ECP-2008-DILI-518001

BHL-Europe

Document agreed standards, best practice and system components

Deliverable number	*D3.2*
Dissemination level	*Restricted*
Delivery date	*26 November 2009*
Status	*Final*
Author(s)	*AIT, ATOS, MfN, NHM*

e*Content*plus

This project is funded under the *e*Content*plus* programme[1],
a multiannual Community programme to make digital content in Europe more accessible, usable and exploitable.

[1] OJ L 79, 24.3.2005, p. 1.

Table of contents

1 Document History

This chapter describes the document's creation events and contributors.

1.1 Contributors

This document is based on the meetings of the technical team in Leiden, Graz and Berlin with the following members contributing to this document.

Person	Partner
Roger Essoh	ATOS
Alexander Herzog	AIT
Gerda Koch	AIT
Walter Koch	AIT
Lee Namba	ATOS
Henning Scholz	MfN
Adrian Smales	NHM
Kai Stalmann	MfN

1.2 Revision History

Revision Date	Author	Version	Change Reference & Summary
Nov 02, 2009	AIT, ATOS, MfN	1.0	Consolidated document
Nov 09, 2009	NHM	1.1	Consistency editing
Nov 10, 2009	EDLF	1.2	Review of version 1.0
Nov 26, 2009	NHM	2.0	Final version

The document itself was written on the BHLwiki to be reviewed by and discussed with the BHL-Europe consortium from 15 October onwards, resulting in the consolidated document provided on 2 November. This document was reviewed again for the final version provided here.

1.3 Reviewers and Approvals

This document requires the following approvals.

Name	Position	Date	Version
Walter Koch	Deliverable responsible	26 Nov 2009	2.0
Henning Scholz	Project coordinator	26 Nov 2009	2.0

1.4 Distribution

This document has been distributed to:

Group	Date of issue	Version
BHL-Europe consortium	2 November 2009	1.0
BHL-Europe consortium	27 November 2009	2.0

2 Purpose and document structure

This document describes agreed standards, best practice and future system components.

The BHL-Europe is primarily a digital collection of material that is important for the study of the natural world. As such, the project will adhere to existing standards and best practice that have already been established.

The Institute of Museum and Library Services (**IMLS**) have established nine principles for the creation and maintenance of digital collections and it is these principles, along with the ISO standard known as the Open Archival Information System (**OAIS**), that provide the foundations for this project.

Principles that apply to good digital collections (see: http://framework.niso.org/) are:

Collections Principle 1: A good digital collection is created according to an explicit collection development policy.

Collections Principle 2: Collections should be described so that a user can discover characteristics of the collection, including scope, format, restrictions on access, ownership, and any information significant for determining the collection's authenticity, integrity, and interpretation.

Collections Principle 3: A good collection is curated, which is to say, its resources are actively managed during their entire lifecycle.

Collections Principle 4: A good collection is broadly available and avoids unnecessary impediments to use. Collections should be accessible to persons with disabilities, and usable effectively in conjunction with adaptive technologies.

Collections Principle 5: A good collection respects intellectual property rights.

Collections Principle 6: A good collection has mechanisms to supply usage data and other data that allows standardized measures of usefulness to be recorded.

Collections Principle 7: A good collection is interoperable.

Collections Principle 8: A good collection integrates into the users own workflow.

Collections Principle 9: A good collection is sustainable over time.

As a result of the technical meetings in spring and summer 2009 it was decided that this document will focus on the Open Archival Information System (**OAIS**)[2] reference model that is used for the Biodiversity Heritage Library for Europe (**BHL-Europe**).

It will describe the characteristics of the transmitted data, the processes within the components and the overall process. OAIS was developed by the Consultative Committee for Space Data Systems (**CCSDS**) of the US National Aeronautics and Space Administration. The OAIS framework is today the most relevant standard in digital preservation "with the purpose to establish a system for archiving information, both digitalized and physical, with an organizational scheme composed of people who accept the responsibility to preserve information and make it available to a designated community". It is published as ISO Standard ISO 14721:2003[3].

Two additional components are needed for the BHL-Europe system. As metadata harmonization and augmentation has to be done before the Submission Information Package (**SIP**) creation to have full control over the created metadata, the Pre-Ingest will handle these activities. To display the augmented metadata correctly a Portal will use the Access interfaces to present multilingual interfaces and search functionality to the user.

To create an in-depth understanding of these components the document will describe the interfaces as well as the communication between these.

In chapter 3 and 4 the document describes the interfaces and payload and the technical considerations for the BHL-Europe system. Chapter 5 lays the focus on the detailed description of each component of the BHL-Europe system. In addition e*ach component and payload* will be described with the *agreed standards* and the *best practice* to combine the various components.

[2] Reference Model for an Open Archival Information System (OAIS). CCSDS 650.0-B-1, Blue Book, January 2002, Online http://public.ccsds.org/publications/archive/650x0b1.pdf [30.10.2009]

[3] ISO 14721:2003, Space data and information transfer systems -- Open archival information system -- Reference model, http://www.iso.org/iso/catalogue_detail.htm?csnumber=24683 [30.10.2009]

3 Interfaces and Payload

This chapter describes the transmitted data within the extended OAIS system used for BHL-Europe. Most of the exchanged data will be XML encoded and will be based on established standards.

↓To \ From →	Pre-Ingest	Ingest	Data Manag-ement	ArchivalStorage	Preserv-ation Planning	Adminis-tration	Access	Portal
Pre-Ingest	**X**	Receipt Confirmation			Surveys	Lien, Final Ingest Report, Submission Agreement		
Ingest	SIP	**X**	Database Update Response, Report Response	Storage Confirmation		Standards / Procedures, [updates] SIP, Audit Report		
Data Management		DI, Database Update request, Report Request	**X**			Policies, Report Requests	Report Request, Query Request	
Archival Storage		AIP		**X**		Policies	AIP Request	
Preservation Planning	Product technology				**X**	Consumer Comments, Migration Goals, Performance Info, Inventory Report		Service Requirements
Administration	Appeal, Submission Agreement	SIP / AIP [audit]	Report, Status Of Updates	Operational Statistics	Proposals / Recommendations, AIP/SIP templates, Migration Packages	**X**	DIP, Billing Info	Payment
Access			DI Report, Result Set	AIP, Notice of Data Transfer		Dissemination Request	**X**	Order Request, Query Request, Assistance Request, Report Request
Portal					Survey	Bill	DIP	**X**

It is necessary to distinguish between an Information Package that is preserved by an OAIS and the Information Packages that are submitted to, and disseminated from, an OAIS. These

variant packages are needed to reflect the reality that some submissions to an OAIS will have insufficient Representation Information or Preservation Description Information (**PDI)** to meet final OAIS preservation requirements. In addition, these may be organized very differently from the way the OAIS organizes the information it is preserving. Finally, the OAIS may provide information to Consumers that does not include all the Representation Information or all the PDI with the associated Content Information being disseminated. These variants are referred to as the Submission Information Package (**SIP**), the Archival Information Package (**AIP**), and the Dissemination Information Package (**DIP**).

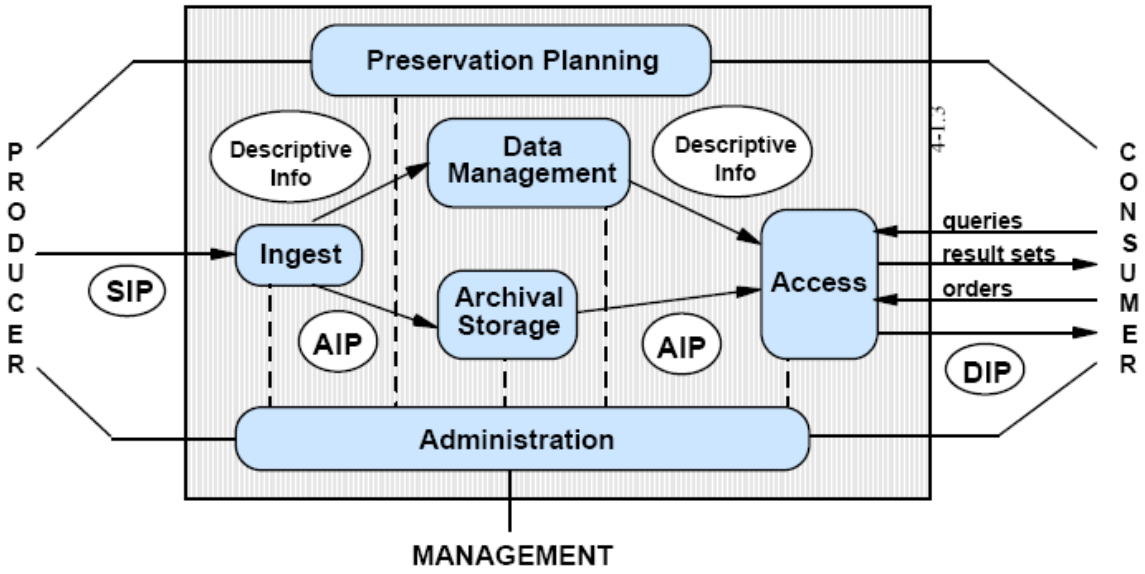


Figure 3-1: OAIS Functional Entities[4]

3.1 Submission Information Package (SIP)

The Submission Information Package (SIP) is the payload used at the interface between PreIngest and OAIS_Ingest. It is the base information package for Descriptive Information (DI) and Archival Information Package (AIP)

The SIP is a Metadata Encoding and Transmission Standard (**METS**) profile[5]. It contains references to scanned digital data, OCR data, structural information for displaying a table of content and related items. The structural information is also used to created composed items. The Descriptive Metadata is represented by **MARCXML**. MARC is a widespread standard

[4] Reference Model for an Open Archival Information System (OAIS). CCSDS 650.0-B-1, Blue Book, January 2002, Online http://public.ccsds.org/publications/archive/650x0b1.pdf [30.10.2009]

[5] http://www.loc.gov/standards/mets/mets-profiles.html [30.10.2009]

in the librarian sector and MARCXML serializes MARC data as XML. As BHL-Europe has more specific needs than the regular MARCXML schema can provide, a semantic description of fields has to be created. This description will be called BHL Deduplication Subset of Metadata (**BDS**). The BDS will be described in Descriptive Information (DI) as it is used for deduplication.

3.1.1 Standards

The standards for this package derive from established XML encoding.

3.1.1.1 MARCXML http://www.loc.gov/standards/marcxml/

The Library of Congress' Network Development and MARC Standards Office developed a framework for working with MARC data in a XML environment.
MARC XML could potentially be used as follows:

- for representing a complete MARC record in XML
- as an extension schema to METS (Metadata Encoding and Transmission Standard)
- to represent metadata for Open Archives Initiative (**OAI**) harvesting
- for original resource description in XML syntax
- for metadata in XML that may be packaged with an electronic resource, forming an information object.

3.1.1.2 Metadata Encoding and Transmission Standard (METS) http://www.loc.gov/standards/mets/

The METS schema is a standard for encoding descriptive, administrative, and structural metadata regarding objects within a digital library, expressed using the XML schema language of the World Wide Web Consortium (**W3C**). The standard is maintained in the Network Development and MARC Standards Office of the Library of Congress, and is being developed as an initiative of the Digital Library Federation (**DLF**).

METS provides an XML document format for encoding metadata necessary for both management of digital library objects within a repository and exchange of such objects between repositories (or between repositories and their users). Depending on its use, a METS document could be used in the role of Submission Information Package (SIP), Archival Information Package (AIP), or Dissemination Information Package (DIP) within the Open Archival Information System (OAIS) Reference Model.

A METS document consists of seven major sections:

1. **METS Header** (metadata describing the METS document itself)
2. **Descriptive Metadata** (may contain external, like MARC, or internally embedded descriptive metadata, or both)
3. **Administrative Metadata** (provides information regarding how the files were created and stored, intellectual property rights, etc.; may be either external to the METS document, or encoded internally.)

4. **File Section** (lists all files containing content)
5. **Structural Map** (outlines a hierarchical structure for the digital library object, and links the elements of that structure to content files and metadata that pertain to each element.)
6. **Structural Links** (allows METS creators to record the existence of hyperlinks between nodes in the hierarchy outlined in the Structural Map.)
7. **Behavior** (this section can be used to associate executable behaviors with content in the METS object.)

3.1.1.3 Metadata Object Description Schema (MODS) - http://www.loc.gov/standards/mods/

Metadata Object Description Schema (MODS) is a schema for a bibliographic element set that may be used for a variety of purposes, and particularly for library applications. The standard is maintained by the Network Development and MARC Standards Office of the Library of Congress with input from users. MODS is expressed using the XML (Extensible Markup Language) schema language. MODS was designed as a compromise between the complexity of the MARC format used by libraries and the extreme simplicity of Dublin Core metadata. It differs from MARCXML in making improved provision for digital material.

3.1.2 Best Practice and Implementation

The METS profile was chosen because it supplies all structural and descriptive information needed for BHL-Europe. It was also chosen by the following digital repository systems:

3.1.2.1 Fedora - http://www.fedora-commons.org

Open source software providing a service-oriented architecture for managing and delivering their digital content. Fedora is a integrated, repository system that enables storage, access and management for virtually any kind of digital content. Supports METS ingest.

3.1.2.2 DSPACE - http://dspace.org

A digital repository system that captures, stores, indexes, preserves, and distributes digital research material. Supports METS for ingest and export, as well as METS AIP

3.2 Archival Information Package (AIP)

The Archival Information Package (AIP) is the payload used at the interface between:

- OAIS_Ingest and OAIS_ArchivalStorage
- OAIS_ArchivalStorage and OAIS_Access

3.2.1 Standards

Because the AIP contains much of the same metadata and information as the SIP it also uses the same standards as the SIP such as MARCXML, METS, and MODS. In addition is uses the SWORD interface for deposit into OAIS_ArchivalStorage.

3.2.1.1 SWORD http://www.swordapp.org/

SWORD is a lightweight protocol for depositing content from one location to another. It stands for Simple Web-service Offering Repository Deposit and is a profile of the Atom Publishing Protocol (known as APP or ATOMPUB).

Funding has been provided by the Joint Information Systems Committee (**JISC**) to develop the SWORD profile and a number of demonstration implementations.

3.2.1.2 AtomPub http://www.ietf.org/rfc/rfc4287.txt

The Atom Publishing Protocol (AtomPub) is an application-level protocol for publishing and editing Web resources. The protocol is based on HTTP transfer of Atom-formatted representations. The Atom format is documented in the Atom Syndication Format.

3.2.2 Best Practice and Implementation

The SWORD protocol was chosen in order to provide an abstraction layer to the OAIS_ArchivalStorage module in the event that the underlying Archival Storage implementation should be changed in the future.

There are SWORD implementations to the following storage repositories:

- Fedora - http://www.fedora-commons.org
- DSPACE - http://dspace.org
- EPRINTS - http://www.eprints.org/

3.3 Descriptive Information (DI)

DI is used to describe packages in a way that allows for identifying records being sent.

DI is needed within Data Management to support the finding, ordering, and retrieving of OAIS information.

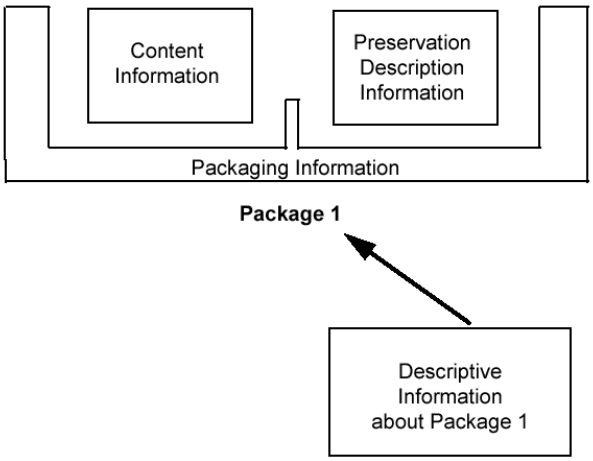
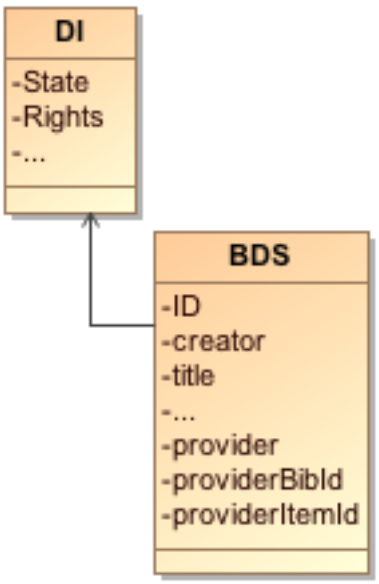


Figure 3-2 DI combines metadata and preservation finding aids

The OAIS must provide adequate features to allow Consumers to locate information of potential interest, analyze that information, and order desired information. This is accomplished through a specialization of the Information Object called Descriptive Information, which contain the data that serves as the input to documents or applications called Access Aids. The latter are documents or applications that can be used to locate, analyze, retrieve, or order information from the OAIS.



The Descriptive Information is generally derived from the metadata. For BHL-Europe there will be tools that allow mapping library data to metadata used for deriving DI.

The OAIS blueprint states that "Descriptive Information associated with a SIP is likely to be provided prior to submitting the SIP to the OAIS, but it may be provided at any time. It may be no more than a text description with a name or title, carried by the Packaging Information, by which the SIP may be recognized." p.4-32

However, BHL-Europe will assign the DI to the SIP after Ingest. For the Pre-Ingest the SIP may only contain the DI which contains information about the state of the object with respect to the planned scanning. It also will contain information about an digital rights agreement that may restrict the accessibility of the record. Finally, and most importantly, it contains the metadata describing the object. The object metadata here is a Dublin Core subset called BHL Deduplication Subset (BDS). The exact format of the BDS is described in a separate document.

The Descriptive Information is used to discover which package has the Content Information of interest. It is used as a blueprint for defining the database schema. The fields contained in the BDS go into the Index that is implemented using Lucene, the search engine library. While additional 'deep retrievals' using Archive data may be supported, the metadata that forms the BDS is the primary source of data retrieval from a client's perspective.

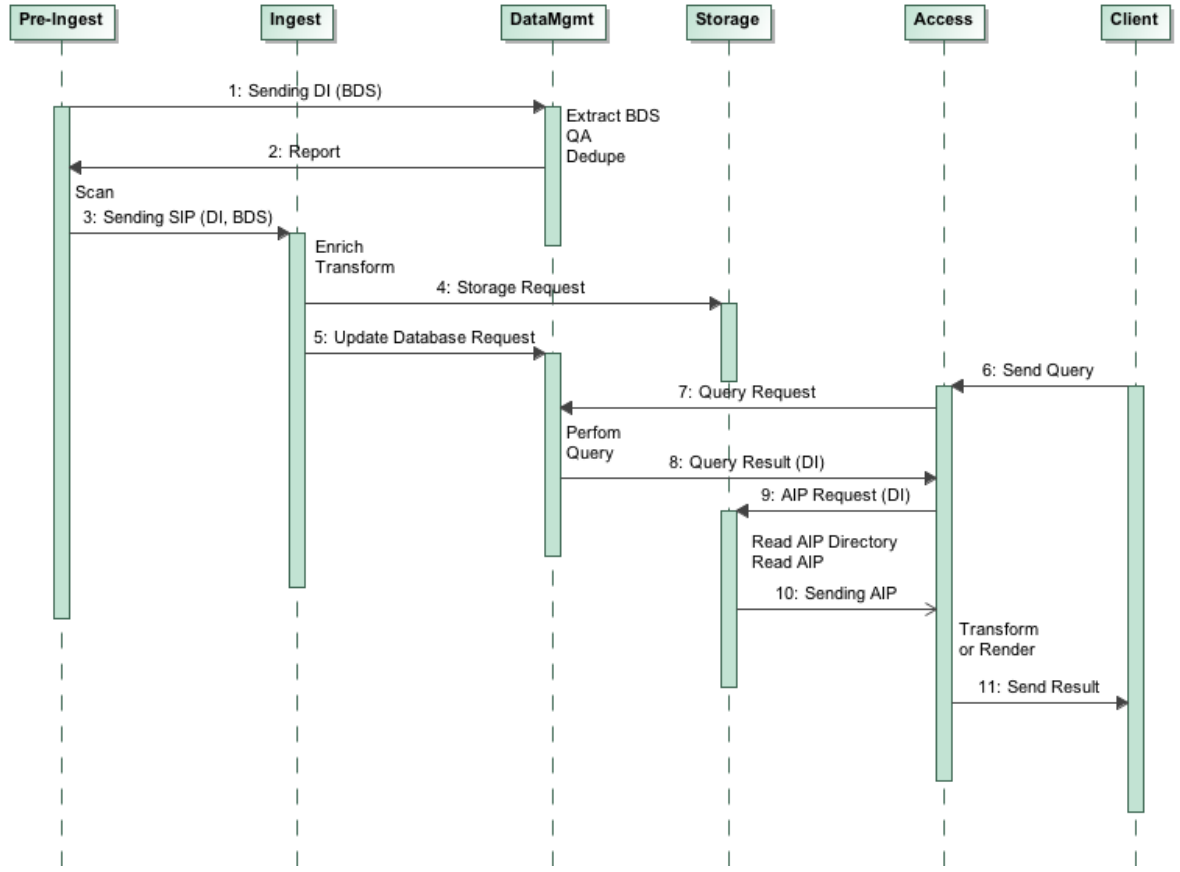


Figure 3-3 Sequence diagram showing how the process from Pre-Ingest to a User or Machine Query evolves around the DI.

3.4 Dissemination Information Package (DIP)

OAIS makes its AIPs visible and available to its Designated Communities (consumers). Multiple views of its holdings, supported by various search aids that may cut across

collections of AIPs, may be provided. They appear as Dissemination Information Packages (DIP)s. Consumers (BHL-Europe portal users) access only DIPs. Each DIP may contain a narrative describing the data file(s), the record layout and codes for the data, a methodology, sample input forms and questionnaires, annotations regarding the data validity, and a bibliography.

Co-operating archives with designated communities may order and ingest data from each other (eg. Europeana - BHL-Europe). At a minimum, co-operating archives must agree to support at least one common Submission Information Package (SIP) and Dissemination Information Package (DIP) for inter-archive requests. When one archive (e.g. Europeana) acts as a consumer of material from another archive the consuming archive must support the DIP format of the producing archive as a SIP format. For BHL-Europe that means that one of its DIPs needs to conform to the ESE schema (current version ESEv3.2 - http://group.europeana.eu/web/guest/provide_content) for collaboration with Europeana.

On the other hand users or consumers of several archives (like in BHL) may wish the following:

- common finding aids to aid in locating information across several archives;
- a common Package Description schema for access;
- a common DIP schema for dissemination; or
- a single global access site.

3.4.1 Standards

The standards for this package depend on its final form. For more details see OAIS_Access

3.4.2 BestPractice

The best practice for this package is based on the supported access methods.. For more details see OAIS_Access in section 5.6.

4 Technical Considerations

Next to the functional implementation of the components, these common services are needed. Infrastructure and messaging issues need to be defined before implementation.

4.1 Security

From a high level security standpoint the BHL-Europe system is a web based centralized system. It can be separated into three different categories which have appropriate levels of protection based upon the value of the information to the application end-users and the perception of threats to it:

1. An external public interface via the Portal component.
2. External private interfaces with partners to the OAIS_Ingest and OAIS_Access components.
3. Internal private interfaces between the components and for the administration of the system via OAIS_Administration.

4.1.1 Best Practices

Security services capabilities and mechanisms to protect sensitive information and treatments in the information system include the following.

4.1.1.1 Identification/authentication

Identification/authentication service confirms the identities of requesters for use of information system resources. In addition, authentication can apply to providers of data. The authentication service may occur at the initiation of a session or during a session.

OpenID – http://www.openid.net

OpenID is an open, decentralized standard for authenticating users which can be used for access control, allowing users to log on to different services with the same digital identity where these services trust the authentication body.

Shibboleth - http://shibboleth.internet2.edu/

Shibboleth is an Internet2 Middleware Initiative project that has created an architecture an open-source implementation for federated identity-based authentication and authorization infrastructure based on the Security Assertion Markup Language (**SAML**).

4.1.1.2 Access Control

Access control service prevents the unauthorized use of information system resources. This service also prevents the use of a resource in an unauthorized way. This service may be applied to specific aspects of access to a resource (e.g., access to communications to the resource; the reading, writing, or deletion of an information/data resource; the execution of a processing resource) or to all access to a resource.

XACML – http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacmlXACML stands for eXtensible Access Control Markup Language. It is a declarative access control policy language implemented in XML and a processing model, describing how to interpret the policies.

4.1.1.3 Data Integrity

Data integrity service provides an assurance that data has not been altered or destroyed in an unauthorized manner. This service applies to data in permanent data stores and to data in communications messages.

MD5

MD5 (Message-Digest algorithm 5) is a widely used cryptographic hash function with a 128-bit hash value. As an Internet standard (RFC 1321), MD5 has been employed in a wide variety of security applications, and is also commonly used to check the integrity of files. By comparing MD5 checksums a user can be assured that the object has been unaltered.

4.1.1.4 Data Confidentiality

Data confidentiality service ensures that data is not made available or disclosed to unauthorized individuals or computer processes. This service will be applied to devices that permit human interaction with the information system. In addition, this service will ensure that observation of usage patterns of communications resources will not be possible.

HTTPS

Hypertext Transfer Protocol Secure (HTTPS) is a combination of the Hypertext Transfer Protocol with the SSL/TLS protocol to provide encryption and secure identification of the server.

SSL/TLS - http://tools.ietf.org/html/rfc5246 Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL), are cryptographic protocols that provide security for communications over networks such as the Internet. TLS and SSL encrypt the segments of network connections at the Transport Layer end-to-end.

TLS is an IETF standards track protocol, last updated in RFC 5246, that was based on the earlier SSL specifications developed by Netscape Corporation.

4.1.1.5 Non-repudiation

Non-repudiation service ensures that entities engaging in an information exchange cannot deny being involved in it. This service may take one or both of two forms. First, the recipient of data is provided with proof of the origin of the data. This protects against any attempt by the sender to falsely deny sending the data or its contents. Second, the sender of data is provided with proof of delivery of data. This protects against any subsequent attempt by the recipient to falsely deny receiving the data or its contents.

X.509 Digital Certificates

X.509 is an ITU-T standard for a public key infrastructure (**PKI**) for single sign-on (**SSO**) and Privilege Management Infrastructure (**PMI**). X.509 specifies, amongst other things, standard formats for public key certificates, certificate revocation lists, attribute certificates, and a certification path validation algorithm.

Certificate Authority

A certificate authority or certification authority (**CA**) is an entity that issues digital certificates for use by other parties. It is an example of a trusted third party. CAs are characteristic of many public key infrastructure (PKI) schemes.

4.2 Service Interoperability

An OAIS-compliant system would most likely by be formed out of various modules that communicate and exchange data. The BHL-Europe Archive System is going to be implemented as a distributed system, which raises the question as to how these parts are glued together.

Since the by far most important core function of Data Management is to organize the means how data packages (SIP, DIP, AIP), database records, and indexes are statically held together (by references / identifiers) and also how they are dynamically kept in synchronization, Data Management can be seen as service organizing the service interoperability.

BHL-Europe decided implementation-specific definitions and to guide the development teams that work on the modules. To facilitate this, data will always travel around as XML, and we will define XML containers containing payload data (like METS data, and other important information).

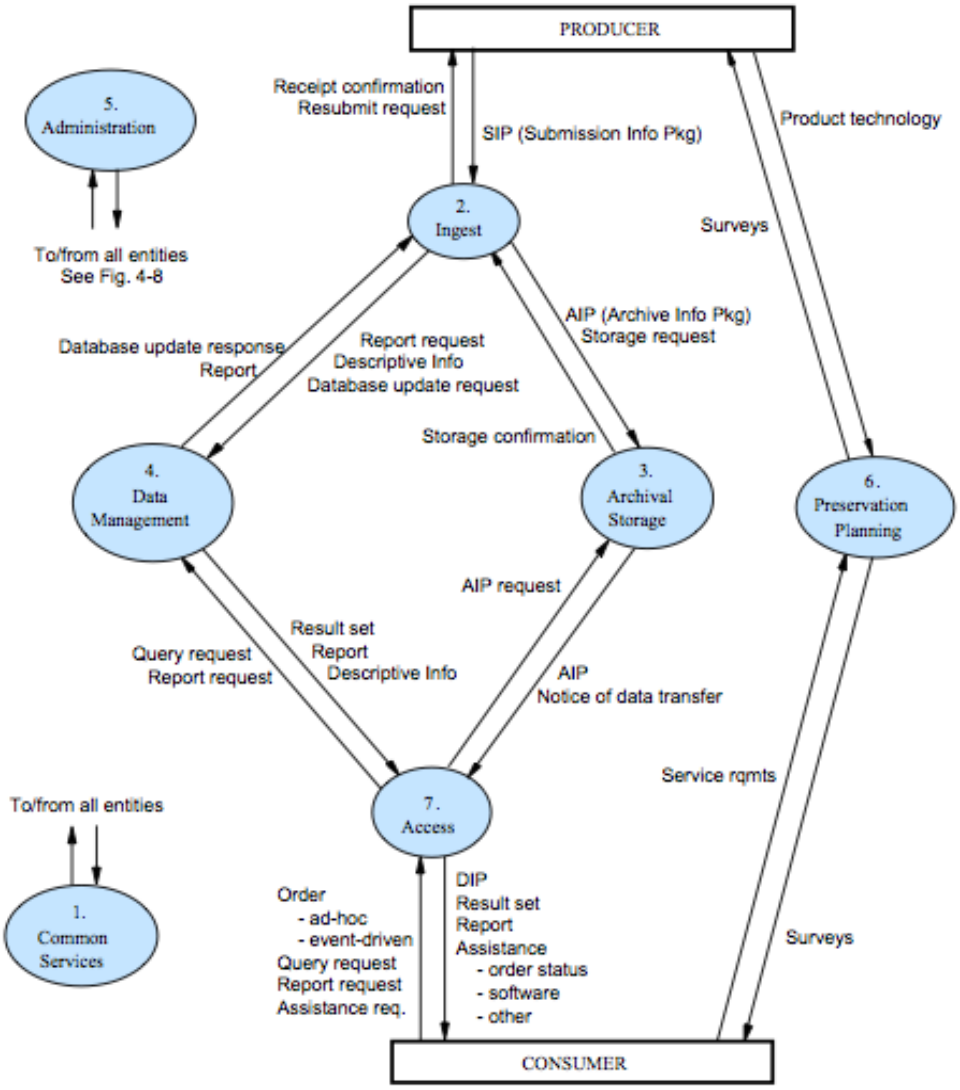


Figure 4-1: OAIS Data Flow Diagram

The OAIS Data Flow shown above revolves around a set of Web services that act as data sinks and sources. To facilitate the communication between these the BHL-Europe Archive System implements service endpoints in a way that complies to the architectural style called Representational state transfer (**REST**).

The REST architectural style describes six constraints:

1. **Client-server:** Clients are separated from servers by a uniform interface. This separation of concerns means that, for example, clients are not concerned with data storage, which remains internal to each server, so that the portability of client code is improved. Servers are not concerned with the user interface or user state, so that servers can be simpler and more scalable. Servers and clients may also be replaced and developed independently, as long as the interface is not altered.

2. **Stateless:** The client-server communication is further constrained by no client context being stored on the server between requests. Each request from any client contains all of the information necessary to service the request, and any state is held in the client. This not only makes servers more visible for monitoring, but also makes them more reliable in the face of partial or network failures as well as further enhancing their scalability.
3. **Cacheable:** As on the World Wide Web, clients are able to cache responses. Responses must therefore, implicitly or explicitly, define themselves as cacheable or not to prevent clients reusing stale or inappropriate data in response to further requests. Well-managed caching partially or completely eliminates some client-server interactions, further improving scalability and performance.
4. **Uniform interface:** The uniform interface between clients and servers, discussed below, simplifies and decouples the architecture, which enables each part to evolve independently. The four guiding principles of this interface are detailed below.
5. **Layered system:** A client cannot ordinarily tell whether it is connected directly to the end server, or to an intermediary along the way. Intermediary servers may improve system scalability by enabling load balancing and by providing shared caches. They may also enforce security policies.
6. **Code on demand (optional):** Servers are able temporarily to extend or customize the functionality of a client by transferring logic to it that it can execute. Examples of this may include compiled components such as Java applets and client-side scripts such as JavaScript.

The web services implemented by the BHL-Europe Archive System fully comply to RESTful web services:

- There is base URI for the web service, such as http://bhl.org:8088/archive/.
- There is a MIME type of the data supported by the web service, for using XML.
- The set of operations is using the HTTP methods POST, GET, PUT or DELETE.

The XML containers moved around by these services contain the payload and possibly routing information in case a rule or content based routing is used to orchestrate some sort of sequence of service calls.

4.3 Data Transformations

The Archival Storage functional entity takes the AIPs produced by the Ingest process and merges it into the permanent archive holdings. The Data Management functional entity takes the Package Descriptions produced by Ingest and augments the existing Collection Descriptions to include their contents. The logical model of the ingested data should map into the logical model of the archives holdings.

For a Data Dissemination Session, however, Access contacts the Storage and Data Management functional areas and requests the AIPs and associated Package Descriptions necessary to populate the DIP requested by the Consumer. The Storage and Data Management functional areas create copies of the requested objects in staging storage. Access then stores those DIPs onto distribution (either physical or communications) media to be delivered to the Consumer in a Data Dissemination Session.

In general, the complexity of this transformation process can differ greatly on the basis of the level of processing services offered by the OAIS and requested by the Consumer's order.

The simplest case is a 1:1 mapping of DIPs and AIPs. In this case the DIP contains duplicates of the AIPs and associated Package Descriptions of interest from Storage and Data Management function. This is very unlikely to be appropriate for the BHL-Europe Archive System, because the AIPs contain a huge amount of data (hundreds of scanned pages in high resolution) that cannot be withdrawn from the archive in a reasonable amount of time.

The OAIS model stipulates more complex cases where the desired Content Information may have to be extracted from the information objects, or where the granularity of the information objects may be changed, and the Dissemination process may generate DIPs and associated Package Descriptions reflecting the new granularity.

This model, which actually is much more likely to be adequate for BHL-Europe requirements, makes use of sub-setting services and other product processing options that would be located inside Data Management to create different DIPs based on combining many AIPs.

Creating a DIP based on combining many AIPs - turns out to be useful for BHL-Europe as it allows for splitting up the SIP into a set of AIPs that would selectively be used to form DIPs.

Three AIPs are logically combined to a compound AIP. This compound AIP is meant as a purely logical unit. Every compound AIP at least has one AIP that serves as a Directory. The Directory would contain title and item metadata, especially the BDS. It should contain IPR information, and it may contain thumbnails if available. This kind of Directory can be retrieved and read very fast, due to the fact that it is small in size. Data Management would always read the directory. In the case that the high resolution data is not in the Open Access but still in Long Time Preservation, it would usually not be delivered to the user. Note that special business models may be developed on grounds of this restriction.

Having the IPR concerning these possible restrictions outside the package containing hundreds of megabytes of page data helps prevent serious performance problems. The high resolution data could also be kept in an archive that is physically distinct from the BHL-Europe Open Access Archive.

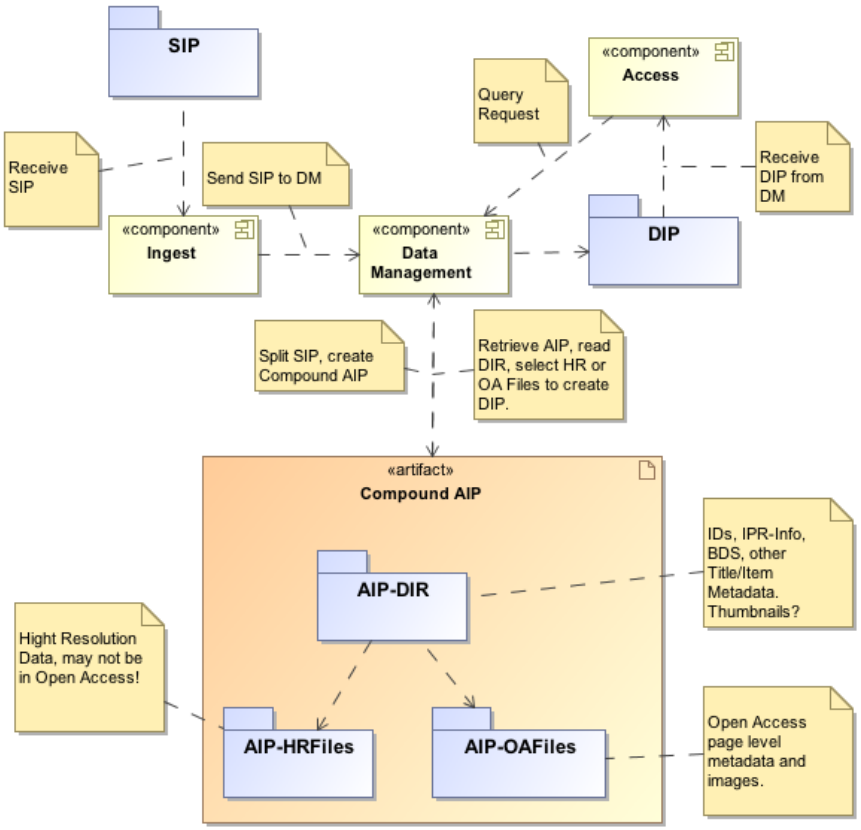


Figure 4-2: Data transformations: a SIP is broken up into three AIP that constitute a Compound AIP. On query side the Compound AIP is transformed into a DIP.

If Data Management decides that the high resolution data cannot be delivered, the related Open Access AIP is being read. This package would usually contain a METS file with structural information, meta data, and page level image and text data. The package would be transformed or simply copied into a DIP for delivery.

Maintaining a directory file that only contains metadata but not the image files (maybe with the exception of thumbnails) would bear the additional advantage that BHL could ingest in an iterative manner. It is likely and makes sense to receive only metadata during an initial Ingest.

After deduplication and consulting a bid list, the provider would start the scanning process - which can take a while. In the meantime BHL-Europe has a hold on the data about material that is planned for scanning, and BHL-Europe thus can prevent duplicate scans but would possibly not show up the records on the portal.

When scanning is done and the BHL-Europe Archive System receives the images during a second Ingest the additional AIPs are being created and the records can be shown on the portal.

5 Components

This chapter describes the components of the BHL-Europe system. As described in chapter 2 the components are based on the OAIS reference model. Each component within the BHL-Europe system will be implemented as already available systems. PreIngest and Ingest as well as Access and the Portal will make use of DISMARC multilingual search functionality concepts. As DISMARC does not have any form of preservation, ArchivalStorage will be implemented using Fedora. Each component described in the following subsections will contain best practice and standards. An overall view of the system can be seen in Figure 5-1 OAIS architecture.

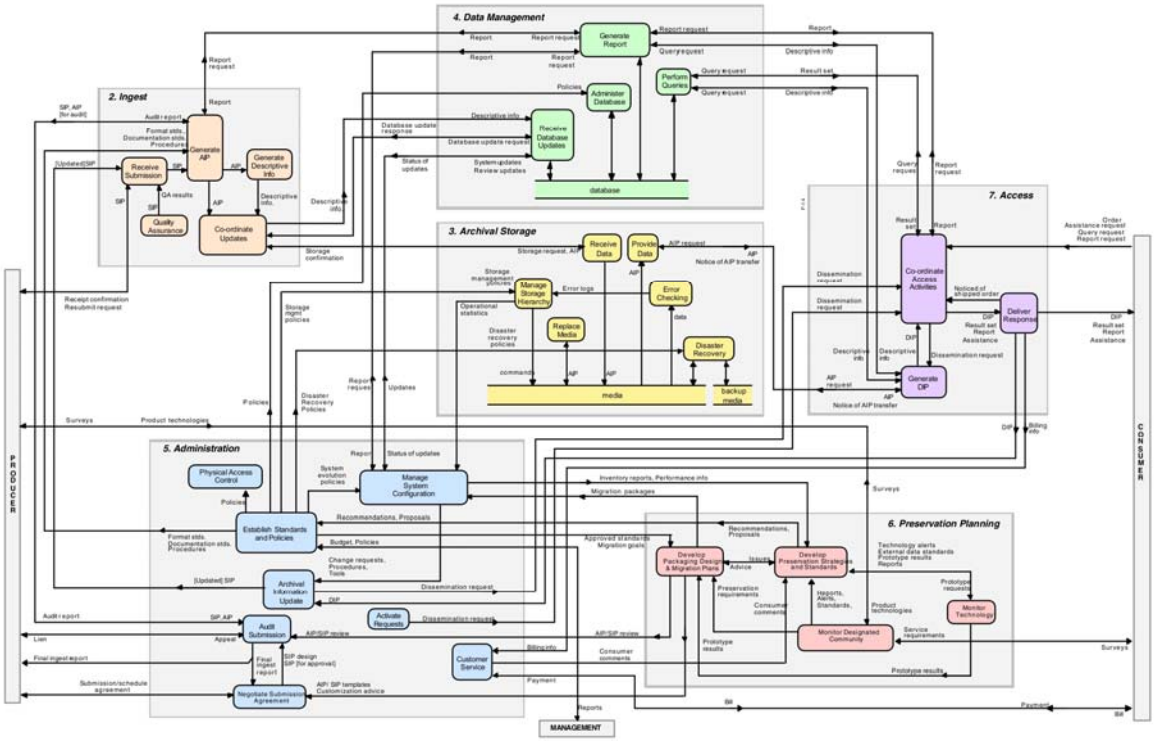


Figure 5-1 OAIS architecture

5.1 PreIngest

This component is the interface to the archives and acts as an adapter for the Ingest module. As external partners store meta data in various formats, the native formats need to be converted, harmonized, enriched and prepared for ingestion. This step is needed for the ingestion, multilingual search, data harmonization, indexation and search requirement.

5.1.1 Interfaces and Payload

- to OAIS_Ingest: Submission Information Package (SIP) via XML REST
- to OAIS_Administration: Lien/Appeal, final ingest report, submission agreement
- to OAIS_PreservationPlanning: Product features

5.1.2 Processes

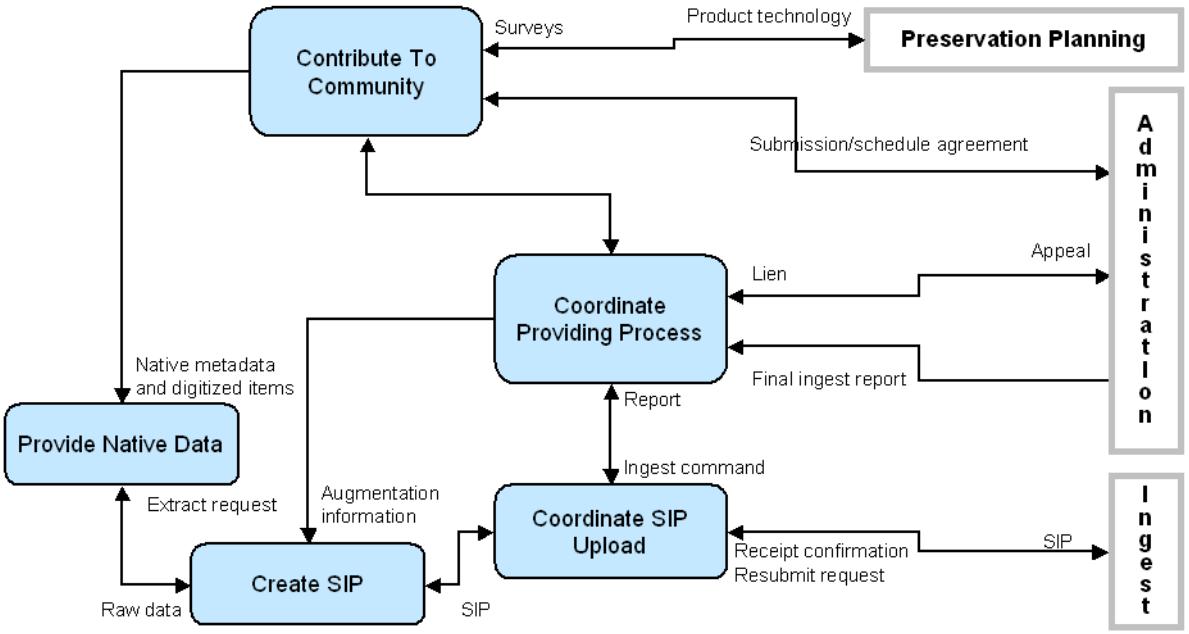


Figure 5-2 PreIngest processes

The **Contribute to Community** function initializes the contribution to BHL. Preservation Planning will be informed about the product technology in use. Administration will negotiate a submission agreement. The Provide Native Data function will be initialized as well as the Coordinate Providing Process.

The **Provide Native Data** function can not be defined generally for all providers. Each provider will carry out metadata collection and digitizing processes on their own with specific tools.

The **Coordinate Providing Process** function has to setup data augmentation information like mappings of raw data and services used to augment the digitized data while creating the SIP. When SIPs have been sent to Ingest, Administration may return Liens for these SIPs. These Liens are appealed by this function or a new ingest command will be sent to Coordinate SIP Upload to resend the SIP with new augmentation information. The final ingest report provided by Administration will be presented by this function as well.

The **Coordinate SIP Upload** function receives an ingest command and starts to request SIPs from the Create SIP function. These SIPs will be sent to Ingest and possible resubmit requests are handled. After all SIPs affected by the ingest command have been sent, a report for the Coordinate Providing Process will be created.

The **Create SIP** function starts by receiving a SIP request from the Coordinate SIP Upload function. Augmentation information defined within the Coordinate Providing Process are used to transform the raw data provided by the Provide Native Data function into a SIP. The raw data will be harmonization and augmented with services like OCR or vocabulary alignment.

5.1.3 Standards

The standards used within this component are based on vocabulary and metadata augmentation. Since this augmented information is used at the Portal, the standards will be used there as well.

5.1.3.1 Vocabulary Representation SKOS [W3C standard]

The Simple Knowledge Organisation Systems (SKOS)[6] is a family of formal languages designed for representation of thesauri, classification schemes, taxonomies, subject-heading systems, or any other type of structured controlled vocabulary. SKOS is built upon RDF[7] and RDFS[8], and its main objective is to enable easy publication of controlled structured vocabularies for the Semantic Web. SKOS is currently developed within the W3C framework.

It is used for export and import of controlled vocabulary within some thesaurus services used by the portal.

5.1.3.2 ISO 2788:1986 Guidelines for the Establishment and Development of Monolingual Thesauri [ISO standard][9]

An International Organization for Standardization (ISO) standard for the construction of monolingual thesauri. Includes guidelines for dealing with references, definitions, abbreviations, vocabulary control, indexing terms, compound terms, basic relationships, display, and management. Available for sale at www.iso.org

---

[6] http://www.w3.org/2004/02/skos/ [30.10.2009]

[7] http://www.w3.org/TR/rdf-concepts/ [30.10.2009]

[8] http://www.w3.org/TR/rdf-schema/ [30.10.2009]

[9] http://www.iso.org/iso/catalogue_detail.htm?csnumber=7776 [30.10.2009]

5.1.3.3 ANSI/NISO Z39.19-2005 Guidelines for the Construction, Format, and Management of Monolingual Controlled Vocabularies. [ANSI standard]

This American standard shows how to formulate descriptors, establish relationships among terms, and present the information in print and on a screen. Included are thesaurus maintenance procedures and recommended features for thesaurus management systems. This standard is available online.

5.1.3.4 ISO 5964 Guidelines for the Establishment and Development of Multilingual Thesauri [ISO standard][10]

An International Organization for Standardization (ISO) standard for the construction of multilingual thesauri. This standard is to be used as an extension of the monolingual guidelines (ISO 2788). Includes guidelines for dealing with degrees of term equivalence and non-equivalence, single-to-multiple term equivalence, etc. Also includes examples of thesaural displays. Print version available for sale.

5.1.4 Implementation and Best Practice

The best practice is aggregating various formats, harmonize them and put them into the SIP.

5.1.4.1 Extract-Transform-Load

ETL is the process of aggregating various databases into one target database. The extract step selects the relevant information from the current database. This information is transformed into the target format, which may also be XML for XML databases like eXist[11]. The load step puts the transformed data into the target database. For BHL-Europe, these steps are used within the PreIngest component to harmonize and augment the data at the archives. As there are multiple tools already available like Pentaho Data Integration[12] or XAware[13]. Some transformations can be made with these tools which lowers costs which would have been caused by creating proprietary tools.

5.1.4.2 Extensible Stylesheet Language Transformation (XSLT)

If XML is already available, transformation with standardized XSLT is recommended. As the SIP is based on standards like METS profiles and MARCXML, the XSLT is reusable.

---

[10] http://www.iso.org/iso/catalogue_detail.htm?csnumber=12159 [30.10.2009]

[11] http://exist.sourceforge.net/ [30.10.2009]

[12] http://www.pentaho.com/ [30.10.2009]

[13] http://www.xaware.org/ [30.10.2009]

5.1.4.3 DISMARC online mapping tool

Within the DISMARC EU-project (http://www.dismarc.eu) a proprietary metadata scheme was developed based on the Dublin Core Library Application Profile[14]. For this schema, an online mapping tool was created. The DISMARC schema of course is tailored for audio objects metadata and the mapping tool will not be suitable for BHL-Europe without amendments and adaptations to the BHL-Europe needs.

In general a mapping tool for the BHL-Europe archives needs to provide graphical tools for vocabulary integration, splitting and combining native records, cross referencing and value normalization. As the mapping should take place locally at the archives' sites, the future mapping tool has to be ported and adapted for BHL-Europe needs, supporting the agreed BHL-Europe metadata schema and to be used as a stand-alone version at the archives' premises .

5.2 OAIS_Ingest

This component provides the services and functions to accept Submission Information Packages (SIPs) from the PreIngest module or Producers directly (or from internal elements under Administration control) and prepare the contents for storage and management within the archive. Ingest functions include receiving SIPs, performing quality assurance on SIPs, generating an Archival Information Package (AIP) which complies with the archive's data formatting and documentation standards, extracting Descriptive Information from the AIPs for inclusion in the archive database, and coordinating updates to Archival Storage and Data Management.

5.2.1 Interfaces and Payload

- from PreIngest: Submission Information Package (SIP)
- to PreIngest: Confirmation, Resubmit Request
- to OAIS_DataManagement: Descriptive Information, Reports, Update Requests
- to OAIS_ArchivalStorage: Archival Information Package (AIP)
- to OAIS_Administration: Submission Information Package (SIP), Archival Information Package (AIP)

[14] http://dublincore.org/documents/library-application-profile/index.shtml [30.10.2009]

5.2.2 Processes

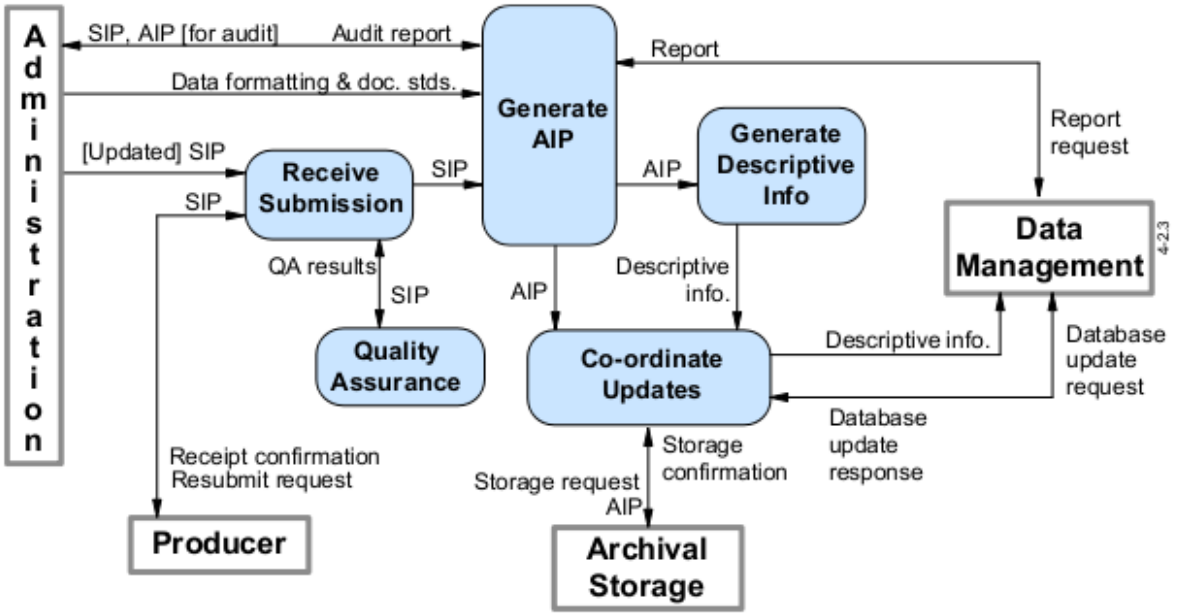


Figure 5-3 Ingest processes

The **Receive Submission** function provides the appropriate storage capability or devices to receive a SIP from the Producer (or from Administration). The SIPs are delivered via electronic transfer (e.g., FTP, REST Service, Cloud Storage Service). The Receive Submission function will check for access rights with the payload send within the SIP. This function provides a confirmation of receipt of a SIP to the Producer, which may include a request to resubmit a SIP in the case of errors resulting from the SIP submission.

The **Quality Assurance** function validates (QA results) the successful transfer of the SIP to the staging area. For digital submissions, these mechanisms include Cyclic Redundancy Checks (CRCs) or checksums associated with each data file as well as syntactical schema validation.

The **Generate AIP** function transforms SIPs into one or more AIPs that conform to the archive's data formatting and documentation standards. This involves file format conversions and file aggregation. The Generate AIP function may issue report requests to Data Management to obtain reports of information needed by the Generate AIP function to produce the Descriptive Information that completes the AIP. This function sends SIPs or AIPs for audit to the Audit Submission function in Administration, and receives back an audit report.

The **Generate Descriptive Information** function extracts Descriptive Information from the AIPs and collects Descriptive Information from other sources to provide to Coordinate Updates, and ultimately Data Management. This includes metadata to support searching and retrieving AIPs (e.g., who, what, when, where, why), and will also include special browse

products (thumbnails, images, document structures) populated by Access and used by the Portal.

The **Coordinate Updates** function is responsible for transferring the AIPs to Archival Storage and the Descriptive Information to Data Management. Transfer of the AIP includes a storage request which results in an electronic transfer to a cloud like storage infrastructure. After the transfer is completed and verified, Archival Storage returns a storage confirmation indicating the storage identification information for the AIP. The Coordinate Updates function also incorporates the storage identification information into the Descriptive Information for the AIP and transfers it to the Data Management entity along with a database update request. In return, Data Management provides a database update response indicating the status of the update. Data Management updates may take place without a corresponding Archival Storage transfer when the SIP contains Descriptive Information for an AIP already in Archival Storage.

5.2.3 Standards

As this component is an adapter and facade for other components, next to the payload and interfaces there are no standards available or they are part of other components.

5.2.3.1 SIP - METS Profile with MARCXML

The standards in this component are reduced to the ingest format which is based on a METS profile and the MARCXML metadata format. These XML schemata are very common in libraries. Please see the SIP for more information about these standards.

5.2.4 Best Practice and Implementation

Within this component, the Best Practice Implementation results from the Bibliothèque Nationale de France project in which ATOS was building an OAIS compliant system. The co-ordinate updates function has to run transactional which includes rollback possibilities coordinated by Ingest. Also, the one SIP equals one Item and therefore one transaction rule will derive from this project as bulk ingest procedures will be handled with multiple transactions.

5.2.4.1 EUROPEANA content checker (Quality Assurance)

This tool is part of the Quality Assurance function. It is developed by Europeana and will be adopted for BHL-Europe as well.

5.3 OAIS_ArchivalStorage

This component provides the services and functions for the storage, maintenance and retrieval of AIPs. Archival Storage functions include receiving AIPs from Ingest and adding them to permanent storage, managing the storage hierarchy, refreshing the media on which archive holdings are stored, performing routine and special error checking, providing disaster recovery capabilities, and providing AIPs to Access to fulfill orders.

5.3.1 Interfaces

- to OAIS_Ingest: Archival Information Package (AIP) via REST/XML
- to OAIS_Access: Archival Information Package (AIP) via REST/XML

5.3.2 Process

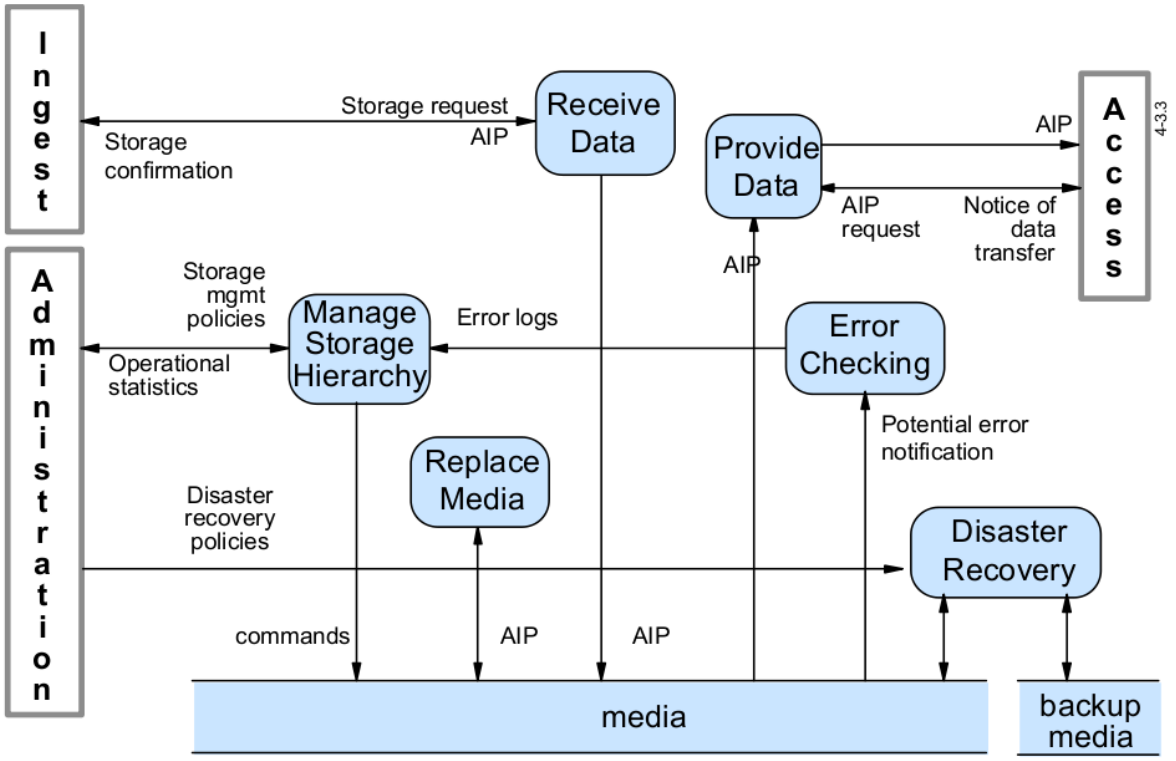


Figure 5-4 Archival Storage processes

The **Receive Data** function receives a *storage request* and an *AIP* from Ingest and moves the *AIP* to permanent storage within the archive. The transfer request may need to indicate the anticipated frequency of utilization of the data objects comprising the AIP in order to allow the appropriate storage devices or media to be selected for storing the AIP. This function will select the media type, prepare the devices or volumes, and perform the physical transfer to the

Archival Storage volumes. Upon completion of the transfer, this function sends a *storage confirmation* message to Ingest, including the storage identification of the AIPs.

The **Manage Storage Hierarchy** function positions, via *command*s, the contents of the AIPs on the appropriate media based on storage *management policie*s, operational statistics, or directions from Ingest via the storage request. It will also conform to any special levels of service required for the AIP, or any special security measures that are required, and ensures the appropriate level of protection for the AIP. These include on-line, off-line or near-line storage, required throughput rate, maximum allowed bit error rate, or special handling or backup procedures. It monitors *error logs* to ensure AIPs are not corrupted during transfers. This function also provides *operational statistics* to Administration summarizing the inventory of media on-hand, available storage capacity in the various tiers of the storage hierarchy, and usage statistics.

The **Replace Media** function provides the capability to reproduce the AIPs over time. Within the Replace Media function the Content Information and Preservation Description Information (PDI) must not be altered. However, the data constituting the Packaging Information may be changed as long as it continues to perform the same function and there is Implementation and

5.3.3 Standards

5.3.3.1 SWORD http://www.swordapp.org/

SWORD is a lightweight protocol for depositing content from one location to another. It stands for Simple Web-service Offering Repository Deposit and is a profile of the Atom Publishing Protocol (known as APP or ATOMPUB).

SWORD has been funded by the Joint Information Systems Committee to develop the SWORD profile and a number of demonstration implementations.

5.3.3.2 AtomPub http://www.ietf.org/rfc/rfc4287.txt

The Atom Publishing Protocol (AtomPub) is an application-level protocol for publishing and editing Web resources. The protocol is based on HTTP transfer of Atom-formatted representations. The Atom format is documented in the Atom Syndication Format.

5.3.4 Implementation and Best Practice

The **ingest module** deposits SIPs via the SWORD protocol. SWORD leverages the existing Atom Publishing Protocol (APP), "an application-level protocol for publishing and editing Web resources". APP is based on the HTTP transfer of Atom-formatted representations yet SWORD has focused on two key aspects of the protocol - the deposit of files, rather than Atom documents, and the extension mechanism for specifying additional deposit parameters.

Also worth noting is that SWORD does not specify the implementation of all of the functionality of APP, rather it supports deposit only. Implementations are free to support update and delete if they wish but this is out of the SWORD remit.

Depositing is a two-stage process within APP and SWORD. First, a request from an authenticated user is sent to the implementation for what APP calls the 'service document', this returns details of the collections that user is allowed to deposit to within the repository. At this point, the user may deposit their file into the chosen collection. Various things may prevent success, for example lack of authentication credentials, unacceptable file format or a corrupt MD5 checksum. The repository will send a respond indicating the success, or otherwise of the deposit.

SWORD detailed specifications: http://www.swordapp.org/docs/sword-profile-1.3.html
The SWORD interface and the Fedora interface can co-exist so we have the option of using both or eliminating the SWORD interface if it does not add enough value. The Fedora interface is the REST API.

The **Fedora Access** service defines an open interface for accessing digital objects. The access operations include methods to do reflection on a digital object (i.e., to discover the kinds of disseminations that are available on the object), and to request disseminations. The major function of the Fedora Access service is to fulfill a client's request for dissemination. To support disseminations, the underlying repository system must evaluate the services specified for a digital object, and figure out how to call it. The service may be internal to the repository, or it may be a web service external to the repository. The underlying repository system facilitates all external service bindings on behalf of the client, simply returning a dissemination result via the access service layer.

The Fedora REST API exposes a subset of the Fedora Access and Management APIs as a RESTful (Representational State Transfer) Web Service.
REST API detailed specifications: http://fedora-commons.org/confluence/display/FCR30/REST+API

5.3.4.1 SWORD (http://www.swordapp.org/)

SWORD is a lightweight protocol for depositing content from one location to another. It stands for Simple Web-service Offering Repository Deposit and is a profile of the Atom Publishing Protocol (known as APP or ATOMPUB).
It offers a standard deposit interface to repositories that will allow more services to be built which can offer functionality such as deposit from multiple locations, e.g. disparate

repositories, desktop drag'n'drop tools or from within standard office applications. SWORD can also facilitates deposit to multiple repositories, increasingly important for depositors who wish to deposit to funder, institutional or subject repositories. Other possibilities include migration of content between repositories, transfer to preservation services and many more. The implementation used is the sword plug-in for fedora. (http://www.fedora-commons.org/confluence/display/FCSVCS/SWORD-Fedora+1.2)

5.3.4.2 Fedora (http://www.fedora-commons.org)

Fedora (Flexible Extensible Digital Object Repository Architecture) is used to manage, preserve, and link the digital assets to be stored. Fedora is the core software component of the Archival Storage module. It also has subordinates such as a relational and an RDF databases.

Java SE Development Kit (JDK) 5 or higher.

Whether installing a binary or source distribution, JDK 5 or higher is required. The JDK should be installed on the machine you intend to use as the Fedora server. It is available from http://java.sun.com/.

Application Server

The fedora repository, the sword-fedora plug-in, as well as other fedora framework services such as the disseminators are all packaged as J2EE web archives (**WAR**). They need to be deployed in a J2EE servlet container or application server that implements Servlet 2.4/JSP 2.0 or higher such as Tomcat, Glassfish, Jetty, etc. The Fedora Repository installer includes Tomcat 5.5.26. however the production environment should have standalone application server in order to run multiple instances of any WARs for scalability purposes.

Database

Fedora uses a relational database to support some of its functions. To simplify installation, the Fedora installer includes and can configure an embedded instance of the Derby SQL Database 10.4.2 which should not be used for any production repository. Fedora supports three external databases: MySQL, Oracle and PostgreSQL. PostgreSQL will be used.

5.3.4.3 Akubra (http://www.fedora-commons.org/confluence/display/AKUBRA/Akubra+Project)

The Akubra Project is an effort to provide a pluggable file storage interface that can be adapted to almost any storage subsystem. Akubra supports both ordinary and transactional storage systems, but makes simplifying assumptions in order to achieve a high level of interoperability between storage systems:
In Akubra:

- a Blob is a finite-length bitstream with an identifier (a URI)

- a Blob Store is primarily concerned with providing read/write access to blobs.

Akubra will allow the file storage system to be more easily replaced if necessary.

5.3.4.4 Storage Service

The storage service based on DuraCloud, S3, etc. must be operational.

5.3.4.5 DuraCloud (http://duraspace.org/duracloud.html)

DuraCloud is software aimed at providing academic libraries, universities, and other cultural heritage organizations with perpetual access to their digital content. DuraCloud leaves the basics of pure storage to storage providers, while providing functionality that ensures long-term access and ease of use. The service replicates and distributes content across multiple cloud providers and enables the deployment of services to support:

- access
- preservation
- re-use
- sharing of cloud content

The use of the Akubra storage abstraction layer allows BHL-Europe to change storage providers that are compatible with the Akubra API.

5.4 OAIS_PreservationPlanning

This component provides the services and functions for monitoring the environment of the OAIS and providing recommendations to ensure that the information stored in the OAIS remains accessible to the Designated User Community over the long term, even if the original computing environment becomes obsolete. Preservation Planning functions include evaluating the contents of the archive and periodically recommending archival information updates to migrate current archive holdings, developing recommendations for archive standards and policies, and monitoring changes in the technology environment and in the Designated Community's service requirements and Knowledge Base. Preservation Planning also designs IP templates and provides design assistance and review to specialize these templates into SIPs and AIPs for specific submissions. . Preservation Planning also develops detailed Migration plans, software prototypes and test plans to enable implementation of Administration migration goals.

5.4.1 Interfaces and Payload

- to OAIS_Administration: via Reports
- to PreIngest: via surveys

- to Portal: via surveys

5.4.2 Process

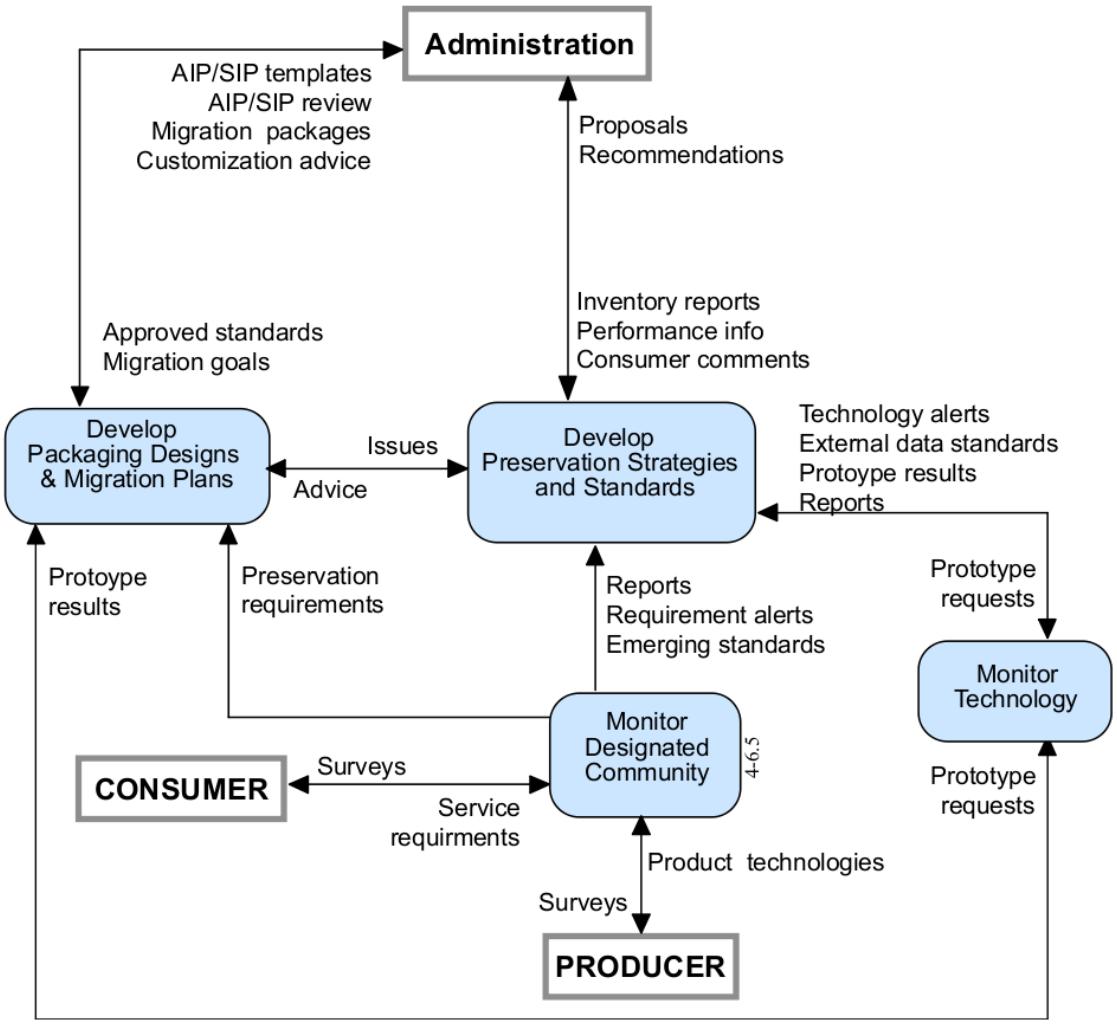


Figure 5-5 Preservation Planning processes

The **Monitor Designated Community** function interacts with archive Consumers and Producers to track changes in their service requirements and available product technologies. Such requirements might include data formats, media choices, preferences for software packages, new computing platforms, and mechanisms for communicating with the archive. This function may be accomplished via surveys, via a periodic formal review process, via community workshops where feedback is solicited or by individual interactions. It provides reports, requirements alerts and emerging standards to the Develop Preservation Strategies and Standards function. It sends preservation requirements to Develop Packaging Designs.

The **Monitor Technology** function is responsible for tracking emerging digital technologies, information standards and computing platforms (i.e., hardware and software) to identify technologies which could cause obsolescence in the archive's computing environment and

prevent access to some of the archives current holdings. This function may contain a prototyping capability for better evaluation of emerging technologies and receive prototype requests from Develop Preservation Strategies and Standards and from Develop Package Designs and Migration Plans. This function sends reports, external data standards, prototype results and technology alerts to Develop Preservation Strategies and Standards. It also sends prototype results to Develop Package Designs and Migration Plans.

The **Develop Preservation Strategies and Standards** function is responsible for developing and recommending strategies and standards to enable the archive to better anticipate future changes in the Designated Community service requirements or technology trends that would require migration of some current archive holdings or new submissions. This function receives reports from the Monitor Designated Communities and Monitor Technology functions, and it receives performance information, inventory reports and summarized consumer comments from Administration. This function sends recommendations on system evolution to Administration. This function also receives external data standards from Monitor Technology and produces profiles of those standards that are sent to Administration as proposals on their potential usage. This function also receives issues from Develop Packaging Designs and Migration Plans in the case of unanticipated submission requirements, and responds with advice to handle the new requirements.

The **Develop Packaging Designs and Migration Plans** function develops new IP designs and detailed migration plans and prototypes, to implement Administration policies and directives. This activity also provides advice on the application of these IP designs and Migration plans to specific archive holdings and submissions. This function receives archive approved standards and migration goals from Administration. The standards include format standards, metadata standards and documentation standards. It applies these standards to preservation requirements and provides AIP and SIP template designs to Administration. This function also provides customization advice and AIP/SIP review to Administration on the application of those designs. If this function encounters submissions that are not covered by existing standards and procedures, it can send issues to Develop Preservation Strategies and Standards and receive advice, including new standards, to assist in meeting the new submission requirements.

The migration goals received by this function tend to involve transformations of the AIP, including transformations of the Content Information to avoid loss of access due to technology obsolescence. The response to the migration goals may involve the development of new AIP designs, prototype software, test plans, community review plans and implementation plans for phasing in the new AIPs. This process may call on expertise or resources from other functions within Preservation Planning, such as prototype development from the Monitor Technology. This effort also will require consultation from the other functional areas and from the Designated Community. Once the migration plan, associated

AIP designs, and software have been tested and approved, this function will send the entire migration package to Administration, which will schedule and perform the actual migration.

5.5 OAIS_DataManagement

Data Management provides services and functions for populating, maintaining, and accessing Descriptive Information which identifies and documents archive holdings and administrative data used to manage the archive.
Data management revolves around use of a graph describing data collections and objects. This graph is usually known as a data graph. It is oriented and non-cyclical. The relations associating a node with its descending nodes are inheritance and composition relations. A data set, also known as a terminal collection, thus inherits the characteristics of all the collections above it. The graphs contain either explicit metadata or references to external files or documents.
Data Management functions include:

- catalogues and inventories on what may be retrieved from Archival Storage,
- administering the archive database functions (maintaining schema and view definitions, and referential integrity),
- performing database updates (loading new descriptive information or archive administrative data),
- performing queries on the data management data to generate result sets,
- producing reports from these result sets,
- processing algorithms that may be run on retrieved data,
- consumer access statistics,
- consumer billing,
- event Based Orders,
- security controls,
- OAIS schedules, policies, and procedures.

For BHL-Europe, the catalogue schema is derived from the fields and semantics of the BHL Deduplication Subset (BDS), which conforms to core Dublin Core fields. BDS was designed as a means to facilitate deduplication, to support fast table views on fields most likely to be used, and for indexing. BDS Fields serve as common denominators, their fields types are known, and they can be indexed generically using an Open Source Indexer like Apache Lucene.

5.5.1 Functions of Data Management

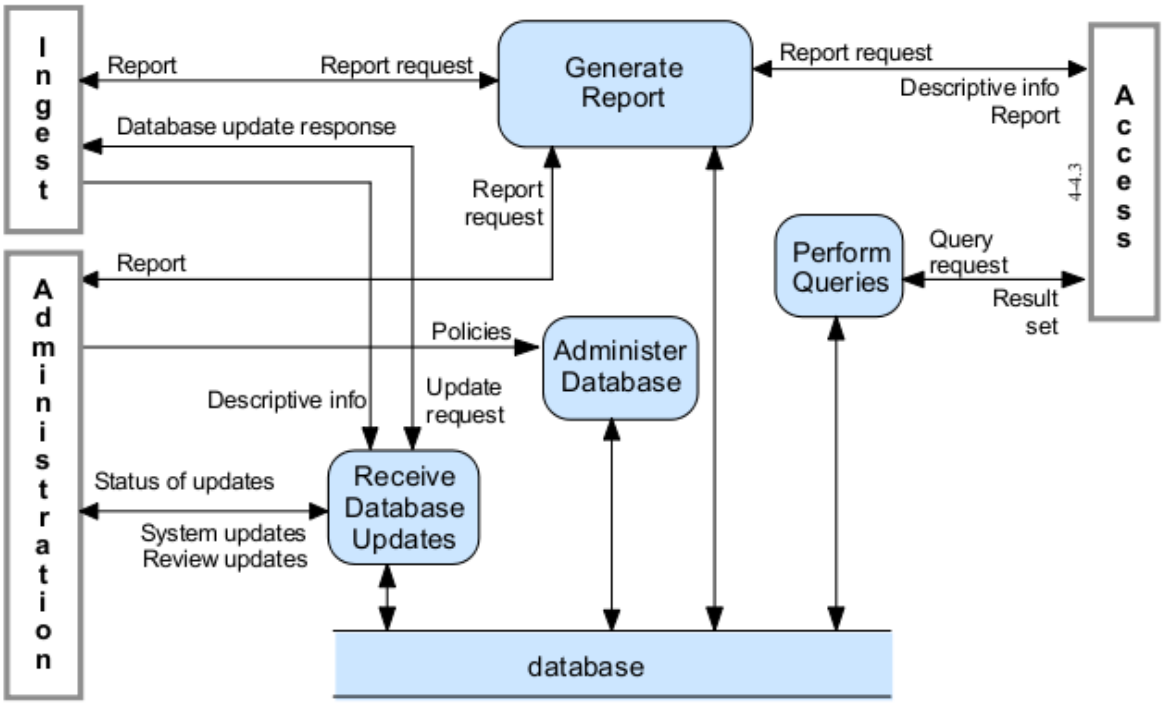


Figure 5-6: Data Management as part of the OAIS model.

For BHL-Europe the Interfaces are being implemented using the REST architectural style, data is encoded as XML. Different versions of interfaces implementations may co-exist.

The Administer Database function is responsible for maintaining the integrity of the Data Management database, which contains both Descriptive Information and system information. Descriptive Information identifies and describes the archive holdings, and system information is used to support archive operations. The Administer Database function is responsible for creating any schema or table definitions required to support Data Management functions; for providing the capability to create, maintain and access customized user views of the contents of this storage; and for providing internal validation (e.g., referential integrity) of the contents of the database. The Administer Database function is carried out in accordance with policies received from Administration.
The Perform Queries function receives a query request from Access and executes the query to generate a result set that is transmitted to the requester.
OAIS distinguishes three categories of Consumer requests:

- Query Requests, which are executed in Data Management and return immediate result sets for presentation to the user. The result sets always contain record-wise one globally unique ID (**GUID**) to identify the records and their related archival contents properly. Records in these result sets may also contain any number of BDS fields.
- Report Requests, which may require a number of queries and produce reports for delivery to the Consumer; and orders, which may access either or both Data

Management and Archival Storage to prepare a formal Dissemination Information Package (DIP) for on- or off-line delivery. An order may be an Ad hoc Order that is executed only once, or an Event Based Order that will be maintained by the Activate Requests function in Administration, and initiated by a dissemination request that may result in periodic deliveries of requested items.

- The Archival Information Update function in Administration also submits Dissemination Requests to obtain DIPs needed to perform its update functions. Other special request types are allowed, but are not detailed. This function will determine if resources are available to perform a request, assure that the user is authorized to access and receive the requested items, and notify the Consumer that a request has been accepted or rejected (possibly with an estimate of request cost and an option to cancel the request). It will then transfer the request to Data Management or to the Generate DIP function for execution.

The Generate Report function receives a report request from Ingest, Access or Administration and executes any queries or other processes necessary to generate the report that it supplies to the requester. Typical reports might include summaries of archive holdings by category, or usage statistics for accesses to archive holdings. It may also receive a report request from Access and provides descriptive information for a specific AIP. An example for a report is the Deduplication Report.
OAIS itself does not cover deduplication but rather assumes that records are deduplicated before ingesting. As for the BHL-Europe archive deduplication will on the long run become very important for the usability of the system, BHL-Europe must address the issue. One option to handle this is by rejecting ingest if duplicates are found. The Deduplication Report would inform the provider which records have been identified as duplicates.

The Receive Database Updates function adds, modifies or deletes information in the Data Management persistent storage. The main sources of updates are Ingest, which provides Descriptive Information for the new AIPs, and Administration, which provides system updates and review updates. Ingest transactions consist of Descriptive Information which identifies new AIPs stored in the archive. Updating the database always includes that the index (Apache Lucene) is being updated.
System updates include all system-related information (operational statistics, Consumer information, and request status). Review updates are generated by periodic reviewing and updating of information values (e.g., contact names, and addresses).
The Receive Database Updates function provides regular reports to Administration summarizing the status of updates to the database, and also sends a database update response to Ingest.

The Generate AIP function transforms one or more SIPs into one or more AIPs that conform to the archive’s data formatting and documentation standards. This may involve file format

conversions, data representation conversions or reorganization of the content information in the SIPs. The Generate AIP function may issue report requests to Data Management to obtain reports of information needed by the Generate AIP function to produce the Descriptive Information that completes the AIP. This function sends SIPs or AIPs for audit to the Audit Submission function in Administration, and receives back an audit report. Fields used for search and retrieval are stored in the Apache Lucene Index.

The Generate Descriptive Information function extracts Descriptive Information from the AIPs and collects Descriptive Information from other sources to provide to Coordinate Updates, and ultimately Data Management. This includes metadata to support searching and retrieving AIPs (e.g., who, what, when, where), and could also include special browse products (thumbnails, images) to be used by Finding Aids.

The Coordinate Updates function is responsible for transferring the AIPs to Archival Storage and the Descriptive Information to Data Management. Transfer of the AIP includes a storage request and may represent an electronic, physical, or a virtual (i.e., data stays in place) transfer. After the transfer is completed and verified, Archival Storage returns a storage confirmation indicating (or verifying) the storage identification information for the AIP. The Coordinate Updates function also incorporates the storage identification information into the Descriptive Information for the AIP and transfers it to the Data Management entity along with a database update request. In return, Data Management provides a database update response indicating the status of the update. Data Management updates may take place without a corresponding Archival Storage transfer when the SIP contains Descriptive Information for an AIP already in Archival Storage.

OAIS is a high level model of pervasive computing and describes how constant flows of data can be ingested into an archive. It does not discuss in detail how a system is to be designed that would comply to OAIS, and is not meant as an implementation blueprint. Real world problems like transactional isolation levels and two phase commits in an distributed system that implements an OAIS system are not in the scope of the OAIS specification. However, BHL-Europe must take care that data loss cannot occur at any point during data transfer from the provider library through the ingestion of the data, and during the update database and update indexes process.

5.5.2 Data Management Information

While Data Objects generally (in OAIS) refer to either a Physical Object or a Digital Object, Data Management Data specifically is data that is created and stored in Data Management persistent storage and refers to operation of an archive. OAIS suggests a hierarchy of business objects for organizing the Data Management Data.

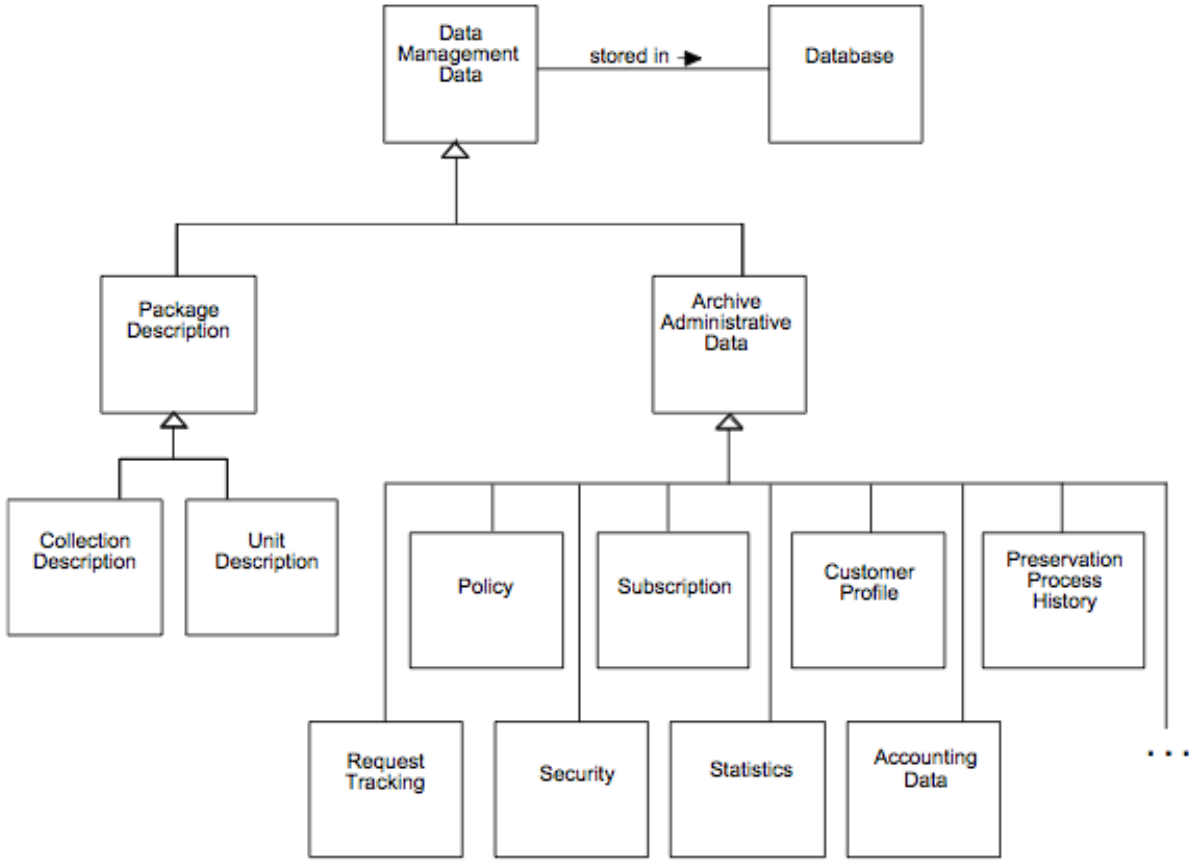


Figure 5-7: illustrates a hierarchy of classes of 'data management information'.

These classes exemplify the data that may be required for archive administration in an OAIS.
Since these classes are conceptual and individual OAIS implementations may vary significantly BHL-Europe will not necessarily implement and use all of these types.
For example, it is currently rather unlikely that the BHL-Europe repository needs to store business specific data.

Package Descriptions often are stored in persistent storage such as database management systems to enable easy, flexible access and update to the contained Associated Descriptions. In addition to the Package Descriptions all the information needed for the operation of an archive could be stored in databases as persistent data classes. Any other appropriate storage may be used instead of an relational database, for example an hierarchical repository as described by JSR 170 (a specification developed under the Java Community Process). The Archive Administration Information represents the entire range of information required for the day-to-day operation of the archive.
This information includes:

- Policy information which provides pricing information and availability constraints for ordering archived information.

- Request tracking information that records the progress of each user transaction with an archive. The request tracking process can be very complicated, involving database events and triggers, or as simple as a flat file tracking Order Requests.
- Security information that includes user names and any passwords or other mechanisms needed to authenticate the identity and privileges of archive users.
- Event Based Order information that provides the information needed to support repeating or future requests.
- Statistical information needed by archive administration and Management to determine future policies and performance tuning for more effective archive operation. Examples of these statistics include the number of times an AIP was ordered over a time period and the average time between receiving an order request and shipping the requested holding
- Preservation process history information that tracks the migrations of AIPs, including media replacements and AIP transformations
- Customer profile information that enables the archive to maintain facts such as user name and address to avoid the user having to re-enter these facts each time he enters a request.
- Accounting information that includes the data necessary for the operation of the archive as a business. The accounting data include payroll data, accounts payable data and accounts receivable data.

Data Management Data is used by Data Management, and is moved around the OAIS during communication with other OAIS modules.

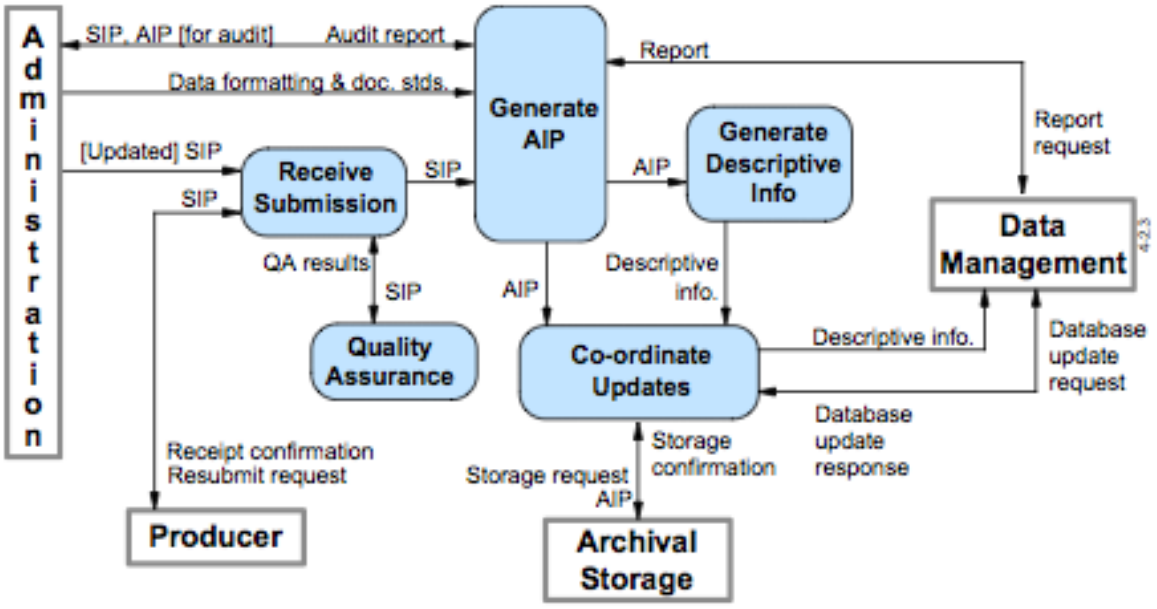


Figure 5-8: illustrates data objects and where they are used within the OAIS.

While Fig. 5-2 describes Data Management Data as business objects, Fig. 5-8 allocates Data Management Data on another level of abstraction as technical objects that are moved around between OAIS modules:

Descriptive Information: The set of information, consisting primarily of Package Descriptions, which is provided to Data Management to support the finding, ordering, and retrieving of OAIS information holdings by Consumers.

Result Set: The set of descriptive records for those AIPs in an OAIS which match the criteria stated in a Consumer query, or other results from a search on Data Management.

Query: The set of criteria encoded in a formal language that form a query. Queries a sent to the Data Management, which evaluates the query.

Report Request: A Request for a report, includes a Query that defines matching criteria for the data the report analyses.

Query Request: The set of descriptive records for those AIPs in an OAIS which match the criteria stated in a Consumer Query, or other results from a search on Data Management.

Report: A formally encoded set of information that corresponds to a query and can be used to render a report.

5.5.3 Standards, Implementation and Best Practice

BHL-Europe is using a subset of core Dublin Core (DC) to define the catalogue schema. Dublin Core is a standard defined by the Dublin Core Metadata Initiative. DC covers a set of 15 metadata field which has been standardized as ISO Standard 15836:2009.

Data management follows OAIS recommendations. The use of data packages named DIP, SIP, and AIP for data transmission complies to the OAIS model. The METS file format is used for storing structural information, metadata, page level image and text data. METS is a Standard maintained by the Library of Congress.

The BHL-Europe Archive System implements service endpoints in a way that complies to the architectural style called Representational State Transfer (REST). REST was first described in Roy Thomas Fielding: Architectural Styles and the Design of Network-based Software Architectures (Dissertation from 2000), and it is widely accepted as a means to implement HTTP based applications that exchange data. The BHL-Europe service components will use REST with data encoded as XML. XML is a standard specified by the W3C consortium.

BHL-Europe will use an Open Source Indexer called Apache Lucene. Lucene is widely accepted, and has been successfully used in numerous web based applications. The success of

Lucene has encouraged tools like SolR that add value to the Lucene framework. Using Lucene is best practice.

Data management will store data in a standard relational database, or possibly in a hierarchical JSR 170 repository. Both techniques are best practise.

5.6 OAIS_Access

This component provides the services and functions that support Consumers in determining the existence, description, location and availability of information stored in the OAIS, and allowing Consumers to request and receive information products. Access functions include communicating with Consumers to receive requests, applying controls to limit access to specially protected information, coordinating the execution of requests to successful completion, generating responses (Dissemination Information Packages, result sets, reports) and delivering the responses to Consumers.

5.6.1 Interfaces

- to OAIS_DataManagement: via REST/XML
- to OAIS_ArchivalStorage: Archival Information Package (AIP) via REST/XML
- to Portal: Dissemination Information Package (DIP)
- to OAIS_Administration: Dissemination Information Package (DIP)

5.6.2 Processes

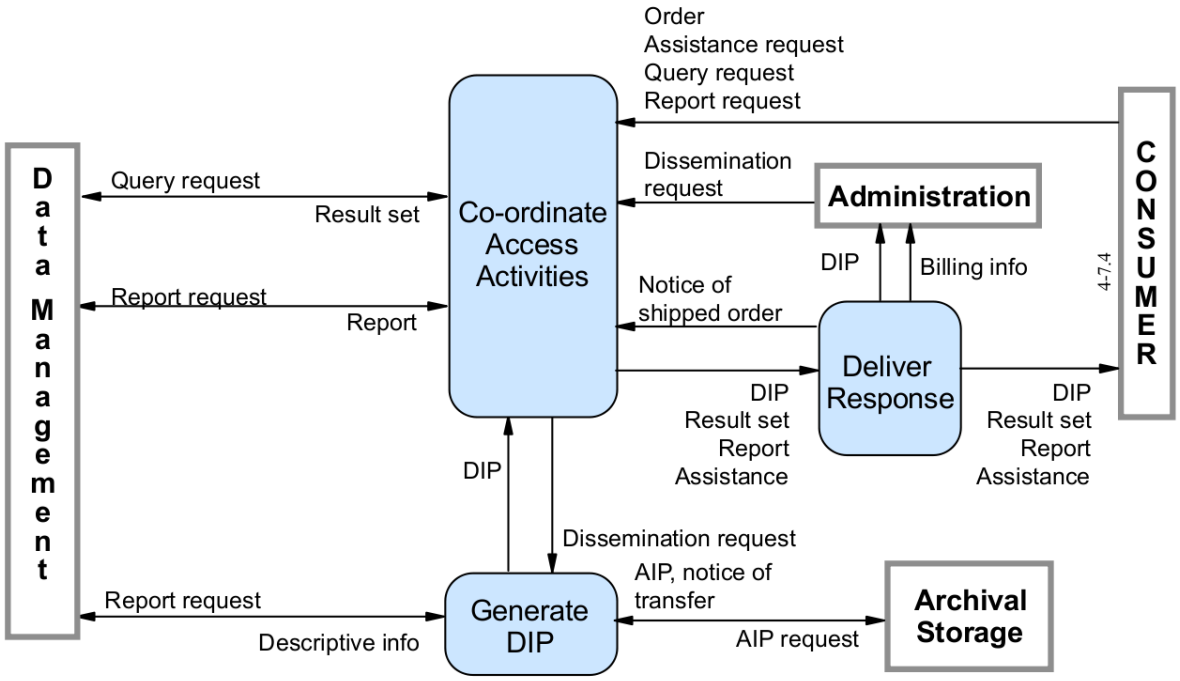


Figure 5-9 Access processes

The **Coordinate Access Activities** function provides a single user interface to the information holdings of the archive. This interface will be implemented as REST and SOAP services and the payload will be encoded in XML and JSON. Three categories of Consumer requests are distinguished:

- query requests, which are executed in Data Management and return immediate result sets for presentation to the user;
- report requests, which may require a number of queries and produce formatted reports for delivery to the Consumer;
- and orders, which may access either or both Data Management and Archival Storage to prepare a formal Dissemination Information Package (DIP) for on- or off-line delivery. An order may be an Adhoc Order that is executed only once, or an Event Based Order that will be maintained by the Activate Requests function in Administration, and initiated by a dissemination request that may result in periodic deliveries of requested items. The Archival Information Update function in Administration also submits dissemination requests to obtain DIPs needed to perform its update functions.
- Other special request types are listed below.

As naming convention in BHL-Europe, DIP is used for all responses and will be prefixed by its purpose.
This function will determine if resources are available to perform a request, assure that the user is authorized to access and receive the requested items, and notify the Consumer that a request has been accepted or rejected (possibly with an estimate of request cost and an option to cancel the request). It will then transfer the request to Data Management or to the Generate DIP function for execution. This function also provides assistance to OAIS Consumers including providing status of orders and other Consumer support activities in response to an assistance request.
The **Generate DIP** function accepts a dissemination request, retrieves the AIP from Archival Storage, and moves a copy of the data to a staging area and caches these packages for further processing. This function also transmits a report request to Data Management to obtain Descriptive Information needed for the DIP. If special processing is required, the Generate DIP function accesses data objects in staging storage and applies the requested processes. The types of operations, which may be carried out, include statistical functions, sub-sampling in temporal or spatial dimensions, conversions between different data types or output formats, and other specialized processing (e.g., image processing). This function places the completed DIP response in the staging area and notifies the Coordinate Access Activities function that the DIP is ready for delivery.
The **Deliver Response** function handles deliveries of responses (DIPs, result sets, reports and assistance) to Consumers. The on-line delivery accepts a response from Coordinate Access Activities and prepares it for on-line distribution in real time via communication links. It identifies the intended recipient, determines the transmission procedure requested, places the response in the staging area to be transmitted, and supports the on-line transmission of the response.

5.6.3 Standards

Access will provide a range of request types and DIPs for the consumer. As the consumer varies, standards have will be used to supply information to the consumer interfaces. Next to the official standards, community based *de facto* standards will be provided as well.

5.6.3.1 OAI-PMH [standard]

Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH)[15] is a protocol developed by the Open Archive Initiative. It is used to provide metadata description of items which will later be harvested through the simple XML over HTTP interface. The metadata will be presented in Dublin Core and additionally defined formats. For more formats see the

[15] http://www.openarchives.org/pmh/ [30.10.2009]

DIP's standards. Search functionality provided by this interface reduces to collections, time frames and formats. This interface will be used by Europeana[16] to get all records in their desired format.

5.6.3.2 SRU/SRW [standard]

Search/Retrieve via URL (SRU)[17] is a standard for libraries created within the Z39.50 International Next Generation (**ZING**). It is a modern equivalent to the Z39.50 protocol based on HTTP and XML. Therefore it is a candidate for cloud services. Search/Retrieve via Web (**SRW**) additionally provides a SOAP Interface. Both standards use CQL as query language.

5.6.3.3 CQL [standard]

The Contextual Query Language (CQL)[18] is a formal language for representing queries to information retrieval systems such as web indexes, bibliographic catalogues and museum collection information. The design objective is that queries be human readable and writable, and that the language be intuitive while maintaining the expressiveness of more complex languages.

Contextual Queries are built using an Index, a Relation and a Search Term. These elements can be used multiple times connected with Boolean operators and embraced by parentheses. Due to Relation Modifiers a wide range of queries can be built covering proximity searches as well as relevance searches.

The query commands are case insensitive. An example query might look like

```
dc.title any fish or (dc.creator any sanderson and dc.identifier =
"id:1234567")
```

5.6.3.4 RSS/ATOM feeds [standard]

Web feeds like Really Simple Syndication (RSS)[19] and Atom Syndication Format (ATOM)[20] are data formats used to provide information about frequently updated content. For BHL-Europe, new items added or changed within the BHL network will be placed onto the feed. The feed will be customizable via GET parameters in CQL/OpenSearch or the BHL Europe Data API and will aggregate the items suitable for the query.

---

[16] http://www.europeana.eu/portal/ [30.10.2009]

[17] http://www.loc.gov/standards/sru/specs/search-retrieve.html [30.10.2009]

[18] http://www.loc.gov/standards/sru/specs/cql.html [30.10.2009]

[19] http://en.wikipedia.org/wiki/RSS [30.10.2009]

[20] http://en.wikipedia.org/wiki/ATOM [30.10.2009]

5.6.3.5 OpenSearch [standard]

OpenSearch[21] is a collection of simple formats for the sharing of search results. The OpenSearch formats helps people to discover and use the search engine and to syndicate search results across the web. Search engines that crawl the surface of the web are picking up only a small fraction of the great content that is available within BHL-Europe. OpenSearch provides a deeper insight on the BHL-Europe content for crawlers like Google and Yahoo. OpenSearch was created by Amazon.

5.6.3.6 OpenUrl [standard]

The OpenURL standard (Z39.88)[22] is designed to support mediated linking from information resources, such as abstracting and indexing databases (sources), to library services (targets), such as academic journals, whether online or in printed or other formats. A "link resolver", or "link-server", parses the elements of an OpenURL and provides links to appropriate targets available through a library. A source is generally a bibliographic citation or bibliographic record used to generate an OpenURL. A target is a resource or service that helps satisfy user's information needs. Examples of targets include full-text repositories, online journals, online library catalogues and other Web resources and services. A source is typically a database that indexes various types of information resources often found in libraries, such as articles, books, patents, etc. Some examples of such databases include Web of Science, Scifinder, Modern Languages Association Bibliography and Google Scholar.

5.6.3.7 BHLE Data API [BHL-Europe Development]

This interface will provide search and retrieval functionality especially designed for the BHL-Europe Portal component. It will just serve specific items needed for high performance portal usage without transformation processes at Portal side.

5.6.3.8 Google Booksearch Data API [proprietary]

The Google Booksearch API[23] may be an additional option for Google My Library users. This interface is based on AtomPub and JSON and can process various queries, authentication, and batch requests. Therefore, this interface my be moved to Portal and will be implemented there for better user experience as users are not stored in Access. Base functionality may still be provided by Access.

---

[21] http://www.opensearch.org/Home [30.10.2009]

[22] http://www.niso.org/kst/reports/standards?step=2&gid=None&project_key=d5320409c5160be4697dc046613f71b9a773cd9e [30.10.2009]

[23] http://code.google.com/intl/de-DE/apis/books/docs/gdata/developers_guide_protocol.html [30.10.2009]

5.6.3.9 Amazon SOAP [proprietary]

As Amazon is the leading book selling institution, their interface is well known. BHL-Europe will implement the same interface for a broad user community.

5.6.4 Implementation and Best Practice

Access is used for authenticated access to DataManagement and ArchivalStorage via a common interface. Access will provide a list of search interfaces based on different standards described above. Most of these standards will provide the DIPs in various formats especially on metadata level. Internal transformations in combination with caching will be used to provide a fast interface for consumers like the portal.

5.7 OAIS_Administration

This component provides the services and functions for the overall operation of the archive system. Administration functions include soliciting and negotiating submission agreements with Producers, auditing submissions to ensure that they meet archive standards, and maintaining configuration management of system hardware and software. It also provides system engineering functions to monitor and improve archive operations, and to inventory, report on, and migrate/update the contents of the archive. It is also responsible for establishing and maintaining archive standards and policies, providing customer support, and activating stored requests.

5.7.1 Interfaces and Payload

- to PreIngest: Lien/Appeal, Final Ingest report. submission/schedule agreement
- to OAIS_Ingest [updated] Submission Information Package (SIP), standards for SIP
- to OAIS_DataManagement: Policies, updates, reports
- to OAIS_ArchivalStorage: Policies, statistics
- to OAIS_PreservationPlanning: Migration Goals, consumer comments, performance info
- to OAIS_Access: Dissemination Information Package (DIP), billing info
- to Portal: via customer services payment and bills

5.7.2 Process

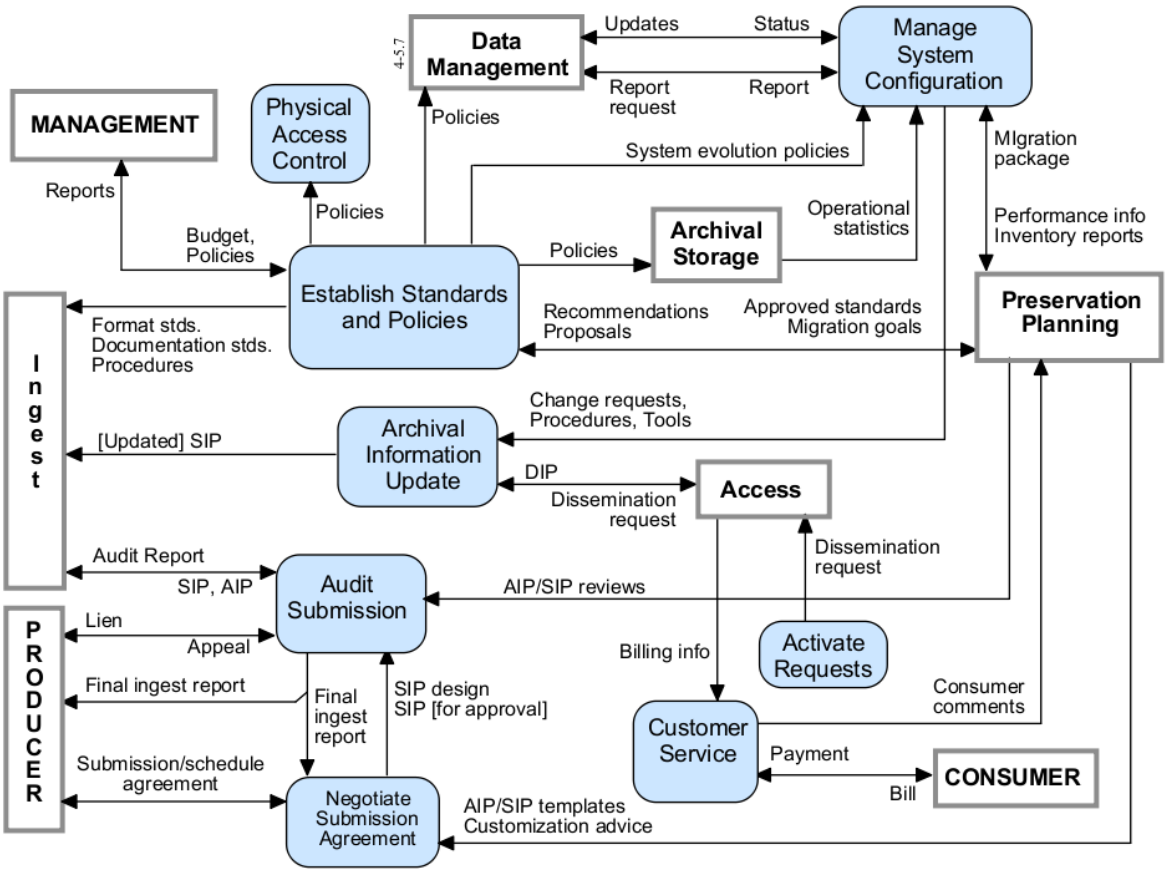


Figure 5-10 Administration processes

The **Negotiate Submission Agreement** function solicits desirable archival information for the OAIS and negotiates Submission Agreements with Producers. This function also negotiates a data submission schedule with the Producer. It maintains a calendar of expected Data Submission Sessions that will be needed to transfer one or more SIPs to the OAIS and the resource requirements to support their ingestion. This function receives AIP/SIP templates and customization advise from Preservation Planning and sends SIP designs and SIPs to the Audit Submission function as part of the submission approval process. The data submission formats and procedures must be clearly documented in the archive's data submission policies, and the deliverables must be identified by the Producer in the Submission Agreement.

The **Manage System Configuration** function provides system engineering for the archive system to continuously monitor the functionality of the entire archive system and systematically control changes to the configuration. This function maintains integrity and tractability of the configuration during all phases of the system life cycle. It also audits system operations, system performance, and system usage. It sends report requests for system information to Data Management and receives reports; it receives operational statistics from Archival Storage. It summarizes those reports and periodically provides OAIS performance

information and archive holding inventory reports to Preservation Planning. It sends performance information to Establish Standards and Policies. It receives migration packages from Preservation Planning. It receives system evolution policies from the Establish Standards and Procedures function. Based on these inputs it develops and implements plans for system evolution. It sends change requests, procedures and tools to Archive Information Update.

The **Archival Information Update** function provides a mechanism for updating the contents of the archive. It receives change requests, procedures and tools from Manage System Configuration. It provides updates by sending a dissemination request to Access, updating the contents of the resulting DIPs and resubmitting them as SIPs to Ingest.

The **Physical Access Control** function provides mechanisms to restrict or allow physical access (doors, locks, guards) to elements of the archive, as determined by archive policies.

The **Establish Standards and Policies** function is responsible for establishing and maintaining the archive system standards and policies. It receives budget information and policies such as the OAIS charter, scope, resource utilization guidelines, and pricing policies from Management. It provides Management with periodic reports. It receives recommendations for archive system enhancement, and proposals for new archive data standards from Preservation Planning. It also receives performance information and archive holding inventories from Manage System Configuration. Based on these inputs, archive standards and policies are established and sent to other Administration functions and the other Functional Entities for implementation. The standards include format standards, documentation standards and the procedures to be followed during the Ingest process. It provides approved standards and migration goals to Preservation Planning. This function will also develop storage management policies (for the Archival Storage hierarchy), including migration policies to assure that archive storage formats do not become obsolete, and database administration policies. It will develop disaster recovery policies. It will also determine security policies for the contents of the archive, including those affecting Physical Access Control and the application of error control techniques throughout the archive.

The **Audit Submission** function will verify that submissions (SIP or AIP) meet the specifications of the Submission Agreement. This function receives AIP/SIP reviews from Preservation Planning and may also involve an outside committee (e.g., science and technical review). The audit process must verify that the quality of the data meets the requirements of the archive and the review committee. It must verify that there is adequate Representation Information and PDI to ensure that the Content Information is understandable and independently usable to the Designated Community. The formality of the review will vary depending on internal archive policies. The Audit process may determine that some portions of the SIP are not appropriate for inclusion in the archive and must be resubmitted or excluded. An audit report is provided to Ingest. After the audit process is completed, any liens are reported to the Producer, who will then resubmit the SIP to Ingest or appeal the decision

to Administration. After the audit is completed, a final ingest report is prepared and provided to the Producer and to Negotiate Submission Agreement. Audit methods potentially include sampling, periodic review, and peer review.

The **Activate Requests** function maintains a record of event-driven requests and periodically compares it to the contents of the archive to determine if all needed data is available. If needed data is available, this function generates a dissemination request that is sent to Access. This function can also generate orders on a periodic basis where the length of the period is defined by the Consumers or on the occurrence of an event (e.g., a database update).

The **Customer Service** function will create, maintain and delete Consumer accounts. It will collect billing information from Access and will send bills and collect payment from Consumers for the utilization of archive system resources. It will respond to general information requests. This function will also collect and respond to feedback on Access services and products. Customer Service will summarize these comments and make them available.

5.8 Portal

This component represents the OAIS consumer. It will help the end user to determine the existence, description, location and availability of items. The portal provides visual aids for browsing the catalogue as well as multilingual search tools to retrieve items on various levels in various languages. In addition to the search and retrieve functionality of DIPs, the portal has to implement surveys and payment features. This derives from the OAIS consumer entity. End users will provide service requirements to OAIS preservation planning with the help of surveys. Payment and billing will be handled by OAIS administration component.

5.8.1 Interfaces

- from OAIS_Access: Dissemination Information Package (DIP)
- to OAIS_PreservationPlanning: Surveys
- to OAIS_Administration: Bills and Payments

5.8.2 Processes

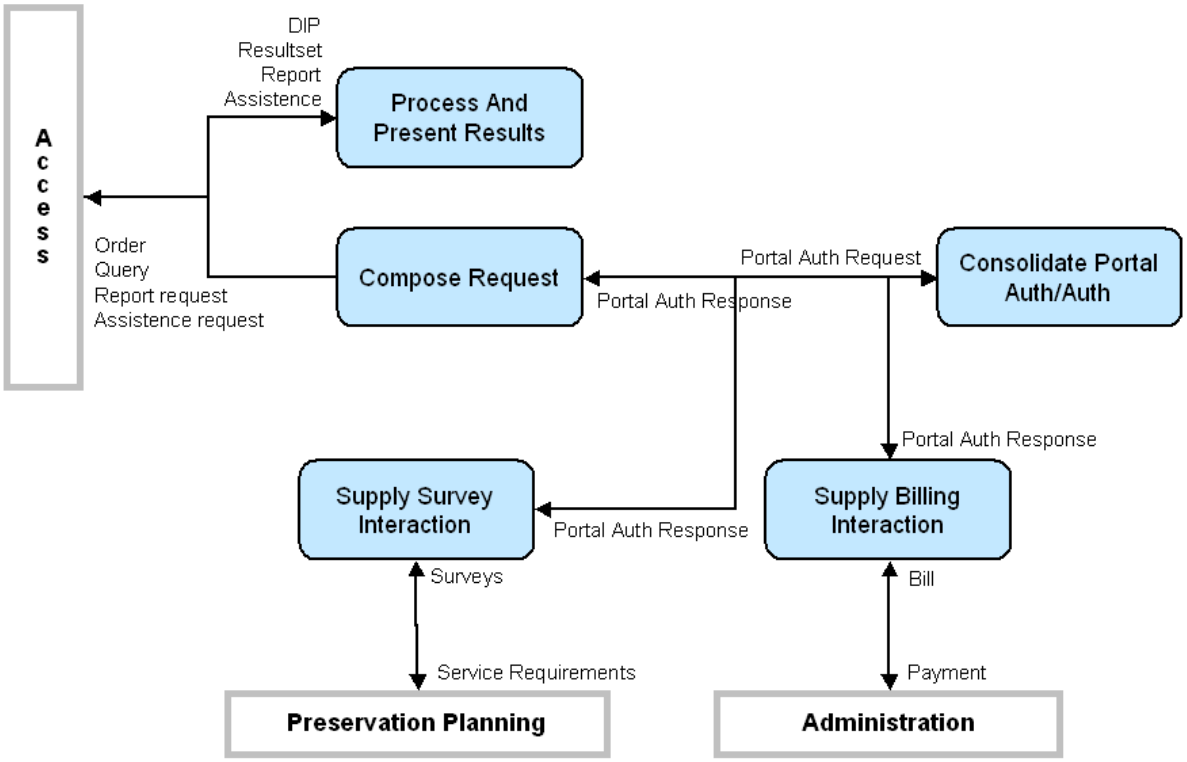


Figure 5-11 Portal processes

The **Compose Requests** function maps user requests to the appropriate Access interface to get the desired Dissemination Information Package (DIP). The portal users will use tools to build their query. These tools will be supplied by this function. Vocabulary services and multilingual search features will enhance the full-text and metadata search.
Within the **Process and Present Results** function, various result sets provided by Access will be prepared for further user interaction. The results may be presented in categories and divided by facets. Additional visualizations for available digitized data are provided by this function as well. The prepared result set can be used to compose new requests if desired.
To improve the user experience, the **Supply Survey Interaction** is implemented by this component. The end-user can use these functions to provide additional service requirements to the provider and the portal.

The **Supply Billing Interaction** functions lets the end-user purchase items from the OAIS. Within BHL-Europe, all items are within public domain and no payment will take place.

As an Auth/Auth convenience, the portal will use the **Consolidate Portal Auth/Auth** functions to provide information about security required by the other OAIS components. Not all users registered at the portal need to be known by the other components which reduces administrative overhead.

5.8.3 Standards

5.8.3.1 Portlet Specification JSR-168 and JSR-286 [Java standard]

As described in the implementation section of this component, the portal will be built on the JSR-168[24] and JSR-286[25] portlet specification. A portal in this scope, is a web based application that provides personalization, authentication, content aggregation from different sources and hosts the presentation layer of information systems. A portlet is an application that provides a specific piece of content (information or service) to be included as part of a portal page. It is managed by a portlet container, that processes requests and generates dynamic content. Portlets are used by portals as pluggable user interface components that provide a presentation layer to information systems.

5.8.3.2 Vocabulary Representation SKOS [W3C standard]

The Simple Knowledge Organisation Systems (SKOS)[26] is a family of formal languages designed for representation of thesauri, classification schemes, taxonomies, subject-heading systems, or any other type of structured controlled vocabulary. SKOS is built upon RDF and RDFS, and its main objective is to enable easy publication of controlled structured vocabularies for the Semantic Web. SKOS is currently developed within the W3C framework. It is used for export and import of controlled vocabulary within some thesaurus services used by the portal.

5.8.3.3 ISO 2788:1986 Guidelines for the Establishment and Development of Monolingual Thesauri [ISO standard]

An International Organization for Standardization (ISO) standard for the construction of monolingual thesauri. Includes guidelines for dealing with references, definitions, abbreviations, vocabulary control, indexing terms, compound terms, basic relationships, display, and management. Available for sale at www.iso.org

5.8.3.4 ANSI/NISO Z39.19-2005 Guidelines for the Construction, Format, and Management of Monolingual Controlled Vocabularies. [ANSI standard]

This American standard shows how to formulate descriptors, establish relationships among terms, and present the information in print and on a screen. Included are thesaurus

[24] http://jcp.org/aboutJava/communityprocess/final/jsr168/ [30.10.2009]

[25] http://jcp.org/aboutJava/communityprocess/final/jsr286/ [30.10.2009]

[26] http://www.w3.org/2004/02/skos/ [30.10.2009]

maintenance procedures and recommended features for thesaurus management systems. This standard is available online.

5.8.3.5 ISO 5964 Guidelines for the Establishment and Development of Multilingual Thesauri [ISO standard]

An International Organization for Standardization (ISO) standard for the construction of multilingual thesauri. This standard is to be used as an extension of the monolingual guidelines (ISO 2788). Includes guidelines for dealing with degrees of term equivalence and non-equivalence, single-to-multiple term equivalence, etc. Also includes examples of thesaural displays. Print version available for sale.

5.8.4 Implementation and Best Practice

Best practice for this component is located at the Compose Requests and Process and Present Results functions. As many portals with search functionality exist, the usability of the portals which provide the best user experience will be used as reference implementation for the BHL-Europe Portal. Also, desired functionality not available at other portals will be addressed within the best practice of this component.

The portal will be implemented as Liferay Portlet Container[27] which provides the framework to run various portlets in jsr-168 and jsr-286 as described above. This approach makes use of the community mentality of Web 2.0 which lets users create additional features on base of the BHL-Europe functionality as well as provide feedback functionality for BHL-Europe. As many community and collaboration features are already available for Liferay, this implementation approach seems to be most suitable for BHL-Europe. The developed portlets may also be used within other jsr-286 implementations like Apache Pluto which provides additional reputability for future developments.

5.8.4.1 BHL Portal - http://www.biodiversitylibrary.org/

- BHL Portal serves images & text files ingested from Internet Archive (IA)
- BHL Portal ingests MARCXML metadata & low resolution JPEG files; High resolution files are retrieved on the fly from IA
- Globally Unique Identifiers (GUIDs) allow links to other services such as EOL
- Taxonomic Intelligence developed at MBL/WHOI allows species name searching by users
    - TI uses sophisticated algorithm to locate name strings in the Optical Character Recognition (OCR) files that match the 11.1 million names in NameBank

[27] http://www.liferay.com/web/guest/home [30.10.2009]

- Iterative processing of texts increases the number of names in NameBank & the accuracy of recognition
- PDF generator enables article-level retrieval[28]

5.8.4.2 The Internet Archive - http://www.archive.org

The Internet Archive is a 501(c)(3) non-profit that was founded to build an Internet library. The Internet Archive portal includes texts, audio, moving images, and software as well as archived web pages in its collections, and is working to provide specialized services relating to training, education, or adaptive reading or information access needs of blind or other persons with disabilities.

Texts: http://www.archive.org/details/texts

Biodiversity: http://www.archive.org/details/biodiversity

5.8.4.3 EUROPEANA - http://www.europeana.eu

The Europeana prototype was launched in November 2008, to date it links to around 4.6 Mio. digital items in Europe (Images, Texts, Sounds, Videos). The site aims to give access to cultural heritage and scientific material all across Europe.

Europeana Version 1.0 is being developed and will launch in July 2010 with links to over 10 million digital objects. Europeana.eu is funded by the European Commission and the member states. The project is run by a core team based in the national library of the Netherlands, the Koninklijke Bibliotheek. It builds on the project management and technical expertise developed by The European Library, which is a service of the Conference of European National Librarians.

The Europeana portal offers a simple search field and an advanced search. Using the advanced search you can search specifically for words in titles, for names of creators (e.g. authors, artists, musicians etc.) or dates (e.g. 1945). Alternatively you can combine your search terms (e.g. title: hamlet and date: 1983).

5.8.4.4 DISMARC - http://www.dismarc.eu

The EU project DISMARC (DIScovering Music ARChives) developed during its project lifetime (2006-2008) a public prototype for an audio metadata platform and was cited a "reference" project within its EU expert evaluation. The platform is constantly growing and additionally functions today as the audio aggregation platform for EuropeanaConnect delivering audio content to Europeana.

---

[28] http://precedings.nature.com/documents/3620/version/1/files/npre20093620-1.pdf [30.10.2009]

DISMARC offers to audio content institutions the opportunity to submit **audio data** (optionally with copyright-free audio samples attached) and **collection information** for the display on the Dismarc metadata platform.

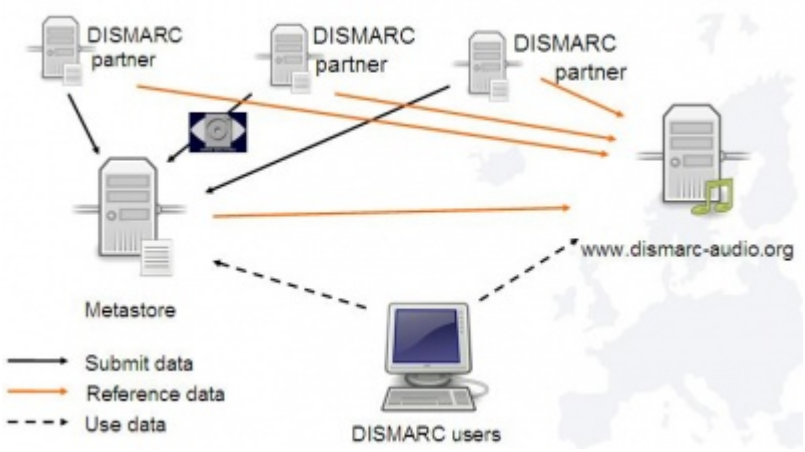


Figure 5-12:DISMARC network

The **DISMARC metastore** offers concise general information on the objects (the object metadata description and a short sound preview if possible), thus allowing users an initial selection of the objects that attract their attention. More detailed information (the original data) can then be retrieved directly from the supplying partner.
DISMARC is not excluding archives that can not yet provide digitized catalogue data of their objects. These archives can add basic information on their collections to the DISMARC metastore via the DISMARC collection description tool.

The metastore **harvests** the metadata from any **different automated systems** (providers) and provides powerful search capabilities for searching this diverse data. This eliminates the need to know how to search objects in individual systems. New metadata is harvested automatically in pre-defined intervals (each day). DISMARC uses the OAI protocol for metadata harvesting (OAI-PMH).

DISMARC offers a Simple (Google-like) Search, an Advanced Search, and Browse Facilities. In the advanced search the user may include selected or all Metadata (Any Field) fields, use Boolean operators, multilingual features, lookup lists and vocabularies when shaping the query. The search may be restricted to those records that carry audio samples and items and collections might be searched individually.

The back end tools are mainly intended for content contributors, translators and portal administrators.

Among these tools count:

- The collection entry tool
- The user administration
- The dictionary

- The portal languages tool
- The data import
- List Thesauri
- List Browse

Among the front end tools count:

- Save Searches
- Save Favorites
- Request IPR Clearance
- Search and Result List Personalization features
- Data mapping tool to the DISMARC metadata scheme (which is based on the Dublin Core Library Application Profile)
- Multilingual search features based on multilingual vocabularies and joint word list (in more than 20 languages).

The whole DISMARC system (eg. OAI Harvester and Provider, local DISMARC Metadata System) is available on a data stick.

5.8.4.5 World Digital Library - http://www.wdl.org

The WDL was developed by a team at the U.S. Library of Congress, with contributions by partner institutions in several countries; the support of the United Nations Education, Scientific, and Cultural Organization (UNESCO); and the financial support of a number of companies and private foundations. The WDL aims to make it possible to discover, study, and enjoy cultural treasures from around the world on one site, in a variety of ways. These cultural treasures include, but are not limited to, manuscripts, maps, rare books, musical scores, recordings, films, prints, photographs, and architectural drawings.
Items on the WDL may be browsed by place, time, topic, type of item, and contributing institution, or can be located by an open-ended search, in several languages. Special features include interactive geographic clusters, a timeline, advanced image-viewing and interpretive capabilities. Item-level descriptions and interviews with curators about featured items provide additional information.

6 Acronyms and Abbreviations

AIP	Archival Information Package
API	Application Programming Interface
APP	Atom Publishing Protocol
BDS	BHL Deduplication Subset (of Metadata)
BHL	Biodiversity Heritage Library
CQL	Contextual Query Language
CRC	Cyclical Redundancy Check
DI	Descriptive Information
DIP	Dissemination Information Package
DISMARC	DIScovering Music ARChives
EOL	Encyclopedia of Life
ESE	Europeana Semantic Elements
ETL	Extract, Transform, Load
Europeana	European Digital Library
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transfer Protocol
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
MARC	MAchine-Readable Cataloging
METS	Metadata Encoding and Transmission Standard
MIME type	Internet Media Type
MODS	Metadata Object Description Scheme
OAI	Open Archives Initiative
OAI-PMH	Open Archives Initiative Protocol for Metadata Harvesting
OAIS	Open Archival Information System
OCR	Optical Character Recognition
ODBC	Open Database Connectivity
PDI	Preservation Description Information
RDF	Resource Description Framework

REST	Representational state transfer
RSS	Really Simple Syndication (RSS) is a lightweight XML format designed for sharing headlines and other Web content.
SIP	Submission Information Package
SOAP	Simple Object Access Protocol providing a simple and lightweight mechanism for exchanging structured and typed information between peers in a decentralised, distributed environment using XML proposed under the W3C.
SQL	Structured Query Language: ISO, ANSI standard user front end to a relational database management system.
SRU	SRU, the Search and Retrieve URL Service, is a companion service to SRW, the Search and Retrieve Web Service. Its primary difference is its access mechanism: SRU is a simple HTTP GET form of the service
SRW	SRW defines a web service for searching databases containing metadata and objects, both text and non-text. The SRW Initiative builds on Z39.50 along with web technologies. Building on Z39.50 semantics enables the creation of gateways to existing Z39.50 systems; web technologies reduce the barriers to new information providers allowing them to make their resources available via a standard search and retrieve service.
SWORD	Simple Webservice Offering Repository Deposit
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
WAR	Web archive
WDL	World Digital Library
XML	EXtensible Markup Language
XSLT	XSL Transformation, language for the transformation of XML documents
Z39.50	National Information Standards Organization Z39.50 Information Retrieval Protocol (Z39.50/ISO 23950), a computer protocol that can be implemented on any platform, defines a standard way for two computers to communicate for the purpose of information retrieval.

7 Figures