CONNECT

OpenVMS

Boot Camp 2011

September 18 – 22

Sheraton Hotel – Needham, MA

Evolving OpenVMS Environments:

An Exercise in Continuous Computing

Robert Gezelter, CSA, CSE

September 2011

As a courtesy to your fellow attendees,

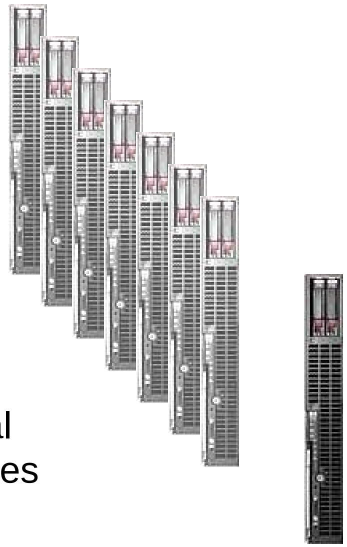
kindly set all devices (electronic and otherwise) to the silent or vibrate mode. If you receive a call, please leave the room to answer it.

This presentation represents my personal opinions. I welcome discussion of these concepts, publicly and privately.

Technical Trends:

SANS

Blades



Virtual Machines

Ubiquitous IP

Blades

- High density
- Common infrastructure, packaging
- Integrated SAN/LAN infrastructure
- Fungible computing elements

Blades

Storage Area Networks

- Modular storage
- Can be configured to be accessible to all systems
- Easy capacity upgrades
- Small physical footprint
- High performance

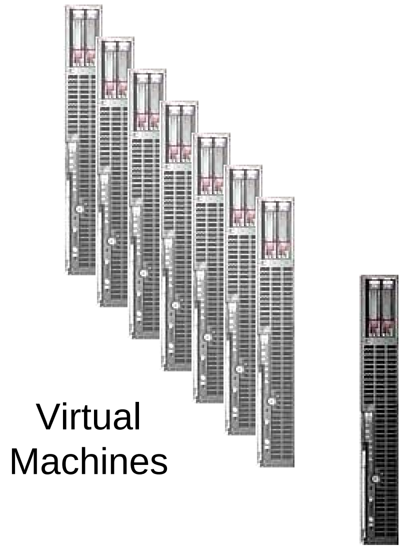
SANS

Ubiquitous IP Connectivity

- Common communications infrastructure
- High bandwidth (100 Mbps common)
- Quickly reconfigurable

Virtual Machines

- Ability to provide the appearance of more hardware than actually exists
- Consolidation of low utilization systems
- Some overhead for virtualization, most significant with high I/O applications



Invention v Discovery

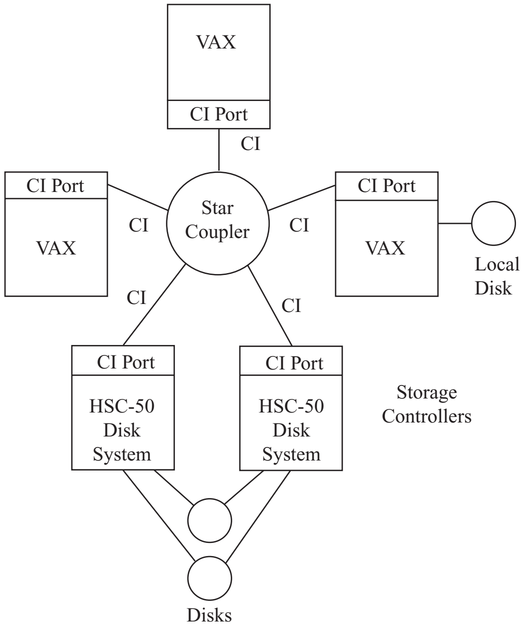
- There is a big difference
  - Discovery identifies something which was already there
  - Invention creates something new
- Discovery
  - William Herschel discovered Uranus
  - Marie and Pierre Curie discovered radium
- Invention
  - Thomas Edison invented the incandescent light
  - Samuel Morse invented the telegraph

What is a "Cloud"?

- A buzzword: "computing somewhere indefinite"
- Capabilities
- Applications Programming Interfaces

OpenVMS

- Quite possibly, the archetypical cloud
- Basic concept – User Virtualization
- OpenVMS clusters –
  - The archetypical peer cluster
  - load sharing
  - Shared file base
  - Resource sharing
  - Interchangeable machines



From Kronenberg, Levy, & Strecker, (1986)
VAXcluster: A closely-coupled distributed system
ACM Transactions on Computer Systems 4(2)

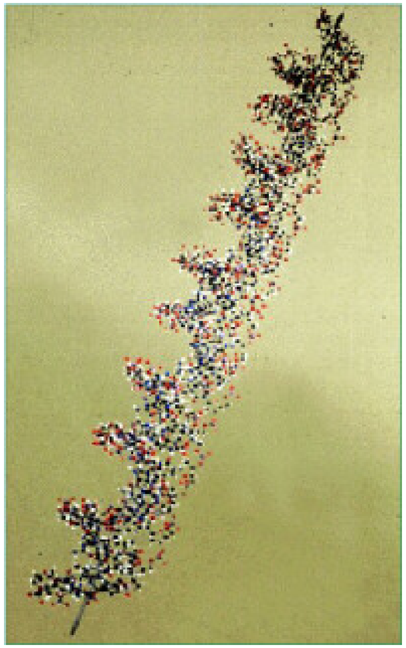
Original VAXcluster Environment

How do OpenVMS environments evolve in a context characterized by blades, SANS, IP, and Virtual Machines?

Evolution is always along the lines of existing pre-adaptations (e.g. consistent with small changes in genome)

The OpenVMS genome:

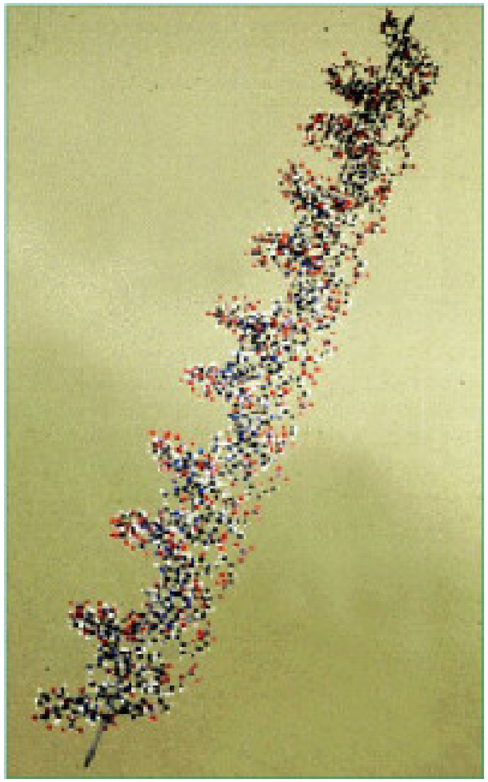
- User virtualization
- OpenVMS clusters
  - Single system image
  - Active-Active cluster
- Host Based Volume Shadowing



DNA Model, NIH Bldg. 38, Rotunda
Photo courtesy NIH

Evolving the OpenVMS Environment

- These basic architectural principles of OpenVMS are its DNA
- A new approach must build on these principals without violating them
- These principals define OpenVMS and its special value to OEMs and end-users alike



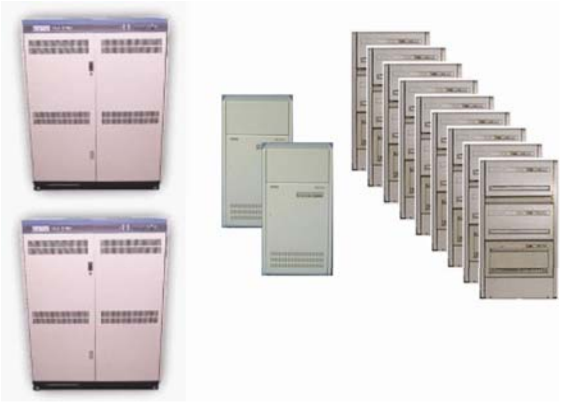
DNA Model, NIH Bldg. 38, Rotunda
Photo courtesy NIH

OpenVMS: The beginning

The context has changed but the basics are unchanging:

- Security
- Integrity
- File management
- Resource management

Remain completely valid.



This is unsurprising –

Good design is timeless. The term "legacy" is not relevant.

Consider aircraft design:

C-130A (circa 1951)

C-130K (circa 2006)

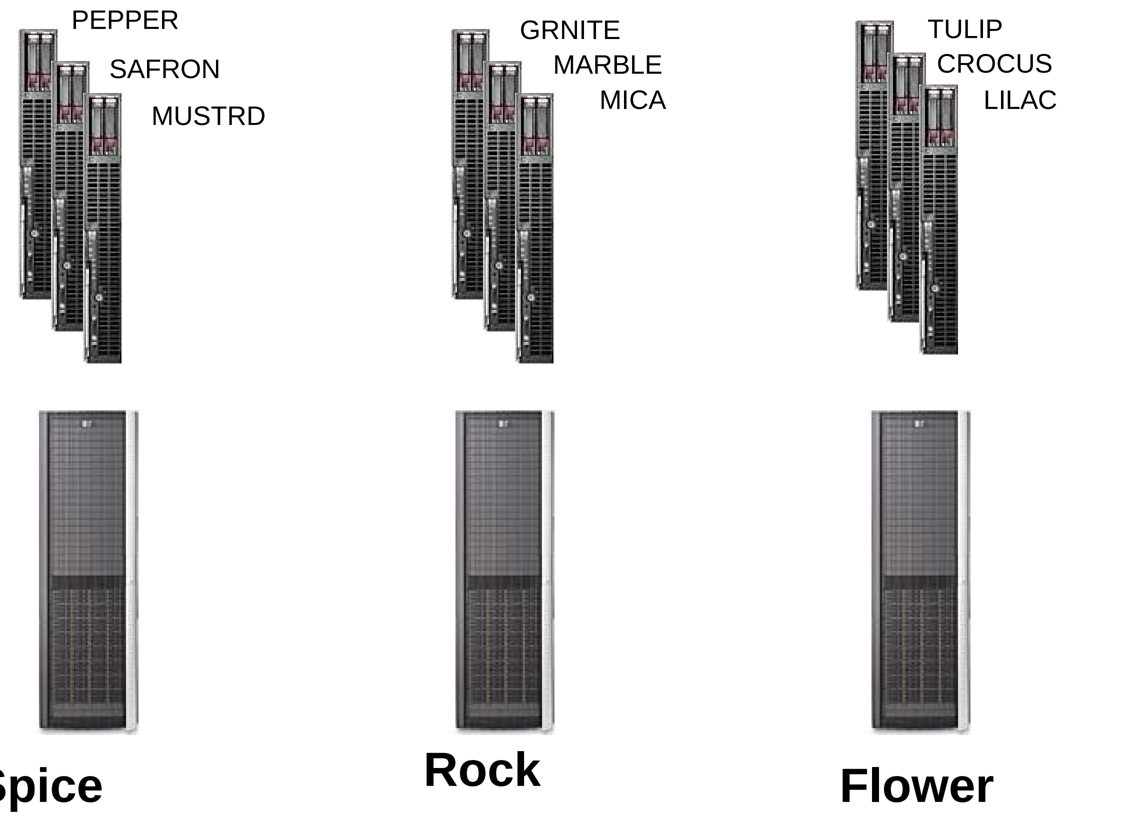
The current challenges are:

Use current technology trends to improve efficiency in:

- Power
- Space
- Capital
- Repair

Improvements in agility are byproducts.

Silos

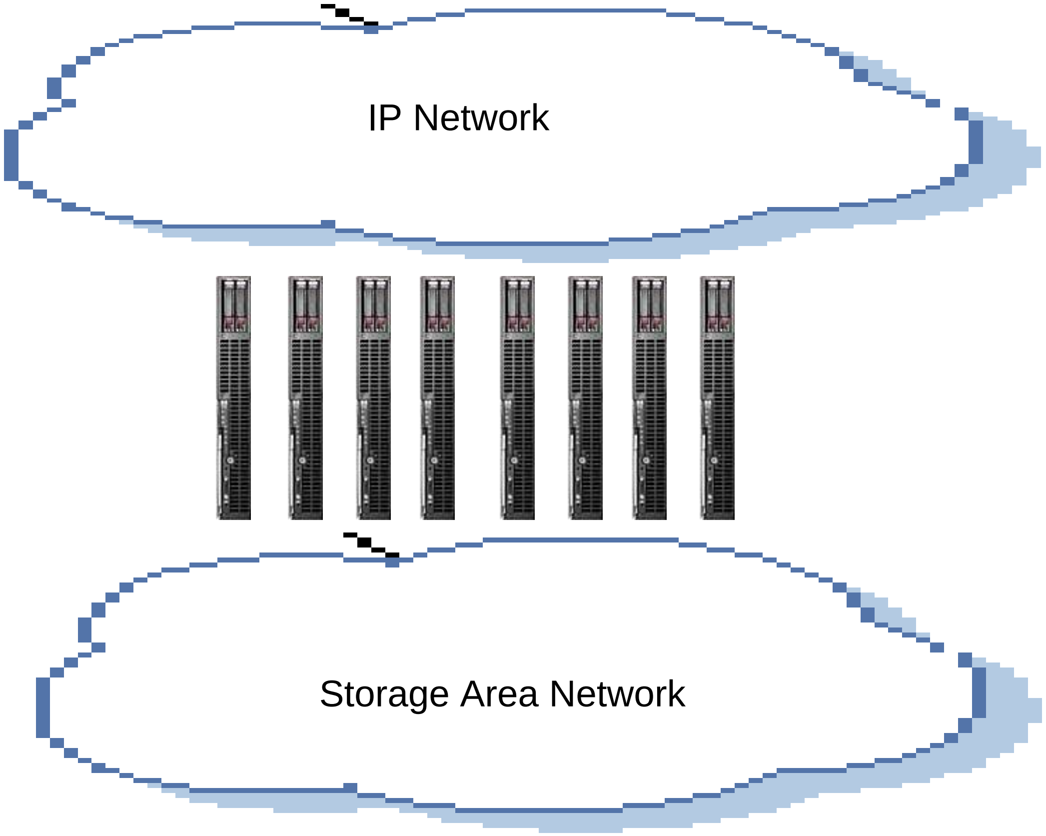


Silos obstruct efficiency in physical plant as much as in logical structure.

Rethink asset allocation

- Are silos appropriate?
- Can clusters be combined?
- What assets can be pooled?
- Common criteria
- OpenVMS dynamic paradigm:
  - Locks using Lock Manager
  - Load Balancing

The current world



Consider: What is this aircraft?

a) VC-25
b) Air Force One
c) SAM 28000
d) Boeing 747

What is this aircraft? The answer.

All, but (b). "Air Force One" is incorrect because:

All are "Air Force One", albeit at different times.

"Air Force One" is:

+

OpenVMS Architecture v. Doctrine

Architecture: How something is built

Doctrine: How something is employed

Reconsider how the OpenVMS architecture is employed in the context of technological developments and trends.

In the context of nodenames and OpenVMS clusters:

- Machine names
- Fixed machine assignment
- Ripe for re-thinking
- Use “Air Force One” example as type specimen

What distinguishes a machine:

Absent special hardware:

- Cluster Group Number
- SCSID / SCSNAME
- IP Address
- DECnet Node Address/Node Name

New paradigm: A cluster member is merely an incarnation of a given Cluster Group/SCSID/SCSNAME

- May be on a different blade/box
- It may be virtualized
- It may be running on a different root
- It may be running on a different volume
- System volumes can run all members on an architecture
- System volume distinguished by release/revision

Doctrinal Questions

- CPU pools
- HPVM (virtual machine)
- Surge Capacity
- System Volumes

Robert Gezelter
Software Consultant

CPU Pools

- Interchangeable blades
- Variants of common configurations
- CPU Pool like “motor pool”, assign as needed

ROBERT GEZELTER
SOFTWARE CONSULTANT

HP Virtual Machine and OpenVMS clusters

- Sub-capacity clusters
- Non-voting worker nodes
- Reserve capacity
- Quorum machines

Surge Capacity

- CPU pool and VM provide options
- Pre-configure add-on nodes
  - May be voting
  - May be non-voting (worker members)
- A way to address
  - Varying workloads
  - Unanticipated surges of demand

System Volumes

Present doctrine:

- Multiple roots
- Host-based Volume Shadowing, RAID sets
- Supplemental disks to avoid overloading

Mass Storage Technology has dramatically reduced economics

- OpenVMS clusters (1983)
  - RA81 – 10.5”, 100+ lb, 456MB
- Now
  - 146GB now less than US$ 500
  - 2.5” form factor
  - Ounces per drive
- Many orders of magnitude in
  - Cost
  - Capacity
  - Physical (size, weight)




System Volumes

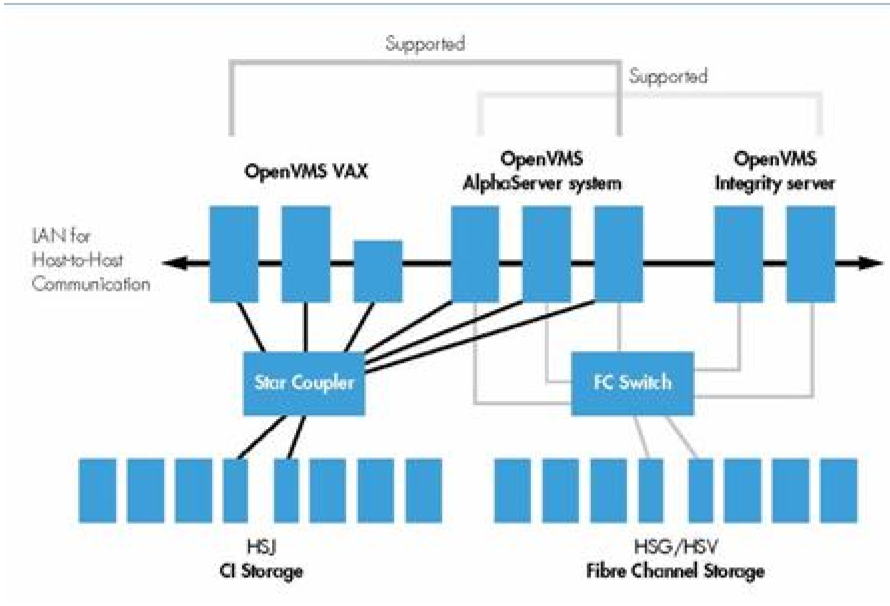
Reconsidered doctrine:

- Multiple roots
- Host-based Volume Shadowing, RAID sets
- Duplicate (in addition to Shadow/Mirror) system volumes
- Update cutover by changing volumes
- One system volume per release or patch level
- Always at least two copies of each system volume (**BEYOND** RAID/Shadowing)

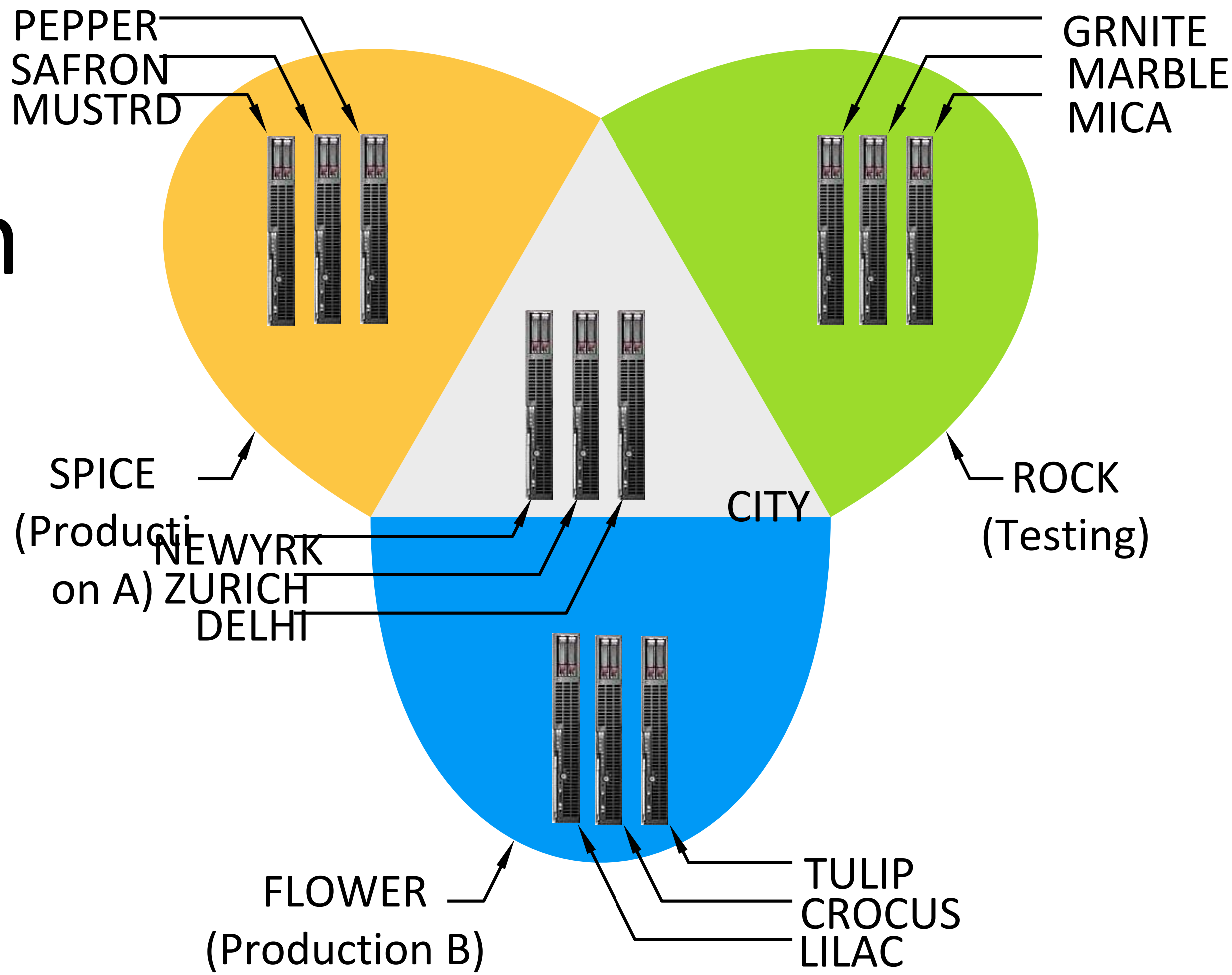
Canonical OpenVMS Cluster Doctrine

Relatively unchanged since 1982

- Provides high availability
- Full file sharing
- Failover

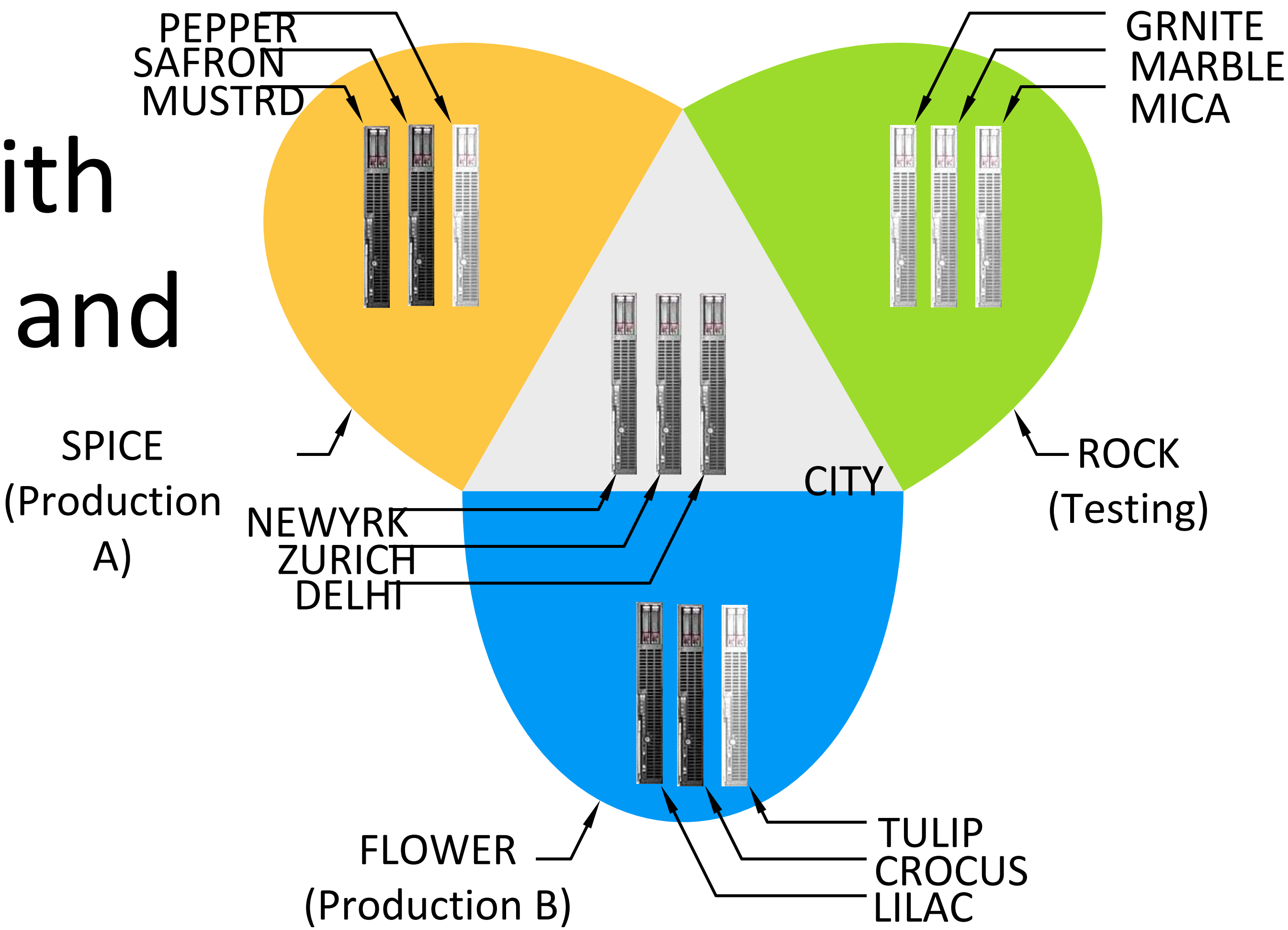


OpenVMS Clusters with CPU Pools

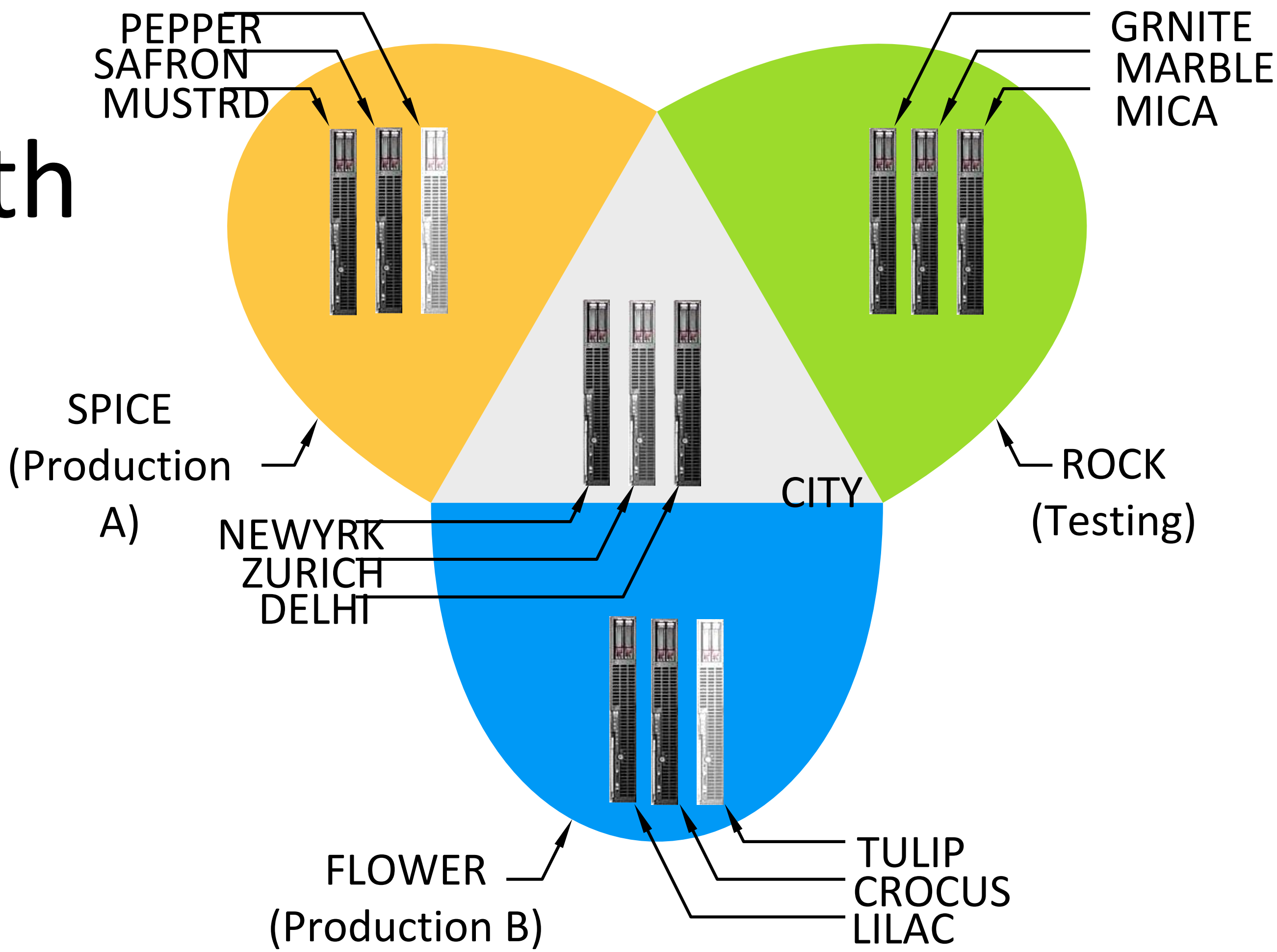


ROBERT GEZELTER
SOFTWARE CONSULTANT

OpenVMS Clusters with CPU Pools and Virtual Machines



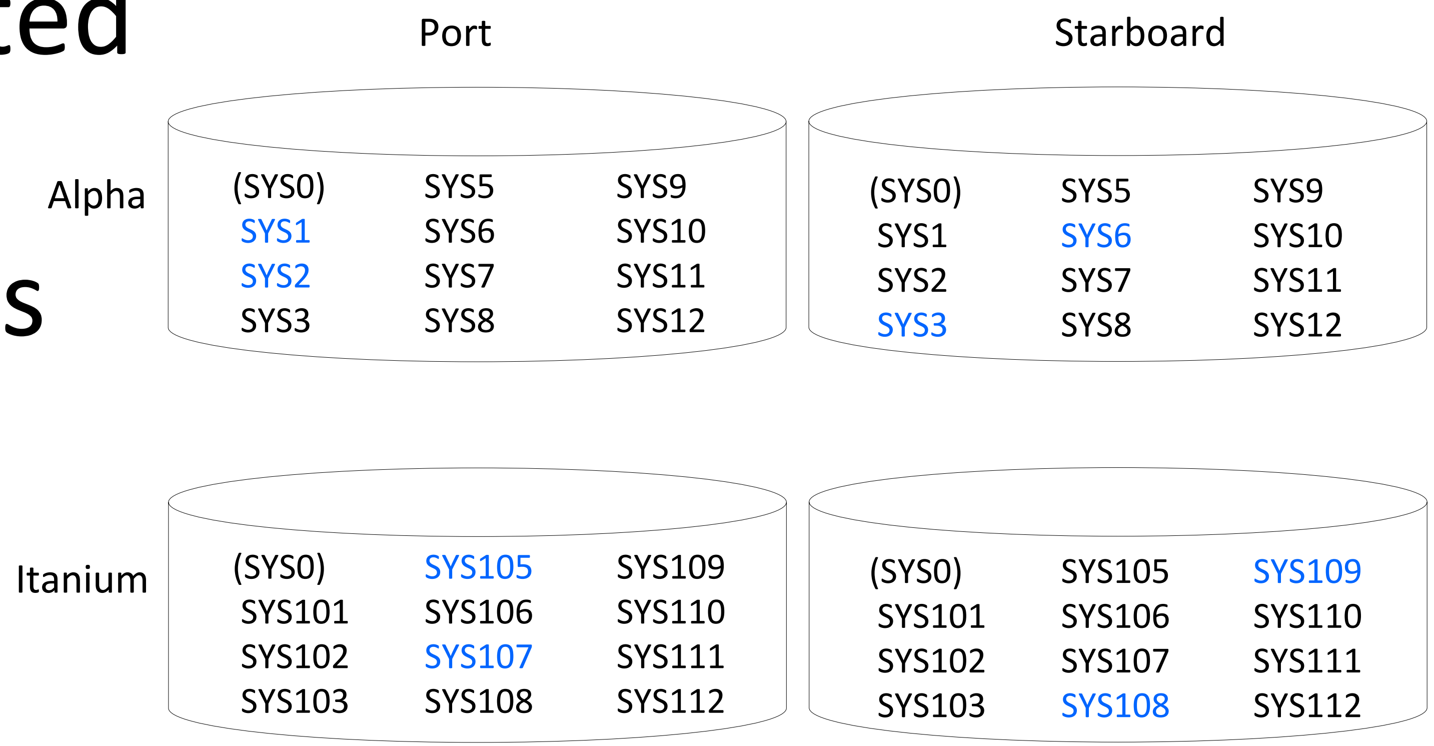
OpenVMS Clusters with VM Hosted Nodes as Quorum Nodes



ROBERT GEZELTER
SOFTWARE CONSULTANT

Connect OpenVMS Bootcamp 2011

Replicated System Volumes



Active System Root
Inactive System Root

ROBERT GEZELTER
SOFTWARE CONSULTANT

Operating Rules for System Volumes

- Mirror/Shadow (RAID 0)
- Split load between replicates
- Do upgrades on isolated blade housing, move drives/drive images
- Use rolling reboot to move to new patch or version

Operating Rules for System Volumes (continued)

- Staged rolling reboot
- Alternate boot roots per node

Logical names

- Create SYS$SITE_SPECIFIC
  - For multi-site clusters
  - See Gezelter, “Inheritance Based Environments for Standalone OpenVMS Systems and OpenVMS clusters”, OpenVMS Technical Journal, Volume 3

Logical names (continued)

- Create SYS$MEMBER_SPECIFIC
  – Inserted in search list between SYS$SPECIFIC and SYS$COMMON
  – Files common to all system roots for a particular cluster member

Summary

Current technological trends provide useful avenues for the evolution of OpenVMS Systems in increasingly flexible patterns of operations.

These changes do not need to sacrifice the strengths that customers rely on when they choose to use OpenVMS.

Virtual Machine technology complements existing OpenVMS paradigms.

Questions

Supplemental Materials, Slides:

http://www.rlgsc.com/openvms-bootcamp/2011/continuous-openvms.html