XAM (eXtensible Access Method)

In Digital Libraries and Preservation Stores

MPeterson, June 2009

SNIA

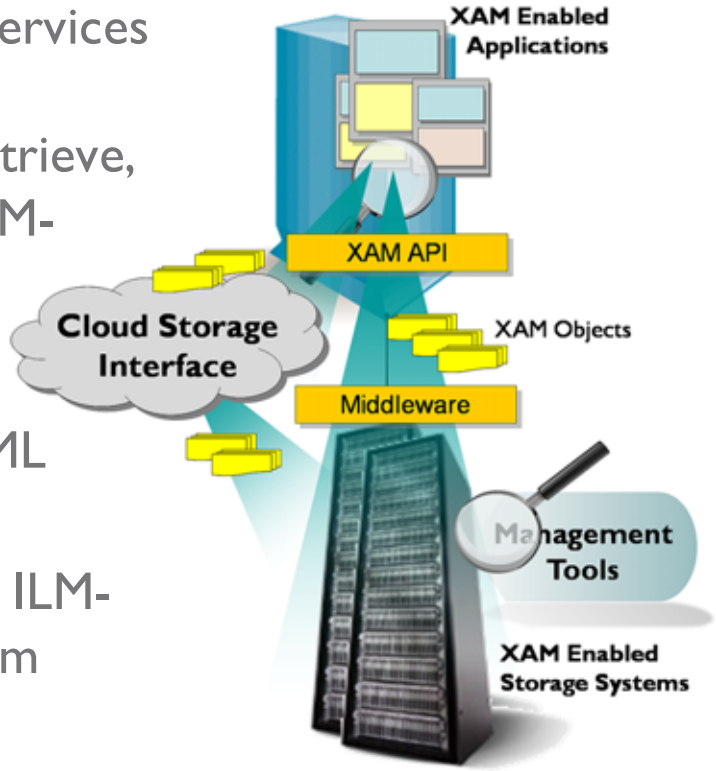
XAM Initiative

Contents

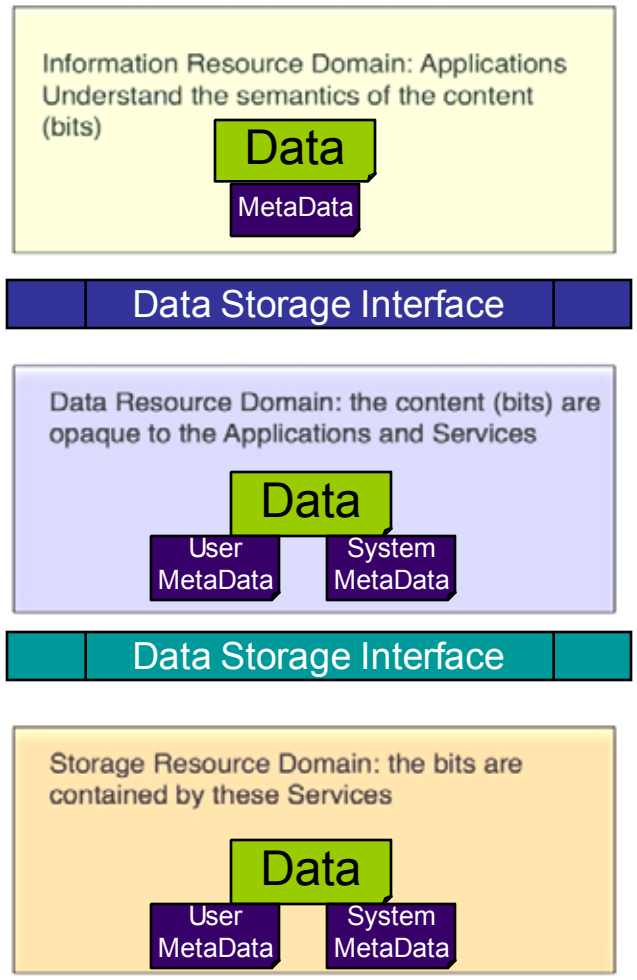
XAM Introduction

SNIA
Advancing storage & information technology

- The open XAM API specification
  - Defines the programming interface and services that enable applications and information management services to define, store, retrieve, search, and manage XAM objects on XAM-enabled storage systems
- XAM objects are:
  - Exportable or importable as standard XML containers
  - Provide for extensible metadata enabling ILM-based management, eDiscovery, long-term retention and preservation
  - Portable, location independent, compliant, secure



XAM is a Data Storage Interface



Resource Domains classify services into specific areas that each deal with a different aspect of the problem

An information domain application creates data and associates metadata with it

Certain Data Storage Interfaces can accommodate both Data and MetaData (XAM, Filesystems with extended attributes)

MetaData aware Data Services interpret Data System MetaData as the requirements for it's lifecycle and implement policies for retention, placement, lifecycle, etc.

Other Data Storage interfaces (based on blocks or objects) provide virtualized Containers for the Data bits and the management of those containers

Storage services are employed to meet those requirements at this point in the data's lifecycle, however the storage services are unaware of the data's requirements

XAM Value Proposition

SNIA
Advancing storage & information technology

- XAM provides choice, flexibility, and makes information portable
  - Choice between storage devices and applications, flexibility to change vendors or use storage devices from multiple vendors
- XAM reduces costs
  - Reduces application porting, testing, and support costs, and improves utilization efficiencies & time to market
- XAM empowers ILM-based management practices
  - Retention, disposition, security, and other requirements can be embedded in the metadata allowing application-based and external services to act on XAM objects automatically based on business rules
  - Reduces cost and improves storage utilization efficiencies

XAM Value Propositions

SNIA
Advancing storage & information technology

- XAM metadata is the foundation for extended e-Discovery, analytics, and litigation support capabilities
- A standard container enabling centralized, shareable, and controlled medical and health care records
- Retention and Preservation systems utilizing XAM provide a standard preservation object format
- XAM provides cloud-based services with missing information management control capabilities

Digital Libraries and Preservation Stores

XAM Value Proposition

May 2009

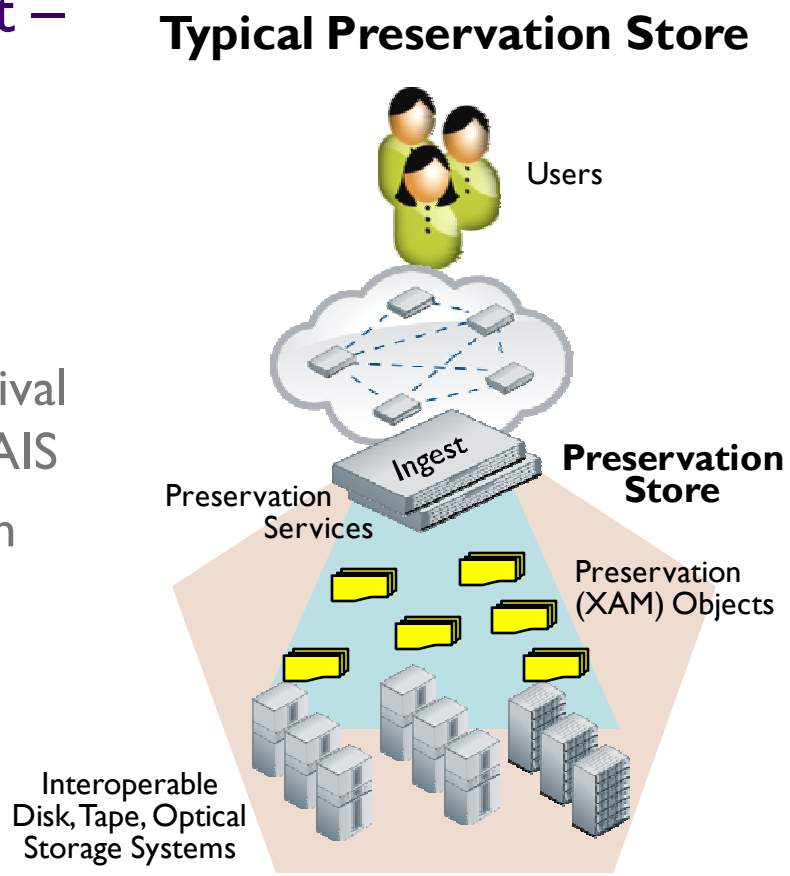
Digital Libraries & Preservation

SNIA
Advancing storage & information technology

- Top Long-term Digital Information Retention Challenges
  - Ever growing Cost
  - Logical & Physical Migration
  - Scaling to keep up with growth
  - Metadata Management
  - Maintaining Access, Security, Confidentiality, Authenticity, Audit logs, and other preservation services over long periods of time
  - Location and storage independence

How XAM Applies

- XAM defines a preservation object – it is the container, the access method, and the application to storage interface
  - A standard container (preservation object) that can function like an Archival Information Package as defined by OAIS
  - Reduces logical and physical migration challenges



Digital Libraries and Preservation Stores

SNIA
Advancing storage & information technology

Vision of Preservation with XAM

- Digital preservation systems that utilize XAM enjoy storage interoperability, resulting in reduced physical migration costs
- Preservation objects based on XAM are produced during the ingestion process, resulting in increased portability, control, interchangeability, and usable life of digital information at lower operating costs

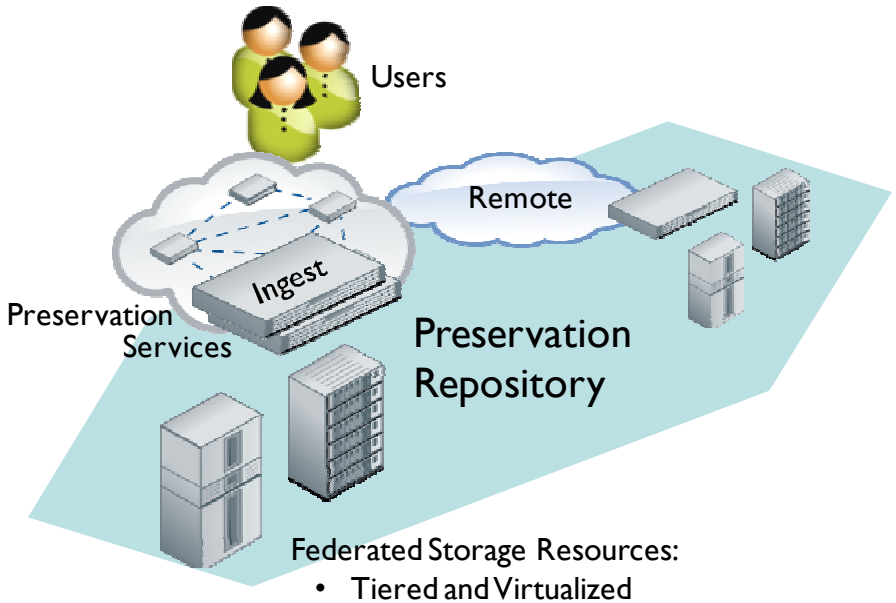
Benefits of Preservation with XAM

- Digital preservation practices become more scalable, cost effective, automated, and able to extend the life of digital information
- XAM objects are self-contained, portable, searchable, location independent preservation objects
- All critical preservation services are supported – authenticity, integrity, availability, discoverability, portability, confidentiality, audit logs, ILM-base management, migration, …
- Retention, Deletion, and Preservation practices can operate automatically based on business rules
- Migration and management costs are greatly reduced

XAM Use Case

- A preservation service ingests content sent to it by client systems
  - During ingestion, XAM objects are created with metadata based on policies
  - Objects are hashed, audit logs started, indexed for discovery, and storage policies initiated based on classification rules
  - ILM-based practices can operate automatically in accordance with business rules

Typical Preservation Store



Federated Storage Resources:
- Tiered and Virtualized
- Distributed
- Multiple vendors
- Multiple architectures
- Interoperable

XAM Value Proposition

SNIA
Advancing storage & information technology

Digital Libraries and Preservation Stores

- Cost:
  - XAM is open source, without proprietary lock-in, directly reduces the cost of migration and of metadata maintenance over time because of its extensibility and portability
- Addresses Logical Migration issues and its cost
  - Makes preservation objects portable - storage & location independent
  - Potential to integrate readers into the container, supporting long-term readability and reducing the need and cost of logical migration
  - Accommodates extended metadata needed for audit logs, authenticity, integrity, security, chain of custody, and future requirements

XAM Value Proposition

SNIA
Advancing storage & information technology

Digital Libraries and Preservation Stores

- Reduces Physical Migration frequency and cost
  - Creates storage system interoperability
  - Portable, application independent objects can be tiered and moved across self-healing storage systems with no migration needed
- Aids in dealing with the metadata management problem
  - Extended metadata tracks audit trail providing for maintenance of integrity, authenticity and chain of custody even through transformations
  - Rich metadata supports enhanced eDiscovery and information repurposing capabilities

XAM Value Propositions

SNIA
Advancing storage & information technology

- XAM provides Retention and Preservation systems additional benefits:
  - Enables digital preservation practices with the ability to contain expanded metadata in the preservation object
    - Potential to embed readers in the preservation object, reducing the cost and impact of periodic logical migration
    - Supports federated search across XAM aware repositories including the metadata
  - XAM bridges multiple data silos because the preservation objects are independent of storage and portable

SNIA and XAM

SNIA
Advancing storage & information technology

- For more information on XAM
  - SNIA XAM Home
    - http://www.snia.org/xam
- Contact information:
  - Michael Peterson, SNIA Chief Strategy Advocate
    (805) 201-3178 or mpeterson@snia-dmf.org

XAM BRIEFING

MP1

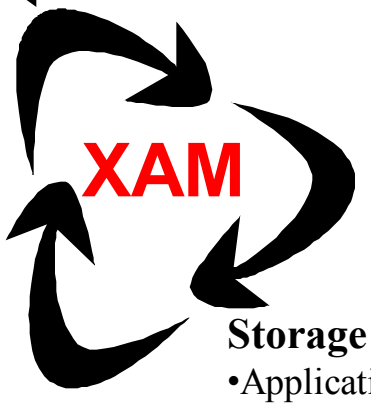
Why XAM?

SNIA
Advancing storage & information technology

The industry will benefit from a standardized access method to storage systems

End Users want:
- Choices between Application Vendors
- Choices between Storage Vendors
- Easy migration between vendors/technology
- Compliance, Scalability, Performance, $/GB, TCO

Application Vendors want:
- Annotate Data with associated Metadata
- Indicate basic Storage Management Policies
- Speak same language to all types of Devices
- Manipulate billions if not trillions or "records"



Storage Vendors want:
- Application Support for their Products
- Efficiently Store Application Data and Metadata
- Integrate Basic Storage Management Capabilities
- Manage billions if not trillions of "records"

Slide 18

MP1 Needs updating with the new value props
Michael Peterson, 5/13/2009

A New Storage API

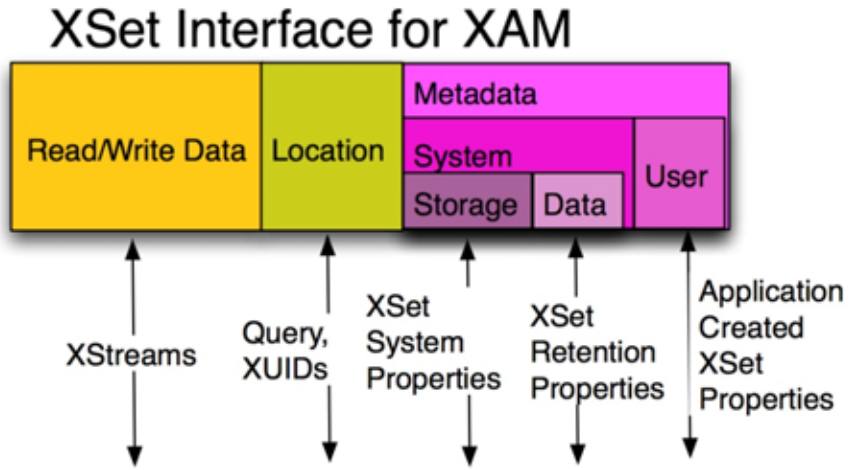
SNIA
Advancing storage & information technology

- XAM Provides:
  - Interoperability: Applications can work with any XAM conformant storage system; information can be migrated and shared
  - Compliance: Integrated record retention and disposition metadata,
  - ILM Practices: Framework for classification, policy, and implementation
  - Migration: Ability to automate migration process to maintain long-term readability
  - Discovery: Application-independent structured discovery avoids application obsolescence.
- XAM is a SNIA Architecture
  - The XAM Architecture spec defines the normative semantics of the API for use by applications and implementation by storage systems
- XAM is an Application Programming Interface (API)
  - The XAM Java and C API specs define bindings of the XAM Architecture to the Java and C Languages
- XAM is SNIA Software – open source
  - The XAM SDK provides a common library and reference implementation to promote widespread adoption of the standard

XAM Interface

- First interface to standardize system metadata for retention of data
- Implements basic capability to Read and Write Data (through Xstreams)
- Locates any XSet with a query or by supplying the XUID
- Allows Metadata to be added to the data and keeps both in an XSet object
- Uses and produces system metadata for each Xset
- For example Access and Commit times (Storage System Metadata)
- But it also uniquely specifies Data System Metadata for Retention Data Services

- XAM User metadata is un-interpretable by the system, but stored with the other data and is available for use in queries
- Given this we can see that XAM is a data storage interface that is used by both Storage and Data Services (functions)



XAM System Metadata

SNIA
Advancing storage & information technology

- XAM specifies *property fields* that are interpreted by an Xsystem as System Metadata
- It does this by using a reserved field namespace
  - .xam.*, .xsystem.* and .xset.* are reserved
- Either the Content Application or a separate Information Management application provides preservation and/or ILM metadata to cause the underlying system to treat the data according to the specified requirements
  - i.e. Retention

Data Storage Interface Standards

SNIA
Advancing storage & information technology

- Other standard data storage APIs have the ability to deal with metadata as well (POSIX filesystems)
- POSIX specifies standard system metadata as part of the data storage interface:
  - File times, Permission (including ACLs), owner, group, etc.
- This metadata is maintained and used (interpreted) by the storage services that implement the API
  - Thus we call it ***storage system metadata***
- The functions that are controlled by this metadata govern the storing and retrieval of the data through the interface
- These functions are described in the abstract as ***storage services***

Metadata

Storage services may provide functions for metadata as part of the data storage interface.

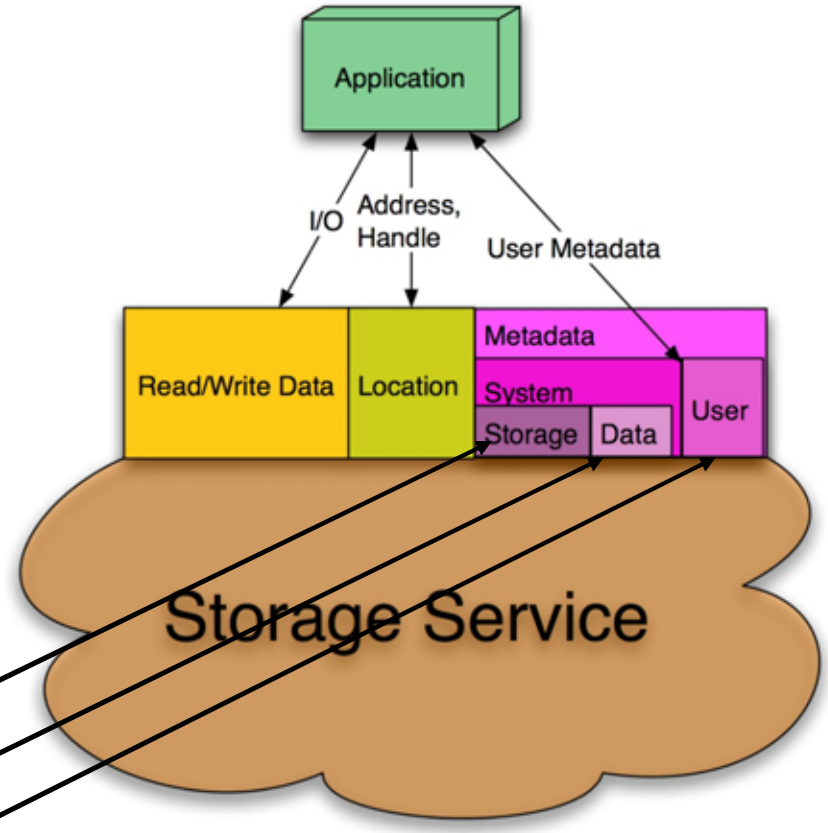
- This is an important capability for managing Data Resources (as opposed to managing Storage Resources).

The metadata may be managed by the storage service, managed by data services, or un-interpreted by either.

- System metadata that is managed by storage services are those properties of a data element that pertain to the primary functions of storing and retrieving the data.

We call this *storage system metadata*, as it is used and managed by storage services.

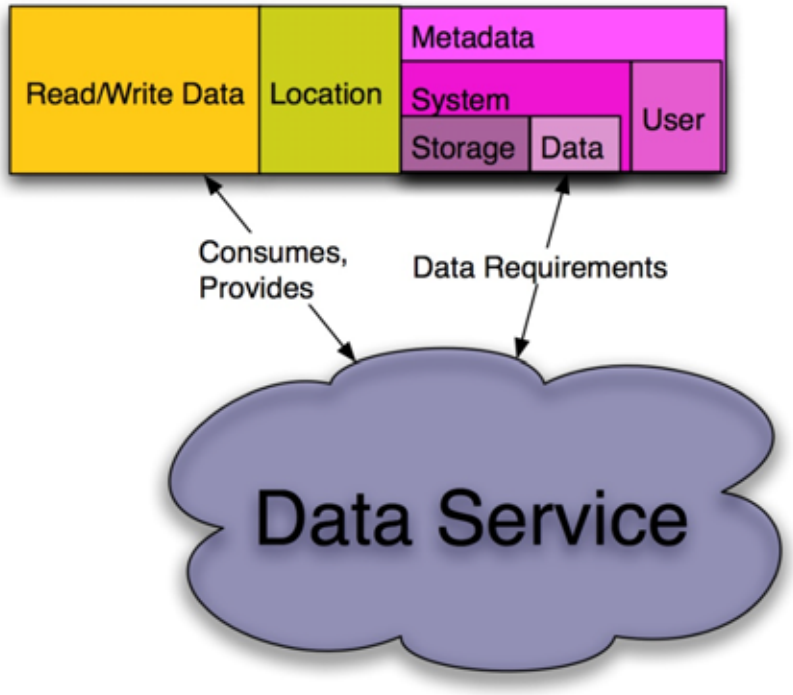
- Other system and user metadata may be preserved on the basis of individual data elements, but is not interpreted by the storage services



Metadata for Data Services

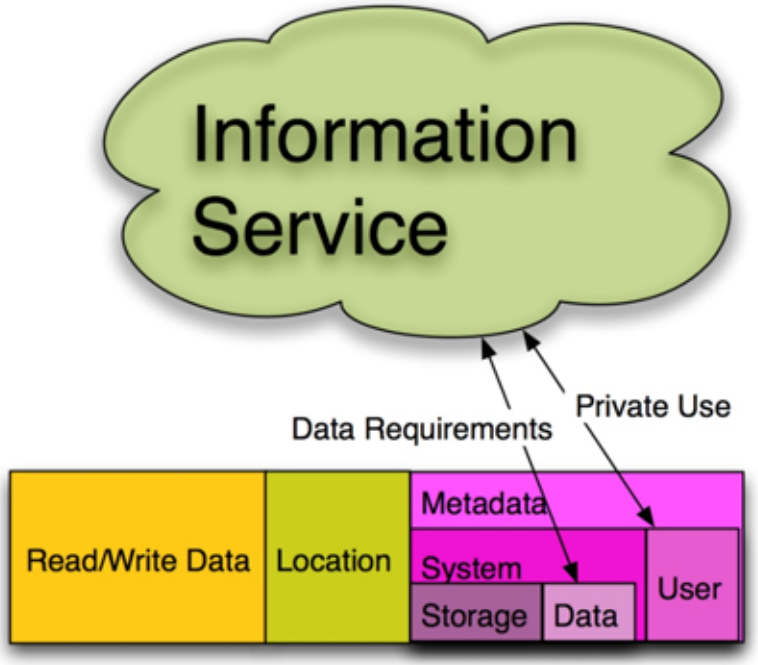
SNIA
Advancing storage & information technology

- Metadata available through the data storage interface may also be managed by data services
- This data service metadata can be used by data services to provide differentiated value to individual data elements
- The model or schema for data service metadata may be defined by each data service and may be standardized



Metadata in Information Services

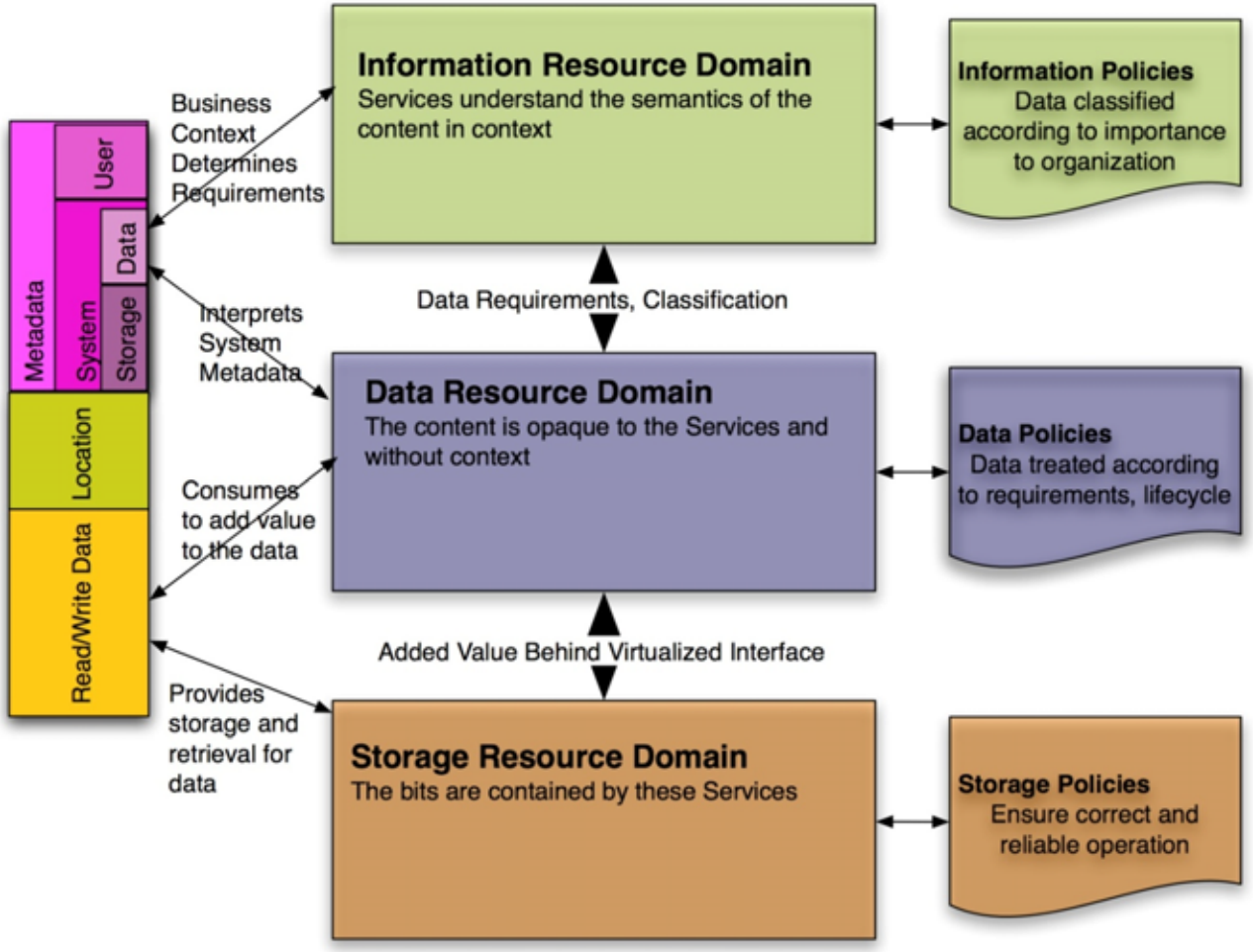
- The role of metadata in information services is as a communication mechanism with the underlying storage services and data services.
- Information services are primarily concerned with the data service system metadata as a means to convey the data's requirements to the underlying data services.
- An information service may also interpret user metadata for purposes of data classification.
- An information service can create its own user metadata that is un-interpreted by the underlying services for its own use.



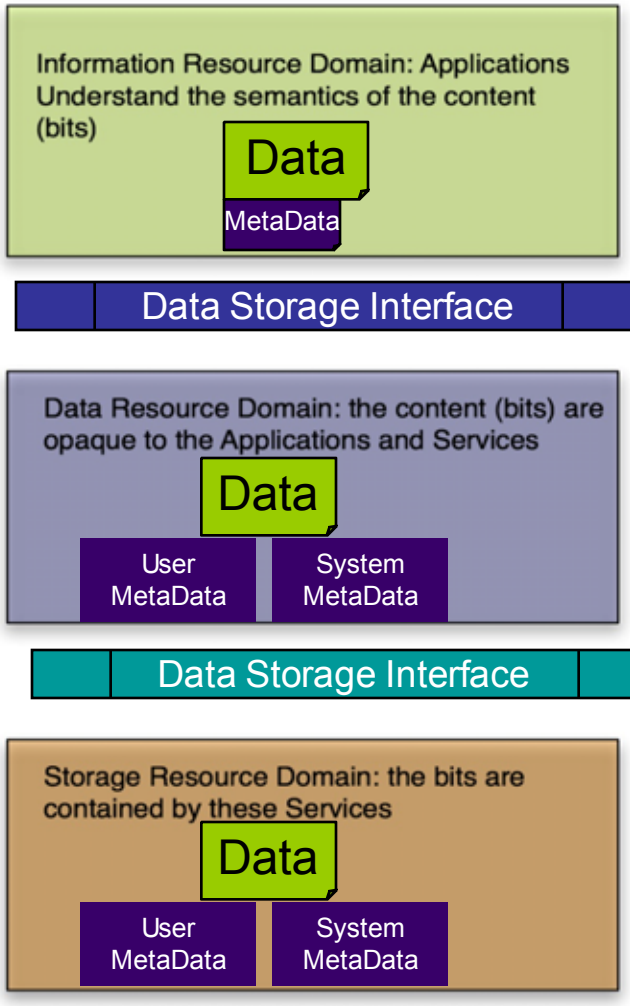
The Resource Domain Model

SNIA
Advancing storage & information technology

- This model shows the logical layering of the different domains and the role of policies for each domain. The services in each domain play a different role, but leverage common, standard interfaces



Resource Domains



Resource Domains are a way of classifying services into specific areas that each deal with a different aspect of the problem

An information domain application creates data and associates MetaData with it

Certain Data Storage Interfaces can accommodate both Data and MetaData (XAM, Filesystems with extended attributes)

MetaData aware Data Services interpret Data System MetaData as the requirements for it's lifecycle and implement policies for retention, placement, lifecycle, etc.

Other Data Storage interfaces (based on blocks or objects) provide virtualized Containers for the Data bits and the management of those containers

Storage services are employed to meet those requirements at this point in the data's lifecycle, however the storage services are unaware of the data's requirements

SIRF BRIEFING

Self-Contained Information Retention Format (SIRF)

SNIA
Advancing storage & information technology

- A logical container format appropriate for long-term retention and preservation
  - A logical data format of a mountable unit e.g. a filesystem, a block device, a stream device, an object store, a tape, etc.
  - Includes a cluster of "interpretable" preservation objects that can be understood in the future
  - Self-describing – can be interpreted by different systems
  - Self-contained – all data needed for the preservation objects interpretation is contained within the preservation objects cluster
    - If a mountable unit is damaged or lost, the effect is contained – the information in the other mountable units is still valid !
- Reduces logical and physical migration costs and complexities

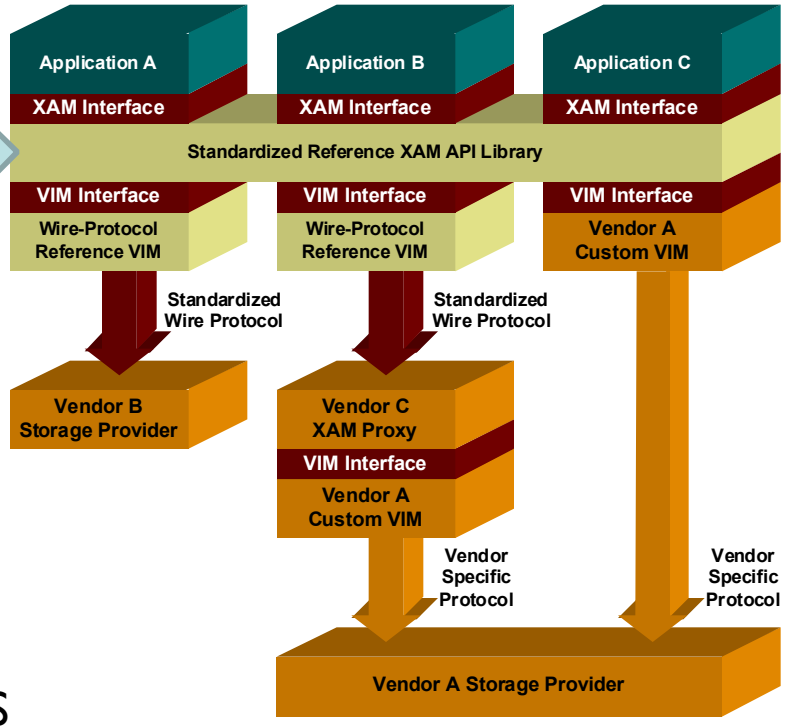
SIRF Objectives

SNIA
Advancing storage & information technology

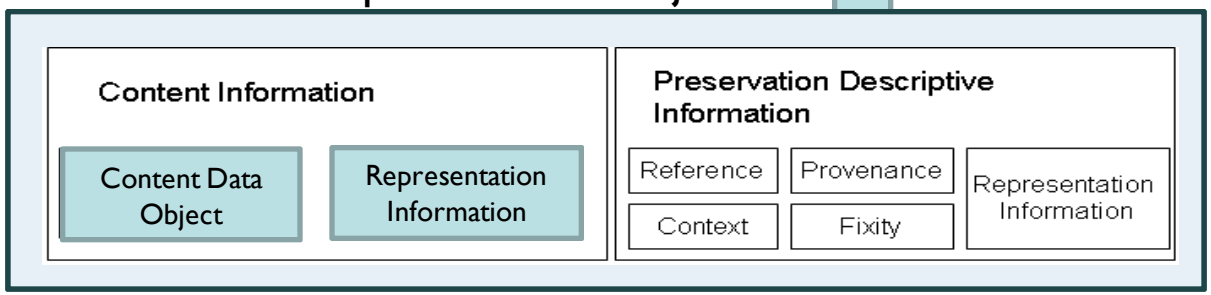
- Facilitate transparent logical and physical migration and movement in order to support long term preservation
  - Media, subsystem or bitstream movement – remove the mountable unit from system A and put it at system B.
  - Transparent – system A is not involved. All the information needed for system B to understand the mountable unit is self-described and self-contained within the mountable unit.
  - Long term readability –embed readers to facilitate logical migration
  - Preservation – sustain the understandability and usability of the data and not just the bits.
- Implementations of SIRF utilize:
  - the Open Archival Information System (OAIS) ISO standard
  - SNIA's eXtensible Access Method (XAM)

Example of XAM Extensibility

- **XAM Library includes a SIRF format container**
  - **Self-Describing, Self Contained Data Format**
- **Applications can now write a standard long-term portable interchange format**



SIRF defines a preservation object based on OAIS

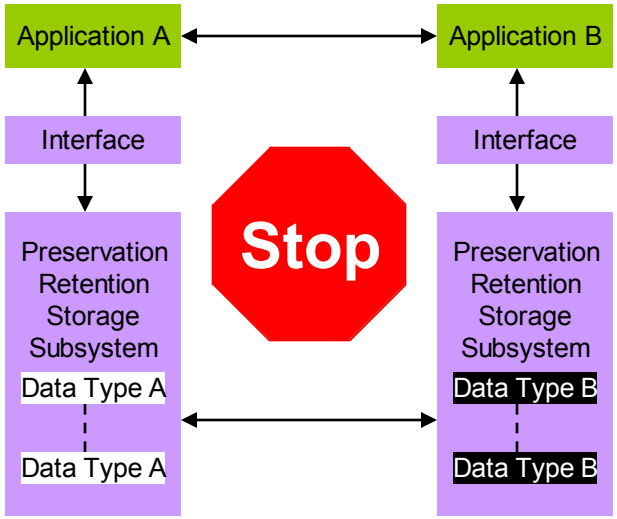


Even 'readers' can be inserted into the preservation object overcoming logical migration barriers

Problem SIRF is Addressing

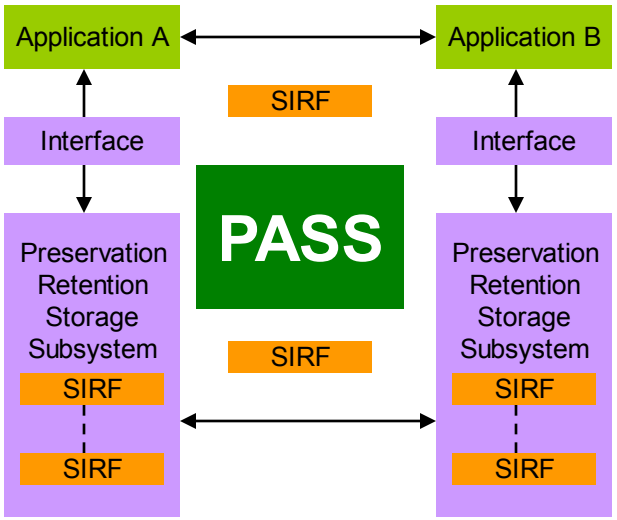
SNIA
Advancing storage & information technology

Without SIRF



- Cannot move cluster of preservation objects between systems by itself
- Only the original application who wrote the preservation objects can read and interpret them
- Utilize export and import processes
- Preservation Objects cannot be sustained over the long-term

With SIRF



- Can move cluster of preservation objects between systems by itself
- Any SIRF compliant application can read and interpret the preservation objects
- No need for export and import processes
- Preservation Objects can survive longer

XAM Introduction

SNIA
Advancing storage & information technology

eXtensible Access Method (XAM) API - An Application Programming Interface (API) standard between applications and retention-managed storage systems

- The XAM API specification defines a consistent interface and set of capabilities between the application and XAM-compliant storage systems
- XAM also defines the structure of the XAM object as stored

The XAM Software Development Kit (SDK)

- Provides the XAM API specification and a set of development tools (libraries, documentation, and sample code) that allows applications to communicate with a XAM-compliant Storage Systems

The SNIA is advancing the XAM Specification through the SNIA Architecture and ANSI / ISO standard process