Towards OAIS-Based Preservation Aware Storage

White Paper

IBM Haifa Labs

Authors: Simona Cohen simona@il.ibm.com
Dalit Naor dalit@il.ibm.com
Leeat Ramati leeat@il.ibm.com
Petra Reshef petra@il.ibm.com

The need to preserve digital information has grown in recent years. Today, most existing digital preservation systems are storage agnostic and may face data loss because they do not exploit features that should be supported at the storage layer. This white paper provides an overview of digital preservation from a storage perspective. The text also addresses key generic OAIS-based preservation requirements that should be supported by storage components.

November 2006
Disclaimer

International Business Machines Corporation provides this publication "as is" without warranty of any kind, either express or implied. Some states do not allow disclaimer of express or implied warranties in certain transactions; therefore this statement may not apply to you.

This publication may contain technical inaccuracies or typographical errors. While every precaution has been taken in the preparation of this document, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Commercial Relations, IBM Corporation.

All trademarks and service marks are the property of their respective owners.

© Copyright IBM Corporation 2006. All rights reserved.
# Table of Contents

**Executive Summary** ................................................................. 5  
**Glossary** .................................................................................... 7  
**Introduction** ................................................................................ 9  
What is Digital Preservation? .......................................................... 9  
Examples for Industry Needs .......................................................... 10  
Preservation vs. Archive ................................................................. 11  
Bit Preservation vs. Logical Preservation ........................................ 14  
Preservation on Tapes vs. Preservation on Disks .............................. 15  
IT Industry Trends ................................................................. 16

**OAIS Reference Model** .............................................................. 18  
OAIS Functional Model ................................................................. 18  
OAIS Information Model ............................................................... 19  
Representation Information .......................................................... 21  
Provenance ................................................................................... 24  
Fixity ......................................................................................... 26  
Migration ....................................................................................... 27  
Context ......................................................................................... 29  
Standards Compliant with OAIS ..................................................... 30

**Storage Standards Initiatives Related to OAIS Implementation** ....... 31  
Extensible Access Method (XAM) ................................................... 31  
100 Years Archive Task Force ........................................................ 32  
Object Storage Devices (OSD) ......................................................... 32
Preservation Aware Storage

Preservation DataStores and CASPAR

IBM Offerings and Preservation DataStores

DIAS

Universal Virtual Computer (UVC)

Intelligent Data Storage (IDS)

Related Preservation Frameworks and Approaches

SRB and iRODS

References

Figures

Figure 1: OAIS Functional Model

Figure 2: AIP Logical Structure

Figure 3: Information Object Model

Figure 4: Representation Information Model

Figure 5: Representation Information Network Example

Figure 6: Object Store Device

Figure 7: OSD Security

Figure 8: CASPAR System

Figure 9: DIAS System

Figure 10: SRB System
Executive Summary

Long-lived digital information is continuously being generated and utilized. The growth of this information, along with new compliance regulations such as HIPAA and Sarbanes-Oxley that require long-term viability of data, have opened the door for a new emerging research area. We must now understand how to preserve myriad types of information—including scientific, financial, healthcare, artistic and cultural data—for tens and hundreds of years.

Digital preservation systems aim at enabling the access and interpretation of digital data in the distant future by providing IT for the twofold preservation challenge: bit preservation and logical preservation. Bit preservation is the ability to restore the bits in the presence of storage media degradation and obsolescence, or even environmental catastrophes like fire or flooding. Logical preservation includes preserving the understandability and usability of the data in the future when current technologies for computer hardware, operating systems, data management products and applications may no longer exist.

Digital preservation systems will be more robust and have less probability for data corruption or loss if their storage component is a preservation aware storage, namely if their storage has built-in support for preservation. The Open Archival Information System (OAIS) is an ISO standard that specifies how digital assets should be preserved for a community of users. OAIS guided us in defining the functionalities required within the storage.

The major requirements of an OAIS-based preservation aware storage are:

- Encapsulate and physically co-locate in the storage the raw data and its complex interrelated metadata objects, such as representation information, provenance, and fixity. This ensures that the metadata needed for interpretation is not separated from the raw data and thus never lost (if the raw data survives).
- Utilize the locality property and execute data intensive functions such as fixity computations within the storage component.
- Include the representation information of metadata such as the representation information of fixity and provenance, so that the metadata can be validated and interpreted when migrating to newer systems.
- Handle the provenance events internally. The applications on top of the preservation aware storage should be freed from managing events that can be handled internally in the storage. Moreover, the types of provenance events are richer and also include events related to migration and transformation.
• Support the loading and execution of external transformations during the migration process. Additionally, it should facilitate on demand triggering of those transformations.

• Support media migration, as opposed to system migration, in which migration from one system to another can be done by physically detaching the media from one system and attaching it to the new system.

• Maintain referential integrity including updating all the links during the migration process such that they remain valid in the new system. This requires an awareness of certain meta-data fields that represent links, both internally to the system and externally.

• Ensure readability of the data by a different system in the future. This is done by developing and supporting global self-described media independent formats.

• Support a graceful loss of data. Some portions of the data are likely to be lost or become corrupted over time. If some data is lost, a good preservation system must minimize the economic effect of this data loss and prevent cases where data in the system that is still intact cannot be read or interpreted.

Preservation DataStores is an OAIS-based preservation aware storage that concentrates mainly on assisting logical preservation. It supports the requirements described above, and initially utilizes the OAIS, Extensible Access Method (XAM), and Object Store Devices (OSD) standards. Preservation DataStores is being developed as an infrastructure component for the European Union (EU) project named Cultural, Artistic, and Scientific knowledge for Preservation, Access and Retrieval (CASPAR).
**Glossary**

**Bit preservation** – The processes used to ensure that the bits comprising a preserved object are not lost or become corrupted over time. These processes include refreshing, backups, and error correcting code modules.

**CASPAR** - A European Union (EU) integrated project that will build an OAIS-based framework and architecture for the preservation of digital information, including cultural, artistic and scientific data.

**Content data object** – The 'raw' data that makes up the content to be preserved.

**Content information object** – The raw data and the metadata needed to interpret it; namely the content data plus its representation information.

**Context** – The relationship of the content information to its environment, initially as perceived by the content data object provider. Later on, more context information may be implied or derived from the preservation process.

**Designated community** - The primary OAIS user group that needs to access and understand the information preserved in the OAIS system. This means that the OAIS must have an appreciation of the community's knowledge base.

**Digital preservation** – The series of managed activities necessary to ensure continued access to digital materials for as long as necessary.

**Fixity** – The information that documents the authentication mechanisms and provides authentication keys to ensure that the content information object has not been altered in an undocumented manner.

**Logical preservation** - The processes used to ensure the understandability and usability of the data, in spite of the unknown changes in technologies and users in the future.

**Metadata** – Information about the content data object that is needed for its preservation.

**Migration** – The act of moving data from one system to another because of a change.

**OAIS** – Open Archival Information System. An ISO standard that specifies a reference model for an archive, consisting of an organization of people and systems that have accepted the responsibility to preserve information for a designated community.
**Preservation system** – An archiving system in which the lifetime of the data it needs to store exceeds the lifetime of the program/format with which the data is interpreted and the lifetime of the media that stores the bits.

**Preservation aware storage** – The storage component of a digital preservation system that has built-in support for preservation.

**Preservation DataStores** – A new OAIS-based preservation aware storage, which is the storage component of CASPAR infrastructure.

**Provenance** – The information that documents the history of the content information. This information describes the origin or source of the content information, any changes that may have taken place since it was originated, and who has had custody of it since it was originated.

**Representation information** – The information that is required to interpret the content data object (raw data) into more meaningful concepts (ultimately more meaningful for humans).
Introduction

What is Digital Preservation?

Digital preservation is the set of processes, activities, and technology utilized to store and access large amounts of heterogeneous digital data for long periods of time—covering tens if not hundreds of years. It enables people or systems to use and understand the data in the far future in spite of the unknown changes in users and technologies such as computer hardware, operating systems, and applications. An example of data that requires preservation is a Word 3.0 document that was running in 1986 on an IBM PC AT with the MS-DOS 3.0 operating system and stored on a 5¼” floppy disk.

Digital preservation is an inherently cross-functional task and there are many stakeholders involved in a preservation environment including IT, records and information management, security, legal, regulatory compliance and even finance. Digital preservation also requires a large spectrum of technology domains such as databases, search, ontologies, digital rights management, ingest and access applications, and of course storage.

There are various approaches to digital preservation. One approach is the museum approach in which the content and rendering devices are kept in their original state and maintained operational. For example, to preserve the Word 3.0 document with this approach, an IBM PC AT with MS-DOS 3.0 and Word 3.0 application as well as the 5¼” floppy disk are kept forever in a designated storage room. This approach doesn’t allow the addition of new applications to the data in the future, and requires maintenance of many software and hardware.

Another approach is the emulation approach, in which up-to-date technology is used to emulate an older or obsolete technology. For example, Xen [1] includes an emulation of a computer hardware and operating system. UVC (see last chapter) emulates applications by rewriting them in the UVC universal language. To preserve the Word 3.0 document with this approach, the Word 3.0 application needs to be rewritten to the UVC language, either manually or via some adequate compiler.

A third approach is the migration approach, in which each time the hardware, operating system, application, or data format becomes obsolete, it is migrated/transformed to the respective new technology. To preserve the Word 3.0 document with this approach, the document would have been constantly
transformed to Word 4.0, 5.0, and so forth until today’s Word 2006 on Windows XP. This approach may require substantial resources when the migration rate is high, and come at the expense of reduced performance in acquiring and preserving new digital data. The repeating transformations also put on risk the fidelity of the data.

A fourth approach is the descriptive approach, which keeps a description that enables reproduction of the data. To preserve the Word 3.0 document with this approach, a description of the content in a non-Word format is kept. Another example would be preserving a music concert by describing the notes and emotions. This approach has difficulty preserving the full interpretation of the data, and thus, is adequate only for specific types of data.

Today, it seems there isn’t a one optimal approach for digital preservation, and the right approach depends on the data to be preserved, the purposes for which it is preserved, the use cases, and the organizational policies. In many cases, a combination of the various approaches is needed. However, all approaches use storage, and all of them can benefit from storage support for digital preservation—including providing self-describing encapsulation of data and its metadata, long term integrity and authenticity, and long term provenance. The storage component of a preservation system should support all the digital preservation approaches noted here.

Examples for Industry Needs

Long-lived digital information is continuously being generated and its use is growing. This information is sometimes also required by new compliance regulations such as HIPAA, Sarbanes-Oxley, Occupational Safety and Health Administration (OSHA), and federal securities laws and regulations. The following lists some examples of industry domains and their need for long-lived data.

Healthcare
- Medical records should be preserved for the life of the individual and beyond.
- X-rays are often stored for periods of 75 years.
- OSHA requires organizations to keep records of both medical and other employees who are exposed to toxic substances and harmful agents. Employers must maintain these records for 30 years [2].
- The retention requirement for the medical records of minors varies from 20 to 43 years of age [2].
Pharmaceutical companies
- Pharmaceutical companies need off-line electronic data storage for 50 to 100 years or longer. They need to keep their research, development and filing application records for decades to comply with FDA 21 CFR Part 11, in addition to protecting themselves against crippling litigation.

Finance
- Rule 17a-4 of the federal securities laws and regulations requires broker-dealers to retain account record information for six years. The six-year period begins either at the time the account is closed or when the information is replaced or updated, so it may total up to 70 years or more [4].
- Life insurance policies have to be kept for the life of the policy plus 6 to 10 years.

Aerospace
- Aircraft designs records have to be retained for the lifetime of each aircraft (30+ years).

Automotive
- European Union regulations require vehicle manufacturers to provide disassembly and material data after vehicle disposal. This could be many years after that particular product had been replaced or discontinued. Furthermore, they must track any modifications or replacements that may have occurred during the lifecycle of the vehicle, so that the exact material content is known at the time of stripping and recycling.

Petroleum
- Oil-field data is used during the entire life of the field (50+ years).

Scientific and Cultural
- Satellite data is kept forever to enhance research and track changes in the earth and the cosmos.
- Federal agencies such as the Library of Congress and the National Archives and Records Administration (NARA) are commissioned to preserve documents, in some cases forever.
- Society would like to keep libraries and art data forever.

Preservation vs. Archive
Digital archiving refers to the ability to safely store and access digital data. Digital preservation is a special case of digital archiving, where the lifetime of the stored data exceeds the lifetime of the program/format with which the data is interpreted as well as the lifetime of the media that stores the bits. It is
therefore natural to ask: What are the aspects that differentiate a data preservation system from an archival system that stores and accesses data for relatively shorter periods?

To address this question, we first characterize the data that is stored in a preservation system, and the unique processes it undergoes during the long preservation life-cycle.

**Characterization** - Digital data in preservation systems has the following characteristics:

i. The content data is Read only.
ii. Most content data is *cold* and is rarely accessed during its lifetime.
iii. New metadata can be added later on.
iv. The content data is heterogeneous in type, size and value (some data is more important than other).
v. The content data quantity may be too large for on-line storage.

**Processes** - The following preservation-related processes may affect the data:

i. The process of packaging the content requires storing metadata (sometimes this will be a substantial amount of additional data) together with the data in order to interpret it. Some of it is referential data, which represents either relationships within various data objects in the system, or references the representation information.

ii. The additional information needed for interpretation (the object’s representation information) may reside elsewhere, outside the system that stores the actual data, and may also evolve over time.

iii. The data must be migrated and possibly transformed, to support obsolescence of formats, hardware or software.

iv. The data is likely to be accessed by a system that is not the same as the system which originally ingested and stored it.

v. The likelihood that some of the data will be lost or become corrupted over time is large, especially as data gets migrated and transformed repeatedly.

Given these characterizations, what are the special requirements of a preservation-aware storage? What should a preservation-aware system support beyond the services supported today by a short-term archive system? Below are some specific examples.
1. Extend the data migration process to handle not only media
degradation but also replication, repackaging and transformation (due
to format conversion and software migration), and to possibly add new
metadata. This requires the storage to react to events that are
external to the storage, for example events triggered by a
preservation planning component. It also extends the notion of
provenance to include the event of migration to another data format or
another physical media.

2. Maintain an extended notion of object integrity that goes beyond bit
integrity, e.g. referential integrity.
   a. Ensure that links point to existing/valid locations. This requires
      an awareness of certain meta-data fields that represent links,
      both internally to the system and externally.
   b. Update links. Representation information links into registries
      may have to be updated due to migration of the registries.
   c. Support partial integrity of the information object, as well as
      information on the parts that intact and those that are
      corrupted. If the information object integrity check fails, it is
      possible to detect the sub-part that was corrupted.
   d. Package data on the media level such that it is self-contained,
      ensuring that all the object's components are co-located
      physically.

3. Support long term aspects of data integrity and encryption.
   a. The algorithms employed to compute data integrity may need
      to change, due to loss of security, or when better algorithms
      are developed as cryptographic algorithms are subject to
      increased brute force attacks over time. The keys of the
      cryptographic algorithms may also need to change. By
      anticipating these changes, this update can be done during the
      migration process.
   b. Export the cryptographic mechanisms that were used to
      compute integrity and encryption (if utilized), to support
      access to the data by a different system. This requires building
      representation information for the fixity.

4. Ensure readability of the data by a different system in the future. This
   is done by developing and supporting global self-described media
   independent formats.
5. Support a *graceful loss* of data. Some portions of the data are likely to be lost or become corrupted over time. If some data is lost, a good preservation system must prevent cases where data in the system that is still intact cannot be read or interpreted.

**Bit Preservation vs. Logical Preservation**

Digital Preservation includes two related technologies, “bit preservation” and “logical preservation”. Bit preservation includes the ability to restore the bits in the presence of storage media degradation and obsolescence, or even environmental catastrophes like fire and flooding. For example, bit preservation is responsible for ensuring the bits can be read after 5 years. A more complex example for bit preservation is ensuring the restoration of bits that were created on a 5¼” floppy disk located in a storage room that was damaged by fire.

Logical preservation includes preserving the understandability and usability of the data, despite unknown future changes that will take place in technologies and users (designated community). The data needs to be properly accessed and interpreted in the far future when current technologies for servers, operating systems and data management products may no longer exist. For example, assuming we can get the bits of a Word 3.0 document, how do we now read, understand and interpret it? Additionally, the logical preservation needs to maintain the provenance of the data, its authenticity, its integrity, and ensure that only legitimate users will access it.

Bit preservation is typically the responsibility of the storage layer. The storage layer includes data protection mechanisms such as RAID, or erasure coding. It may include mirroring services that maintain at least two copies of each data object, with each copy held at a physically separate location. Because the value of the data objects is mixed, the degree of protection for each data object may vary according to its importance. Because the data is located in multiple places, replicated data may introduce difficulties in refreshing, migrating, versioning, and access control.

Moreover, media may deteriorate. To support bit preservation, the storage layer refreshes the data according to a refreshment schedule. Generally, this is done every three years for hard disks and every five years for tapes. Although the life expectancy of tape is 10 to 30 years, refreshment is done sooner because of tape technology updates. Bit preservation strategies are well
known and have been well tested in applied information technology. The challenge is mostly organizational rather than technical.

In contrast, logical preservation is still an unresolved problem. It is a recursive problem, where in addition to storing the raw data, it must also store the separately-born (in time and place) metadata that helps interpret and use the data itself. Moreover, this metadata (representation information) may need recursively additional metadata to help interpret itself. The recursion ends when the representation information is non-digital and preserved by the designated community (see example in the representation information section).

To further support logical preservation, the raw data is associated with metadata that describes its context, logs its provenance, and assures its fixity. All these forms of data need to be migrated over time from current systems to newer systems as the present systems become obsolete. Moreover, data transformation may occur within the migration process to replace obsolete formats or add new formats.

*Preservation aware storage should support logical preservation in addition to the traditional bit preservation.* The storage should encapsulate the raw data with its complex interrelated metadata objects, so they are inseparable during the migration processes and when accessing the data in the future. The system should decrease the data transfer between the applications and the storage by offloading data intensive functions such as fixity to the storage. The system should also simplify the applications by transferring the responsibility of managing the storage-related events, such as provenance events, to the storage itself. The storage should handle migration including the ability to execute externally specified transformations.

**Preservation on Tapes vs. Preservation on Disks**

Tape storage systems and disk storage systems are currently the prominent types of media on which data is preserved. Individual disk drives provide random access and sub-second performance for transferring 50 MB, but they are not reliable and tend to deteriorate approximately every three years. Tapes provide sequential access and their transfer time is slower than that of disks. However, tapes are more reliable than disks and their expected lifetime is 3 to 10 times higher than that of disks. Additionally, tapes consume 25 times less power than disks. Hence, the power and cooling costs for tapes, which over five years comprises 10% to 20% of total midrange storage system
cost, is much lower [55]. Overall, tapes are much more cost-effective than disks.

Tapes may be used as near-line storage by using a tape library in which there is no need to manually load and unload tapes. Monitoring tape bar code sequences is an automated process and even predicting media failure is automated with alerts sent to a remote site. Tapes can also be used as off-line storage in which the tapes are not connected to the system and manual intervention is needed prior to access. Off-line tapes are considered more secure as they are less susceptible to viruses and user errors.

The amount of digital preservation data may become very large in the future. The U.S National Archives and Records Administration (NARA) [6] projected that in the year 2010, they will have 10,000 TB of data to be preserved forever, which will then rapidly increase to 230,000 TB in the year 2020, and to 350,000 TB in the year 2022. This is accompanied by large volumes of metadata that must be stored to interpret the raw data. Consequently, the actual volume is even larger. Nevertheless, over the years, most preserved data is rarely accessed. Some peaks may occur in which the data is hot and frequently accessed such as when the data was born, or when some disruptive event happens. Over time, the preservation data tends to be cold (inactive) and is seldom accessed. We generally expect to see the use of a media blend of disks and tapes, where the disks are used for caching purposes and tapes are the ultimate place for the data. But we don’t really know what the media of the future will be.

*The preservation aware storage should be agnostic to the media and support various types of media.* The media format and layout should take into account the random access mode of disks versus the serial access mode of tapes.

## IT Industry Trends

Digital archive capacity (short-term and long-term) is expected to significantly increase in the coming years. This trend is being fueled by regulatory compliance, corporate governance, the desire to protect the investment put into creating the information, as well as addressing the risks of litigation and loss of prestige. The content of the archive data mainly includes a myriad of different record types including scientific data, artistic data, cultural data, e-mail, office documents, invoices and other customer records, financial statements, digital images, historical trading data, medical records, contracts, and web pages.
Today, many organizations do not know the volume of digital information they need to preserve or the amount of time they need to preserve it. Most organizations are still in the early stages of contemplating digital preservation. Moreover, there are no existing products that will be around to interpret digital information in the future the way it is done today, and ensure the integrity of the archive after their underlying technologies and even the companies and organizations that created them have become extinct.

Today, existing products address only some aspects of preservation but not the whole problem. Vendors like EMC, IBM, and Vignette market products for archiving and organizing content. Some, including IBM, Archivas, EMC, and PermaBit are building content-addressable and object-based archives that will inexpensively organize a very large numbers of digital objects. Other vendors are working on virtualization, data movement, management, and compression technologies. All these products need to be orchestrated and enhanced to building a coherent suite of products and services for long term information preservation.
OAIS Reference Model

The Open Archival Information System (OAIS) [7, 8, 9], an ISO standard since 2003 (ISO 14721:2003 OAIS), specifies a way in which digital assets should be preserved for a community of users from the moment digital material is ingested into the digital storage area, through subsequent preservation strategies, to the creation of a dissemination package for the end user. The OAIS is a high-level reference model, which means it is flexible enough to use in a wide variety of environments. More detailed steps and workflows stages will have to be developed by the implementing institution.

OAIS Functional Model

The reference functional model identifies and describes the core set of high-level functional components, which together provide the designated community with long-term preservation and access.

Figure 1: OAIS Functional Model
• **Ingest** – Responsible for accepting information submitted by Producers and preparing it for inclusion in the archival store.

• **Archival Storage** – Responsible for ensuring that archived content resides in appropriate forms of storage and that it remains complete and renderable over the long-term. This is done by periodic media refreshment or format migration, as well as implementation of safeguard mechanisms such as error-checking procedures and disaster recovery policies.

• **Data Management** – Maintains databases of descriptive metadata identifying and describing the archived information. It supports search and retrieval of the OAIS' archived content.

• **Preservation Planning** – Responsible for monitoring the environment and developing recommendations for updating the OAIS policies and procedures to accommodate these changes.

• **Access** – Manages the services and processes by which consumers locate, request and receive delivery of items from the archive.

• **Administration** – Responsible for managing the day-to-day operations of the OAIS and coordinating the activities with the other five high-level OAIS services.

---

**OAIS Information Model**

The OAIS information model provides a high-level description of the information objects managed by the archive.

1. **SIP** – Submission Information Package. The information package that is transferred from the producer to the OAIS.

2. **AIP** – Archival Information Package. The information package that is stored and preserved by the OAIS. It consists of the preserved information accompanied by a complete set of metadata.

3. **DIP** – Dissemination Information Package. The information package delivered to the consumer. It contains the data and relevant metadata. It
may differ from the AIP in the format of the content, the amount of data and metadata, depending on the designated community.

**AIP – Archival Information Package**

![AIP Logical Structure](image)

*Figure 2: AIP Logical Structure*

*Content Information* contains the content data object that is the focus of the preservation. OAIS also requires the archive to maintain the representation information required to render the object intelligible to its designated community. This may include information regarding the hardware and software environment needed to view the content data object.

OAIS breaks the Preservation Description Information (*PDI*) down into four sections:

- *Reference information* – A unique and persistent identifier of the content information within the OAIS, and externally (e.g., UUID).
- *Provenance information* – The history and the origin of the archived object.
- *Context information* – The relationship to other objects, for example, the hierarchical structure of a digital archive.
- *Fixity information* – A demonstration of authenticity, such as a checksum.

*Packaging information* is used to bind all of these information components into a single logical package.
**Representation Information**

*Representation information* is defined as the information required to interpret a data object into more meaningful concepts (ultimately more meaningful for humans).

![Information Object Model Diagram]

**Figure 3: Information Object Model**

In a digital archive, representation information is the information necessary to render/display, understand, and interpret the content data object. Therefore, representation information is the metadata that translates the bit stream of the content data object into accessible knowledge (or information) that is meaningful to current and future users. Such metadata might include the description of the hardware and software environment or specify different standards. One example of metadata is the ASCII definition that describes how a sequence of bits (i.e., a content data object) is mapped into a symbol.
Figure 4: Representation Information Model

OAIS divides representation information into several components.

*Structure information* provides a technical description of the content data object’s structured organization, including format, data structures, encoding, and more, in particular as it relates to rendering or displaying the object in a digital environment. For example, it maps bit streams to common computer types such as characters, numbers, and pixels, along with aggregations of these types, such as character strings and arrays.

*Semantic information* imparts higher level meaning to the structural components of the content data object, beyond what is expressed by structure information. Thus, semantic information might indicate that a sequence of alphanumeric characters should be interpreted as English prose, or that a sequence of integers represents temperature readings from a chemistry experiment. In this sense, semantic information contributes towards an understanding or appropriate interpretation of the intellectual content of the content data object.

The representation information applicable to a content data object is in itself a digital object and may itself require further representation information to enable its use. In this way, a recursive *representation network* may be created. This recursive process ends with information that is non-digital. This
information (or knowledge) can be represented either by a physical object or by using the concept of a designated community and its knowledge base.

Note that the scope of the designated community impacts the amount of metadata required to support the preservation process. In general, the broader the scope of the designated community, the less specialized the knowledge base associated with that community – hence, more representation information is needed. Over time, evolution of the Designated Community's Knowledge Base may require updates to the Representation Information to ensure continued understanding.

The ISO 9660 standard [1010], which defines a file system for CD-ROM media, can be considered an example of a representation network. Consider data file stored on a CD-ROM. The file and directory structure of many CD-ROMs conforms to ISO 9660. This file system standard describes text as conforming to the ASCII standard, but does not actually describe how the ASCII is to be implemented. It simply references the ASCII standard, which is additional representation information that is needed for a full understanding. Therefore, the ASCII standard is a part of the representation network associated with ISO 9660 and needs to be obtained in some form to ensure that the ISO 9660 representation information is fully understandable.

![Diagram of Information Object Flow](image)

**Figure 5: Representation Information Network Example**

Note that representation information is most likely a resource that is common to many content data objects. Meaning, that many content data objects can share one representation information that is relevant to all of them. For example, the ASCII standard can be common representation information of a
Word file and a PDF file, when both describe text as conforming to the ASCII standard.

Other representation information may indicate software, standards, algorithms, packaging information, and documentation that relate to the digital object [11].

There are two special types of software representation information:

- **Representation rendering software** is a type of software that displays the representation information of an information object in forms understandable to humans. One such example is PDF software that is used to render the description of ISO 9660.

- **Access software** is a type of software that presents part of or all of the information content of an information object in forms understandable to humans or systems. Often the lack of adequate, associated, representation information is hidden by the use of currently available access software. However, the OAIS reference model takes the position that software is not an adequate substitute for representation information because it cannot be relied upon to continue working and because it further obscures the underlying information on which it depends.

A preservation aware storage should support representation information. It should ensure that the representation information is never lost or separated from the raw data and that the referential integrity of the network of representation information is maintained.

**Provenance**

In the archival community, provenance refers to the chain of ownership and the transformations a document has undergone.

In traditional archives, provenance is the key according to which records are arranged. It serves two functions [12]:
- Maintains the *chain of custody*. This is important for the reliability of the data and for intellectual property considerations.
- Helps in information retrieval, especially over time, when consumers vary in their perspectives and the single best way to organize records is according to their provenance.

In a digital archive, *provenance information* documents the history of the content data object. This tells the origin or source of the content data object, any changes that may have taken place since it was originated, and who has had custody of it since it was originated. Provenance information can be considered chronological event-based metadata. The evolutionary process associated with the object is driven by the occurrence of important events such as the object's creation, transfer of ownership, migration to another data format, etc. Recording these events and their impact on the content data object is the key function of provenance information.

The following delineate the major phases or aspects of the content data object's life cycle [13]:

- **Origin** – describes the process by which the content data object was created.
- **Pre-ingest** – describes the history of the content data object in terms of maintenance, changes in content, custody, and so forth, from its creation to its submission to the archive.
- **Ingest** – describes the process by which the content data object is prepared and ingested into the archive (e.g., object migrated to archive's standard storage format, AIP assembled).
- **Archival retention** – describes the maintenance, changes in content, management, and so forth of the content data object during its retention in the archival store (e.g., migration history, media refreshment history, etc.).
- **Rights management** – specifies the legal uses of the content data object. Documents the archive’s scope to preserve and disseminate the content data object (e.g., access permissions, legal deposit responsibilities).
Each of these phases/aspects may consist of the occurrence of one or more events that impact some aspects of the content data object. A description of such an event should include the following elements:

- **Designation** – name of the event. (e.g., change in custody, migration, media refreshment).
- **Procedure** – details of the procedures constituting an occurrence of the event (e.g., timing and procedural steps associated with a format migration).
- **Date** – date the event occurred. Establishes appropriate chronology of the events.
- **Responsible agency** – entity responsible for the successful occurrence of the event (e.g., archival system administration staff).
- **Outcome** – documents the impact of the event on the relevant aspects of the content data object (e.g., object successfully migrated from Microsoft Word 97 to PDF).

Some of the events that influence provenance may only occur in a preservation system and not in an archive system. For instance, content data object transformation is an event that only occurs in systems in which the content data object lives longer than the application that created it.

In a preservation system the representation information of the provenance needs to be kept as well to enable future interpretation of the content of the provenance information.

A preservation aware storage should support provenance. Since most of the provenance events are executed in the storage, it is natural that the logging of these events will also be performed within the storage. This will also free the application from unnecessary extra responsibilities.

**Fixity**

Fixity information provides the data integrity checks or validation/verification keys used to ensure that the particular content data object has not been altered in an undocumented manner. Fixity information includes special encoding and error detection schemes that are specific to instances of content objects.
A variety of techniques are available for authenticating content data objects - check sums, digital signatures, watermarks, etc. We need to allow updates to the fixity algorithms to keep up with the increasing computing power and the development of new algorithms.

The object authentication metadata must include the following:

- **Authentication type** – identifies and describes the authentication method applied to the content data object.
- **Authentication procedure** – provides enough information and tools to carry out the authentication type applied to the content data object (e.g., pointer to software implementation of this authentication type).
- **Authentication date** – specifies the date of the most recent archival use of this authentication type.
- **Authentication result** – specifies the result of most recent archival use of this authentication type.

In a preservation system the representation information of the fixity needs to be kept as well to enable fixity validation in the future.

The fixity algorithm should be executed on divided parts of the Content Data Object in a way that the various partial fixities will be combined using a computation tree. This will reduce computation when only some of the data changes as well as help identify the invalid part when fixity validation fails.

A *preservation aware storage should support fixity*. Since fixity is a data intensive function that gets large data as input and generates a short output, it is preferable to utilize the locality property and compute the fixity within the storage. This gives better performance and is more robust as it saves the bandwidth of transferring large amounts of data to the application.

**Migration**

Migration is the act of moving data from one system to another because of a change. The nature of the change may be one of the following:

- Possible decay of storage media
- Obsolete hardware or software (encompasses obsolete file formats)
- Change in availability of software or documentation (copyright issues)
- Change in external environment e.g., organization, staff
Migration is a major characteristic in preservation environments. The OAIS reference model identifies four primary digital migration types:

1. **Refreshment** – bit-to-bit copy of the entire media onto newer media of the same type, without changing the bit sequence of either the packaging or content information, or the placement of the data objects. As a result, the existing archival storage mapping infrastructure, without alteration, is able to continue to locate and access the AIP.

2. **Replication** – copying data onto newer media that is not necessarily of the same type, but without changing the bit sequence of either the packaging or content information. Note that refreshment is also replication, but replication may require changes to the archival storage mapping infrastructure.

3. **Repackaging** – copying data while changing the placement of the components within a data object. This changes the bits of the packaging information but not the content information object itself.

4. **Transformation** – copying data while performing format change on the data. This may change the bit sequence of both the packaging and content information object. Data that is transformed runs the risk of losing some type of functionality since newer formats may be incapable of capturing all the functionality of the original format, or the converter itself may be unable to interpret all the nuances of the original format. The latter is often a concern with proprietary data formats.

Digital migrations are time consuming, costly, and expose the preservation system to greatly increased probabilities of information loss. The ways in which AIPs are implemented will have a major influence on both the level of automation and the probability of information loss during migrations. A well-designed AIP can both increase migration automation and reduce information loss probabilities. The described migration types are ordered according to their rate of being time consuming and their risk of information loss. Refreshment is the migration type that is the least time consuming and has the lowest risk of information loss.

*In order to improve migration performance and reduce the risk of information loss, a preservation aware storage should support migration within the storage.* It should utilize a self-describing self-contained media format that will reduce the amount of external information needed to interpret the data of a single media. It should check and maintain referential integrity within the migration process. It should also support the execution of transformations within the migration process.
Context

Context information documents the relationships of the content data object to its environment, initially as perceived by the content data object provider. Later on, more context information may be implied or derived from the preservation process.

One aspect of the context information is to place the content data object in the context of the motivation or purpose for its creation. For example, a TIFF file was created to serve as a digital surrogate for a rare, fragile paper document, in order to facilitate access and protect the original resource.

Context information also documents significant relationships among the preserved object and other content data objects, which are important for managing the preservation process.

Many relationships can be identified between the preserved object and the other content data objects. Here are two important relationship categories:

- **Relationship to other manifestations of the content data object.** This relationship category is essential for maintaining a change history for the object (i.e., recording outcome of a format migration process), or relating alternative versions of the current object. This category would include versions of the object in alternate software formats such as HTML, PDF, and Microsoft Word versions of the same document. It would also include different versions of the object in the same software format such as Microsoft Word 6.0, 97, and 2000 versions of the same document.

- **Relationship to other content data objects whose intellectual content is related to that of the preserved object.** This relationship category identifies groups of related objects that exist within the archive. For example, this may include: a set of PDF documents, each representing a chapter of a book; a collection of objects representing digitized images of an art collection; a set of individual objects (HTML, GIF files, etc.) which, in aggregate, form a web page.

A preservation aware storage should support context. Because the storage supports format migration, which is a process that changes the context, it is preferable to utilize the locality property and update the context information within the storage. The semantic relationship described by the context can be used by the storage to generate better layout of data. For example, the set of PDF files, each representing a chapter of a book, should probably be stored together physically.
Standards Compliant with OAIS

- **Content packaging techniques**
  - METS - a standard for encoding descriptive, administrative, and structural metadata regarding objects within a digital library [14].
  - XFDU – an emerging CCSDS recommendation to package data and metadata, including software, into a single package to facilitate information transfer and archiving [15, 16].

- **Preservation metadata schemata**
  - PREMIS – a preservation metadata schema supported by a data dictionary, intended for use in a variety of digital preservation situations [17].
  - Dublin Core Metadata initiative – an open forum engaged in the development of interoperable online metadata standards that support a broad range of purposes and business models [18].

- **Domain specific standards**
  - LOTAR – an attempt to specify standards for the long term archiving of (3D) CAD models and PDM documents for Aerospace projects [19].
  - SAFE – a standard compliant with OAIS and XFDU designed to act as a common format for archiving and conveying data within ESA earth observation archiving facilities [20].
Storage Standards Initiatives Related to OAIS Implementation

OAIS requires support for complex interrelated objects with large metadata that have globally unique IDs. It requires storage that understands objects and can associate metadata with the content. Extensible Access Method (XAM) and Object Storage Devices (OSD) are new storage standards that can support this, and are thus relevant to OAIS implementations.

Extensible Access Method (XAM)

XAM [2121] is a Storage Networking Industry Association (SNIA) initiative to define a standard interface between consumers (application and management software) and providers (storage systems). It was initiated by IBM and EMC in Q4 2004 and now is supported by major vendors including Seagate, Microsoft, ByCast, Hitachi, HP, HDS and SUN.

A XAM storage system includes one or more XSystems with each XSystem being a logical container of XSet records. An XSet, which is the basic artifact in XAM, is a data structure that is a package of multiple pieces of data and metadata, bundled together for access under a common globally unique external name, called an XUID. An XSet is a collection of XSet Fields. There are two types of XSet Fields:

- **Properties** – fields that usually include metadata and can therefore be indexed and used in queries. Their type is a “simple” type and is one of Boolean, int64, uint64, float64, string, datetime, xuid. The property type is checked and enforced by the storage system. The API to these fields is via get/set methods.

- **XStreams** – fields that include unbounded byte streams. Their type is a valid MIME-type but the type is not checked or enforced by the storage system. The API to these fields is based on Posix I/O methods such as open, read, write, and close.

Future versions of XSet may introduce a new type of field called Structured Streams, which like XStreams include unbounded byte streams and are manipulated by Posix-like methods. However, their type is expected to be an XML schema and validation with that XML schema is checked and enforced by the storage system.
Each XSet field has a fixed set of attributes for its content and behavior that are manipulated via get/set methods. The attributes are:

- **type** – the type of the value of the field; namely simple type for properties, MIME-type for XStreams.
- **value** – the actual value (content) of the field.
- **fixed** - a Boolean value indicating whether the field is immutable within the XSet. If true, then field modification causes the automatic creation of a new XSet with a different XUID.
- **readonly** - a Boolean value indicating whether the field can be modified by an application. If true, then an attempt to modify this field will return an error.
- **length** – the actual size of the field value in bytes. This attribute value is provided by the XAM system and is read-only for the application.

### 100 Years Archive Task Force

The SNIA 100 years archive task force aims to define best practices and storage standards for long term digital information retention. The task force goals are [22]:

- With a multi-disciplinary team, produce a paper on "best practices for long term digital information retention”.
- Influence Information Lifecycle Management (ILM) standards to support management and automation practice for long term archive.
- Guide the impact of new storage technologies such as XAM and Grid to improve long term retention methods.
- Define a standard for on-media formats and long term readability.

### Object Storage Devices (OSD)

Object-Based Storage Devices (OSD) enable the creation of self-managed, heterogeneous, shared and secure storage by moving low-level storage functions into the storage device itself and accessing the device through a standard object interface rather than a traditional block-based interface. The OSD technical working group at SNIA [23] develops models and guidelines,
requirement statements, preliminary standards definitions, reference code and prototype demonstrations for OSD storage subsystems.

The first standardization effort of an Object Storage Device specification is embodied over the SCSI protocol and is being realized as a new set of SCSI commands. Version 1 of the T10 standard was publicly reviewed and approved in late 2004; the OSD standard was published as ANSI INCITS 400-2004 [24]. Many companies were involved and contributed to the standard. Among them are: HP, IBM, Intel, Panasas, Seagate, Veritas (now Symantec) and Sun. Presently, the standard is being extended to Version 2. Extensions include advanced functions such as snapshots, multi-object operations, collections, and error handling.

What is an OSD object?

An OSD user-object is a container of data and attributes, indexed by a 128-bit object identifier. The object identifier is composed of a 64-bit partition ID and 64-bit object ID.

User objects are grouped into partitions, where each user-object belongs to a single partition. An object can be a member of a collection of objects, all of which reside in the same partition. An object can belong to multiple collections.

An object is created and deleted via the Create/Remove command respectively. The data of the object can be accessed via OSD I/O commands (e.g. Read, Write, Append), while its attributes can be accessed via the Set/Get attribute commands. I/O commands may access any number of bytes at any logical offset of the object.

User Data - An object is a sparse collection of data bytes, addressed by their logical offset in the object. That is, the object may be composed of ‘holes’ corresponding to offsets that were never written to with data. Semantically, such holes contain the value of ‘zero’. An object has two associated parameters – its logical length and its size. The logical length is defined as the largest offset that has been written to and the size is the actual number of bytes it consumes on the physical media.

Attributes – the object’s attributes are data bytes that are maintained and stored persistently with the object’s data. The standard defines a basic set of attributes, some of which can be user-defined, whereas others are not. The standard provides a mechanism to extend this basic set into user-defined attributes. All attributes are type-less.
Attributes are grouped logically into ‘pages’ and accessed via a pair of indices (page number, attribute number). The pages are:

- Directory page
- Information page
- Quota page
- Timestamps page
- Collections page
- Policy/Security page

![Figure 6: Object Store Device](image)

**OSD Security**

An important aspect of an object-based storage device is its security model, which enforces access control on every command. The security aspect of the OSD protocol is based on a cryptographically secured capability. Credentials are issued by the Security/Policy Admin, a trusted entity, which is assumed to enforce a given policy, as shown in the following figure.
Figure 7: OSD Security
Digital preservation systems will be more robust and have less probability for data corruption or loss if their storage component is a preservation aware storage, namely it has built-in support for preservation. An OAIS-based preservation aware storage is required to:

- Support logical preservation in addition to the traditional bit preservation e.g. support handling AIPs.
- Be agnostic to the media and support random access media as well as serial access media.
- Support global persistent identifiers, so identifiers of preserved objects are maintained over many years as the objects are migrated.
- Encapsulate and manage within the storage the raw data and its complex interrelated metadata objects, such as representation information, provenance, and fixity. This ensures that the metadata needed for interpretation is not separated from the raw data and thus never lost (if the raw data survived).
- Utilize the locality property and execute data intensive functions such as fixity computations and transformations within the storage component.
- Include the representation information of metadata such as the representation information of fixity and provenance, so that the metadata can be validated and interpreted when migrating to newer systems.
- Handle internally the provenance events. The applications on top of the preservation aware storage should be freed from managing events that can be handled internally in the storage. Moreover, the types of provenance events are richer and also include events related to migration and transformation.
- Support the loading and execution of external transformations during the migration process. Additionally, it should facilitate on demand triggering of those transformations.
- Support a self-contained self-describing medium format adquate for random access and serial access. Self-describing facilitates a portable format when migrating from one system to another. Self-contained
means that all the information needed to access and interpret the data resides within the same media; thus, in the migration process the new system is able to interpret the data of the new media without additional external information.

- Ensure graceful loss of data. Since each media is self-contained and contains its metadata, a single tape/disk loss or corruption should not affect other tapes/disks, except the one damaged. Also, ensure that losing some number of sectors on a single medium does not result in loss of all data on the medium.
- Support media migration (as opposed to system migration) in which migration from one system to another can be done by physically detaching the media from one system and simply attaching it to the new system.
- Maintain the data protection according to externally set policies on the importance of data taking into account physical placement and encoding schemes.
- Maintain referential integrity including updating all the links during the migration process such that they remain valid in the new system.
- Maintain the context information and update it in the migration process and ensure its maintained validity.
- Provide fine-grain security per content data object.

**Preservation DataStores and CASPAR**

**Preservation DataStores** is a value-add preservation aware storage that assists in the storage aspects of digital preservation, concentrating mainly on logical preservation. It supports OAIS-derived concepts and functions as well as the preservation aware storage requirements described above. Our initial realization of Preservation DataStores will mainly utilize OAIS, XAM and OSD.

The Preservation DataStores is being designed as an infrastructure component for a new European Union (EU) project named Cultural, Artistic, and Scientific knowledge for Preservation, Access and Retrieval (CASPAR) [25]. The project builds an OAIS-based framework and architecture for digital information preservation and demonstrates its validity by preserving cultural, artistic and scientific knowledge. CASPAR also plans active collaboration and teaming with relevant digital preservation initiatives outside the EU, with national and
international projects such as Chronopolis [26], InterPARES [27] and NARA (US National Archives and Records Administration) to bring global dimension to its work.

The CASPAR mission is practical yet ambitious: provide secure, reliable and cost-effective access for digitally encoded information for an indefinite time period. The motivation of the CASPAR project is to define the methodology and infrastructure for digital preservation in Europe. This objective will be pursued by following four guidelines.

- Building a preservation environment based on the OAIS reference model.
- Demonstrating its ability to handle the preservation of the digital resources of diverse user communities.
- Adapting and integrating current state of the art technology in digital preservation.
- Developing technological solutions aimed at sustaining expansion.

The CASPAR three-year project intends to produce a reference implementation for OAIS. CASPAR has a stack of software components, including components to: generate and access AIPs, components to capture, manage and re-construct designated community knowledge, components to extract, index and maintain the metadata, components for directory services, and the archival storage component, which includes the ultimate place of the data. The following figure illustrates a schematic view of the CASPAR system with the Preservation DataStores at the bottom, serving as the storage layer:
Preservation DataStores will be utilized in CASPAR by ingesting and accessing AIPs. During ingest time, the upper layers of CASPAR generate AIPs and then provide it to the Preservation DataStores. For each AIP, the Preservation DataStores assigns or validates a persistent globally unique ID, calculates its fixity and logs the provenance. Then, the AIP is assigned to a cluster that includes AIPs intended for co-location on the same medium. The cluster assignment is performed according to some configured parameters such as clustering by the content information object format and creation date. Each cluster will eventually become a self-contained self-describing object on the media. Finally, metadata that is needed to search for and manage the ingested
AIP is stored in the Preservation Registry and Directory Service which are outside the Preservation DataStores.

During access time, the upper layers of CASPAR and search aids can be used to determine the required AIP. This AIP is then fetched from the Preservation DataStores. The upper layers then unpack the AIP and assist the user in interpreting it.
IBM Offerings and Preservation Data Stores

This section includes a partial list of IBM offerings and research projects related to preservation. DIAS and UVC are focused on digital preservation. The IDS research project is considering adding a preservation focus as well.

DIAS

In 2000, IBM and the National Library of the Netherlands (KB, Koninklijke Bibliotheek) started building an electronic deposit system called Digital Information Archiving System (DIAS), the technical core of the infrastructure for e-Deposit [28] in the Netherlands.

The DIAS [29] provides an open deposit library solution for storing and retrieving electronic documents and multimedia files. It conforms to the OAIS standard and supports physical and logical digital preservation.

The DIAS solution allows the manual and automated ingestion of digital information (assets) into the system. Once an asset is successfully stored, it is managed for preservation and permanent access. Stored assets can be accessed either via a web-based interface (for assets having standard file types) or via a specific work environment on a reference workstation.

IBM constructed DIAS using as many off-the-shelf components as possible, including WebSphere, DB2, Tivoli Storage Manager, and Content Manager.

Figure 9: DIAS System
Preservation DataStores complements DIAS, whose archival storage component can utilize the Preservation DataStores to enhance the solution with preservation aware storage and provide better support for OAIS.

### Universal Virtual Computer (UVC)

The UVC [30] concept was proposed in 2000 by Raymond Lorie of IBM, in a research paper written for IBM, and later published more widely in an article in *RLG DigiNews* [31]. The UVC approach relies partially on emulation concepts and aims to allow digital objects to be retained in their original format along with a program that can decode the data and present it in an understandable form.

The UVC is a simple virtual computer architecture that will run on any existing hardware platform. The UVC is a computer in its functionality. It is virtual because it will never have to be built physically, and it is universal because its definition is very basic and is guaranteed to remain unchanged in the present and future.

Given a UVC interpreter, a program compiled to (or written in) the machine language of UVC is completely independent of the architecture of the computer on which it runs. To access data in the future on a given platform, a UVC interpreter should be written for the hardware/software configuration on which the data will be accessed.

Criticism of emulation has come from David Bearman who comments that “Rothenberg [the original and committed advocate of emulation] is fundamentally trying to preserve the wrong thing by preserving information systems functionality rather than records” [32]. Bearman’s criticism can be leveled at the UVC approach, which would have similar disadvantages.

Preservation DataStores is complementary to UVC and the emulation programs can utilize the Preservation DataStores to enhance the solution with preservation aware storage and provide better support for OAIS e.g. provide fixity computation, support context, handle provenance.

### Intelligent Data Storage (IDS)

Intelligent Data Storage is a research project at the Almaden Research Center. The project aims to create a new generation of scalable archival storage that
integrates storage, file systems, HSM, full-text indexing/search, analytics, retention and compliance. The system is designed to cost-effectively retain large volumes of varied information and facilitate effective usage of the retained information for business advantage. It lowers cost and complexity by integrating the solution stack and system management into an appliance, while supporting multiple APIs. Its main features are:

- Flexibility through multiple different APIs (e.g. NFS, CIFS, JSR 170, TSM API, XAM)
- Integrated full-text search and domain-specific analytics (discovery)
- "Semantics-aware" lifecycle management using the policy engine

The integration of Preservation DataStores and IDS can provide a preservation archival system that will support active short-lived data as well as dormant long-lived data. It will support timely access to the data in the short term as well as generation of OAIS-based AIPs and support migration for the long-term retention of the data.
Related Preservation Frameworks and Approaches

SRB and iRODS

The Storage Resource Broker (SRB) [33] is a data grid technology, developed and owned by the San Diego Supercomputing Center (SDSC). It manages distributed data, enabling the creation of data grids that focus on the sharing of data, and was recently extended to *persistent* archives that focus on the preservation of data. Data grid technology provides the fundamental management mechanisms for distributed data in a scaleable manner. This includes support for managing data on remote storage systems, a uniform name space for referencing the data, a catalog for managing information about the data, and mechanisms for interfacing to the preferred access method. The SRB is a middleware software - it builds on top of standard file systems, commercial archives and storage systems.

Over the past years, SRB has been used as a foundation technology for providing a persistent archive in the context of preservation projects. Many of these projects were commissioned by NARA, the National Archives and Records Administration, and supported by the Library of Congress and NSF. As a persistent archive, it managed the retention of the digital record as well as the context that describes the origin, relevance and authenticity of the record.

SRB focuses more on the bit preservation aspect of the problem. It handles media failures, data mirroring and distribution of data. It stores the data records as files on the storage system and assumes that the data object is packaged into an AIP. The AIP is written to the storage repository but a separate database is used to store the metadata related to the electronic record.

The basic architecture of SRB consists of an SRB server and a Metadata Catalog (MCAT) server. The SRB server exposes various file-like APIs to the application and interacts with the storage system. The MCAT server handles the information stored in the SRB database. The picture below is taken from SRB documents.
The SRB code is currently being re-written in a new system called iRODS, *Intelligent Rule-Oriented Data management System*. It is a rule-based system, which allows the user to make assertions about the rules under which the data is being maintained.
References

1. Xen. See http://www.xensource.com/
3. Recommended Film Retention Policy for Medical Facilities, A Work Product of CRCPD’s H-3 Committee on Medical Practice. See http://www.crcpd.org/WG_Products/Film_Retention_Policy.htm
6. NARA – National Archives and Records Administration. See http://www.archives.gov/
8. PARADIGM. See http://www.paradigm.ac.uk/workbook/oais/index.html


18. Dublin Core Metadata Initiative. See http://dublincore.org/


20. SAFE - Standard Archive Format for Europe. See http://earth.esa.int/SAFE/


30. UVC – Universal Virtual Computer. See 
33. SRB – Storage Resource Broker. See 
http://www.sdsc.edu/srb/index.php/Main_Page