Using OpenVMS Clusters for Disaster Tolerance

Keith Parris

Systems/Software Engineer
HP

HP WORLD 2003
Solutions and Technology Conference & Expo

Speaker Contact Info:

- Keith Parris
- E-mail: parris@encompasserve.org
- **or** keithparris@yahoo.com
- **or** Keith.Parris@hp.com
- Web: http://encompasserve.org/~parris/
- **and** http://www.geocities.com/keithparris/
- **and** http://www2.openvms.org/kparris/

High Availability (HA)

- Ability for application processing to continue with high probability in the face of common (mostly hardware) failures
- Typical technologies:
  - Redundant power supplies and fans
  - RAID for disks
  - Clusters of servers
  - Multiple NICs, redundant routers
  - Facilities: Dual power feeds, n+1 air conditioning units, UPS, generator

Fault Tolerance (FT)

- The ability for a computer system to continue operating despite hardware and/or software failures
- Typically requires:
  - Special hardware with full redundancy, error-checking, and hot-swap support
  - Special software
- Provides the highest availability possible within a single datacenter

Disaster Recovery (DR)

- Disaster Recovery is the ability to resume operations after a disaster
  - Disaster could be destruction of the entire datacenter site and everything in it
- Implies off-site data storage of some sort

Disaster Recovery (DR)

- Typically,
  - There is some delay before operations can continue (many hours, possibly days), and
  - Some transaction data may have been lost from IT systems and must be re-entered

Disaster Recovery (DR)

- Success hinges on ability to restore, replace, or re-create:
  - Data (and external data feeds)
  - Facilities
  - Systems
  - Networks
  - User access

DR Methods: Tape Backup

- Data is copied to tape, with off-site storage at a remote site
- Very-common method. Inexpensive.
- Data lost in a disaster is: all the changes since the last tape backup that is safely located off-site
- There may be significant delay before data can actually be used

DR Methods: Vendor Recovery Site

- Vendor provides datacenter space, compatible hardware, networking, and sometimes user work areas as well
  - When a disaster is declared, systems are configured and data is restored to them
- Typically there are hours to days of delay before data can actually be used

DR Methods: Data Vaulting

- Copy of data is saved at a remote site
  - Periodically or continuously, via network
  - Remote site may be own site or at a vendor location
- Minimal or no data may be lost in a disaster
- There is typically some delay before data can actually be used

DR Methods:
Hot Site

- Company itself (or a vendor) provides pre-configured compatible hardware, networking, and datacenter space
- Systems are pre-configured, ready to go
  - Data may already resident be at the Hot Site thanks to Data Vaulting
- Typically there are minutes to hours of delay before data can be used

Disaster Tolerance vs. Disaster Recovery

- Disaster Recovery is the ability to *resume* operations after a disaster.
- Disaster Tolerance is the ability to *continue operations uninterrupted* despite a disaster

Disaster Tolerance

- Ideally, Disaster Tolerance allows one to continue operations uninterrupted despite a disaster:
  - *Without any appreciable delays*
  - *Without any lost transaction data*

Disaster Tolerance

- Businesses vary in their requirements with respect to:
  - Acceptable recovery time
  - Allowable data loss
- Technologies also vary in their ability to achieve the ideals of *no data loss* and *zero recovery time*
  - OpenVMS Cluster technology today can achieve:
    - zero data loss
    - recovery times in the single-digit seconds range

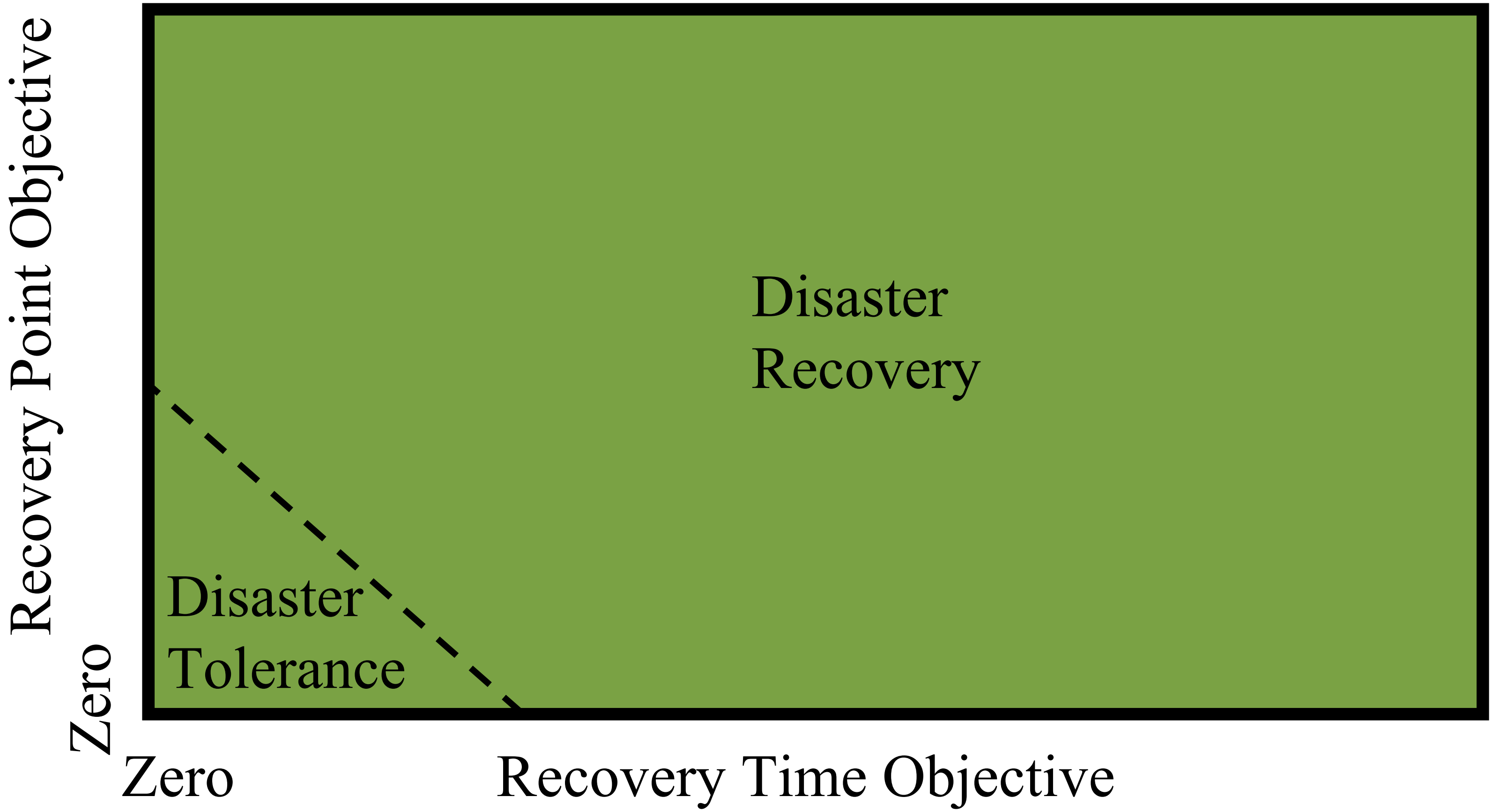
Measuring Disaster Tolerance and Disaster Recovery Needs

- Determine requirements based on *business needs* first
  - Then find acceptable technologies to meet the needs of the business

Measuring Disaster Tolerance and Disaster Recovery Needs

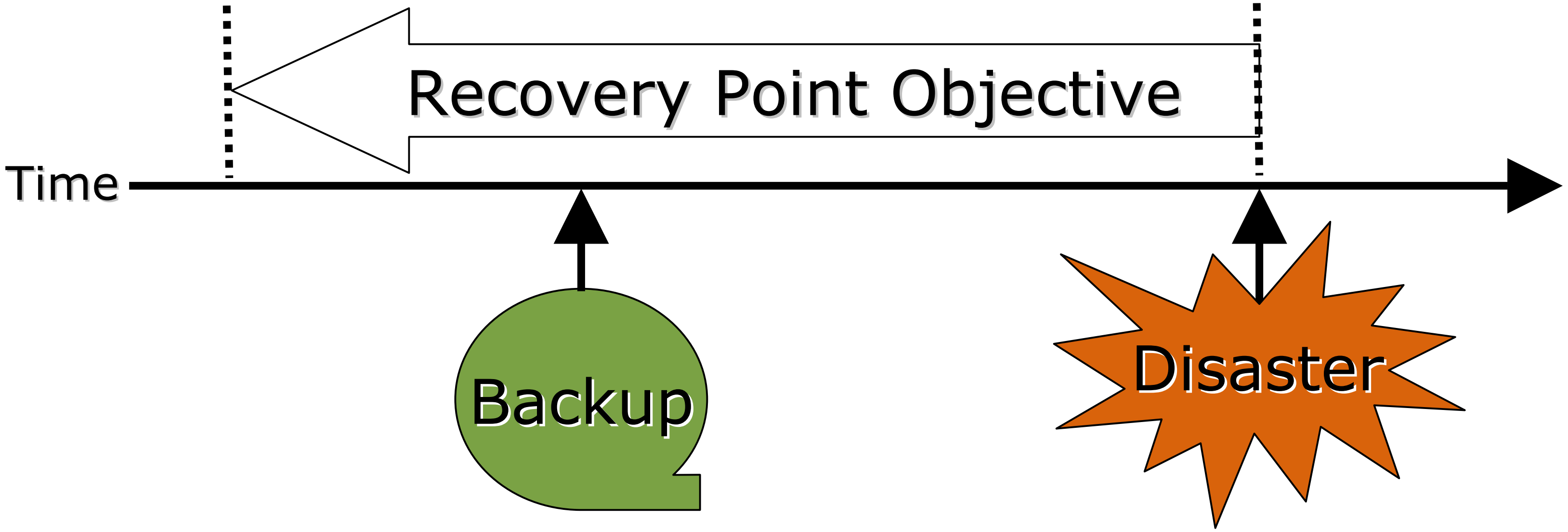
- Commonly-used metrics:
  - Recovery Point Objective (RPO):
    - Amount of data loss that is acceptable, if any
  - Recovery Time Objective (RTO):
    - Amount of downtime that is acceptable, if any

Disaster Tolerance vs. Disaster Recovery



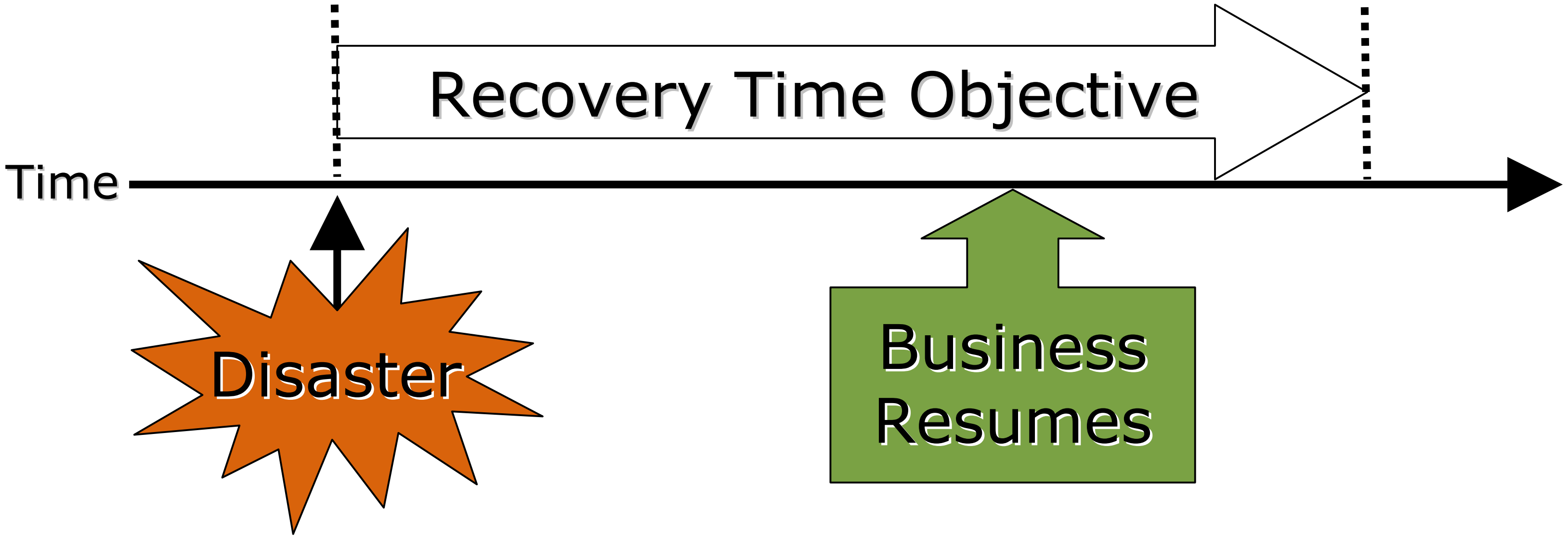
Recovery Point Objective (RPO)

- Recovery Point Objective is measured in terms of time
- RPO indicates the point in time to which one is able to recover the data after a failure, relative to the time of the failure itself
- RPO effectively quantifies the amount of data loss permissible before the business is adversely affected

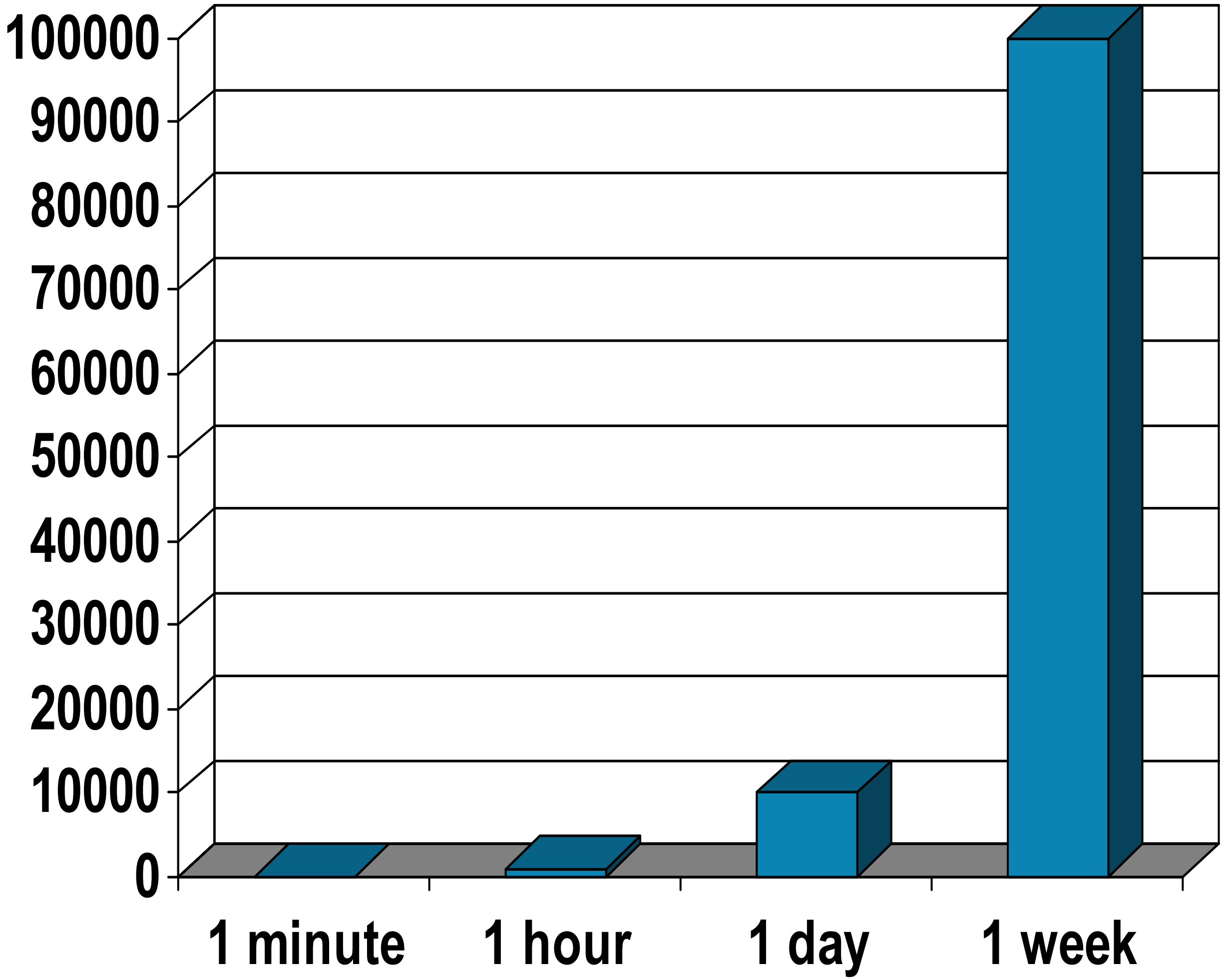


Recovery Time Objective (RTO)

- Recovery Time Objective is also measured in terms of time
- Measures downtime:
  - from time of disaster until business can continue
- Downtime costs vary with the nature of the business, and with outage length



Downtime Cost Varies with Outage Length



Examples of Business Requirements and RPO / RTO

- Greeting card manufacturer
  - RPO zero; RTO 3 days
- Online stock brokerage
  - RPO zero; RTO seconds
- Lottery
  - RPO zero; RTO minutes

Examples of Business Requirements and RPO / RTO

- ATM machine
  - RPO minutes; RTO minutes
- Semiconductor fabrication plant
  - RPO zero; RTO minutes; but data protection by geographical separation not needed

Recovery Point Objective (RPO)

- RPO examples, and technologies to meet them:
  - RPO of 24 hours: Backups at midnight every night to off-site tape drive, and recovery is to restore data from set of last backup tapes
  - RPO of 1 hour: Ship database logs hourly to remote site; recover database to point of last log shipment
  - RPO of zero: Mirror data strictly synchronously to remote site

Recovery Time Objective (RTO)

- RTO examples, and technologies to meet them:
  - RTO of 72 hours: Restore tapes to configure-to-order systems at vendor DR site
  - RTO of 12 hours: Restore tapes to system at hot site with systems already in place
  - RTO of 4 hours: Data vaulting to hot site with systems already in place
  - RTO of 1 hour: Disaster-tolerant cluster with controller-based cross-site disk mirroring
  - RTO of seconds: Disaster-tolerant cluster with bi-directional mirroring (precluding need for mirroring failover time); Distributed Lock Manager and Cluster File System, allowing applications to run at both sites simultaneously (precluding need for database and application failover and startup times); plus tie-breaking vote at 3rd site to prevent quorum loss

Technologies

- Clustering
- Inter-site links
- Foundation and Core Requirements for Disaster Tolerance
- Data replication schemes
- Quorum schemes

Clustering

- Allows a set of individual computer systems to be used together in some coordinated fashion

Cluster types

- Different types of clusters meet different needs:
  - *Scalability Clusters* allow multiple nodes to work on different portions of a sub-dividable problem
    - Workstation farms, compute clusters, Beowulf clusters
  - *Availability Clusters* allow one node to take over application processing if another node fails
- Our interest here is *Availability Clusters*

Availability Clusters

- Transparency of failover and degrees of resource sharing differ:
  - “Shared-Nothing” clusters
  - “Shared-Storage” clusters
  - “Shared-Everything” clusters

“Shared-Nothing” Clusters

- Data is partitioned among nodes
- No coordination is needed between nodes

“Shared-Storage” Clusters

- In simple “Fail-over” clusters, one node runs an application and updates the data; another node stands idly by until needed, then takes over completely
- In Shared-Storage clusters which are more advanced than simple “Fail-over” clusters, multiple nodes may access data, but typically one node at a time “serves” a file system to the rest of the nodes, and performs all coordination for that file system

“Shared-Everything” Clusters

- “Shared-Everything” clusters allow any application to run on any node or nodes
  - Disks are accessible to all nodes under a *Cluster File System*
  - File sharing and data updates are coordinated by a *Lock Manager*

Cluster File System

- Allows multiple nodes in a cluster to access data in a shared file system simultaneously
- View of file system is the same from any node in the cluster

Lock Manager

- Allows systems in a cluster to coordinate their access to shared resources:
  - Devices
  - File systems
  - Files
  - Database tables

Multi-Site Clusters

- Consist of multiple sites in different locations, with one or more systems at each site
- Systems at each site are all part of the same cluster, and may share resources
- Sites are typically connected by bridges (or bridge-routers; pure routers don't pass the special cluster protocol traffic required for many clusters)
  - e.g. SCS protocol for OpenVMS Clusters

Multi-Site Clusters: Inter-site Link(s)

- Sites linked by:
  - DS-3/T3 (E3 in Europe) or ATM circuits from a telecommunications vendor
  - Microwave link: DS-3/T3 or Ethernet bandwidths
  - Free-Space Optics link (short distance, low cost)
  - “Dark fiber” where available. ATM over SONET, or:
    - Ethernet over fiber (10 mb, Fast, Gigabit)
    - FDDI (up to 100 km)
    - Fibre Channel
    - Fiber links between Memory Channel switches (up to 3 km from hub)
  - Wave Division Multiplexing (WDM), in either Coarse or Dense Wave Division Multiplexing (DWDM) flavors
    - Any of the types of traffic that can run over a single fiber

Bandwidth of Inter-Site Link(s)

- Link bandwidth:
  - DS-3: 45 Mb/sec (or E3 at 34 Mb/sec)
  - ATM: Typically 155 or 622 Mb/sec
  - Ethernet: Fast (100 Mb/sec) or Gigabit (1 Gb/sec)
  - Fibre Channel: 1 or 2 Gb/sec
  - Memory Channel: 100 MB/sec
  - [D]WDM: Multiples of ATM, GbE, FC, etc.

Bandwidth of Inter-Site Link(s)

- Inter-site link minimum standards are in OpenVMS Cluster Software Product Description (SPD):
  - 10 megabits minimum data rate
    - This rules out T1 (1.5 Mb) links (and E1at 2 Mb in Europe), Fractional T1, Frame relay (256 Mb), ISDN (128 Mb), and DSL
  - “Minimize” packet latency
  - Low SCS packet retransmit rate:
    - Less than 0.1% retransmitted. Implies:

      Low packet-loss rate for bridges

      Low bit-error rate for links

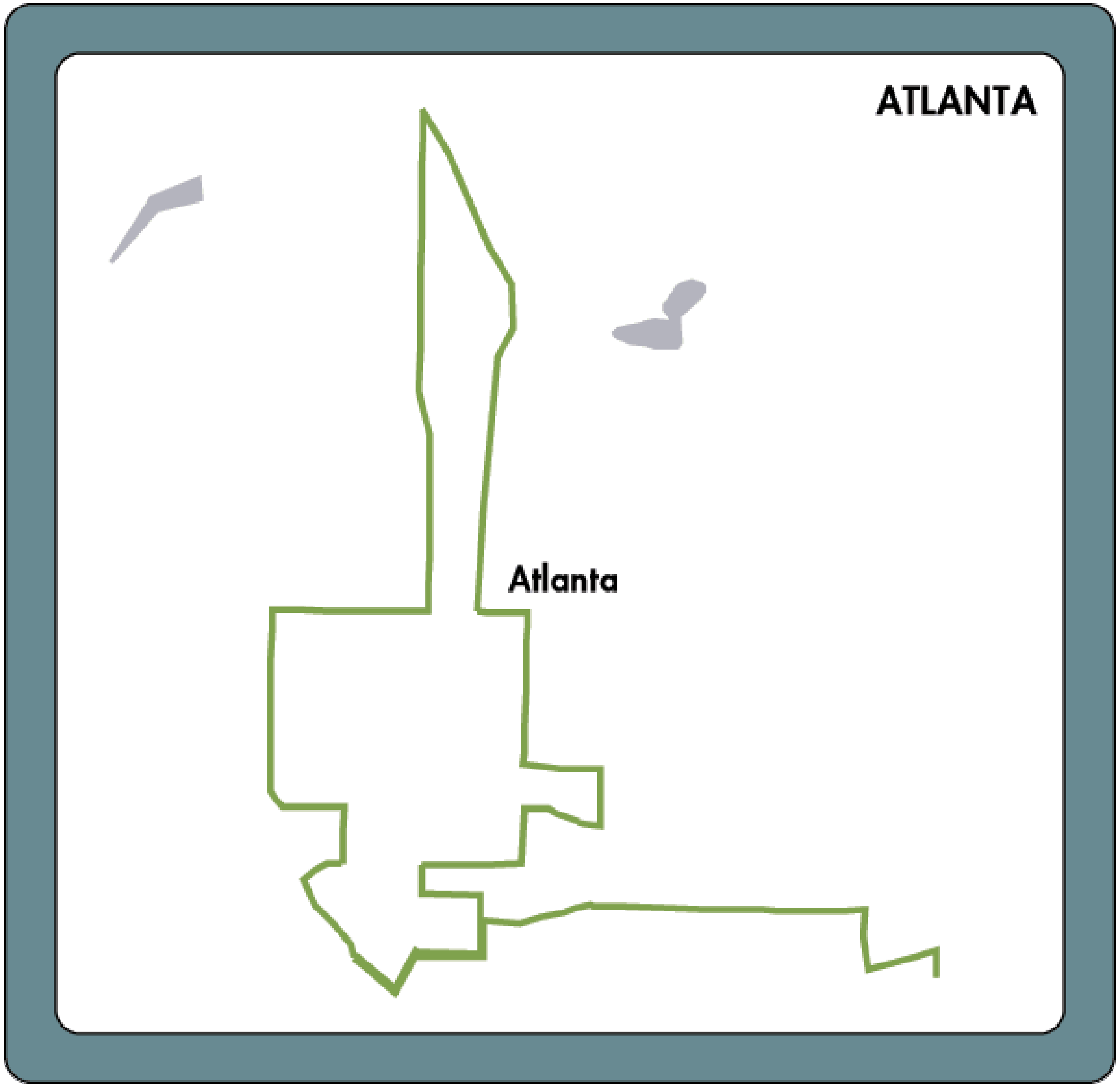
Bandwidth of Inter-Site Link

- Bandwidth affects performance of:
  - Volume Shadowing full copy operations
  - Volume Shadowing merge operations
- Link is typically only fully utilized during shadow copies
  - Size link(s) for acceptably-small shadowing Full Copy times
  - OpenVMS (PEDRIVER) can use multiple links in parallel quite effectively
    - Significant improvements in this area in OpenVMS 7.3

Inter-Site Link Choices

- Service type choices
  - Telco-provided data circuit service, own microwave link, FSO link, dark fiber?
  - Dedicated bandwidth, or shared pipe?
  - Single or multiple (redundant) links? If multiple links, then:
    - Diverse paths?
    - Multiple vendors?

Dark Fiber Availability Example



Source: Metro Media Fiber Networks mmfn.com

Inter-Site Link: Network Gear

- Bridge implementations must not drop small packets under heavy loads
  - SCS Hello packets are small packets
  - If two in a row get lost, a node without redundant LANs will see a Virtual Circuit closure; if failure lasts too long, node will do a CLUEXIT bugcheck

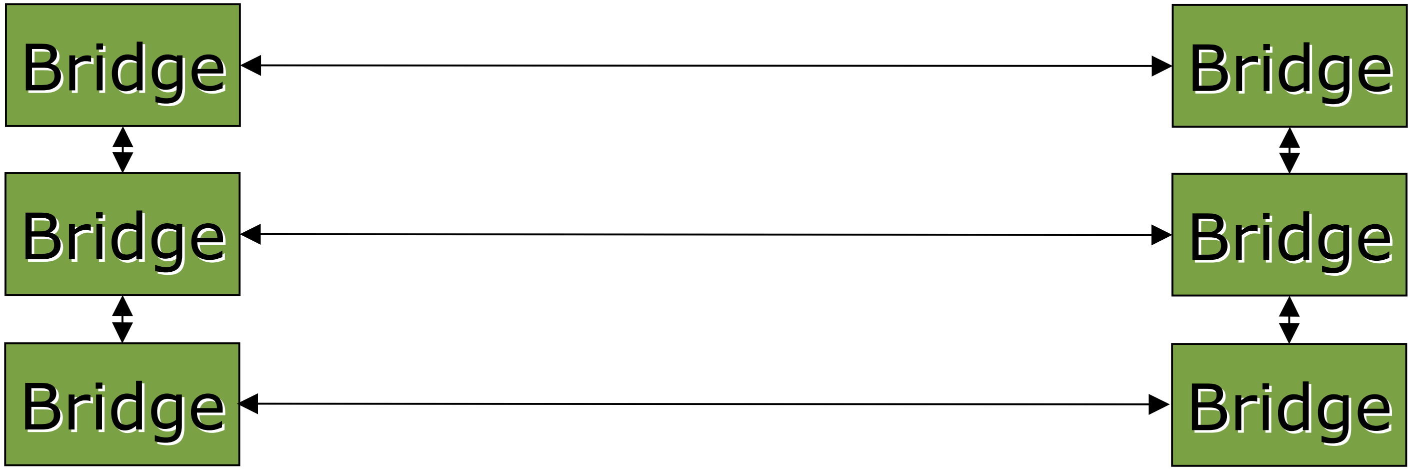
Inter-Site Links

- It is desirable for the cluster to be able to survive a bridge/router reboot for a firmware upgrade or switch reboot
  - If only one inter-site link is available, cluster nodes will just have to wait during this time
- Spanning Tree reconfiguration takes time
  - Default Spanning Tree protocol timers often cause delays longer than the default value for RECNXINTERVAL
  - Consider raising RECNXINTERVAL parameter
    - Default is 20 seconds
    - It's a dynamic parameter

Redundant Inter-Site Links

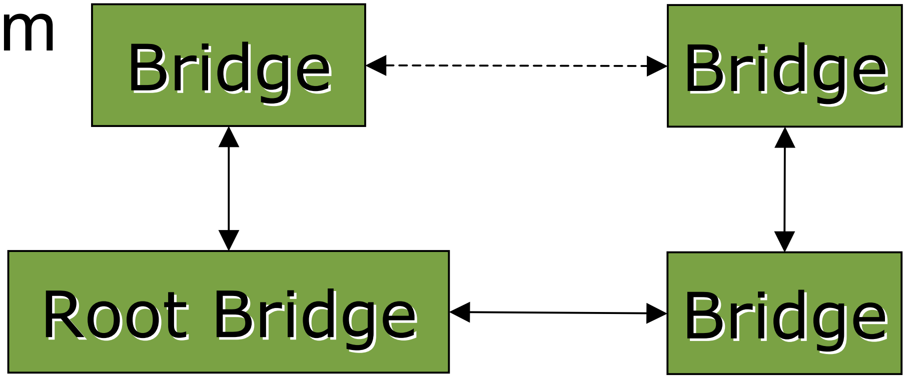
– If multiple inter-site links are used, but they are joined together into one extended LAN:

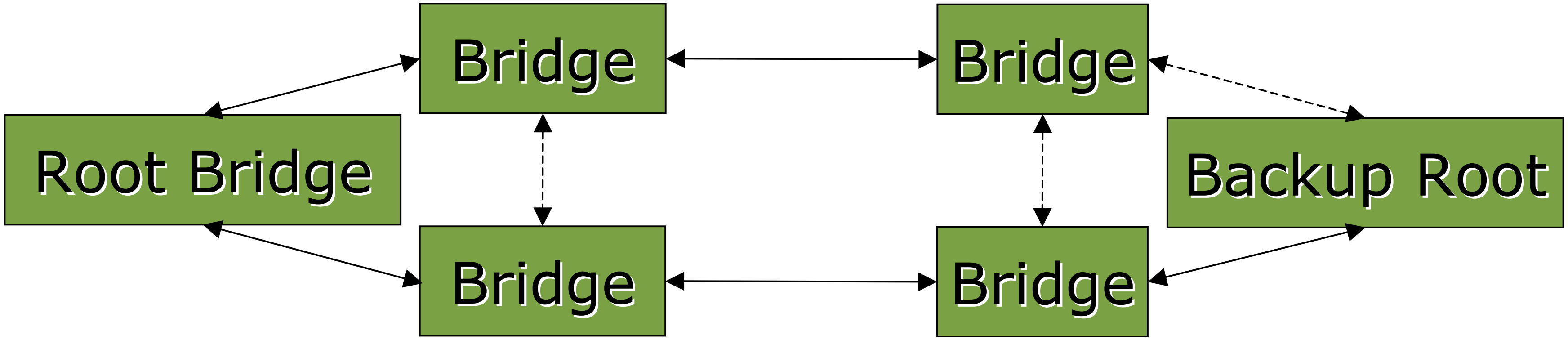
1. One or more of the inter-site links may be disabled by the Spanning Tree protocol
2. The Spanning Tree reconfiguration time (with default timer values) is typically too long for the default value of RECNXINTERVAL (20 seconds)



Redundant Inter-Site Links

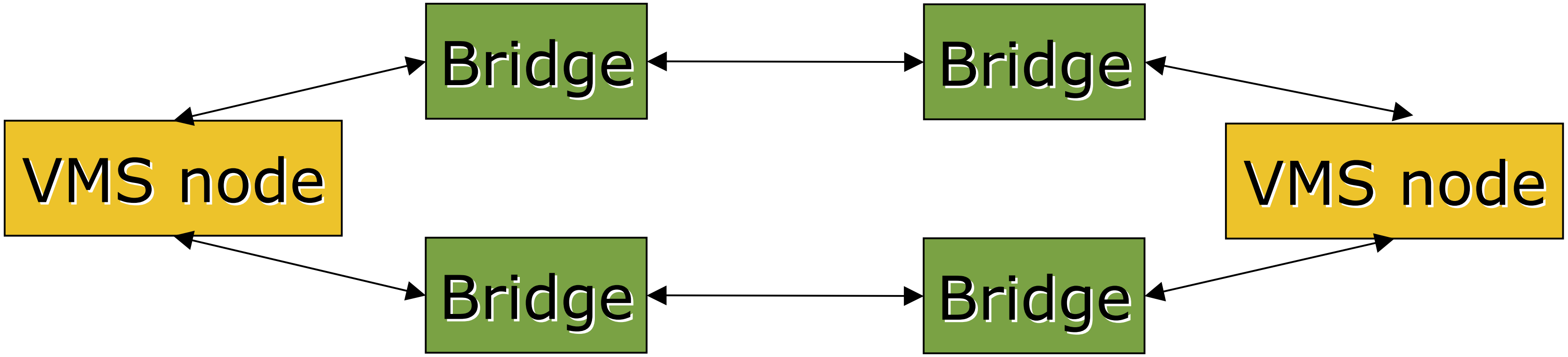
- One may want to carefully select bridge root priorities so that one of the (expensive) inter-site links is not turned off by the Spanning Tree algorithm





Redundant Inter-Site Links

- Multiple inter-site links can instead be configured as isolated, independent LANs, with independent Spanning Trees
  - There is a very low probability of experiencing Spanning Tree reconfigurations at once on multiple LANs when they are completely separate
  - Use multiple LAN adapters in each system, with one connected to each of the independent inter-site LANs



Inter-Site Link Monitoring

- Where redundant LAN hardware is in place, use the LAVC$FAILURE_ANALYSIS tool from SYS$EXAMPLES:
  - It monitors and reports, via OPCOM messages, LAN component failures and repairs
  - More detail later

Disaster-Tolerant Clusters: Foundation

- Goal: Survive loss of up to one entire datacenter
- Foundation:
  - Two or more datacenters a “safe” distance apart
  - Cluster software for coordination
  - Inter-site link for cluster interconnect
  - Data replication of some sort for 2 or more identical copies of data, one at each site:
    - Volume Shadowing for OpenVMS, StorageWorks DRM or Continuous Access, database replication, etc.

Disaster-Tolerant Clusters

- Foundation:
  - Management and monitoring tools
    - Remote system console access or KVM system
    - Failure detection and alerting, for things like:
      - Network (especially inter-site link) monitoring
      - Shadowset member loss
      - Node failure
    - Quorum recovery tool (especially for 2-site clusters)

Disaster-Tolerant Clusters

- Foundation:
  - Configuration planning and implementation assistance, and staff training
    - HP recommends Disaster Tolerant Cluster Services (DTCS) package

Disaster-Tolerant Clusters

- Foundation:
  - History of packages available for Disaster-Tolerant Cluster configuration planning, implementation assistance, and training:
    - HP currently offers Disaster Tolerant Cluster Services (DTCS) package
      - Monitoring based on tools from Heroix
    - Formerly Business Recovery Server (BRS)
      - Monitoring based on Polycenter tools (Console Manager, System Watchdog, DECmcc) now owned by Computer Associates
    - and before that, Multi-Datacenter Facility (MDF)

Disaster-Tolerant Clusters

- Management and monitoring toolset choices:
  - Remote system console access
    - Heroix RoboCentral; CA Unicenter Console Management for OpenVMS (formerly Command/IT, formerly Polycenter Console Manager); TECSys Development Inc. ConsoleWorks; Ki Networks Command Line Interface Manager (CLIM)
  - Failure detection and alerting
    - Heroix RoboMon; CA Unicenter System Watchdog for OpenVMS (formerly Watch/IT, formerly Polycenter System Watchdog); BMC Patrol
    - HP also has a software product called CockpitMgr designed specifically for disaster-tolerant OpenVMS Cluster monitoring and control, and based on the principle of using only OpenVMS systems to monitor and control OpenVMS systems. See http://www.hp.be/cockpitmgr/ and http://h71000.www7.hp.com/openvms/journal/v1/mgclus.pdf

Disaster-Tolerant Clusters

- Management and monitoring toolset choices:
  - Network monitoring (especially inter-site links)
    - HP OpenView; Unicenter TNG; Tivoli; ClearViSN; CiscoWorks; etc.
  - Quorum recovery tool
    - DECamds / Availability Manager
    - DTCS or BRS integrated tools (which talk to the DECamds/AM RMDRIVER client on cluster nodes)

Disaster-Tolerant Clusters

- Management and monitoring toolset choices:
  - Performance Management
    - HP ECP (CP/Collect & CP/Analyze)
    - Perfcap PAWZ, Analyzer, & Planner
    - Unicenter Performance Management for OpenVMS (formerly Polycenter Performance Solution Data Collector and Performance Analyzer, formerly SPM and VPA) from Computer Associates
    - Fortel SightLine/Viewpoint (formerly Datametrics)
    - BMC Patrol
    - etc.

Disaster-Tolerant Clusters

- Foundation:
  - Carefully-planned procedures for:
    - Normal operations
    - Scheduled downtime and outages
    - Detailed diagnostic and recovery action plans for various failure scenarios

Disaster Tolerance: Core Requirements

- Foundation:
  - Complete redundancy in facilities and hardware:
    - Second site with its own storage, networking, computing hardware, and user access mechanisms is put in place
      - No dependencies on the 1st site are allowed
    - Monitoring, management, and control mechanisms are in place to facilitate fail-over
    - Sufficient computing capacity is in place at the 2nd site to handle expected workloads by itself if the 1st site is destroyed

Disaster Tolerance: Core Requirements

- Foundation:
  - Data Replication:
    - Data is constantly replicated to or copied to a 2nd site, so data is preserved in a disaster
    - Recovery Point Objective (RPO) determines which technologies are acceptable

Planning for Disaster Tolerance

- Remembering that the goal is to continue operating despite loss of an entire datacenter
  - All the pieces must be in place to allow that:
    - User access to both sites
    - Network connections to both sites
    - Operations staff at both sites
  - Business can't depend on *anything* that is *only* at either site

Disaster Tolerance: Core Requirements

- If all these requirements are met, there may be as little as zero data lost and as little as seconds of delay after a disaster before the surviving copy of data can actually be used

Planning for Disaster Tolerance

- Sites must be carefully selected to avoid hazards common to both, and loss of both datacenters at once as a result
- Make them a “safe” distance apart
  - This must be a compromise. Factors:
    - Business needs
    - Risks
    - Interconnect costs
    - Performance (inter-site latency)
    - Ease of travel between sites
    - Politics, legal requirements (e.g. privacy laws)

Planning for Disaster Tolerance: What is a "Safe Distance"

- Analyze likely hazards of proposed sites:
  - Fire (building, forest, gas leak, explosive materials)
  - Storms (Tornado, Hurricane, Lightning, Hail, Ice)
  - Flooding (excess rainfall, dam failure, storm surge, broken water pipe)
  - Earthquakes, Tsunamis

Planning for Disaster Tolerance: What is a "Safe Distance"

- Analyze likely hazards of proposed sites:
  - Nearby transportation of hazardous materials (highway, rail, ship/barge)
  - Terrorist (or disgruntled customer) with a bomb or weapon
  - Enemy attack in war (nearby military or industrial targets)
  - Civil unrest (riots, vandalism)

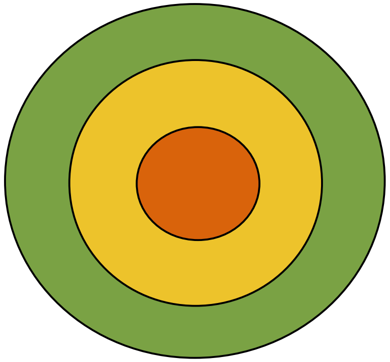
Planning for Disaster Tolerance: Site Separation

- Select site separation direction:
  - Not along same earthquake fault-line
  - Not along likely storm tracks
  - Not in same floodplain or downstream of same dam
  - Not on the same coastline
  - Not in line with prevailing winds (that might carry hazardous materials)

Planning for Disaster Tolerance: Site Separation

- Select site separation distance (in a “safe” direction):
  - 1 mile: protects against most building fires, gas leak, terrorist bombs, armed intruder
  - 10 miles: protects against most tornadoes, floods, hazardous material spills, release of poisonous gas, non-nuclear military bombs
  - 100 miles: protects against most hurricanes, earthquakes, tsunamis, forest fires, “dirty” bombs, biological weapons, and possibly military nuclear attacks

Threat Radius



Planning for Disaster Tolerance: Providing Redundancy

- Redundancy must be provided for:
  - Datacenter and facilities (A/C, power, user workspace, etc.)
  - Data
    - And data feeds, if any
  - Systems
  - Network
  - User access

Planning for Disaster Tolerance

- Also plan for continued operation *after* a disaster
  - Surviving site will likely have to operate alone for a long period before the other site can be repaired or replaced

Planning for Disaster Tolerance

- Plan for continued operation *after* a disaster
  - Provide redundancy *within* each site
    - Facilities: Power feeds, A/C
    - Mirroring or RAID to protect disks
      - Obvious solution for 2-site clusters would be 4-member shadowsets, but the limit is 3 members. Typical workarounds are:
        - Shadow 2-member controller-based mirrorsets at each site, or
        - Have 2 members at one site and a 2-member mirrorset as the single member at the other site
        - Have 3 sites, with one shadow member at each site
    - Clustering for servers
    - Network redundancy

Planning for Disaster Tolerance

- Plan for continued operation *after* a disaster
  - Provide enough capacity within each site to run the business alone if the other site is lost
    - and handle normal workload growth rate

Planning for Disaster Tolerance

- Plan for continued operation *after* a disaster
  - Having 3 sites is an option to seriously consider:
    - Leaves two redundant sites after a disaster
    - Leaves 2/3 of processing capacity instead of just ½ after a disaster

Cross-site Data Replication Methods

- Hardware
  - Storage controller
- Software
  - Host software Volume Shadowing, disk mirroring, or file-level mirroring
  - Database replication or log-shipping
  - Transaction-processing monitor or middleware with replication functionality

Data Replication in Hardware

- HP StorageWorks Data Replication Manager (DRM) or Continuous Access (CA)
- HP StorageWorks XP (formerly HP SureStore E Disk Array XP) with Continuous Access (CA) XP
- EMC Symmetrix Remote Data Facility (SRDF)

Data Replication in Software

- Host software volume shadowing or disk mirroring:
  - Volume Shadowing Software for OpenVMS
  - MirrorDisk/UX for HP-UX
  - Veritas VxVM with Volume Replicator extensions for Unix and Windows
  - Fault Tolerant (FT) Disk on Windows
- Some other O/S platforms have software products which can provide file-level mirroring

Data Replication in Software

- Database replication or log-shipping
  - Replication
    - e.g. Oracle DataGuard (formerly Oracle Standby Database)
  - Database backups plus “Log Shipping”

Data Replication in Software

- TP Monitor/Transaction Router
  - e.g. HP Reliable Transaction Router (RTR) Software on OpenVMS, UNIX, and Windows

Data Replication in Hardware

- Data mirroring schemes
  - Synchronous
    - Slower, but less chance of data loss
      - Beware: Some hardware solutions can still lose the last write operation before a disaster
  - Asynchronous
    - Faster, and works for longer distances
      - but can lose minutes’ worth of data (more under high loads) in a site disaster
  - Most products offer you a choice of using either method

Data Replication in Hardware

- Mirroring is of sectors on disk
  - So operating system / applications must flush data from memory to disk for controller to be able to mirror it to the other site

Data Replication in Hardware

- Resynchronization operations
  - May take significant time and bandwidth
  - May or may not preserve a consistent copy of data at the remote site until the copy operation has completed
  - May or may not preserve write ordering during the copy

Data Replication: Write Ordering

- File systems and database software may make some assumptions on write ordering and disk behavior
  - For example, a database may write to a journal log, wait until that I/O is reported as being complete, then write to the main database storage area
    - During database recovery operations, its logic may depend on these write operations having been completed to disk in the expected order

Data Replication: Write Ordering

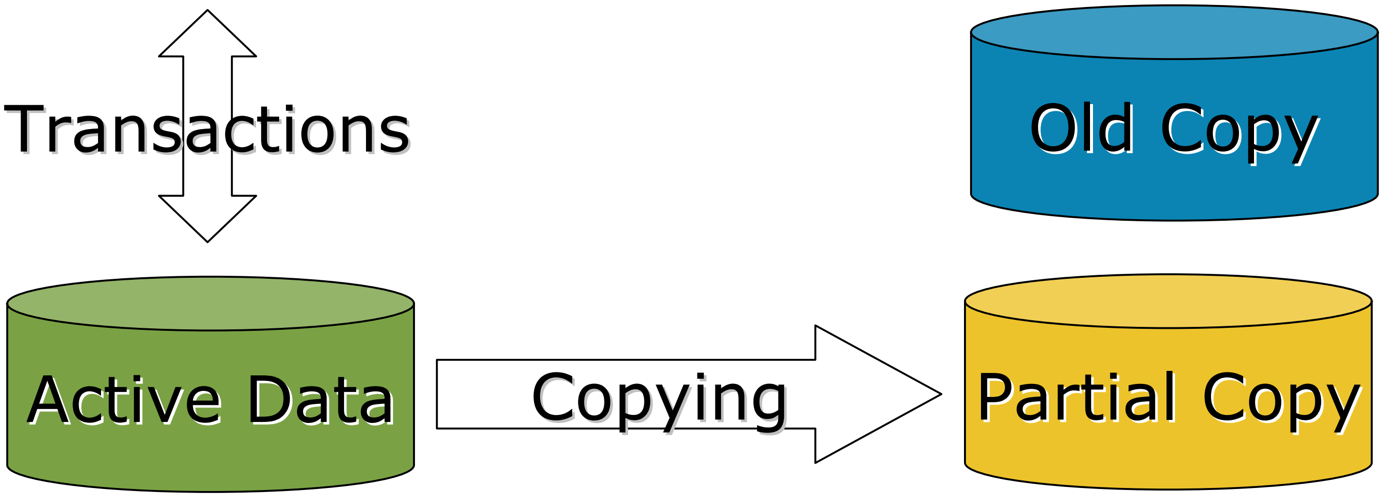
- Some controller-based replication methods copy data on a track-by-track basis for efficiency instead of exactly duplicating individual write operations
  - This may change the effective ordering of write operations within the remote copy

Data Replication: Write Ordering

- When data needs to be re-synchronized at a remote site, some replication methods (both controller-based and host-based) similarly copy data on a track-by-track basis for efficiency instead of exactly duplicating writes
- This may change the effective ordering of write operations within the remote copy
- The output volume may be inconsistent and unreadable until the resynchronization operation completes

Data Replication: Write Ordering

- It may be advisable in this case to preserve an earlier (consistent) copy of the data, and perform the resynchronization to a different set of disks, so that if the source site is lost during the copy, at least one copy of the data (albeit out-of-date) is still present



Data Replication in Hardware: Write Ordering

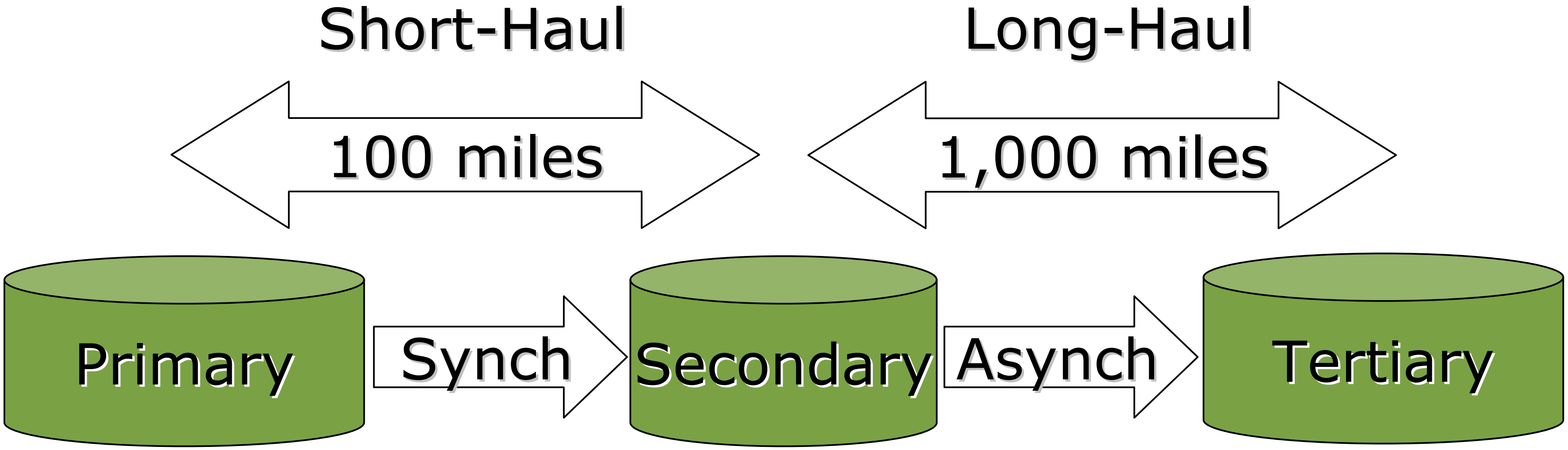
- Some products provide a guarantee of original write ordering on a disk (or even across a set of disks)
- Some products can even preserve write ordering during resynchronization operations, so the remote copy is always consistent (as of some point in time) during the entire resynchronization operation

Data Replication: Performance over a Long Distance

- Replication performance may be affected by latency due to the speed of light over the distance between sites
- Greater (and thus safer) distances between sites implies greater latency
- Rule of thumb:
  - 1 millisecond per 100 miles, one-way, or
  - 1 millisecond per 50 miles, round-trip

Data Replication: Performance over a Long Distance

- With some solutions, it may be possible to synchronously replicate data to a nearby "short-haul" site, and asynchronously replicate from there to a more-distant site
  - This is sometimes called "cascaded" data replication

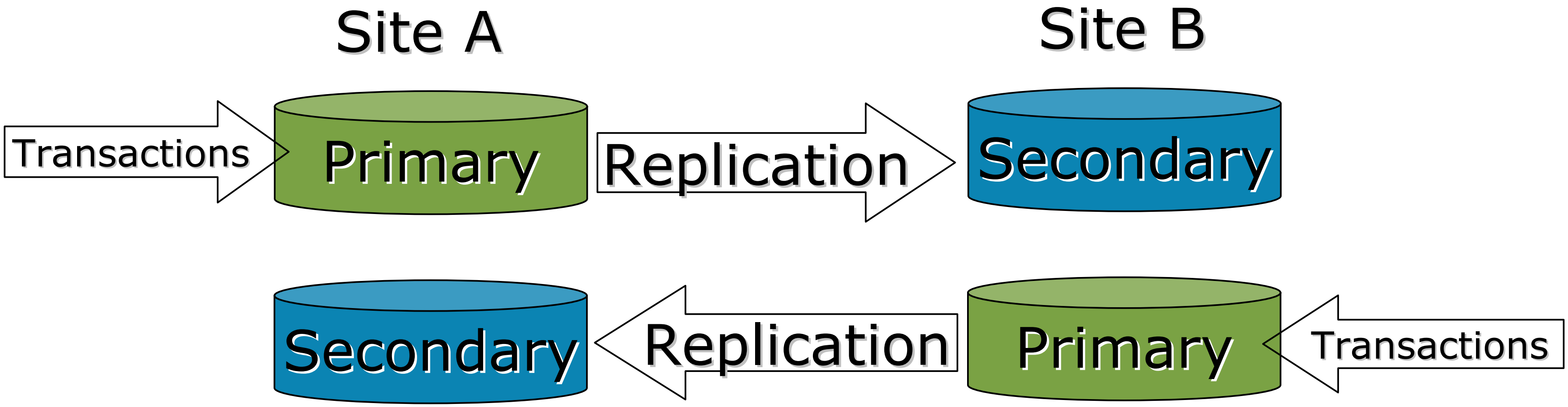


Data Replication: Performance During Re-Synchronization

- Re-synchronization operations can generate a high data rate on inter-site links
- Excessive re-synchronization time increases Mean Time To Repair (MTTR) after a site failure or outage
- Acceptable re-synchronization times and link costs may be the major factors in selecting inter-site link(s)

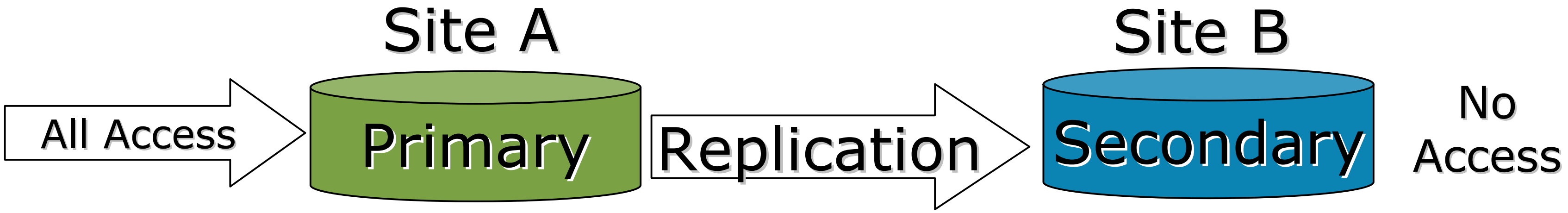
Data Replication in Hardware: Copy Direction

- Most hardware-based solutions can only replicate a given set of data in one direction or the other
- Some can be configured replicate some disks on one direction, and other disks in the opposite direction
  - This way, different applications might be run at each of the two sites



Data Replication in Hardware: Disk Unit Access

- All access to a disk unit is typically from only one of the controllers at a time
  - Data cannot be accessed through the controller at the other site
    - Data might be accessible to systems at the other site via a Fibre Channel inter-site link, or by going through the MSCP Server on a VMS node
  - Read-only access may be possible at remote site with some products (so-called "Productive Protection")
  - Failover involves controller commands
    - Manual, or manually-initiated scripts
      - 15 minutes to 1 hour range of minimum failover time



Data Replication in Hardware: Multiple Copies

- Some products allow replication to:
  - A second unit at the same site
  - Multiple remote units or sites at a time (“M x N“ configurations)
- In contrast, OpenVMS Volume Shadowing allows up to 3 copies, spread across up to 3 sites

Data Replication in Hardware: Copy Direction

- Few or no hardware solutions can replicate data between sites in both directions on the same shadowset/mirrorset
  - But Host-based OpenVMS Volume Shadowing can do this
  - If this could be done in a hardware solution, host software would still have to coordinate any disk updates to the same set of blocks from both sites
    - e.g. OpenVMS Cluster Software, or Oracle Parallel Server or 9i/RAC
- This capability is required to allow the same application to be run on cluster nodes at both sites simultaneously

Managing Replicated Data

- With copies of data at multiple sites, one must take care to ensure that:
    - Both copies are always equivalent, or, failing that,
        - Users always access the most up-to-date copy

Managing Replicated Data

- If the inter-site link fails, both sites might conceivably continue to process transactions, and the copies of the data at each site would continue to diverge over time
- This is called a “Partitioned Cluster”, or “Split-Brain Syndrome”
- The most common solution to this potential problem is a Quorum-based scheme
  - Access and updates are only allowed to take place on one set of data

Quorum Schemes

- Idea comes from familiar parliamentary procedures
- Systems are given votes
- Quorum is defined to be a simple majority (just over half) of the total votes

Quorum Schemes

- In the event of a communications failure,
  - Systems in the minority voluntarily suspend (OpenVMS) or stop (MC/ServiceGuard) processing, while
  - Systems in the majority can continue to process transactions

Quorum Scheme

- If a cluster member is not part of a cluster with quorum, OpenVMS keeps it from doing any harm by:
  - Putting all disks into Mount Verify state, thus stalling all disk I/O operations
  - Requiring that all processes have the QUORUM capability bit before they can run
  - Clearing the QUORUM capability bit on all CPUs in the system, thus preventing any process from being scheduled to run on a CPU and doing any work
    - OpenVMS many years ago looped at IPL 4 instead

Quorum Schemes

- To handle cases where there are an even number of votes
  - For example, with only 2 systems,
  - Or half of the votes are at each of 2 sites

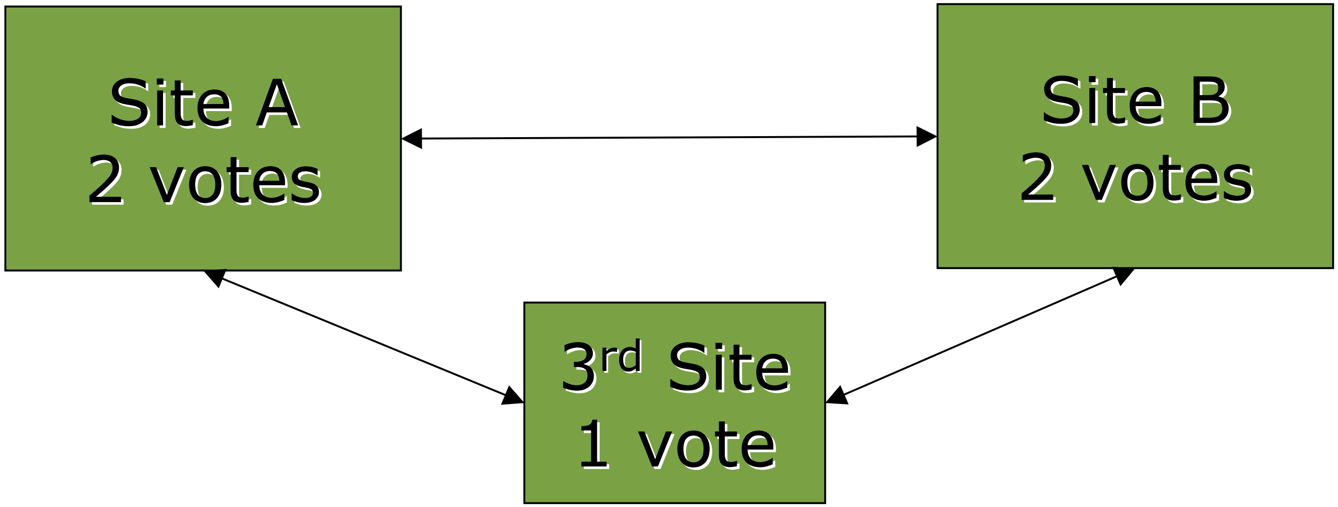
  provision may be made for
  - a tie-breaking vote, or
  - human intervention

Quorum Schemes: Tie-breaking vote

- This can be provided by a disk:
  - Quorum Disk for OpenVMS Clusters or TruClusters or MSCS
  - Cluster Lock Disk for MC/ServiceGuard
- Or by a system with a vote, located at a 3rd site
  - Additional cluster member node for OpenVMS Clusters or TruClusters (called a "quorum node") or MC/ServiceGuard clusters (called an "arbitrator node")
  - Software running on a non-clustered node or a node in another cluster
    - e.g. Quorum Server for MC/ServiceGuard

Quorum configurations in Multi-Site Clusters

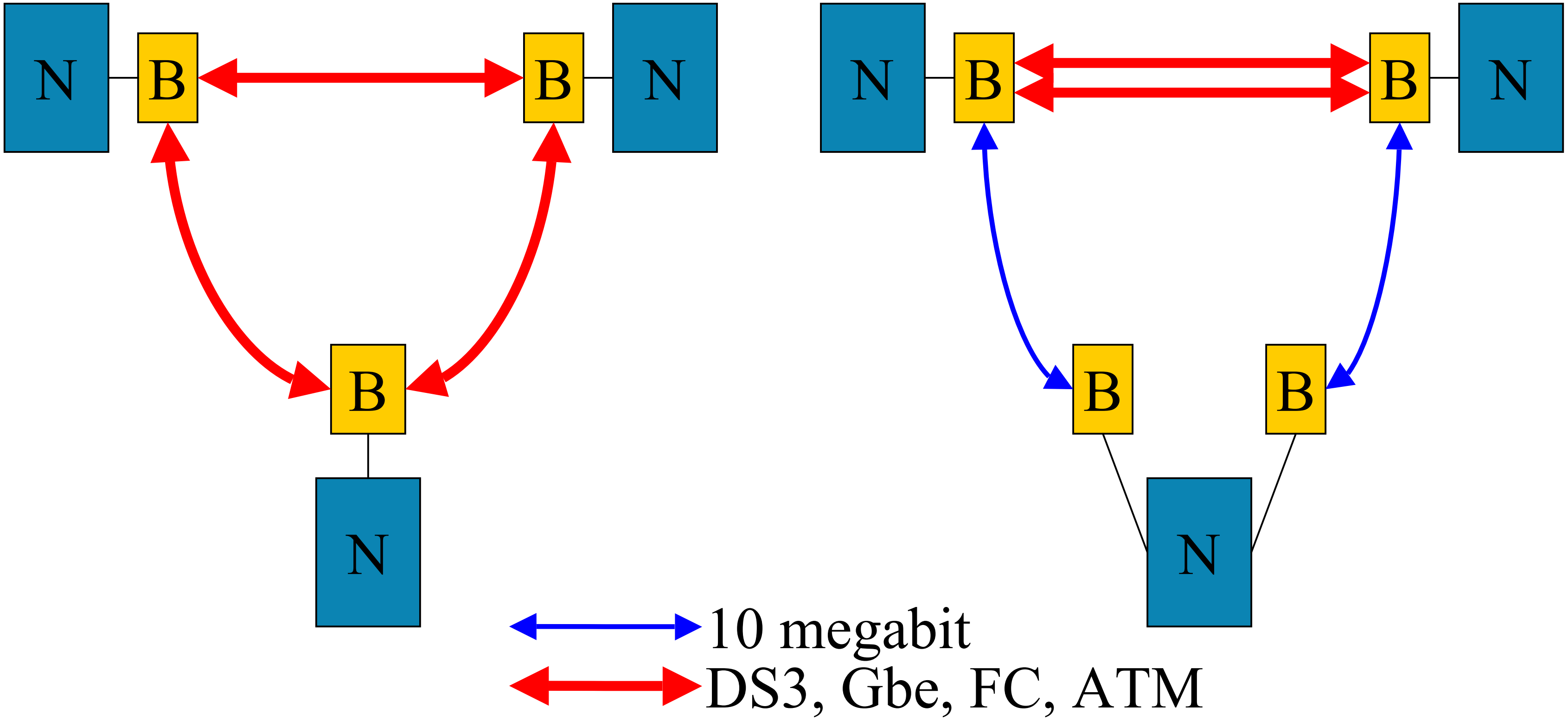
- 3 sites, equal votes in 2 sites
  - Intuitively ideal; easiest to manage & operate
  - 3rd site serves as tie-breaker
  - 3rd site might contain only a “quorum node”, “arbitrator node”, or “quorum server”



Quorum configurations in Multi-Site Clusters

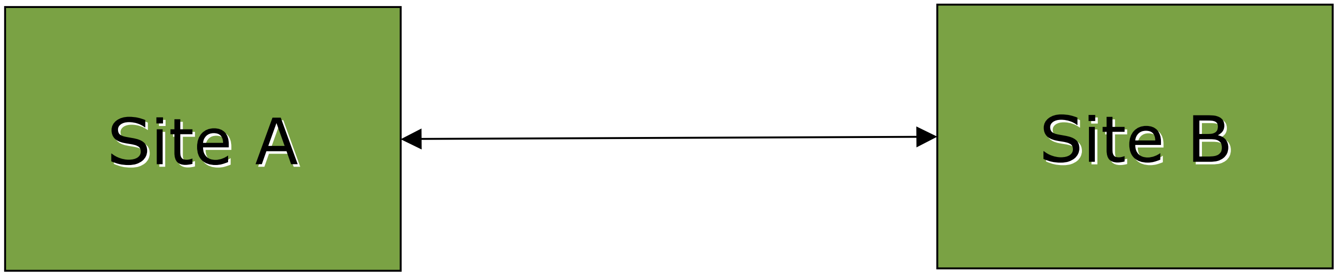
- 3 sites, equal votes in 2 sites
  - Hard to do in practice, due to cost of inter-site links beyond on-campus distances
    - Could use links to quorum site as backup for main inter-site link if links are high-bandwidth and connected together
    - Could use 2 less-expensive, lower-bandwidth links to quorum site, to lower cost
      - But OpenVMS SPD requires a minimum of 10 megabits bandwidth for any link

Quorum configurations in 3-Site Clusters



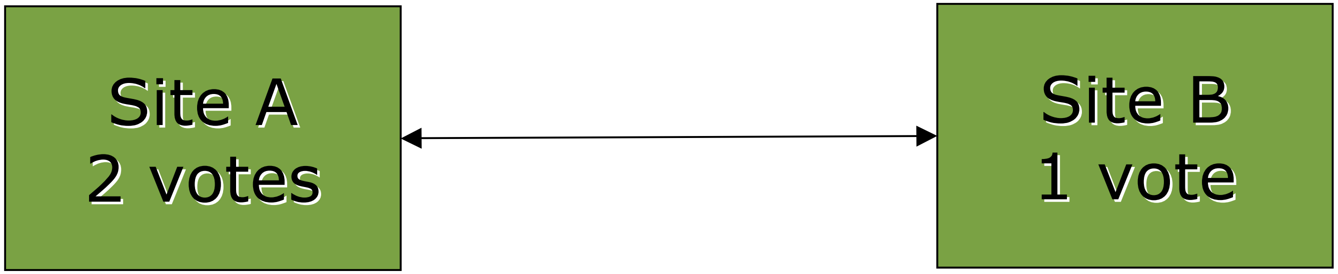
Quorum configurations in Multi-Site Clusters

- 2 sites:
  - Most common & most problematic:
    - How do you arrange votes? Balanced? Unbalanced?
    - If votes are balanced, how do you recover from loss of quorum which will result when either site or the inter-site link fails?



Quorum configurations in Two-Site Clusters

- One solution: Unbalanced Votes
  - More votes at one site
  - Site with more votes can continue without human intervention in the event of loss of the other site or the inter-site link
  - Site with fewer votes pauses or stops on a failure and requires manual action to continue after loss of the other site



Site A
2 votes

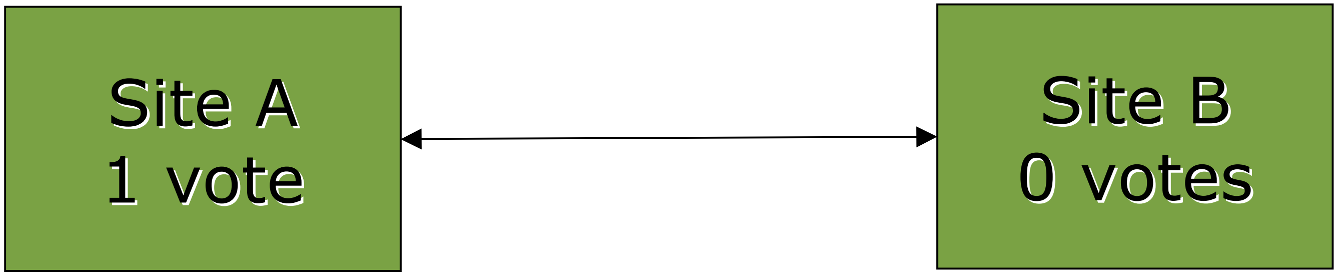
Site B
1 vote

Can continue automatically

Requires manual intervention to continue alone

Quorum configurations in Two-Site Clusters

- Unbalanced Votes
  - Very common in remote-shadowing-only clusters (not fully disaster-tolerant)
    - 0 votes is a common choice for the remote site in this case
      - but that has its dangers



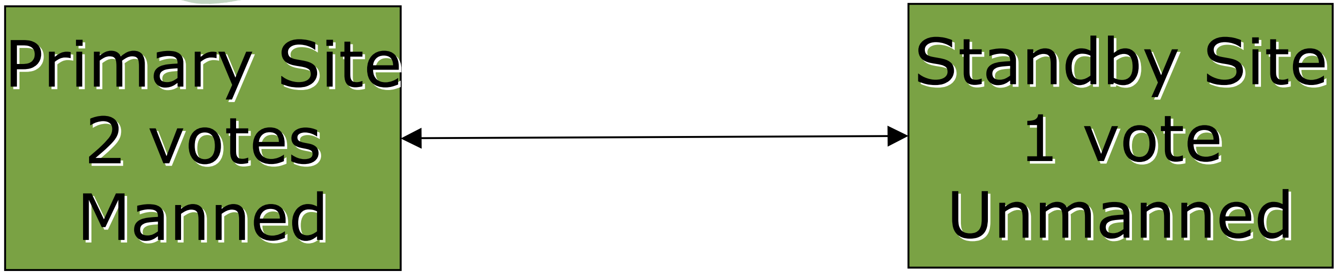
Can continue automatically

Requires manual intervention to continue alone

Quorum configurations in Two-Site Clusters

- Unbalanced Votes
  - Common mistake:
    - Give more votes to Primary site, and
    - Leave Standby site unmanned
    - Result: cluster can’t run without Primary site or human intervention at the (unmanned) Standby site

Lights-out



Primary Site
2 votes
Manned

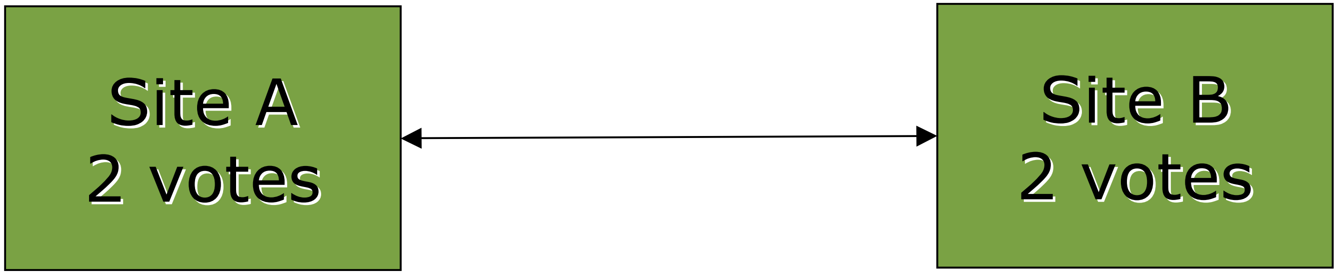
Standby Site
1 vote
Unmanned

Can continue automatically

Requires manual intervention to continue alone

Quorum configurations in Two-Site Clusters

- Balanced Votes
  - Equal votes at each site
  - Manual action required to restore quorum and continue processing in the event of either:
    - Site failure, or
    - Inter-site link failure



Requires manual intervention to continue alone

Requires manual intervention to continue alone

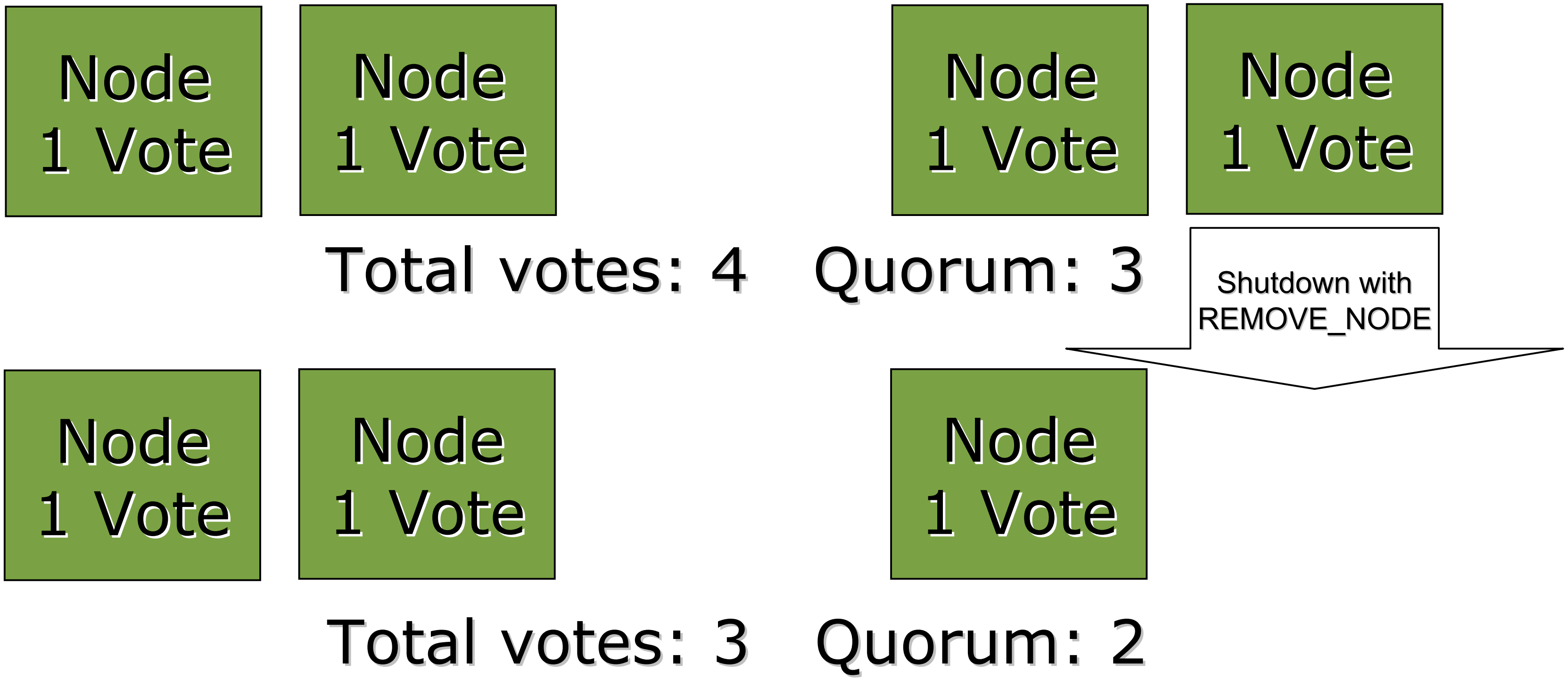
Quorum Recovery Methods

Methods for human intervention to restore quorum

- Software interrupt at IPL 12 from console
  - IPC> Q
- DECamds or Availability Manager Console:
  - System Fix; Adjust Quorum
- DTCS or BRS integrated tool, using same RMDRIVER (DECamds/AM client) interface

Quorum configurations in Two-Site Clusters

- Balanced Votes
  - Note: Using REMOVE_NODE option with SHUTDOWN.COM (post V6.2) when taking down a node effectively “unbalances” votes:



Node 1 Vote | Node 1 Vote | Node 1 Vote | Node 1 Vote

Total votes: 4 Quorum: 3

Shutdown with REMOVE_NODE

Node 1 Vote | Node 1 Vote | Node 1 Vote

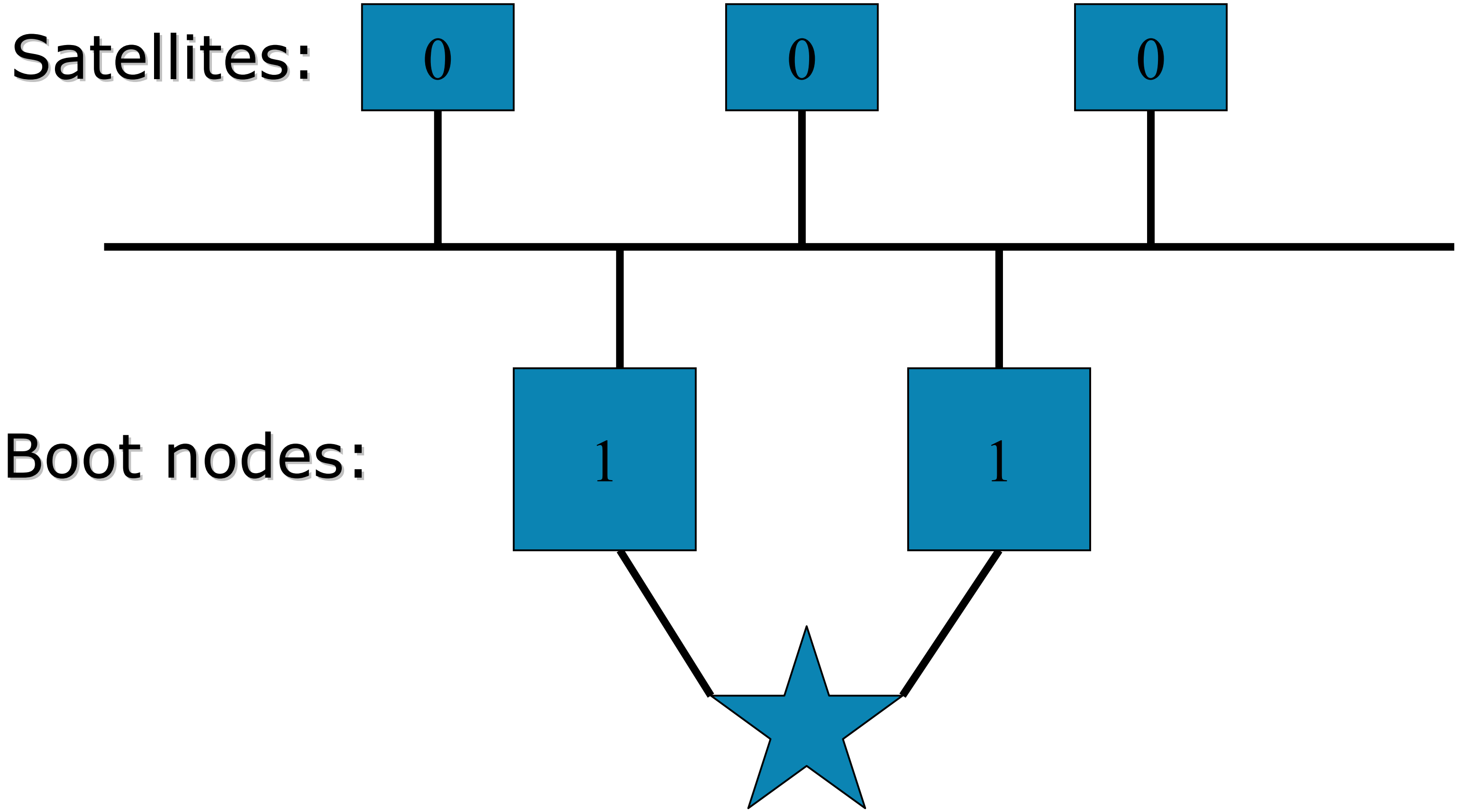
Total votes: 3 Quorum: 2

Optimal Sub-cluster Selection

- Connection Manager compares potential node subsets that could make up the surviving portion of the cluster
  - Picks sub-cluster with the most votes; or,
  - If vote counts are equal, picks sub-cluster with the most nodes; or,
  - If node counts are equal, arbitrarily picks a winner
    - based on comparing SCSSYSTEMID values within the set of nodes with the most-recent cluster software revision

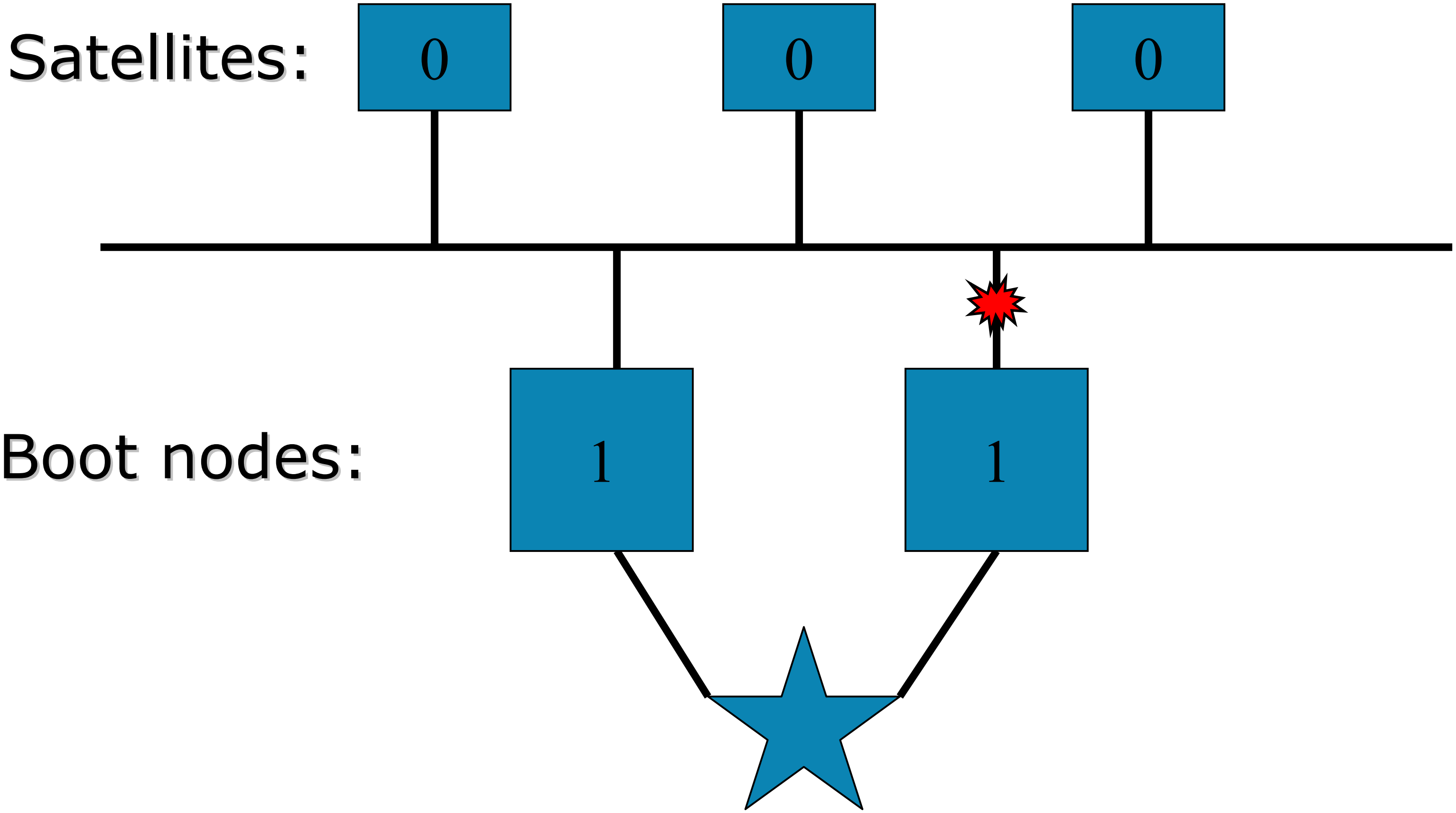
Optimal Sub-cluster Selection: Examples
Boot nodes and satellites

- Most configurations with satellite nodes give votes to disk/boot servers and set VOTES=0 on all satellite nodes

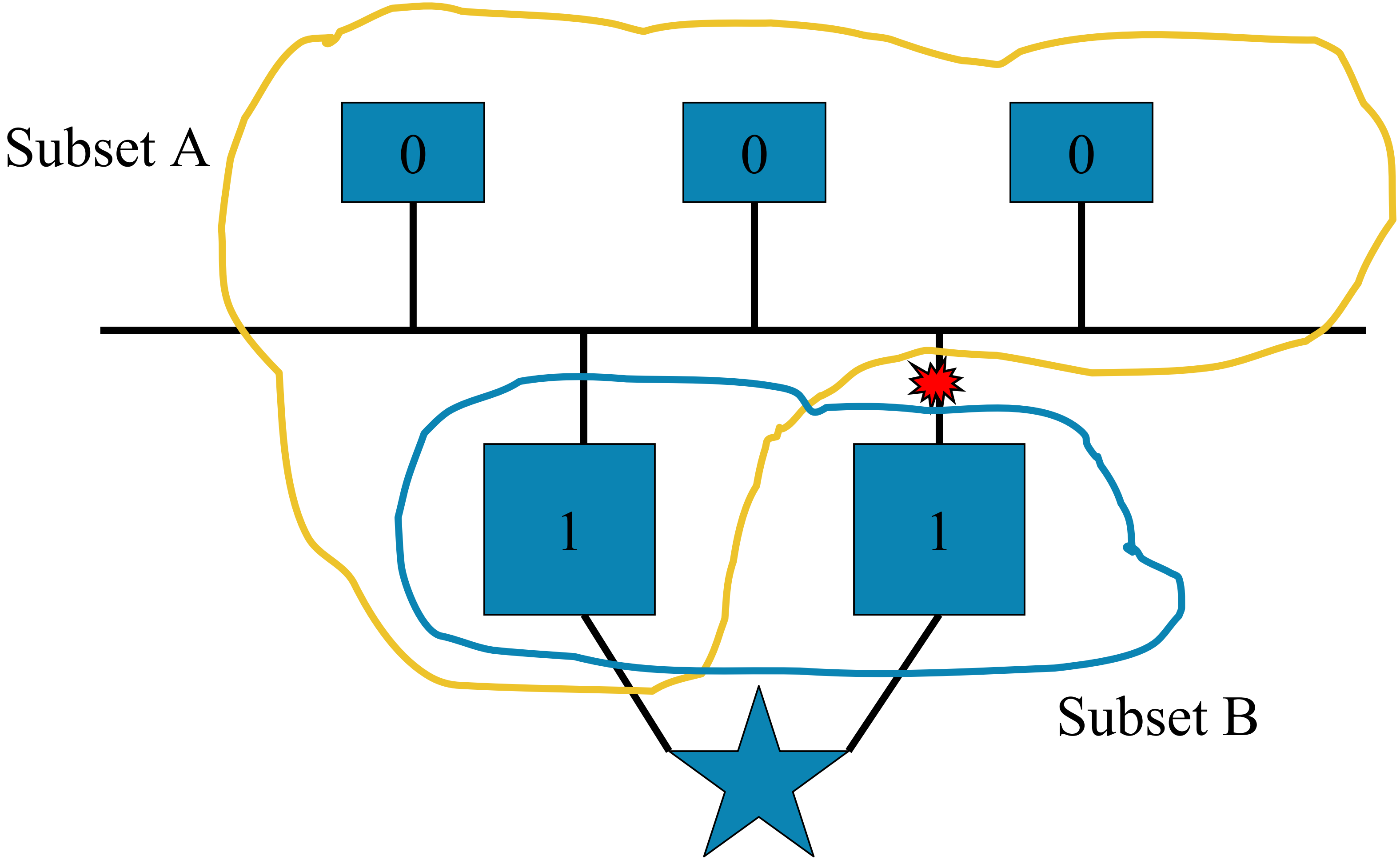


Optimal Sub-cluster Selection: Examples
Boot nodes and satellites

- What happens if the LAN connection on a boot node fails?

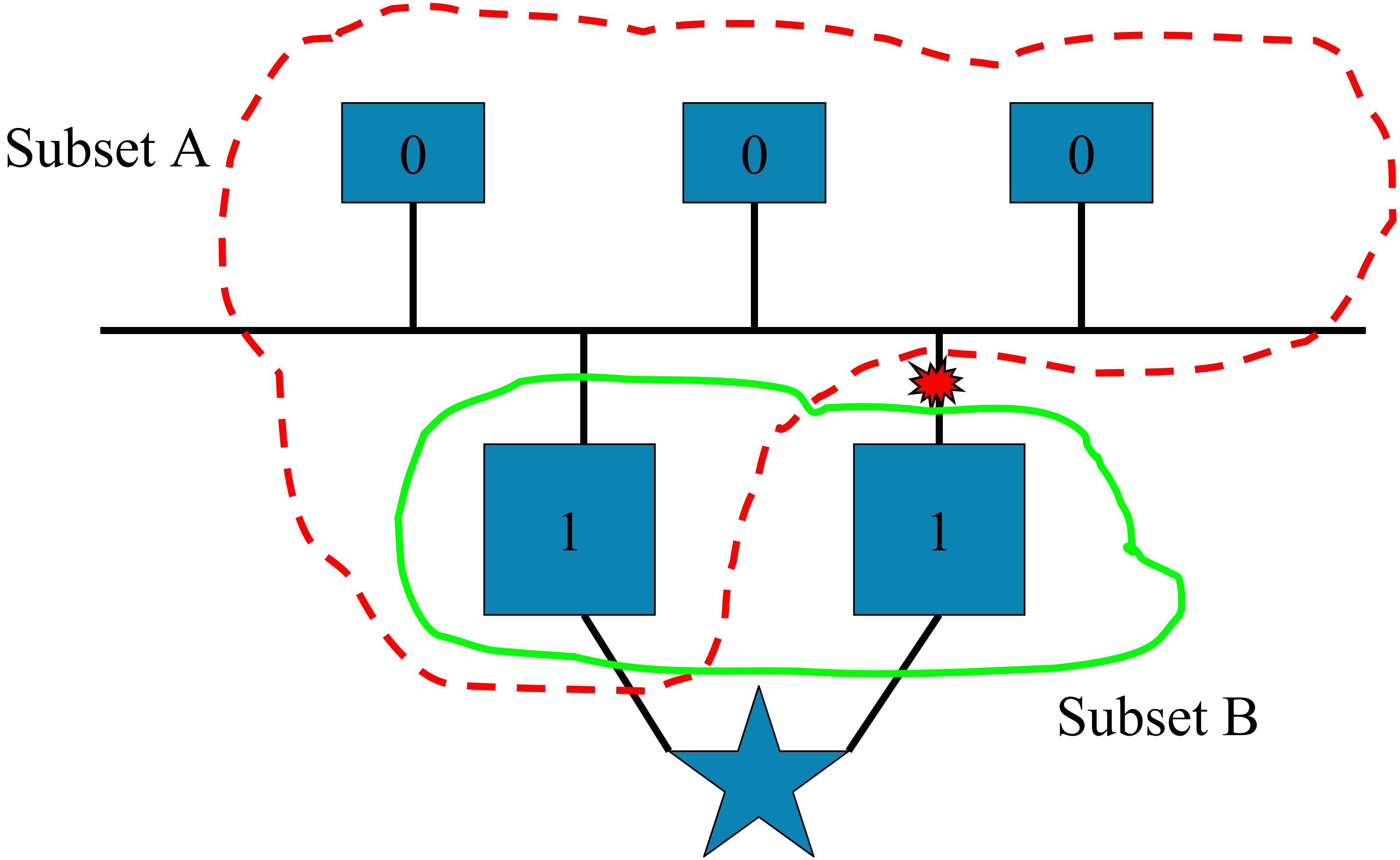


Optimal Sub-cluster Selection: Examples
Boot nodes and satellites



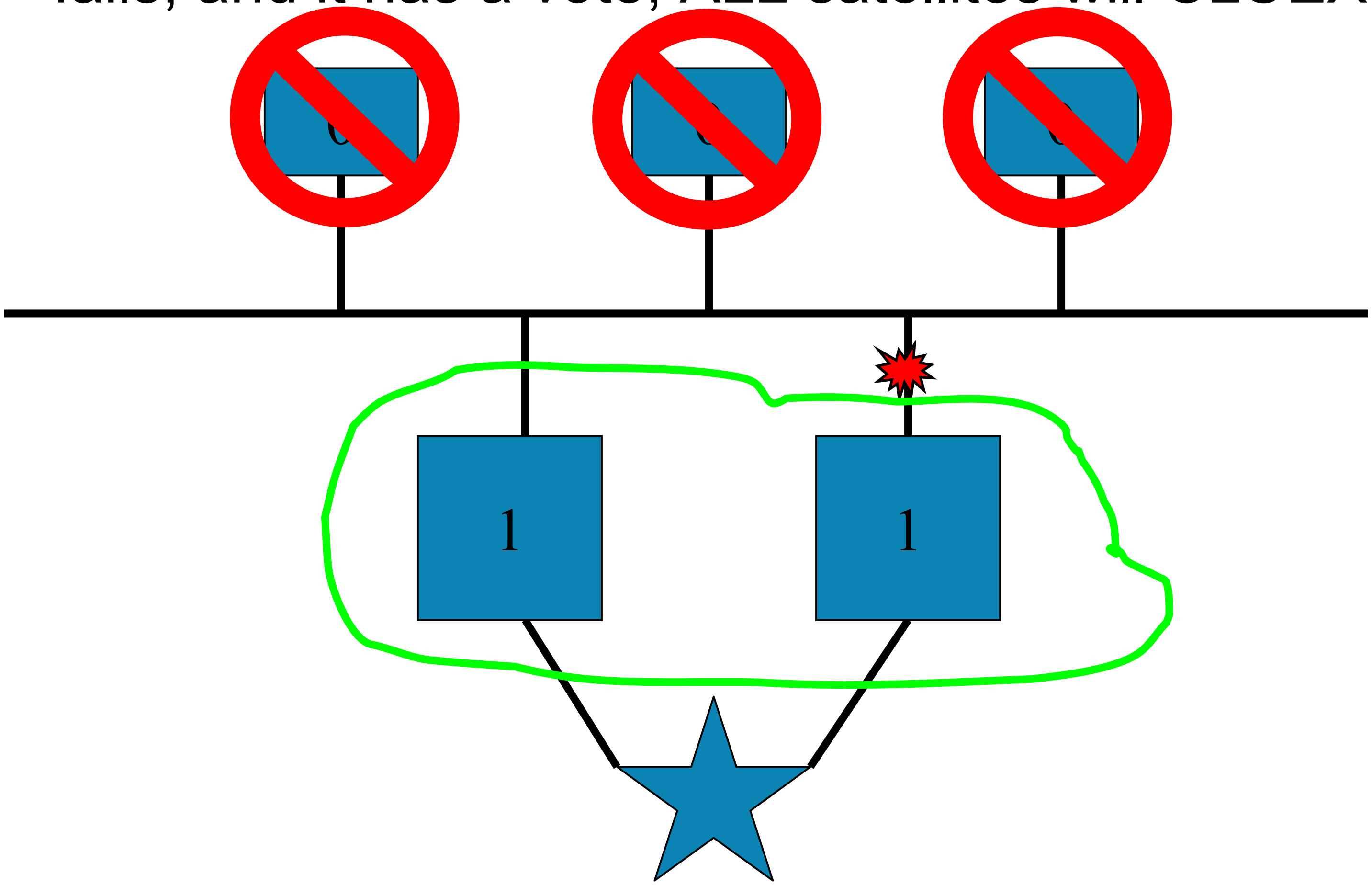
Which subset of nodes does VMS select as the optimal subcluster?

Optimal Sub-cluster Selection: Examples
Boot nodes and satellites



Optimal Sub-cluster Selection: Examples
Boot nodes and satellites

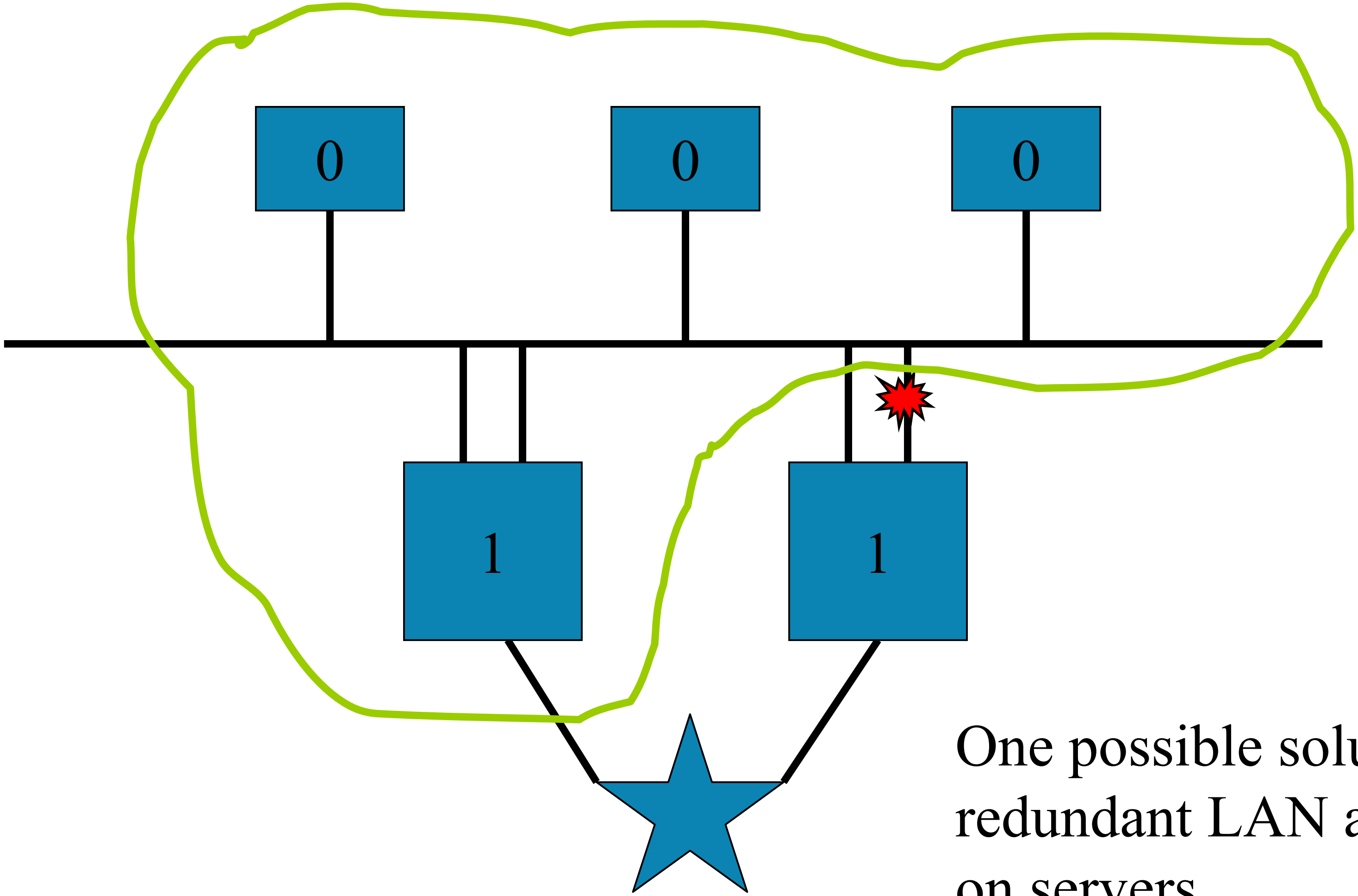
- Conclusion: If the sole LAN adapter on a disk/boot server fails, and it has a vote, ALL satellites will CLUEXIT!



Optimal Sub-cluster Selection: Examples Boot nodes and satellites

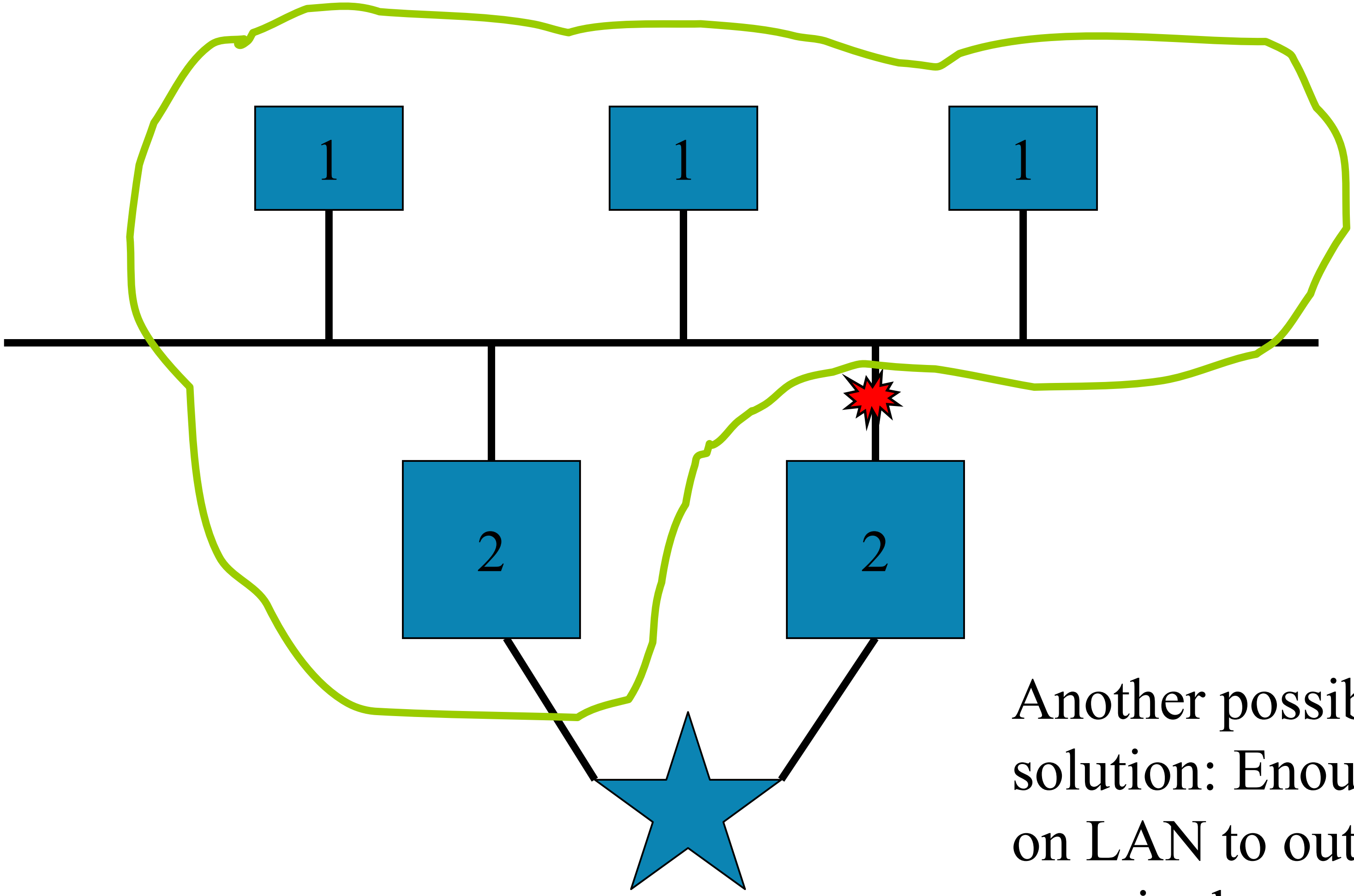
- Advice: give at least as many votes to node(s) on the LAN as any single server has, or configure redundant LAN adapters

Optimal Sub-cluster Selection: Examples
Boot nodes and satellites



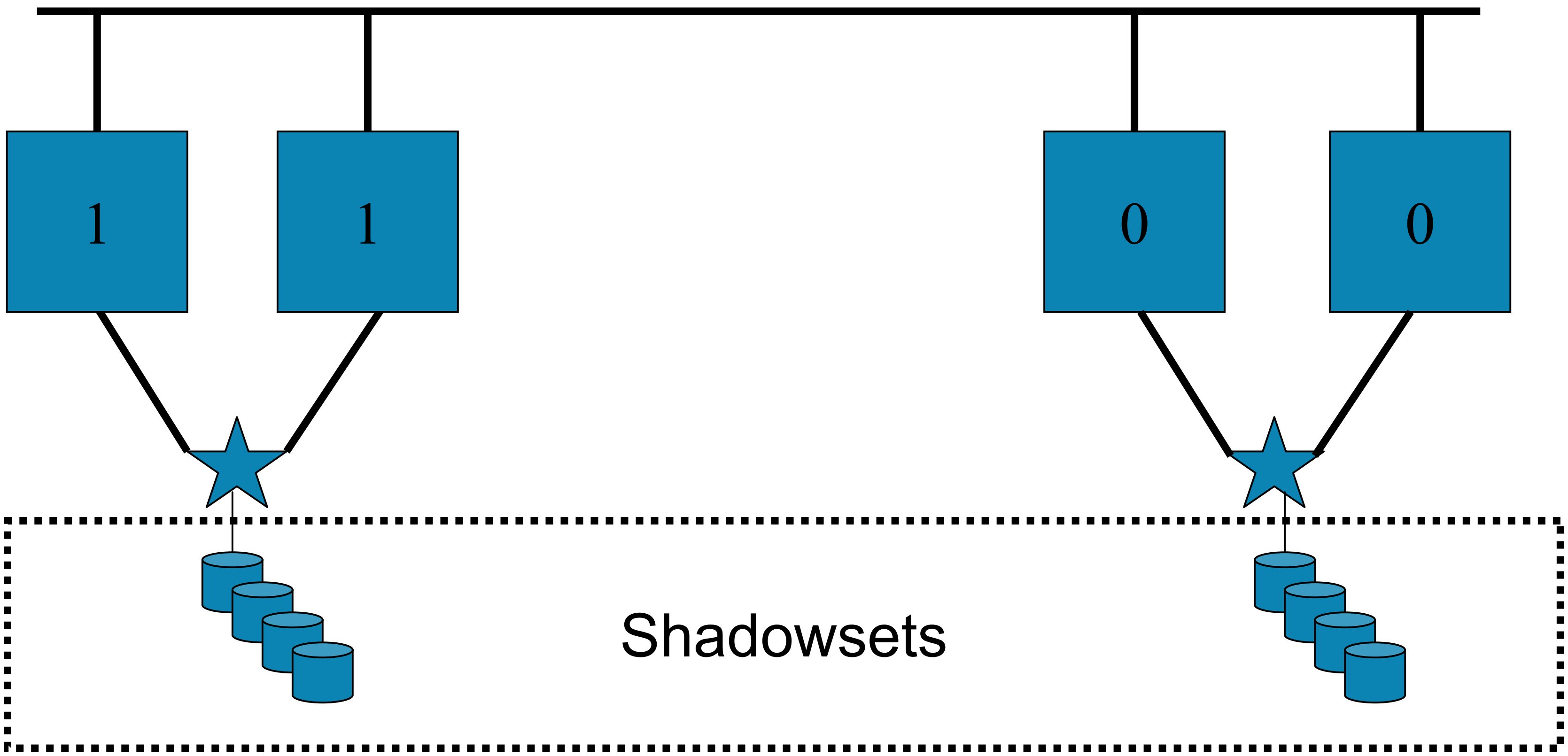
One possible solution: redundant LAN adapters on servers

Optimal Sub-cluster Selection: Examples
Boot nodes and satellites

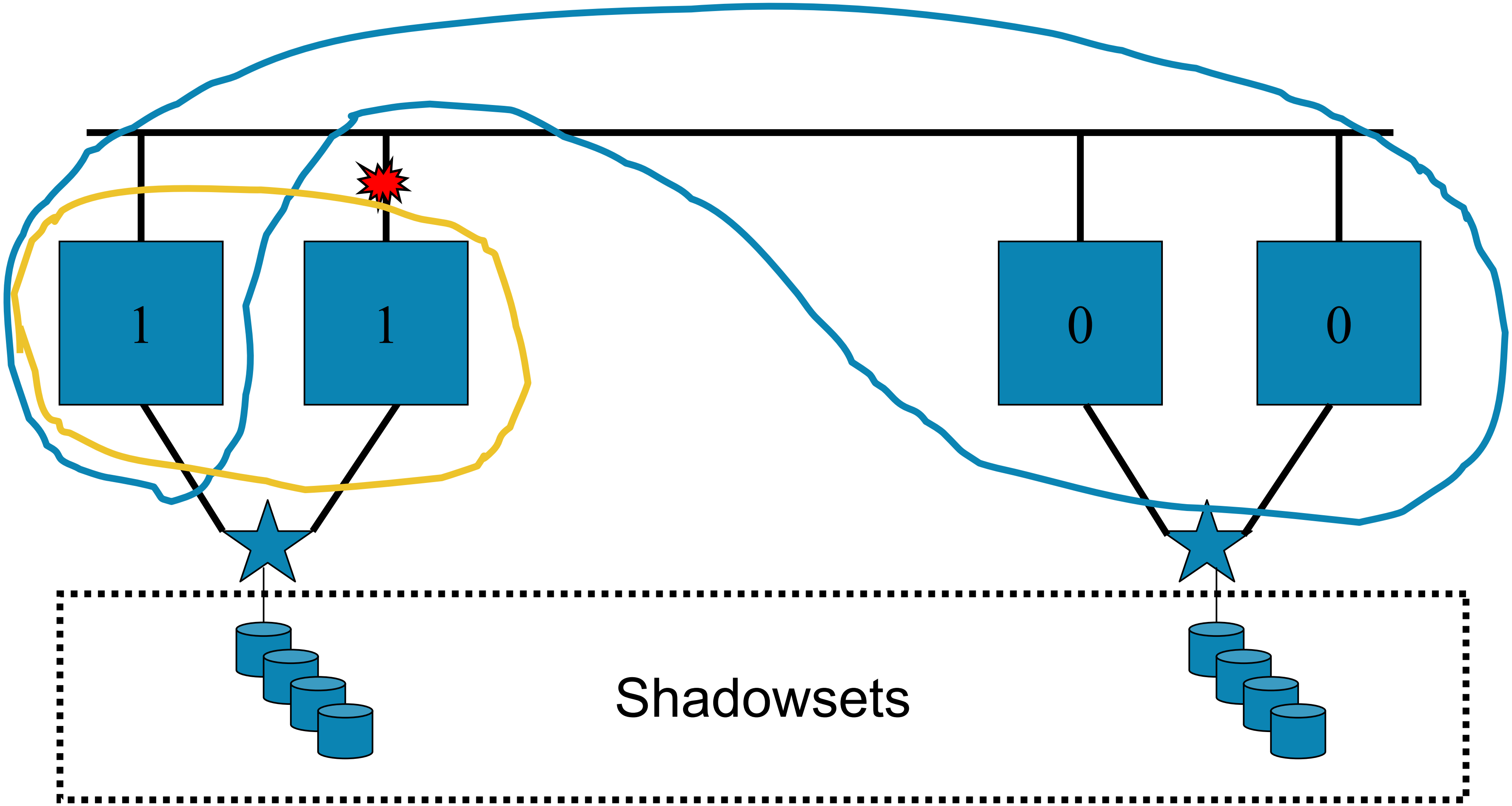


Another possible solution: Enough votes on LAN to outweigh any single server node

Optimal Sub-cluster Selection: Examples
Two-Site Cluster with Unbalanced Votes

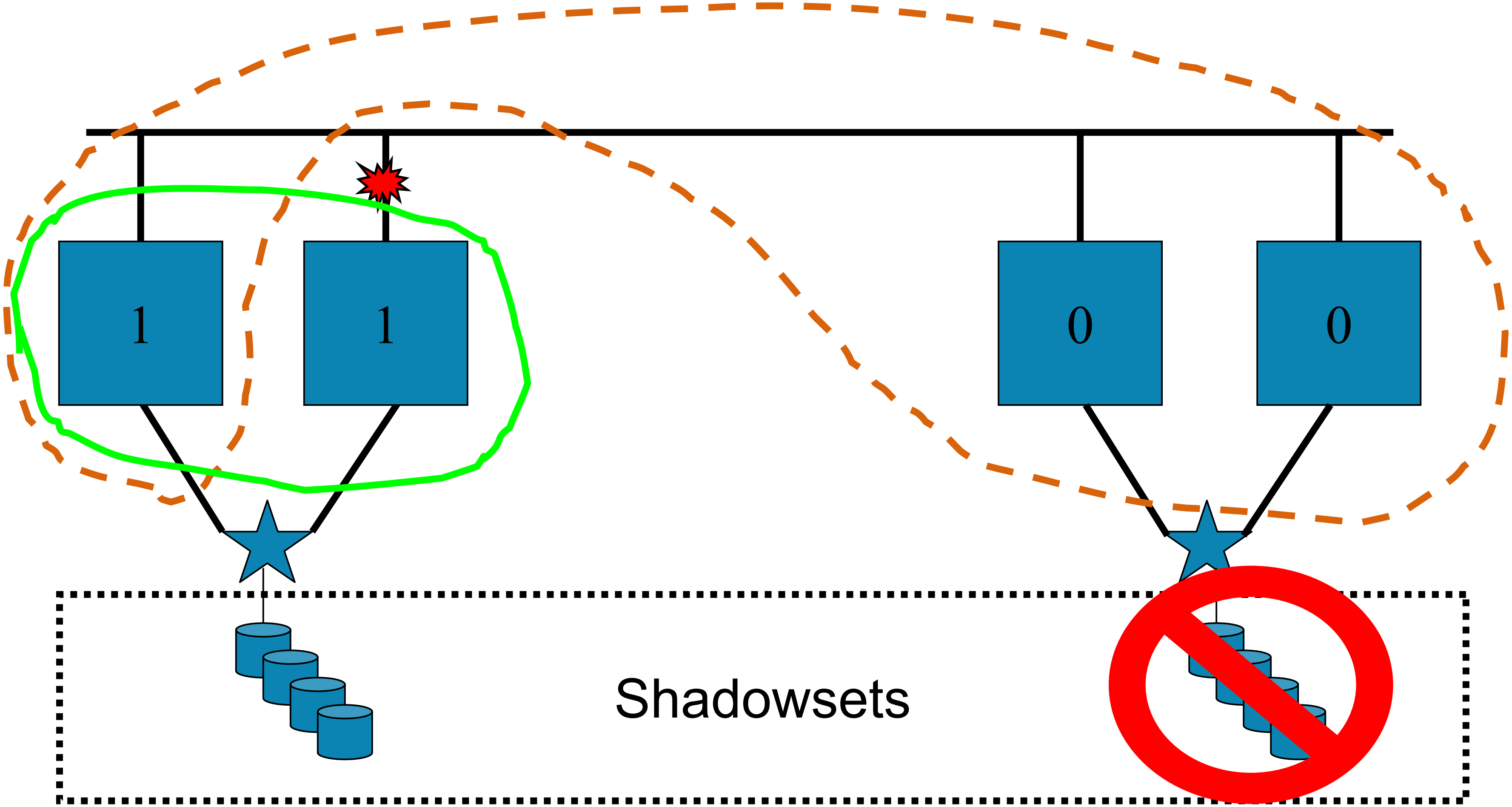


Optimal Sub-cluster Selection: Examples Two-Site Cluster with Unbalanced Votes



Which subset of nodes does VMS select as the optimal subcluster?

Optimal Sub-cluster Selection: Examples
Two-Site Cluster with Unbalanced Votes

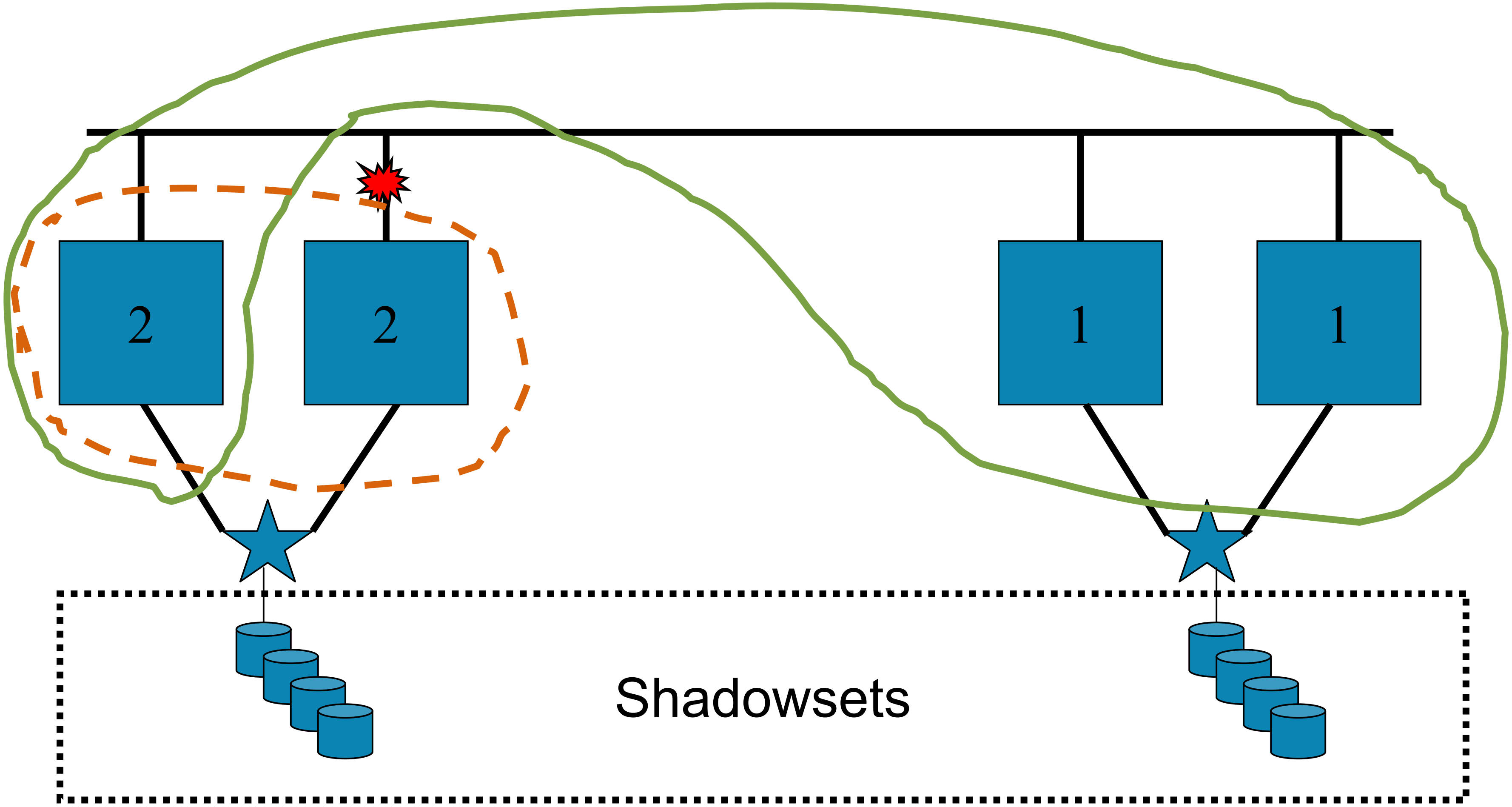


Nodes at this site continue

Nodes at this site CLUEXIT

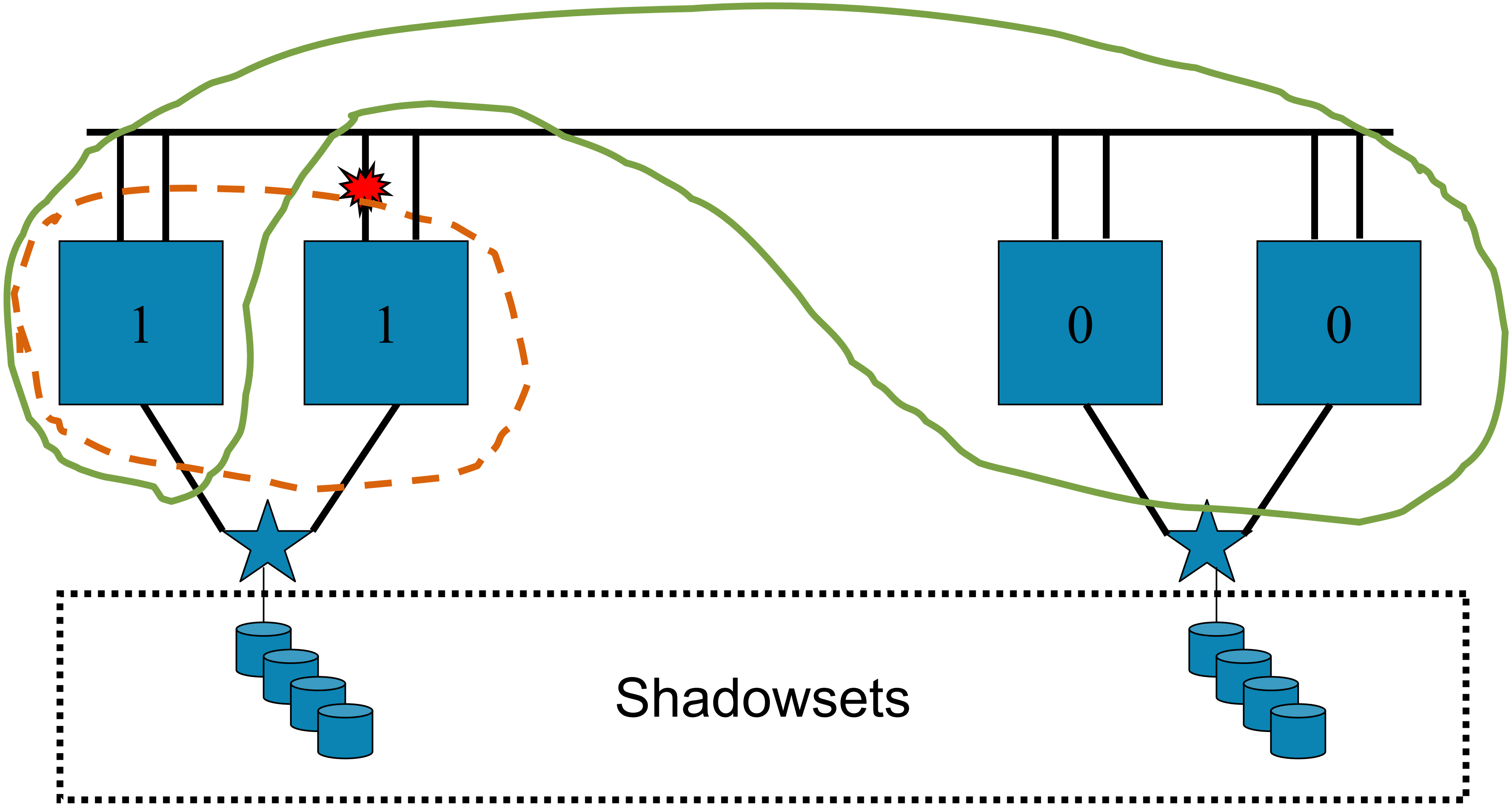
Optimal Sub-cluster Selection: Examples
Two-Site Cluster with Unbalanced Votes

One possible solution: Give more votes to nodes at 2nd site:



Optimal Sub-cluster Selection: Examples Two-Site Cluster with Unbalanced Votes

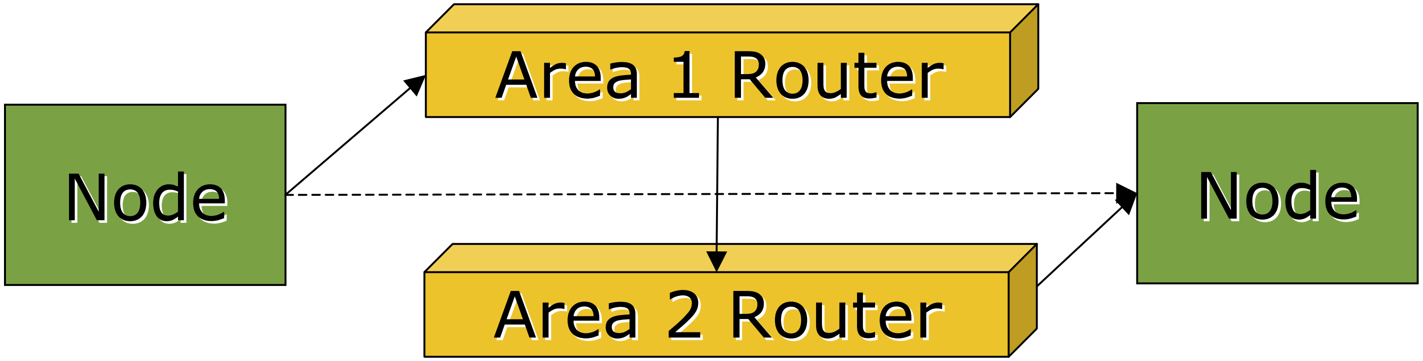
Another possible solution: Redundant LAN adapters:



Network Considerations

- Best network configuration for a disaster-tolerant cluster typically is:
  - All nodes in same DECnet area
  - All nodes in same IP Subnet

  despite being at two separate sites, for these reasons:
  - For ease of cluster alias operations and failover, and
  - To allow direct communications and minimize hops through routers



Shadowing Between Sites

- Shadow copies can generate a high data rate on inter-site links
- Excessive shadow-copy time increases Mean Time To Repair (MTTR) after a site failure or outage
- Acceptable shadow full-copy times and link costs will typically be the major factors in selecting inter-site link(s)

Shadowing Between Sites

- Because:

1) Inter-site latency is typically much greater than intra-site latency, at least if there is any significant distance between sites, and
2) Direct operations are perhaps 0.5 to 1 ms lower in latency than MSCP-served operations, even when the inter-site distance is small,

It is most efficient to direct **Read** operations to the local disks, not remote disks

- (All **Write** operations have to go to all disks in a shadowset, remote as well as local members, of course)

Shadowing Between Sites: Local vs. Remote Reads

- Directing Shadowing Read operations to local disks, in favor of remote disks:
  - Bit 16 (%x10000) in SYSGEN parameter SHADOW_SYS_DISK can be set to force reads to local disks in favor of MSCP-served disks
  - OpenVMS 7.3 (or recent VOLSHAD ECO kits) allow you to tell OpenVMS at which site member disks are located, and the relative “cost” to read a given disk

Shadowing Between Sites: Local vs. Remote Reads on Fibre Channel

- With an inter-site Fibre Channel link, VMS can’t tell which disks are local and which are remote, so we give it some help:
  - To tell VMS what site a given <u>disk</u> is at, do:

    ```
    $ SET DEVICE/SITE=xxx $1$DGAnnn:
    ```

  - To tell VMS what site a given VMS <u>node</u> is at, do:

    ```
    $ SET DEVICE/SITE=xxx DSAnn:
    ```

    for each virtual unit, from each VMS node, specifying the appropriate SITE value for each VMS node.
  - New SYSGEN parameter SHADOW_SITE (introduced at 7.3-1 but still latent) will eventually tell Shadowing what site a node is located at

Shadowing Between Sites: Local vs. Remote Reads on Fibre Channel

- Here's an example of setting /SITE values:

```
$ DEFINE/SYSTEM/EXEC ZKO 1
$ DEFINE/SYSTEM/EXEC LKG 2
$! Please note for this example that:
$! $1$DGA4: is physically located at site ZKO.
$! $1$DGA2: is physically located at site LKG.
$! The MOUNT command is the same at both sites:
$ MOUNT/SYSTEM DSA42/SHAD=($1$DGA4, $1$DGA2) VOLUME_LABEL

$! At the ZKO site ...
$ SET DEVICE/SITE=ZKO DSA42:

$! At the LKG site ...
$ SET DEVICE/SITE=LKG DSA42:

$! At both sites, the following commands would be used to
$! specify at which site the disks are located
$ SET DEVICE/SITE=ZKO $1$DGA4:
$ SET DEVICE/SITE=LKG $1$DGA2:
```

Shadowing Between Sites: Local vs. Remote Reads and Read Cost

- VMS assigns a default “Read Cost” for each shadowset member, with values in this order (lowest cost to highest cost):

1. DECram device
2. Directly-connected device in the same physical location
3. Directly-connected device in a remote location
4. DECram served device
5. All other served devices

VMS adds the Read Cost to the queue length and picks the member with the lowest value to do the read

- Spreads reads in round-robin fashion across members with equal values for Read Cost + queue length

Shadowing Between Sites: Local vs. Remote Reads and Read Cost

- For even greater control, one can override the default "read cost" to a given disk from a given node:

$ SET DEVICE /READ_COST=nnn $1$DGAnnn:

For reads, VMS adds the Read Cost to the queue length and picks the member with the lowest value to do the read

- To return all member disks of a shadowset to their default values after such changes have been done, one can use the command:

$ SET DEVICE /READ_COST = 1 DSAnnn:

Shadowing Between Sites

- Mitigating Impact of Remote Writes
  - Impact of round-trip latency on remote writes:
    - Use write-back cache in controllers to minimize write I/O latency for target disks
  - Remote MSCP-served writes
    - Check SHOW CLUSTER/CONTINUOUS with CR_WAITS and/or AUTOGEN with FEEDBACK to ensure MSCP_CREDITS is high enough to avoid SCS credit waits
    - Use MONITOR MSCP, SHOW DEVICE/SERVED, and/or AUTOGEN with FEEDBACK to ensure MSCP_BUFFER is high enough to avoid segmenting transfers

Shadowing Between Sites: Speeding Shadow Copies

- Host-Based Volume Shadowing Full-Copy algorithm is non-intuitive:

1. Read from source disk
2. Do Compare operation with target disk
3. If data is different, write to target disk, then go to Step 1.

Speeding Shadow Copies

- Shadow_Server process does copy I/Os
- Does one 127-block segment at a time, from the beginning of the disk to the end, with no double-buffering or other speed-up tricks
- Odd algorithm is to ensure correct results in cases like:
  - With application writes occurring in parallel with the copy thread
    - application writes during a Full Copy go to the set of source disks first, then to the set of target disk(s)
  - On system disks with a VMS node booting or dumping while shadow copy is going on

Speeding Shadow Copies

- Implications:
  - Shadow copy completes fastest if data is identical beforehand
    - Fortunately, this is the most-common case – re-adding a shadow member into shadowset again after it was a member before

Speeding Shadow Copies

- If data is very different, empirical tests have shown that it is faster to:
  1. Do BACKUP/PHYSICAL from source shadowset to /FOREIGN-mounted target disk,
  2. Then do shadow copy afterward

  than to simply initiate the shadow copy with differing data.
    - But be sure to clobber SCB on target disk with an $INITIALIZE (or $MOUNT/OVERRIDE=SHADOW) command before adding new member to shadowset, or $MOUNT gets fooled by the identical SCBs and adds disk as a merge member instead of a full-copy target (this is fixed in 7.3-2)

Speeding Shadow Copies

- For even more speed-up, perform the BACKUP/PHYSICAL on a node on the target side
  - Because remote (MSCP-served) writes take a minimum of 2 round trips, whereas remote reads take a minimum of only 1 round trip

Speeding Shadow Copies

- Doing shadow copy work from a node at target site, not source site, is also most efficient, for the same reason. It also uses less inter-site bandwidth (only Reads go across the link).
  - But with limited inter-site link bandwidth, it may help to do some of the copies in the reverse direction and thus utilize the “reverse” direction bandwidth, too

Speeding Shadow Copies

- To control which node does shadow copy:
  - 1) Set dynamic SYSGEN parameter SHADOW_MAX_COPY to a large positive value on desired (e.g. target-site) node(s)
  - 2) Set SHADOW_MAX_COPY to 0 on all other nodes
  - 3) Do $MOUNT to add member to shadowset; wait briefly for shadow copy operation to start
  - 4) Reset SHADOW_MAX_COPY parameter to original values on all nodes

Speeding Shadow Copies

- Determining which node is performing a shadow copy:
  - Using SDA:
    - From each cluster node, do:

      SDA> SET PROCESS SHADOW_SERVER

      SDA> SHOW PROCESS/CHANNELS

      and look for Busy channel to disk of interest
    - Or look for node holding a lock in Exclusive mode on a resource of the form “$DSAnnnn$_COPIER”

Speeding Shadow Copies

- Because of the sequential, non-pipelined nature of the shadow-copy algorithm, to speed shadow copies:
  - Rather than forming controller-based stripesets and shadowing those across sites, shadow individual disks in parallel, and combine them into RAID-0 arrays with host-based RAID software
  - Dividing a disk up into 4 partitions at the controller level, and shadow-copying all 4 in parallel, takes only 40% of the time to shadow-copy the entire disk as a whole

Speeding Shadow Copies: Mini-Copy Operations

- If one knows ahead of time that one shadowset member will temporarily be removed, one can set things up to allow a Mini-Copy instead of a Full-Copy when the disk is returned to the shadowset

Speeding Shadow Copies

- Read-ahead caching on new controllers can speed shadow copies.

- New command $SET DEVICE/COPY_SOURCE can be used to specify the source disk for a Full-Copy operation (e.g. to copy from a 2nd member at the same site instead of from a member at the remote site)

Avoiding Shadow Copies

- New $INITIALIZE/SHADOW command allows a shadowset to be formed with multiple members from the start, instead of always having to do a Full-Copy from the initial member
  - But consider including the /ERASE qualifier or the first Full-Merge operation will be slow as it fixes up non-file-system areas.

Speeding Merge Operations

- Logical names SHAD$MERGE_DELAY_FACTOR and SHAD$MERGE_DELAY_FACTOR_DSAnnn allow control of merge thread I/O rates in the face of application I/Os
  - Logical names checked every 1,000 I/Os
  - Default is 200; lower values slow merge thread I/Os; higher values increase merge thread I/Os
  - Default merge thread back-off threshold is now too low for modern controllers, so Shadowing is too cautious; use logical names to speed merges
- MSCP controllers allow Mini-Merge operations
- Project is underway with goal of host-based mini-merges

Data Protection Scenarios

- Protection of the data is obviously extremely important in a disaster-tolerant cluster
- We'll look at one scenario that has happened in real life and resulted in data loss:
  - "Wrong-way shadow copy"
- We'll also look at two obscure but potentially dangerous scenarios that theoretically could occur and would result in data loss:
  - "Creeping Doom"
  - "Rolling Disaster"

Protecting Shadowed Data

- Shadowing keeps a “Generation Number” in the SCB on shadow member disks
- Shadowing “Bumps” the Generation number at the time of various shadowset events, such as mounting, or membership changes

Protecting Shadowed Data

- Generation number is designed to constantly increase over time, never decrease
- Implementation is based on OpenVMS timestamp value, and during a “Bump” operation it is increased to the current time value (or, if it’s already a future time for some reason, such as time skew among cluster member clocks, then it’s simply incremented). The new value is stored on all shadowset members at the time of the Bump.

Protecting Shadowed Data

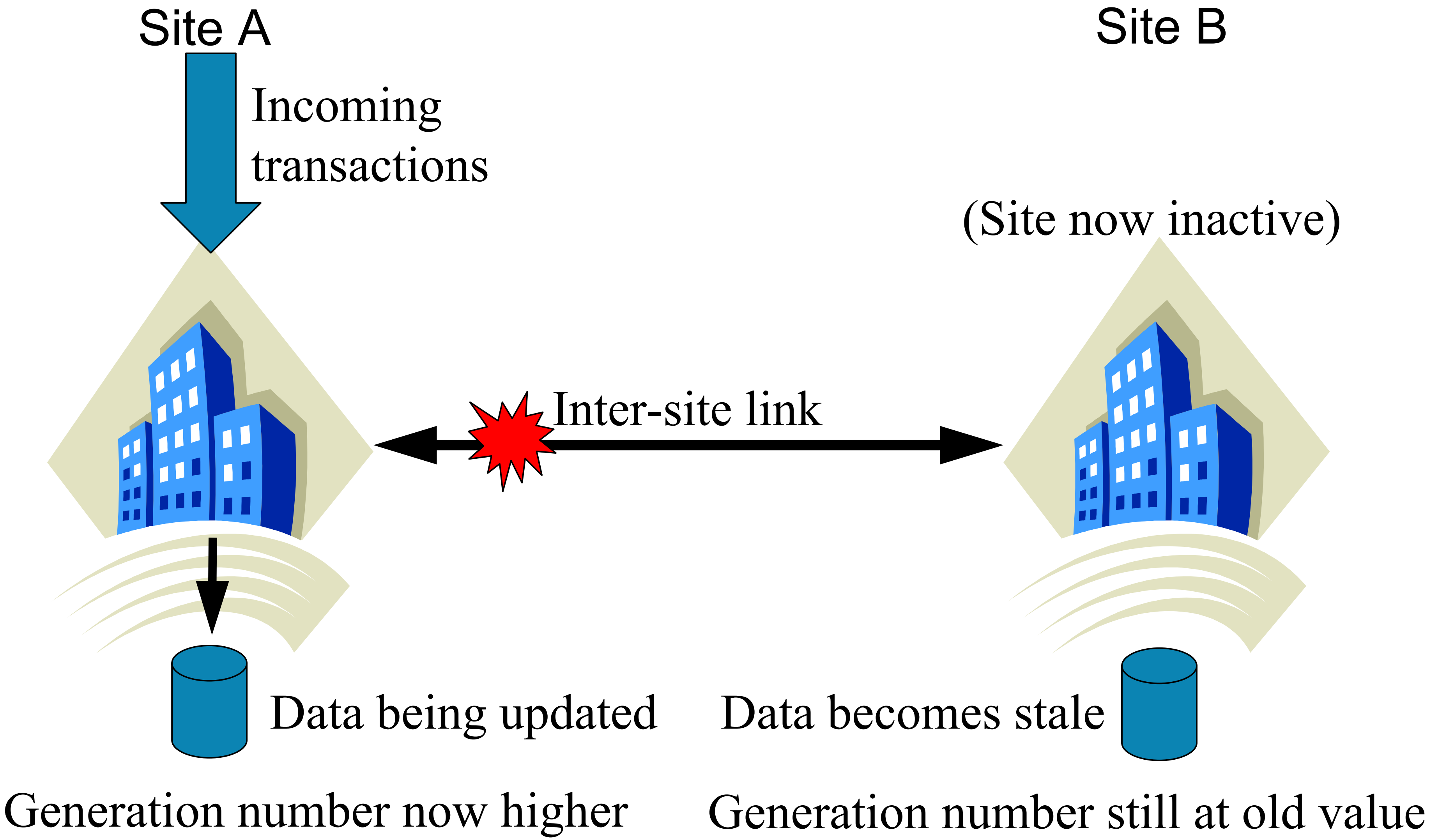
- Generation number in SCB on removed members will thus gradually fall farther and farther behind that of current members
- In comparing two disks, a later generation number should always be on the more up-to-date member, under normal circumstances

"Wrong-Way Shadow Copy" Scenario

- Shadow-copy "nightmare scenario":
  - Shadow copy in "wrong" direction copies old data over new

- Real-life example:
  - Inter-site link failure occurs
  - Due to unbalanced votes, Site A continues to run
    - Shadowing increases generation numbers on Site A disks after removing Site B members from shadowset

Wrong-Way Shadow Copy



Site A

Site B

Incoming transactions

(Site now inactive)

Inter-site link

Data being updated

Data becomes stale

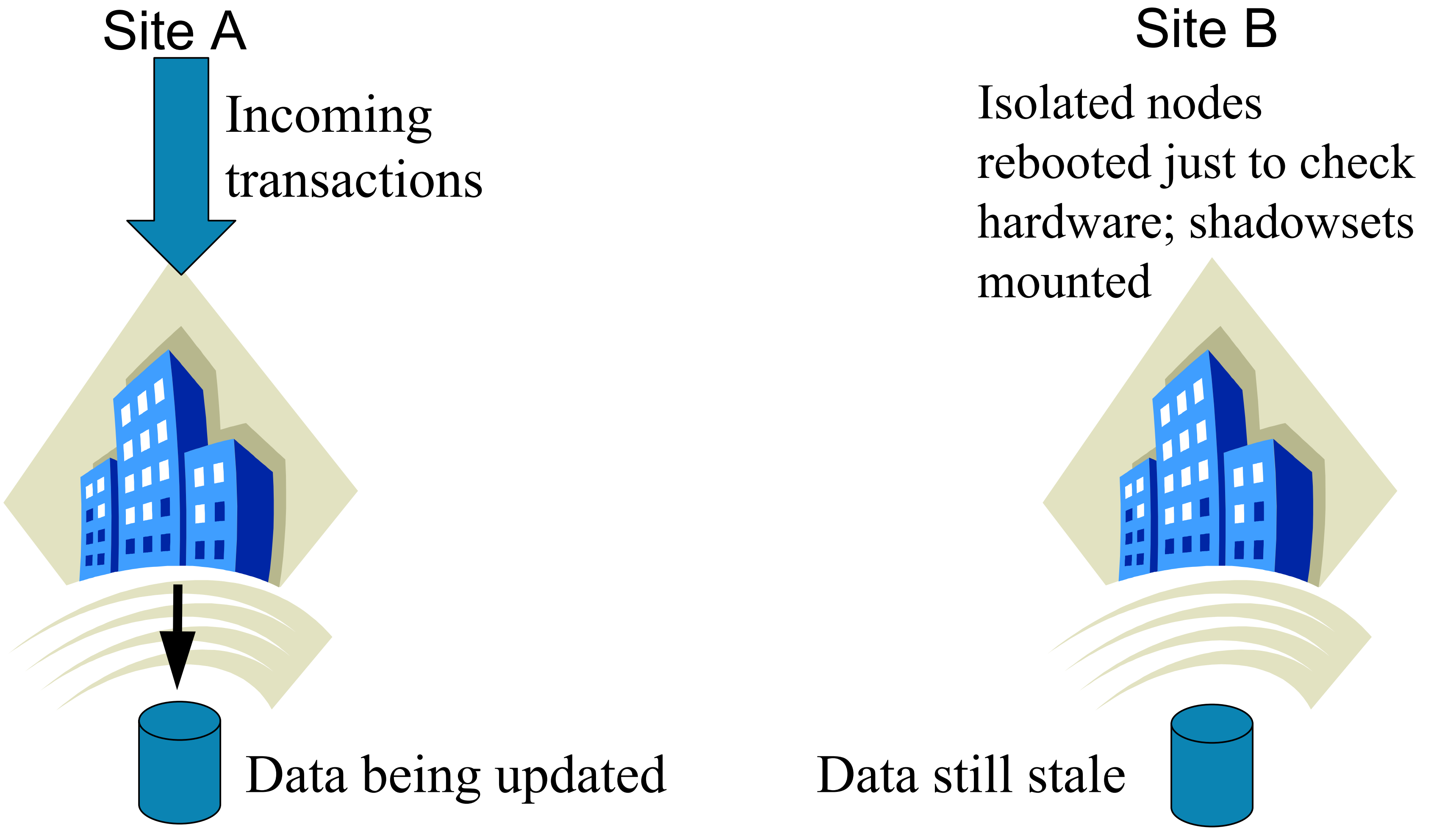
Generation number now higher

Generation number still at old value

Wrong-Way Shadow Copy

- Site B is brought up briefly by itself for whatever reason (e.g. “to test the hardware”)
  - Shadowing can’t see Site A disks. Shadowsets mount with Site B disks only. Shadowing bumps generation numbers on Site B disks. Generation number is now greater than on Site A disks.

Wrong-Way Shadow Copy



Site A

Incoming transactions

Data being updated

Site B

Isolated nodes rebooted just to check hardware; shadowsets mounted

Data still stale

Generation number unaffected

Generation number now highest

Wrong-Way Shadow Copy

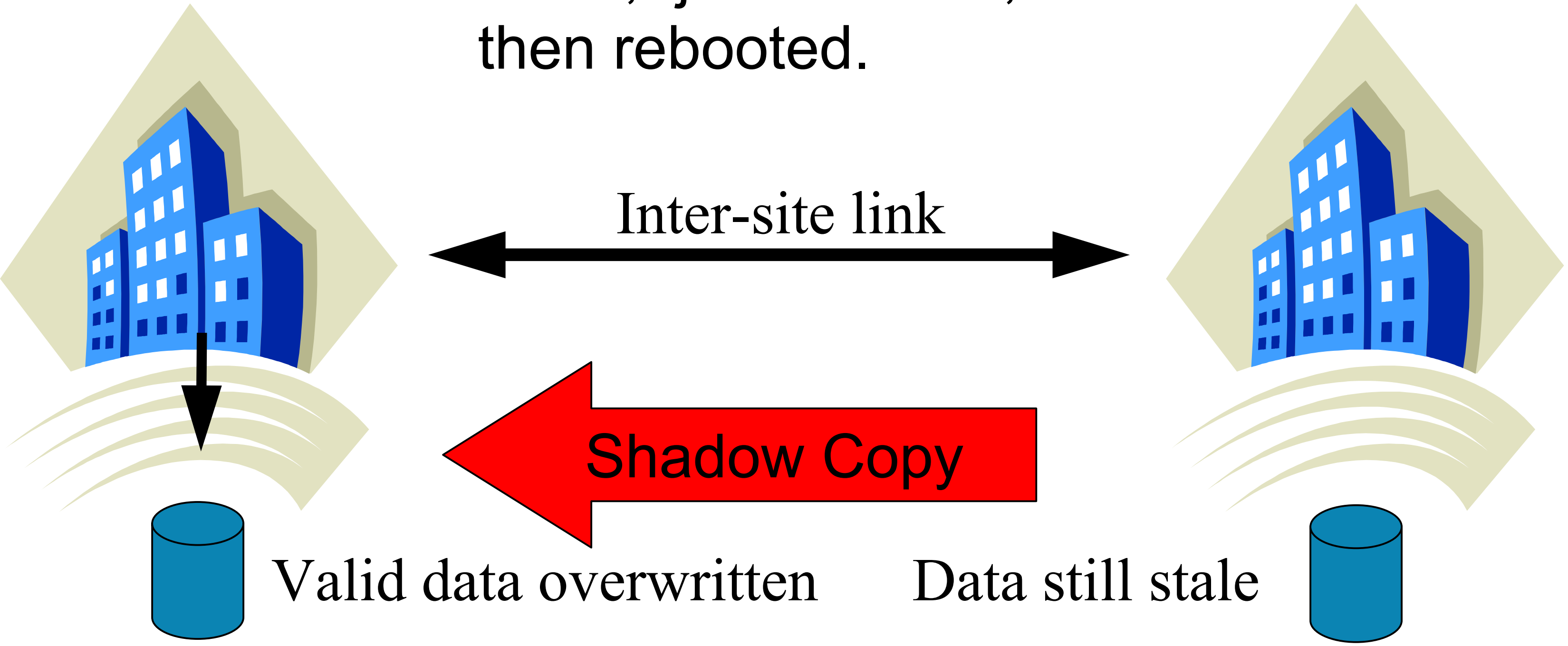
- Link gets fixed. Both sites are taken down and rebooted at once.
- Shadowing thinks Site B disks are more current, and copies them over Site A’s. Result: Data Loss.

Wrong-Way Shadow Copy

Site A

Site B

Before link is restored, entire cluster is taken down, “just in case”, then rebooted.



Inter-site link

Shadow Copy

Valid data overwritten

Data still stale

Generation number is highest

Protecting Shadowed Data

- If shadowing can't "see" a later disk's SCB (i.e. because the site or link to the site is down), it may use an older member and then update the Generation number to a current timestamp value
- New /POLICY=REQUIRE_MEMBERS qualifier on MOUNT command prevents a mount unless all of the listed members are present for Shadowing to compare Generation numbers on
- New /POLICY=VERIFY_LABEL qualifier on MOUNT means volume label on member must be SCRATCH_DISK, or it won't be added to the shadowset as a full-copy target

Avoiding Untimely/Unwanted Shadow Copies

- After a site failure or inter-site link failure, rebooting the downed site after repairs can be disruptive to the surviving site
- Many DT Cluster sites prevent systems from automatically rebooting without manual intervention
  - Easiest way to accomplish this is to set console boot flags for conversational boot

Avoiding Untimely/Unwanted Shadow Copies

- If MOUNT commands are in SYSTARTUP_VMS.COM, shadow copies may start as soon as the first node at the downed site reboots
- Recommendation is to ***not*** mount shadowsets automatically at startup; manually initiate shadow copies of application data disks at an opportune time

Avoiding Untimely/Unwanted Shadow Copies

- In bringing a cluster with cross-site shadowsets completely down and back up, you need to preserve both shadowset members to avoid a full copy operation
- Cross-site shadowsets must be dismounted while both members are still accessible
- This implies keeping MSCP-serving OpenVMS systems up at each site until the shadowsets are dismounted
  - Easy way is to use the CLUSTER_SHUTDOWN option on SHUTDOWN.COM

Avoiding Untimely/Unwanted Shadow Copies

- In bringing a cluster with cross-site shadowsets back up, you need to ensure both shadowset members are accessible at mount time, to avoid removing a member and thus needing to do a shadow full-copy afterward
- If MOUNT commands are in SYSTARTUP_VMS.COM, the first node up at the first site up will form 1-member shadow sets and drop the other site's shadow members

Avoiding Untimely/Unwanted Shadow Copies

- Recommendation is to not mount cross-site shadowsets automatically in startup; wait until at least a couple of systems are up at each site, then manually initiate cross-site shadowset mounts
- Since MSCP-serving is enabled before a node joins a cluster, booting systems at both sites simultaneously works *most* of the time

Avoiding Untimely/Unwanted Shadow Copies

- New Shadowing capabilities help in this area:

```
$ MOUNT DSAnnn: label
```

without any other qualifiers will mount a shadowset on an additional node using the existing membership, without the chance of any shadow copies being initiated.

  - This allows you to start the application at the second site and run from the first site’s disks, and do the shadow copies later

Avoiding Untimely/Unwanted Shadow Copies

- DCL code can be written to wait for both shadowset members before MOUNTing, using the /POLICY=REQUIRE_MEMBERS and /NOCOPY qualifiers as safeguards against undesired shadow copies
- The /VERIFY_LABEL qualifier to MOUNT prevents a shadow copy from starting to a disk unless its label is “SCRATCH_DISK”:
  - This means that before a member disk can be a target of a full-copy operation, it must be MOUNTed with the /OVERRIDE=SHADOW qualifier and a

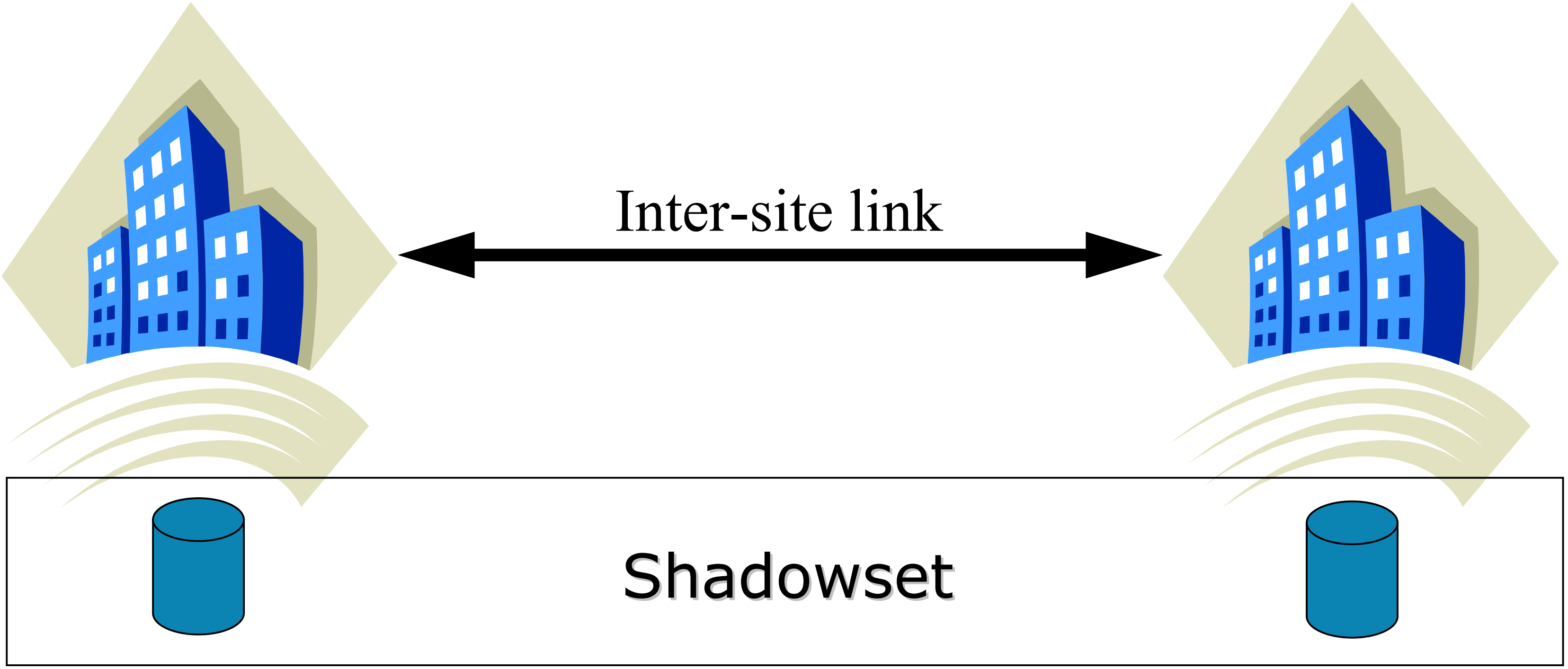
    ```
    $ SET VOLUME/LABEL=SCRATCH_DISK
    ```

    command executed to change the label

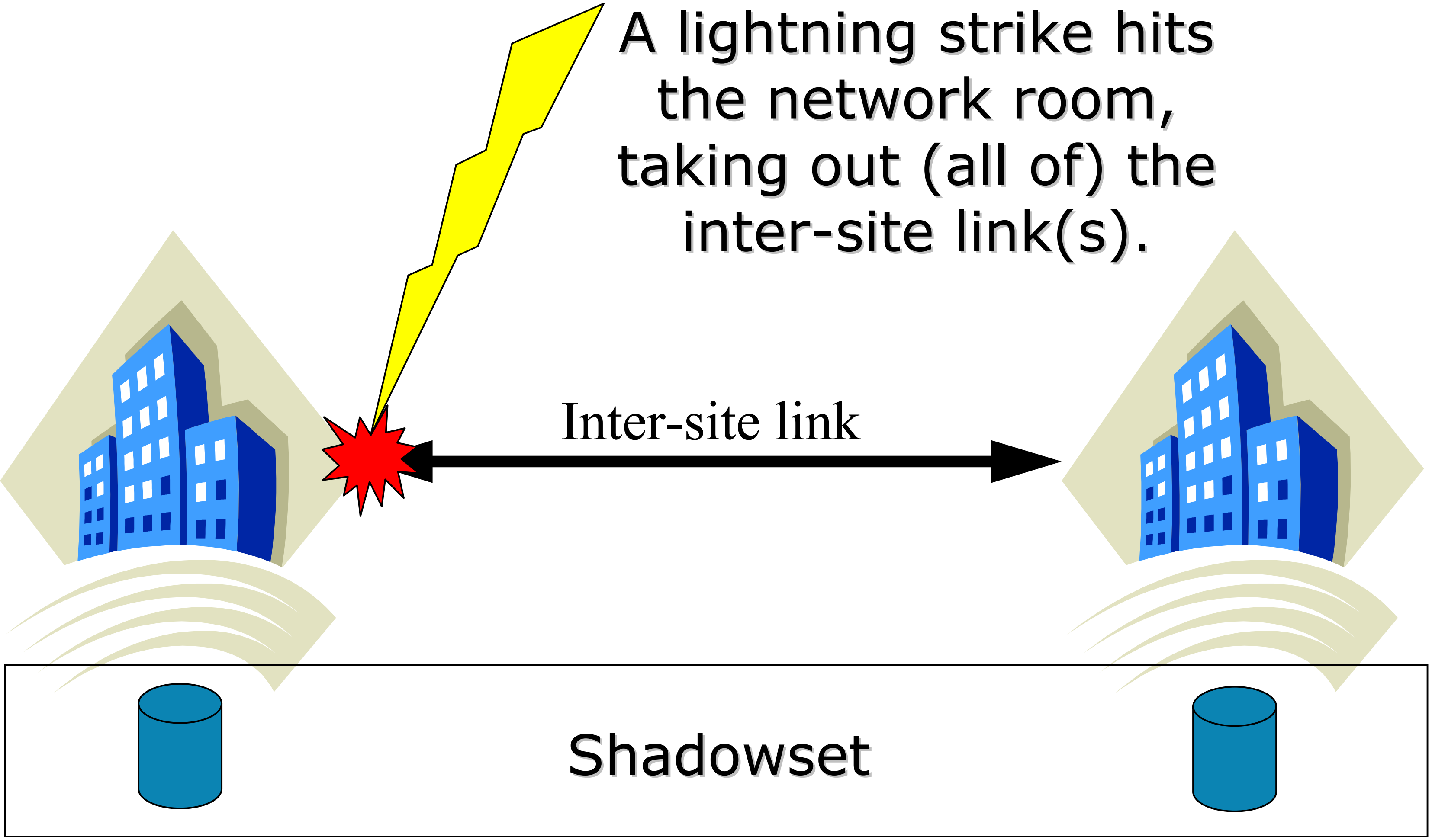
Avoiding Untimely/Unwanted Shadow Copies

- One of the USER* SYSGEN parameters (e.g. USERD1) may be used to as a flag to indicate to startup procedures the desired action:
  - Mount both members (normal case; both sites OK)
  - Mount only local member (other site is down)
  - Mount only remote member (other site survived; this site is re-entering the cluster, but we're deferring shadow copies until later)

“Creeping Doom” Scenario



"Creeping Doom" Scenario

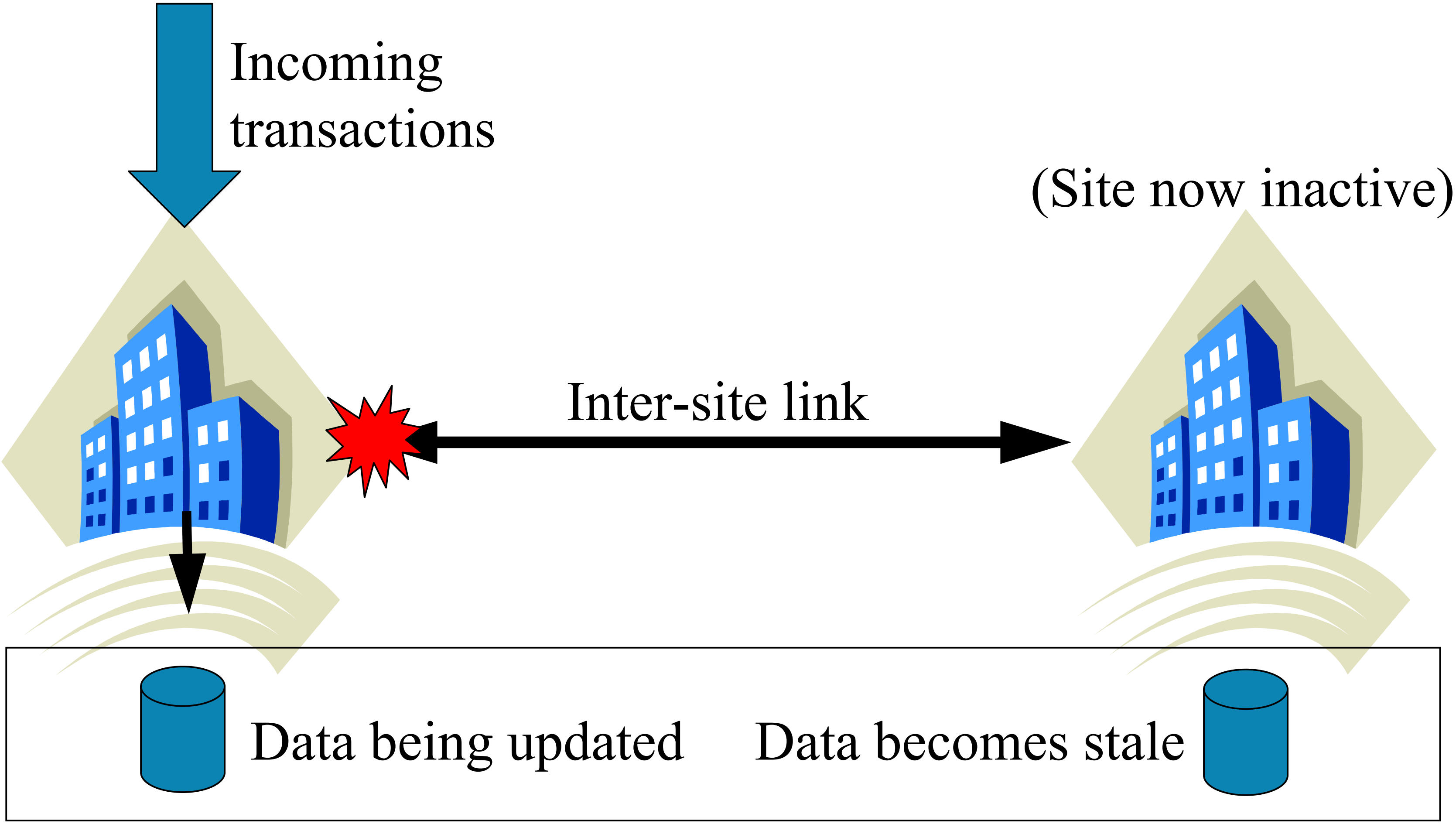


A lightning strike hits the network room, taking out (all of) the inter-site link(s).

“Creeping Doom” Scenario

- First symptom is failure of link(s) between two sites
  - Forces choice of which datacenter of the two will continue
- Transactions then continue to be processed at chosen datacenter, updating the data

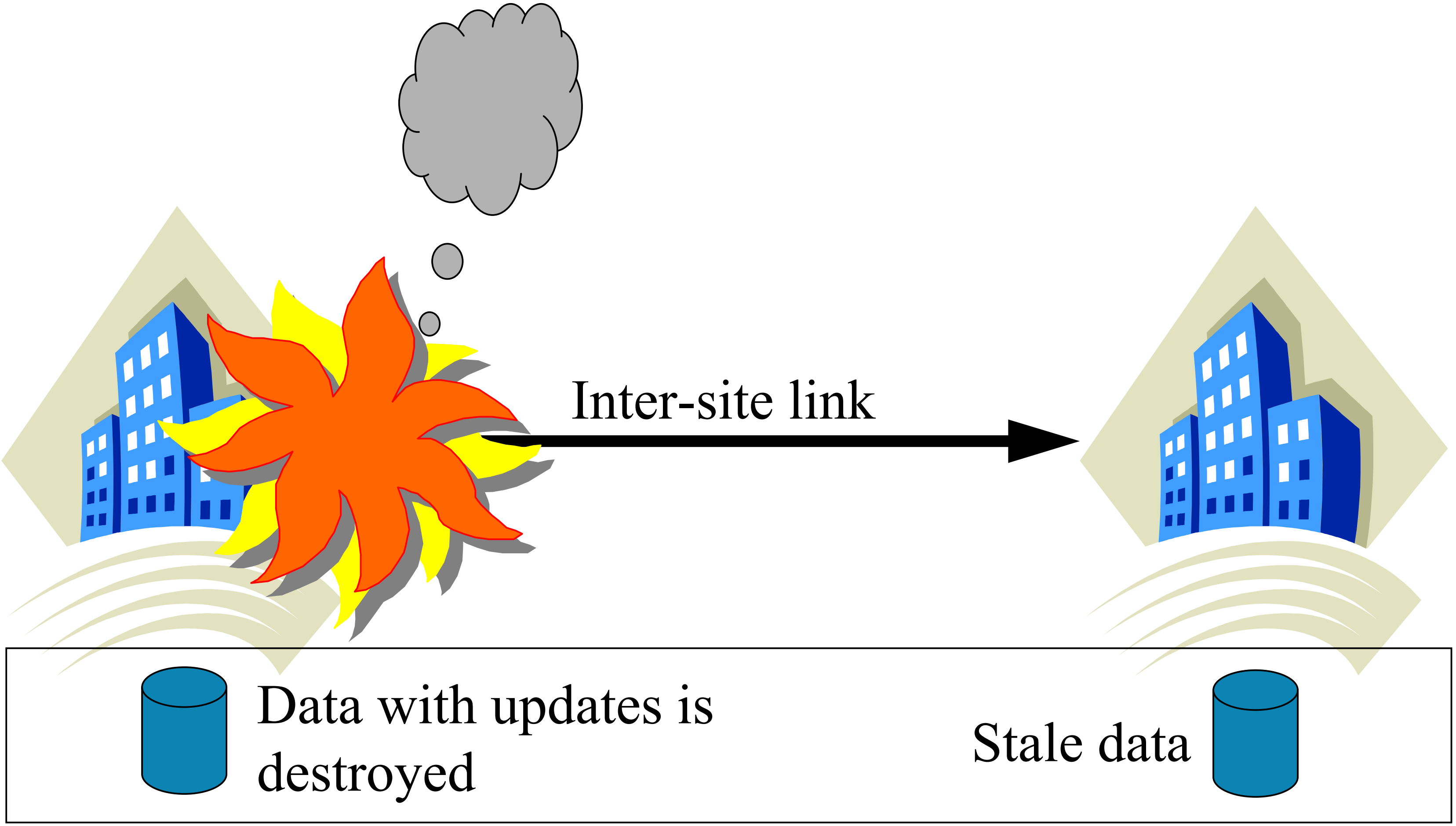
“Creeping Doom” Scenario



“Creeping Doom” Scenario

- In this scenario, the same failure which caused the inter-site link(s) to go down expands to destroy the entire datacenter

“Creeping Doom” Scenario



Inter-site link

Data with updates is destroyed

Stale data

“Creeping Doom” Scenario

- Transactions processed after “wrong” datacenter choice are thus lost
  - Commitments implied to customers by those transactions are also lost

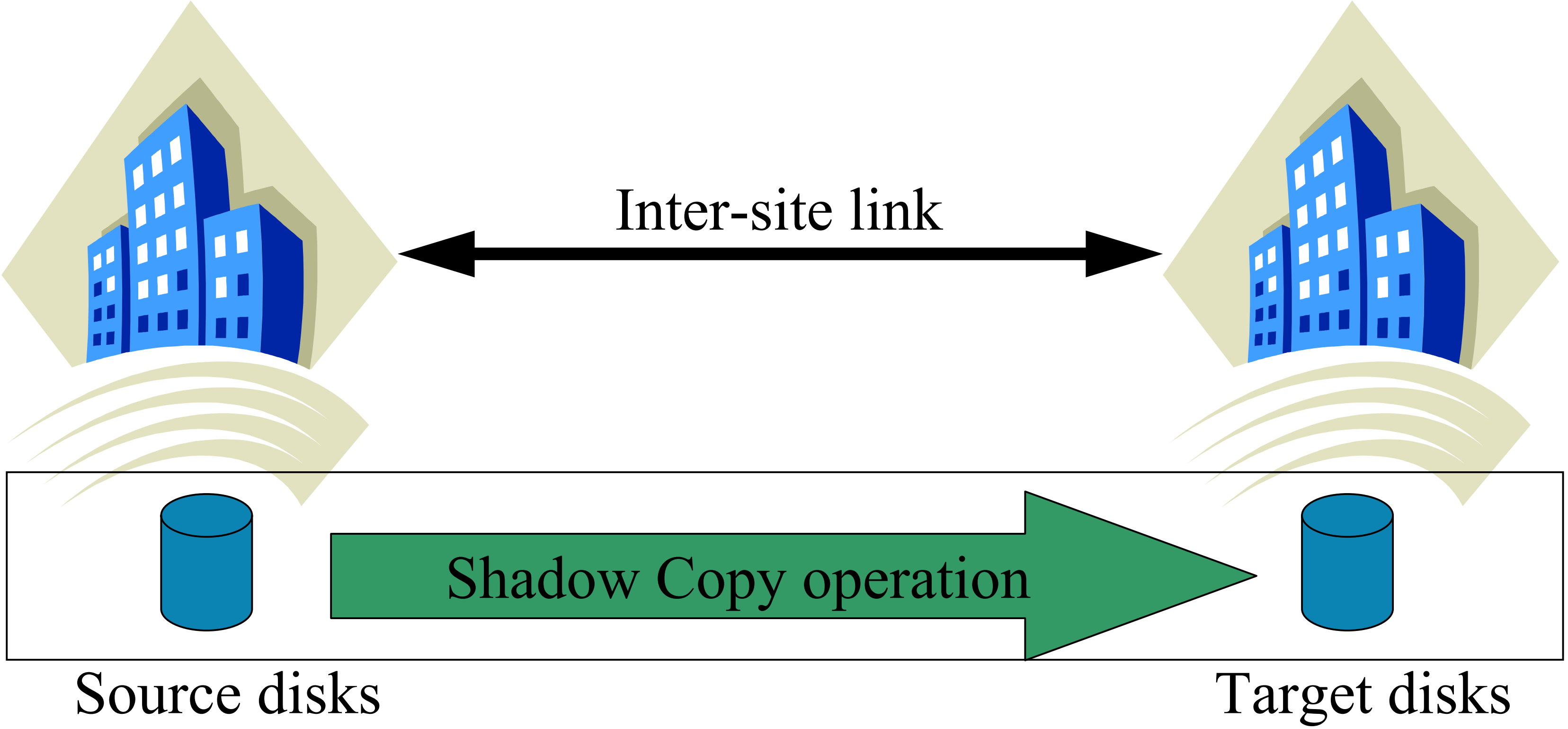
“Creeping Doom” Scenario

- Techniques for avoiding data loss due to “Creeping Doom”:
  - Tie-breaker at 3rd site helps in many (but not all) cases
  - 3rd copy of data at 3rd site

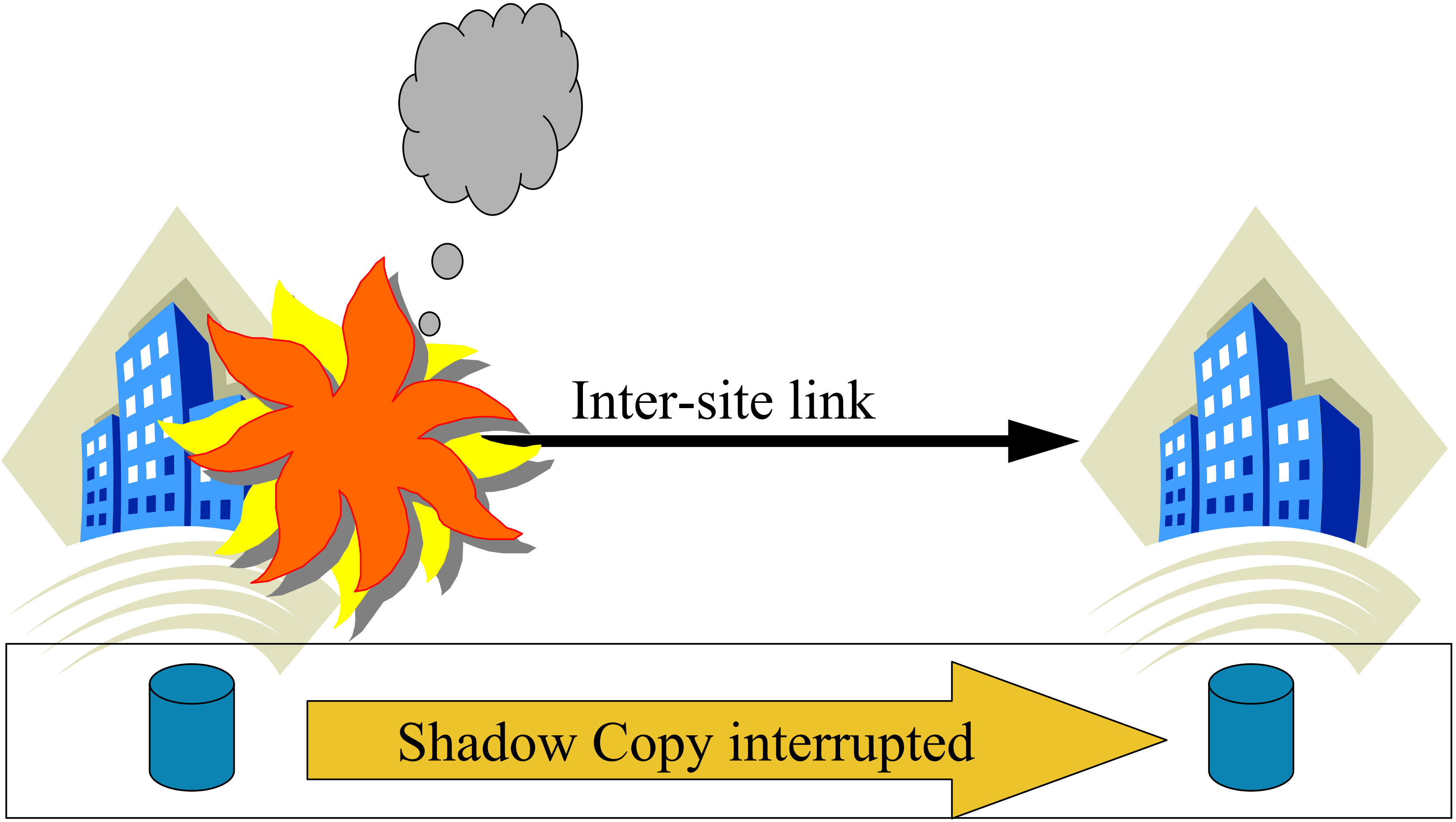
“Rolling Disaster” Scenario

- Disaster or outage makes one site’s data out-of-date
- While re-synchronizing data to the formerly-down site, a disaster takes out the primary site

“Rolling Disaster” Scenario



“Rolling Disaster” Scenario



Source disks destroyed

Partially-updated disks

“Rolling Disaster” Scenario

- Techniques for avoiding data loss due to “Rolling Disaster”:
  - Keep copy (backup, snapshot, clone) of out-of-date copy at target site instead of over-writing the only copy there, or
  - Use a hardware mirroring scheme which preserves write order during re-synch
    - In either case, the surviving copy will be out-of-date, but at least you’ll have some copy of the data
- Keeping a 3rd copy of data at 3rd site is the only way to ensure there is no data lost

Primary CPU Workload

- MSCP-serving in a disaster-tolerant cluster is typically handled in interrupt state on the Primary CPU
  - Interrupts from LAN Adapters come in on the Primary CPU
    - A multiprocessor system may have no more MSCP-serving capacity than a uniprocessor
      - Fast_Path may help
- Lock mastership workload for remote lock requests can also be a heavy contributor to Primary CPU interrupt state usage

Primary CPU interrupt-state saturation

- OpenVMS receives all interrupts on the Primary CPU (prior to 7.3-1)
- If interrupt workload exceeds capacity of Primary CPU, odd symptoms can result
  - CLUEXIT bugchecks, performance anomalies
- OpenVMS has no internal feedback mechanism to divert excess interrupt load
  - e.g. node may take on more trees to lock-master than it can later handle
- Use $MONITOR MODES /CPU=n /ALL to track primary CPU interrupt state usage and peaks (where “n” is the Primary CPU shown by $SHOW CPU)

Interrupt-state/stack saturation

- FAST_PATH:
  - Can shift interrupt-state workload off primary CPU in SMP systems
    - IO_PREFER_CPUS value of an even number disables CPU 0 use
      - Consider limiting interrupts to a subset of non-primaries rather than all
  - FAST_PATH for CI since about 7.1
  - FAST_PATH for SCSI and FC is in 7.3 and above
  - FAST_PATH for LANs (e.g. FDDI & Ethernet) in 7.3-2
  - FAST_PATH for Memory Channel probably “never”
  - Even with FAST_PATH enabled, CPU 0 still received the device interrupt, but handed it off immediately via an inter-processor interrupt
    - 7.3-1 allows interrupts for FAST_PATH devices to bypass the Primary CPU entirely and go directly to a non-primary CPU

Making System Management of Disaster-Tolerant Clusters More Efficient

- Most disaster-tolerant clusters have multiple system disks
  - This tends to increase system manager workload for applying upgrades and patches for OpenVMS and layered products to each system disk
- Techniques are available which minimize the effort involved

Making System Management of Disaster-Tolerant Clusters More Efficient

- Create a “cluster-common” disk
  - Cross-site shadowset
  - Mount it in SYLOGICALS.COM
  - Put all cluster-common files there, and define logicals in SYLOGICALS.COM to point to them:
    - SYSUAF, RIGHTSLIST
    - Queue file, LMF database, etc.

Making System Management of Disaster-Tolerant Clusters More Efficient

- Put startup files on cluster-common disk also; and replace startup files on all system disks with a pointer to the common one:
  - e.g. SYS$STARTUP:STARTUP_VMS.COM contains only:

```
$ @CLUSTER_COMMON:SYSTARTUP_VMS
```

- To allow for differences between nodes, test for node name in common startup files, e.g.

```
$ NODE = F$GETSYI(“NODENAME”)
$ IF NODE .EQS. “GEORGE” THEN ...
```

Making System Management of Disaster-Tolerant Clusters More Efficient

- Create a MODPARAMS_COMMON.DAT file on the cluster-common disk which contains system parameter settings common to all nodes
  - For multi-site or disaster-tolerant clusters, also create one of these for each site
- Include an AGEN$INCLUDE_PARAMS line in each node-specific MODPARAMS.DAT to include the common parameter settings

Making System Management of Disaster-Tolerant Clusters More Efficient

- Use “Cloning” technique to replicate system disks and avoid doing “n” upgrades for “n” system disks

System disk “Cloning” technique

- Create “Master” system disk with roots for all nodes. Use Backup to create Clone system disks.
  - To minimize disk space, move dump files off system disk for all nodes
- Before an upgrade, save any important system-specific info from Clone system disks into the corresponding roots on the Master system disk
  - Basically anything that’s in SYS$SPECIFIC:[*]
  - Examples: ALPHAVMSSYS.PAR, MODPARAMS.DAT, AGEN$FEEDBACK.DAT
- Perform upgrade on Master disk
- Use Backup to copy Master to Clone disks again.

Implementing LAVC$FAILURE_ANALYSIS

- Template program is found in SYS$EXAMPLES: and called LAVC$FAILURE_ANALYSIS.MAR
- Written in Macro-32
  - but you don’t need to know Macro to use it
- Documented in Appendix D of OpenVMS Cluster Systems Manual
  - Appendix E (subroutines the above program calls) and Appendix F (general info on troubleshooting LAVC LAN problems) are also very helpful

Using LAVC$FAILURE_ANALYSIS

- To use, the program must be

1. Edited to insert site-specific information
2. Compiled (assembled on VAX)
3. Linked, and
4. Run at boot time on each node in the cluster

Maintaining LAVC$FAILURE_ANALYSIS

- Program must be re-edited whenever:
  - The LAVC LAN is reconfigured
  - A node's MAC address changes
    - e.g. Field Service replaces a LAN adapter without swapping MAC address ROMs
  - A node is added or removed (permanently) from the cluster

How Failure Analysis is Done

- OpenVMS is told what the network configuration should be
- From this info, OpenVMS infers which LAN adapters should be able to “hear” Hello packets from which other LAN adapters
- By checking for receipt of Hello packets, OpenVMS can tell if a path is working or not

How Failure Analysis is Done

- By analyzing Hello packet receipt patterns and correlating them with a mathematical graph of the network, OpenVMS can tell what nodes of the network are passing Hello packets and which appear to be blocking Hello packets
- OpenVMS determines a Primary Suspect (and, if there is ambiguity as to exactly what has failed, an Alternate Suspect), and reports these via OPCOM messages with a “%LAVC” prefix

Getting Failures Fixed

- Since notification is via OPCOM messages, someone or something needs to be scanning OPCOM output and taking action
- ConsoleWorks, Console Manager, CLIM, or RoboMon can scan for %LAVC messages and take appropriate action (e-mail, pager, etc.)

Gathering Info

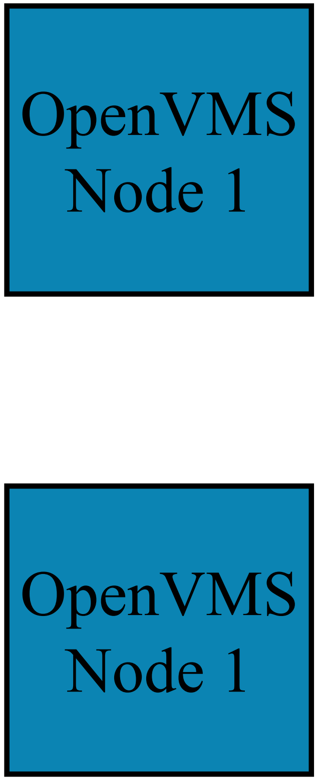
- Data required:
  - Local Area Network configuration:
    - VMS Nodes
    - LAN adapters in each node
    - Bridges
    - Hubs
    - Links between all of the above

Network Information

- OpenVMS considers LAN building blocks as being divided into 4 classes:
  - NODE: The OpenVMS systems
  - ADAPTER: LAN host-bus adapters in each OpenVMS system
  - COMPONENT: Hubs, bridges, bridge-routers
  - CLOUD: Combinations of components that can't be diagnosed directly (more on this later)

Network building blocks

NODEs

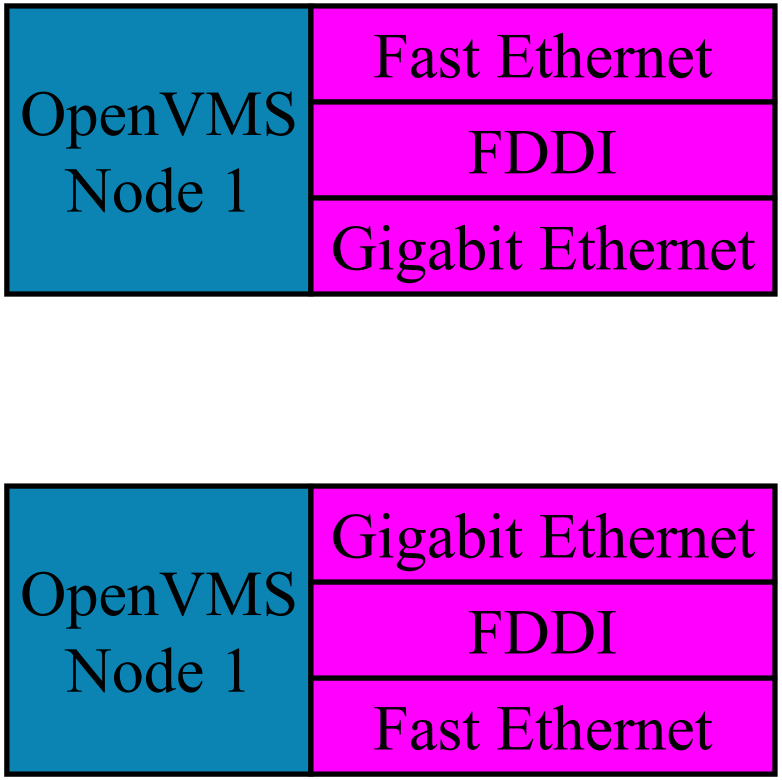


OpenVMS Node 1

OpenVMS Node 1

Network building blocks

NODEs ADAPTERs



OpenVMS Node 1

Fast Ethernet

FDDI

Gigabit Ethernet

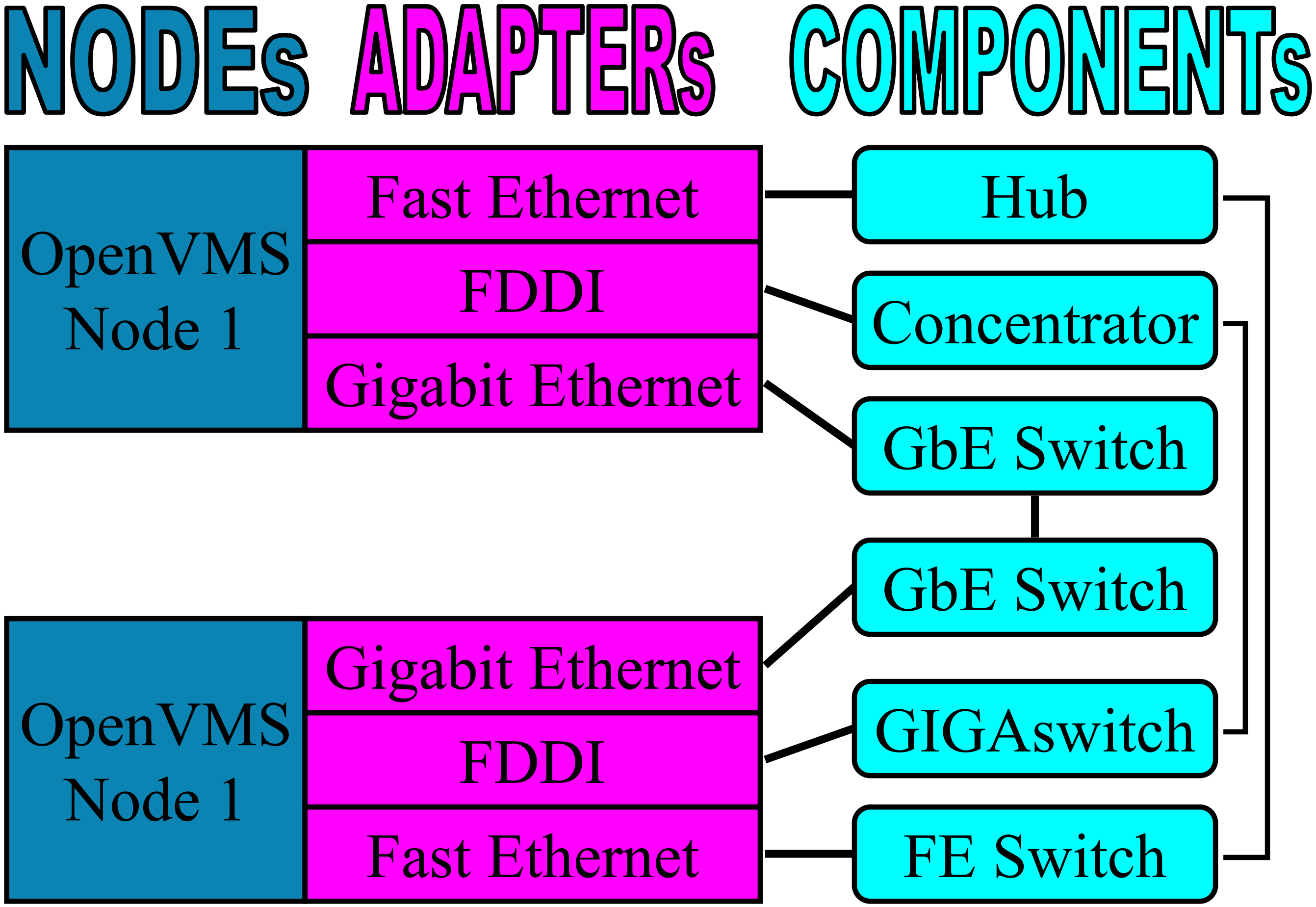
OpenVMS Node 1

Gigabit Ethernet

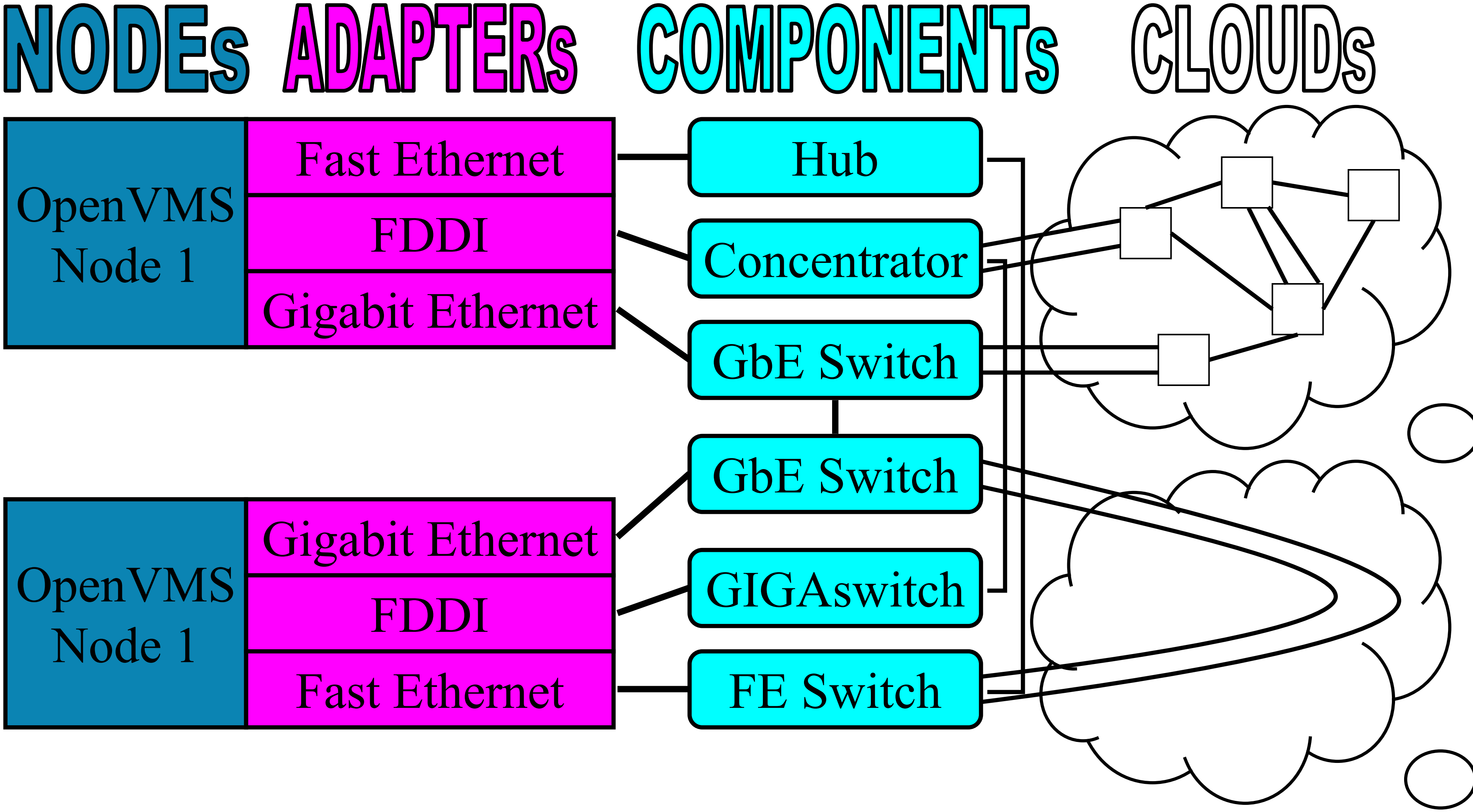
FDDI

Fast Ethernet

Network building blocks



Network building blocks



Handling Network Loops

- The algorithm used for LAVC$FAILURE_ANALYSIS can't deal with loops in the network graph
  - Yet redundancy is often configured among LAN components
  - The bridges' Spanning Tree algorithm shuts off backup links unless and until a failure occurs
    - Hello packets don't get through these backup links, so OpenVMS can't track them
  - For these cases, you replace the redundant portion of the network with a "network cloud" that includes all of the redundant components
    - Then OpenVMS can determine if the network "cloud" as a whole is functioning or not

Handling Redundancy

- Multiple, completely separate LANs don’t count as “loops” and OpenVMS can track each one separately and simultaneously

Gathering Info

- Data required (more detail):
  - Node names and descriptions
  - LAN adapter types and descriptions, and:
    - MAC address

      e.g. 08-00-2B-xx-xx-xx, 00-00-F8-xx-xx-xx
    - plus DECnet-style MAC address for Phase IV

      e.g. AA-00-04-00-yy-zz

Getting MAC address info

```
$! SHOWLAN.COM
$!
$       write sys$output "Node ",f$getsyi("nodename")
$       temp_file := showlan_temp.temp_file
$       call showlan/out='temp_file'
$       search 'temp_file' "(SCA)","Hardware Address" -
               /out='temp_file`-1
$       delete 'temp_file';*
$       search/window=(0,1) 'temp_file`-1 "(SCA)"
$       delete 'temp_file`-1;*
$       exit
$!
$ showlan: subroutine
$        analyze/system
show lan/full
exit
$       endsubroutine
```

Editing the Program

- Once the data is gathered, you edit the program
- There are 5 sections to edit, as follows:

Edit 1

- In Edit 1, you can give descriptive names to nodes, adapters, components, and clouds
- These names become names of macros which you'll create invocations of later in the code

Edit 1

```
;       Edit 1.
;
;              Define the hardware components needed to
describe
;              the physical configuration.
;

        NEW_COMPONENT  SYSTEM         NODE
        NEW_COMPONENT  LAN_ADP        ADAPTER
        NEW_COMPONENT  DEMPR          COMPONENT
        NEW_COMPONENT  DELNI          COMPONENT
        NEW_COMPONENT  SEGMENT        COMPONENT
        NEW_COMPONENT  NET_CLOUD      CLOUD
```

Edit 2

- In Edit 2, you create ASCII art to document the LAVC LAN configuration
- This has no functional effect on the code, but helps you (and others who follow you) understand the information in the sections which follow
- In the drawing, you choose brief abbreviated names for each network building block (Node, Adapter, Component, or Cloud)
  - These abbreviated names are only used within the program, and do not appear externally

Edit 2

```
;       Edit 2.
;
;                         Diagram of a multi-adapter LAV cluster.
;
;
;        Sa   -------+---------------+---------------+---------------+-------
Sa
;                    |               |               |               |
;                    |             MPR_A             |               |
;                    |          .----+----.          |               |
;                    |         1|        1|         1|               |
;                   BrA       ALPHA     BETA       DELTA            BrB
;                    |         2|        2|         2|               |
;                    |          `----+----'          |               |
;                    |             LNI_A             |               |
;                    |               |               |               |
;        Sb   -------+---------------+---------------+---------------+-------
Sb
```

Edit 3

- In Edit 3, you name and provide a text description for each system and its LAN adapter(s), and the MAC address of each adapter
  - The name and text description will appear in OPCOM messages indicating when failure or repair has occurred
  - The MAC address is used to identify the origin of Hello messages

Edit 3

- For DECnet Phase IV, which changes the MAC address on all circuits it knows about from the default hardware address to a special DECnet address when it starts up, you provide both:
  - The hardware MAC address (e.g. 08-00-2B-nn-nn-nn) and
  - The DECnet-style MAC address which is derived from the DECnet address of the node (AA-00-04-00-yy-xx)
- DECnet Phase V does not change the MAC address, so only the HW address is needed

Edit 3

```
;      Edit 3.
;
;              Label Node                     Description                     LAN HW Addr         DECnet Addr
;              ----- -----   ------------------------------------------   --------------------   -------------------

      SYSTEM   A,    ALPHA,  < - MicroVAX II; In the Computer room>
      LAN_ADP  A1,   ,       <XQA; ALPHA - MicroVAX II; Computer room>,   <08-00-2B-41-41-01>,  <AA-00-04-00-01-04>
      LAN_ADP  A2,   ,       <XQB; ALPHA - MicroVAX II; Computer room>,   <08-00-2B-41-41-02

      SYSTEM   B,    BETA,   < - MicroVAX 3500; In the Computer room>
      LAN_ADP  B1,    ,      <XQA; BETA - MicroVAX 3500; Computer room>,  <08-00-2B-42-42-01>,  <AA-00-04-00-02-04>
      LAN_ADP  B2,    ,      <XQB; BETA - MicroVAX 3500; Computer room>,  <08-00-2B-42-42-02>

      SYSTEM   D,    DELTA,  < - VAXstation II; In Dan's office>
      LAN_ADP  D1,    ,      <XQA; DELTA - VAXstation II; Dan's office>,  <08-00-2B-44-44-01>,  <AA-00-04-00-04-04
      LAN_ADP  D2,    ,      <XQB; DELTA - VAXstation II; Dan's office>,  <08-00-2B-44-44-02>
```

Edit 4

- In Edit 4, you name and provide a text description for each Component and each Cloud
  - The name and text description will appear in OPCOM messages indicating when failure or repair has occurred

Edit 4

```
;         Edit 4.
;
;                   Label each of the other network components.
;

          DEMPR     MPR_A, , <Connected to segment A; In the Computer room>
          DELNI     LNI_A, , <Connected to segment B; In the Computer room>

          SEGMENT   Sa,  , <Ethernet segment A>
          SEGMENT   Sb,  , <Ethernet segment B>

          NET_CLOUD BRIDGES, , <Bridging between ethernet segments A and B>
```

Edit 5

- In Edit 5, you indicate which network building blocks have connections to each other
- This is a list of pairs of devices, indicating they are connected

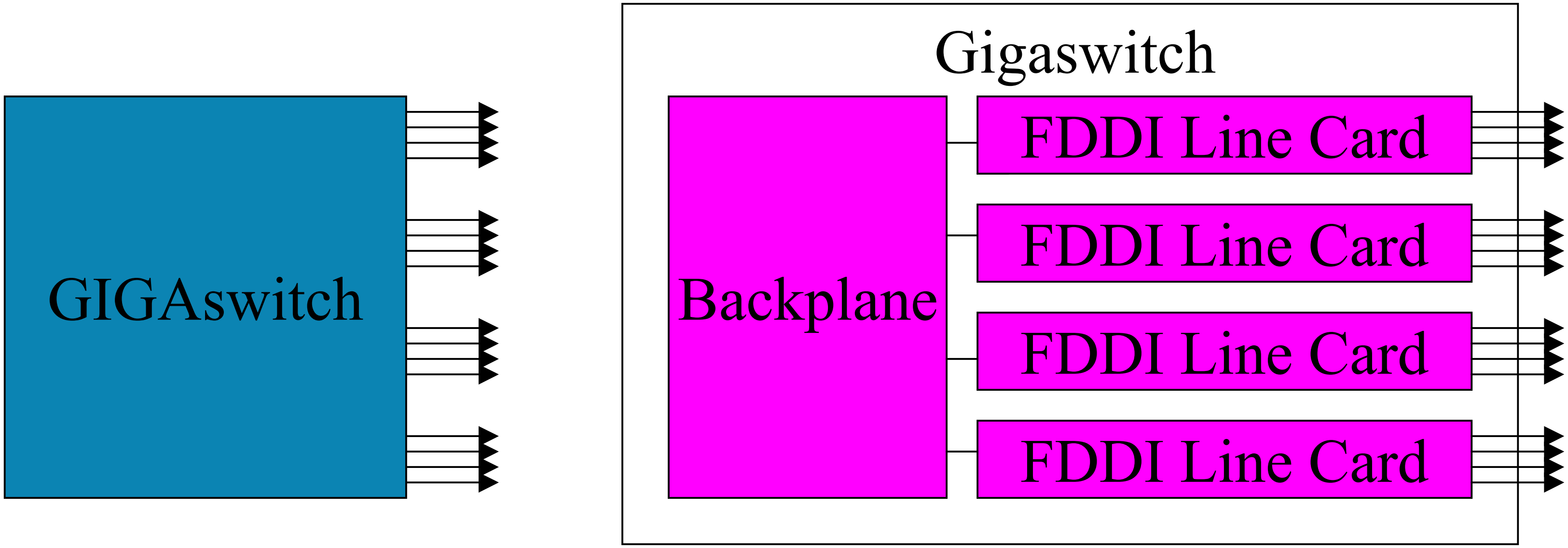
Edit 5

```
;           Edit 5.
;
;                       Describe the network connections.
;

            CONNECTION          Sa,         MPR_A
            CONNECTION                      MPR_A,      A1
            CONNECTION                                  A1,         A
            CONNECTION                      MPR_A,      B1
            CONNECTION                                  B1,         B

            CONNECTION          Sa,         D1
            CONNECTION                      D1,         D

            CONNECTION          Sa,         BRIDGES
            CONNECTION          Sb,         BRIDGES

            CONNECTION          Sb,         LNI_A
            CONNECTION                      LNI_A,      A2
            CONNECTION                                  A2, A
            CONNECTION                      LNI_A,      B2
            CONNECTION                                  B2, B

            CONNECTION          Sb,         D2
            CONNECTION                      D2,         D
```

Level of Detail

- There is a trade-off between level of detail in diagnostic info and the amount of work required to initially set up and to maintain the program over time
  - More detail means more work to setup, and more maintenance work, but can provide more-specific diagnostic info when failures occur

Level of Detail Example



EDIT_LAVC.COM Tool

- A DCL command procedure is available to gather the information and create an example LAVC$FAILURE_ANALYSIS.MAR program customized for a given cluster. See
  - http://encompasserve.org/~parris/edit_lavc.com and EDIT_LAVC_DOC.TXT at the same location

- To turn off LAVC Failure Analysis, use the LAVC$FAILURE_OFF.MAR program found at the same location

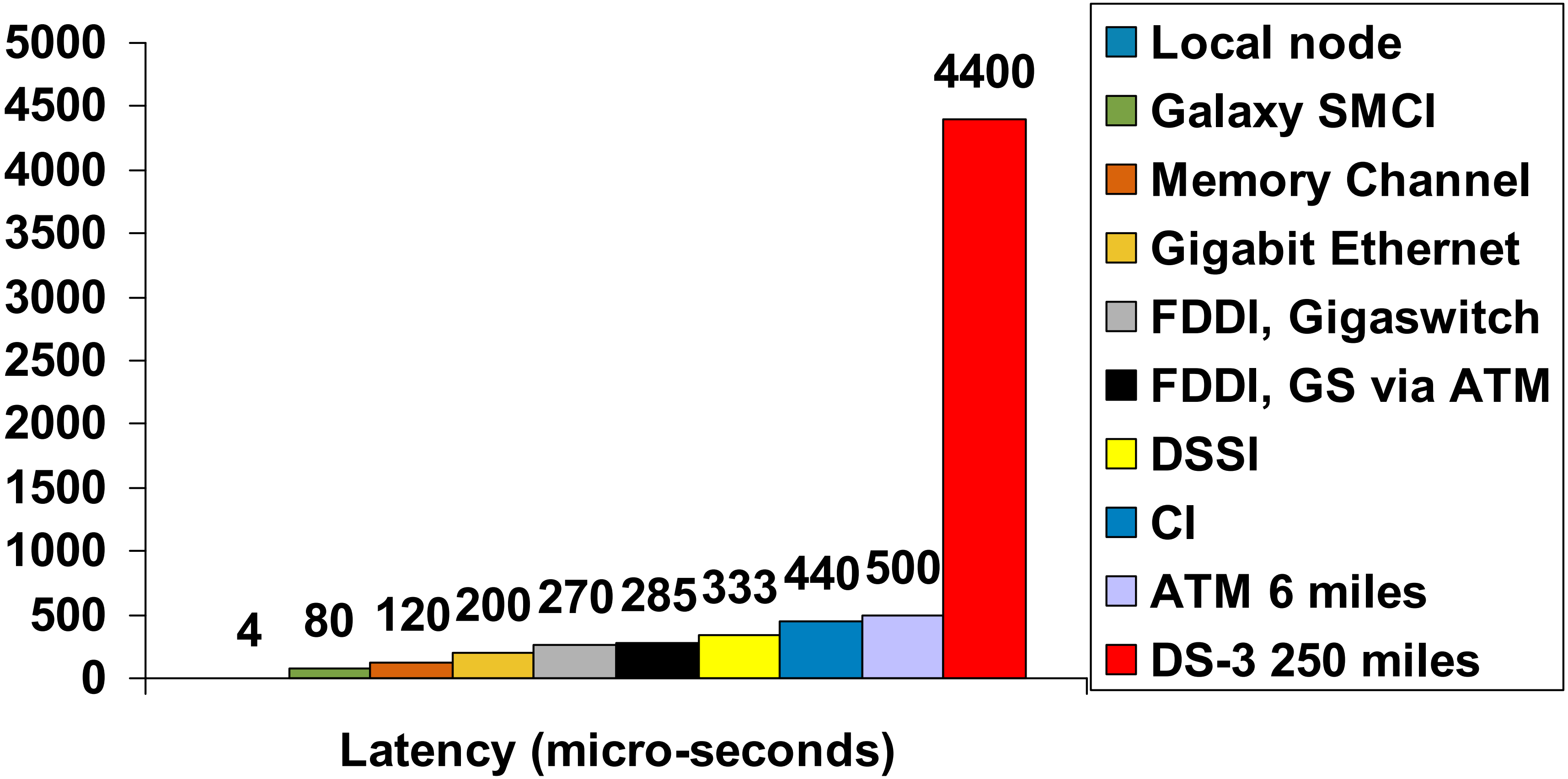
Long-Distance Clusters

- OpenVMS SPD supports distance of up to 150 miles (250 kilometers) between sites
  - up to 500 miles (833 kilometers) with DTCS or BRS
- Why the limit?
  - Inter-site latency, and its effect on performance

Long-distance Cluster Issues

- Latency due to speed of light becomes significant at higher distances. Rules of thumb:
  - About 1 millisecond per 100 miles, one-way or
  - About 1 millisecond per 50 miles, round-trip latency
- Actual circuit path length can be longer than highway driving distance between sites
- Latency primarily affects performance of:
  1. Remote lock requests (including lock directory lookups)
  2. Remote I/Os

Inter-site Round-Trip Lock Request Latencies



Differentiate between latency and bandwidth

- Can’t get around the speed of light and its latency effects over long distances
  - Higher-bandwidth link doesn’t mean lower latency
    - Multiple links may help latency somewhat under heavy loading due to shorter queue lengths, but can’t outweigh speed-of-light issues

Latency of Inter-Site Link

- Latency affects performance of:
  - Lock operations that cross the inter-site link
    - Lock requests
    - Directory lookups, deadlock searches
  - Write I/Os to remote shadowset members, either:
    - Over SCS link through the OpenVMS MSCP Server on a node at the opposite site, or
    - Direct via Fibre Channel (with an inter-site FC link)

  Both MSCP and the SCSI-3 protocol used over FC take a minimum of two round trips for writes

Mitigating Impact of Inter-Site Latency

- Locking
  - Try to avoid lock requests to master node at remote site
    - OpenVMS does move mastership of a resource tree to the node with the most activity
  - Lock directory lookups with directory node at remote site can only be avoided by setting LOCKDIRWT to zero on all nodes at the remote site
    - This is typically only satisfactory for Primary/Backup or remote-shadowing-only clusters

Mitigating Impact of Inter-Site Latency

- Check SHOW CLUSTER/CONTINUOUS with ADD CONNECTIONS, ADD REM_PROC_NAME and ADD CR_WAITS to check for SCS credit waits. If counts are present and increasing over time, increase the SCS credits *at the remote end* as follows:

Mitigating Impact of Inter-Site Latency

- For credit waits on VMS$VAXcluster SYSAP connections:
  - Increase CLUSTER_CREDITS parameter
  - Default is 10; maximum is 128

- For credit waits on VMS$DISK_CL_DRVR / MSCP$DISK connections:
  - For OpenVMS server node, increase MSCP_CREDITS parameter. Default is 8; maximum is 128.
  - For HSJ/HSD controller, lower MAXIMUM_HOSTS from default of 16 to actual number of OpenVMS systems on the CI/DSSI interconnect

Local versus Remote operations

- Optimize local operations:
  - Read I/Os:
    - Specify SITE or READ_COST for member disks with OpenVMS 7.3 or above or recent VOLSHAD ECO kits, or, for earlier versions,
    - Set SHADOW_SYS_DISK bit %X10000 (bit 16) to select local-read optimization (to favor CI/DSSI disks over MSCP-served disks), if applicable

Application Scheme 1: Hot Primary/Cold Standby

- All applications normally run at the primary site
  - Second site is idle, except for volume shadowing, until primary site fails, then it takes over processing
- Performance will be good (all-local locking)
- Fail-over time will be poor, and risk high (standby systems not active and thus not being tested)
- Wastes computing capacity at the remote site

Application Scheme 2: Hot/Hot but Alternate Workloads

- All applications normally run at one site or the other, but not both; data is shadowed between sites, and the opposite site takes over upon a failure
- Performance will be good (all-local locking)
- Fail-over time will be poor, and risk moderate (standby systems in use, but specific applications not active and thus not being tested from that site)
- Second site’s computing capacity is actively used

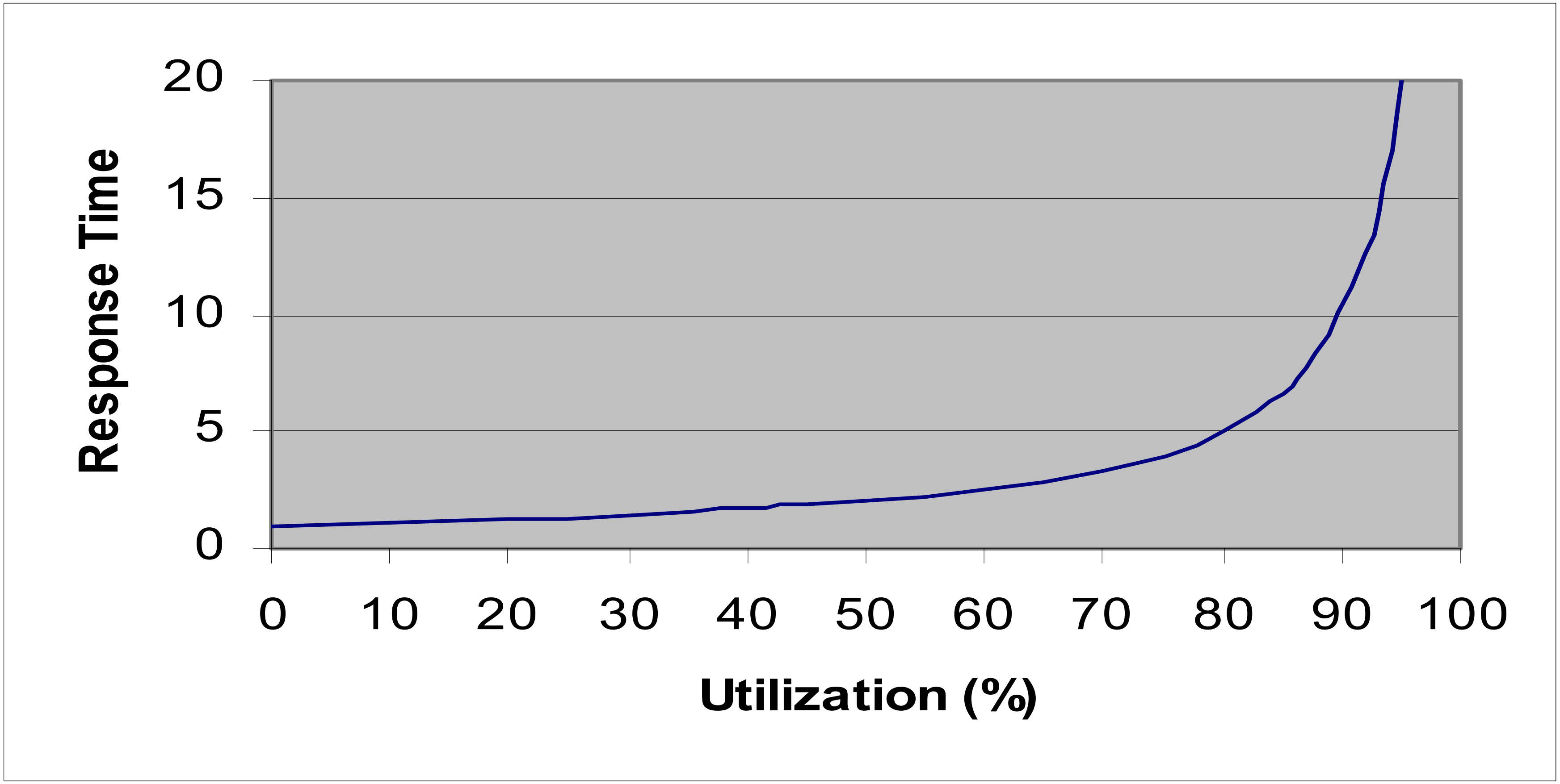
Application Scheme 3: Uniform Workload Across Sites

- All applications normally run at both sites simultaneously; surviving site takes all load upon failure
- Performance may be impacted (some remote locking) if inter-site distance is large
- Fail-over time will be excellent, and risk low (standby systems are already in use running the same applications, thus constantly being tested)
- Both sites' computing capacity is actively used

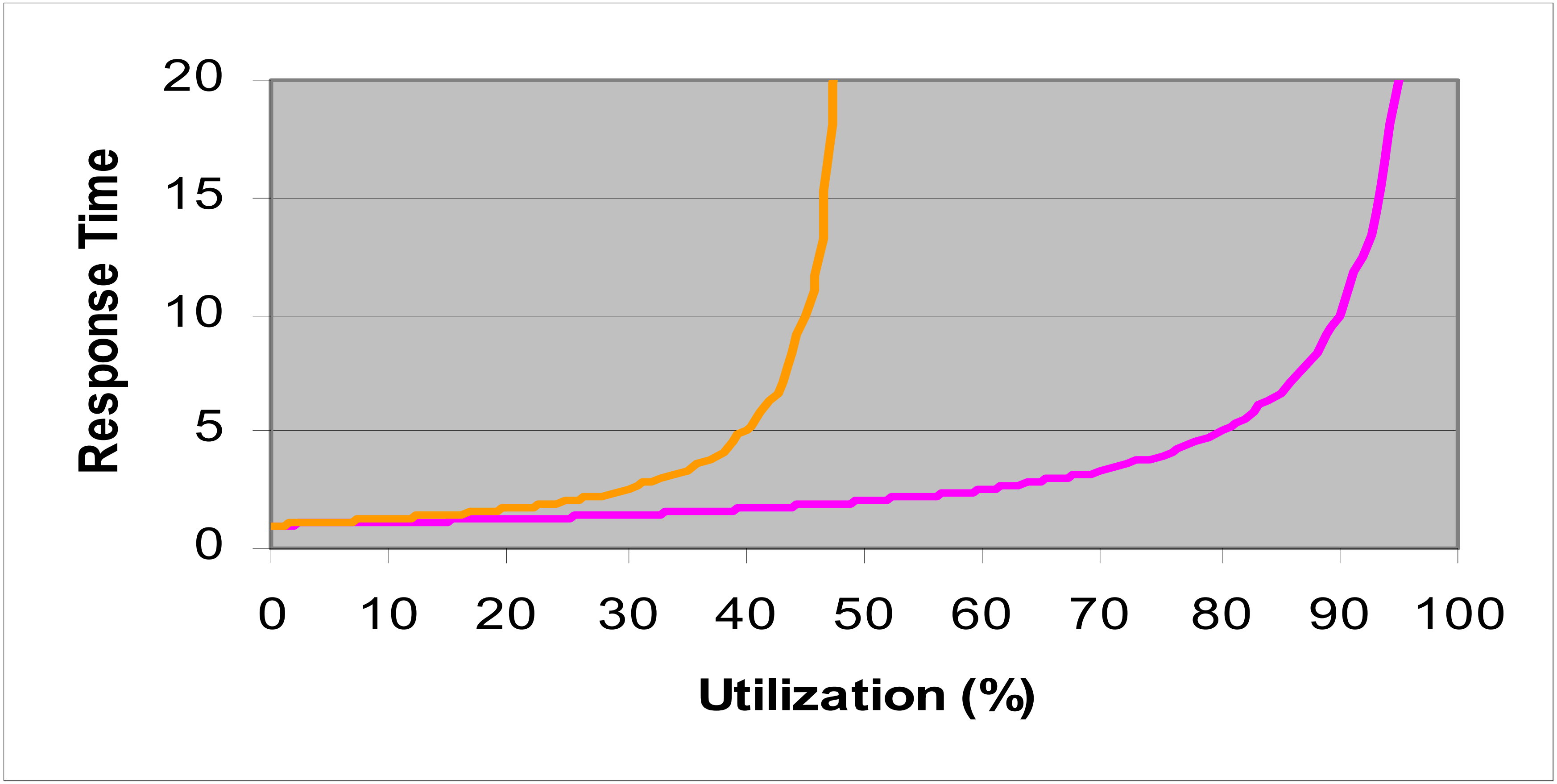
Capacity Considerations

- When running workload at both sites, be careful to watch utilization.
- Utilization over 35% will result in utilization over 70% if one site is lost
- Utilization over 50% will mean there is no possible way one surviving site can handle all the workload

Response time vs. Utilization



Response time vs. Utilization: Impact of losing 1 site



Testing

- Separate test environment is very helpful, and highly recommended
- Good practices require periodic testing of a simulated disaster. This allows you to:
  - Validate your procedures
  - Train your people

Setup Steps for Creating a Disaster-Tolerant Cluster

- Let’s look at the steps involved in setting up a Disaster-Tolerant Cluster from the ground up.
  - Datacenter site preparation
  - Install the hardware and networking equipment
    - Ensure dual power supplies are plugged into separate power feeds
  - Select configuration parameters:
    - Choose an unused cluster group number; select a cluster password
    - Choose site allocation class(es)

Steps for Creating a Disaster-Tolerant Cluster

- Configure storage (if HSx or EVA controllers)
- Install OpenVMS on each system disk
- Load licenses for Open VMS Base, OpenVMS Users, Cluster, Volume Shadowing and, for ease of access, your networking protocols (DECnet and/or TCP/IP)

Setup Steps for Creating a Disaster-Tolerant Cluster

- Create a shadowset across sites for files which will be used on common by all nodes in the cluster. On it, place:
  - SYSUAF and RIGHTSLIST files (copy from any system disk)
  - License database (LMF$LICENSE.LDB)
  - NETPROXY.DAT, NET$PROXY.DAT (DECnet proxy login files), if used; NETNODE_REMOTE.DAT, NETNODE_OBJECT.DAT
  - VMS$MAIL_PROFILE.DATA (OpenVMS Mail Profile file)
  - Security audit journal file
  - Password History and Password Dictionary files
  - Queue manager files
  - System login command procedure SYS$SYLOGIN:
  - LAVC$FAILURE_ANALYSIS program from the SYS$EXAMPLES: area, customized for the specific cluster interconnect configuration and LAN addresses of the installed systems

Setup Steps for Creating a Disaster-Tolerant Cluster

- To create the license database:
  - Copy initial file from any system disk
  - Leave shell LDBs on each system disk for booting purposes (we'll map to the common one in SYLOGICALS.COM)
  - Use LICENSE ISSUE/PROCEDURE/OUT=xxx.COM (and LICENSE ENABLE afterward to re-enable the original license in the LDB on the system disk), then execute the procedure against the common database to put all licenses for all nodes into the common LDB file
  - Add all additional licenses to the cluster-common LDB file (i.e. layered products)

Setup Steps for Creating a Disaster-Tolerant Cluster

- Create a minimal SYLOGICALS.COM that simply mounts the cluster-common shadowset, defines a logical name CLUSTER_COMMON to point to a common area for startup procedures, and then invokes

```
$ @CLUSTER_COMMON:SYLOGICALS.COM
```

Setup Steps for Creating a Disaster-Tolerant Cluster

- Create "shell" command scripts for each of the following files. The "shell" will contain only one command, to invoke the corresponding version of this startup file in the CLUSTER_COMMON area. For example. SYS$STARTUP:SYSTARTUP_VMS.COM on every system disk will contain the single line:

  ```
  $ @CLUSTER_COMMON:SYSTARTUP_VMS.COM
  ```

  Do this for each of the following files:

  - SYCONFIG.COM
  - SYPAGSWPFILES.COM
  - SYSECURITY.COM
  - SYSTARTUP_VMS.COM
  - SYSHUTDWN.COM

  Any command procedures that are called by these cluster-common startup procedures should also be placed in the cluster-common area

Setup Steps for Creating a Disaster-Tolerant Cluster

- Create AUTOGEN “include” files to simplify the running of AUTOGEN on each node:
  - Create one for parameters common to systems at each site. This will contain settings for a given site for parameters such as:
    - ALLOCLASS
    - TAPE_ALLOCLASS
    - Possibly SHADOW_SYS_UNIT (if all systems at a site share a single system disk, this gives the unit number)

Setup Steps for Creating a Disaster-Tolerant Cluster

– Create one for parameters common to every system in the entire cluster. This will contain settings for things like:
  - VAXCLUSTER
  - RECNXINTERVAL (based on inter-site link recovery times)
  - SHADOW_MBR_TMO (typically 10 seconds larger than RECNXINTERVAL)
  - EXPECTED_VOTES (total of all votes in the cluster when all nodes are up)
  - Possibly VOTES (i.e. if all nodes have 1 vote each)
  - DISK_QUORUM=" " (no quorum disk)
  - Probably LOCKDIRWT (i.e. if all nodes have equal values of 1)
  - SHADOWING=2 (enable host-based volume shadowing)
  - NISCS_LOAD_PEA0=1
  - NISCS_MAX_PKTSZ (to use larger FDDI or this plus LAN_FLAGS to use larger Gigabit Ethernet packets)
  - Probably SHADOW_SYS_DISK (to set bit 16 to enable local shadowset read optimization if needed)
  - Minimum values for:
    CLUSTER_CREDITS
    MSCP_BUFFER
    MSCP_CREDITS
  - MSCP_LOAD, MSCP_SERVE_ALL; TMSCP_LOAD, TMSCP_SERVE_ALL
  - Possibly TIMVCFAIL (if faster-than-standard failover times are required)

Setup Steps for Creating a Disaster-Tolerant Cluster

- Pare down the MODPARAMS.DAT file in each system root. It should contain basically only the parameter settings for:
  - SCSNODE
  - SCSSYSTEMID

plus a few AGEN$INCLUDE_PARAMS lines pointing to the CLUSTER_COMMON: area for:

- MODPARAMS_CLUSTER_COMMON.DAT (parameters which are the same across the entire cluster)
- MODPARAMS_COMMON_SITE_xxx.DAT (parameters which are the same for all systems within a given site or lobe of the cluster)
- Architecture-specific common parameter file (Alpha vs. VAX vs. Itanium), if needed (parameters which are common to all systems of that architecture)

Setup Steps for Creating a Disaster-Tolerant Cluster

- Typically, all the other parameter values one tends to see in an individual stand-alone node's MODPARAMS.DAT file will be better placed in one of the common parameter files. This helps ensure consistency of parameter values across the cluster and minimize the system manager's workload and reduce the chances of an error when a parameter value must be changed on multiple nodes.

Setup Steps for Creating a Disaster-Tolerant Cluster

- Place the AGEN$INCLUDE_PARAMS lines at the beginning of the MODPARAMS.DAT file in each system root. The last definition of a given parameter value found by AUTOGEN is the one it uses, so by placing the "include" files in order from cluster-common to site-specific to node-specific, if necessary you can override the cluster-wide and/or site-wide settings on a given node by simply putting the desired parameter settings at the end of a specific node's MODPARAMS.DAT file. This may be needed, for example, if you install and are testing a new version of OpenVMS on that node, and the new version requires some new SYSGEN parameter settings that don't yet apply to the rest of the nodes in the cluster.
  - (Of course, an even more elegant way to handle this particular case would be to create a MODPARAMS_VERSION_xx.DAT file in the common area and include that file on any nodes running the new version of the operating system. Once all nodes have been upgraded to the new version, these parameter settings can be moved to the cluster-common MODPARAMS file.)

Setup Steps for Creating a Disaster-Tolerant Cluster

- Create startup command procedures to mount cross-site shadowsets

Recent Disaster-Tolerant Cluster Developments

- Fibre Channel storage, and Fibre Channel inter-site links
- Data Replication Manager / Continuous Access

Fibre Channel and SCSI in Clusters

- Fibre Channel and SCSI are Storage-Only Interconnects
  - Provide access to storage devices and controllers
    - Storage can be shared between several nodes
      - SCSI Bus or SCSI Hub
      - FC Switch
    - Each node can access the storage directly

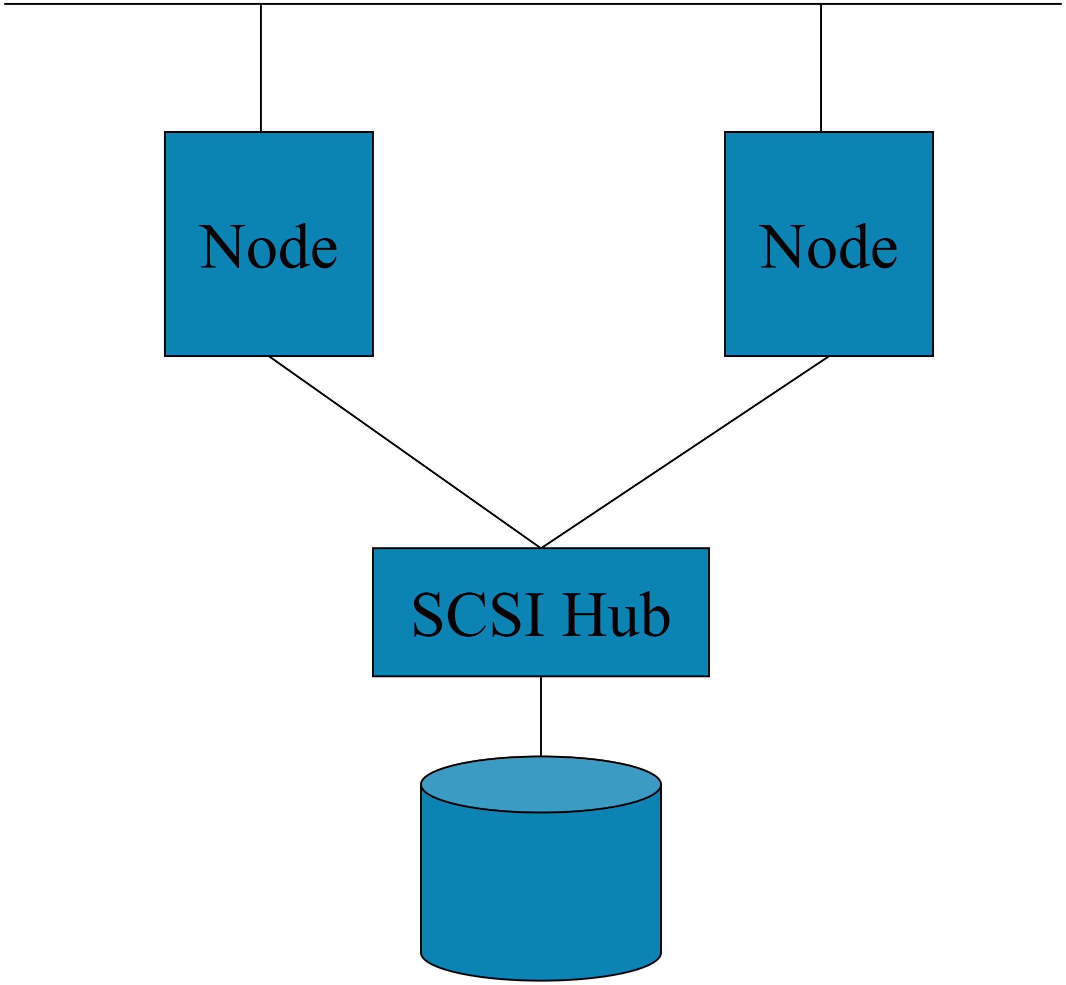
Fibre Channel and SCSI in Clusters

- Fibre Channel or SCSI are Storage-Only Interconnects
  - Cannot carry SCS protocol
    - e.g. Connection Manager and Lock Manager traffic
  - Need SCS-capable Cluster Interconnect also
    - Memory Channel, CI, DSSI, FDDI, Ethernet, or Galaxy Shared Memory C.I. (SMCI)

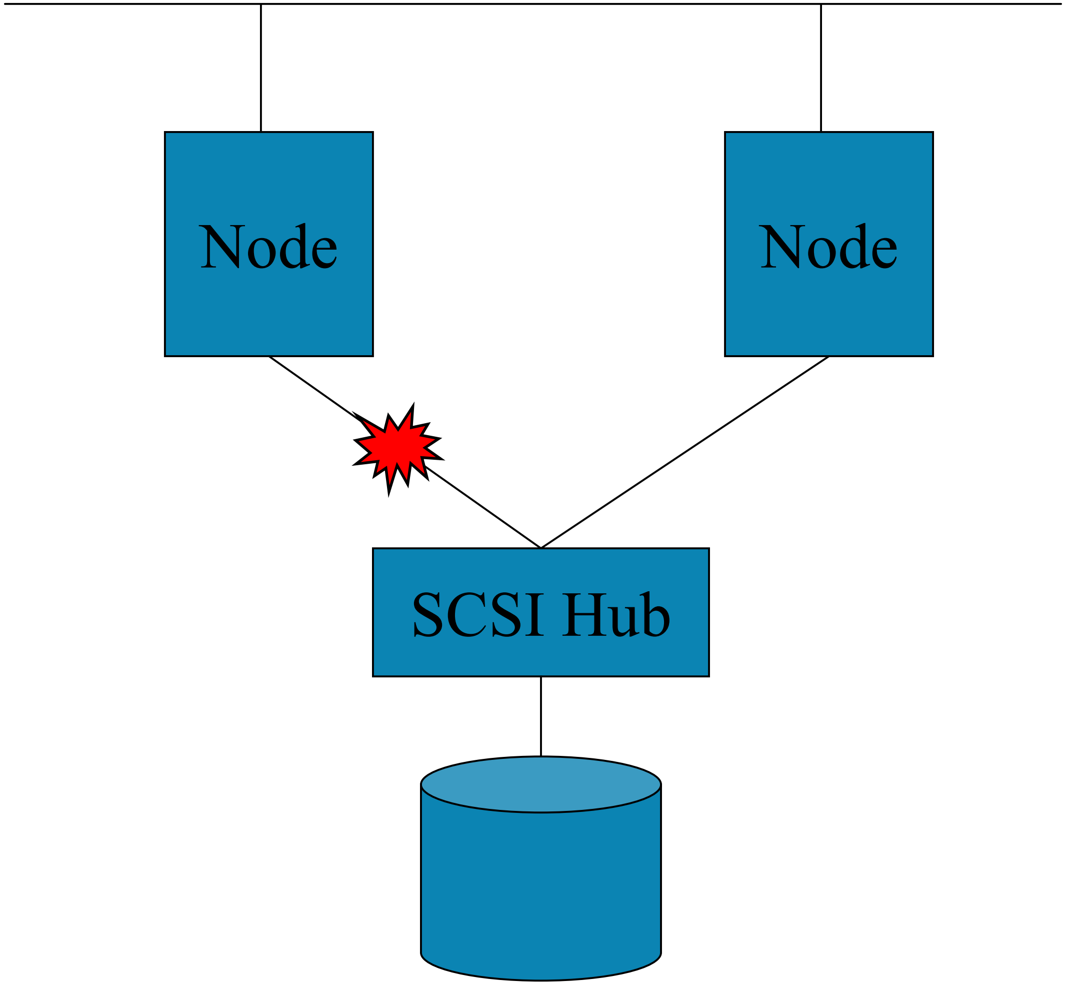
SCSI and FC Notes

- Fail-over between a direct path and an MSCP-served path is first supported in OpenVMS version 7.3-1

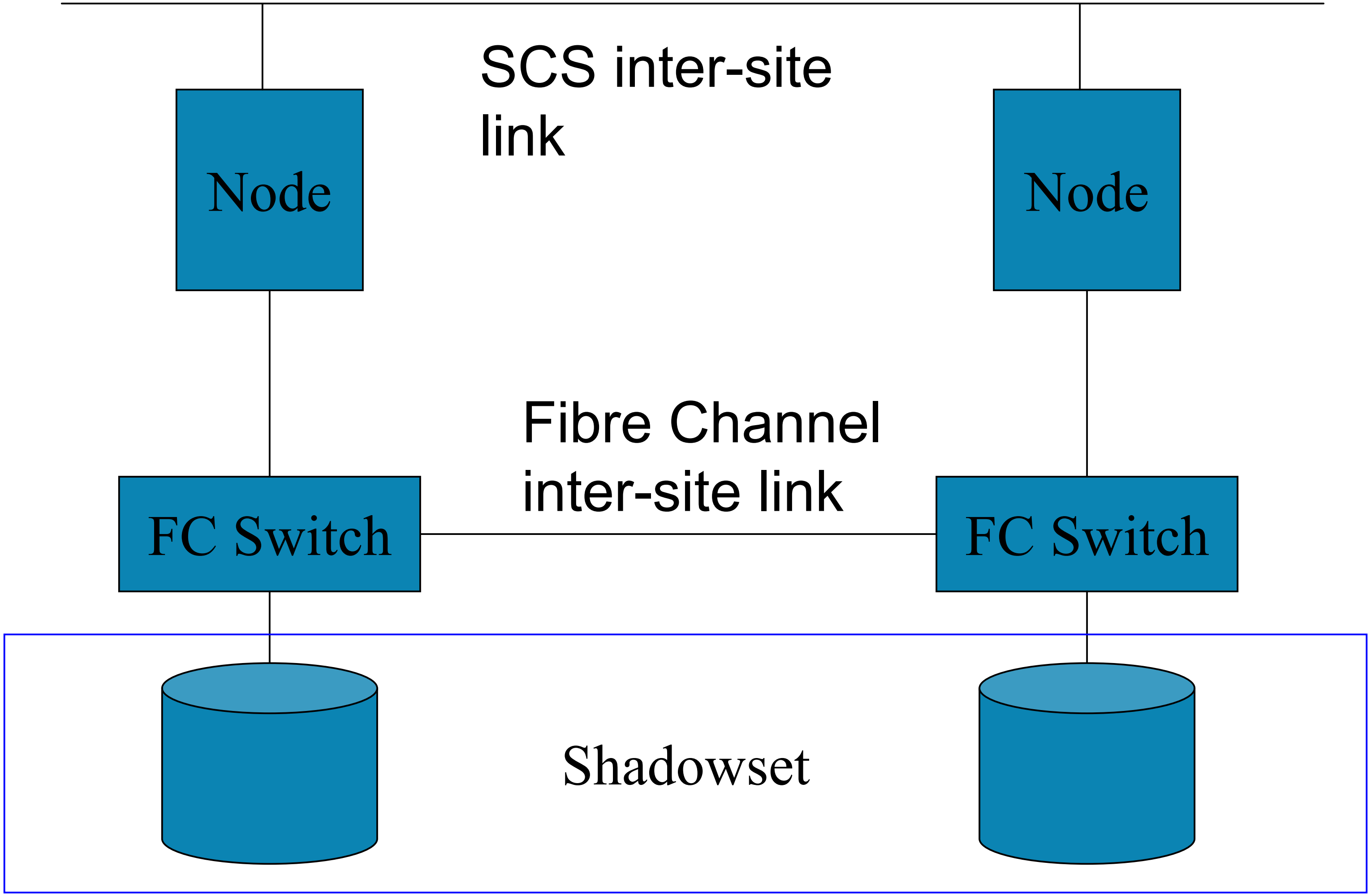
Direct vs. MSCP-Served Paths



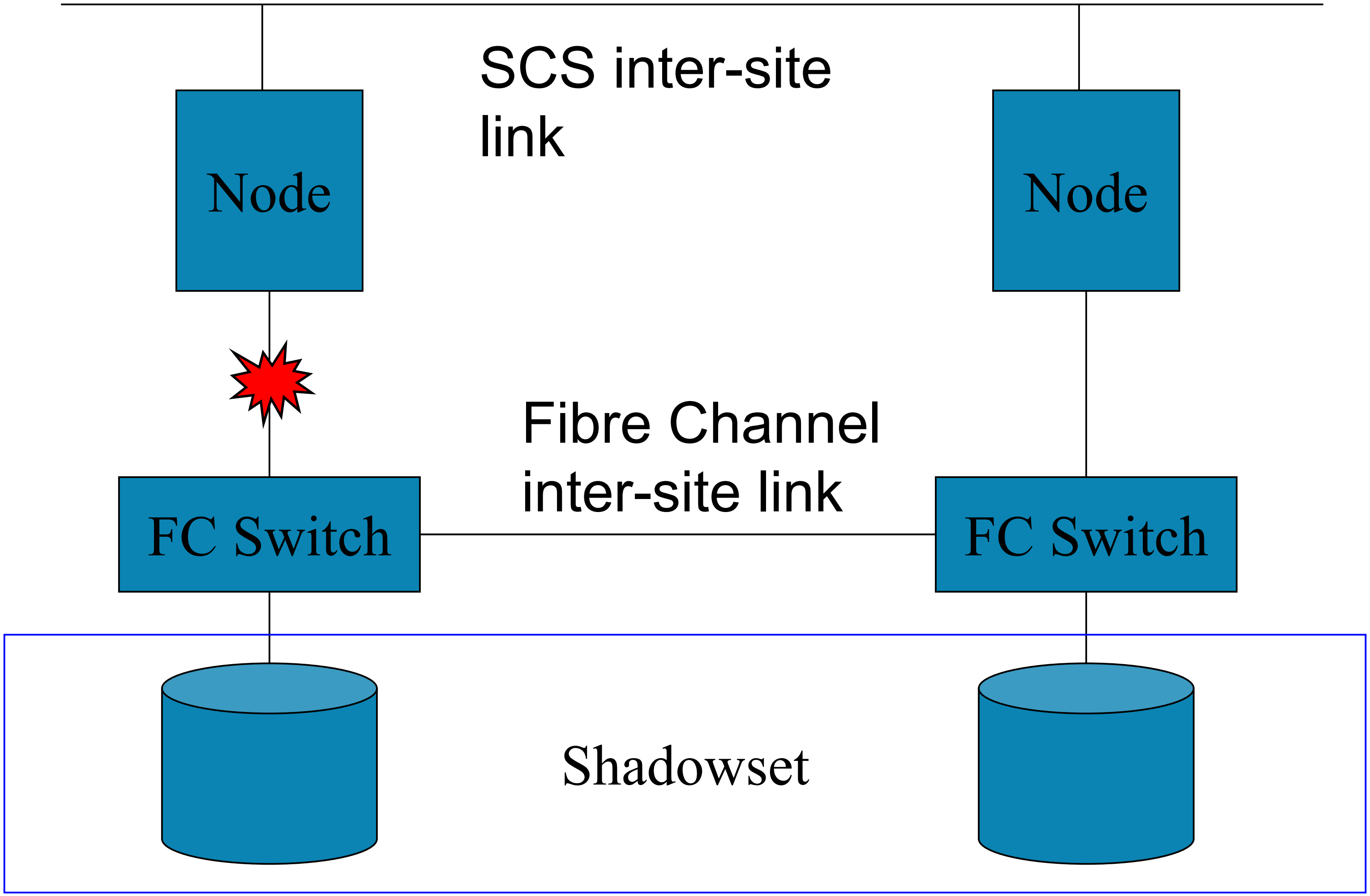
Direct vs. MSCP-Served Paths



Direct vs. MSCP-Served Paths



Direct vs. MSCP-Served Paths



Fibre Channel Notes

- After FC switch, cable, or disk virtual unit reconfiguration, the documentation recommends you do:

```
SYSMAN> IO SCSI_PATH_VERIFY
```

and

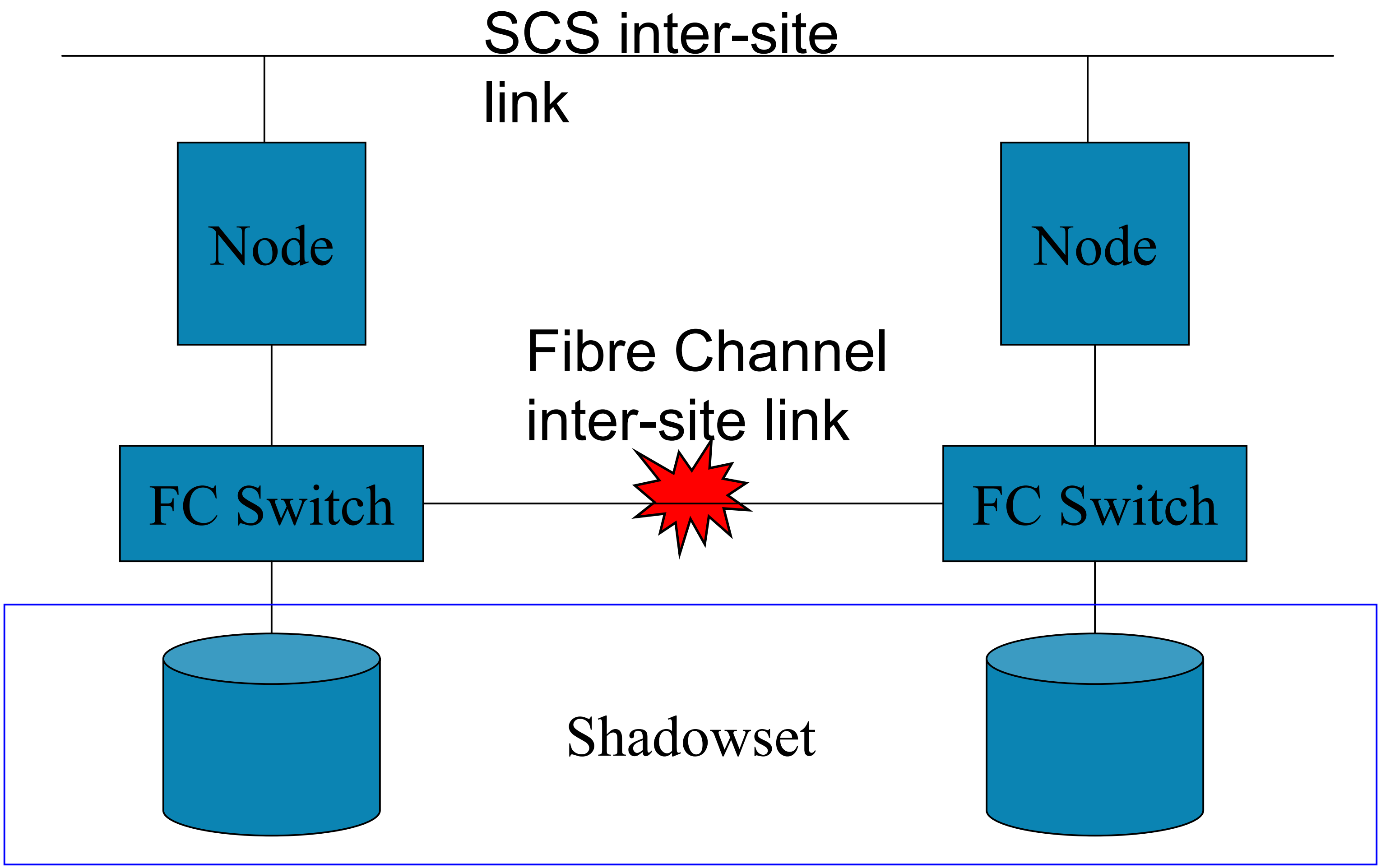
```
SYSMAN> IO AUTOCONFIGURE
```

  - This is unlike CI or DSSI, where path changes are detected and new devices appear automatically

Fibre Channel Notes

- Volume Shadowing Mini-Merges are not supported for SCSI or Fibre Channel
  - Full Merge will occur after any node crashes while it has a shadowset mounted
- HSG80-specific Mini-Merge project was cancelled; general Host-Based Mini-Merge capability is now under development

New Failure Scenario: SCS link OK but FC link broken



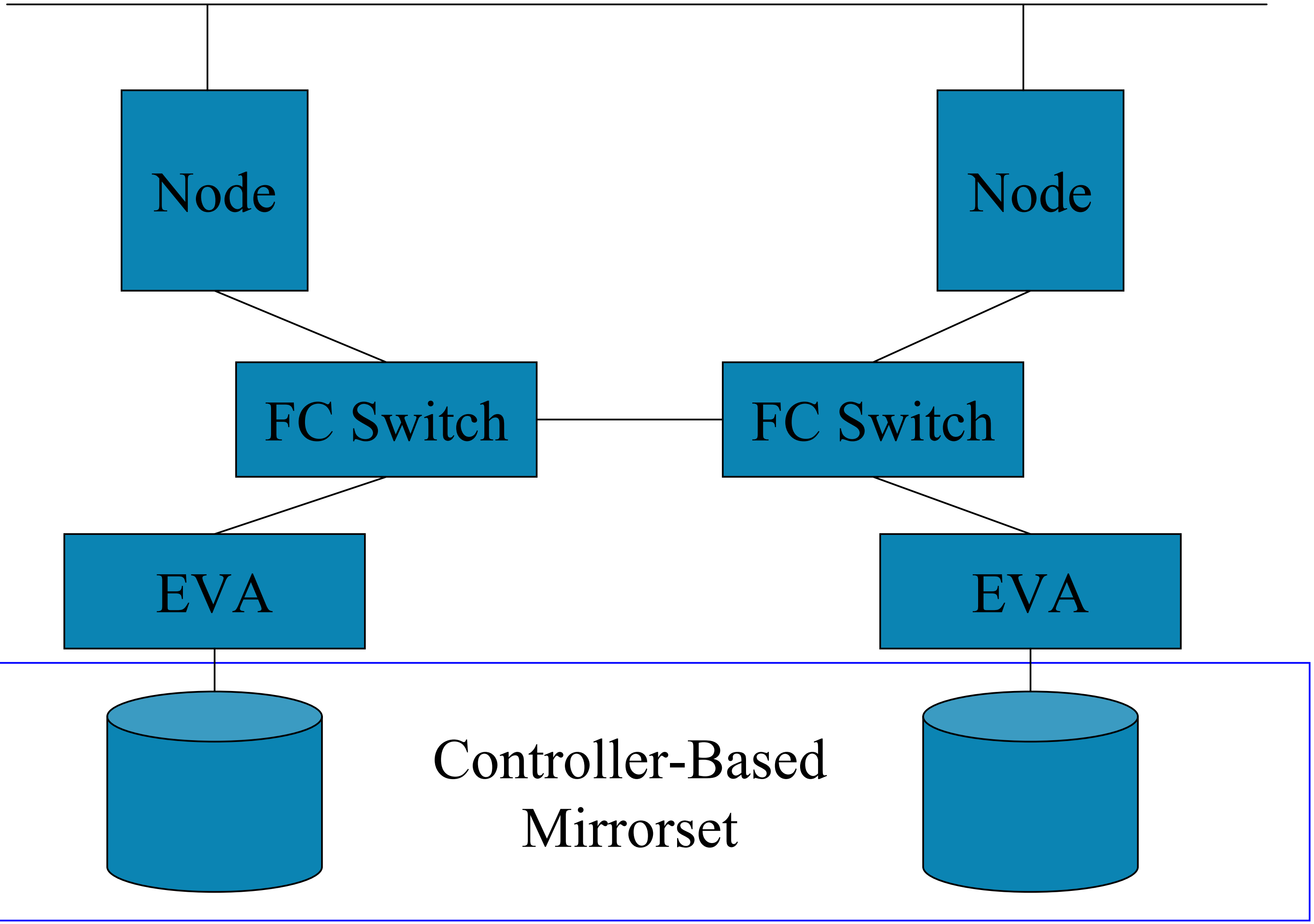
New Failure Scenario: SCS link OK but FC link broken

- With inter-site FC link broken, Volume Shadowing doesn’t know which member to remove
  - Node on left wants to remove right-hand disk, and vice-versa
- New DCL commands allow you to force removal of one member manually
  - See white paper on “Fibre Channel in a Disaster-Tolerant OpenVMS Cluster System” at http://h71000.www7.hp.com/openvms/fibre/fc_hbvs_dtc_wp.pdf
  - Assumes you are logged in already, or SYSUAF, etc. aren’t on a cross-site shadowed disk
    - Not easy to use in practice, in my experience
- Problem solved by direct-to-MSCP-served failover in 7.3-1

Cross-site Shadowed System Disk

- With only an SCS link between sites, it was impractical to have a shadowed system disk and boot nodes from it at multiple sites
- With a Fibre Channel inter-site link, it becomes possible to have a single system disk shadowed across sites, instead of a separate system disk per site.
  - But this is probably still not a good idea, because:
    - The single system disk would be a single point of failure for the cluster

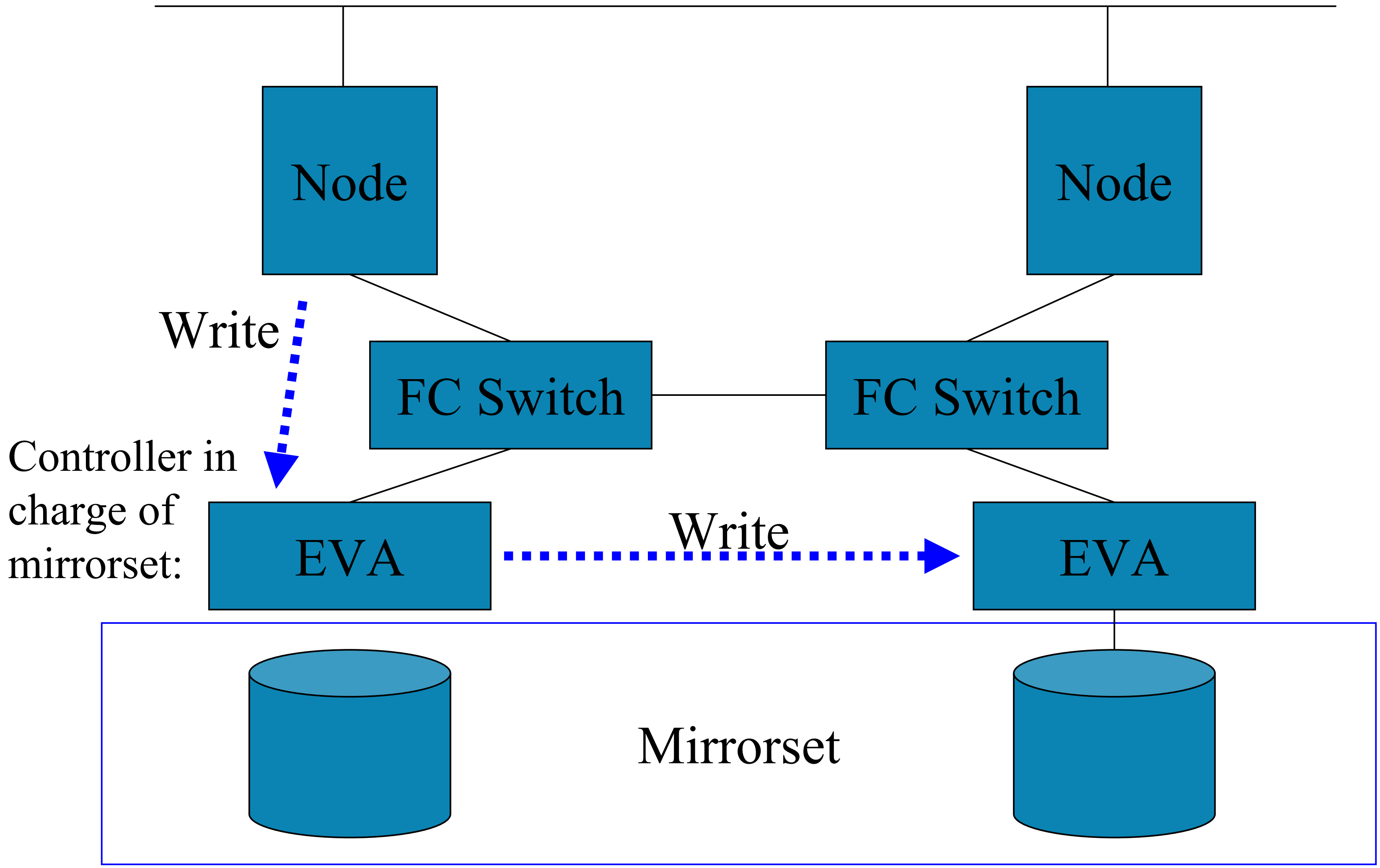
Data Replication Manager / Continuous Access



Data Replication Manager / Continuous Access

- Multi-site mirroring between HSG80 or EVA controllers
- Host writes to one controller
  - That controller sends data to the controller at the opposite site through Fibre Channel

Data Replication Manager / Continuous Access



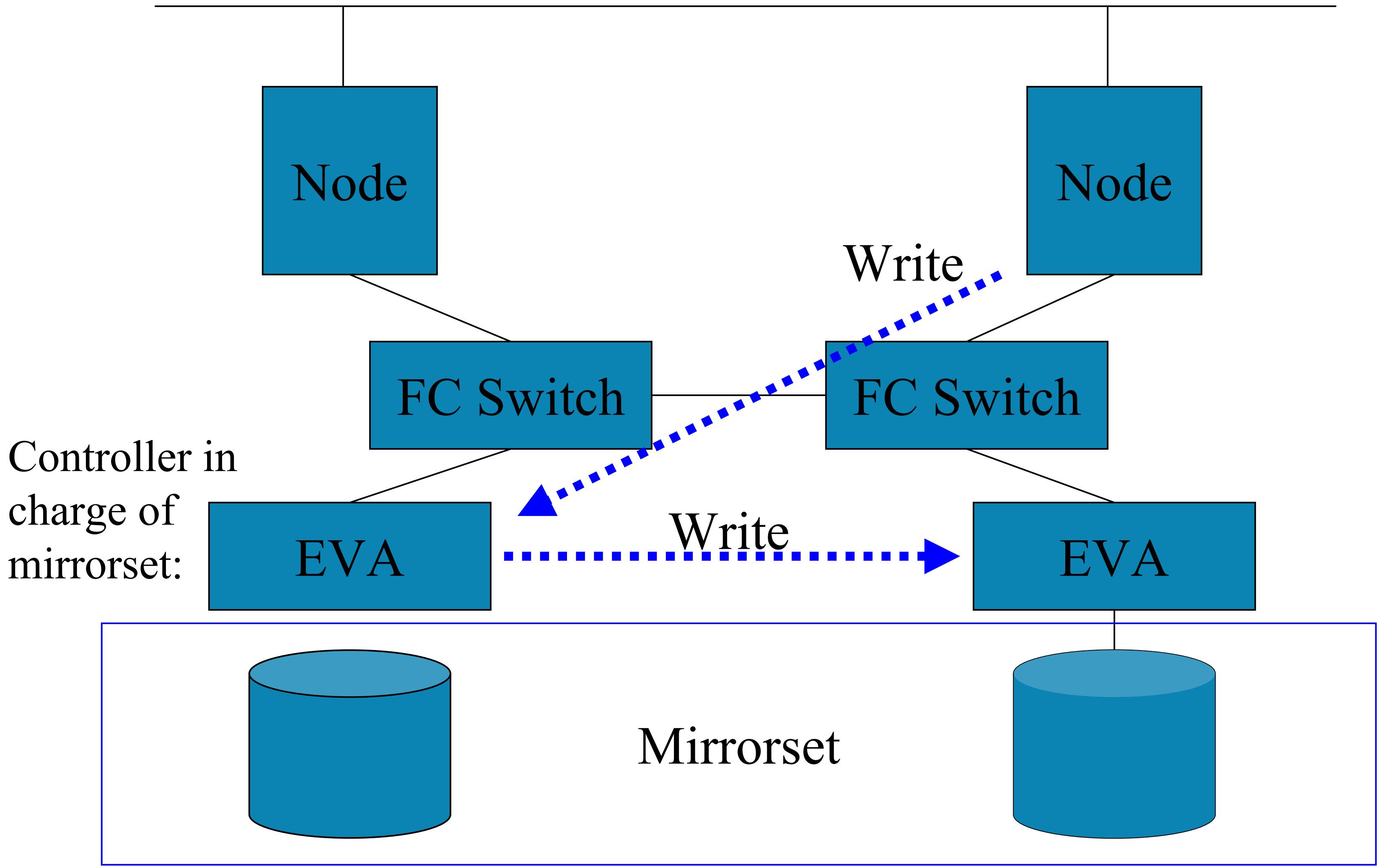
Data Replication Manager / Continuous Access

- Since the controller coordinates writes to a mirrorset:
  - Mirrorset is visible on and can only be accessed through one controller at a time

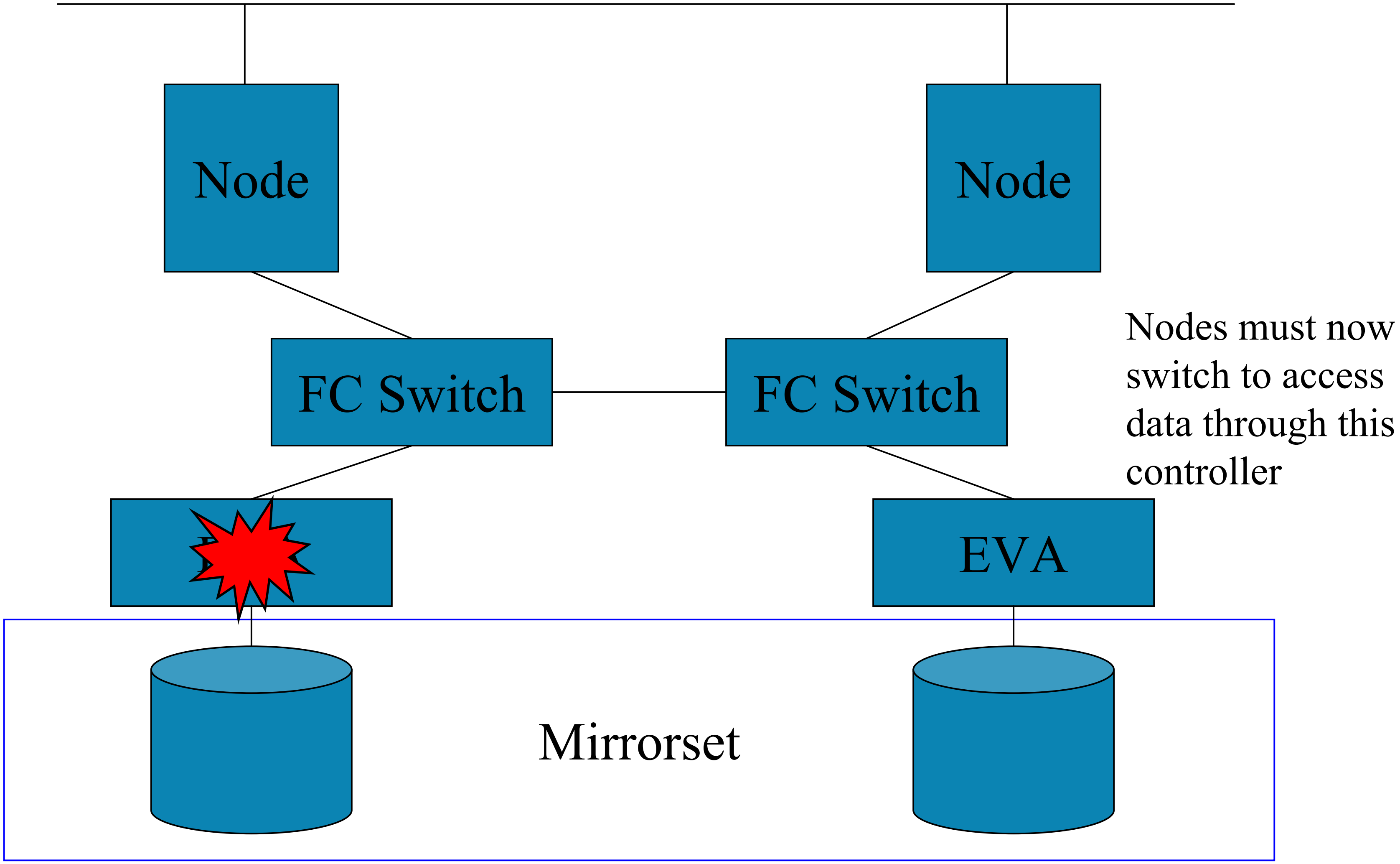
Data Replication Manager / Continuous Access

- Nodes at opposite site from controller in charge of mirrorset will have to do remote I/Os, both for writes *and for reads*
  - Storage Engineering plans the ability to access data simultaneously from both sites in a future “DRM II” implementation

Data Replication Manager / Continuous Access



Data Replication Manager / Continuous Access



Nodes must now switch to access data through this controller

Data Replication Manager / Continuous Access

- Because data can be accessed only through one controller at a time; failover is messy and involves manual or, at best, scripted changes

Achieving Very-High Availability

- In mission-critical environments, where OpenVMS Disaster-Tolerant Clusters are typically installed, there is typically a completely different mindset than at less-critical sites
- Extra effort is typically taken to ensure very-high availability
- This effort occurs across a broad range of areas

Achieving Extremely High Availability

- Configure “extra” redundancy
  - 2N redundancy instead of N+1, and 3N instead of 2N
    - i.e. shadowing instead of RAID-5
  - Layer redundancy when possible
    - e.g. shadowsets of mirrorsets
  - Recognize all potential single points of failure
    - e.g. dual-redundant controller pair tightly coupled
    - e.g. non-mirrored cache memory in controller

Achieving Extremely High Availability

- Monitor closely
  - Fix broken hardware quickly because it reduces redundancy and subsequent hardware failures can cause outages
  - Quicker mean-time-to-repair (MTTR) means better mean-time-between-failures (MTBF) in practice

Achieving Extremely High Availability

- Configure reserve capacity
  - Avoid saturation and recovery scenarios, infrequently-used code paths

Achieving Extremely High Availability

- Avoid using very-new OpenVMS versions, ECO patch kits, firmware releases, and hardware products
  - Let new OpenVMS releases, ECO kits, and products “age”
  - This lets other sites find the bugs first, and helps avoid installing an ECO kit only to find shortly that it has been put on “Engineering Hold” due to a problem

Achieving Extremely High Availability

- Allow fail-over mechanisms the freedom to work
  - Leave options open for fail-over in case they are needed:
    - e.g. Rather than leaving preferred paths set permanently down at the HSJ/HSD controller level, give OpenVMS the freedom to select the path, in anticipation of future failure conditions
    - e.g. Consider leaving 10–megabit or Fast Ethernet LAN paths enabled for SCS traffic even though Gigabit Ethernet (or FDDI) is present and larger packet size is in use; use SCACP to set at lower priority if desired
    - e.g. Load MSCP server with MSCP_LOAD=2 even on a node which you normally don’t desire to take MSCP serving load

Achieving Extremely High Availability

- Give time for hardware to mature
  - The second or third revisions of a product often have higher performance and fewer problems than the very first version (and often cost less)
- However, one aspect of “availability” is acceptable performance, so
  - You may have to trade off hardware / software maturity in favor of higher performance to handle high workload growth rates

Achieving Extremely High Availability

- Test new code and hardware (and any changes) first in a test environment separate from the production environment
  - Ideally, have a full-scale test environment which exactly duplicates the production environment, to allow testing under full (simulated) loading
- Introduce any changes into production cautiously, one node at a time

Achieving Extremely High Availability

- Introduce new technology using “generational diversity”
  - Instead of using all new technology, use a mix of old (proven) and new (fast, but unproven) technology, backing each other up in redundant configurations

Achieving Extremely High Availability

- Consider using multiple clusters
  - One approach:
    - Partition data among clusters
    - Provide in-bound user routing across clusters to where their data resides
    - Allow user data to be migrated between clusters on-the-fly, transparent to the user
    - Failure of any single cluster affects only a fraction of the users
    - Partial failure of any cluster can be mitigated by migrating its users to another cluster
  - Cleanest approach to multiple clusters:
    - Design with RTR in mind from the beginning

Achieving Extremely High Availability

- Do cluster configuration and application design or re-design with availability in mind:
  - On-line backups
  - Disk defragmentation
  - Indexed file reorganization

Business Continuity

- Although we've been talking about tolerating disasters in the IT area, true ability to survive a disaster involves more than using a disaster-tolerant cluster in the IT department
- The goal of *Business Continuity* is the ability for the entire business, not just IT, to continue operating despite a disaster

Business Continuity: Not just IT

- Not just computers and data:
  - People
  - Facilities
  - Communications
    - Networks
    - Telecommunications
  - Transportation

Business Continuity Resources

- Disaster Recovery Journal:
  - http://www.drj.com/
- Contingency Planning & Management Magazine
  - http://www.contingencyplanning.com/

- Both are free to qualified subscribers
- Both hold conferences as well as publishing high-quality journals

Real-Life Examples: Credit Lyonnais

- Credit Lyonnais fire in Paris, May 1996
- Data replication to a remote site saved the data
- Fire occurred over a weekend, and DR site plus quick procurement of replacement hardware allowed bank to reopen on Monday

Real-Life Examples: Online Stock Brokerage

- 2 a.m. on 29 December, 1999, an active stock market trading day
- Just 3 days before Y2K
  - Media were watching like hawks to detect any system outages that might be related to inadequate Y2K preparation
  - Customers fearing inadequate Y2K preparation would likely pull their money out in a hurry
- UPS Audio Alert alarmed security guard on his first day on the job, who pressed emergency power-off switch, taking down the entire datacenter

Real-Life Examples: Online Stock Brokerage

- Disaster-tolerant cluster continued to run at opposite site; no disruption
- Ran through that trading day on one site alone
- Performed shadow copies to restore data redundancy in the evening after trading hours
- Procured a replacement for the failed security guard by the next day

Real-Life Examples: Commerzbank on 9/11

- Datacenter near WTC towers
- Generators took over after power failure, but dust & debris eventually caused A/C units to fail
- Data replicated to remote site 30 miles away
- One AlphaServer continued to run despite 40° C temperatures, running off the copy of the data at the opposite site after the local disk drives had succumbed to the heat
- See http://h71000.www7.hp.com/openvms/brochures/commerzbank/

Real-Life Examples: Online Brokerage

- Dual inter-site links
  - From completely different vendors
- Both vendors sub-contracted to same local RBOC for local connections at both sites
- Result: One simultaneous failure of both links within a 4 year time period

Real-Life Examples: Online Brokerage

- Dual inter-site links from different vendors
- Both used fiber optic cables across the same highway bridge
  - El Niño caused flood which washed out bridge
- Vendors' SONET rings wrapped around the failure, but latency skyrocketed and cluster performance suffered

Real-Life Examples: Online Brokerage

- Vendor provided redundant storage controller hardware
  - Despite redundancy, a controller pair failed, preventing access to the data behind the controllers
- Host-based volume shadowing was in use, and the cluster continued to run using the copy of the data at the opposite site

Real-Life Examples: Online Brokerage

- Dual inter-site links from different vendors
  - Both vendors’ links did fail sometimes
- Redundancy and automatic failover masks failures
  - Monitoring is crucial
    - One link outage lasted 6 days before discovery

Speaker Contact Info:

- Keith Parris

- E-mail: parris@encompasserve.org
- or keithparris@yahoo.com
- or Keith.Parris@hp.com
- Web: http://encompasserve.org/~parris/
- and http://www.geocities.com/keithparris/
- and http://www2.openvms.org/kparris/

HP WORLD 2003

Solutions and Technology Conference & Expo

Interex, Encompass and HP bring you a powerful new HP World.

interex
shared knowledge • shared power

encompass
AN HP USER GROUP

hp
invent