Reaching MoU Targets at Tier0

December 20th 2005

Tim Bell
IT/FIO/TSI

How to do it

- Choose fail-safe hardware
- Have ultra-reliable networking
- Write bug-free programs
- Use administrators who never make mistakes
- Find users who read the documentation

Agenda

- MoU Levels
- Procedures
- High Availability approaches

LCG Services Class

Class	Description	Downtime	Reduced	Degraded	Avail
C	Critical	1 hour	1 hour	4 hours	99%
H	High	4 hours	6 hours	6 hours	99%
M	Medium	6 hours	6 hours	12 hours	99%
L	Low	12 hours	24 hours	48 hours	98%
U	Unmanaged	None	None	None	None

- Ref: https://uimon.cern.ch/twiki/bin/view/LCG/ScFourServiceDefinition
- Defines availability rather than raw performance metrics

Downtime from a failure

Failure Occurs	Something breaks
Failure Detected	Latencies due to polling of status
Failure Noticed	Console, E-Mail, Siren,…
Investigation Started	Login, have a look
Problem Identified	Root cause found
Procedure Found	How to solve it
Problem Solved	Execute the procedure
Restore Production	Cleanup

MoU is not very ambitious

- 99% uptime
  - 1.7 hours / week down
  - 4 days / year down
- Does not cover impact of failure
  - Lost jobs / Recovery / Retries
  - Problem Analysis
  - Glitch effects
- Core services have domino effects
  - MyProxy, VOMS, SRMs, Network
- User Availability is sum of dependencies
  - FTS, RB, CE

Coverage

- Standard availability does not cover
  - Weekends
  - Night time
- Working Time = 40 hours / week = 24%
- Dead time
  - Meetings / Workshops
  - No checks before morning status reviews and coffee
  - Illness / Holidays
- Response Time (assuming available)
  - If on site, < 5 minutes
  - If at home and access sufficient, < 30 minutes
  - If on-site required, ~ 1 hour ?

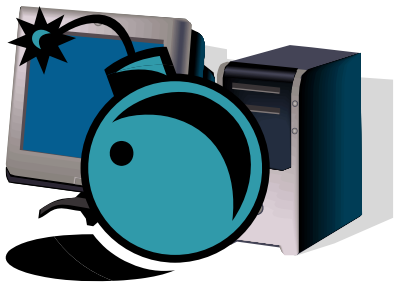
Changes

- New release needed rapidly
  - Security patches
  - Interface changes
- Slow quiesce time to drain
  - 1 week for jobs to complete
  - 1 week proxy lifetime
- Many applications do not provide drain or migrate functionality
  - Continue to serve existing requests
  - Do not accept new requests

How to Reconcile

- People and Procedures
  - Call trees and on-call presence coverage
  - Defined activities for available skills
- Technical
  - Good quality hardware
  - High availability
  - Degraded services

People and Procedures - Bottom Up



Application Specialist

Sysadmin on Call

Lemon Alerts

People and Procedures

- Alerting
  - 24x7 Operator receives problem from Lemon
  - Follows per-alert procedure to fix or identify correct next level contact
- SysAdmin / Fabric Services
  - 24x7 for more complex procedures
- Application Expert
  - As defined by the grid support structure

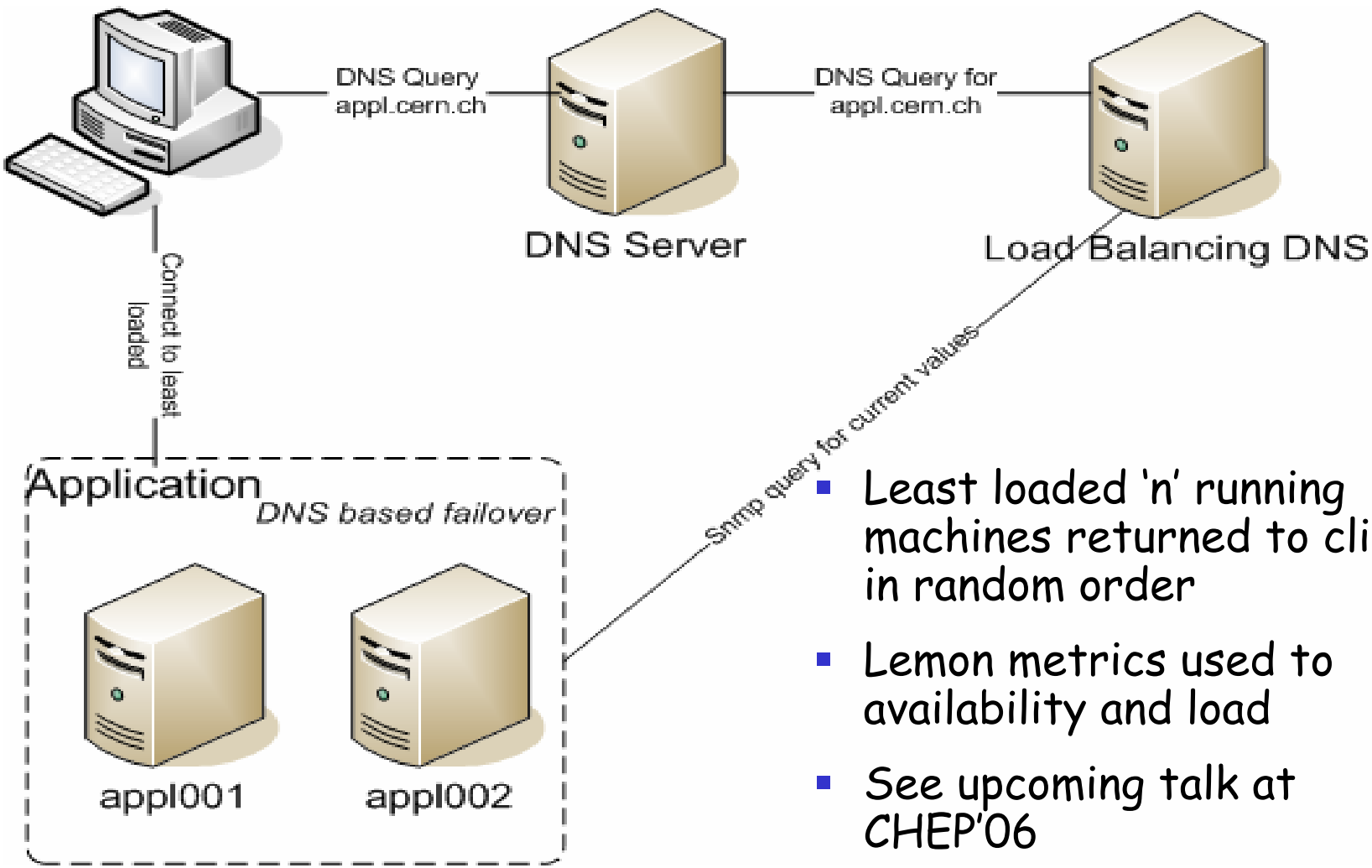
Technical Building Blocks

- Minimal Hardware for Servers
- Load Balancing
- RAC Databases
- High Availability Toolkits
- Cluster File Systems

Server Hardware Setup

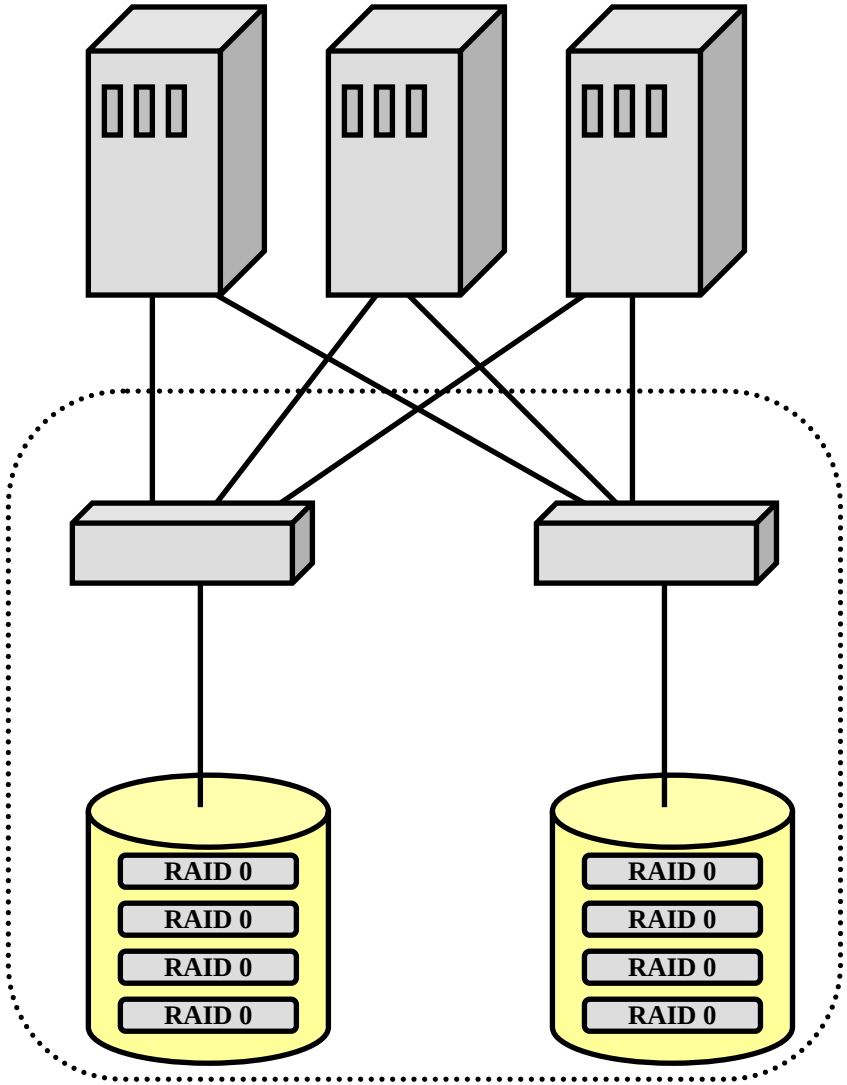
- Minimal Standards
  - Rack mounted
  - Redundant power supplies
  - RAID on system and data disks
  - Console access
  - UPS
  - Physical access control
- Batch worker nodes do not qualify even if they are readily available

Load Balancing Grid Applications



- Least loaded 'n' running machines returned to client in random order
- Lemon metrics used to availability and load
- See upcoming talk at CHEP'06

State Databases

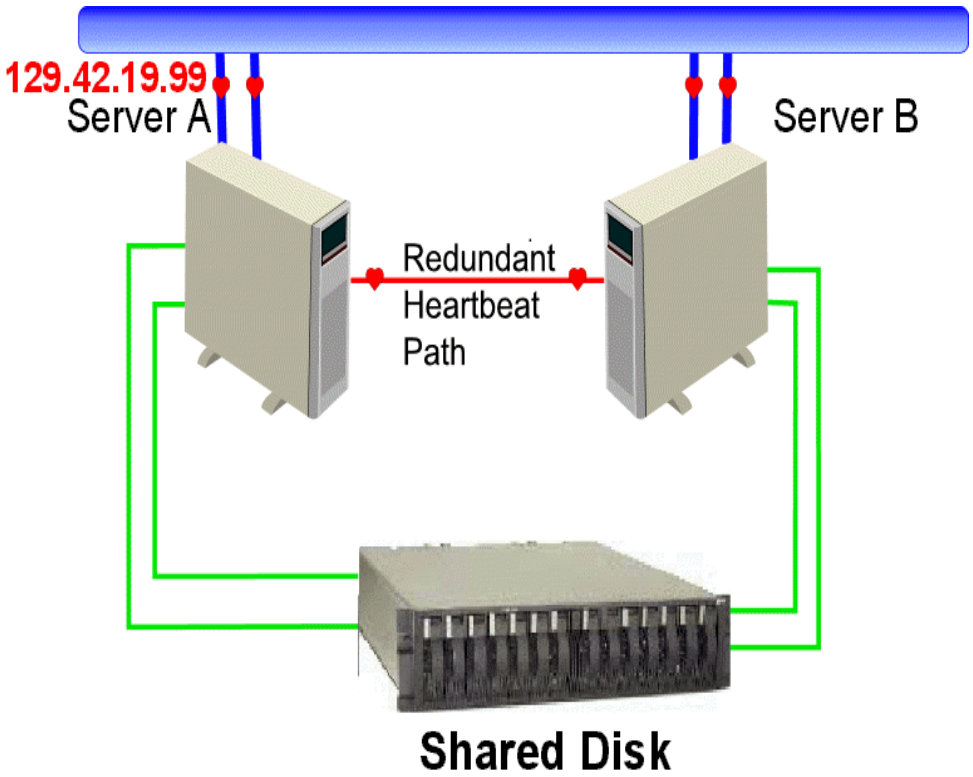


- Oracle RAC configuration with no single points of failure
- Used for all grid applications which can support Oracle
- Allows stateless load balanced application servers
- It really works ☺

High Availability Toolkits

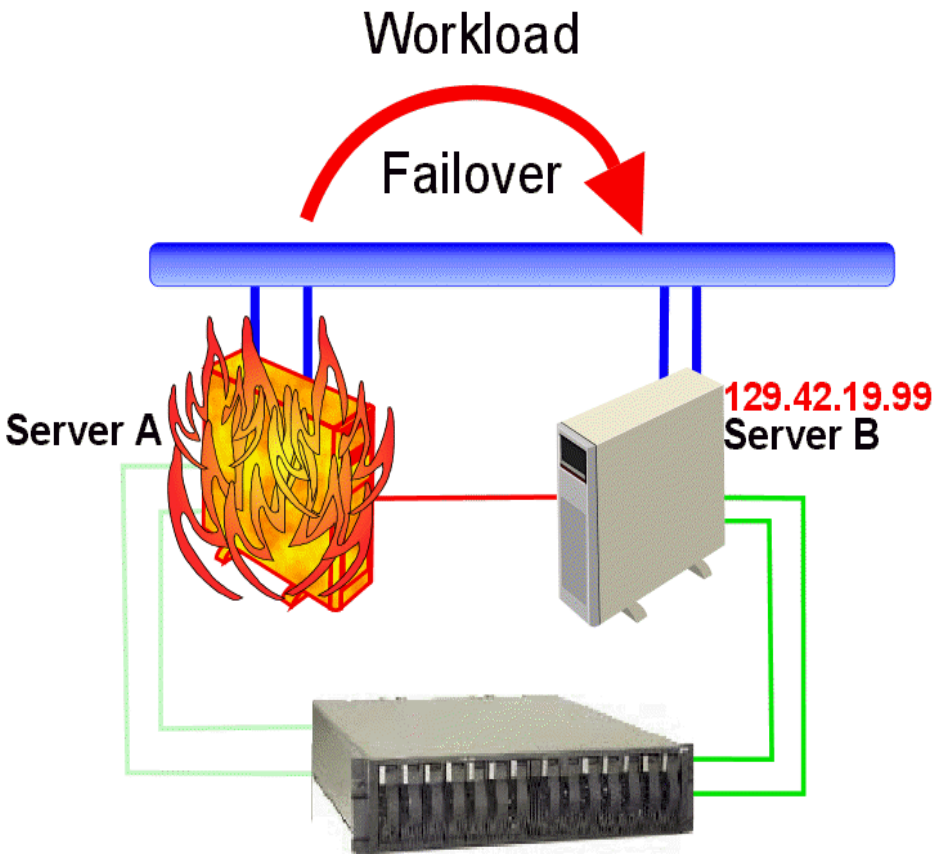
- FIO is using Linux-HA
  - http://www.linux-ha.org/ running at 100s of sites on Linux, Solaris and BSD.
- Switch when
  - Service goes down
  - Administrator request
- Switch with
  - IP Address of master machine
  - Shared disk (requires Fibre Channel)
  - Application specific procedures

Typical Configuration with HA



- Redundancy eliminates Single Points Of Failure (SPOF)
- Monitoring determines when things need to change
- Can be administrator initiated for planned changes

Failure Scenario with HA

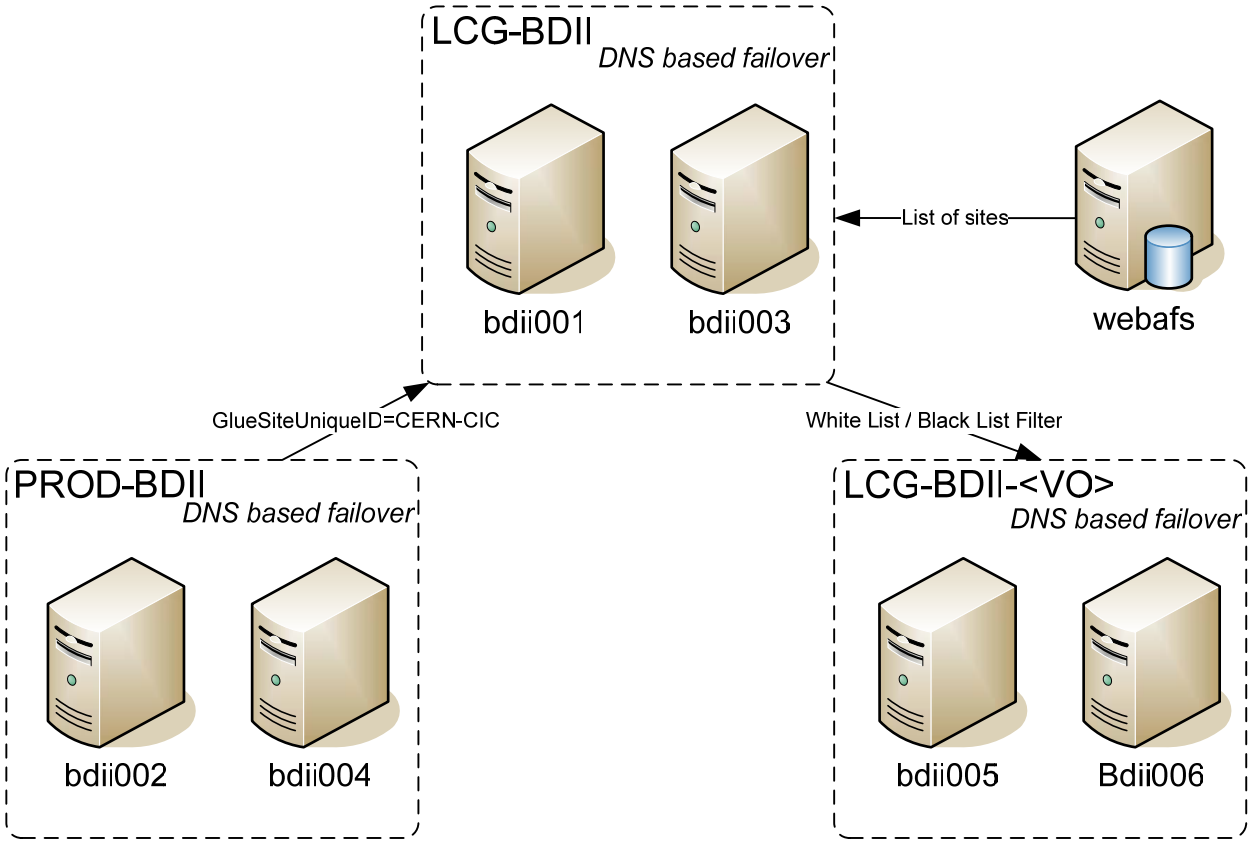


- Monitoring detects failures (hardware, network, applications)
- Automatic Recovery from failures (no human intervention)
- Managed restart or failover to standby systems, components

Cluster File Systems

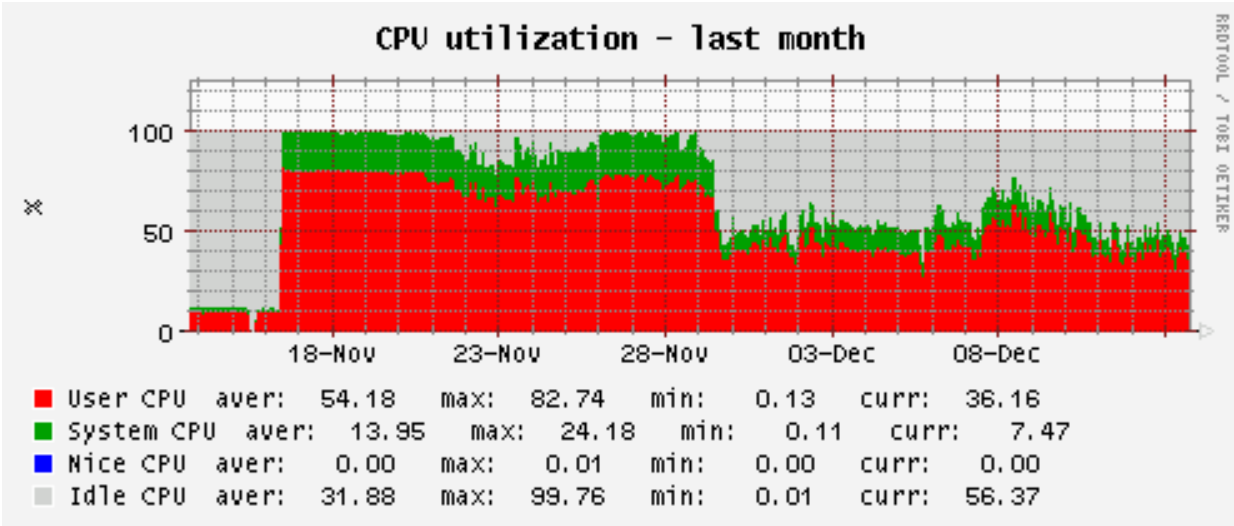
- NFS does not work in production conditions under load
- FIO has tested 7 different cluster file systems to try to identify a good shared highly available file system
- Basic tests (disconnect servers, kill disks) show instability or corruption
- No silver bullet as all solutions are immature in the high availability area
- Therefore, we try to avoid any shared file systems in the CERN grid environment

BDII



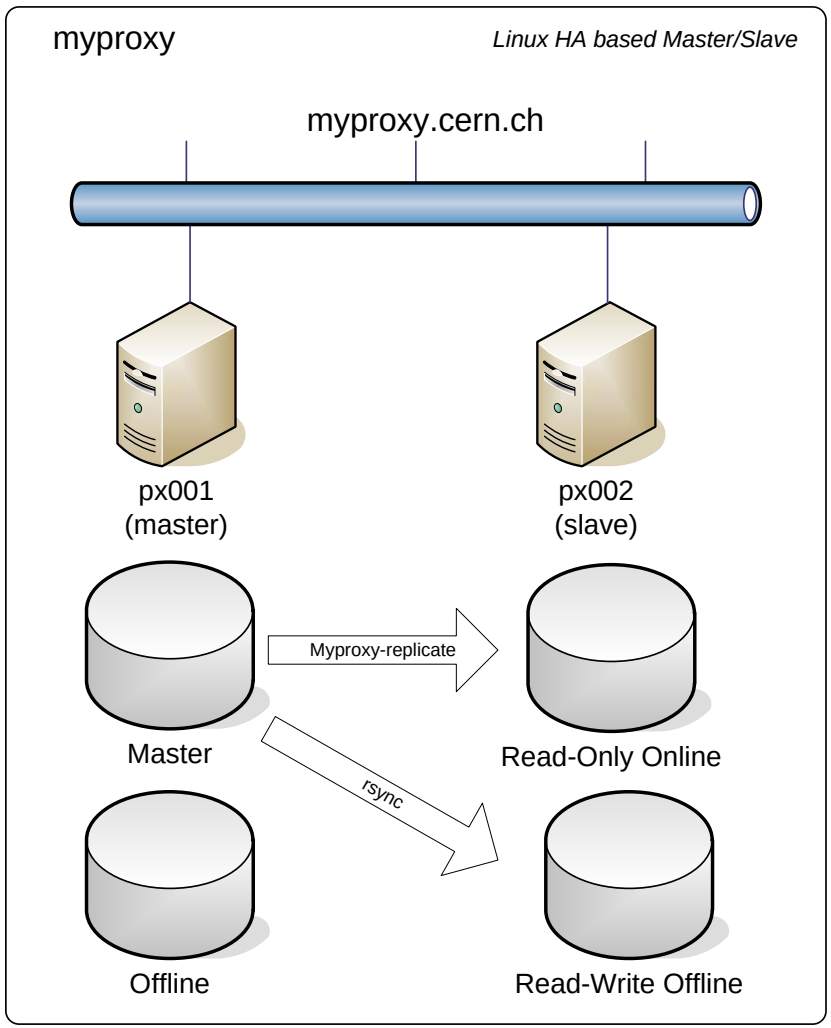
- BDII is easy since the only state data is the list of sites
- Load Balancing based on Lemon sensor which checks the longitude/latitude of CERN
- Lemon monitoring of current load based on number of LDAP searches

BDII Lemon Monitoring



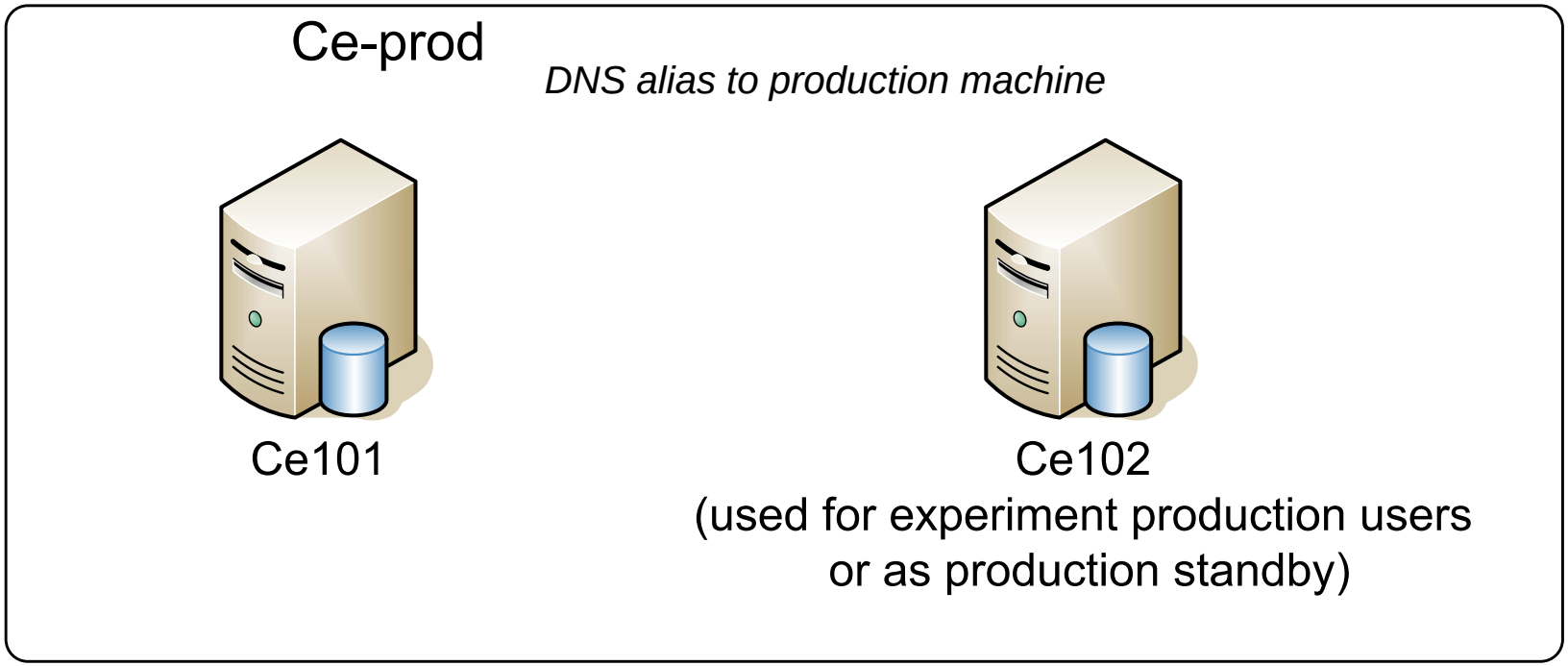
- New machine started production mid November
- Load Balancing turned on at the end November

MyProxy



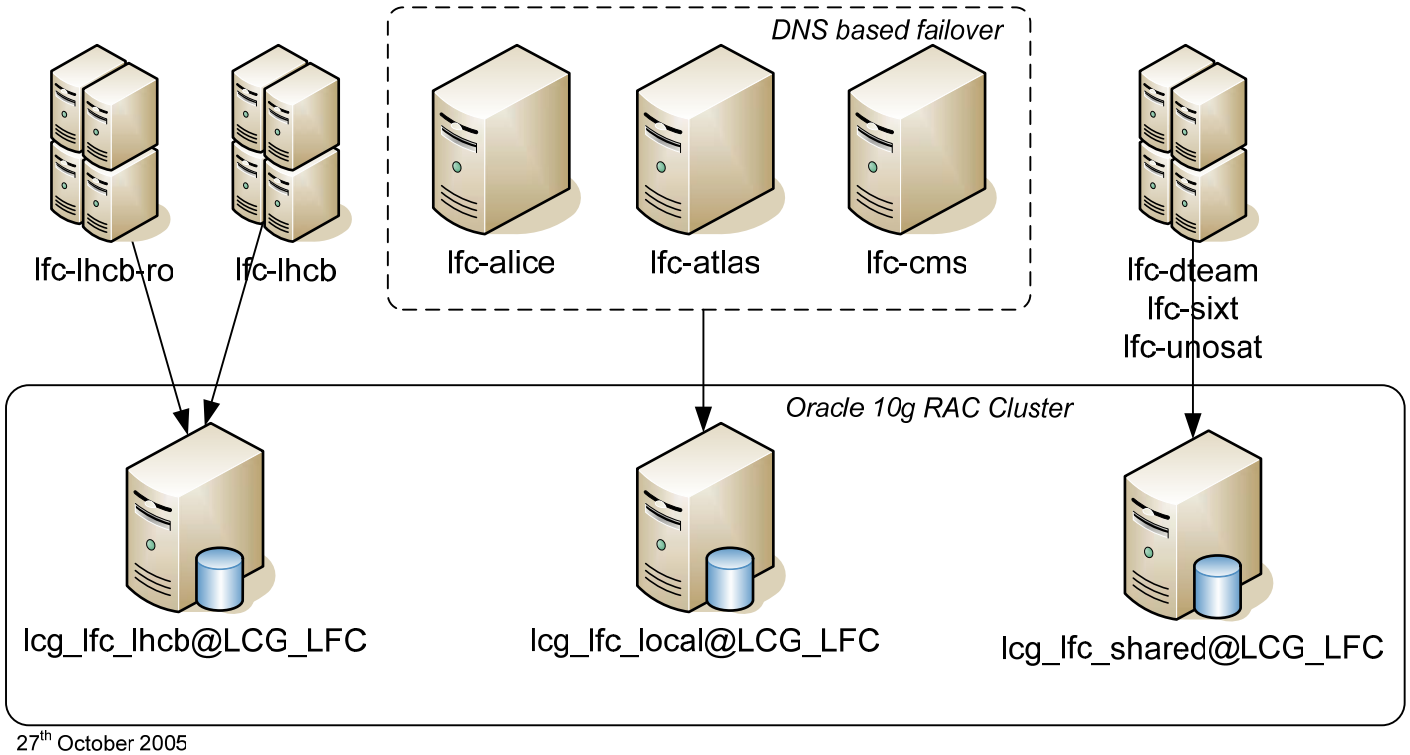
- MyProxy has a replication function to create a slave server
- Slave server is only read-only for proxy retrieval
- Second copy made at regular intervals in case of server failure
- TCP/IP network alias switched by Linux-HA in the event of the master proxy server going down
- Slave monitors the master to check all is running ok

RBs and CEs – No HA solution



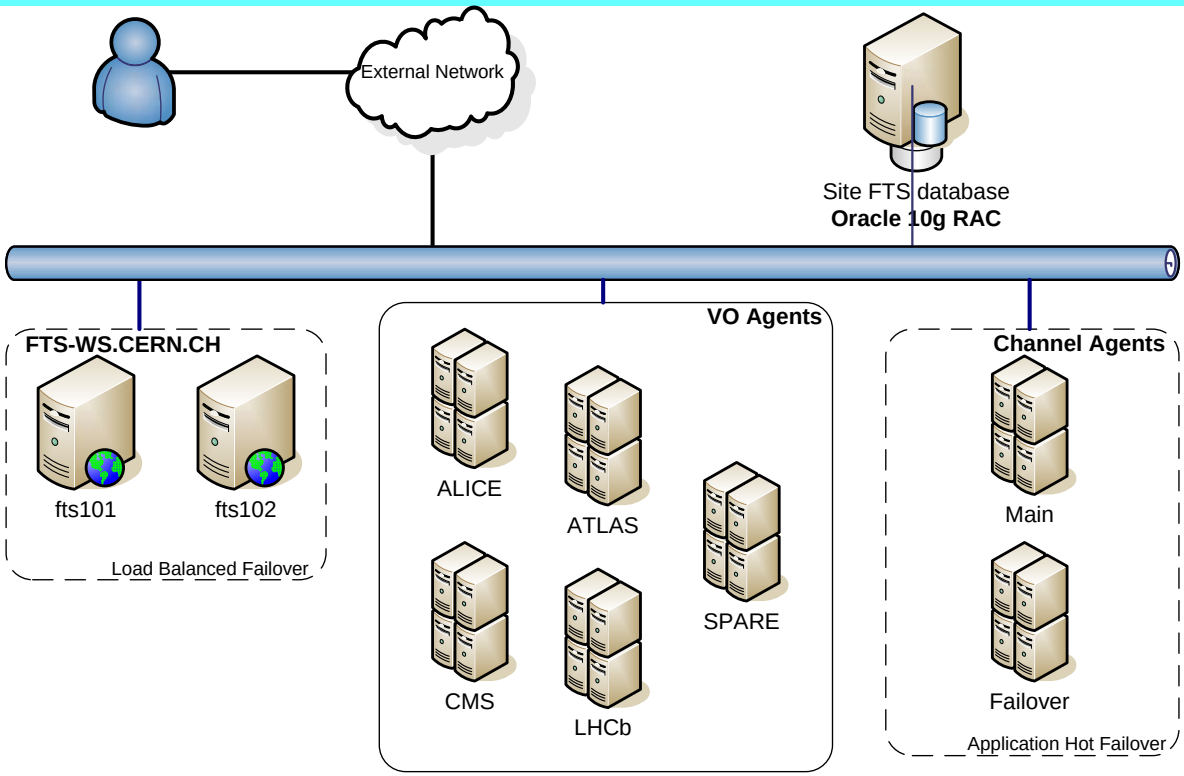
- Currently no High Availability solution as state data is on local file system
- Plan to run two machines with manual switch over using an IP alias
- 2nd machine can be used by production super-users when 1st machine is running ok
- Could consider shared disk solution with standby machine
- Drain time is around 1 week

LFC



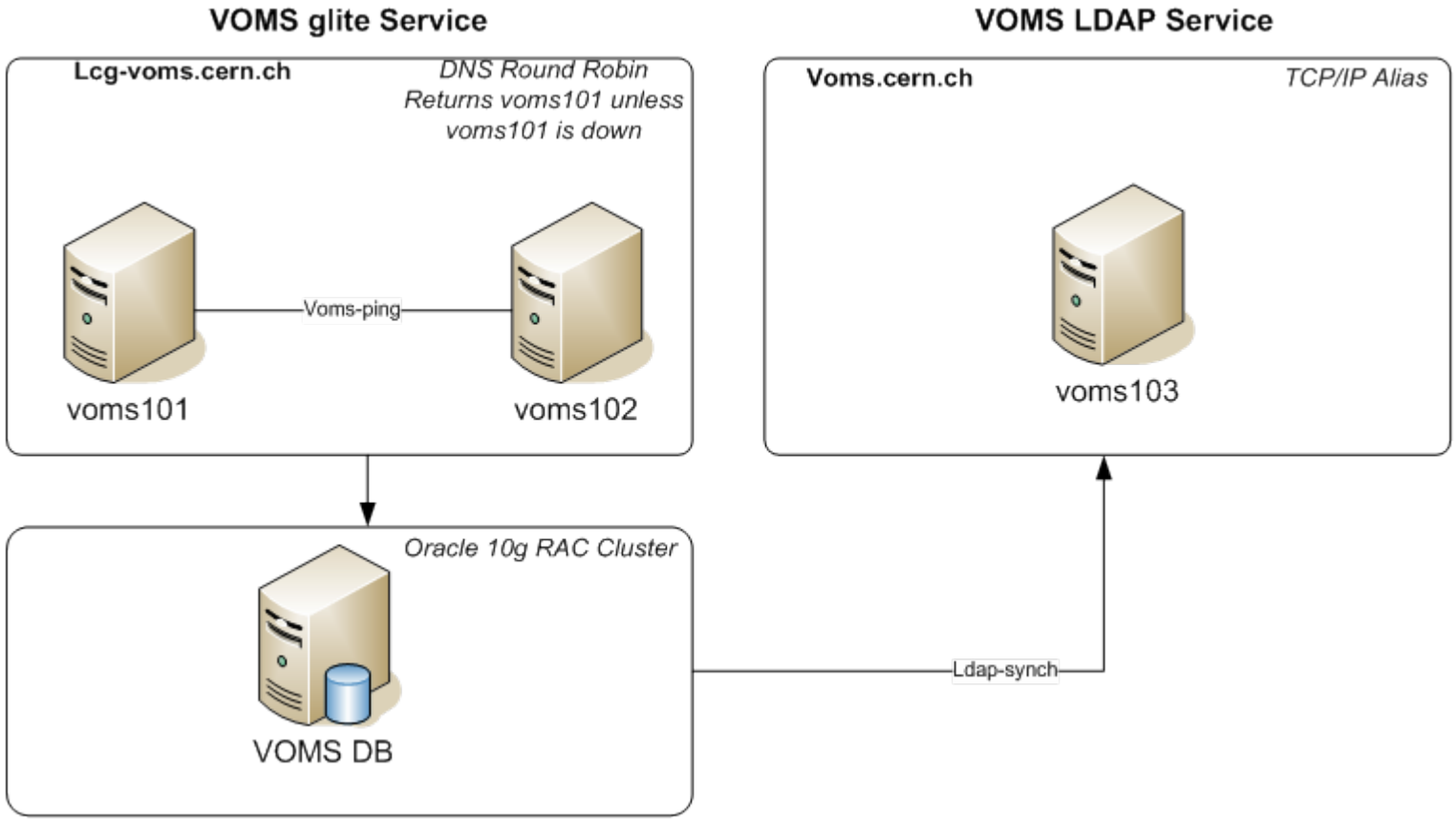
- Application front ends are stateless
- RAC databases provide state data

FTS



- Load Balanced front end
- Agents are warm, becoming hot

VOMS



- VOMS gLite is highly available front end using DNS load balancing. Slave reports itself as very low priority compared to master for log stability
- LDAP access is to be reduced so less critical

Summary of Approaches

- Highly available service using HA toolkits / Oracle RAC – single failure is covered by switch to alternative system
- VO based services with spares – single failure may cause one VO to lose function but other VOs remain up
- File system based stateful services problematic - Need
  - Cluster file system or
  - Application re-architecting
  - User acceptance of increased time to recover / manual intervention

Other Applications

- Only critical and high products considered for high availability so far
- Others may be worth considering
  - SFT, GridView, GridPeek
  - R-GMA, MonBox

Current Status

- BDIIs now in production with procedures in place
- MyProxy, CEs nearing completion of automatic software installation and setup
- FTS, LFC, VOMS, GridView hardware ready
- RB not there yet

Conclusions

- Adding High Availability is difficult but sometimes possible at fabric level
- Applications need to be designed with availability in mind (FTS, LFC are good examples of this)
- Planned changes are more frequent than hardware failures. Change automation reduces impact
- Procedures and problem determination guides to minimise downtime