IBM XIV at Red Hat

Architecture, Performance and Tuning

Will Foster

Senior Sysadmin and IT Storage Lead, Red Hat

June 4, 2012

RED HAT: WHAT WE DO.

We offer a range of mission-critical software and services covering:

CLOUD

MIDDLEWARE

OPERATING SYSTEM

VIRTUALIZATION

STORAGE

HOW WE DO IT.

We develop everything using an OPEN SOURCE model.

Shared development reduces costs & accelerates innovation.

Open collaboration offers products that genuinely meet customers' requirements.

THE BENEFITS.

- ✓ Flexibility
- ✓ Faster technology innovation
- ✓ Better quality
- ✓ Better price & performance
- ✓ Alignment to your needs

IBM XIV Footprint - Architecture

Architecture

1,800 Servers

2.13PB Storage

2 Datacenters

6 IBM XIV Frames

- 5 x Gen2
- 1 x Gen3

Cisco MDS SAN Fabric

- 9513 core switches
- *110+* BC switches
- IBM HS22, 22V
- IBM X Series for Databases

602TB XIV Capacity

2079 Volumes

- 356 attached hosts
- 21 cluster groups

XIV is our standard for SAN/FC and transactional environments
with moderate to high performance requirements

IBM XIV Footprint - Environment

Environment

100% Virtualized
Non-production

42% Virtualized
Production

Database Environments

- Oracle 11g
- Oracle RAC
- MySQL & PostgreSQL

Performance Data-sets

- Business Intelligence
- Data Warehouse
- Enterprise Service Bus

KVM-based Dev Cloud

- iDataplex
- RHEL5 & 6
- SoNAS

RHEV Virtualization

- Production
- Stage
- Test/Dev/QA

Disruptive Growth in Compute and Storage

2012 – 2.13PB
New Datacenter

+73%

2011 – 1.23PB
Two new Datacenters

+102%

2010 – 622TB
New Datacenter

+43%

2009 – 432TB

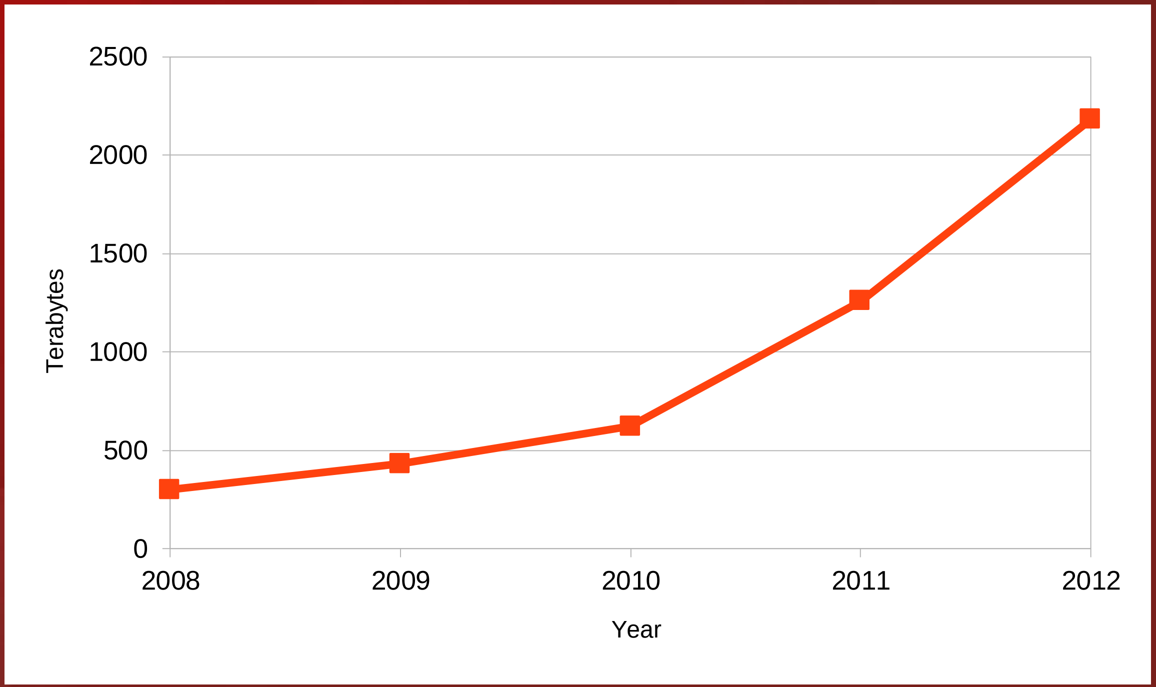
+44%

2008 – 300TB

Problem:
Explosive Growth in Storage Capacity

Problem:
Surge in Business Demand and Delivery

- Scaleable Storage
- RHEV & Bladecenter Architecture
- Automation and Config Management

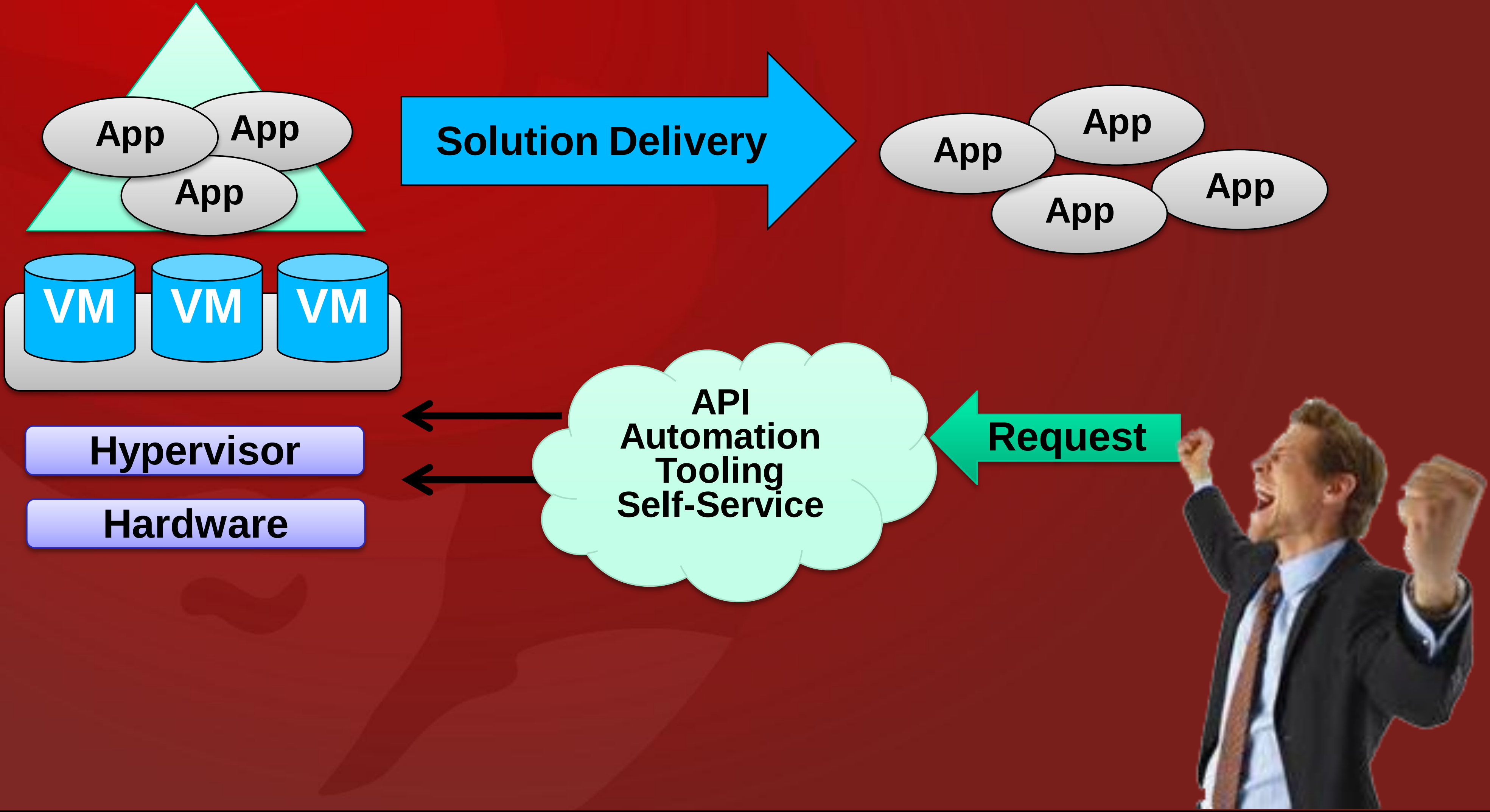


Virtualization, Cloud and Self-Service

“Everything is different, but the same... things are more moderner than before... bigger, and yet smaller... it's computers...”

- Ox, Bill and Ted's Excellent Adventure (1989)

Virtualization, Cloud and Self-Service



App

App

App

VM

VM

VM

Hypervisor

Hardware

Solution Delivery

App

App

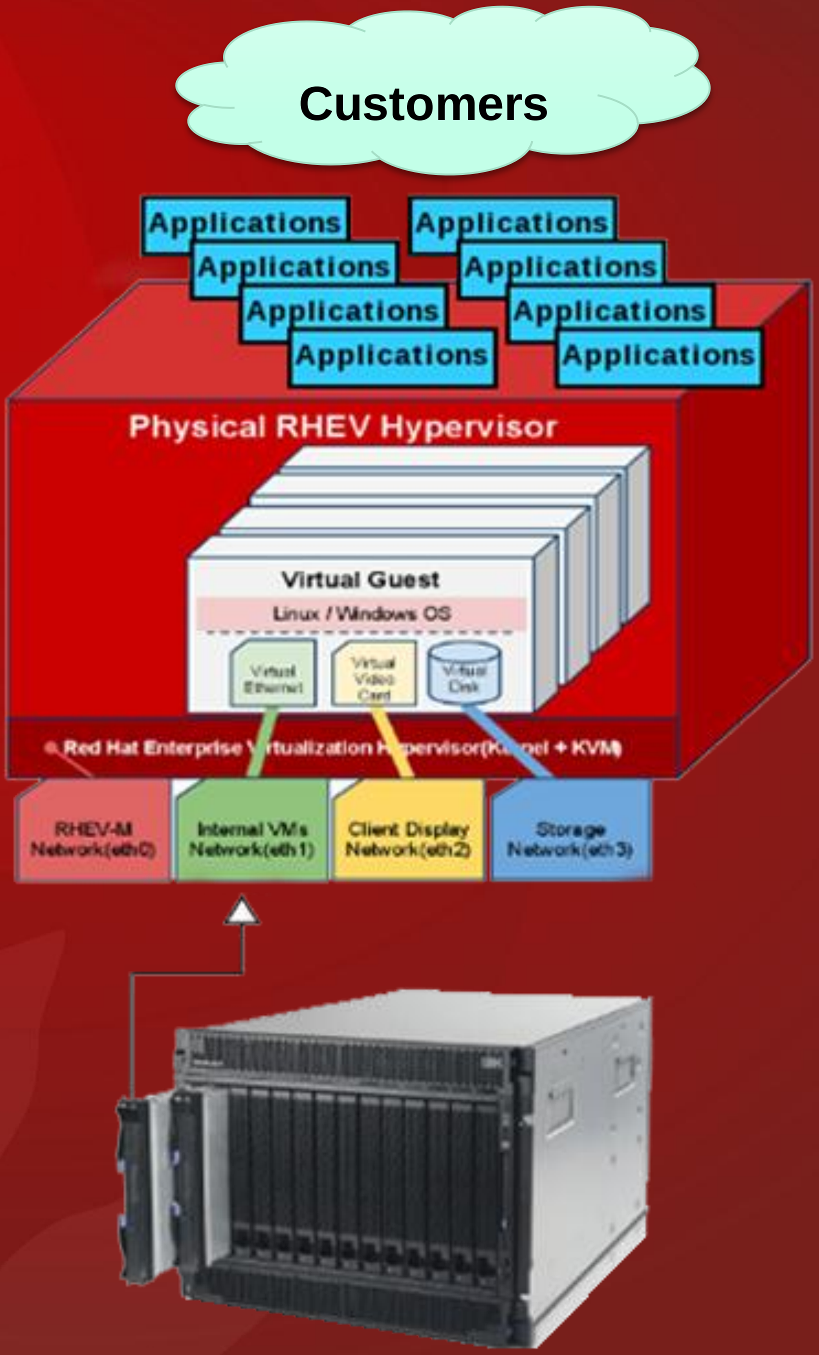
App

App

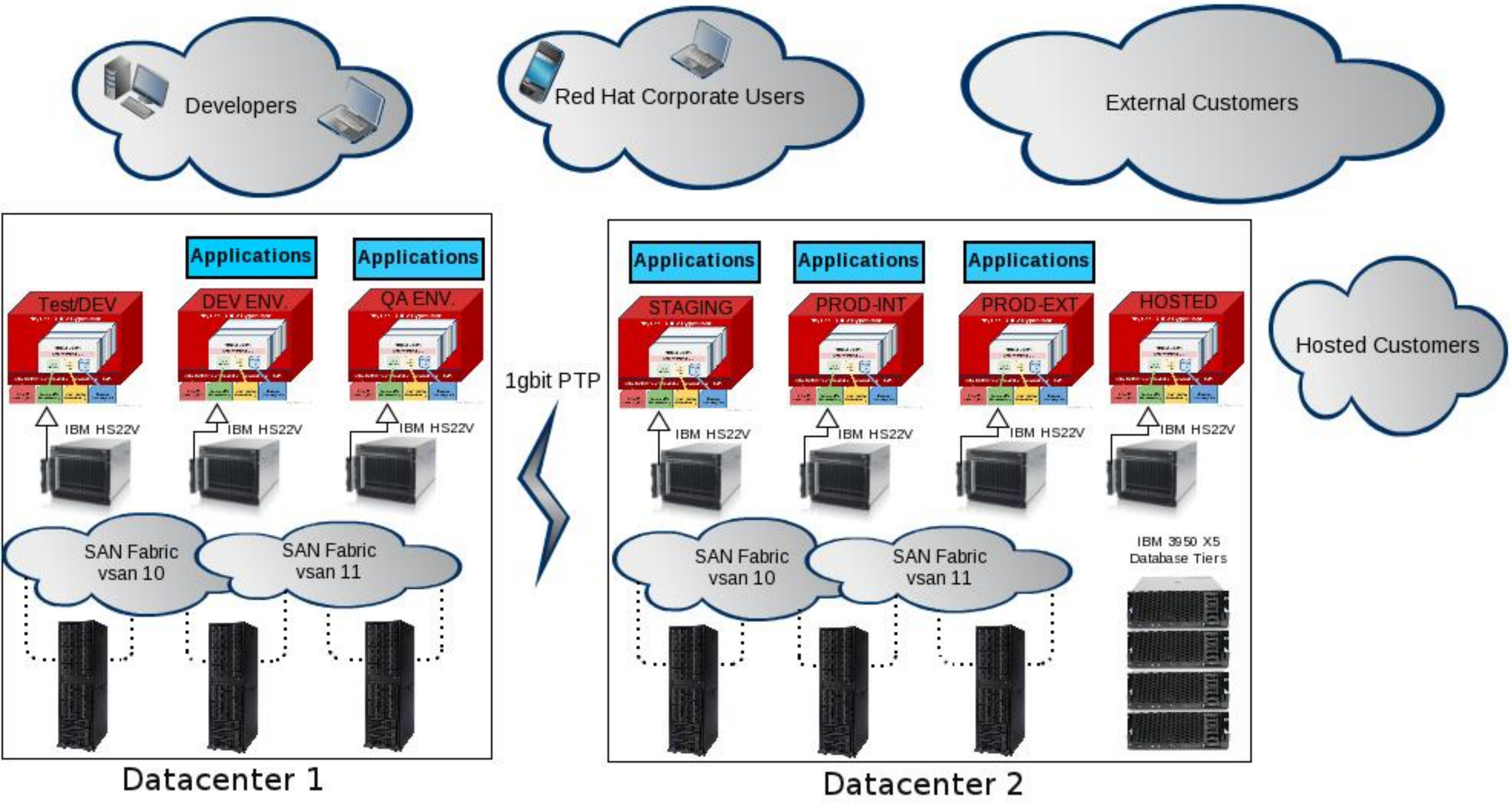
API
Automation
Tooling
Self-Service

Request

RHEV Design (per Hypervisor)



RHEV on XIV in the Datacenter



Performance in Transactional Workloads

We use 1-3TB sized LUNS

- Databases range from 1-2TB in size
- Average **2 to 5ms** latency or service time

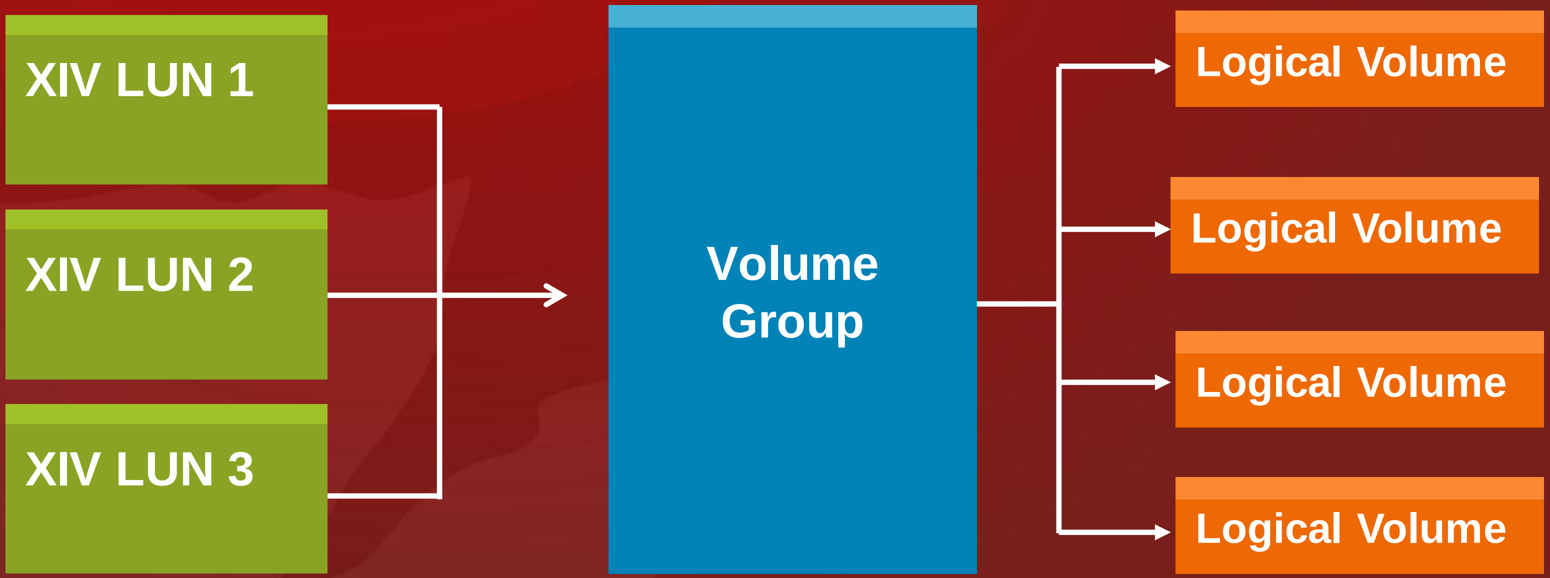
With XIV grid architecture:

- Do not have to balance X sized LUNs in order to achieve optimal cache and spindles, chunks go to every disk

LVM Concatenation > To grow datasets:

- New LUN as a 'PV'
- Extend the VolumeGroup and LogicalVolume (non-disruptive)

Filesystem: Ext3 or Ext4 (on top of LVM Logical Volume)



XIV SAN Multipath Internals on Red Hat

Example output of 'multipath -ll' on Red Hat with XIV SAN

```
mpath2 (20017380004130b5f) dm-1 IBM,2810XIV
[size=1.0T][features=1 queue_if_no_path][hwhandler=0][rw]
\_ round-robin 0 [prio=8][active]
 \_ 8:0:0:3  sdab 65:176 [active][ready]
 \_ 8:0:1:3  sdah 66:16  [active][ready]
 \_ 10:0:0:3 sdan 66:112 [active][ready]
 \_ 10:0:1:3 sdat 66:208 [active][ready]
 \_ 3:0:0:3  sdd  8:48   [active][ready]
 \_ 3:0:1:3  sdj  8:144  [active][ready]
 \_ 5:0:0:3  sdp  8:240  [active][ready]
 \_ 5:0:1:3  sdv  65:80  [active][ready]
```

- On XIV, all paths are active at the same time
- IOPS balance across all paths so queue_depth needs to be **tuned lower** than traditional active/passive arrays
- Application uses may determine queue_depth as well i.e. number of Oracle DB writers/threads etc.

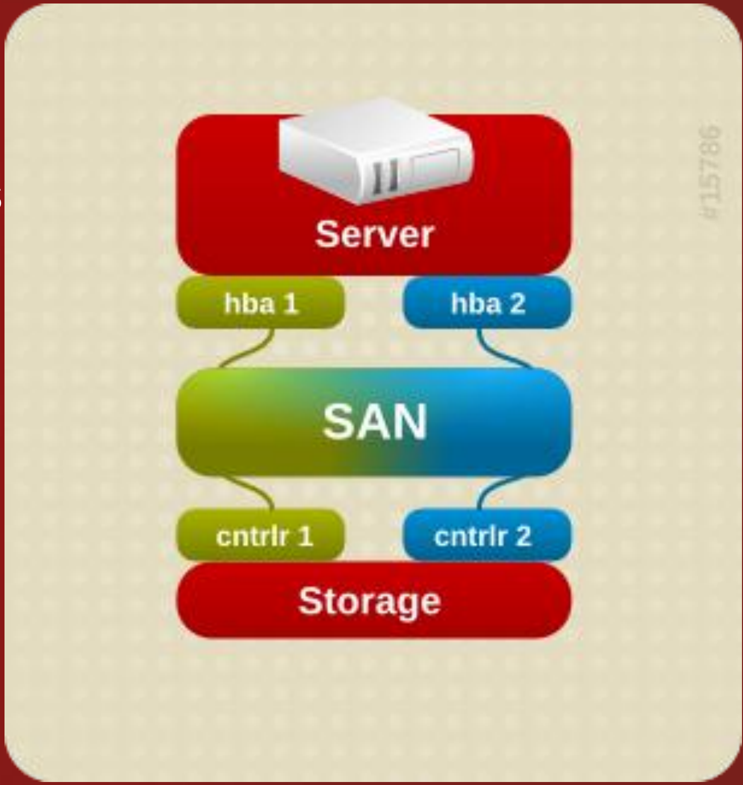
XIV SAN Multipath Internals on Red Hat

- **dm-multipath** is the default RHEL mpath subsystem
- Supports active-active and active-passive round-robin
- Some configurations support load shifting to different paths
- For optimal performance, tune the "*min_rr_io*" parameter
- "*min_rr_io*" determines how many requests to service before moving to the next path

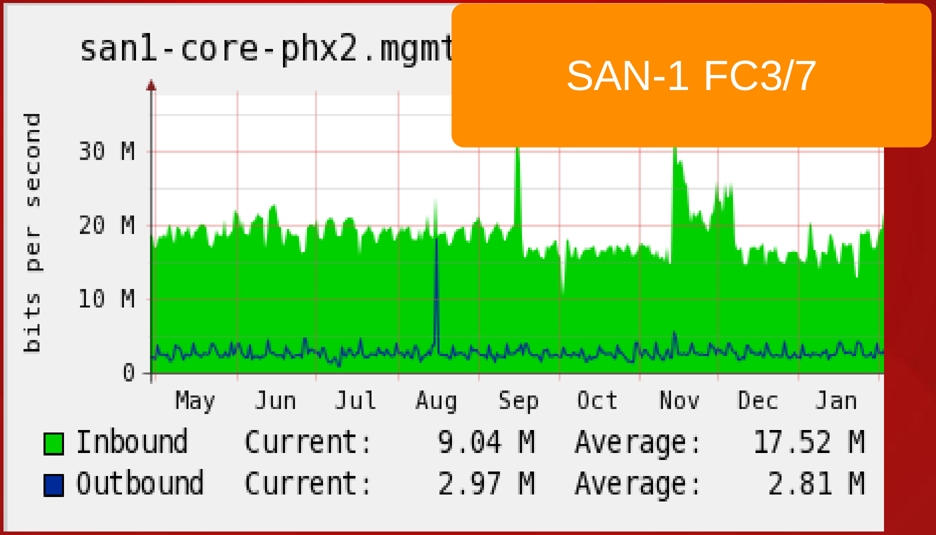
❑ For transactional environments, keep *min_rr_io* between 10-32 (or 10 less than your queue_depth)

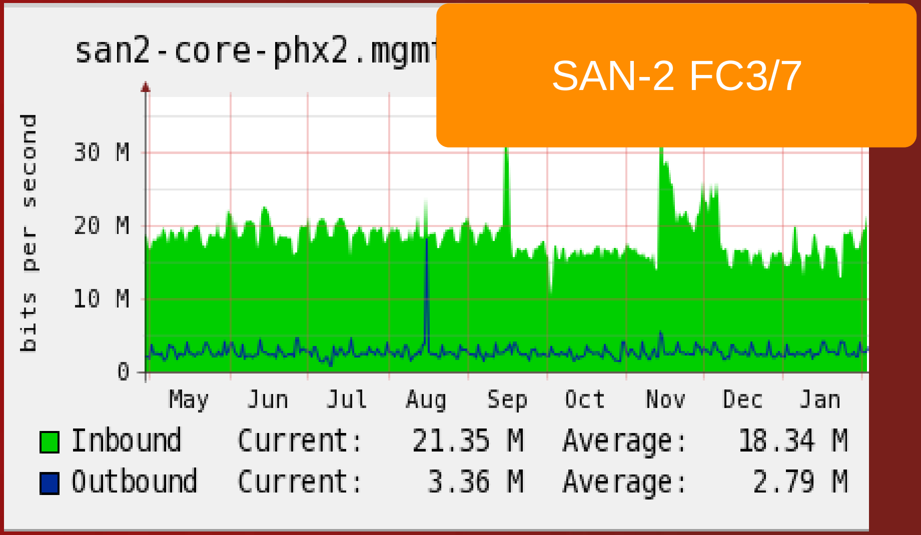
❑ For sequential workloads, keep min_rr_io around 100-200

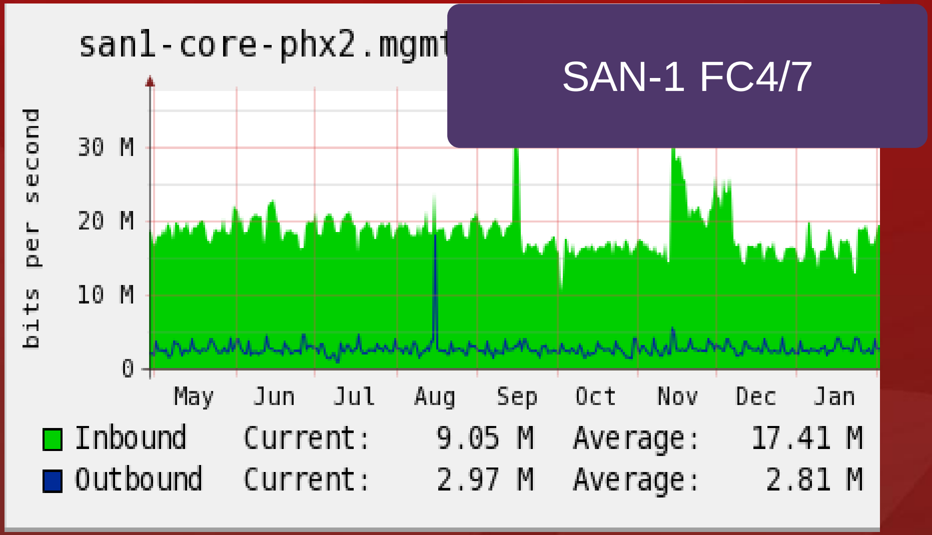
❑ Experiment to see what works best for you

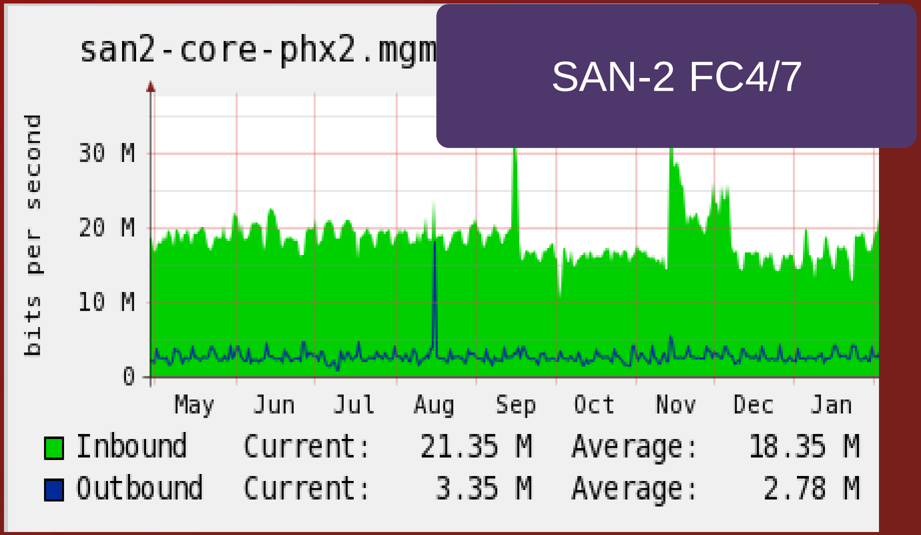


Active: Active SAN pathing (4paths)


SAN-1 FC3/7


SAN-2 FC3/7


SAN-1 FC4/7


SAN-2 FC4/7

Tuning Queue_Depth on your HBA

- **Tuning HBA and host queue_depth**
- **Determine your HBA defaults**

```
cat /sys/class/scsi_device/*/device/queue_depth
```

- **Use a tool like “nmon” to monitor in-flight queues**
- **You may need to increase if:**
  - Queue_depth buffer gets consistently saturated on your paths
  - You experience paths bouncing around erratically
  - Tune your multipathing subsystem appropriately
  - Multiply #paths against your queue_depth to obtain optimal settings
- Example: 4 x paths, HBA queue_depth of 64 = *256 queue_depth*

Tuning Queue_Depth on Local Disk

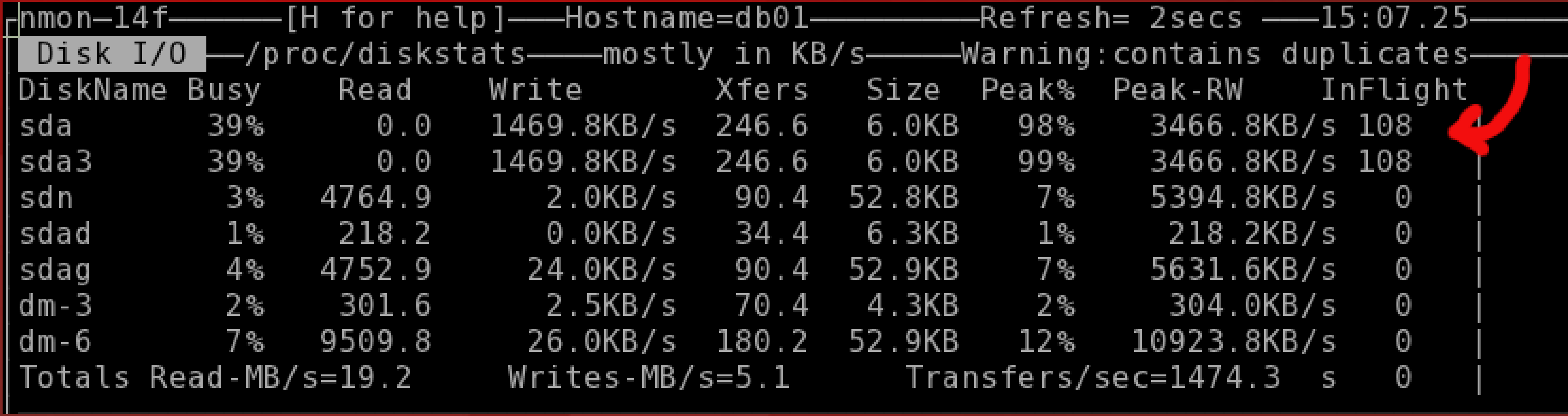
- **Tuning HBA and host queue_depth**
- Monitor your local disk queue_depth in the same way
- If it is too low, performance may be affected
- You can easily increase this on the fly via:

```
echo "128" > /sys/block/sda/device/queue_depth
```

- For best practices > Consult your HBA and disk manufacturer
- Consider what makes sense for your workload (sequential/random, etc.)

Using NMON on Linux to Monitor Performance

- **Look at “InFlight” queues to determine your profile**
- **Balance disk & HBA queue_depth with your multipath settings**
- **Test, Test, Test!**
- Tools like hdparm, dd, bonnie++ and iozone can test general IO
  Tools like orion can simulate database load.



```
┌nmon─14f─────────[H for help]───Hostname=db01──────────────Refresh= 2secs ───15:07.25───
│ Disk I/O ──/proc/diskstats────mostly in KB/s──────Warning:contains duplicates───
│DiskName Busy    Read     Write       Xfers   Size  Peak%   Peak-RW    InFlight
│sda       39%     0.0  1469.8KB/s  246.6   6.0KB   98%   3466.8KB/s 108
│sda3      39%     0.0  1469.8KB/s  246.6   6.0KB   99%   3466.8KB/s 108    |
│sdn        3%  4764.9     2.0KB/s   90.4  52.8KB    7%   5394.8KB/s   0    |
│sdad       1%   218.2     0.0KB/s   34.4   6.3KB    1%    218.2KB/s   0    |
│sdag       4%  4752.9    24.0KB/s   90.4  52.9KB    7%   5631.6KB/s   0    |
│dm-3       2%   301.6     2.5KB/s   70.4   4.3KB    2%    304.0KB/s   0    |
│dm-6       7%  9509.8    26.0KB/s  180.2  52.9KB   12%  10923.8KB/s   0    |
│Totals Read-MB/s=19.2     Writes-MB/s=5.1      Transfers/sec=1474.3  s   0    |
```

Summary

- XIV Footprint: Architecture
- XIV Footprint: Environment
- The “Growth Problem”
- Virtualization, Cloud and Self-Service
- Performance in Transactional Workloads
- Linux and Multipath Tuning

THANK YOU!

will@redhat.com

RedHatVideos

youtube.com/RedHatVideos

Red Hat

facebook.com/RedHatInc

Red Hat

linkedin.com/company/3545

@RedHatNews

twitter.com/RedHatNews