International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol.1, Issue.2, pp-648-652 ISSN: 2249-6645

Cloud Data Management Interface for Multiple Cloud Platforms

Mr. Pranav Dadlani[1], Ms. Kanika Dubey[2], Prof. Meeta Kumar[3]
(Department of Computer , Maharashtra Institute of Technology's College of Engineering (MITCOE), Pune)

ABSTRACT
There are various resources that are offered as services by different clouds. CDMI (Cloud Data Management Interface) focuses on the data storage service offered by various clouds. We propose a model for an interface that can be mapped to the data storage offering of multiple clouds. This specification defines an interface for interoperable transfer and management of data in a multiple cloud storage environment. This interface provides the means to access multiple cloud storage and to manage the data stored there.

Keywords **- Clustering, DaaS, Information resource management, Interoperability, Information Storage and Retrieval, Information Storage, Information Search and Retrieval,.**

.

I. INTRODUCTION
Cloud computing has grown from being a promising business concept to one of the fastest growing segments of the IT industry. In the current economic climate with its increasing competition, companies must respond to changes faster and more effectively than ever before. In today's world, to put your ideas into action in the computational world you need applications. Business apps have always been too expensive. Users wish to access Internet services over lightweight portable devices rather than through some descendant of the traditional desktop PC
Cloud computing is a recent trend in IT that moves computing and data away from desktop and portable PCs into large data centers. Its major goal is reducing the cost of IT.

Cloud-service clients will also be able to add more capacity at peak demand, experiment with new services and remove unneeded capacity.

1.1 Taxonomy for Cloud Computing
1.1.1 Service consumer
The service consumer is the end user or enterprise that actually uses the service, whether it is Software, Platform or Infrastructure as a Service.

1.1.2 Service provider
The service provider delivers the service to the consumer. The actual task of the provider varies depending on the type of service:

- For Software as a Service, the provider installs, manages and maintains the software.
- For Platform as a Service, the provider manages the cloud infrastructure for the platform, typically a framework for a particular type of application.
- For Infrastructure as a Service, the provider maintains the storage, database, message queue or other middleware, or the hosting environment for virtual machines.

Crucial to the service provider's operations is the management layer. At a low level, management requires metering to determine who uses the services and to what extent, provisioning to determine how resources are allocated to consumers, and monitoring to track the status of the system and its resources. At a higher level, management involves billing to recover. costs, capacity planning to ensure that consumer demands will be met.
Security applies to all aspects of the service provider's operations. A well-rounded set of standards simplify operations within the provider and interoperability with other providers.

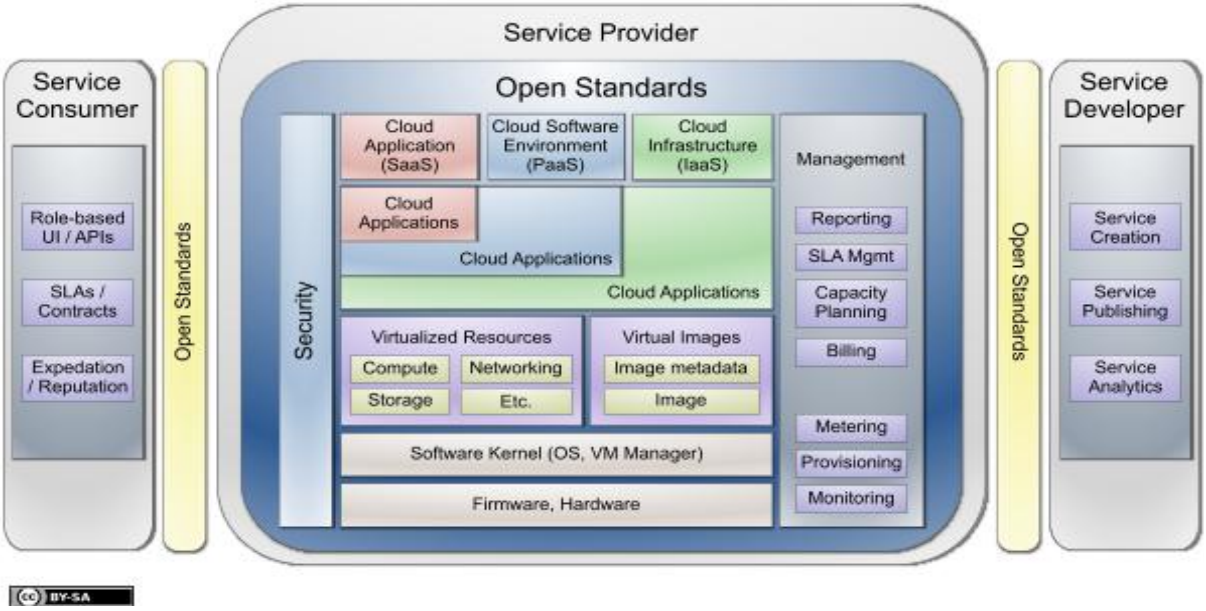


Fig1: In this diagram, service consumers use the services provided through the cloud, service providers manage the cloud infrastructure and service developers create the services themselves

1.1.3 Service Developer

The service developer creates, publishes and monitors the cloud service. These are typically "line-of-business" applications that are delivered directly to end users via the SaaS model. Applications written at the IaaS and PaaS levels will subsequently be used by SaaS developers and cloud providers

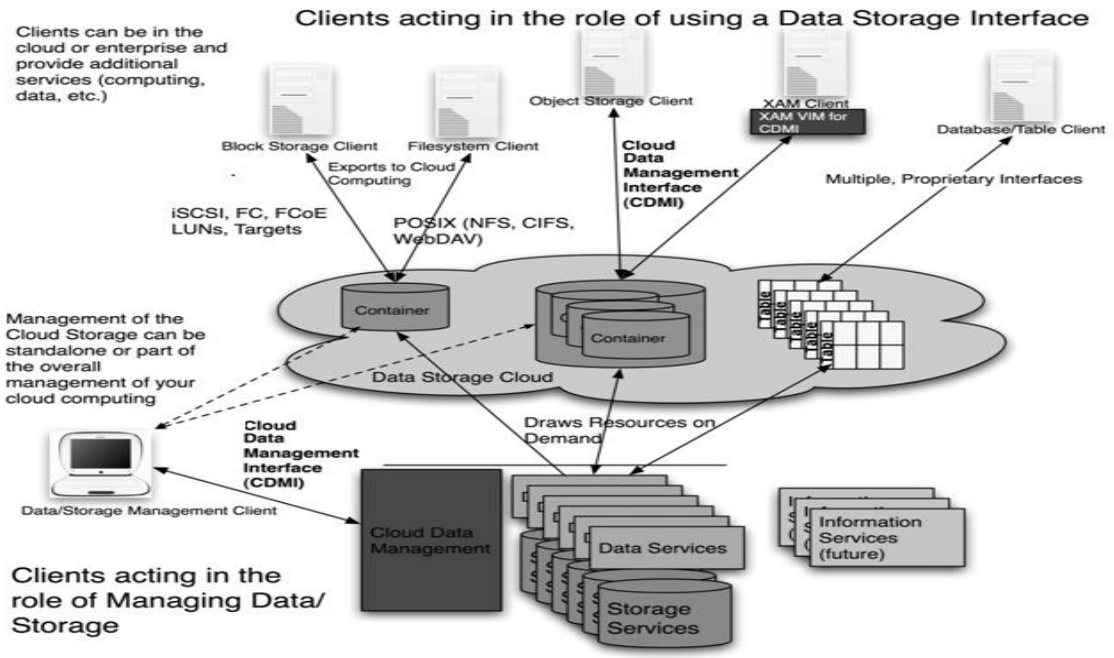
2. CLOUD STORAGE

Cloud storage is a subset of cloud computing. The appeal of cloud storage is due to some of the same attributes that define other cloud services: pay as you go, the illusion of infinite capacity (elasticity), and the simplicity of use/management. It is therefore important that any interface for cloud storage support these attributes, while allowing for a multitude of business cases and offerings, long into the future.

Thus, cloud storage is simply the delivery of virtualized storage on demand. The formal term or this is Data Storage as a Service (DaaS).

2.1 Data storage as service

- By abstracting data storage behind a set of service interfaces and delivering it on demand.
- The difference between the purchase of a dedicated appliance and that of cloud storage is not the functional interface, but merely the fact that the storage is delivered on demand. The customer pays for either what they actually use, or in other cases, what they have allocated for use.
- We abstract the underlying storage space exposed by these interfaces using the notion of a container. A container is not only a useful abstraction for storage space, but also serves as a grouping of the data stored in it and a point of control for applying data services in aggregation. idea is to determine what features of the website encourage the desired result. "Fallout analysis," a subset of path analysis, looks at "black holes" on the site, or paths that lead to a dead end most frequently, paths or features that confuse or lose potential customers.



This model shows multiple types of cloud data storage interfaces that are able to support both legacy and new applications. All of the interfaces allow storage to be provided on demand, drawn from a pool of resources. The capacity is drawn from a pool of storage capacity provided by storage services. The data services are applied to individual data elements, as determined by the data system metadata. Metadata specifies the data requirements on the basis of individual data elements or on groups of data elements (containers).

3. NEED FOR INTEROPERABILITY

With the presence of numerous vendors, the need is emerging for interoperability between clouds so that a complex and developed business application on clouds is interoperable.

Every new cloud service provider have their own way on how a user or cloud application interacts with their cloud leading to cloud API propagation . This kills the cloud ecosystem by limiting cloud choice because of vendor locking, portability, ability to use the cloud services provided by multiple vendors including the ability to use an organization's own existing data center resources seamlessly. There is a need for complex developed business applications on the clouds to be interoperable. Cloud adoption will be hampered if there is not a good way of integrating data and applications a cross clouds.

If cloud computing is to move beyond the hype cycle, vendors need to put aside their differences and agree on common principles related to security and the interoperability of cloud platforms.

4. CDMI

The Cloud Data Management Interface (CDMI) may be used to create, retrieve, update, and delete objects in a cloud. The features of the CDMI include functions that:

- Allow clients to discover the capabilities available in the cloud storage offering,
- Manage containers and the data that is placed in them, and
- Allow metadata to be associated with containers and the objects they contain.

CDMI may also be used by administrative and management applications to manage containers, domains, security access, and monitoring/billing information, even for storage that is functionally accessible by legacy or proprietary protocols. The capabilities of the underlying storage and data services are exposed so that clients may understand the offering.

5. CDMI METADATA

CDMI uses many different types of metadata, data system metadata, user metadata, and storage system metadata.

Data system metadata is metadata that is specified by a CDMI client and is a component of objects. Data system metadata abstractly specifies the data requirements associated with data services that are deployed in the cloud storage system.

User metadata is arbitrarily-defined JSON strings that are specified by the CDMI client and is a component of objects. The namespace used for user metadata names is self-administered (e.g., using the reverse domain name) and user metadata names shall not begin with the prefix "cdmi_".

Storage system metadata is metadata that is generated by the storage services in the system (e.g., creation time, size) to provide useful information to a CDMI client.

6. IMPLEMENTATION PLAN

The project mainly consists making a common interface (drivers) which can be used to access different clouds , loading the required APIs for the required cloud(s) , performing all user operations and lastly a convenient GUI for the users to use our system with ease.

1. **Common Interface:**
   a. Initiator :To load the interface and authenticate the user on the interface
   b. Loader : Loads cloud APIs to connect to the cloud
   c. Drivers
   d. Simulator : performs file handling and database browsing
   e. Terminator: logs out and shuts down device
   f. Database: Data base has 4 parts:

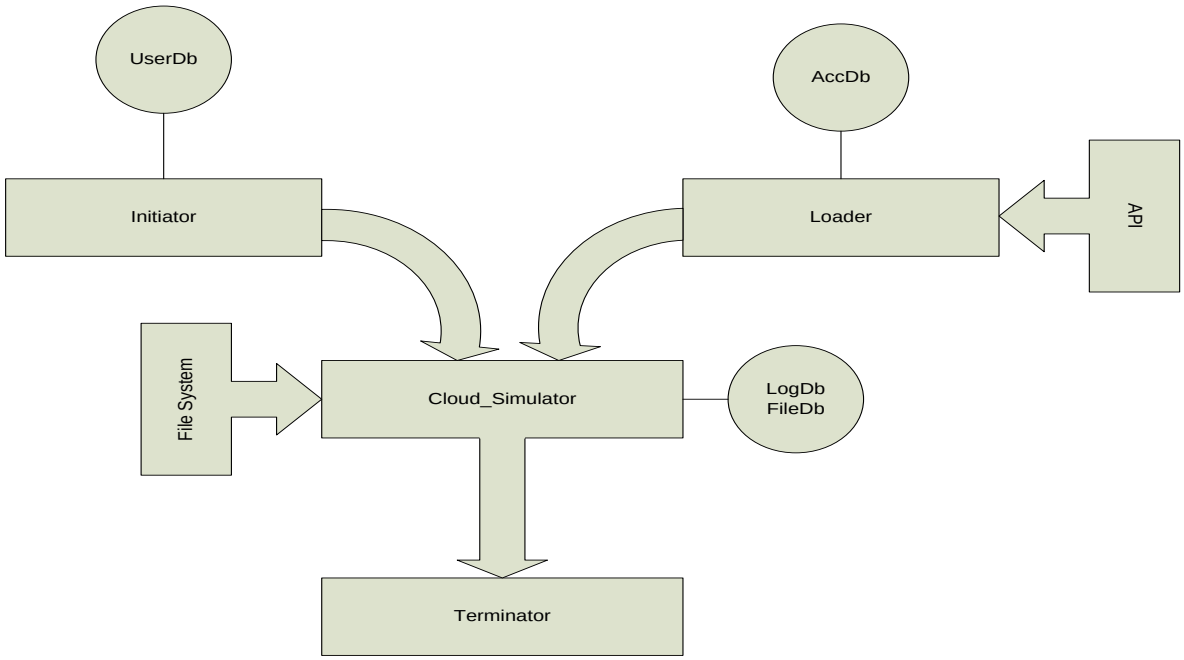
      1. File database : It is used to store the unique file ids of files uploading.

      2.Logs database : It is used to store information of where (which cloud and container) every file is located.

      3. Account database : It is used to store the details (uid,pwd) of every account of every user (on different clouds).

      4.User database: It stores the username and password for the software.

2. **GUI:** The GUI should be well designed for the user's convenience using any standard front end language.



6.1Block Diagram

The Interface mainly consists of 4 parts:

1. The Initiator: The Initiator is responsible to load the interface on the screen and authenticate the user on the interface. It checks for the username and password for the interface user stored in the UserDb (User database) and loads the respective log files and account details.

2. Loader: The function of the loader is to load the required API's to connect to the respective clouds. It uses the AccDb(Account database) to retrieve the user's account details.

3.Cloud Simulator: The cloud simulator simulates the cloud data on the interface. It shows all the files/folders/containers that are stored on the cloud. It browses the cloud and is useful for performing CRUD functionality. It is connected to the user's file system to enable user to upload data or retrieve it.

4.Terminator: The terminator is required to shut down the interface properly. It logs the user out of the interface for security purpose.

7. FUTURE ENHANCEMENTS

- Mobile Portability-Can be extended for Mobile Operating systems like Symbian Anna ,Windows Mango, Android and iOS
- Integrating Future Clouds –Clouds which are not yet integrated into the interface as well as those which yet in existences can be added in the future.

8. CONCLUSION:

Interoperability has had a huge impact on the cloud adoption and usage and thus the industry is witnessing high amount of energy and thrust towards these from different stakeholders viz., users, vendors and standard bodies. By developing common interface ,one can increase and accelerate the adoption of cloud computing as users will have a wider range of choices in cloud without vendor lock-in, portability and ability to use the cloud services provided by multiple vendors. This will also include the ability to use an organization's own existing data center resources seamlessly. It further promises to help towards complexly developed business applications on the cloud to be interoperable and ensure data and application integration across clouds. It also provides business opportunities to users to choose and use services provided by many different cloud vendors based on various criteria. Cdmi will thus pave the way towards realizing the true potential/benefits of cloud computing.

REFERENCES

THESES

[1]By A V Parameswaran and Asheesh Chaddha "Cloud Interoperability and Standardization" Infosys Paper 2009

[2] Mark Carlson "CDMI Specification for Developers "

[3]Deni Connor, Patrick H. Corrigan and James E. Bagley Client Relations: Phylis Bockelman "Cloud Storage: Adoption, Practice and Deployment "

BOOKS

[4]Williams, Ross, "A Painless Guide to CRC Error Detection Algorithms", Chapter 16, August 1993, http://www.repairfaq.org/filipg/LINK/F_crc_v3.html

JOURNALS

[5] Jiehui JU, Jiyi WU, Jianqing FU, Zhijie LIN "A Survey on Cloud Storage "
Journal of Computers, Vol 6, No 8 (2011), 1764-1771, Aug 2011
doi:10.4304/jcp.6.8.1764-1771

[6] Ronald L. Krutz, Russell Dean Vines "Cloud Security: A Comprehensive Guide to Secure Cloud Computing"