Linux for System z at Nationwide
From Woe to Whoa!
Part II
Where do we go from here?
Session 9213

Rick Barlow
Richard.Barlow@nationwide.com

August 15, 2006
SHARE 107

Overview and Disclaimer

Disclaimer:

The content of this presentation is for information only and is not intended to be an endorsement by Nationwide Insurance. Each site is responsible for their own use of the concepts and examples presented.

Overview:

With a few exceptions, this is an overview! Where possible there are technical details you may be able to use.
As you frequently hear, when anyone asks for recommendations, **"IT DEPENDS"**!
The information in this session is based on my experiences as a long-time VM-er adding virtual Linux.
Interactive is good! Please ask questions. We'll all get the most out of this session that way.

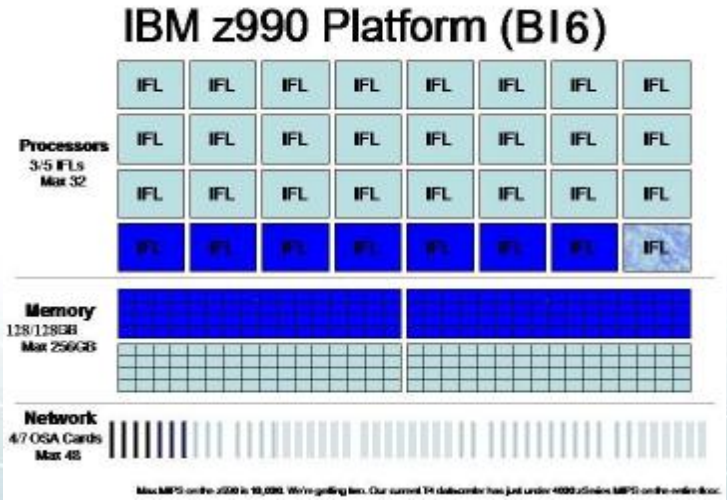
Topics

- Our Environment
- Why Virtualize Hardware?
- Virtual Networking
- High Availability
- Disaster Recovery Enablement
- Performance
- Conclusions

Our Environment

Environment

- 2 z990 installed in 2005, each with:
  - § ~~5~~ 8 IFL engines on development box
    ~~3~~ 7 IFL engines on production box
  - § ~~64~~ 128GB memory on development box
    ~~56~~ 128GB memory on production box
  - § 4 z/VM LPARs (1 additional test LPAR on development box)

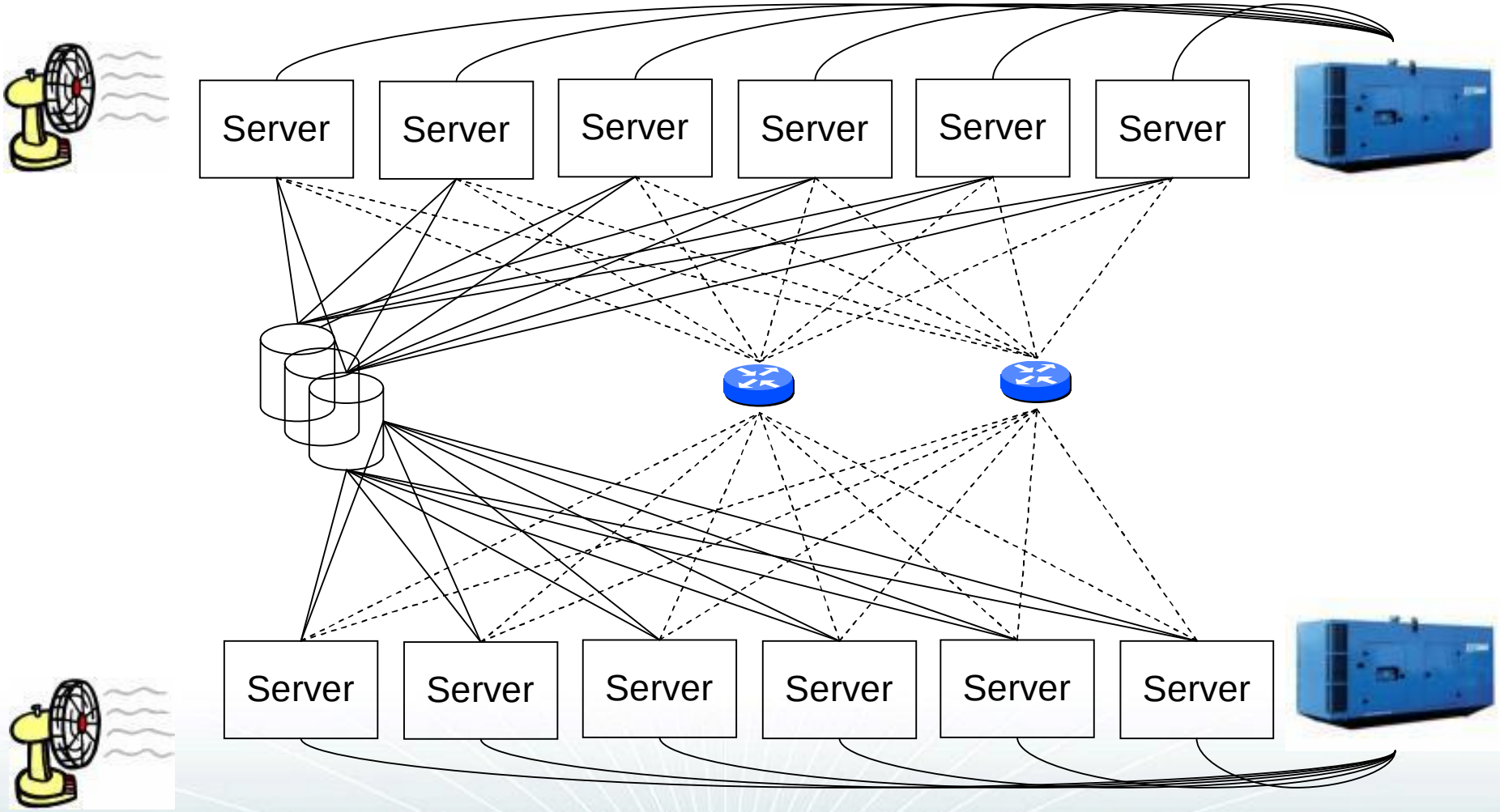


Why Virtualize Hardware?

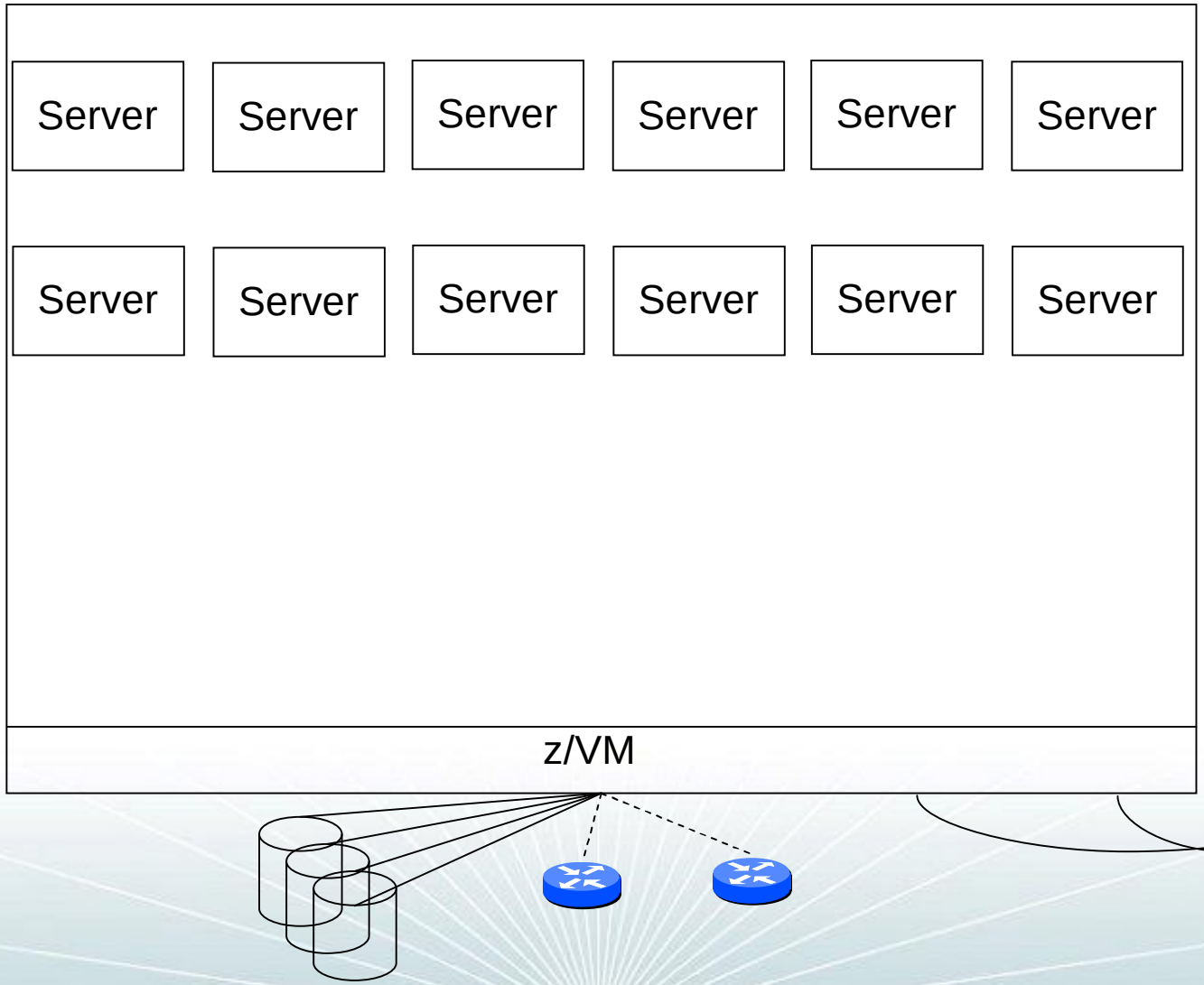
Why Hardware Virtualization?

- Reduce complexity
  - Physical servers
  - Network connections
  - Disk connections
- Reduce facility resources
  - Floor space
  - Power consumption
  - Cooling
- Opportunities
  - Shared disk
  - Shared memory
  - Reduced server capacity because of sharing

Distributed Server Model



Virtual Server Model



Virtual Networking

Learning

- Overcoming Terminology
  - VLAN, VLAN, Guest LAN
    - § VLAN – native, hardware, management – the one the routers, switches and OSAs use
    - § VLAN – logical – the ones used to separate/isolate servers
    - § Guest LAN – a VM emulation of a network
  - Switches, Routers , VSWITCH
    - § Switch – a device that acts as a connector to create a network
    - § Router – a device that forwards data packets between computer networks
    - § VSWITCH – a logical extension of the physical network inside the zSeries

Learning

- More Terminology
  - Layer 3, Layer 2
    - § Layer 3
      - o Forwarding based on IP address
      - o Sufficient for most implementations
    - § Layer 2
      - o Forwarding based on MAC address
      - o Allows non-IP protocols like NETBIOS or IPX

Learning

- zSeries Hardware
  - Open System Adapter (OSA) Express 2
    - § Gigabit adapter with a smart network controller
    - § zSeries LPAR microcode allows:
      - o Sharing of the same OSA across LPARs
      - o Multiple Read/Write/Data groups to be attached to virtual server or defined as a VSWITCH
    - § Gigabit Ethernet
      - o Fiber
    - § 1000BaseT
      - o Copper Cat6
      - o Can be configured as Integrated Console Controller (ICC)

Learning

- zSeries Hardware with z/VM
  - Virtual Switch (VSWITCH)
    - § Combination of zSeries microcode and z/VM CP code to create an extension of a network switch
    - § Layer 3
      - o Defined as "IP"
      - o Common MAC included for all guests
      - o z/VM 4.4.0 or higher

Learning

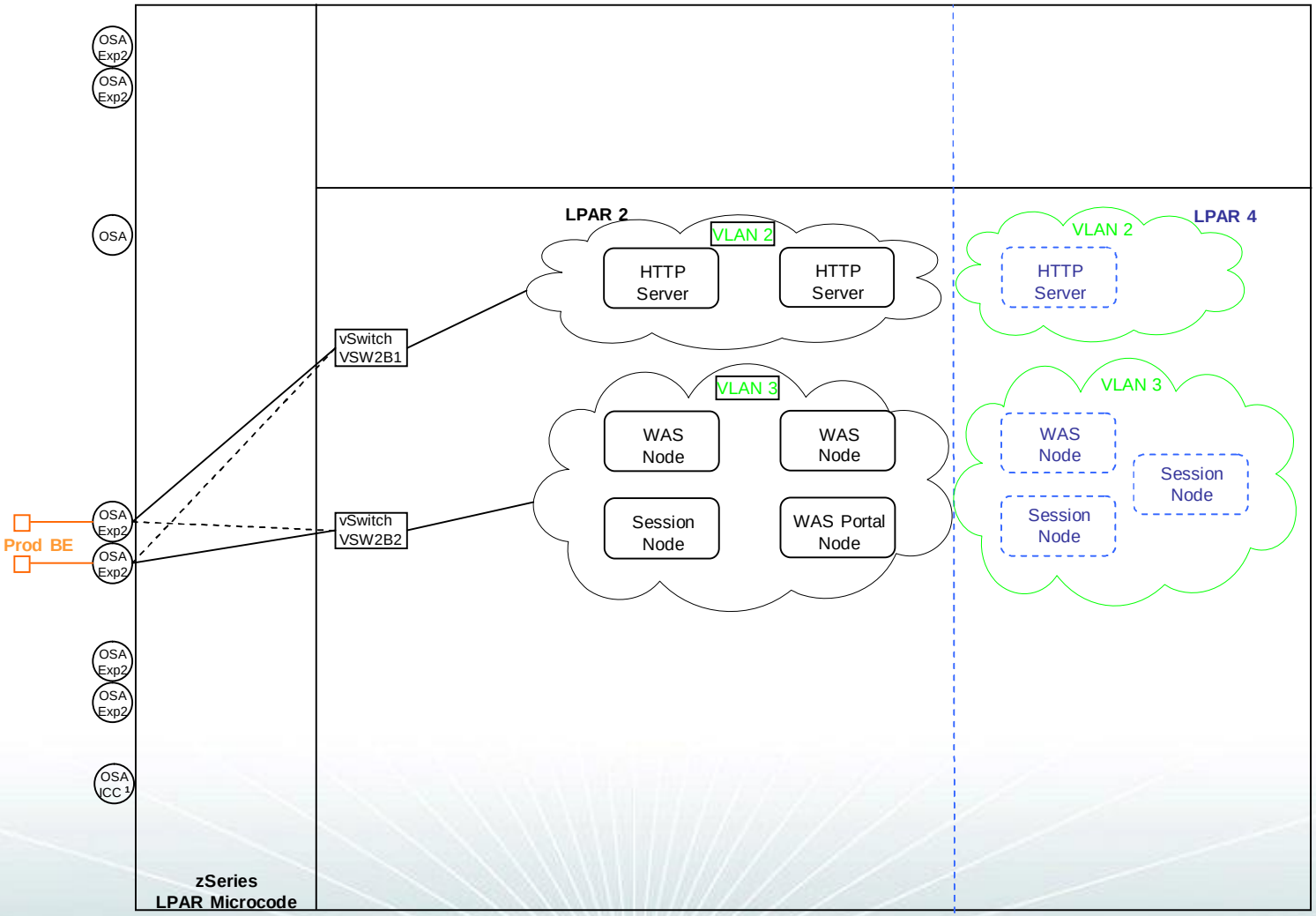
- zSeries Hardware with z/VM
  - Virtual Switch (VSWITCH)
    - Layer 2
      - Defined as "Ethernet"
      - New on z990 with OSA Express 2 and z/VM 5.1.0
      - Recommended by IBM
      - Unique MAC for each virtual server
        - Local MAC addressing must be administered
      - z/VM TCPIP cannot connect to a Layer 2 VSWITCH
      - Higher VM CPU
      - Allows “sniffing” by authorized guest

Learning

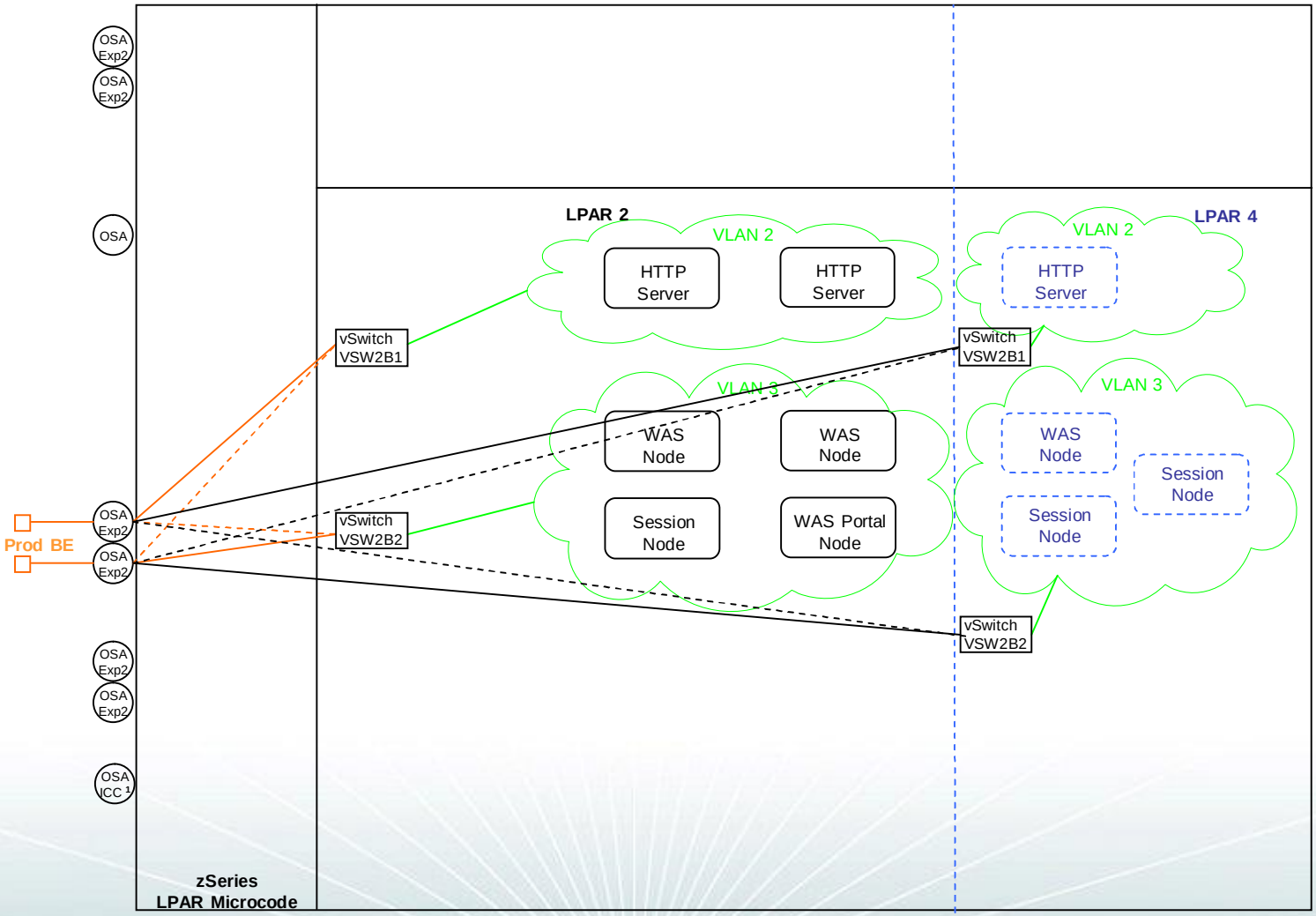
- z/VM
  - Guest LAN
    - § Use to create isolated LAN within a z/VM LPAR
      - o Can be owned by SYSTEM or a virtual machine
      - o Can be restricted to authorized users or open to anyone
    - § HIPERSOCKET – emulate zSeries HIPERSOCKET hardware
    - § QDIO – emulate gigabit ethernet
    - § Define in CP SYSTEM CONFIG or by CP command (syntax is the same)

```
DEFINE LAN GLAN1 OWNERID SYSTEM TYPE HIPERS MAXCONN INFINITE
DEFINE LAN GLAN2 OWNERID SYSTEM TYPE QDIO MAXCONN INFINITE
```

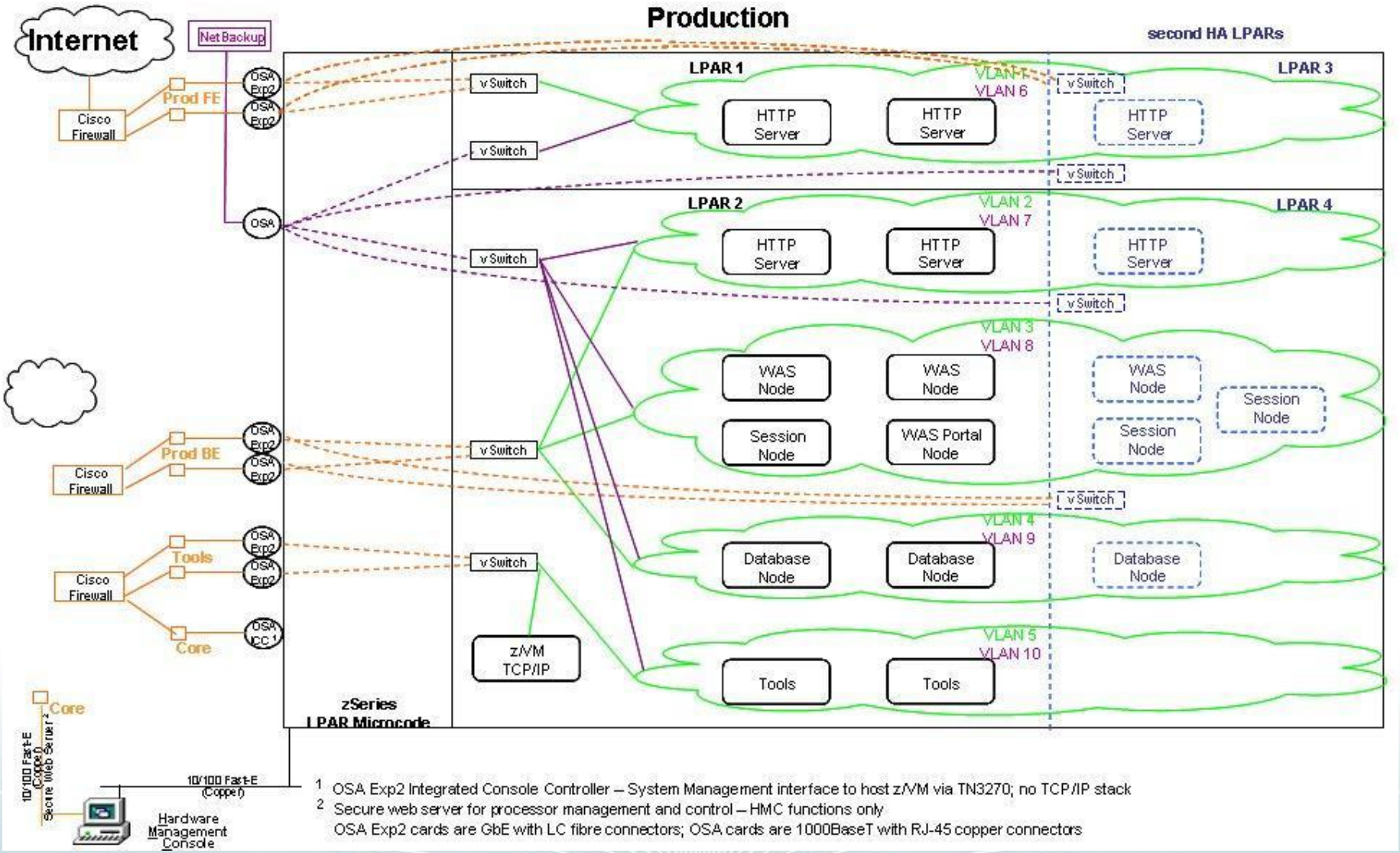
Network



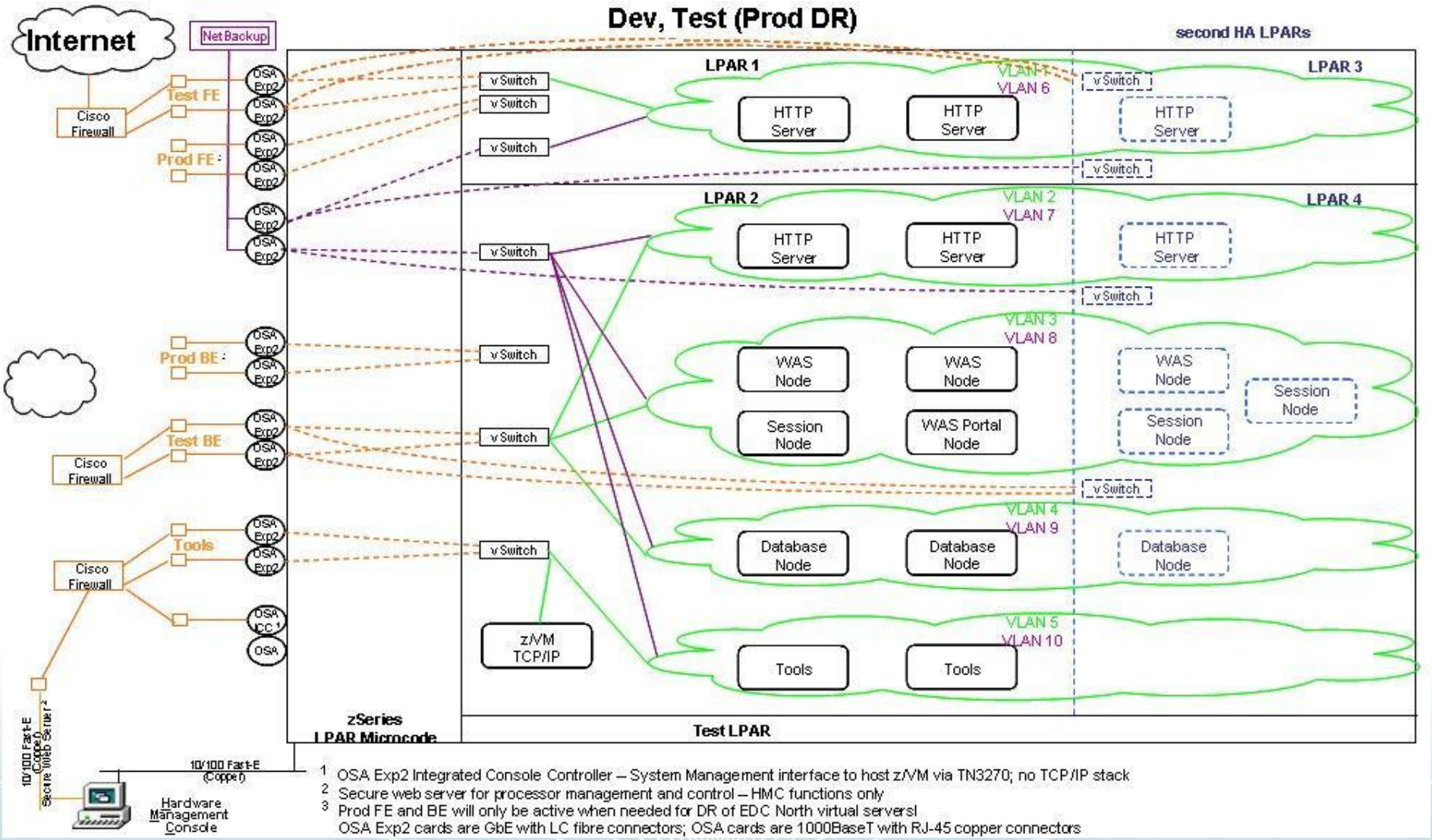
Network



Network


Production
second HA LPARs
Internet
Net Backup
Cisco Firewall
Prod FE
OSA Exp2
OSA
LPAR 1
LPAR 2
LPAR 3
LPAR 4
v Switch
HTTP Server
WAS Node
Session Node
WAS Portal Node
Database Node
Tools
z/VM TCP/IP
VLAN 1
VLAN 6
VLAN 2
VLAN 7
VLAN 3
VLAN 8
VLAN 4
VLAN 9
VLAN 5
VLAN 10
Prod BE
Tools
Core
OSA ICC 1
zSeries LPAR Microcode
10/100 Fast-E (Copper)
Secure Web Server 2
Hardware Management Console
1 OSA Exp2 Integrated Console Controller – System Management interface to host z/VM via TN3270; no TCP/IP stack
2 Secure web server for processor management and control – HMC functions only
OSA Exp2 cards are GbE with LC fibre connectors; OSA cards are 1000BaseT with RJ-45 copper connectors

Network



VSWITCH Detail

- Our OSA / VSWITCH configuration
  - 3 / 6 OSA Express 2 Gigabit Ethernet cards (6 / 12 Gb ports)
  - 1 OSA Express 1000BaseT (2 ports: 1 ICC)
  - 6 different network zones; 12 VSWITCHes defined
    - 2 VSWITCHes on each pair of OSA ports for redundancy and load distribution
      - Paired OSA ports are on separate cards for redundancy
    - Each pair of ports is in a specific network zone
      - Each OSA port in a pair is connected to a different physical switch

VSWITCH Detail

- Defining VSWITCH
  - In SYSTEM CONFIG or via CP command by authorized user (same syntax in both places)
    - Example of a pair of VSWITCHes:

```
CP DEFINE VSWITCH NWZONE1A RDEV C100 C204 CONTROLLER * IP VLAN 4094
CP DEFINE VSWITCH NWZONE1B RDEV C200 C104 CONTROLLER * IP VLAN 4094
```

    - VLAN on the VSWITCH is the default VLAN used by the **hardware** switches.

VSWITCH Detail

- Authorizing virtual servers to use VSWITCH
  - SYSTEM CONFIG format
    - Example 1: 2 virtual servers in same zone on opposite VSWITCHes

```
MODIFY VSWITCH NWZONE1A GRANT LINSERV1 VLAN 1001
MODIFY VSWITCH NWZONE2A GRANT LINSERV2 VLAN 1001
```

    - Example 2: 1 virtual server on 2 VSWITCHes in different zones

```
MODIFY VSWITCH NWZONE1A GRANT LINSERV1 VLAN 1001
MODIFY VSWITCH NWZONE2B GRANT LINSERV1 VLAN 2001
```

  - CP command format
    - Example 1: 2 virtual servers in same zone on opposite VSWITCHes

```
CP SET VSWITCH NWZONE1A GRANT LINSERV1 VLAN 1001
CP SET VSWITCH NWZONE2A GRANT LINSERV2 VLAN 1001
```

    - Example 2: 1 virtual server on 2 VSWITCHes in different zones

```
CP SET VSWITCH NWZONE1A GRANT LINSERV1 VLAN 1001
CP SET VSWITCH NWZONE2B GRANT LINSERV1 VLAN 2001
```

VSWITCH Detail

- Defining Guest NIC
  - CP DIRECTORY format

```
NICDEF 5708 TYPE QDIO DEVICES 3 LAN SYSTEM TOOL2
NICDEF 1E00 TYPE QDIO DEVICES 3 LAN SYSTEM NETBKUP1
```

  - CP command format

```
CP DEFINE NIC 5708 TYPE QDIO DEVICES 3
CP COUPLE 5708 TO SYSTEM TOOL2
CP DEFINE NIC 1E00 TYPE QDIO DEVICES 3
CP COUPLE 1E00 TO SYSTEM NETBKUP1
```

VSWITCH Detail – Linux definitions

- Hardware configuration script

```
cat /etc/sysconfig/hardware/hwcfg-qeth-bus-ccw-0.0.5708
#!/bin/sh
#
# hwcfg-qeth-bus-ccw-0.0.5708
#
# Hardware configuration for a qeth device at 0.0.5708
# Automatically generated by netsetup
#
STARTMODE="auto"
MODULE="qeth"
MODULE_OPTIONS=""
MODULE_UNLOAD="yes"
# Scripts to be called for the various events.
SCRIPTUP="hwup-ccw"
SCRIPTUP_ccw="hwup-ccw"
SCRIPTUP_ccwgroup="hwup-qeth"
SCRIPTDOWN="hwdown-ccw"
# CCW_CHAN_IDS sets the channel IDs for this device
# The first ID will be used as the group ID
CCW_CHAN_IDS="0.0.5708 0.0.5709 0.0.570a"
# CCW_CHAN_NUM set the number of channels for this device
# Always 3 for an qeth device
CCW_CHAN_NUM=3
# CCW_CHAN_MODE sets the port name for an OSA-Express device
CCW_CHAN_MODE="suselin7"
```

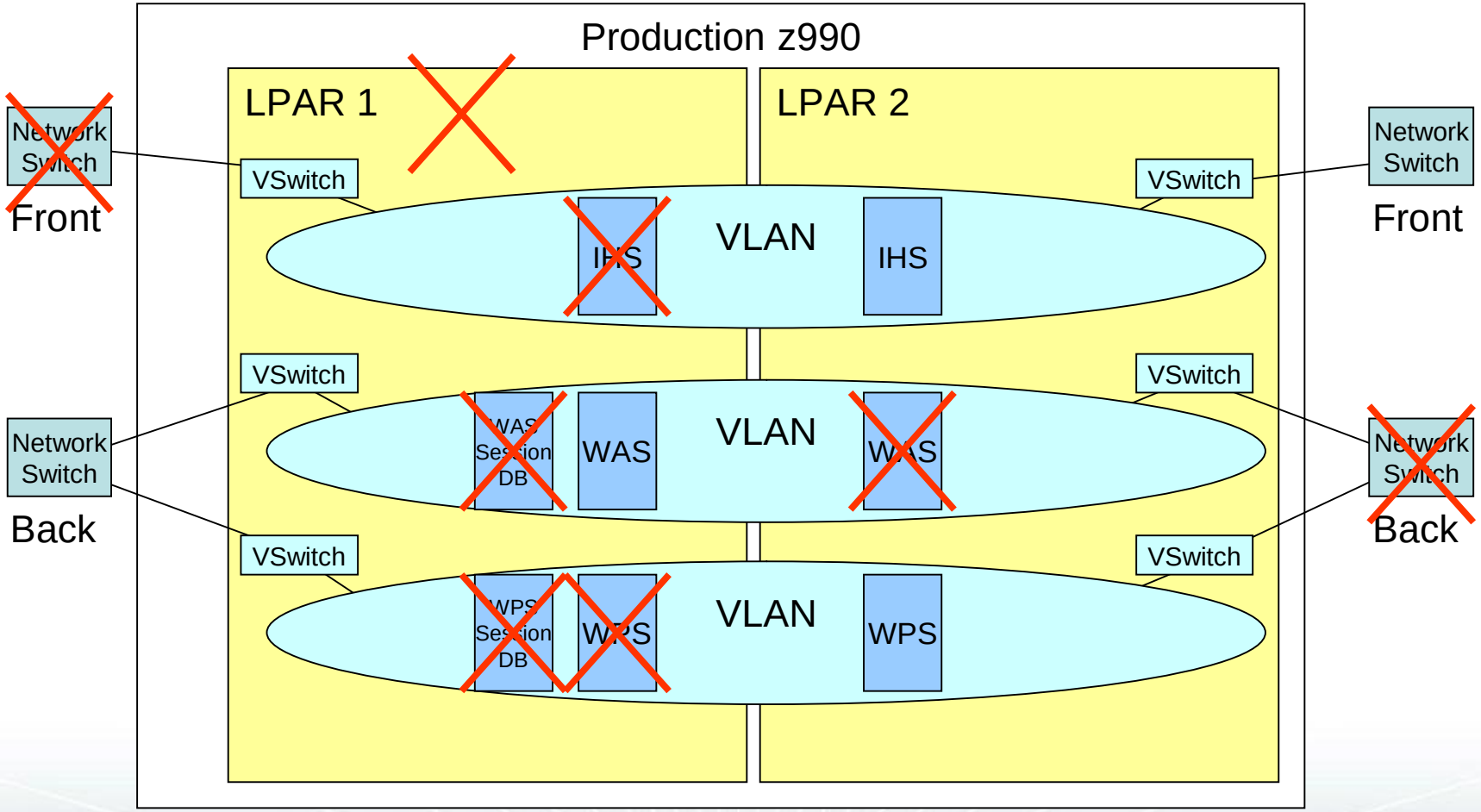
VSWITCH Detail – Linux definitions

- Confirmation

```
ifconfig
eth0      Link encap:Ethernet  HWaddr 02:00:00:00:00:05
          inet addr:10.220.228.4  Bcast:10.220.228.255  Mask:255.255.255.0
          inet6 addr: fe80::200:0:100:5/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 b)  TX bytes:652 (652.0 b)
eth1      Link encap:Ethernet  HWaddr 02:00:00:00:00:04
          inet addr:10.220.168.12  Bcast:10.217.70.255  Mask:255.255.255.0
          inet6 addr: fe80::200:0:100:4/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:150122 errors:0 dropped:0 overruns:0 frame:0
          TX packets:66742 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:32348101 (30.8 Mb)  TX bytes:17319537 (16.5 Mb)
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)
```

High Availability

High Availability Clustering



High Availability Clustering

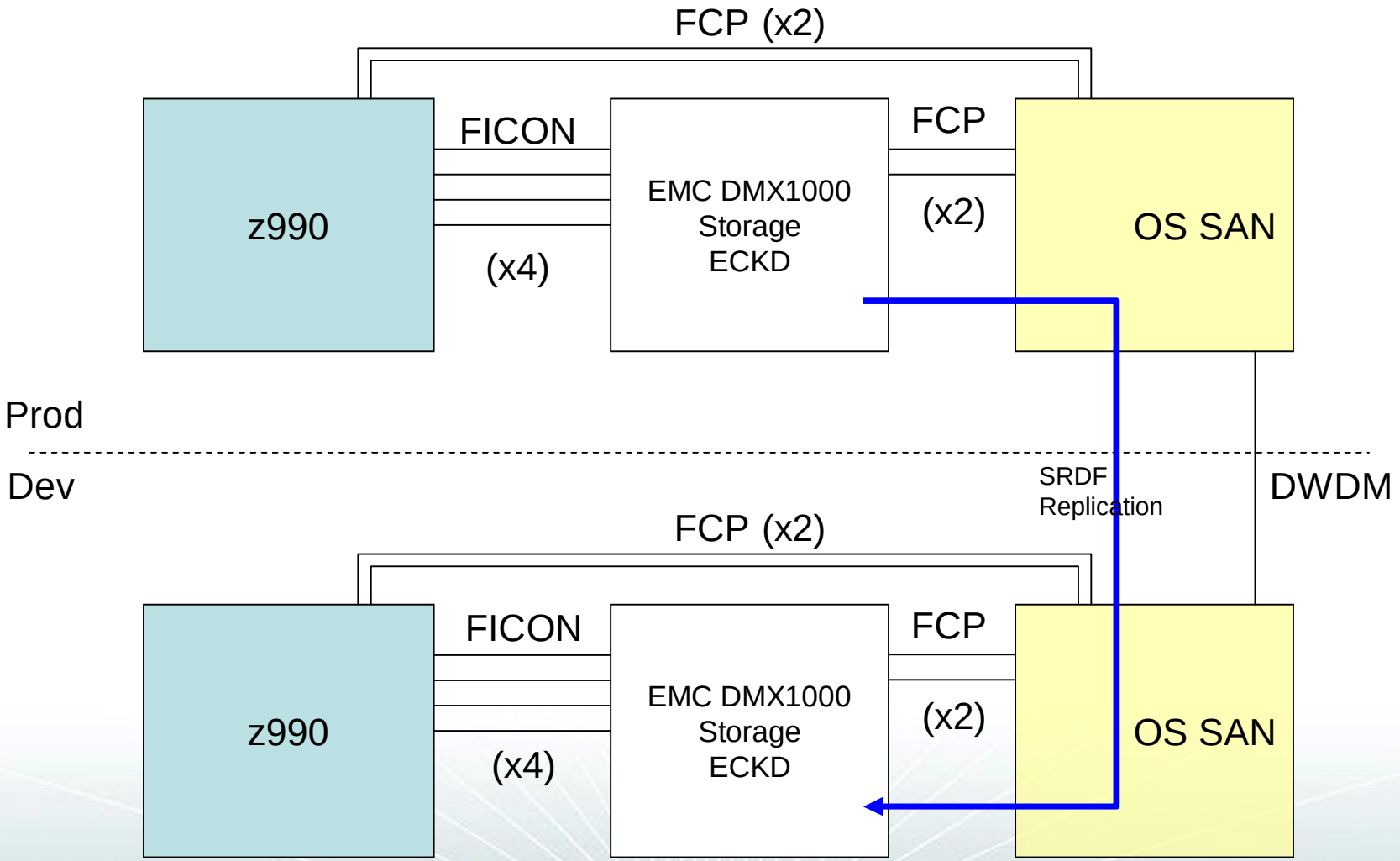
- Scenarios tested
  - Loss of clustered web server
  - Loss of network switch
  - Loss of clustered application server
  - Loss of entire z/VM LPAR
- Current Limitations
  - Single z990
    Increased availability if LPARs are spread across CPCs

Disaster Recovery Enablement

Disaster Recovery

- Included with High Availability offering
  - Disk replication between sites
  - Complete server definition (VM Directory) at second site
  - Physical network connections in place
  - Standby network definitions
  - Automated script for network personality at second site
    - Script on virtual server "asks" where it is running and sets network parameters
    - External DNS swap process must be performed
  - If primary site is unavailable, virtual servers are booted at second site

DR - Disk



Measuring Virtual Servers

Tools

Performance 001 (way less than 101)

- Basic metrics to watch – z/VM
  - CPU utilization
    - § While zSeries runs fine at 100%, Linux workload is much more demanding than traditional mainframe workloads. Keep peak periods at 85-90%.
  - Memory
    - § Many Linux guests have huge working set sizes and many don't go idle
    - § Keep memory over-commit less than 2:1 (ratio of combined working set sizes to real memory available)
  - Paging
    - § z/VM has no problem with high page rates
      - o Keep Expanded Storage for high-speed page buffer
    - § Guests may not be tolerant
    - § Allocate enough page space for twice the total of the working set of expected guests

Performance 001 (way less than 101)

- Basic metrics to watch – Linux Guests
  - Don't wake guests to ask
    - § Choose performance tools that understand that Linux is running on z/VM
  - Pick **one** tool
    - § Multiple monitoring tools adds a lot of overhead
    - § ½% CPU per server adds up fast when there are 100s of servers
  - CPU measured inside guest is not very meaningful (today – watch this space)
    - § Avoid TOP – significant overhead
    - § Use vmstat or nmon

Performance 001 (way less than 101)

- Basic metrics to watch – Linux Guests
  - Memory
    - § Don't over allocate. Large virtual storage sizes drive up z/VM paging.
    - § Use a swap hierarchy with z/VM VDISK as the highest priority swap space. It is not a problem for Linux to do some swapping.
    - § Show all snapshot of memory/swap: free or cat /proc/meminfo
    - § Avoid multiple caching
      - o DB2: Use directio=yes to prevent it from doing its own I/O caching and rely on Linux
    - § Default Linux memory management may not be optimal
      - o Kernel parm: vm.swapiness=60
        Default may be too high – causes memory to be consumed
        Lower values cause Linux to reuse memory allocations more often to reduce memory demand
  - Paging
    - § Prevent Linux from paging. z/VM paging is much more efficient.
    - § Show Linux pagein/pageout: cat /proc/vmstat | grep pgpg

Performance 001 (way less than 101)

- Basic metrics to watch – Linux Guests
  - Look at guest CPU demand from z/VM
  - Watch for excessive paging on behalf of a guest.
    - § May indicate inefficient memory usage or excessive virtual storage allocation
  - Watch for guests with poor I/O response
    - § zSeries handles high I/O rates fine but bottlenecks can occur
  - Watch for % of active time that guests spend in various queues
    - § Run
    - § CPU queue
    - § Page queue
    - § etc

Performance 001 (way less than 101)

- Linux Guests internal performance
  - Tools to analyze guests functions vary greatly
    - Some have a lot of tools – WAS
    - Some have little to offer – other purchased software
  - Application developers debugging skills may be limited
    - Accustomed to working with excessive capacity
    - Not accustomed to shared environment

Performance 001 (way less than 101)

- Ideas that may help
  - Enable the timer patch!
  - Utilize Cryptographic hardware
    - § Dramatically improves SSL calls in ssh and scp
      - o Moving a 165MB tar ball went from 430K/sec to 1.2M/sec
  - Minimize external network hops
    - § Use virtual firewall solutions
    - § Staying inside the zSeries hardware operates at memory speeds
  - Turn off NTP (or only run occasionally)
  - Minimize or stagger cron scheduling

Performance Future Options

- Cooperative Memory Management
- Fixed I/O Buffers
- Execute In Place (xipfs)
- Shared Read-Only disks
  - Requires separation of code from configurations or perhaps use of union mount
- DCSS – shared code in z/VM storage

Conclusions

Conclusions

- Linux virtualization on zSeries can:
  - Reduce complexity
  - Improve provisioning time
    - § No hardware acquisition
    - § No physical installation to perform
  - Reduce environmental demand
    - § Less cooling
    - § Less power
    - § Less floor space

Conclusions

- Things are changing rapidly
- Performance is as much an art as a science
- Be careful what you ask for because you may get it!

References

- Other sessions this week:
  - Tue 08:00 9125 Virtual Networking with z/VM Guest LANs and the z/VM Virtual Switch
  - Tue 01:30 9112 z/VM TCP/IP Stack Configuration
  - Wed 11:00 9131 TCP/IP Routing
  - Thu 03:00 9216 The Virtualization Cookbook: Day 1 - z/VM
  - Thu 04:30 9217 The Virtualization Cookbook: Day 2 - Linux
- Documentation
  - REDP3719 **Linux on IBM eServer zSeries and S/390: VSWITCH and VLAN Features of z/VM 4.4**

Contact Information

"And I thought we were busy *before Linux showed up!*"

Rick Barlow
Systems Engineering Consultant

Phone: (614) 249-5213

Internet: Richard.Barlow@nationwide.com