



Building a Distributed Block Store on Xen

Julian Chesterfield
Storage & Virtualization Architect, OnApp

julian.chesterfield@onapp.com

Overview

- Introduction to OnApp and the Storage Platform
- High-Level Goals of the Project
- Project Architecture
- Technical Details
- Additional Feature Set
- Conclusions

About OnApp

Cloud platform provider for hosts,
service providers & other enterprises



OnApp

Cloud



OnApp

CDN



OnApp

Storage

Founded July 1st 2010

400+ clients

~1,000 cloud deployments

93 CDN PoPs

Storage beta underway

OnApp Cloud Platform

- Infrastructure as a Service platform runs on commodity hardware
- Supports OSS Xen, KVM and VMWare ESX HV platforms
- Install and go – includes user-based interface, billing integration, over 400 VM templates etc..
- Diskless boot of Hypervisors named *CloudBoot*
 - New feature in 3.0 release
 - Uses standalone ramdisk image for control domain
 - No persistent installation requirements, servers identified by MAC address

=> Over 50% of public cloud deployments today are managed by OnApp software

The OnApp Storage Platform

- Work began mid-2011
- Storage development team based in Cambridge, UK
 - Former XenSource/XenServer developers
 - UI experts
- Bringing affordable enterprise-class storage to cloud providers
- Bundled for free with the core OnApp cloud product

Common Storage Challenges – Cost

- Storage platform is often the largest investment for cloud providers
- Proprietary lock-in ensures costs continue to rise
- Forklift upgrade cost for higher capacity/IO throughput

Common Storage Challenges – IOPS Scalability

- Virtualization infrastructure must share and have access to the same storage platform
- Cloud hypervisor infrastructure growth is essentially unlimited
- Scaling a centralized SAN requires massive investment in IO controller capacity
 - Proportional with the HV infrastructure
 - Often proprietary hardware + licensing involved

Centralized SAN can scale - but slowly,
and at significant cost

Common Storage Challenges – Capacity Scale

- Growth of HV infrastructure demands dynamic capacity scale:
 - Proprietary drives are costly
 - Adding disk array units is costly and requires space
 - Adjusting RAID group membership can be cumbersome

Common Storage Challenges – Physical Space

- SAN disk arrays have significant physical requirements:
 - Additional rackspace for all head units, disk array units, interconnects etc...
 - Independent cooling
 - Redundant power supplies
 - Redundant IO controllers
 - Significant cabling overhead

Platform Design Goals

Platform Goals [1]

- Utilize commodity integrated *storage drives within Hypervisors*
- Hypervisor agnostic
- Simplify *content location and awareness* for data
 - No complex RAID methodology across servers
 - Content location fully exposed through platform APIs
 - Optimised co-location of customer VMs with storage data
- Distribute IO load
 - Content replicated across physical drives in separate servers
 - Read local or read distributed policies

Platform Goals [2]

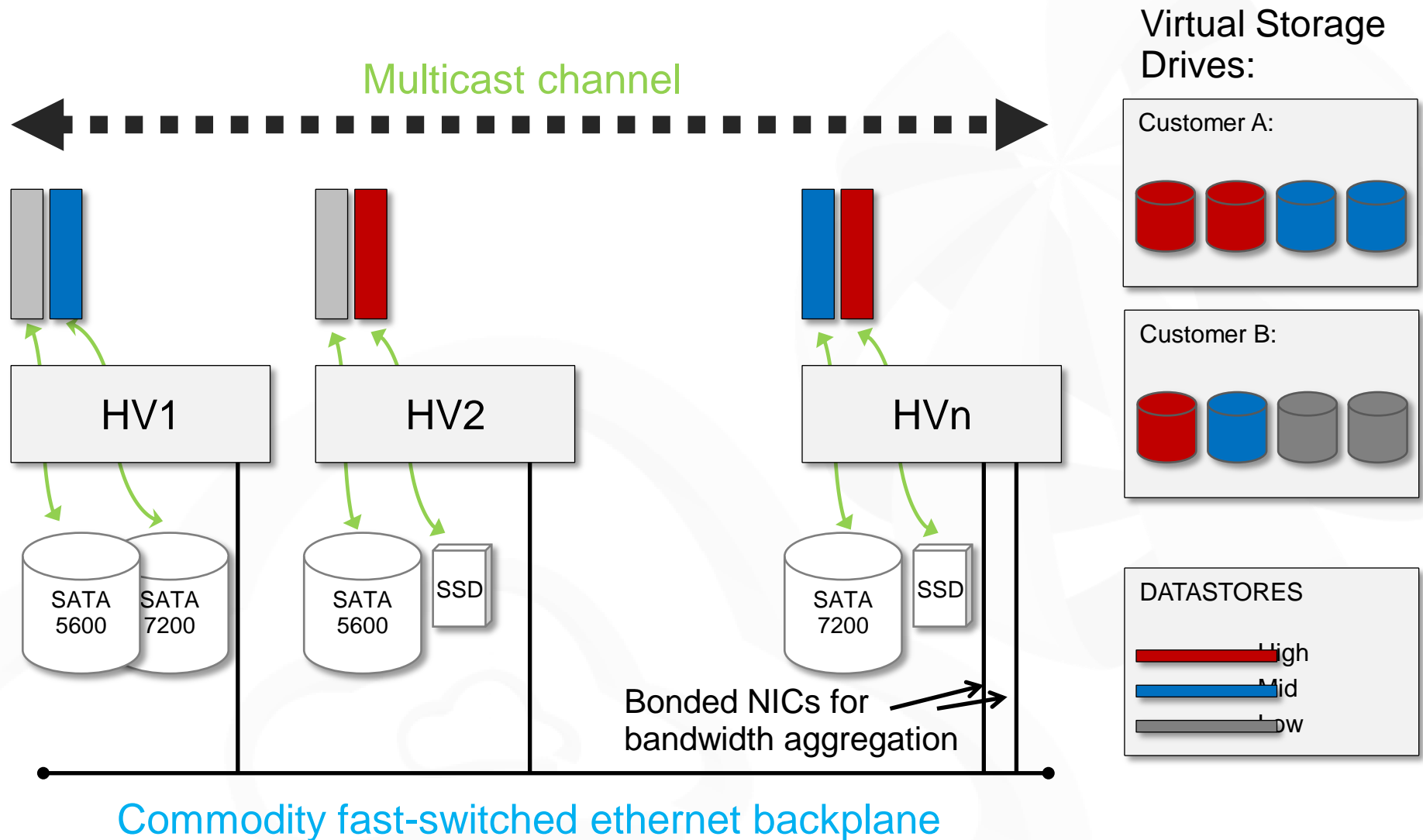
- Fully Decentralized content management
 - No single point of failure across whole system
 - All drives operate independently, and can be relocated across any HV chassis
- High Performance using commodity hardware
 - Negligible metadata overhead
 - As close to raw drive IO throughput as possible
- Storage platform runs as a service without impacting any other VM workloads
 - Utilize HV platform to provide resource and performance isolation
- Resilient to drive and/or server failure without impacting data availability

How Did We Meet These Goals?

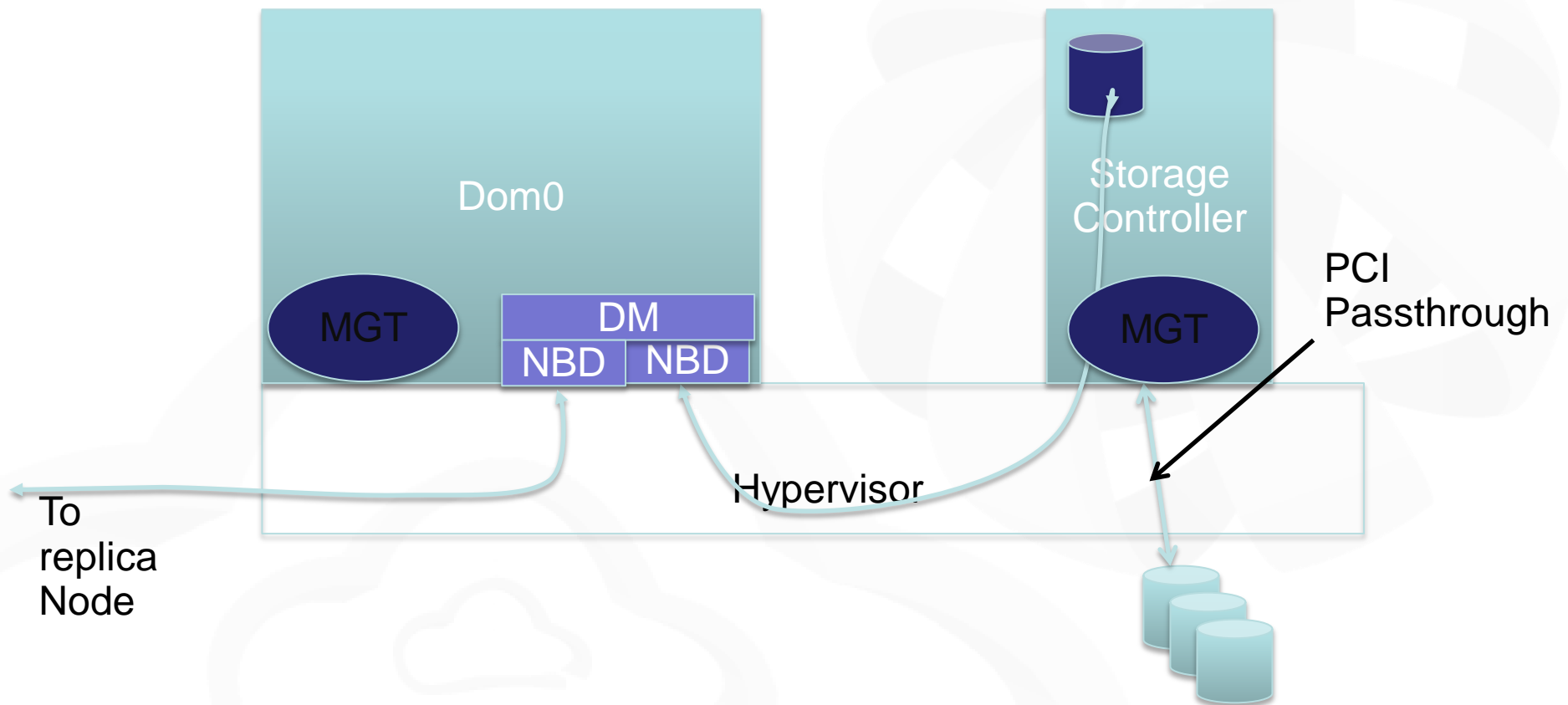
OnApp Distributed Storage Platform

- Each HV has a storage VM 'appliance' that manages content on it's physical drives
- Appliance VMs communicate over a fast ethernet SAN backbone
 - Dom0 has access to the SAN network also
 - Appliance VMs can have dedicated NIC passthrough hardware, or virtual bridge access
- Drives are fully plug and play and auto-discover on startup
 - Content remains accessible if replicas are online
 - Each drive has it's own unique ID in the system and can be located in any HV chassis
- Drives are classified into performance tiers and logically grouped together
- SAN utilises private 10.x/16 address space. Each HV is responsible for it's own /24 space

A Smart, Independent Array of Storage Nodes



Internal Server view



Storage Controller Appliance





- Minimal Busybox Ramdisk image – approx 10MB compressed
- Allocated 256MB RAM at runtime
- Pvops linux kernel for best performance on Xen and KVM

SAN Management UI

The screenshot displays the OnApp web interface for SAN management. The browser address bar shows the URL `http://212.44.45.214/storage/data_stores`. The interface includes a sidebar with navigation options: Dashboard, APPLIANCES (Virtual Machines, Load Balancers, Hypervisors), Integrated Storage (selected), USERS (Users and Groups, Roles, Billing Plans), and TOOLS (Settings, Logs, Stats, Sysadmin Tools, Alerts, Help). The main content area is titled "Data Stores" and contains a table listing existing data stores. Below the table is a "Storage Nodes" section showing four nodes (hv1_xen, hv2_xen, hv3_xen, hv4_xen) with their respective disk configurations. A "Create new Integrated Storage DataStore" button is located at the bottom right of the Data Stores section.

Data Stores

The data stores in this cloud. To view or change data store's settings, click its label.

Label	Identifier	No Disks	Volume	Performance	Actions
tds	39bnfxh5kuwpgm	2	2 TB (2 TB free)	Normal	 
igor-datastore-10	y97lnp3q4a0rzj	3	119 GB (108 GB free)	High	 

Create new Integrated Storage DataStore

Storage Nodes

Four storage nodes are shown, each with a label and disk configuration:

- hv1_xen: 119 GB, 466 GB
- hv2_xen: 119 GB, 466 GB
- hv3_xen: 149 GB, 466 GB, 119 GB
- hv4_xen: 119 GB, 466 GB, 466 GB

Powered by OnApp

SAN Management UI – Create Datastore

The screenshot displays the OnApp web interface for creating a new integrated storage datastore. The browser address bar shows the URL `http://212.44.45.214/storage/data_stores/new`. The page title is "OnApp > Add New Integrated Storage DataStore".

Left Sidebar:

- Dashboard
- APPLIANCES
 - Virtual Machines
 - Load Balancers
 - Hypervisors
 - hvv1
 - hvv2
 - Unassigned HVs / Reserved
 - Templates
- Integrated Storage (highlighted)
- USERS
 - Users and Groups
 - Roles
 - Billing Plans
- TOOLS
 - Settings
 - Logs
 - Stats
 - Sysadmin Tools
 - Alerts
 - Help

Main Content Area:

Add Data Store
To add a new data store, give it a label, specify the disk capacity in GB, then click the Save Data Store button.

Add New Integrated Storage DataStore

Properties

Name:

Advanced: ☒

Advanced

Redundancy:

Stripes:

Storage Nodes

Filter by Hypervisor:

Performance:

☒ 465.762 GB (Normal, HV1_perf_test)

☒ 465.762 GB (Normal, HV3_perf_test)

Footer: Dashboard > Data Stores > Add New Integrated Storage DataStore

Performance Isolation

- Hardware pass-through of storage controllers to an IO controller VM (Hard or soft IOMMU)
 - Dom0 has no access to integrated storage drives
 - Diskless boot of HVs
- Can also assign dedicated passthrough NICs for complete network separation as well
- Controller VM acts as a driver domain on behalf of the whole cluster, and manages IO/content handling for all physical drives that are present

Decentralized Consistency

- Physical drives are grouped in Datastores (Diskgroups)
 - Each drive has unique identifier
 - Drives are location independent
 - Datastores have a particular replication and striping policy associated
- A new virtual disk object (or LUN) is assigned to a datastore
 - Content owning members are selected from the diskgroup
 - # of Content replicas/logical stripe members is based on Datastore policy
 - Vdisks map directly to VMs as VBDs
- Each vdisk content instance stores information about the other owners *For that unique piece of content*
- Simplified Paxos-like commit protocol used for vdisk operations

Vdisk operations only involve direct communication with a small number of nodes that are owning members

Datapath Performance

- Virtual block device queue can be instantiated from any Hypervisor control domain
- IO path is optimal between front end device queue and actual content members
 - Distributed IO load across the whole infrastructure ensures that there's no centralized IO bottleneck
- Read local policy vs read distributed policy
 - VMs can be co-located with content replicas for localized read path only (network constrained environments)
 - VMs can be hosted anywhere with distributed read policy (over-provisioned network bandwidth)

Some Performance Stats

Setups	Devices	READS		WRITES	
		256KB	512KB	256KB	512KB
LOCAL					
Raw Single Device	SSD	168	222	169	164
OnApp Single Device	SSD	167	182	130	106
Raw Single Device	SATA HDD	116	117	94	105
OnApp Single Device	SATA HDD	115	115	86	97
OnApp 2 Device Striped	SATA HDD	191	189	123	127
OnApp 4 Device Striped	SATA HDD + SSD	413	361	251	291
REMOTE					
OnApp Single Device	SSD	116	117	112	108
OnApp Single Device	SATA HDD	108	109	61	72
LOCAL + REMOTE (2HVs)					
OnApp 2 Stripe	SSD	228	230	226	216
OnApp 2 Stripe	SATA HDD	217	180	128	115
OnApp 4 Stripe	SATA HDD	232	230	214	221

Resilience to Failure

- Content owning members get selected across HVs
- IO queues depend on TCP-based NBD device protocol
- Custom device-mapper failover driver operates above individual device queues to failover reads and writes
- Data consistency handled across members via the control path
 - Explicitly validated before activating the device queues

Hot Migrate of Data

- Online content migration from one physical drive to another (same HV or across HVs)
- Uses content hash mechanism for smarter differencing block identification
- Can be used to repair out of synch members and to move data replicas to another place

Additional Features

- Online disk snapshot
- Thin provisioned at source to enable disk over-commit
- De-duplication within drives
- Fully integrated into the OnApp cloud management platform

Conclusions

- Enterprise SANs costly to purchase and to scale
- OnApp storage platform exposes local storage drives across HVs as SAN
- Distributed IO paths decentralise traffic flows to provide much better scalability properties
- Smarter location of end-user VMs and storage replicas improves IO performance and reduces network traffic

Questions?

julian.chesterfield@onapp.com

...and here's some I prepared earlier 😊

1. Is this Open Source?

- Not at this time but keen to do so

2. How far will this scale?

- Scalability limits are only really the number of Vdisk objects that can be cached and processed usefully by endpoints in the SAN

3. Is server hardware really good enough to meet or outperform enterprise SANs?

- Yes. Performance comes from massively distributed IO controller points, not the IO throughput of any particular node

4. What are the major differences to other distributed blockstore/filesystems?

- Primarily content location awareness for more optimal localised read path and no centralised metadata component

Decentralized Consistency [2]

- Storage controller VMs report information only about content for which they are authoritative
- Content membership changes are strictly handled across the ownership group
- The same vdisk information is always reported by all members of the ownership set

A vdisk object is therefore known about correctly by any other member of the SAN, or not at all.