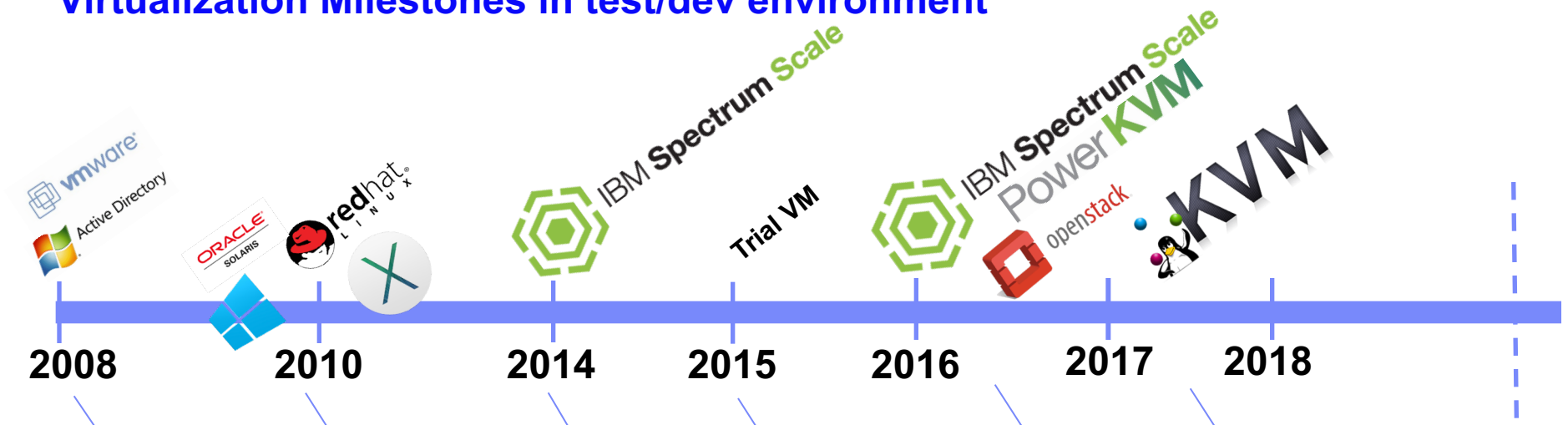
May 17th, 2018 GPFS User's Group (Boston)

Spectrum Scale Virtualization in Test and Development

Aaron Palazzolo (aspalazz@us.ibm.com)

Virtualization Milestones in test/dev environment



2008 — VMware Cloud created for SONAS Authentication test

2010 — Growth to include IO drivers & clients

2014 — 20+ Virtual Spectrum Scale clusters used by all scrum teams

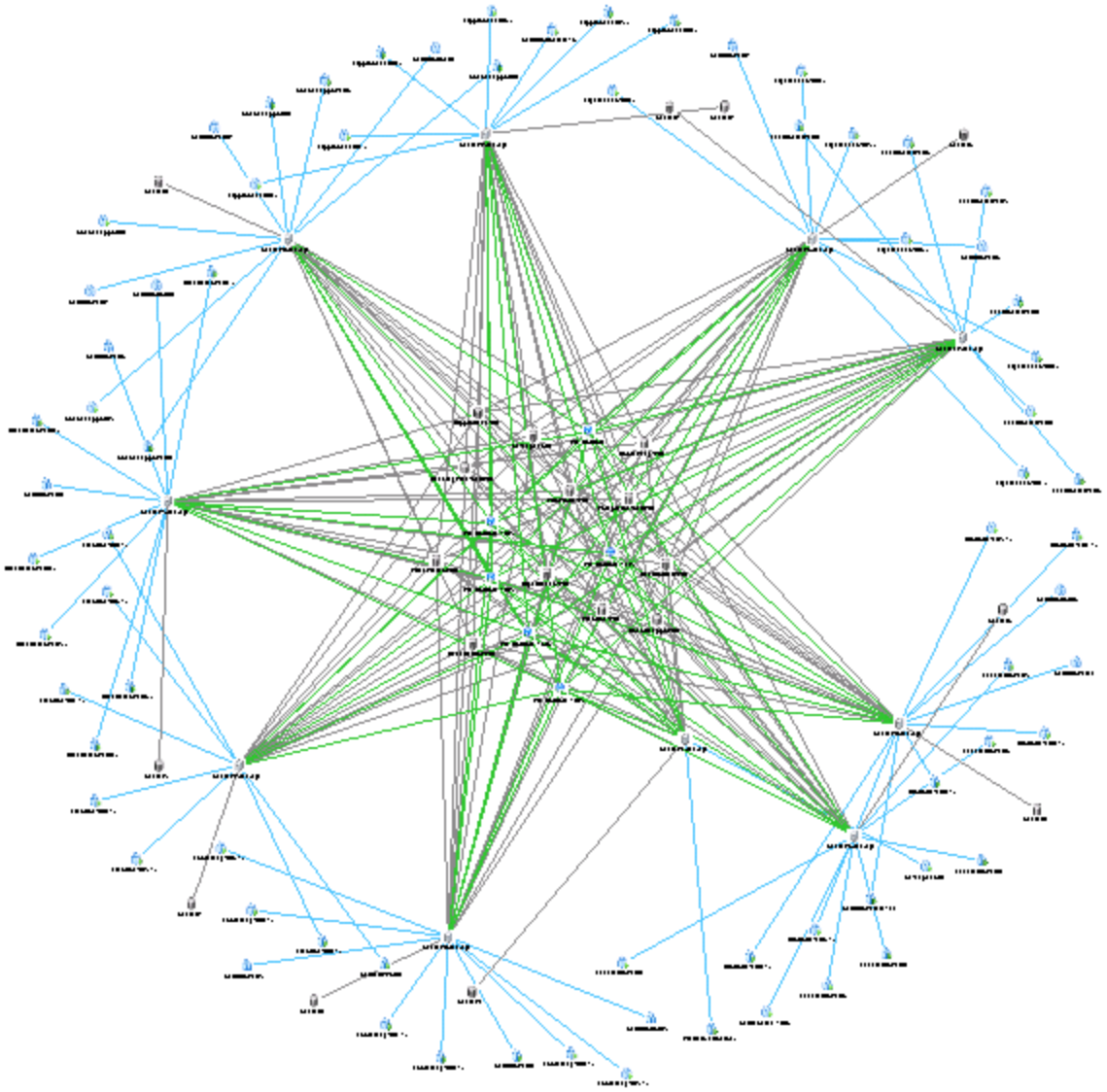
2015 — Spectrum Scale Trial VM created

2016 — Spectrum Scale + KVM + PowerKVM starting to replace VMware in Test/Dev

OpenStack RedPaper

2017 — Conversion to RHEL + KVM across both Power and X

VMware cloud – prior to transition to Spectrum Scale



How?

Use the pieces we already have

CISCO

Mellanox TECHNOLOGIES

DataDirect NETWORKS

BROCADE

vmware

IBM Spectrum Scale

PowerKVM

xiv

IBM Storwize V7000

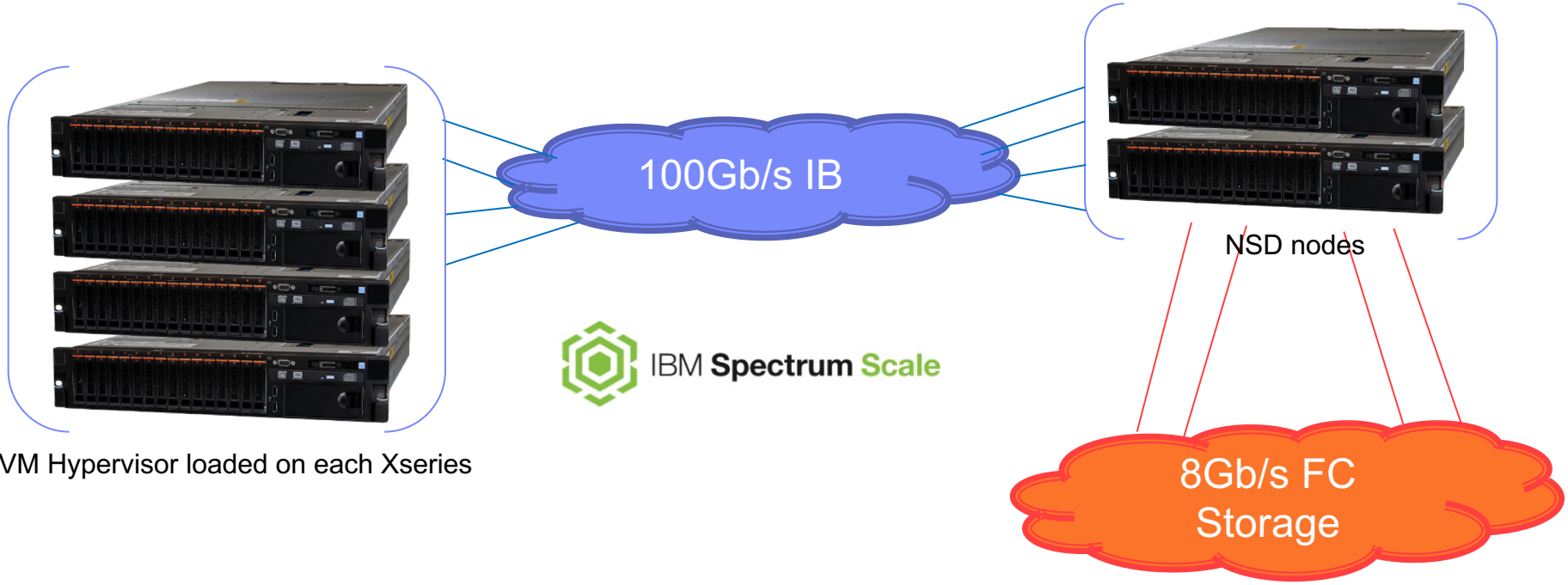
Power

NetApp

Lenovo

Build a big GPFS cluster

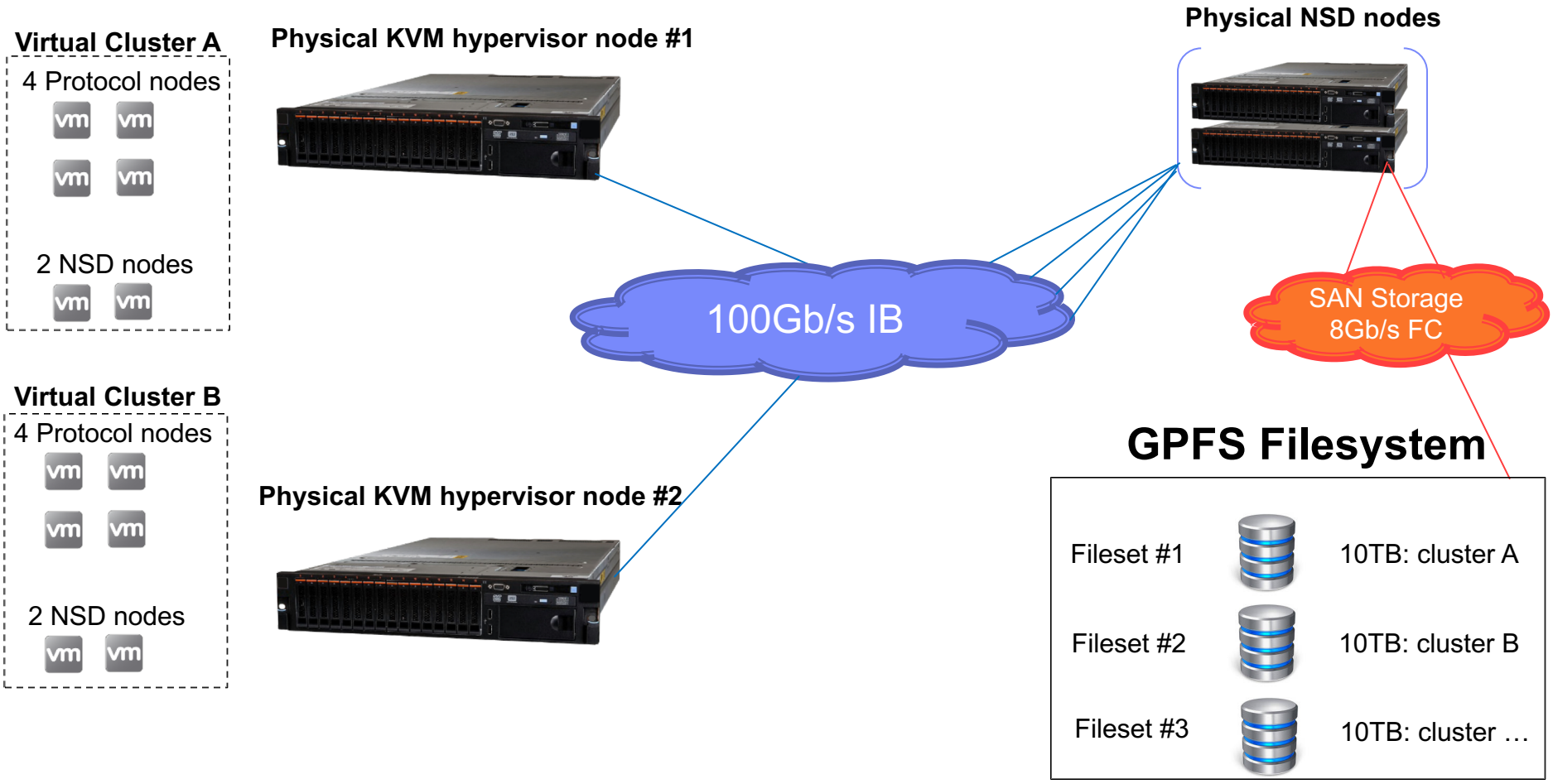
- ~100 Total Xseries GPFS nodes (Xseries M2, M3, M4, M5)
- 8 Power8 nodes
- >10TB of total Memory
- >600 core total CPU
- 100Gb/s IB network for GPFS cluster (*EDR mixed with some FDR and DDR*)
- 10Gb/s Ethernet network for I/O to VMs and for communication between VMs
- 8Gb/s Fibre Channel network (both direct and SAN) on NSD nodes
- ~30 NSD nodes



KVM Hypervisor loaded on each Xseries

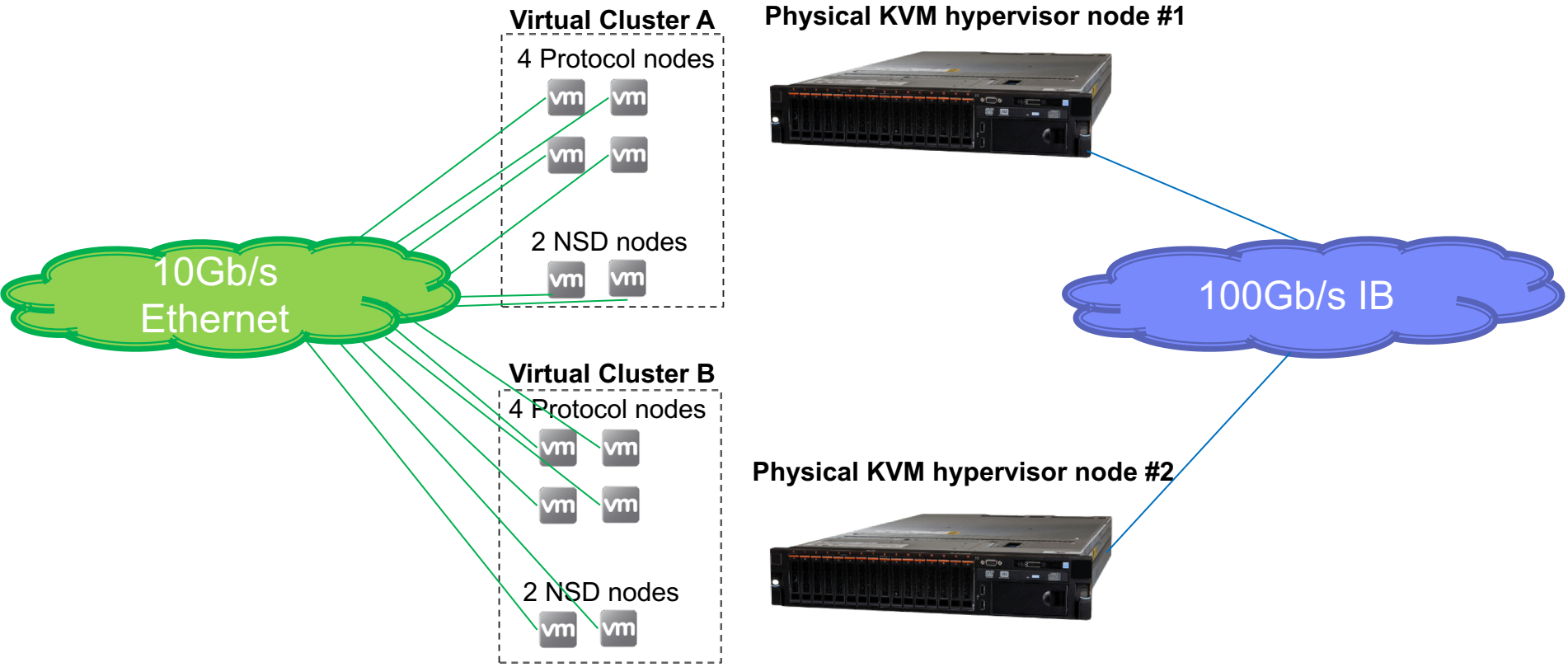
All nodes become VM hypervisors

- Each hypervisor node can run 1 or more VMs
- Possible to run one or more entire virtual GPFS clusters on a single hypervisor node
- Each Virtual GPFS cluster is stored in its own Fileset within the baremetal cluster's FS



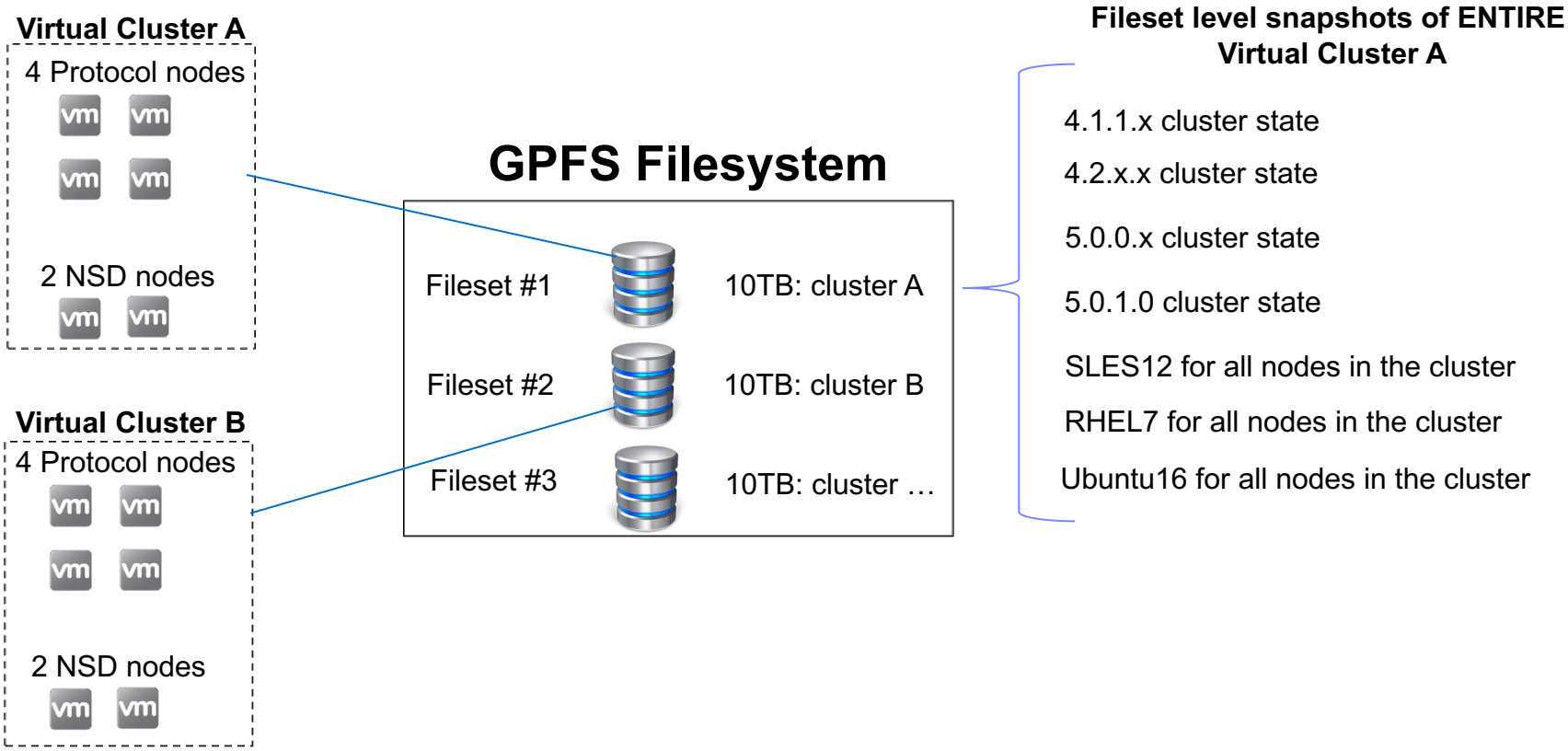
How do the VMs talk to each other?

- Each hypervisor node has at least 1 10GbE connection
- Each 10GbE link has access to multiple VLANs
- 1 or more VLANs are used for VM->VM GPFS admin/daemon network
- 1 or more VLANs are used for external access to the VMs *(ssh, SMB, NFS, OBJ, iSCSI)*



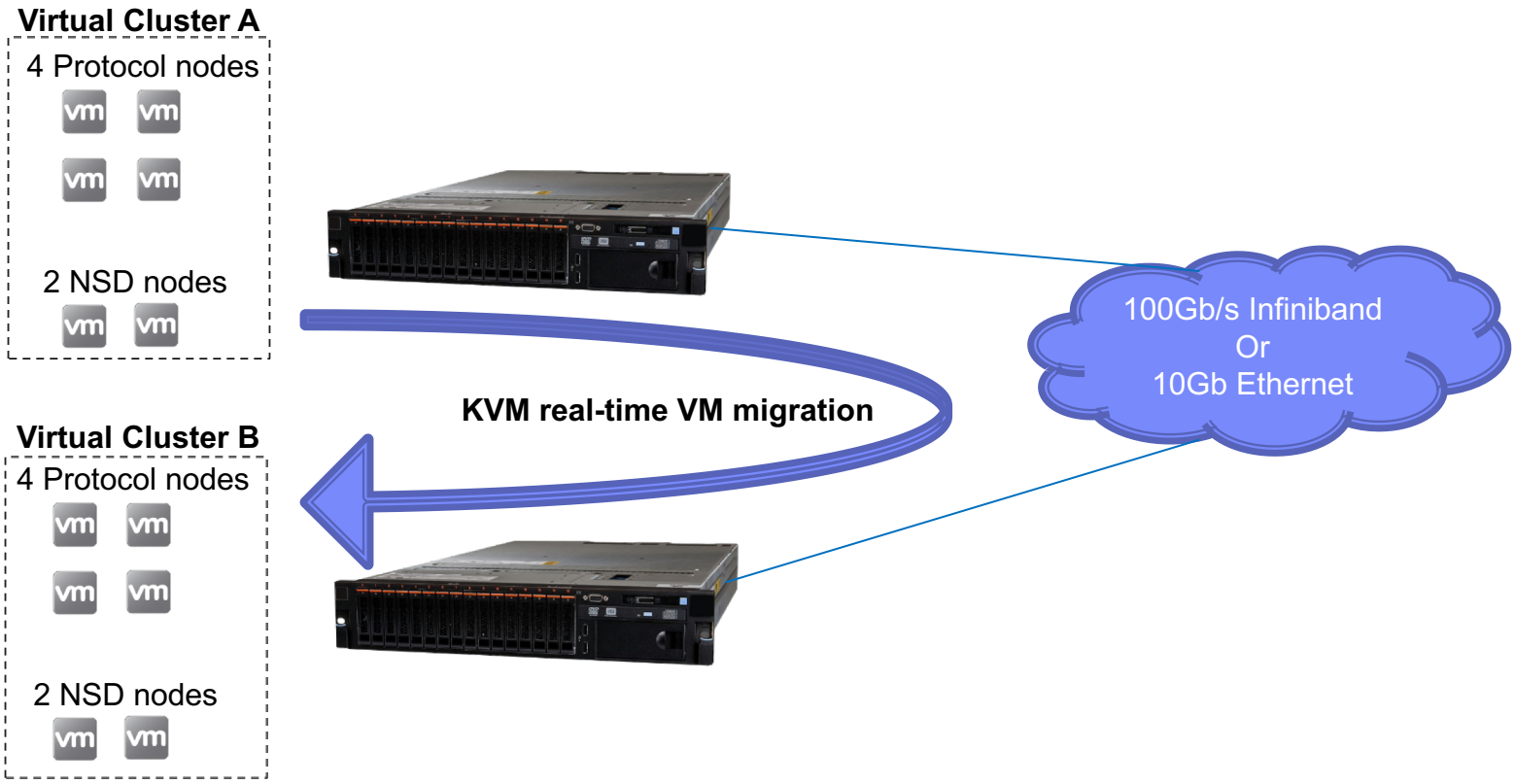
Why is this useful?

- Snapshots of any virtual cluster state
- Quick restore of prior cluster states
- Full use of hypervisor nodes instead of having them sit idle
- Using GPFS own functionality to run our test/dev environments (file clones, snapshots, independent filesets, sudo user, ...))



What happens if a physical hypervisor needs to be serviced?

- Simple: Move the VMs in real-time from one node to another



How-TO create a virtual GPFS cluster inside of GPFS

Method #1: Using GPFS for everything

1. Grab an OS iso image
2. Prepare a template by pre-installing the OS, adjusting everything you'd like, running virt-sysprep to change MACs, UUIDs, system ids, etc...
3. Create an independent file set
4. Copy the template VM to the file set
5. Clone the template for each VM using mmclone
6. Repeat cloning for each and every VM you want
7. Add virtual disks (NSDs) to each VM designated as an NSD server
8. Snapshot the new set of nodes using mmcrsnapshot
9. Power up all VMs

Method #2: Using external storage for snapshot operations

1. Create a single LUN on external storage (ex: XIV)
2. Setup multipath on all NSD nodes to see this single LUN
3. Create a single NSD from this LUN
4. Create a single Filesystem based off this NSD
5. Create VMs on this new Filesystem
6. Use XIV to snapshot... But first, poweroff VMs and unmount GPFS FS (or flush cache to disk).
7. Entire content of GPFS file system (VM cluster within) is snapshottable / restoreable in any state

Method #1: Hypervisor node setup

```
[root@basecluster-node1 ~]# mmlscluster

GPFS cluster information
========================
  GPFS cluster name:         basecluster.gpfs.net
  GPFS cluster id:           3460983905314557028
  GPFS UID domain:           basecluster.gpfs.net
  Remote shell command:      /usr/bin/ssh
  Remote file copy command:  /usr/bin/scp
  Repository type:           CCR

 Node  Daemon node name  IP address    Admin node name  Designation
-------------------------------------------------------------------
   1   basecluster-node1-ib    172.31.132.1  basecluster-node1-ib   quorum
   2   basecluster-node2-ib    172.31.132.2  basecluster-node2-ib   quorum
   3   basecluster-node3-ib    172.31.132.3  basecluster-node3-ib   quorum
   4   basecluster-node4-ib    172.31.132.4  basecluster-node4-ib   manager
   5   basecluster-node5-ib    172.31.132.5  basecluster-node4-ib   manager
   6   basecluster-node6-ib    172.31.132.6  basecluster-node6-ib   manager
   7   basecluster-node7-ib    172.31.132.7  basecluster-node7-ib   manager
```

- *GPFS cluster created on hypervisor nodes*
- *IB network used for admin/daemon network (IPoIB with RDMA enabled for GPFS)*

```
# brctl show
bridge name bridge id               STP enabled interfaces
br2059          8000.6805ca5790bd       no              ens5f1.2059             vnet1
br86            8000.6805ca5790bc       no              ens5f0                  vnet0
virbr0          8000.525400b56717       yes             virbr0-nic
```

- *Bridge to VMs for use with CES protocols (10.18.xx network)*
- *Bridge to VMs to use for virtual GPFS cluster creation/communication and CES protocols (9.11.86.xx network)*

Method #1: Listing all the VMs across the GPFS cluster

```
[root@basecluster-node1 ~]# mmdsh virsh list --all
basecluster-node1-ib:   Id    Name                           State
basecluster-node1-ib:  ----------------------------------------------------
basecluster-node1-ib:   1     VM-cluster1-nsd01              running
basecluster-node1-ib:
basecluster-node2-ib:   Id    Name                           State
basecluster-node2-ib:  ----------------------------------------------------
basecluster-node2-ib:   1     VM-cluster1-nsd02              running
basecluster-node2-ib:
basecluster-node4-ib:   Id    Name                           State
basecluster-node4-ib:  ----------------------------------------------------
basecluster-node4-ib:   1     VM-cluster1-client01           running
basecluster-node4-ib:
basecluster-node6-ib:   Id    Name                           State
basecluster-node6-ib:  ----------------------------------------------------
basecluster-node6-ib:   1     VM-cluster1-client02           running
basecluster-node6-ib:
basecluster-node5-ib:   Id    Name                           State
basecluster-node5-ib:  ----------------------------------------------------
basecluster-node5-ib:   1     VM-cluster1-ces01              running
basecluster-node5-ib:
basecluster-node3-ib:   Id    Name                           State
basecluster-node3-ib:  ----------------------------------------------------
basecluster-node3-ib:   1     VM-cluster1-ces02              running
basecluster-node3-ib:
basecluster-node7-ib:   Id    Name                           State
basecluster-node7-ib:  ----------------------------------------------------
basecluster-node7-ib:   1     VM-cluster1-ces03              running
```

This example shows a virtual cluster consisting of:

- *2 NSD nodes*
- *2 client nodes*
- *3 CES protocol nodes*

Method #1: GPFS File Systems on the Hypervisor nodes

```
[root@basecluster-node1 ~]# df -h
Filesystem                Size  Used Avail Use% Mounted on
/dev/mapper/system-root   542G  102G  441G  19% /
devtmpfs                   16G     0   16G   0% /dev
tmpfs                      16G  4.0K   16G   1% /dev/shm
tmpfs                      16G  226M   16G   2% /run
tmpfs                      16G     0   16G   0% /sys/fs/cgroup
/dev/sda2                 509M  170M  340M  34% /boot
/dev/sda1                  50M  9.8M   41M  20% /boot/efi
tmpfs                     3.2G     0  3.2G   0% /run/user/0
KVM-Sandbox1              9.0T  107G  8.9T   2% /gpfs/KVM-Sandbox1
KVM-Templates             2.7T   60G  2.7T   3% /gpfs/KVM-Templates
```

```
[root@basecluster-node1 ~]# ls /gpfs/KVM-Templates/
........
RHEL7.2_x86.raw   RHEL7.2_ppc64le.raw   RHEL7.2_ppc64.raw   SLES12_SP3_x86.raw      Ubuntu16.04_x86.raw
RHEL7.3_x86.raw   RHEL7.3.ppc64le.raw   RHEL7.3_ppc64.raw   SLES12_SP3_ppc64le.raw  Ubuntu16.04_ppc64.raw
RHEL7.4_x86.raw   RHEL7.4.ppc64le.raw   RHEL7.4_ppc64.raw   SLES12_SP3_ppc64.raw    Ubuntu16.04_ppcle64.raw
RHEL7.5_x86.raw   RHEL7.5.ppc64le.raw   RHEL7.5_ppc64.raw                           Ubuntu18.04_x86.raw
...... etc

RHEL7.2_x86.xml
RHEL7.3_x86.xml
...... etc

CentOS-7-x86_64-DVD-1511.iso
RHEL-7.4-20170711.0-Server-x86_64-dvd1.iso
RHEL-7.4-20170711.0-Server-ppc64-dvd1.iso
RHEL-7.4-20170711.0-Server-ppc64le-dvd1.iso
ubuntu-16.04-server-amd64.iso
ubuntu-16.04-server-ppc64el.iso
SLE-12-Server-DVD-ppc64le-GM-DVD1.iso
SLE-12-SP2-Server-DVD-x86_64-GM-DVD1.iso
...... etc
```

What do we keep in the KVM-Templates FS?

- *Pre-built OS templates for cloning*
- *Template .xml files*
- *OS ISOs*

Method #1: Filesets inside the Sandbox File System

```
[root@basecluster-node1]# mmlsfileset KVM-Sandbox1
Filesets in file system 'KVM-Sandbox1':
Name                     Status    Path
root                     Linked    /gpfs/KVM-Sandbox1
testing                  Linked    /gpfs/KVM-Sandbox1/testing
VM_cluster1              Linked    /gpfs/KVM-Sandbox1/VM_cluster1
VM_cluster2              Linked    /gpfs/KVM-Sandbox1/VM_cluster2
VM_cluster3              Linked    /gpfs/KVM-Sandbox1/VM_cluster3
VM_cluster4              Linked    /gpfs/KVM-Sandbox1/VM_cluster4
```

Each Virtual GPFS Cluster is contained in it's own GPFS fileset

- *See below for the file layout of VM_cluster1*

```
[root@basecluster-node1 ]# ls -ltr /gpfs/KVM-Sandbox1/VM_cluster1/
total 13985509380
-rw-r--r-- 1 root root  536870912000 Jan  5 13:01 sparse_VM-cluster1_RHEL7.4_2Master.raw
-rw------- 8 root root  536870912000 Jan  8 14:43 sparse_RHEL7.4_Master.raw
drwxr-xr-x 2 root root          4096 Apr 21 13:22 XML
-rw------- 1 root root  536870912000 May 15 03:10 VM-cluster1-client01.img
-rw------- 1 root root  536870912000 May 15 03:10 VM-cluster1-client02.img
-rw------- 1 root root  536870912000 May 15  2018 VM-cluster1-ces01.img
-rw------- 1 root root  536870912000 May 15  2018 VM-cluster1-ces02.img
-rw------- 1 root root  536870912000 May 15  2018 VM-cluster1-ces03.img
-rw------- 1 root root  536870912000 May 15  2018 VM-cluster1-nsd01.img
-rw------- 1 root root  536870912000 May 15 07:04 VM-cluster1-nsd02.img

-rw-r--r-- 1 root root 1825361100800 May  1 07:37 VM-cluster1-gpfs1_2.img
-rw-r--r-- 1 root root 1825361100800 May  1 07:37 VM-cluster1-gpfs1_6.img
-rw-r--r-- 1 root root 1825361100800 May 14 21:44 VM-cluster1-gpfs1_1.img
-rw-r--r-- 1 root root 1825361100800 May 14 21:59 VM-cluster1-gpfs1_4.img
-rw-r--r-- 1 root root 1825361100800 May 14 22:05 VM-cluster1-gpfs1_7.img
-rw-r--r-- 1 root root 1825361100800 May 14 22:09 VM-cluster1-gpfs1_8.img
-rw-r--r-- 1 root root 1825361100800 May 14 22:13 VM-cluster1-gpfs1_9.img
-rw-r--r-- 1 root root 1825361100800 May 14 22:13 VM-cluster1-gpfs1_3.img
-rw-r--r-- 1 root root 1825361100800 May 14 22:27 VM-cluster1-gpfs1_5.img
-rw-r--r-- 1 root root 1825361100800 May 15 07:04 VM-cluster1-gpfs0_1.img
-rw-r--r-- 1 root root 1825361100800 May 15 07:04 VM-cluster1-gpfs0_5.img
-rw-r--r-- 1 root root 1825361100800 May 15 07:04 VM-cluster1-gpfs0_3.img
-rw-r--r-- 1 root root 1825361100800 May 15 07:04 VM-cluster1-gpfs0_7.img
-rw-r--r-- 1 root root 1825361100800 May 15 07:04 VM-cluster1-gpfs0_9.img
-rw-r--r-- 1 root root 1825361100800 May 15  2018 VM-cluster1-gpfs0_6.img
-rw-r--r-- 1 root root 1825361100800 May 15  2018 VM-cluster1-gpfs0_4.img
-rw-r--r-- 1 root root 1825361100800 May 15  2018 VM-cluster1-gpfs0_2.img
-rw-r--r-- 1 root root 1825361100800 May 15  2018 VM-cluster1-gpfs0_8.img
```

VM disk files

Virtual GPFS NSD files

- *Attach to the VM NSD nodes*
- *Shared by both VM NSD nodes*
- *Notice names are for gpfs0 & gpfs1 – the VM NSD nodes will be hosting 2 FSs*

Method #1: Virtual Machine disk files are all clones

```
[root@basecluster-node1]# mmclone show *
Parent  Depth    Parent inode   File name
------  -----  --------------  ---------
                                VM-cluster1-gpfs0_1.img
                                VM-cluster1-gpfs0_2.img
                                VM-cluster1-gpfs0_3.img
                                VM-cluster1-gpfs0_4.img
                                VM-cluster1-gpfs0_5.img
                                VM-cluster1-gpfs0_6.img
                                VM-cluster1-gpfs0_7.img
                                VM-cluster1-gpfs0_8.img
                                VM-cluster1-gpfs0_9.img
                                VM-cluster1-gpfs1_1.img
                                VM-cluster1-gpfs1_2.img
                                VM-cluster1-gpfs1_3.img
                                VM-cluster1-gpfs1_4.img
                                VM-cluster1-gpfs1_5.img
                                VM-cluster1-gpfs1_6.img
                                VM-cluster1-gpfs1_7.img
                                VM-cluster1-gpfs1_8.img
                                VM-cluster1-gpfs1_9.img
   no       1          1581058  VM-cluster1-client01.img
   no       1          1581058  VM-cluster1-client02.img
   no       1          1581058  VM-cluster1-ces01.img
   no       1          1581058  VM-cluster1-ces02.img
   no       1          1581058  VM-cluster1-ces03.img
   no       1          1581058  VM-cluster1-nsd01.img
   no       1          1581058  VM-cluster1-nsd02.img
                                sparse_VM-cluster1_RHEL7.4_2Master.raw
  yes       0                   sparse_RHEL7.4_Master.raw
```

VM disk files are all clones of the master

Method #1: Fileset level snapshots of an entire VM cluster state

```
[root@basecluster-node1 gpfs0]# mmlssnapshot KVM_Sandbox1 | grep VM_cluster1
Snapshots in file system gpfs0:
Directory                                SnapId    Status  Created                   Fileset
VM-cluster1-blank                          1         Valid   Mon Aug 28 12:47:47 2017  VM_cluster1
VM-cluster1-rhel7.2_noNSDs                 2         Valid   Tue Aug 29 11:20:13 2017  VM_cluster1
VM-cluster1-rhel7.3_noNSDs                 3         Valid   Tue Aug 29 14:20:13 2017  VM_cluster1
VM-cluster1-rhel7.4_noNSDs                 4         Valid   Tue Aug 29 16:23:44 2017  VM_cluster1
VM-cluster1-rhel7.4_withNSD                5         Valid   Wed Aug 30 11:20:13 2017  VM_cluster1
VM-cluster1-Ubuntu16.04_noNSDs             6         Valid   Sat Sep  2 08:57:20 2017  VM_cluster1
VM-cluster1-Ubuntu16.04_withNSDs           7         Valid   Sat Sep  2 10:10:27 2017  VM_cluster1
VM-cluster1-rhel7.4_Ubuntu16.04_ESS_mix    8         Valid   Sat Sep  2 14:10:27 2017  VM_cluster1
VM-cluster1-rhel7.4_gpfs5.0.0.2            9         Valid   Sat Sep  2 18:10:27 2017  VM_cluster1
VM-cluster1-rhel7.4_gpfs5.0.1.0            10        Valid   Sat Sep  2 23:10:27 2017  VM_cluster1
```

Snapshots encompass the entire VM cluster

Test/Dev can easily jump between:

- *GPFS code versions*
- *OS levels*
- *Mixes of Oss*
- *Mixes of products (ESS / Scale combos)*

Key GPFS commands to create this:

- *mmcrfileset, mmlinkfileset, mmclone, mmcrsnapshot, mmrestorefs*

Method #1: Here's what the Virtual GPFS cluster looks like

```
# mmlscluster

GPFS cluster information
========================
  GPFS cluster name:         VM_cluster1.tuc.stglabs.ibm.com
  GPFS cluster id:           7871738103681496291
  GPFS UID domain:           VM_cluster1.tuc.stglabs.ibm.com
  Remote shell command:      /usr/bin/ssh
  Remote file copy command:  /usr/bin/scp
  Repository type:           CCR

 Node  Daemon node name                            IP address  Admin node name                            Designation
-------------------------------------------------------------------------------------------------------------------
   1   VM-cluster1-nsd01.tuc.stglabs.ibm.com      9.11.86.15  VM-cluster1-nsd01.tuc.stglabs.ibm.com      quorum-perfmon
   2   VM-cluster1-nsd02.tuc.stglabs.ibm.com      9.11.86.13  VM-cluster1-nsd02.tuc.stglabs.ibm.com      quorum-manager-perfmon
   3   VM-cluster1-client01.tuc.stglabs.ibm.com   9.11.86.14  VM-cluster1-client01.tuc.stglabs.ibm.com   quorum-manager-perfmon
   4   VM-cluster1-client02.tuc.stglabs.ibm.com   9.11.86.34  VM-cluster1-client02.tuc.stglabs.ibm.com   perfmon
   5   VM-cluster1-ces01.tuc.stglabs.ibm.com      9.11.86.36  VM-cluster1-ces01.tuc.stglabs.ibm.com      perfmon
   6   VM-cluster1-ces02.tuc.stglabs.ibm.com      9.11.86.37  VM-cluster1-ces02.tuc.stglabs.ibm.com      perfmon
   7   VM-cluster1-ces03.tuc.stglabs.ibm.com      9.11.86.38  VM-cluster1-ces03.tuc.stglabs.ibm.com      perfmon
```

```
# mmces service list -a
Enabled services: OBJ SMB NFS
VM-cluster1-ces01.tuc.stglabs.ibm.com:  OBJ is running, SMB is running, NFS is running
VM-cluster1-ces02.tuc.stglabs.ibm.com:  OBJ is running, SMB is running, NFS is running
VM-cluster1-ces03.tuc.stglabs.ibm.com:  OBJ is running, SMB is running, NFS is running

# mmlscluster --ces

GPFS cluster information
========================
  GPFS cluster name:         VM-cluster1.tuc.stglabs.ibm.com
  GPFS cluster id:           7871738103681496291

Cluster Export Services global parameters
-----------------------------------------
  Shared root directory:                /ibm/ces/ces
  Enabled Services:                     OBJ SMB NFS
  Log level:                            0
  Address distribution policy:          node-affinity

 Node  Daemon node name            IP address       CES IP address list
-----------------------------------------------------------------------
   5   VM-cluster1-ces01.tuc.stglabs.ibm.com 9.11.86.36       10.18.44.201 10.18.44.202 10.18.44.203 10.18.44.204 9.11.86.101 9.11.86.102 9.11.86.103
   6   VM-cluster1-ces02.tuc.stglabs.ibm.com 9.11.86.37       10.18.44.205 10.18.44.206 10.18.44.207 10.18.44.208 9.11.86.104 9.11.86.105 9.11.86.106
   7   VM-cluster1-ces03.tuc.stglabs.ibm.com 9.11.86.38       10.18.44.209 10.18.44.210 10.18.44.211 10.18.44.212 9.11.86.107 9.11.86.108 9.11.86.109
```

Networks that the VMs see

9.11.86.xx network

- *Used for GPFS cluster operation*
- *Used for CES IPs*

10.18.44.xx network

- *Used for CES IPs*

THE END