SNIA DEVELOPER CONFERENCE

SDC 24

BY Developers FOR Developers

September 16-18, 2024
Santa Clara, CA

CDMI-enabled Web Applications

Standards-based Multi-protocol Namespace Management

Presented by David Slik

Presentation Agenda

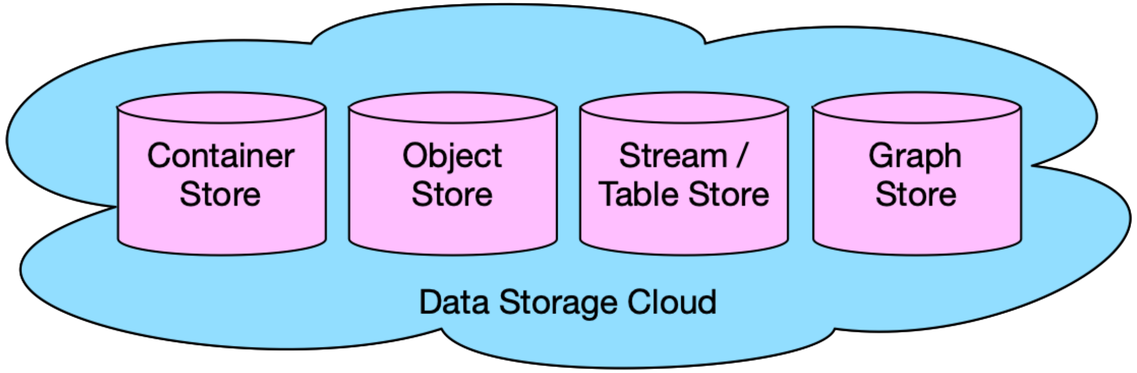
- Cloud Data Management Interface Refresher
- CDMI as a Superset Representation
- CDMI-enabled Web Applications

Cloud Data Management Interface Refresher

A Quick Overview of CDMI

CDMI Refresher

- The Cloud Data **Management** Interface helps clients manage data stored in clouds
- CDMI solves three problems:
  1. How do we support rapid cloud innovation?
  2. How do we generalize managing this data?
  3. How do we provide a standard management interface for clients?



CDMI Refresher

- Clouds provide value-added data storage services and features.
- The SNIA Storage Industry Reference Domain Model (SIRDM) categorizes these services into storage services, data services, and information services:
  - Some are standardized
  - Some are app-specific
  - Some are vendor-specific
  - Some are experimental

Storage Services, e.g.

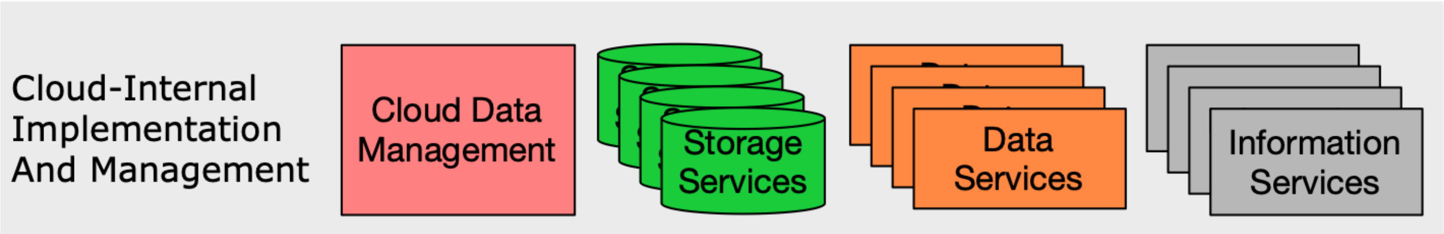
- Basic data storage
- Naming
- Metadata
- Access Control

Data Services, e.g.

- Data protection
- Data integrity
- Data versioning
- Data retention

Information Services, e.g.

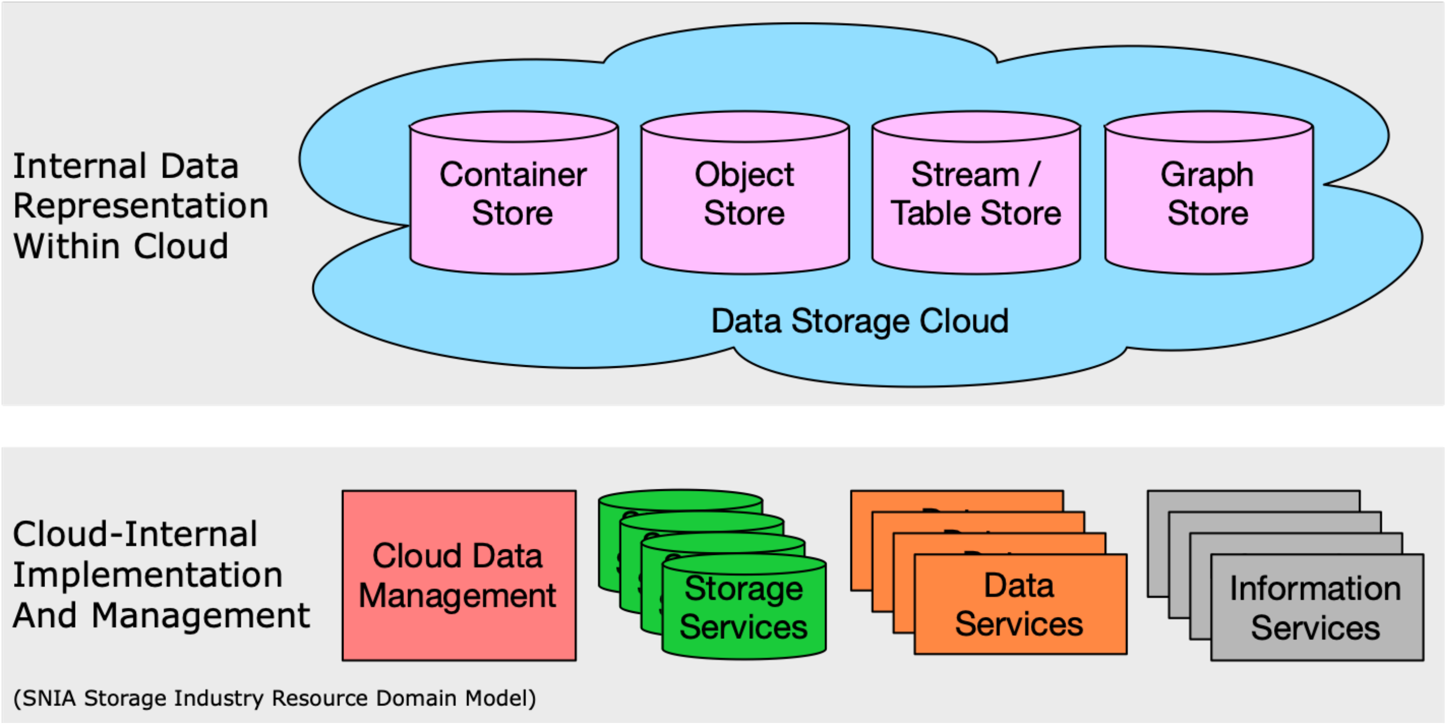
- Data transformation
- Feature extraction
- Automated tagging
- Triggers and FaaS



(SNIA Storage Industry Resource Domain Model)

CDMI Refresher

- CDMI leaves the internal data representation, internal implementation, and internal data management up to the cloud.
  - This enables cloud vendors to innovate, solving problem #1



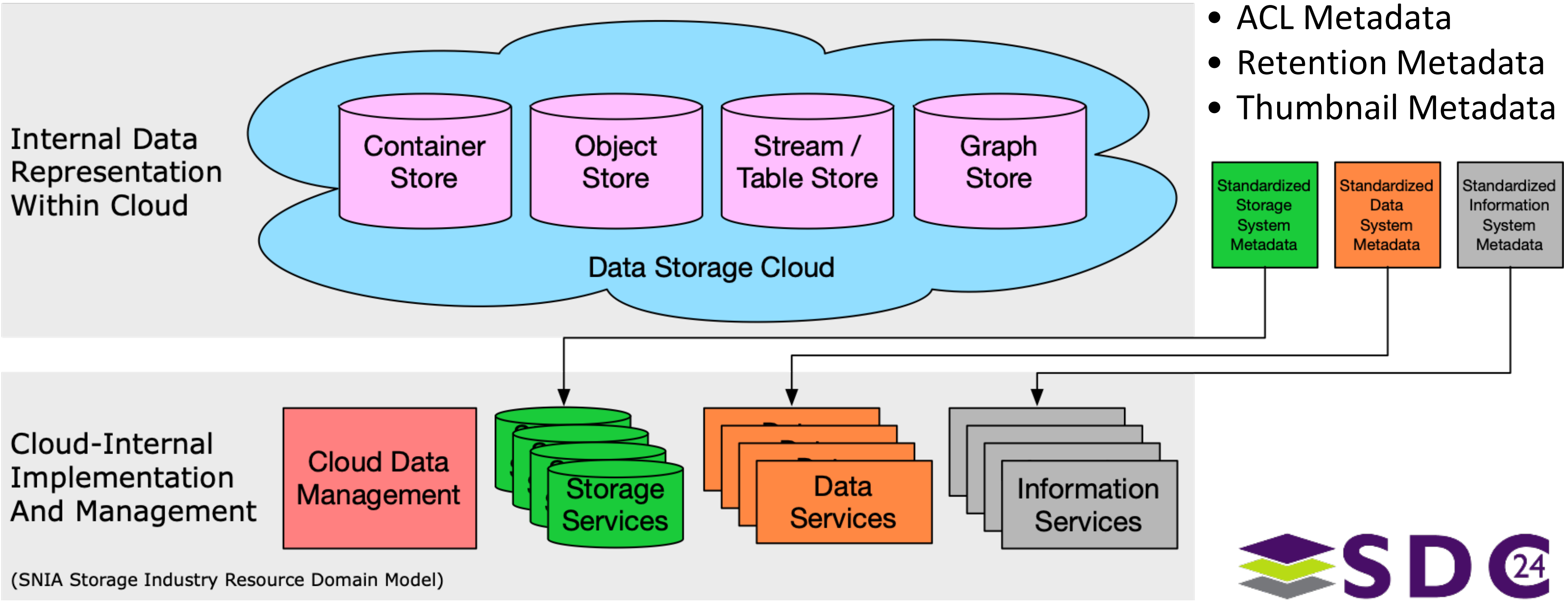
(SNIA Storage Industry Resource Domain Model)

CDMI Refresher

- To enable interoperability, CDMI defines metadata to declaratively specify which services should be used, and how they should be used.
  - This generalizes managing data and data services, solving problem #2.
  - This metadata is extensible and flexible.

For Data XYZ:

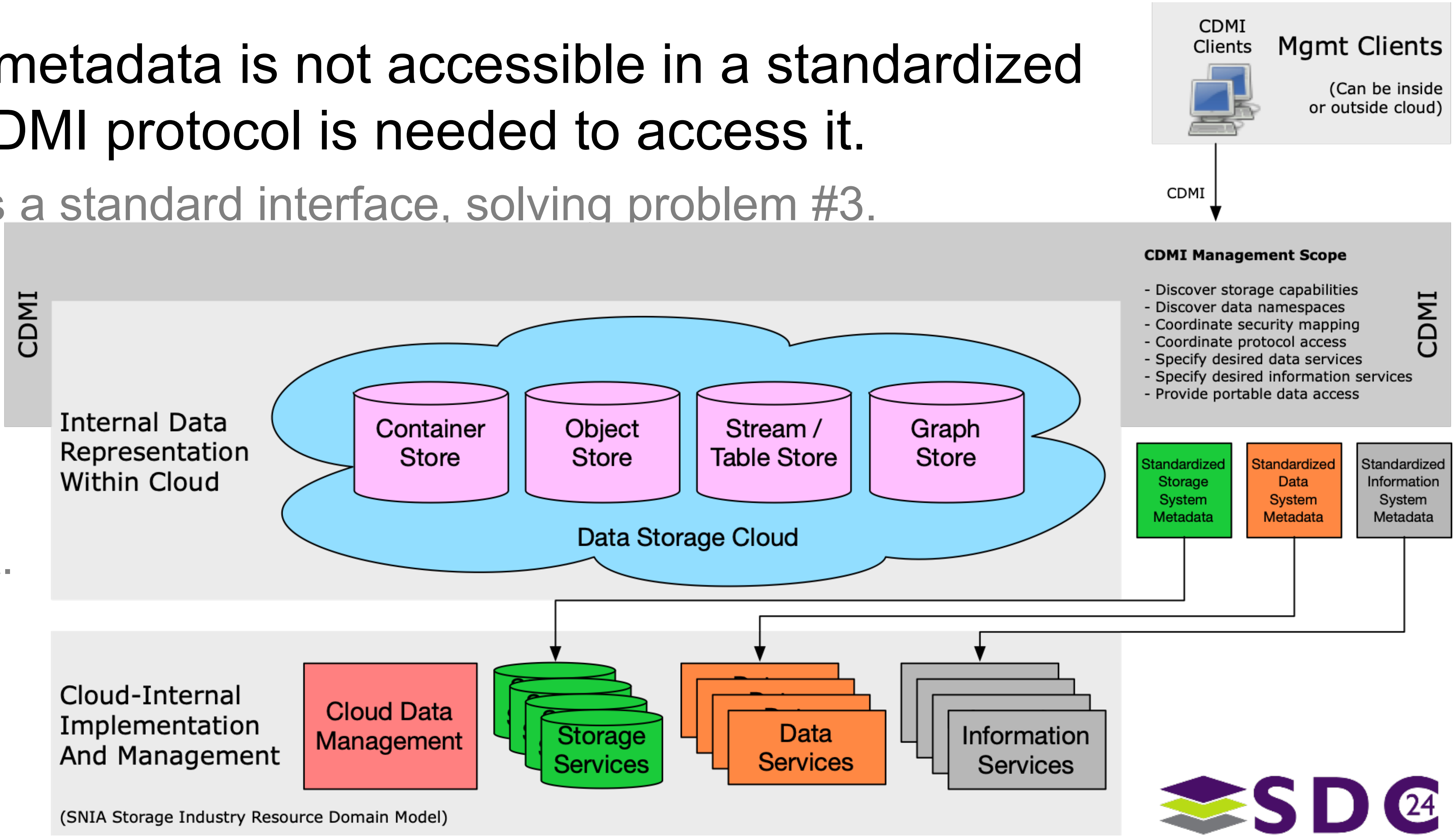
- ACL Metadata
- Retention Metadata
- Thumbnail Metadata


Internal Data Representation Within Cloud
Container Store
Object Store
Stream / Table Store
Graph Store
Data Storage Cloud
Standardized Storage System Metadata
Standardized Data System Metadata
Standardized Information System Metadata
Cloud-Internal Implementation And Management
Cloud Data Management
Storage Services
Data Services
Information Services

(SNIA Storage Industry Resource Domain Model)

CDMI Refresher

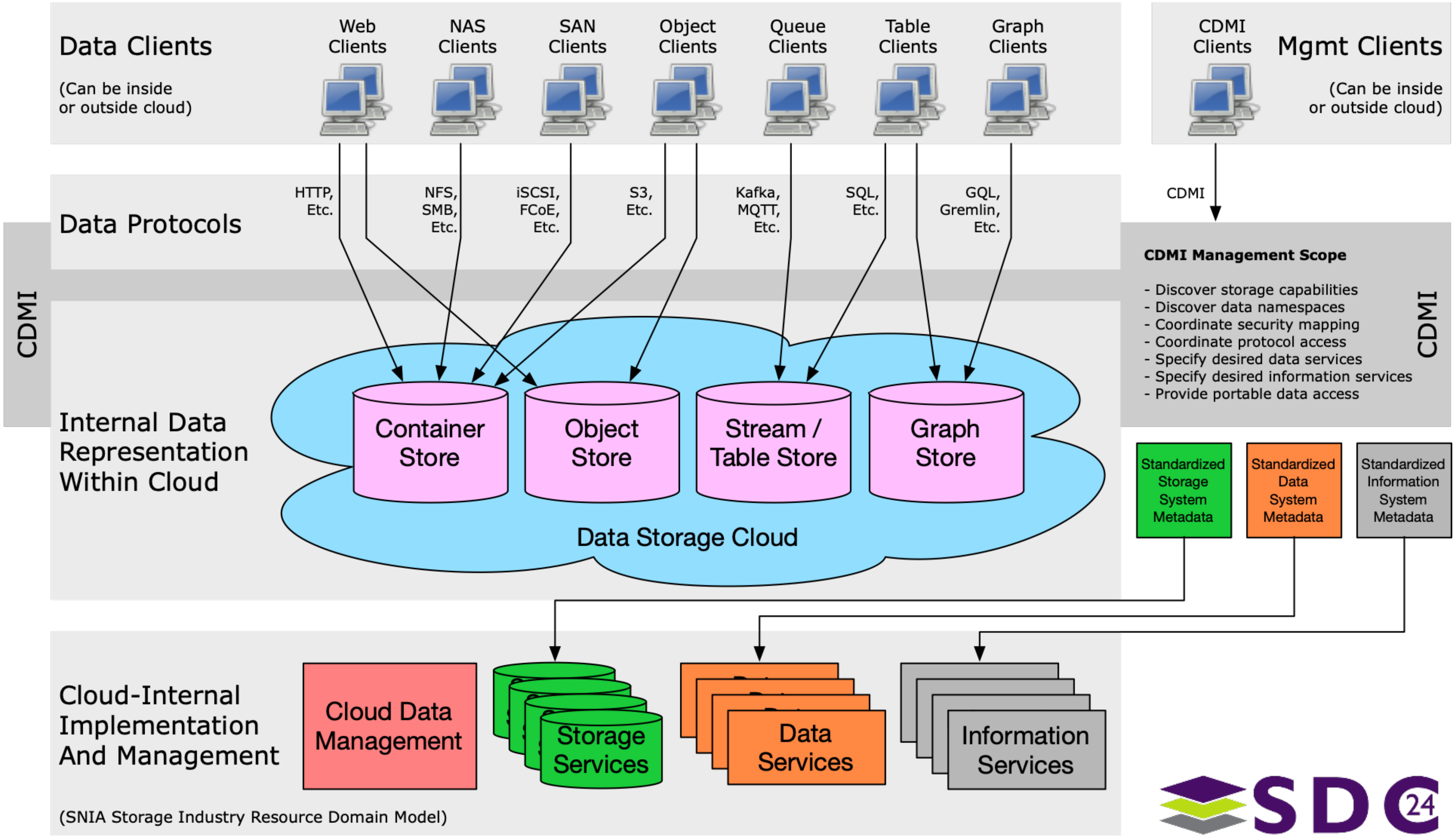
- Management metadata is not accessible in a standardized way, so the CDMI protocol is needed to access it.
  - This provides a standard interface, solving problem #3.
  - This enables namespace discovery and access for metadata management.


CDMI Clients
Mgmt Clients
(Can be inside or outside cloud)
CDMI
CDMI
CDMI
CDMI Management Scope
- Discover storage capabilities
- Discover data namespaces
- Coordinate security mapping
- Coordinate protocol access
- Specify desired data services
- Specify desired information services
- Provide portable data access
Internal Data Representation Within Cloud
Container Store
Object Store
Stream / Table Store
Graph Store
Data Storage Cloud
Standardized Storage System Metadata
Standardized Data System Metadata
Standardized Information System Metadata
Cloud-Internal Implementation And Management
Cloud Data Management
Storage Services
Data Services
Information Services

(SNIA Storage Industry Resource Domain Model)

CDMI Refresher

- CDMI access is separate from data access.
- Complements data access protocols.
- Manages data access protocols via exports and imports.



Data Clients
(Can be inside or outside cloud)
Web Clients
NAS Clients
SAN Clients
Object Clients
Queue Clients
Table Clients
Graph Clients
CDMI Clients
Mgmt Clients
(Can be inside or outside cloud)
Data Protocols
HTTP, Etc.
NFS, SMB, Etc.
iSCSI, FCoE, Etc.
S3, Etc.
Kafka, MQTT, Etc.
SQL, Etc.
GQL, Gremlin, Etc.
CDMI
CDMI
CDMI Management Scope
- Discover storage capabilities
- Discover data namespaces
- Coordinate security mapping
- Coordinate protocol access
- Specify desired data services
- Specify desired information services
- Provide portable data access
CDMI
Internal Data Representation Within Cloud
Container Store
Object Store
Stream / Table Store
Graph Store
Data Storage Cloud
Standardized Storage System Metadata
Standardized Data System Metadata
Standardized Information System Metadata
Cloud-Internal Implementation And Management
Cloud Data Management
Storage Services
Data Services
Information Services

(SNIA Storage Industry Resource Domain Model)

CDMI Refresher

CDMI DEMO #1

This demonstration illustrates the following CDMI capabilities:

1. For data accessible from a cloud, use CDMI to view the corresponding management metadata

CDMI as a Superset Representation

LUNs and Files and Objects, Oh My!

CDMI as a Superset Representation

- One of CDMI's goals is to be able to represent any type of data:

Type	Organization	Identifier	Parent	Access	Metadata
File	Linear array of bytes	Name	Directory	Path	Yes
Directory	Collection of resources	Name	Directory	Path	Yes
Queue	Zero or more values	Name or ID	Sometimes	Varies	Sometimes
LUN	Linear array of bytes	ID	Sometimes	IQN, etc.	Sometimes
Object	Linear array of bytes	Key or ID	Sometimes	ID	Yes
Bucket	Collection of objects	Name	No	Name	Yes
Table	Set of structured rows	ID	Sometimes	Name	No
Graph	Object with relationships	Varies	Many	Varies	Sometimes

- This requires being able to express these variations

CDMI as a Superset Representation

- CDMI specifies two primary views into data: Objects and Containers
  - Data objects have values, which can be strings (e.g. files), arrays (e.g. queues), or structured data (e.g. tables)
  - Containers have children and parents (e.g. directories), and typed links (e.g. graphs)
- Data may be accessed simultaneously via different representations:
  - Exporting a file in a filesystem as a LUN (e.g. VMDK or ISO files)
  - Exporting a LUN as a filesystem (e.g. “mounting” a LUN)
  - Exporting an S3 bucket as a file system
  - Exporting a filesystem as a S3 bucket
- Some representations may have values and children simultaneously!

CDMI as a Superset Representation

- Simultaneous value and children example:
  - S3 simulates a hierarchical file system using object names (keys).
  - An S3 bucket may contain the following four objects:
    - “/"
    - “/files”
    - “/files/”
    - “/files/readme.txt”
  - Each of these objects can have a value!
    - In a file system, “/” and “/files/” represent directories, and do not have a value
    - In a file system, “/files” and “/files/” cannot exist simultaneously.
- CDMI must represent data even if a given protocol forbids it.

CDMI as a Superset Representation

- For S3, this is currently being standardized in the SNIA Cloud Storage Technical Working Group:
  - Each URI has a CDMI “data object” representation that focuses on the value, if present
  - Each URI has a CDMI “container” representation that focuses on the parents/children, if present
- Both of these representations are for the same underlying data item.
- CDMI defines how these are mapped when exported using different protocols.

CDMI as a Superset Representation

CDMI DEMO #2

This demonstration illustrates the following CDMI capabilities:

1. Show object and container representations for a file
2. Show object and container representations for a directory

CDMI-enabled Web Applications

Where a container is a filesystem is a web site is an application

CDMI-enabled Web Applications

- We can now put together CDMI's ability to manage exports and manage data/container duality to enable the deployment and management of web applications:
  - Web applications consist of files (Javascript, HTML, CSS, images, etc)
  - CDMI enables file systems to be exported via HTTP
  - CDMI enables directories to have values
  - Web applications can be launched from a directory URI
    - E.g. "https://localhost/myApplication/

- Since containers can be exported as file systems (and via HTTP), this provides an easy way to manage deployment of web applications.

CDMI-enabled Web Applications

CDMI DEMO #3

This demonstration illustrates the following CDMI capabilities:

1. Exporting a file system via HTTP
2. Setting the value of a directory
3. Launching a web application

CDMI-enabled Web Applications

Why this is of value:

1. We have demonstrated how CDMI provides a standardized protocol for exporting local and cloud-resident data.
2. We have demonstrated how CDMI exports facilitate multi-protocol access to local and cloud-resident data.
3. We have demonstrated how CDMI exports simplify deployment of web applications.

These are all significant challenges for storage systems, and CDMI can help!

Questions and Discussion

And don’t forget to join us in the SNIA Cloud Storage Technical Working Group if you found this work interesting!

Please take a moment to rate this session.

Your feedback is important to us.