Storage Networking Industry Association
Technical White Paper

Multi-cloud White Paper

Version 1.2 - October, 2022

Abstract: As storing data in the cloud has become ubiquitous and mature, many organizations have adopted a multi-cloud strategy. Eliminating dependence on a single cloud platform is quite a compelling case with benefits of increased reliability, availability, performance, and the avoidance of vendor lock-in and/or specific vendor vulnerabilities, to name a few.

SNIA Cloud Data Management Interface (CDMI™) v2.0 is an ISO/IEC standard, ISO/IEC 17826. CDMI enables cloud solution vendors to meet the growing need of interoperability for data stored in the cloud by providing a storage abstraction layer that insulates the application from the underlying cloud provider's interfaces which allows an application to be easily used with multiple clouds.

Abstract

Eliminating dependence on a single cloud platform is quite a compelling case with benefits of increased reliability, availability, performance, and the avoidance of vendor lock-in and/or specific vendor vulnerabilities to name a few.

But multi-cloud environments are not without challenges. Taking advantage of the benefits without increasing complexity requires a strategy that ensures applications are not tightly coupled to cloud-specific technologies. Supporting a storage abstraction layer that insulates the application from the underlying cloud provider's interfaces allows an application to be easily used with multiple clouds. It allows storage features specific to a cloud to be exposed in a standardized manner and enables data to be transparently accessed and migrated as needed in order to take advantage of cloud-specific features without the application being aware of the underlying mechanics, thus reducing or eliminating the limits and vulnerabilities of any one cloud. What is missing is a comprehensive description of where interoperability is needed and where standards can best be applied.

Why use multiple clouds?

Risk reduction

Avoid lock-in

Applications that are built on technology or features only provided by a single cloud are tightly coupled to that cloud provider. Moving to another cloud requires development work that may be expensive or take a long time to complete. If features are not present on other clouds, that technology may need to be re-implemented within the application, or functionality that depends on that technology may need to be removed before moving to another cloud.

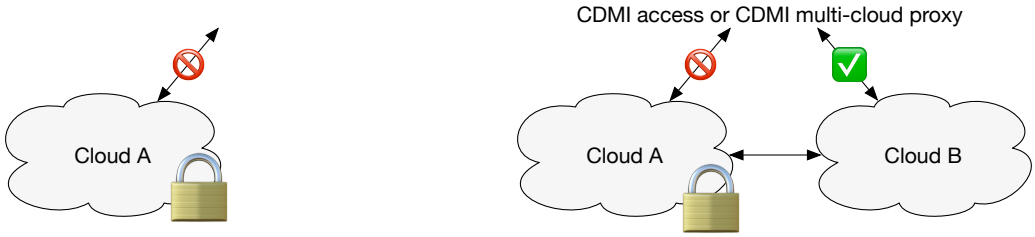

Figure 1: Avoid single-cloud lock-in

Supporting a storage abstraction layer that insulates the application from the underlying cloud provider's interfaces (such as using a CDMI proxy server that communicates with multiple clouds, or by using CDMI proxies on each cloud) allows an application to be used with multiple clouds, thus reducing or eliminating the cost to exit any given cloud. This also mitigates risk by ensuring that applications are not tightly coupled to cloud-specific technologies.

Avoid single points of failure

Applications that are built on a single cloud are dependent on the reliability and availability of that cloud. By spanning more than one cloud, reliability, availability and performance can be increased.

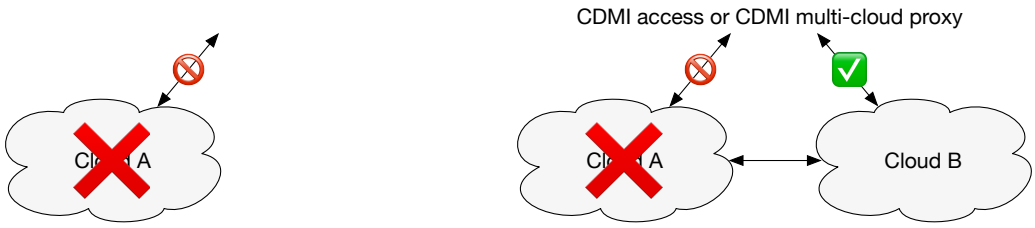

Figure 2: Avoid single-point of failure

Supporting a storage abstraction layer allows an application to store and access data across multiple clouds, and the placement of data can be transparently changed without having to change the application. This allows greater control over the cost vs. reliability, availability, and performance tradeoffs.

Political, regulatory and compliance restrictions

Regulatory, compliance and political constraints are constantly evolving, which quickly can change which clouds are viable options for organizations.

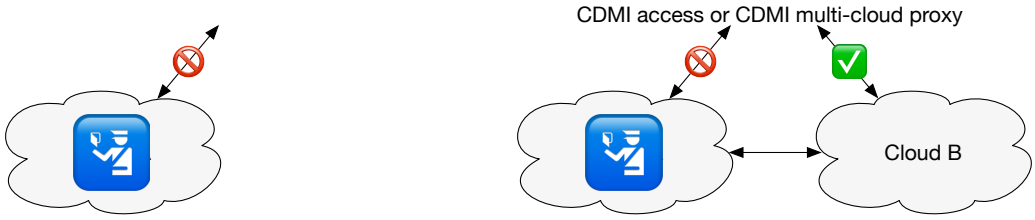

Figure 3: Avoid Regulatory Compliance Violations

Supporting a storage abstraction layer allows data to be evacuated or moved from cloud to cloud transparently without requiring changes to the application. For example, data governed by different regulations can be placed in different clouds that are compliant with those regulations, and for data where cloud storage is not acceptable, the data can be stored on-premises.

Cost reduction

Operational cost reduction

Applications that are built on a single cloud can only use service offerings made available by that cloud provider. By selecting services offered by multiple cloud providers, a lower cost service offering that meets the needs of the application may be available on one cloud that is not available on another cloud.

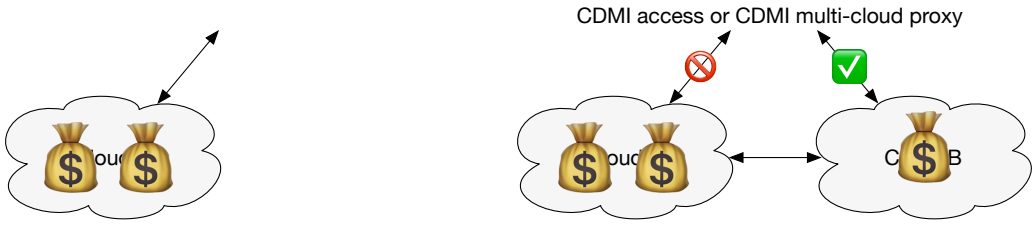

Figure 4: Select lowest cost services

Supporting a storage abstraction layer allows data to be placed on storage services that most closely meet the desired service level objective (SLO), thus minimizing overpaying for SLO levels that are not required. Data can also be transparently moved from cloud service to cloud service if pricing and SLOs change over time (when egress costs are less than the potential savings).

Exit cost reduction

Applications that are built on a single cloud are unable to stop using that cloud. By supporting multiple clouds, an application can choose to migrate services to an alternate cloud, or choose to stop using services on a given cloud, without disrupting application operation.

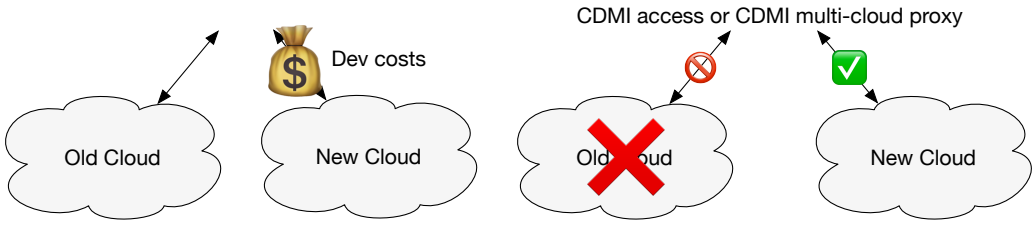

Figure 5: Lower cost to switch cloud providers

Container orchestration solutions, such as Kubernetes, enable workload packaging and mobility between clouds. A multi-cloud storage abstraction layer will make such workload migrations even more transparent.

Supporting a storage abstraction layer allows data movement to evacuate (or archive) data stored on a given cloud to be transparent to the application. It also allows data movement to be sourced from the most cost-effective location.

Running work in parallel across clouds

Applications that are built on a single cloud have to process data within the same cloud where the data is stored. By supporting multiple clouds, data can be packaged and spread across multiple clouds to take advantage of computing costs, available computing resources and types of computing resources, including running processing in multiple clouds in parallel. However, this approach does not work well for large datasets where the cloud-resident application must access the full dataset.

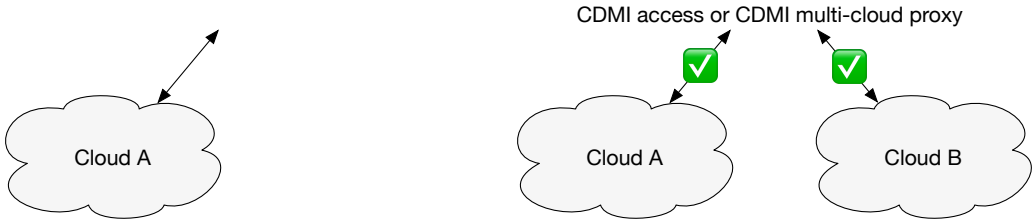

Figure 6: Run workloads in multiple clouds

Supporting a storage abstraction layer allows collections of data to be stored together on the same cloud, so that computing performed on that cloud can be performed against locally stored data. An abstraction layer can also provide transparent access to remotely stored data for computing that is running in any given cloud, and can transparently migrate data when computing moves from cloud to cloud.

Hybrid cloud / Burst to cloud enablement

Supporting a storage abstraction layer allows uniformal data access on-premises and in the cloud. This may ease burst to cloud when on-premises capacity is exhausted, or to allow remote batch execution using cloud resources.

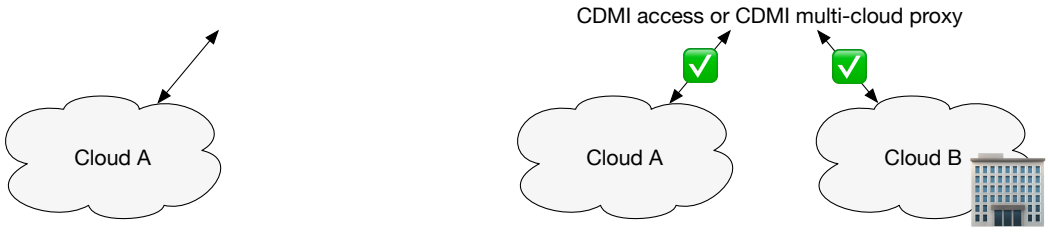

Figure 7: Run workloads across on-prem as well as public clouds

Access to features/performance

Access to cloud-specific features

Applications that are built on a single cloud are limited to the specific features provided by that cloud. By supporting multiple clouds, features that are present only a given cloud can still be taken advantage of, without having to use that cloud exclusively.

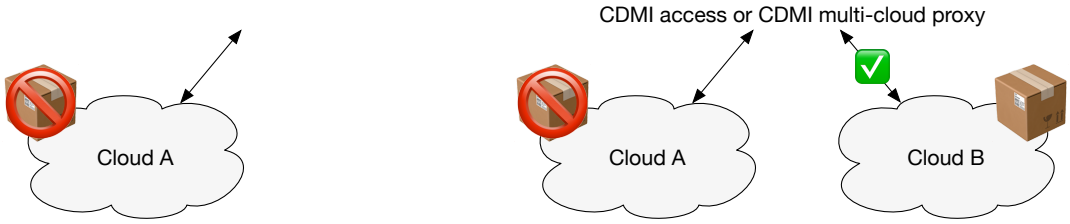

Figure 8: Access features not supported by all clouds

Supporting a storage abstraction layer allows storage features specific to a cloud to be exposed in a standardized manner and allows data to be transparently accessed and migrated as needed in order to take advantage of cloud-specific features without the application being aware of the underlying mechanics.

Latency reduction

When storing the same file in multiple clouds, a simultaneous request can be made for the file from each of those clouds. The response will vary for each cloud, but the first response can be used to satisfy the request. The other requests can be aborted to save egress charges. This also can be hidden behind a proxy CDMI server such that applications need not do anything different.

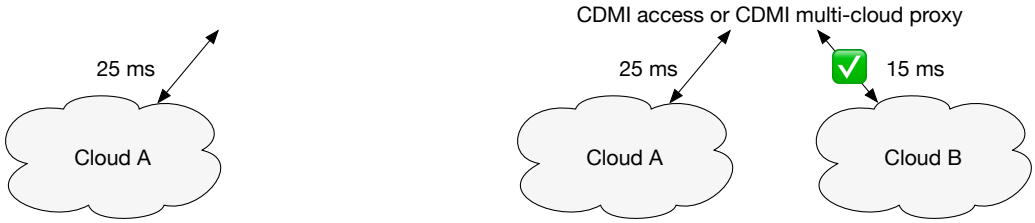

Figure 8: Selectively access clouds based on latency

Throughput increase (parallelism)

Objects stored on-prem can be made accessible to multiple clouds. There are often limits to the rate at which on-prem objects can be processed by any given cloud at any given time. By processing on-prem objects in parallel across multiiple clouds, those limits can be exceeded and the aggregate total throughput can be increased.

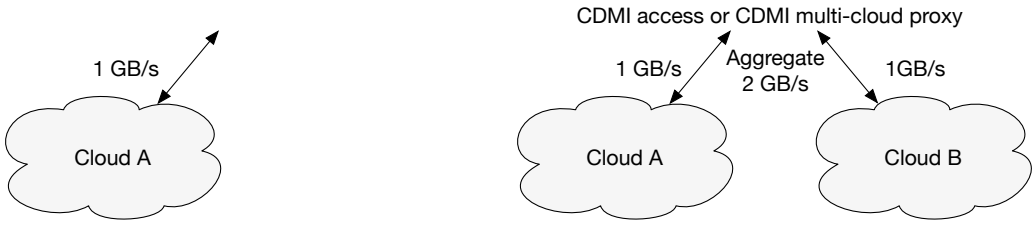

Figure 9: Increase throughput across multiple clouds.

Storage objects can be sharded and erasure coded such that each shard is stored on a different cloud. Parallel requests to each cloud return the shards. When a sufficient number of shards are returned, the object can be reconstituted. This gets around any individual cloud provider's bandwidth limits. Sharding also increases the security since no one cloud has all the data.

Egress cost reduction

Cloud providers have varying charges for egress. Some providers' egress charges are quite high, making retrieving data a very expensive proposition, however others may not charge at all for egress. By using multiple clouds to store the data, the cloud with the cheapest egress charges can be used to fetch the object.

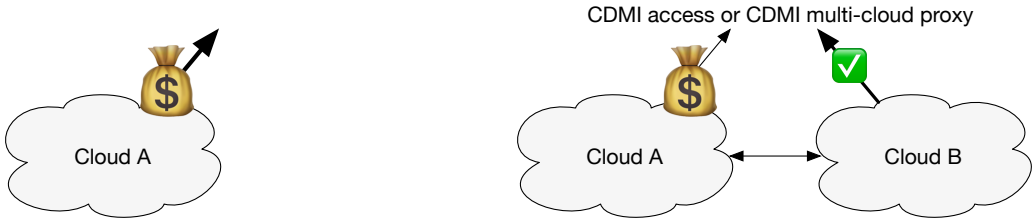

Figure 10: Reduce cost by retreiving from lower cost cloud

Summary

Using multiple clouds has become the norm for many organisations. A multi-cloud strategy enables organizations to reduce risk and dependence on a single cloud platform.

CDMI enables cloud solution vendors to meet the growing need of interoperability for data stored in the cloud. The CDMI standard is applicable to all types of clouds – private, public and hybrid. It provides end users with the ability to control the destiny of their data and ensures data access, data protection and data migration from one cloud service provider to another.

The CDMI standard can be downloaded from the SNIA web site at: https://www.snia.org/sites/default/files/technical-work/cdmi/release/CDMI-v2.0.0.pdf

The CDMI standard can be downloaded from the International Organization for Standardization web site at: https://www.iso.org/standard/83451.html