Data networking with OpenVMS

OpenVMS Bootcamp 2017

Session 237

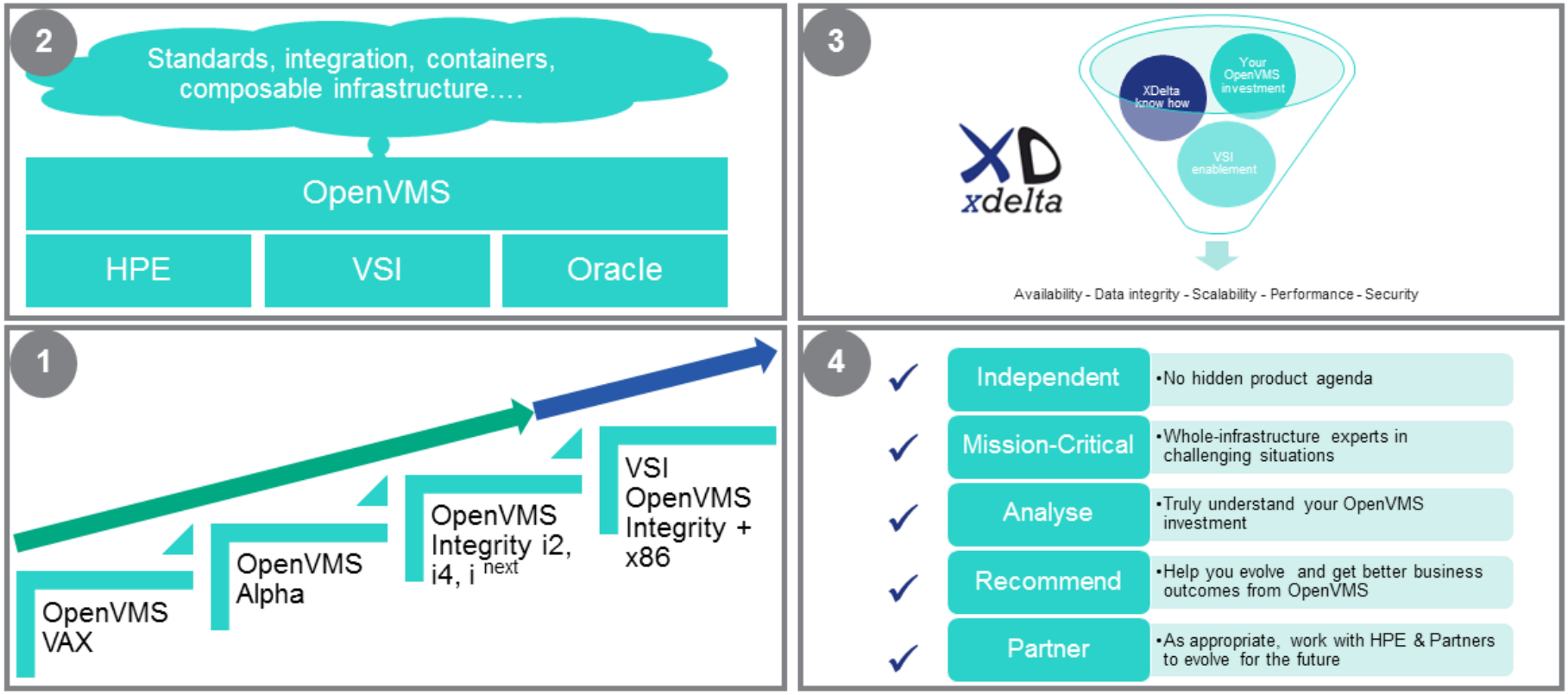
Colin Butcher CEng FBCS CITP
Technical director, XDelta Limited
www.xdelta.co.uk

XDelta: Who we are

vms Software

- VSI Professional Services Alliance member
- Independent consulting engineers since 1996:
  - UK based with international reach
  - Delivering OpenVMS based systems for 30+ years
- Technical leadership for business-critical systems
  - Design, planning and implementation
  - Mentoring and skills transfer
  - Systems engineering background
- Gartner (2009):
  - Identified XDelta as one of few companies world-wide capable of OpenVMS platform migration projects

XDelta - a trusted advisor to advance your critical OpenVMS application infrastructure



Hewlett Packard Enterprise

Agenda

- Data network infrastructure
- Network protocols
- OpenVMS network connectivity
- Integration with other systems

Part1: Basic principles

- Ethernet packet format
- Infrastructure components

Basic principles: Data networks

- Local-Area Networks (LANs)
  - Ethernet technologies
  - Physical components and cabling
  - Protocols and addressing
  - Network Interfaces
  - Network Switches
- LAN segmentation
- LAN extension
- Wide-Area networks (WANs)

Basic principles: OSI seven-layer model

7	Application	Provides for distributed processing and access, contains application programs and supporting protocols (eg FTAM)
6	Presentation	Coordinates conversion of data and data formats to meet the needs of the individual applications
5	Session	Organises and structures the interactions between pairs of communicating applications
4	Transport	Provides reliable transparent transfer of data between end systems with error recover and flow control
3	Network	Permits communication between network entities
2	Data link	Specifies the technique for moving data along network links between defined points on the network, and how to detect and correct errors in the Physical layer (layer 1)
1	Physical	Connects systems to the physical communications media

Basic principles: TCPIP V4 four-layer model

- Layer 4 – Layer 3 sub-protocol specific, identified by TCPIP port number
- Layer 3 – TCPIP V4 addressing and routing layer, needs protocol address to MAC address translation
- Layer 2 – MAC address layer, Ethernet V2 or IEE802.3 format packets.
- Layer 1 – Physical layer (transmission media)

Network infrastructure: Cabling

- Transmission properties, transmitter components and receiver components are important - a signal at one end needs to be recognisable at the other end

- Copper:
  - Co-axial (thick-wire, thin-wire)
  - Twisted pair (Category 5, 5E, Category 6 etc.)

- Fibre-optic:
  - Monomode (typically 9 micron)
  - Multimode (typically 50 or 62.5 micron)

Network infrastructure: Ethernet

- 10 Mbit/sec
- 100 Mbit/sec (Fast ethernet)
- 1,000 Mbit/sec (Gigabit ethernet)
- 10,000 Mbit/sec (10Gigabit ethernet)
- Copper / fibre (different transmission characteristics)

- Wireless ethernet (WiFi)
  - Access control and data privacy are major issues

Network infrastructure: NICs / switch ports

- NIC = Network Interface Card
- LoM = LAN on motherboard
- ASIC = Application Specific Integrated Circuit

- Provide connection between IO subsystem and network
- Copper / fibre / wireless physical interfaces
- Processing capability:
  - Address filtering
  - Packet processing (Checksum offload – CKO)
  - Protocol processing (TCP/IP offload - TOE)
  - Needs operating system / firmware support

Network infrastructure: Ethernet packets

- Hardware MAC address
- Physical MAC address
- Broadcast address
- Multicast addresses
- Point to point addresses
- Ethernet packet format v IEEE802.3 packet format
- Packet size (normal frames and jumbo frames)

LANCP> SHOW DEVICE /CHAR
SDA> SHOW LAN

Network infrastructure: Segmentation

Why segment a network?

- Availability
- Performance
- Security
- Separation of traffic flows

How can you segment a network?

- Multiple NICS in systems
- Switches
- VLANs
- Routers

History: Repeaters

- Layer 1 devices (“flat” network)
- Provide electrical fault isolation
- Simply re-time and re-transmit signal
- No control of bandwidth
- Beware of cumulative end to end delay exceeding maximum permissible frame timing – which leads to ‘folklore’ such as the “three repeater rule”

- *TIP: Beware of the generic term “hub”*

History: Bridges

- Packet content based (Layer 2)
- Store and Forward
- Easy to use and configure
- Poor control of bandwidth (filtering)
- Spanning tree algorithm
- Provides an extended LAN
- Not all protocols can tolerate the inherent delays in working over an extended LAN
- Remote booting (MOP, BOOTP etc.) will absorb bandwidth

Network infrastructure: Switches

- Introduces parallelism
- Speed of chipsets (latency & bandwidth)
- Full duplex operation on a single device per port basis
- Traffic monitoring (mirror ports)
- Link aggregation
- Bandwidth control
- “Store and forward” versus “Cut through” switching
- Layer 2, Layer 3, Layer 4 switching
  - Layer 2 is protocol independent – MAC address based
  - Layer 3 generally refers to TCP/IP routing layer
  - Layer 4 generally refers to TCP/IP ports, eg: HTTP port 80

Network infrastructure: VLANs

VLANs are another way to segment a network for performance and security

- Implemented within core switches
- Also implemented in NICs / device drivers (LLdriver)
- VLAN tagging of packets (802.1Q)
- Port based VLANs
- Protocol based VLANs
- Connectivity between VLANs
- QoS (Quality of Service) and bandwidth reservation

Network infrastructure: Multiple paths

- Different manufacturers (Cisco, HP, Extreme, etc.) have slightly different terminology and features (eg: Cisco ‘etherchannel’; Procurve ‘meshing’; Extreme ‘EAPS ring’)
- Inter-switch and switch to server links can be aggregated to provide sufficient bandwidth
- Packets may not arrive in the order in which they were sent
- Link “glitches” can cause traffic disruption
- VLANs can extend across multiple switches
- Extended distance inter-switch links

Network infrastructure: WAN

- ISDN, POTS
- Leased Line (KiloStream, MegaStream, T1 etc.)
- Frame Relay
- ATM
- MPLS and “pseudo-wires”
- “Dark fibre” and Wave Division Multiplexing
- SONET / SDH etc.
- ADSL / SDSL
- VPNs
- Encapsulation and tunnelling
- FC over IP, FC over Ethernet

Network infrastructure: Routers

- Routers build knowledge of address reachability (node or interface) on a per-protocol basis
- Protocol address based (Layer 3 devices)
- Routing table updates are propagated between routers
- Separate devices or can be integrated into the core
- Need to design protocol addressing scheme and areas
- IPV6 is common in big core routers
- Rare to find DECnet routing in modern routers – it's a TCP/IP dominated world in the WAN
- Can set up OpenVMS systems as dedicated multiprotocol routers if you need both DECnet and TCP/IP routing

Part 2: Interfacing OpenVMS to networks

- OpenVMS network protocols

OpenVMS networking: Multiple protocols

Very different to the Linux / Windows world, which is predominantly TCPIP V4. Scares network administrators!

- TCPIP V4
- TCPIP V6
- DECnet Phase IV
- DECnet-Plus and DECnet over IP
- SCS (use SCACP) and Clustering over IP
- AMDS (Availability Manager)
- LAT / MOP / Remote Console (Terminal Servers)
- LAD / LAST (Infoserver)
- SMB CIFS (Samba)

OpenVMS networking: Background

- VAX VMS V4.x introduced SCS for LAVC
- Infoserver introduced LAD / LAST for serving remote disc containers. Also used by RSM. Available in OpenVMS V8.2-1 onwards for Integrity to provide network upgrade.
- Pathworks (Advanced Server) introduced DECnet for PC operating systems and LANmanager functionality for OpenVMS systems. Replaced by CIFS (based on SAMBA)
- Galaxy introduced SMCI pseudo-LAN interconnect
- V8.3 introduced PEdriver compression

OpenVMS networking: Background

- OpenVMS V7.1 introduced LANCP / LANACP for MOP loading without DECnet (needed to load cluster satellites)
- OpenVMS V7.3-2 introduced “LAN failover” for improved LAN availability (all protocols)
- TCP/IP V5.4 introduced “failsafe IP” for improved TCP/IP availability within a cluster

- LLdriver: LAN failover
- VLdriver: VLAN tagging
- Jumbo frames (frame size varies with device type)
- PEdriver: compression, checksumming

OpenVMS networking: Multiple addresses

- DECnet naming is “per node”

- TCPIP addressing is “per interface”
- TCPIP allows multiple addresses per interface – and they can move between interfaces
- TCPIP - use “service addresses” that can be moved and enabled / disabled as needed
- TCPIP - use multiple subnets

- SCS, AMDS, LAT etc. are layer 2 non-routable protocols

OpenVMS networking: LAN failover

- Single high-availability LAN device for protocols that do not handle multi-path LANs well (TCP/IP V4, AMDS, etc.)
- Equivalent to Proliant NIC teaming in failover mode
- No load balancing
- Inter-operability with switches – stacking etc. needed
- Use LANCP to set up the failover group of NICs
- Booting – LANCP now runs before Pedriver, so everything is in place before SCS starts

OpenVMS networking: LAN failover

- Group NICS into LAN failover sets
- LLdriver presents “logical lan” devices

```
$ lancp define device lla/enable/failover=(ewa,ewc)
$ lancp define device llb/enable/failover=(ewb,ewd)
```

- One device in the failover set gives a layer of indirection

```
$ lancp define device lld/enable/failover=(ewd)
```

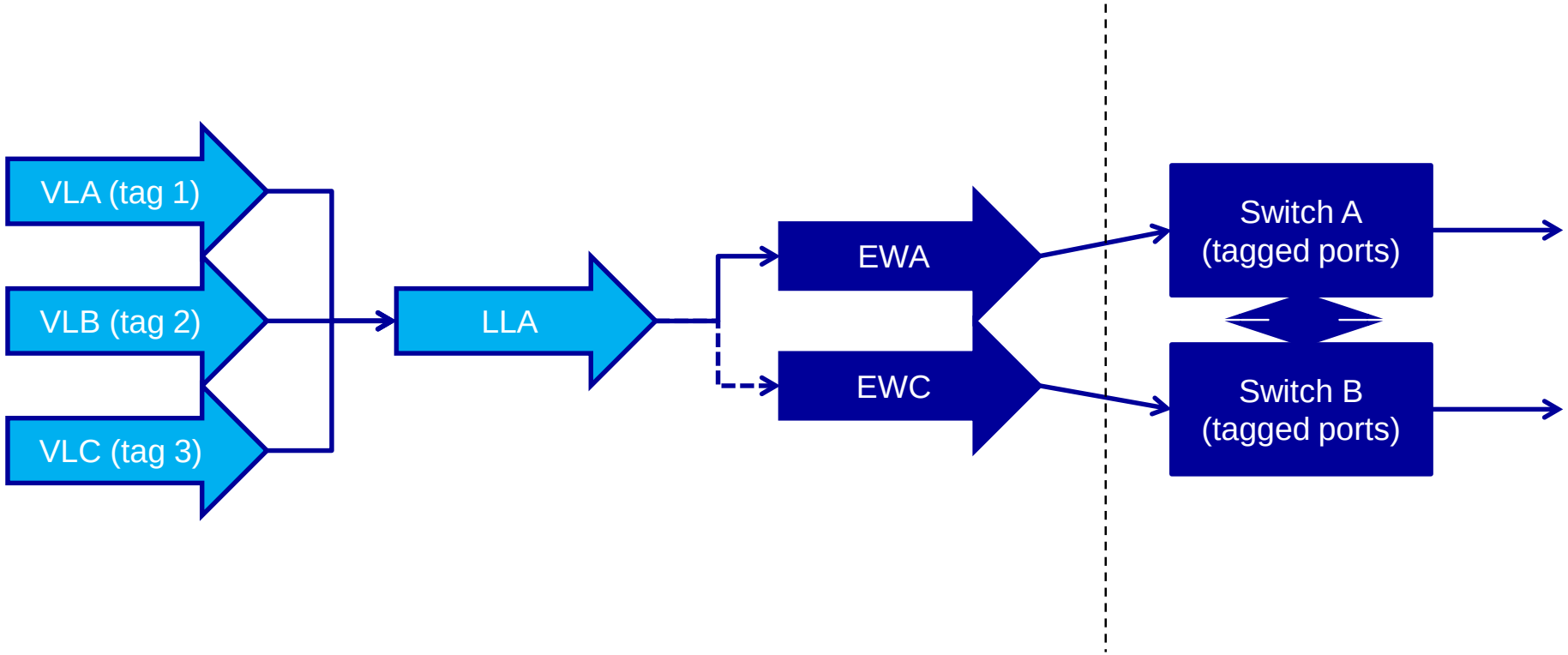
OpenVMS networking: VLAN tagging

- VLdriver applies 802.1Q tags to packets

```
$ lancp define device vla/tag='scs_tag'/vlan_device=lla
$ lancp define device vlb/tag='decnet_tag'/vlan_device=lla
$ lancp define device vlc/tag='userip_tag'/vlan_device=llb
$ lancp define device vld/tag=‘sysip_tag'/vlan_device=llc
$ lancp define device vle/tag=‘bkpip_tag'/vlan_device=lld
$ lancp define device vlf/tag=‘amds_tag'/vlan_device=lld
```

- Configure the switch ports to accept tagged packets

OpenVMS networking: connectivity



OpenVMS networking: jumbo frames

- Performance is better
- All intervening devices in the network infrastructure must have jumbo frame support enabled
- LAN_FLAGS bit 6 (64) enables jumbo frame support
- LANCP /JUMBO qualifier enables / disables jumbo frame support on a per device basis

OpenVMS networking: Configuration

- LANCP
  - See what's running on which interface: SHOW CONFIG/USER
  - Load client setup (MOP / BOOTP)
  - Device setup (EI,EW, LL, VL, etc.)
- SCACP
  - See what SCS is up to (VC, ECS, etc.)
  - Stop SCS on devices where you don't need it enabled
- @TCPIP$CONFIG
- @NET$CONFIGURE ADVANCED
- MCR DECNET_REGISTER

TCPIP V4

- Layer 4 – port or socket layer, “well known” ports allocated by convention (eg: HTTP = port 80)
- Layer 3 – IP addressing and routing layer, subnet based (eg: 192.168.0.n/24), DNS/BIND resolver maps hostnames to interface addresses
- Layer 2 – MAC address layer (ARP used to convert IP interface addresses to MAC addresses)
- Layer 1 – Physical layer (transmission media)

TCPIP: sub-protocols / services

- DNS and the BIND resolver
- DHCP address provision
- BOOTP services
- SSH / TELNET terminal access
- XDM and X11 Xterm access (DECwindows)
- SFTP / FTP / TFTP file transfer
- NFS file serving
- Monitoring with SNMP
- SMTP / POP / IMAP
- Printing (LPR / LPD)

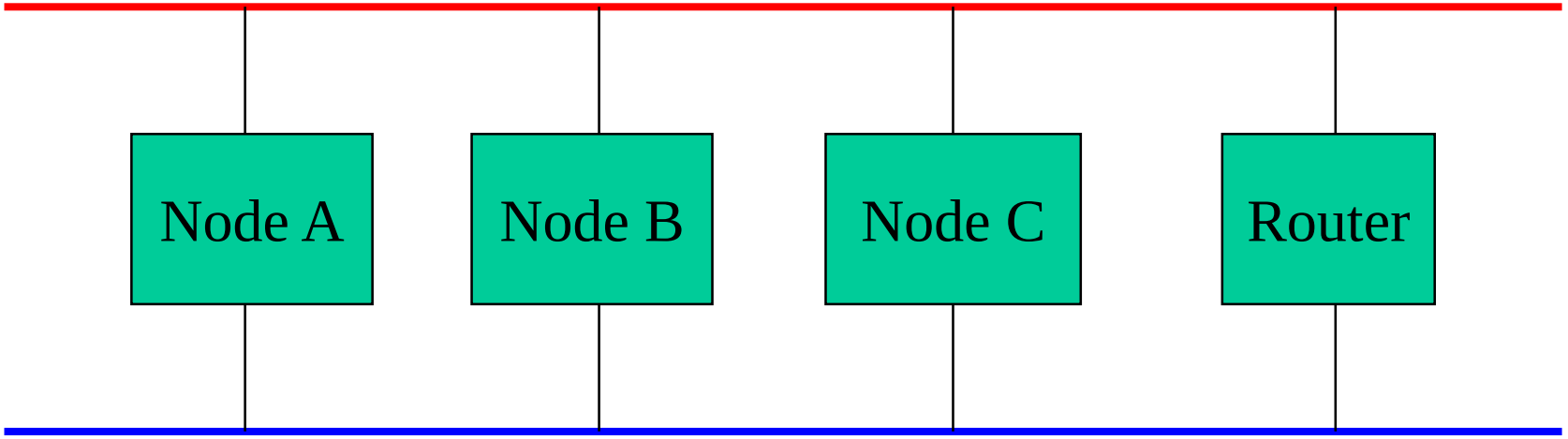
Also BSD style commands as well as DCL commands

TCPIP: Multiple addresses

- Consider “service addresses” and their failover, in addition to “interface addresses”:
  - Have per interface addresses for manageability and path availability “ping tests”
  - Have per machine addresses for systems management access (eg: one failsafe IP address across all NICs running IP in that machine)
  - Have per application addresses for user access and manual load distribution across NICs.
- Think about restricting access by subnet / address with TCPIP SET COMMUNICATION/SECURITY

  Use the IFCONFIG commands

Dual LAN: TCPIP V4 behaviour



- Per-interface addressing, no out-of-order packet cache
- Multiple NICs per subnet
- Different physical networks must be in different subnets
- Use LAN failover for high-availability dual connect

DECnet Phase IV addressing

- End Node
- Routing Nodes: Level 1 & Level 2 (Area) Routers
- MAC Address formed from Node address:
  - Area 1 - 63, Node: 1 - 1023
  - 16 bit address = (Area x 1024) + Node number
  - SCSSYSTEMID = same 16 bit value
  - AA-00-04-00-nn-mm
  - nn-mm = byte reversed hexadecimal 16 bit address

DECnet Phase IV “behind the scenes”

- DECnet “hidden information”:
  - End Node to Routers (end node hello packets)
  - Routers to Routers (routing updates)
  - Routers to End Nodes (router hello packets)

- DECnet Phase IV bases the MAC address on the node number, so no need for routers on LAN except for determining adjacencies.

  Router on LAN will give fast “node unreachable” rather than slow “timeout” when attempting to connect to a node that is not on the LAN.

DECnet Phase IV limitations

- Number of nodes in a private network can exceed the address range (eg: Easynet)
- MOP loader needs fake node entries
- Sets MAC address on all LAN adapters based on DECnet node address, so cannot connect multiple LAN adapters to the same LAN (or extended LAN).

Can route between parallel LANs, but cannot bridge between them due to the risk of duplicate MAC addresses.

DECnet Phase V: DECnet-Plus

- The obvious big difference - NCL in place of NCP
- Name Services
- DECnet over IP (RFC 1006)
- Permanent database is NCL script files (text)
- Time Synchronisation Service
- Routing algorithms (Phase V routers)
- Multiple path behaviour (multi-homed End System)
- Startup early in boot sequence
- Phase IV compatible addressing on first adapter only (by default)

DECnet Phase V: Addressing

- Phase IV compatible addressing
  - The AA-00-04-00-xx-xx address
  - Multiple CSMA-CD adapters
  - Select which adapters run in Phase IV mode

- Synonyms and FullNames
- Address Towers
  - Transport selection (NSP or TP4)
  - Session Control version selection (SC2 or SC3)

DECnet Phase V: Entities (7 layer model)

- session control
  - applications and ports
- transports
  - NSP and OSI (plus OSI templates)
- routing
- routing circuits
- csma-cd station (ethernet and FDDI)
- <datatype> links (HDLC, DDCMP etc.)
  - <datatype> link logical station
- modem connect lines

DECnet Phase V: DECnet-Plus

- Can disable DTSS by defining the NET$DISABLE_DTSS logical name in SYLOGICALS.COM.
- DTSS server can receive time from NTP
- See AUTO_DLIGHT_SAV system parameter and DST timezone rules
- Phase IV migration improvements (databases, FDDI)
- Improved NCL help
- Reduced NCL output on boot by default (NET$STARTUP_QUIET_NCL logical)

OpenVMS: Time synchronisation

Relative time is more useful than absolute time

- Need to be able to order events across the network based on timestamps
- Timestamp format
  - Time value
  - Inaccuracy component
- External reference Time Providers
- DTSS Servers and Clerks
- NTP servers

OpenVMS: DECnet over IP

- Preserves DECnet APIs for existing applications
- Performance and availability are determined by underlying IP network infrastructure
- DECnet uses TCPIP as a pseudo-transport layer
- Need to have RFC1006 and RFC1869 (aka RFC1006-Plus) OSI transport templates - ports 102 and 399
- Streams interface
- Need to have PWIP driver enabled

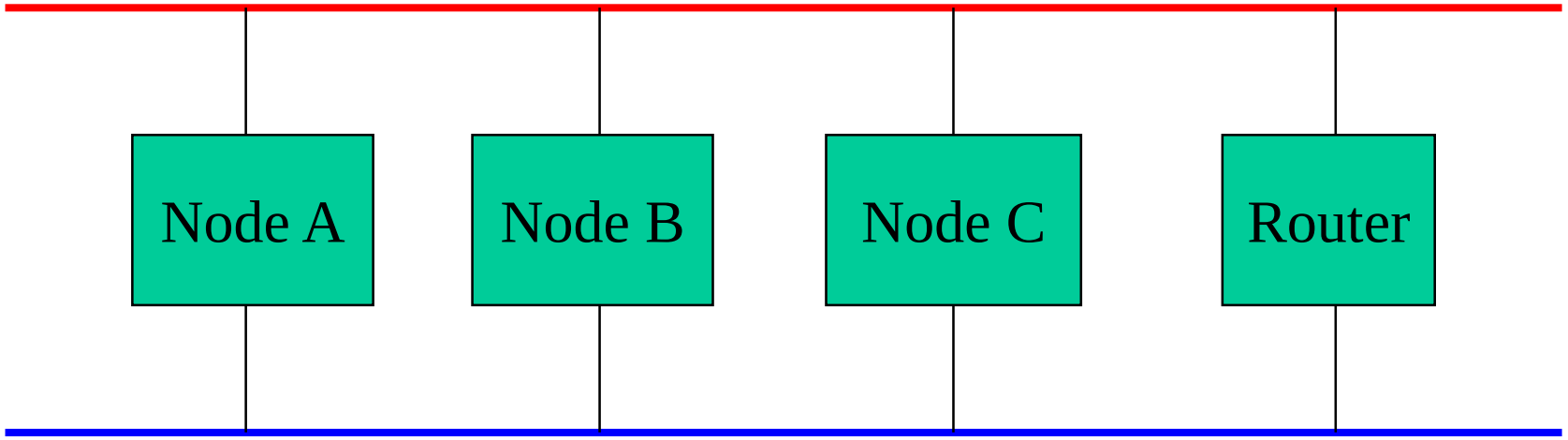
OpenVMS: DECnet over IP

- Need to have DNS/BIND name service in list for access to local name resolver:
  - @NET$CONFIGURE ADVANCED
  - Naming services: “LOCAL,DOMAIN”
  - Need a BIND resolver (can use 127.0.0.1)
- Can enable DECnet over IP “on the fly”:
  - change the naming (remove from LOCAL or DECdns naming database with DECNET_REGISTER and add to HOSTS database or DNS/BIND server)
  - NCL> FLUSH SESSION CONTROL NAMING CACHE ENTRY “*”

DECnet-Plus

- Don’t be scared of Phase V - it works, generally at least as well as or better than Phase IV
- NCL is a much better tool than NCP
- Keep it simple - use Local naming, Phase IV compatible addressing and the NSP transport
- DECnet over IP lets you use applications with the DECnet APIs over a TCP/IP only infrastructure
- Multi-Homed End Systems get you load balancing for the price of an end-node licence

Dual LAN: DECnet behaviour



- Phase IV – different costs, all L1 routers
- Phase IV Plus – same costs, all L1 routers
- Phase IV Plus – same costs, end node failover
- DECnet/OSI – both active, Multi-homed End Systems
- DECnet-Plus – both active, Multi-homed ES or IS

Part 3: Putting it all together

- Network infrastructures - putting it all together

Design issues: performance

- Bandwidth – determines throughput
  - Large packets shift more data with less overhead
- Signal path quality and reliability
  - Retransmits severely affect overall throughput
- Latency – determines round trip delay
  - Determines how much data is in transit at any given instant
  - Data in transit is at risk if there is a failure
- Jitter ("div latency" or variation of latency with time) – determines predictability of round trip delay
  - Understanding jitter is important for establishing timeout values
  - Severe latency fluctuations can cause system failures

Design issues: packet flow

- Traffic flow, end-to-end packet delivery, delivery failure notification and performance are key parts of the design of any network protocol, as are the addressing scheme and the naming scheme

- Multicast packets are inherently “fire and forget”
- Multiple paths – packets may no longer arrive in the order in which they were sent
- What happens when paths fail or are intermittent?
- How do we cope with bad latency or jitter?

- Time synchronisation across the infrastructure

OpenVMS: planning the configuration

- Node naming, addressing schemes and routing mechanisms
- Multiple NICs and multiple LANs
- Map functions to NICs:
  - Management (ILO, SAN appliance, etc.)
  - Clustering
  - Network backups
  - Data transfers (eg: FTP, NFS etc.)
  - Interactive users

Part 4: Integrating OpenVMS

- External authentication
- User access
- File sharing
- Web serving
- Monitoring and management tools

LDAP / Windows AD - authentication

- ACME
- LDAP
- Configure LDAP / Windows AD to allow access from OpenVMS LDAP authentication agent
- Swap loginout image etc. (requires reboot – security)
- Configure ACME server to use LDAP agent
- Configure LDAP agent on VMS to have access to LDAP directory service / Windows Active Directory
- Configure “mapping of principals” if needed
- Set “extauth” flag in UAF

LDAP / Windows AD - example

SYSTEM on RX2660 $ set h 0

OpenVMS IA64 V8.4 on node RX2660

Username: butcher_ca
Password:
    Last interactive login on Thursday, 16-APR-2015 19:46:32.29
**** Logon authenticated by LDAP ****
BUTCHER_CA on RX2660 $

File sharing – CIFS (SAMBA) and NFS

- Uses SMB (Server Message Block) over TCPIP
- CIFS V1.2-ECO01 PS2_13 is latest from HPE
- TCPIP V5.7-ECO05A is latest from HPE
- NFS V3 client
- Pathworks (Advanced Server) to CIFS migration

CIFS (SAMBA)

- Install and perform initial configuration of CIFS
- Edit SMB.CONF to add shares and set values for your site
- Connect to domain (net rpc join)
- Start CIFS
- SWAT (if used) needs local webserver (CSWS)

CIFS (SAMBA)

- File formats:
  - File format conversion may be needed:
    $ convert/fdl="record; format stream_lf" <infile> <outfile>
- Security:
  - File ownership and protection mapping

CIFS (SAMBA) – shares – SMB.CONF

Share specific sections in file: [sharename]

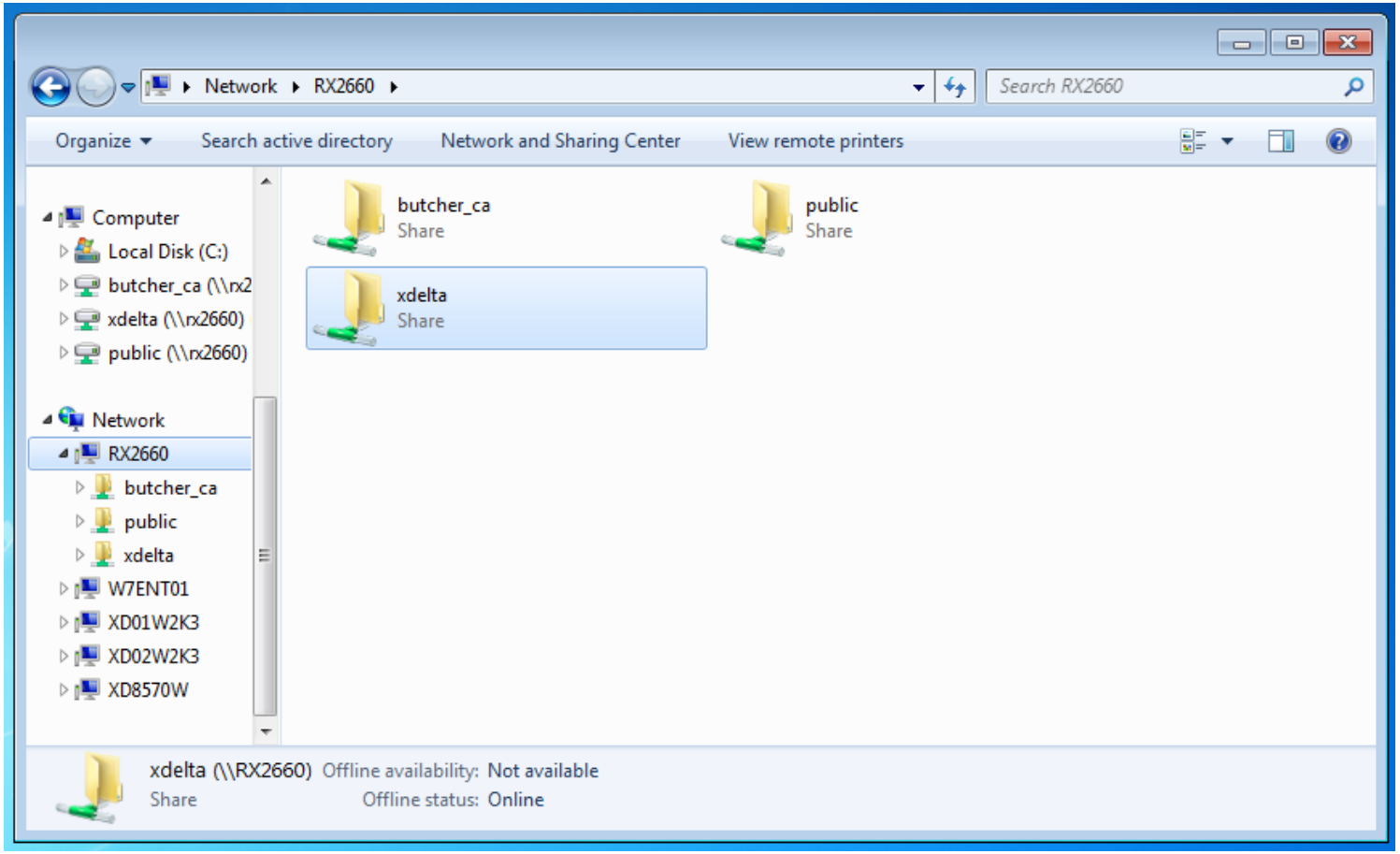
```
[xdelta]
        comment = XDelta working area
        path = xd_tree_toplevel:
        writable = yes
        read only = no
        printable = no
        browseable = yes
        inherit owner = yes
        vfs objects = varvfc
        vms rms format = streamlf
        vms ods5 volume = yes
```

CIFS (SAMBA) – [homes] share

Special case: [homes] share

- Per-user “home” directories on server
- Needs domain access on both client and server
- WINBIND and username mapping
- See OpenVMS Technical Journal V14 for additional information

CIFS (SAMBA) – the view from Windows 7



NFS server and NFS client

- File format conversion may be needed:

  $ convert/fdl="record; format stream_lf" <infile> <outfile>

- TCPIP V5.7-ECO05A fixes a lot of NFS problems

- New features coming soon with the new TCPIP stack!

NFS client

- Always use a separate copy of the XQP for NFS mounts, e.g.:

```
$ tcpip mount dnfsNNN: /host=<hostname / ip_address>/path="<path>" -
        /uid=XXX/gid=YYY -
        /structure=5/convert/system -
        /processor=unique/acp=(dump,buffer_limit=1000000,page_file=500000) -
        /transport=tcp/version=3 –
        /sync=file_sync
        /noadf -
        <volume_name>
```

Terminal access

- SSH (and TELNET)
  - Terminal emulation products
  - ILO on Integrity Servers
- Serial port access (useful for ILO setup etc.):
  - Console servers (SSH and TELNET)
  - DECservers (no SSH / SSL)

DECwindows access

- Xterms
- X11
- XDM
- SSH and port forwarding

Monitoring

- Tools such as PTRG will collect and display SNMP data graphically over time
- TCPIP has a built-in MIB, other MIBs are available

- T4 can monitor and record traffic flows per network interface (or pseudo-interface)

- SCACP
- LANCP
- DECnet-Plus
- TCPIP (@SYS$MANAGER:TCPIP$DEFINE_COMMANDS)

Availability Manager

- Uses AMDS protocol to interact with OpenVMS at driver level
- Can diagnose and bypass / fix a range of problems (with care!)
- New: VMS Software Availability Manager on 64bit platform

Future plans (VMS Software)

- New TCPIP stack
- Up-to-date SSL
- Up-to-date CIFS (SAMBA)
- Up-to-date CSWS (Apache), PHP, PERL, etc.
- Up-to-date JAVA and TOMCAT
- If you have a wish list, let us know!

Data networking with OpenVMS

OpenVMS Bootcamp 2017

Session 237

Colin Butcher CEng FBCS CITP
Technical director, XDelta Limited
www.xdelta.co.uk