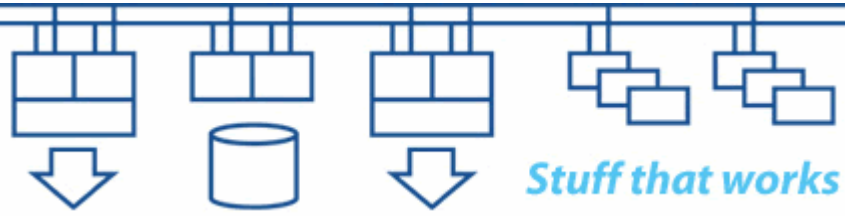
OpenVMS advanced technical symposium – May 2007

Designing OpenVMS systems for performance and availability

Colin Butcher
Technical Director, XDelta Limited

Web: http://www.xdelta.co.uk
E-mail: vms@xdelta.co.uk



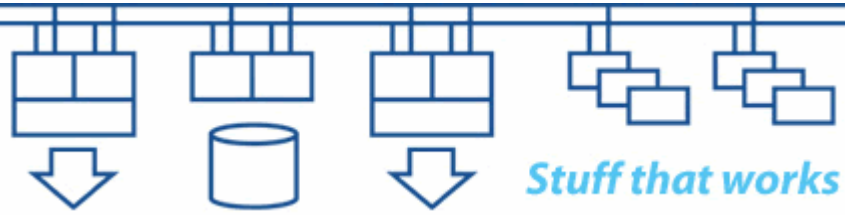
Overview

Overall system availability and performance depends on matching the system configuration to the requirements of the application. Performance issues are often the cause of system failures and disruption, thus affecting availability.

System designers need to understand both the application and the behaviour of their target systems in order to write code that is reliable, which has minimal performance impact and which scales well in a production environment.

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



Topics (1 of 2)

Principles of performance:

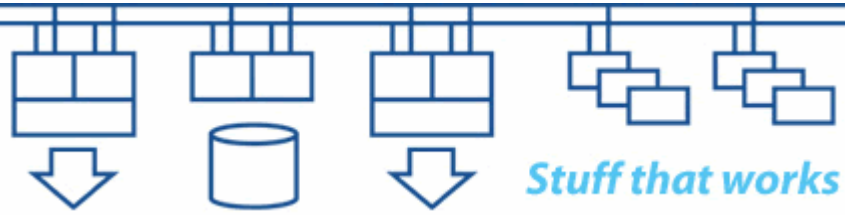
Understanding throughput, response times and the underlying mechanisms that determine the overall performance characteristics of a system.

Principles of high availability:

An overview of availability analysis, state transitions and failure detection.

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



Topics (2 of 2)

Network connectivity and protocols:

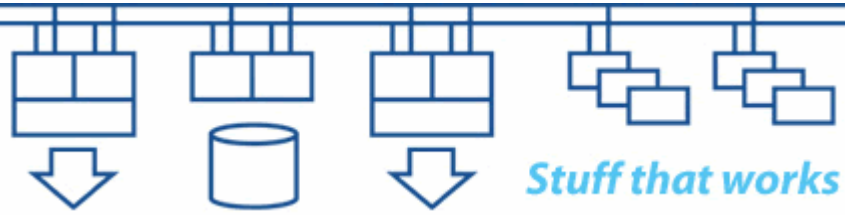
An overview of networking and connectivity to systems.

Designing for availability and performance:

This will use case studies and examples to bring the different strands together in a practical manner to solve real-world problems.

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

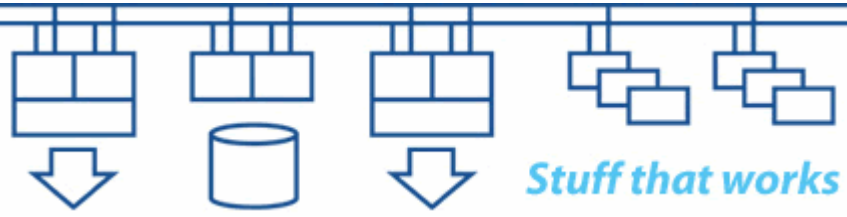


Principles of performance:

Understanding throughput, response times and the underlying mechanisms that determine the overall performance characteristics of a system.

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

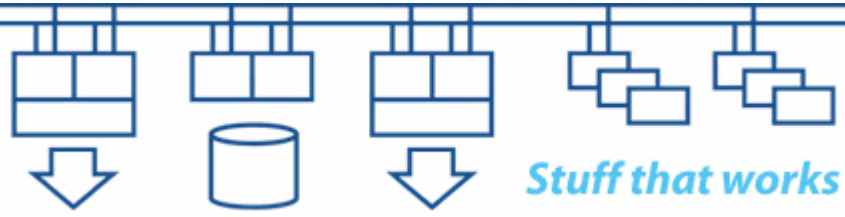


The performance envelope

- **What does the business require the systems to do?**
- **What happens to the business if the systems fail?**
- **What happens if you push beyond the limits?**
- **How far from the edge are you?**
- **How do you know?**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

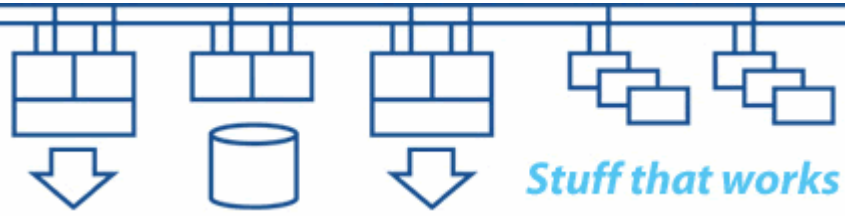


How can we characterise it?

- **What is “fast” or “slow”?**
- **What are the performance requirements of the system we’re designing or investigating?**
- **What can we measure?**
- **What comparisons can we make?**
- **What “footprint” can we look for?**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



How do we go looking?

- **Time series performance data is just data – it does not tell us what to do**
- **Time series data sets allow us to make comparisons**
- **You must understand your workload**
- **You must understand what “normal” looks like**
- **There is no substitute for experience to interpret and analyse the data**

Web: http://www.xdelta.co.uk

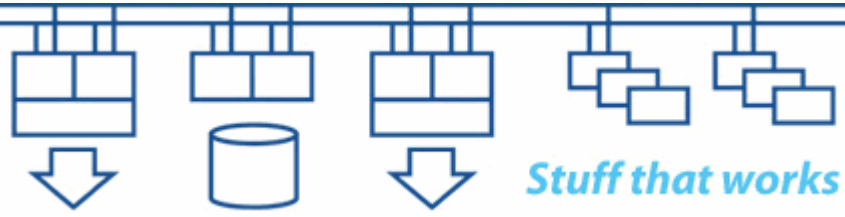
E-mail: vms@xdelta.co.uk

Basic principles

- **Bandwidth – determines throughput**
- **Latency – determines response time**
- **Jitter ("div latency" or variation of latency with time) – determines predictability of response**
- **Think about a motorway (freeway):**
  - **What is throughput?**
  - **What is latency?**
  - **What is jitter?**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

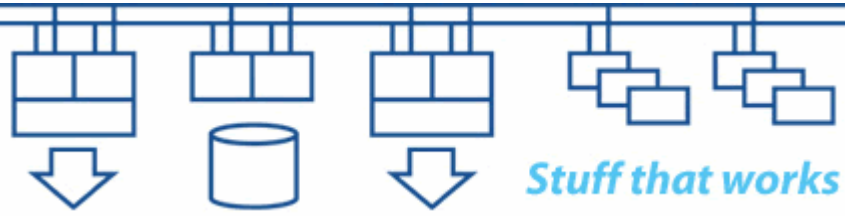


Bandwidth

- **Bandwidth – determines throughput**
  - **Worst case is the maximum capacity of the smallest path in the system (the “bottleneck”)**
  - **It’s not just “speed”, it’s throughput in terms of “units of stuff per second”**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

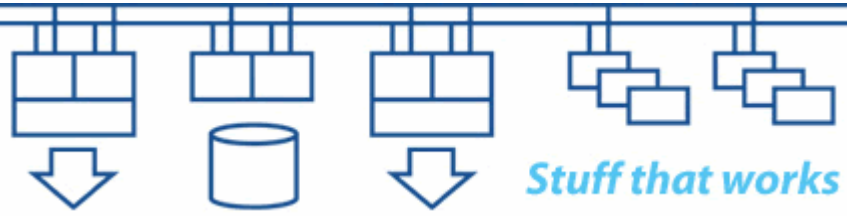


Latency

- **Latency – determines response time**
  - **Worst case is the sum of all the delays in the system**
  - **Latency determines how much "stuff" is in transit through the system at any given instant**
  - **"Stuff in transit" is the data at risk if there is a failure**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

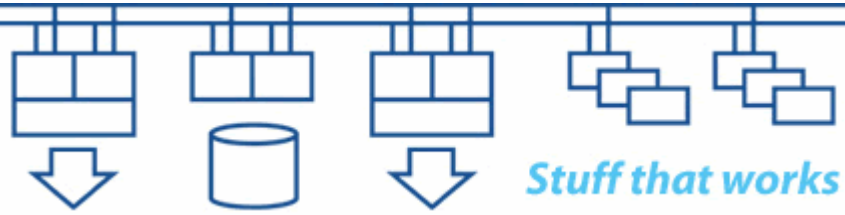


Jitter

- **Jitter ("div latency" or variation of latency with time)**
  - **Determines predictability of response, ie: timeouts**
  - **Ideally zero**
  - **Wildly fluctuating latency is a "bad thing"**
  - **Poor jitter can cause system failures under peak load**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

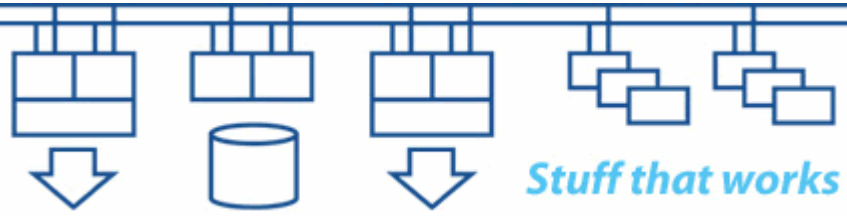


What can we do?

- **Parallelism and reduction in “wait states”**
- **Caches at various levels**
- **Synchronisation of access to data structures**
- **Serialisation of access to data structures**
- **Scalability – do as much as possible once only, do little as possible every time**

Web: http://www.xdelta.co.uk

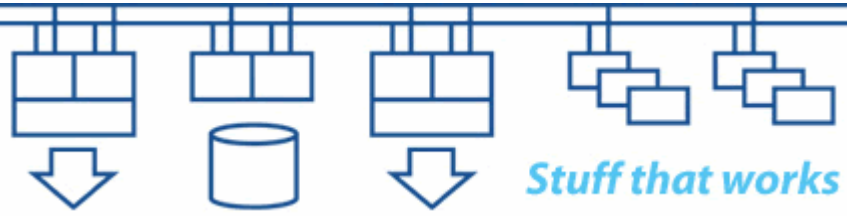
E-mail: vms@xdelta.co.uk



Principles of high availability

An overview of availability analysis, state transitions and failure detection.

Web: http://www.xdelta.co.uk
E-mail: vms@xdelta.co.uk



Overview

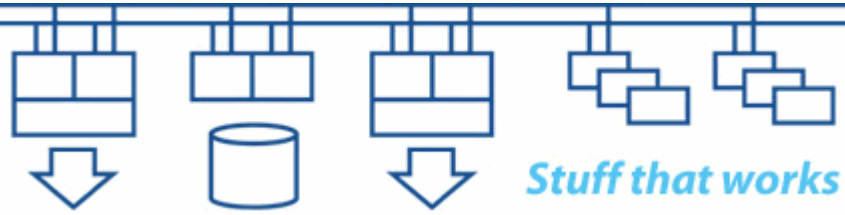
- **Availability is more important than performance**

- **We need to understand how systems fail and what failure looks like**

- **Business continuity:**
  - **It's not just the systems – it's everything!**
- **Disaster tolerance:**
  - **Surviving major site outages without loss of data**
- **High availability:**
  - **Surviving equipment and software failures**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

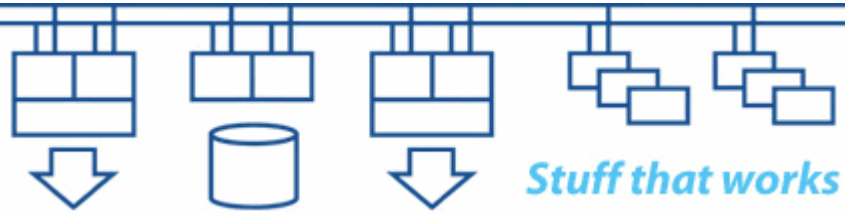


Availability assessment

- **How can we look for points of failure?**
- **How can we assess the impact of failure?**
- **Which parts of the system are mission-critical?**
- **What kind of failure do we prefer?**
- **What happens to our data?**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



Reliability and Availability

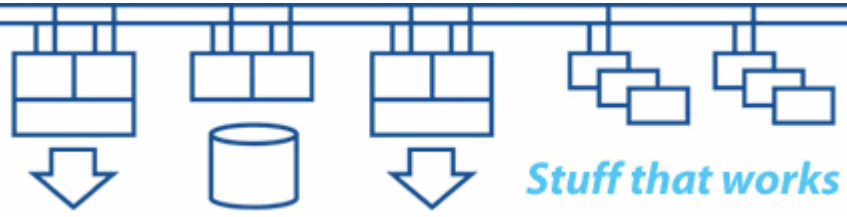
- **Reliability = Probability of failure at a given point in time, usually expressed as MTBF (Mean Time Between Failures)**

- **Availability = Probability of system being up and running at the instant when need it. Function of MTBF and MTTR (Mean Time To Repair), usually expressed as a % uptime , eg: 99.999%**

- **99.999% uptime (five nines) is equivalent to 5.26 minutes loss of service in a year for a 24x365 system. For a 12hour x 5day operational window it's only 1.87 minutes permissible outage in a year. That's difficult to achieve.**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



Techniques

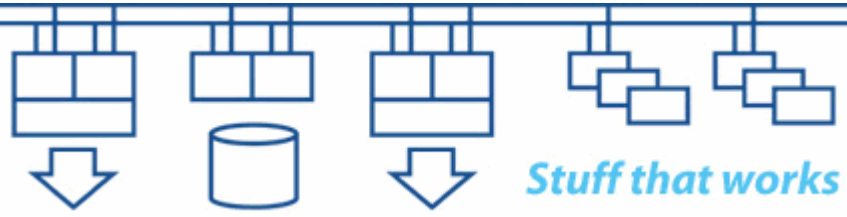
There are many techniques which have evolved over the years and there are tools to help you apply them. For example:

- **Reliability Block Diagrams (RBD)**
- **Failure Modes, Effects and Criticality Analysis (FMECA)**
- **Fault Tree Analysis (FTA)**

These techniques can be applied with software tools available from a number of vendors.

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



State transitions

A	B
Off to Master	Off
Master	Off to Standby
Master to Off	Standby to Master
Master to Off	Off to Master
Master to Standby	Standby to Master
Master	Standby to Off
And so on…	

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

Timing issues

- **Nothing happens instantaneously**
- **How long do state transitions last for?**
- **Can we ensure that there are no timing windows?**

Web: http://www.xdelta.co.uk

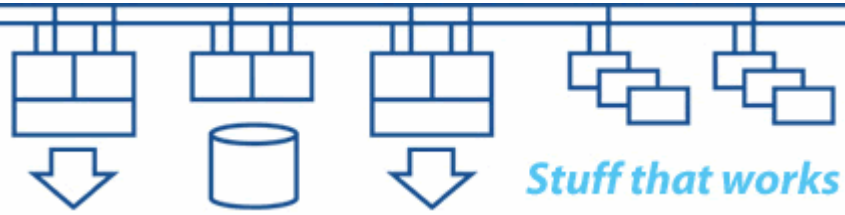
E-mail: vms@xdelta.co.uk

Dealing with failures

- **How can we get good information?**
- **What does “failure” look like?**
- **How quickly do we need to react to a failure?**
- **Should we automate decision making?**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

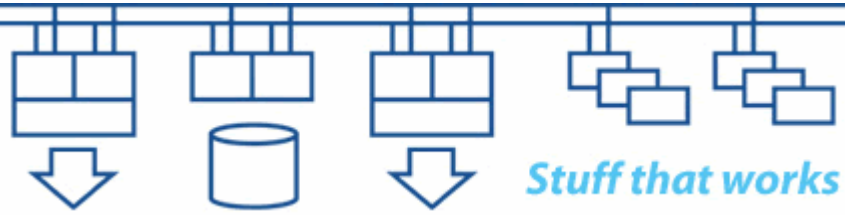


Network connectivity and protocols

An overview of networking and connectivity to systems.

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

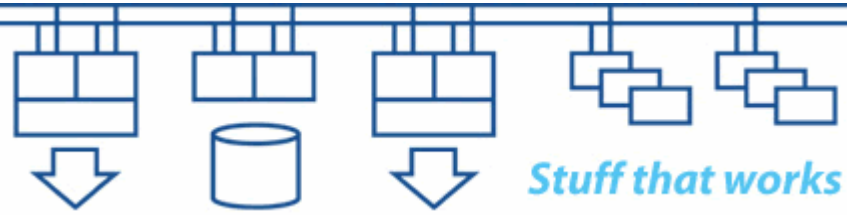


Network connectivity

- **Network protocols used with OpenVMS systems**
- **Network basics (bridging, switching etc.)**
- **Node naming, addressing schemes and routing mechanisms**
- **Multiple NICs and multiple LANs**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



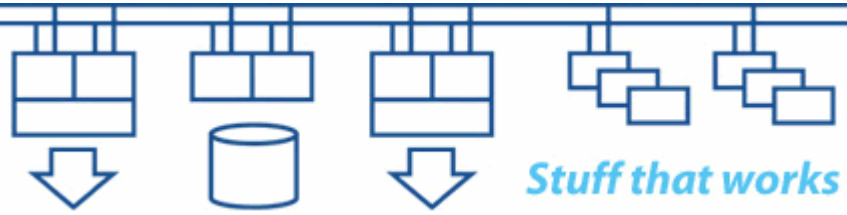
Common protocols

Typical network protocols in use with OpenVMS systems:

- **SCS (clustering)**
- **TCP/IP (and all it's components)**
- **DECnet-Plus (NSP,OSI and DECnet over IP) or DECnet Phase IV (NSP only)**
- **DECdns (not to be confused with TCP/IP's DNS/BIND)**
- **LAT (DECserver terminal access etc.)**
- **MOP + Remote Console (DECserver, LAVC boot etc.)**
- **DTSS (can be disabled)**
- **LAD + LAST (Infoserver etc.)**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



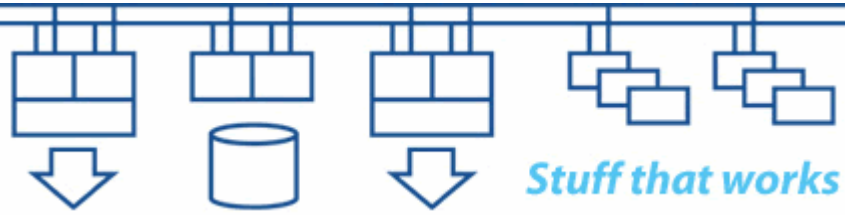
Ethernet addressing

- **Hardware MAC address**
- **Physical MAC address**
- **Broadcast address**
- **Multicast addresses**
- **Point to point addresses**
- **Ethernet packet format v IEEE802.3 packet format**

- ***TIP: SDA>SHOW LAN/FULL***

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



LAN - Segmentation

Why segment a network?

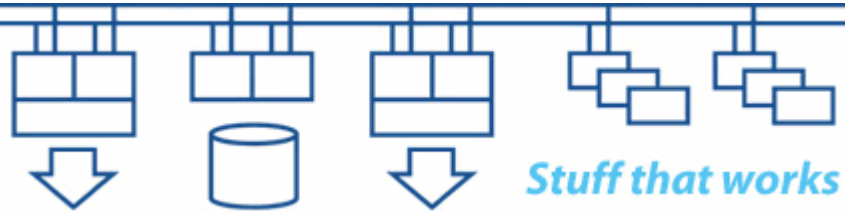
- Performance
- Security
- Availability

How can you segment a network?

- Repeaters
- Bridges
- Switches
- VLANs
- Routers

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

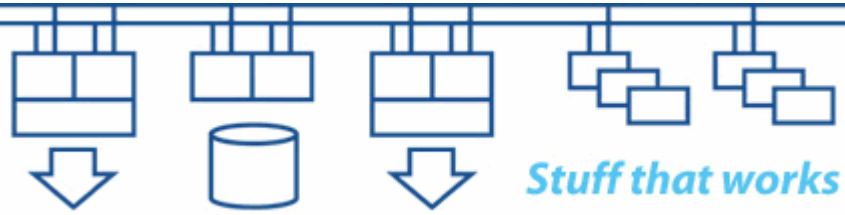


Using VLANs to segment a network:

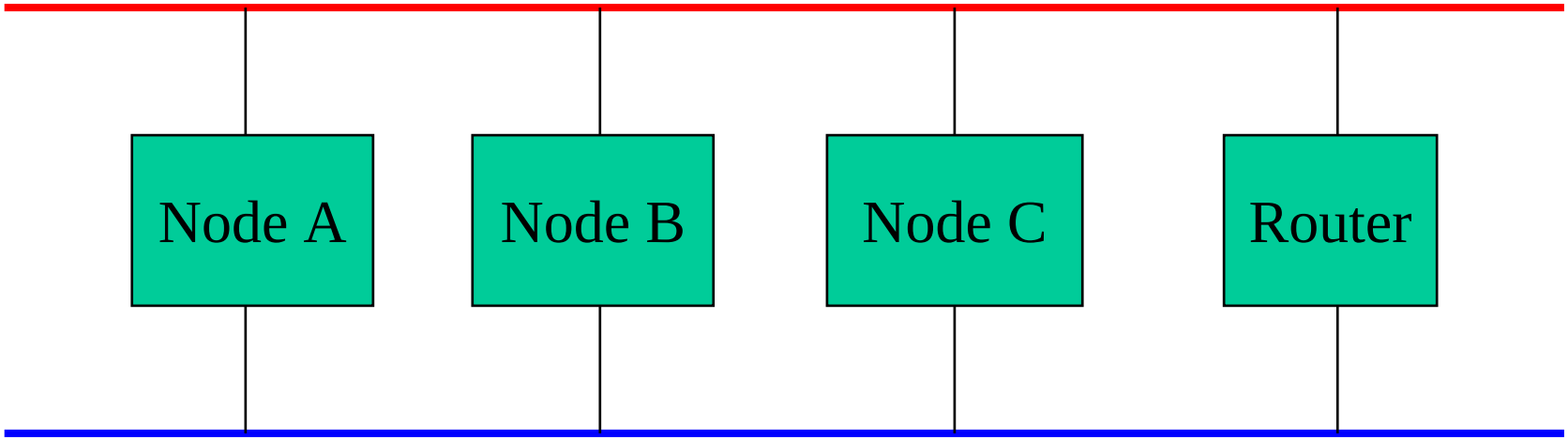
- Implemented within core switches
- Layer 2, Layer 3 etc.
- Port based VLANs
- Protocol based VLANs
- Connectivity between VLANs
- VLAN tagging of packets (802.1Q)

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



Dual LAN behaviour



- **DECnet Phase IV– L1 routers or end-node failover**
- **DECnet-Plus –Multi-homed ES or IS, load balancing**
- **TCP/IP – multiple NICs per subnet, dynamic routing**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

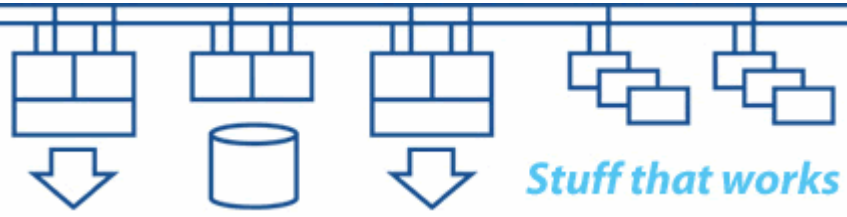
Multiple LAN interface behaviours

- **LAN failover (LLdriver)**
- **DECnet Phase IV and V - load balancing**
- **TCP/IP - Failsafe IP**
- **SCS – stopping and starting per adapter**

- **MOP and LANCP (network booting)**
- **LAD / LAST (InfoServer)**
- **LAT (DECservers)**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

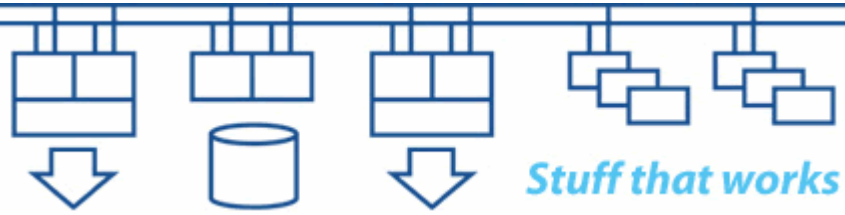


Relative time is more useful than absolute time

- Need to be able to order events across the network based on timestamps
- UTC Timestamp format
  - Time value
  - Inaccuracy component
- External reference clocks
- NTP
- DTSS

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

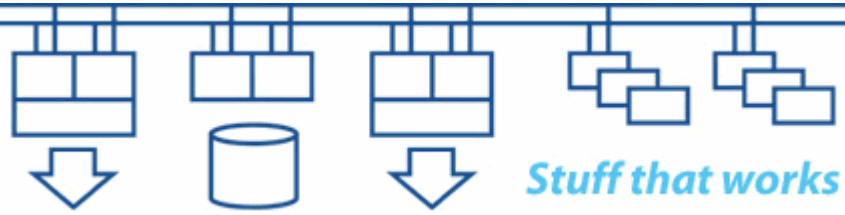


Designing for availability and performance

This will use case studies and examples to bring the different strands together in a practical manner to solve real-world problems.

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



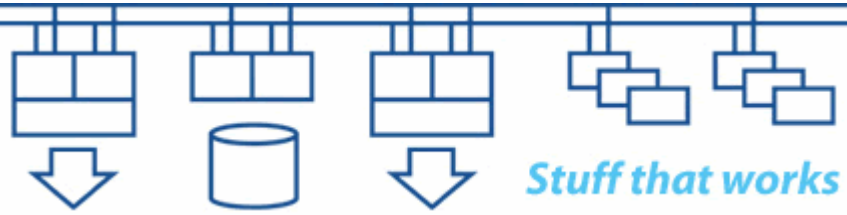
Mission Critical systems

Mission critical systems need to be able to:

- **Survive failures (resilience and failover)**
- **Survive changes (adapt and evolve)**
- **Survive people (simplify and automate)**
- **Never corrupt or lose critical data (data integrity)**
- **Requirements never remain static over an extended period of time, so we need to be able to make changes during the operational lifetime of the system.**
- **Circumstances change, so we often need to be able to extend the operational lifetime and scope of a system.**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



Do we really need 100% uptime?

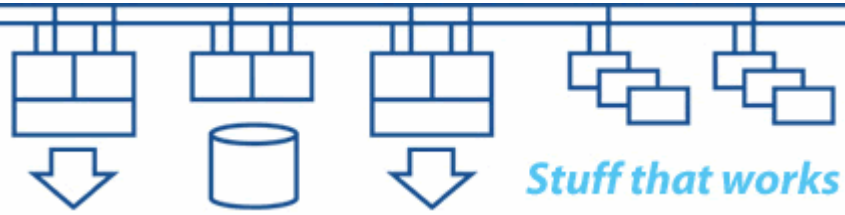
- **Safety-critical systems (especially safety-critical real-time monitoring and control systems such as air traffic control) require exceedingly high levels of availability. They also have to be fail-safe in order not to endanger lives.**

- **True 24x365 mission-critical systems are fairly rare. With these there is no "downtime window" to take backups, fix faults or to make changes. So, whatever you do has to be done "live" – and very carefully!**

- **The closer you get to 100% uptime the more expensive a satisfactory solution will become.**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

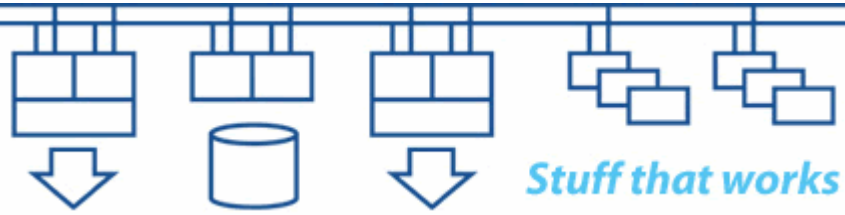


"Survivability test"

Cause of Outage:	Planned (Maintenance)	Unplanned (Failure)
Hardware	?	?
Operating System	?	?
Network Layer	?	?
Layered Products	?	?
Application Software	?	?
Application Data	?	?
Environment	?	?
Staff	?	?

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

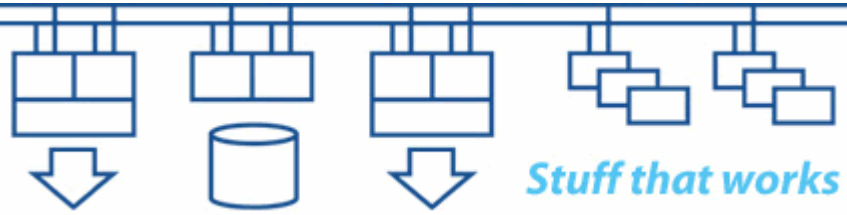


System configurations

- System configuration – the art is to select components that work well together and which provide the bulk of what you need with minimal additional work
- Establish the minimum requirements that have to be met – and do it as well as possible
- Availability and performance have to be designed in to the application
- Monitoring and automation are key components
- Understand the typical behaviour of your systems and be aware of changes
- Configuration control – hardware, settings, software etc.

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

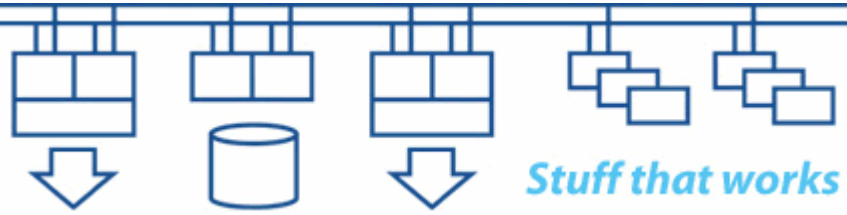


Systems infrastructure

- **Use multiple systems within a site – and have appropriate physical separation between them**
- **Use multiple sites with appropriate geographical separation**
- **Understand what happens when a failure occurs and how the overall system configuration is likely to respond**
- **Avoid the risk of more than one system thinking that it's in charge when a failure has happened**
- **Ensure that the surrounding infrastructure and environment is appropriate to the needs of your systems**
- **Configure the individual components of the systems in a manner that maximises interchangeability**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

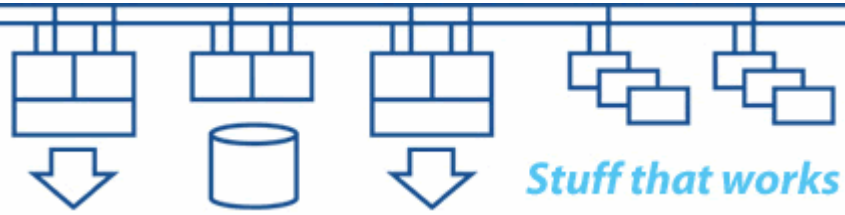


Storage infrastructure

- **Losing data is a disaster**
- **Availability is more important than performance**
- **Size storage subsystem based on minimum components and maximum estimated throughput**
- **Segment storage subsystem to provide gradual degradation rather than wholesale failure**
- **Need adequate backup capacity and throughput in order to meet permissible backup windows**
- **Understand application behaviour and storage performance requirements (bandwidth and latency)**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

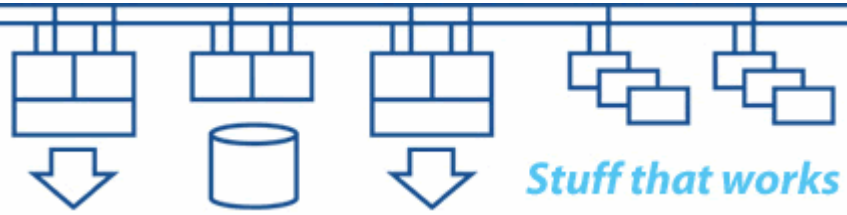


Network infrastructure

- **Scale network so that overall performance is based on minimum essential number of paths and maximum estimated traffic**
- **May wish to take advantage of installed bandwidth capacity to provide additional functionality when everything is working**
- **There is no such thing as a single protocol network**
- **Understand the behaviours of the different protocols under failure conditions**
- **Segment the network to provide gradual degradation rather than wholesale failure**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



Software considerations

- Operating system design
- System configuration
- Performance tuning
- Application design (scalability, data integrity, availability)
- Compilers generate code better than you can
- Use the OS features (eg: lock manager)
- Portability
- Reliability
- Supportability
- Testing and documentation

Web: http://www.xdelta.co.uk

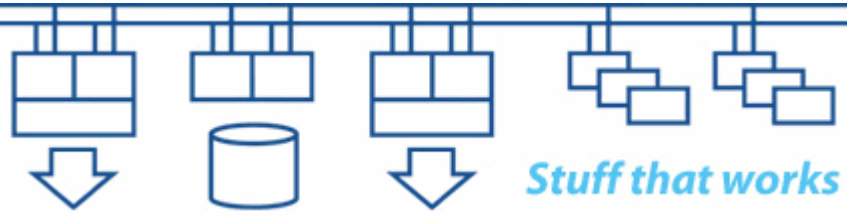
E-mail: vms@xdelta.co.uk

Limits to software performance

- Ability to make use of hardware parallelism
- Granularity of data structures
- Synchronisation techniques
- Serialisation techniques
- Scalability techniques
- Compilers
- Application design
- Designing and writing very good code requires very good programmers

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

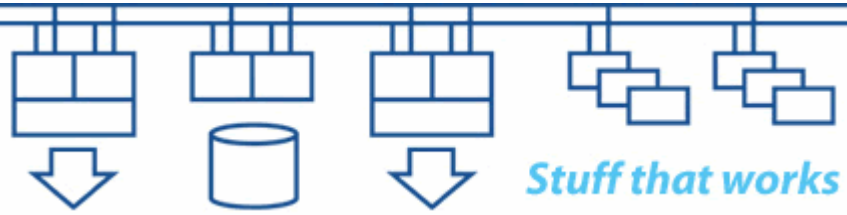


Design decisions

- **Big decisions which have long-term implications and constraints, eg: disc block size = 512bytes**
- **Small decisions which seem big at the time, eg: memory page size = 512 bytes, now variable**
- **Requirements you don't yet understand or know about**
- **Design for change**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



Problem solving concepts

	Before	Now	After
Environment			
System			
Component			

Web: http://www.xdelta.co.uk

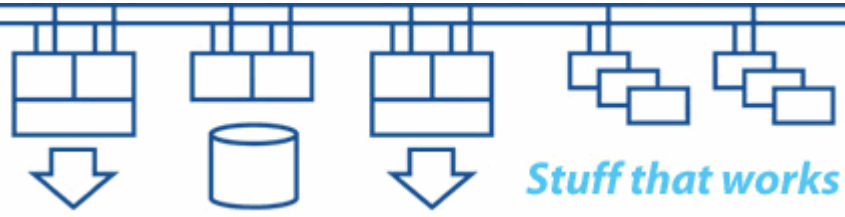
E-mail: vms@xdelta.co.uk

Design and implementation

- Think big, implement small
- Documentation
- Portability
- Modularity
- Reliability
- Scalability
- Software build process
- Software installation process
- Configuration control
- Testing (development, pre-deployment, post-deployment)
- Support and planning

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk

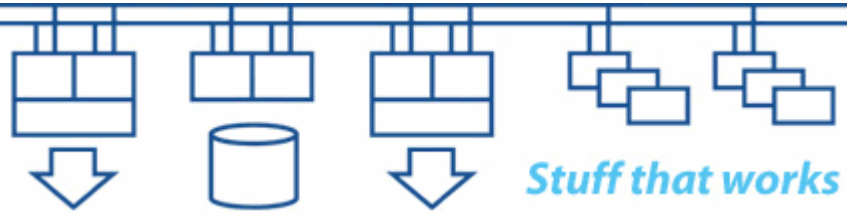


Examples and discussion

- **Satellite flight control**
- **Air traffic monitoring**
- **Finance data processing**
- **Healthcare infrastructure**
- **Process control**
- **What do you want to discuss? What examples have you got tht we can work with?**

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk



Thank you for your participation

Q & A

Web: http://www.xdelta.co.uk

E-mail: vms@xdelta.co.uk