infinite io

White Paper

Maximizing the Performance and Usability of Object/Cloud Storage with the NSC-110 Network-Based Storage Controller

Overview

File-based data is growing at such a rapid pace that existing storage solutions cannot cost-effectively scale to meet the challenge. A study conducted by IBM in 2012 found that ninety percent of all the data in the world was created during the proceeding twenty four months (*Bringing Big Data to the Enterprise." IBM - What Is Big Data? 08 Mar. 2012)*. To complicate matters even more, over eighty percent of the new data created will be accessed a few times over the first few days of its creation and never accessed again.

Traditional storage systems were not designed to accommodate this type of rapid data growth and the existence of massive stores of inactive data. The underlying use of RAID (Redundant Array of Independent Disks) as a fault-tolerance and disk failure protection technology breaks down to a point where it is impractical to use. Low-cost, super high-density disks, which are ideal for inactive data storage, don't work well in traditional RAID-based storage systems for two reasons: If a super high-density drive fails in a traditional RAID group, it can take days to rebuild the RAID group. During this time the entire array, or that RAID group, is totally out of service and data access is stopped. The other reason high-density drives don't make sense in the traditional RAID deployment model is: why would you want an expensive controller architecture tuned for performance, accessing low performance high-density drives? It's a major performance mismatch and not the ideal systems architecture.

Object or cloud storage provides the perfect solution for the storage of inactive data. Object stores by their very design accommodate high density drives and have built-in fault-tolerance using forward error correction know as erasure coding and multi-site distribution for disaster recovery. Unfortunately, the low-cost storage and fault-recovery functionality of object storage typically comes at the price of performance. In general, object storage does not perform as well as RAID-based file systems.

Using Object Storage as Primary Storage – Gateways Just Don't' Work Well

To use an object store with traditional file systems like Network File System (NFS) or Simple Message Block (SMB) has traditionally required a gateway to convert between the object and file worlds. The challenge in this approach is the performance mismatch between traditional file-based applications and an object store. Gateways work well in low-performance, non-mission-critical applications; but for the bulk of enterprise applications, gateways have major performance issues.

Gateways operate similar to data caches; files get moved into the gateway from the object store after the file has been accessed several times or deemed "hot". The problem with this approach is for a file to be deemed "hot" it has to suffer through poor performance while it is warming up. Once warmed up to "hot" and in the gateway, the next challenge that gateways have is they contain a limited amount of storage that is typically only a fraction of what a traditional storage array would have installed. So once

a file gets promoted to “hot” status it must stay hot or it will eventually be demoted to “warm” and then “cold” status and kicked out of the gateway back to the object store until it deemed “hot” again.

This architectural shortcoming of gateways causes what is called a “ping-pong” effect as files go back and forth between the gateway and the object store as their status alternates from “hot” to “cold”. If a public object store or cloud is used, this can rack up significant in and out usage and bandwidth charges. In a private object store or cloud this effect generates very unpredictable performance making a gateway in front of an object store or cloud an unsuitable replacement for primary storage arrays in most applications.

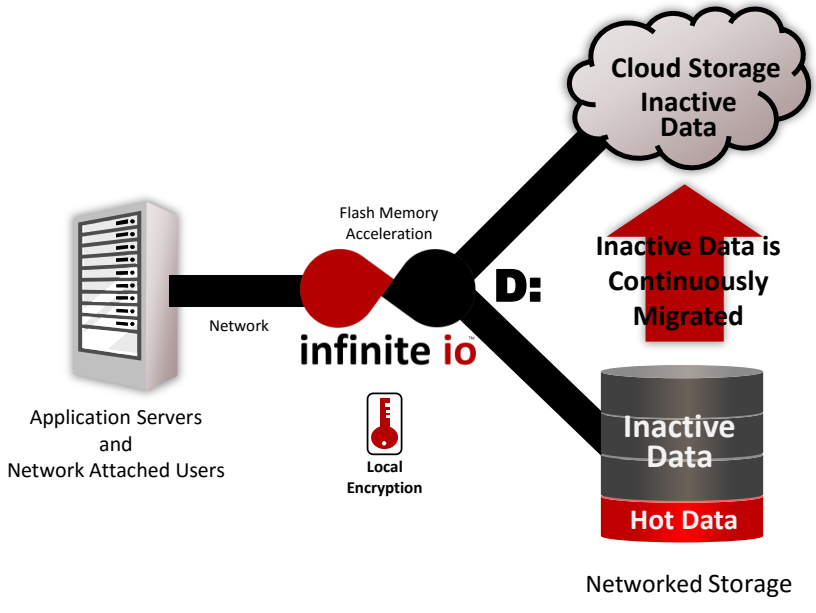
A Different Solution is Needed

A solution that bridges the performance gap of traditional storage systems with the low-cost and fault-tolerance of object storage is clearly needed to make object/cloud storage an integral part of today’s datacenter. One that preserves the investment made in existing storage systems and applications that doesn’t require the rip-and-replace of existing systems or the total reinstallation of applications.

The NSC-110 Network-based Storage Controller – Something Completely Different

Infinite io has taken a radically different approach to integrating object/cloud storage into existing or new environments with the NSC-110 Network-based Storage Controller. The NSC-110 is based on a layer-seven proxy or bump-on-a-wire and installs between servers and/or clients and existing or new primary storage systems.

Infinite io NSC-110 Network-based Storage Controller



The NSC-110 installs as a bump-on-a-wire and continuously migrates inactive data to an object store or cloud while being totally invisible to installed storage and applications

By installing in-line in front of primary storage the NSC-110 is able to use deep packet inspection inspect all traffic that goes between servers/clients and primary storage systems. The NSC-110 connects to private or public object/cloud storage through a standard Ethernet connection supporting multiple object interfaces including Amazon's S3 and OpenStack.

WHAT! - No Overlay File System, No New Mount Points or Gateways Required

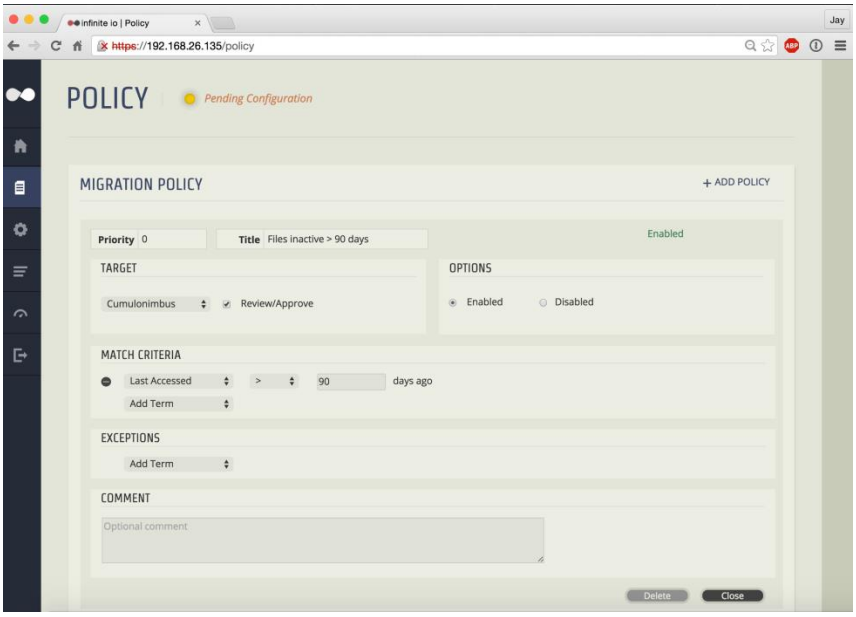
The NSC-110 Network-based Storage Controller is unique in that it exposes no IP address, no new mount points and requires no domain mapping. Servers and clients continue to see and work with the same drive letters and mount point names they did before the NSC-110 was installed.

The NSC-110 accomplishes this functionality by using two technologies typically not associated with network storage devices. Based on a layer 7 proxy or bump-on-a-wire, the NSC-110 is totally invisible to installed storage systems, networking devices and servers/clients. In place of an overlay file system, the NSC-110 uses deep packet inspection to discover file activity. Deep packet inspection literally analyses the contents of packets passing through the NSC-110 at 10 Gigabit wire rates. By inspecting packets from Layer-1 to Layer-7 of the OSI model, the NSC-110 is able to discover file read/write activity and produce a wealth of real-time analytics. Deep packet inspection eliminates the need for a complicated overlay file system and also preserves all file handles relative to their native storage system.

Continuous Policy-based Automated File Migration

Unlike cloud gateways, the NSC-110 proactively migrates inactive or "cold" files off primary storage to an object store based on IT defined policies. Once defined, policies continuously migrate and delete files off primary storage systems to a private object store, a public cloud or a low-cost NFS system.

The NSC-110 Policy Creation User Interface



Policies are easily created using the NSC-110 user interface and can be global across all storage systems or limited in scope to specific systems, directories, and even users

Policies can be as simple as “move all files that have not been accessed in the last xx days”. Policies can be multi-tiered and have exceptions similar to “always migrate all *.mpeg2 files” or “never move files from \directory” or “always move files from \directory”.

Policies can be limited to one mount point or global. Polices can also migrate data to different object stores and have the option to geographically restricted data movement. With the NSC-110 data is migrated based on solid IT defined logic that goes well beyond just access patterns.

Unheard of Performance for Object/Cloud Storage – The Metadata Map

Cloud gateways act like a cache and move files deemed hot into the gateway over time as specific file usage increases. This is kind of like closing the door after the cows have already left, since you’re always behind the real-time data flow and experiencing poor performance while data is warming up.

Statistics have shown that for most workloads, eighty percent or more of all traffic is nothing more than metadata lookups. The NSC-110 takes an elegantly simple approach to solve the object storage performance issue – it keeps all metadata for all supported storage in DRAM and flash memory. It’s always hot and needs no warm up. And since the NSC-110 leaves active data on the primary arrays it’s supporting, if an entire file is requested, the NSC-110 just passes that request to the array to respond to.

If migration polices are properly created, files in the object store should rarely be accessed and eighty percent of that access will likely be metadata the NSC-110 can serve directly out of memory. On the rare occasion a file is actually read from the object store, the NSC-110 servers up the metadata from memory while it streams the file to the server and recreates it back on the supported storage array where it stays until a migration policy transfers it back to the object store.

Local files systems will also perform better as they become uncluttered with inactive or “cold” data. With the NSC-110 files on both primary storage and object/cloud migrated files get maximum performance.

Summary

By creating a new class of product the “Network-based Storage Controller”, the NSC-110 solves one of the biggest IT challenges today – how to cost effectively store inactive or “cold” data. The NSC-110’s total invisibility to installed applications and its continuous automatic file migration is unique to the industry and provides a non-disruptive means of object/cloud enabling any file-based application or storage system.