the VuFind bibliographic catalog, full text search, and Collection Builder. All of these mechanisms are served by Solr search indexes, and share a similar architecture: a Solr index stands between one or more data sources and an outward-facing application.

The Solr index serving the VuFind catalog includes full MARC records as well as rights determination information, facilitating bibliographic search, and faceting of results by

[24] HathiTrust plans to expand to other formats such as audio as well.
[25] The Data API does not deliver images and text for objects that are in copyright, or those that were digitized by Google.
[26] A illustrated description of the accessible interface is included in Suzanne Chapman, Heather Christensen, Paul Fogel and Jeremy York “HathiTrust: Preservation as a Platform for Collaboration and Expanded User Services”, a poster presentation given at the iPRES conference in October 2009.
[27] http://www.xfront.com/REST-Web-Services.html
[28] The specification for the API is located at http://www.hathitrust.org/data_api.
[29] Staff from the University of Michigan have built a demonstration application showing how the Data and Bibliographic APIs, and HathiTrust metadata files might be used to browse and retrieve content from the repository. It is located at http://www.lib.umich.edu/two-over-threehundred.

bibliographic data and viewability status.[30] The VuFind Solr index also plays a role in full text search indexing. Full text search uses this index to determine the queue of volumes that are eligible for indexing or re-indexing and as the source for the bibliographic data used in search results display. The full text search index includes these data, OCR text extracted from the repository, and information from the rights database, which is used to allow or limit display of search results.[31] The full text index is replicated at Michigan and Indiana, as are the web servers that handle user queries.

Collection Builder employs a third Solr index, containing the full text of all volumes that have been saved to collections by users, as well as an indication of the collections the volumes belong to. Individual collections can thus be searched independently of others. Volumes saved in one collection can be readily copied into an existing or new collection, facilitating searching across multiple collections. Collection Builder also extracts OCR directly from the repository for indexing, and limited descriptive information from the bibliographic management system is saved into a collections database.

Metadata Access

Users can retrieve metadata about objects in HathiTrust in several ways. These include OAI harvesting, HathiTrust tab-delimited metadata files, a Bibliographic API, and a Data API.[32] The first three mechanisms are commonly used to insert HathiTrust records into local catalogs. They are also tools that partners and others can use to analyze and repurpose repository content. The tab-delimited files serve a key reporting function for the repository, delivering a daily feed of ingested volumes including rights information, a variety of standard identifiers, and some descriptive information. With the exception of the Data API, distribution of these metadata relies on the data management components of the repository, not on archival storage.

Conclusion

Leveraging existing community standards such as OAIS and TRAC and collective resources, HathiTrust has in a relatively short period of time constructed a secure and robust environment for digitally preserving and providing access to the published content of its member libraries. The technical infrastructure of the repository is only one aspect of the broader organization and resources framework on which the longevity of HathiTrust depends, but it exemplifies how libraries, through cooperation and shared resources, can create a whole that is greater than the sum of their individual parts. HathiTrust was created to preserve the rich record of scholarship and research that libraries have collected over centuries of time. By successfully doing so, and making this record broadly available, HathiTrust is simultaneously building a foundation for scholarship and advanced inquiry to continue long into the future.

[30] Viewability status is based on rights status. Public domain and open access volumes in HathiTrust are "Full view" to the public, meaning they can be searched and read. In copyright works are "Limited (search-only)" – bibliographic and full text information can be searched, but content of the volume is not available for reading.
[31] Similar to reading restrictions on in copyright volumes, the view of search results for in copyright works is more limited than works that are in the public domain.
[32] More information about these methods of data distribution is available at http://www.hathitrust.org/data. The Data API is included in the Metadata Access section because it can be queried to return volume metadata independently of content files in an object package.

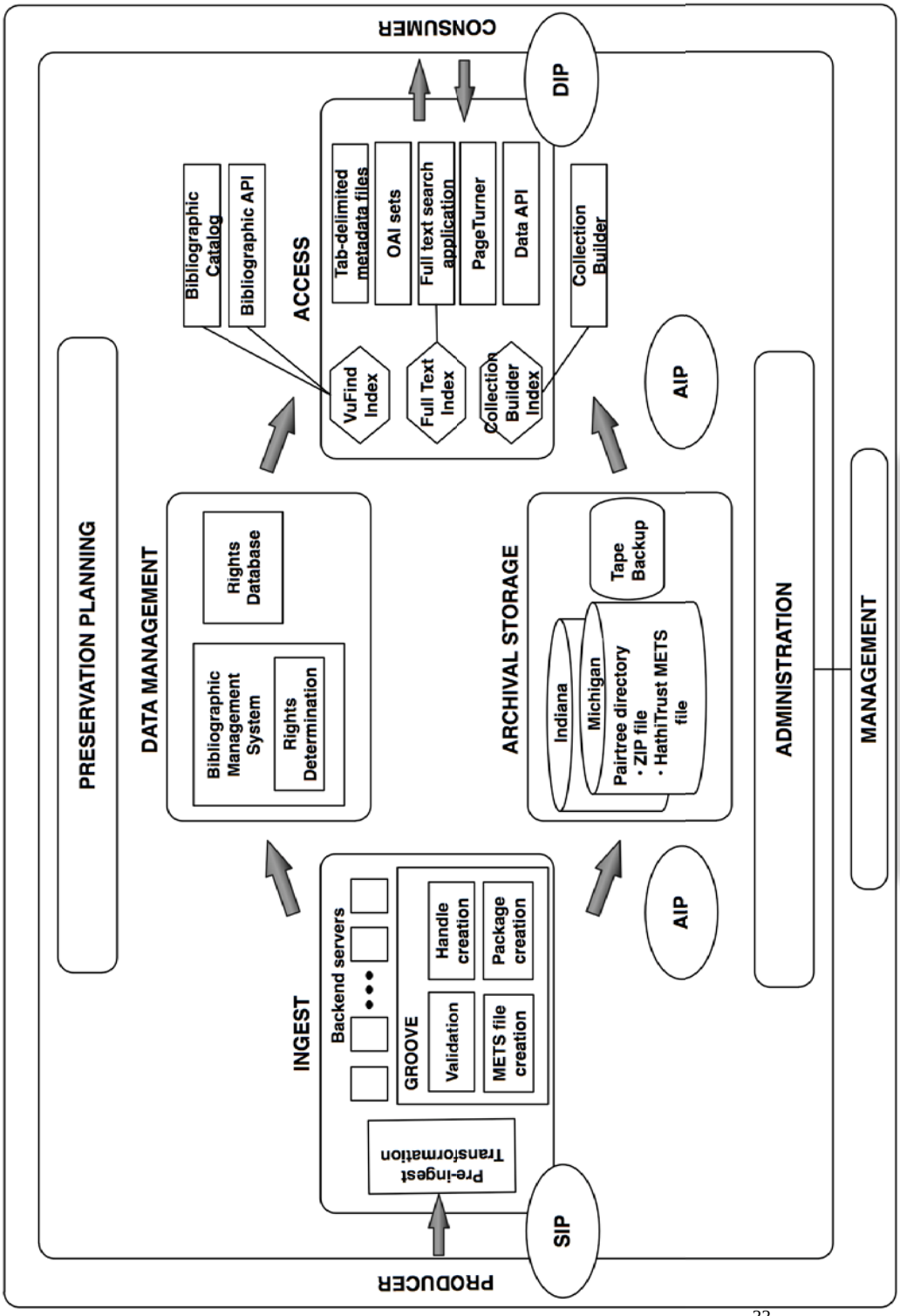

Figure 1. HathiTrust architecture according to OAIS framework[33]

[33] OAIS, pg.4-1. Functional Model.