High Availability with Linux Using DRBD and Heartbeat

- short introduction to linux high availability
- description of problem and solution possibilities
- linux tools
  - heartbeat
  - drbd
  - mon
- implementation at GSI
- experiences during test operation

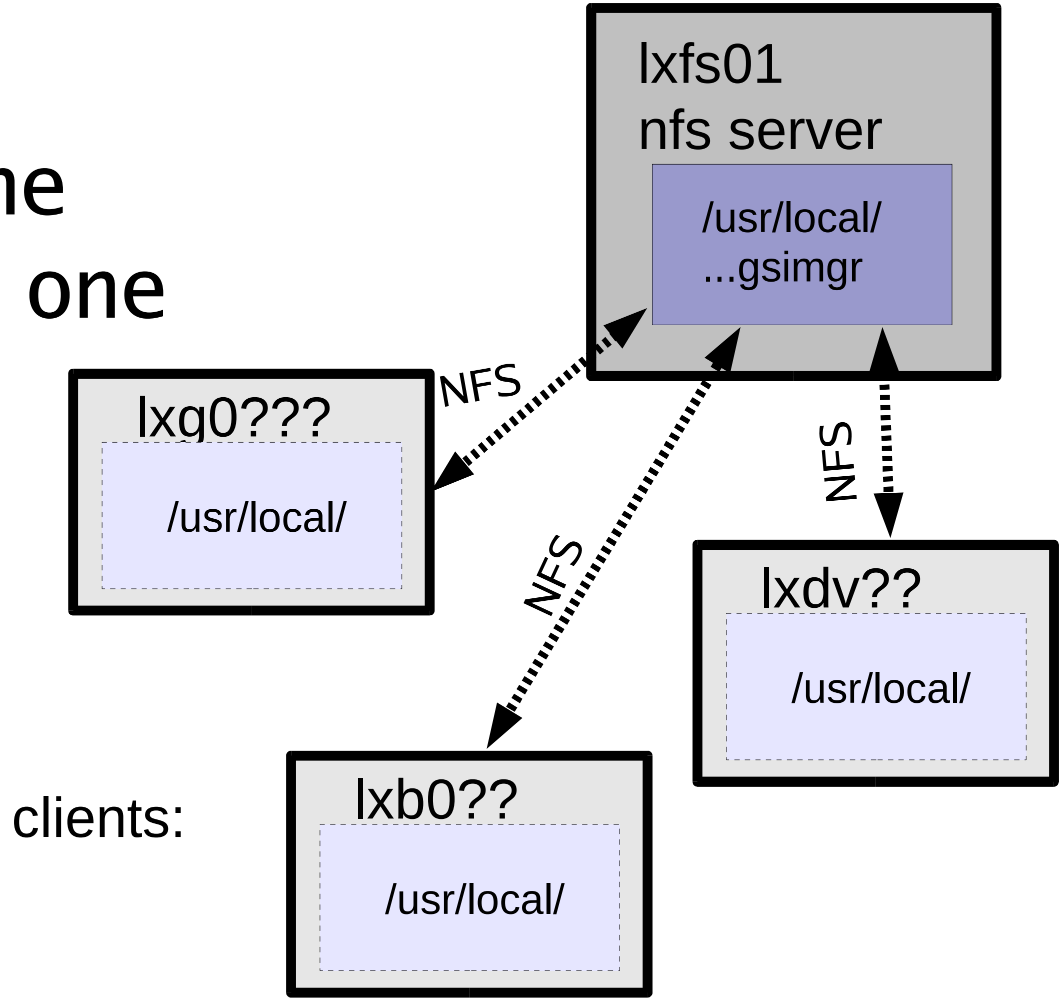
High Availability

- reduction of downtime of critical services (name service, file service ...)
- Hot Standby - automatical failover
- Cold Standby - exchange of hardware
- reliable / special hardware components (shared storage, redundant power supply...)
- special software, commercial and Open Source (FailSafe, LifeKeeper/Steeleye Inc., heartbeat ...)

Problem

central NFS service and administration:

- all linux clients mount the directory /usr/local from one central server
- central administration including scripts, config files ...



In Case of Failure...

if the central nfs server is down:

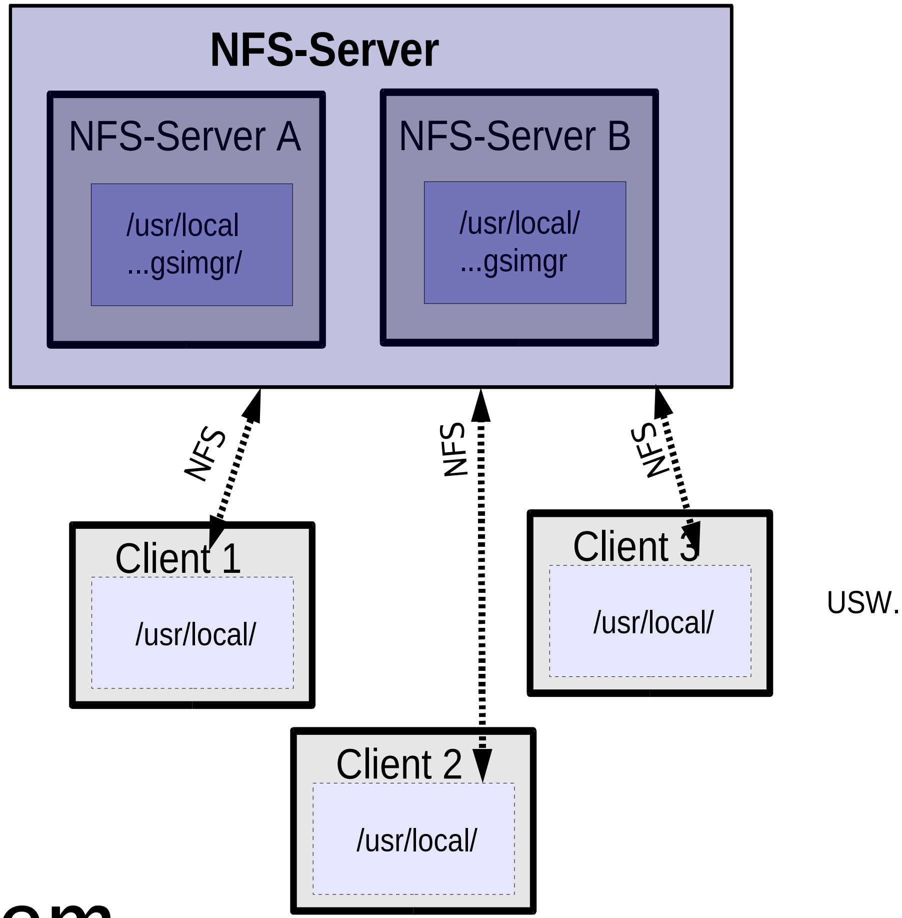
- no access of /usr/local
- most clients cannot work anymore
- administration tasks are delayed or hang

after work continues:

- stale nfs mounts

Solution

hot-standby / shared nothing:
2 identical servers with
individual storage
(instead of shared storage)



---> advantage:

- /usr/local exists twice

---> problems:

- synchronisation of file system
- information about nfs mounts

Linux Tools

heartbeat

- communication between the two nodes
- starts the services

drbd

- synchronisation of the file system (/usr/local)

mon

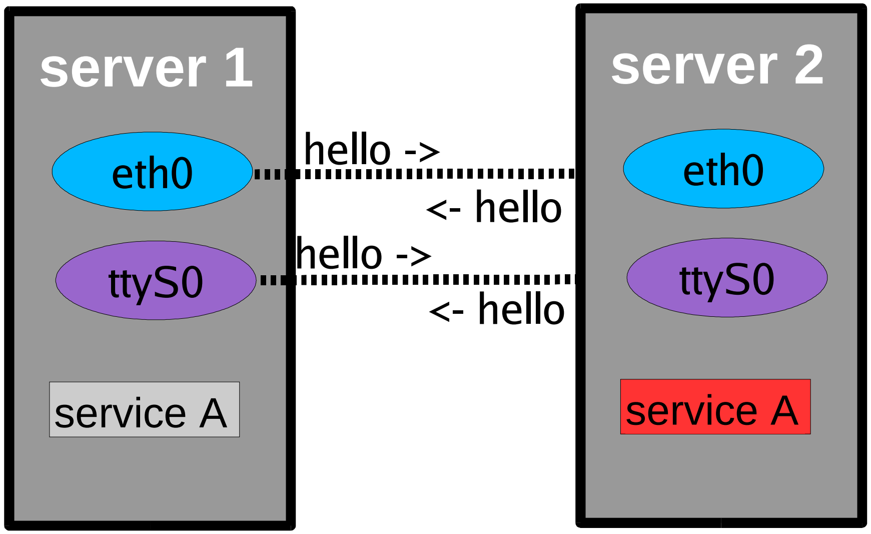
- system monitoring

all tools are OpenSource, GPL or similar

Heartbeat

- how does the slave server knows that the master node is dead?
- both nodes are connected by ethernet or serial line
- both nodes exchange pings in regular time intervals
- if all pings are missing for a certain dead time the slave assumes that the master failed
- slave takes over the IP and starts the service

Heartbeat



normal operation:

server 2 - master for service A

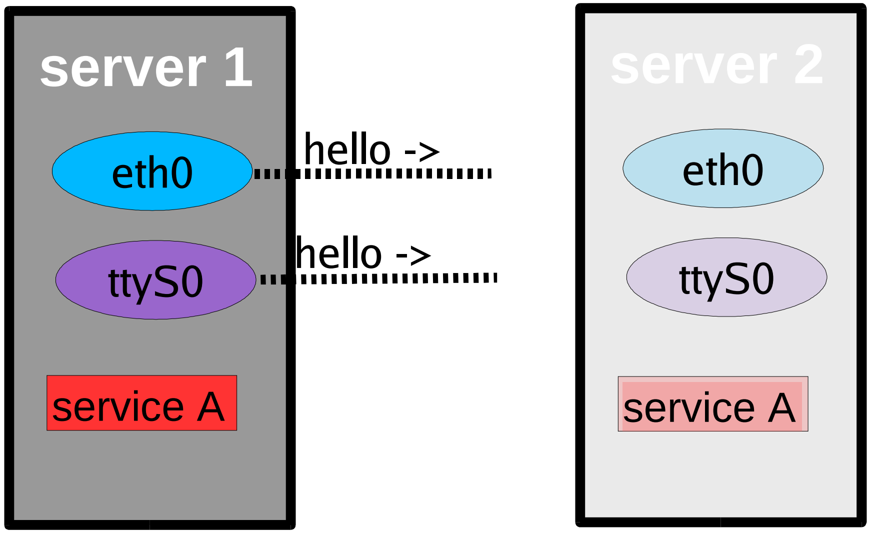
server 1 - slave for service A

failure:

server 2 fails

heartbeat-ping stops

server 1 takes over service A



Heartbeat Problems

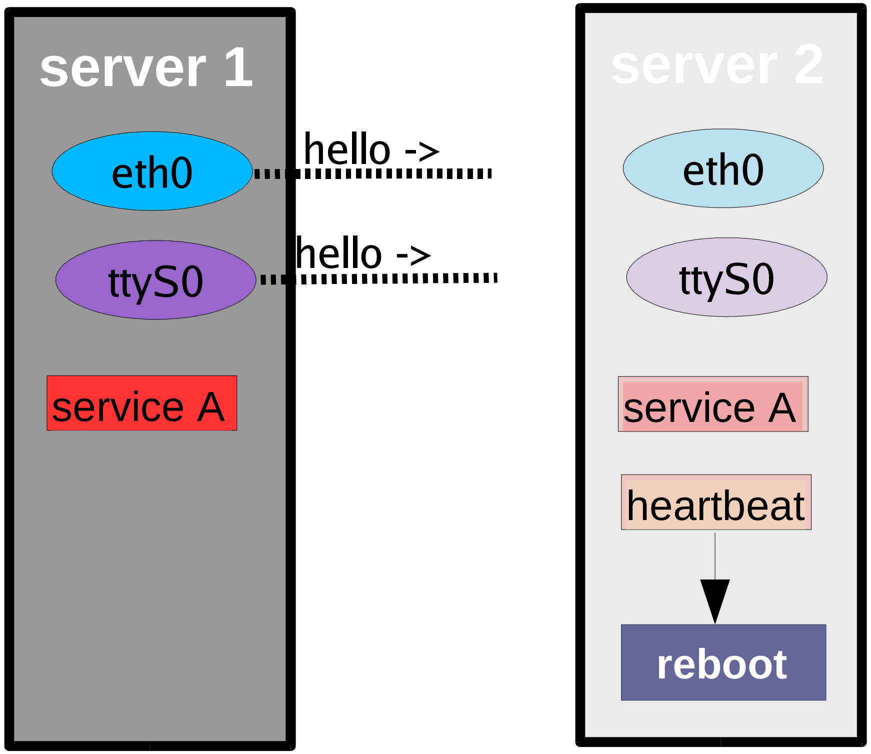
- heartbeat only checks whether the other node replies to ping
- heartbeat does not investigate the operability of the services
- even if ping works, the service could be down
- heartbeat could fail, but the services still run

To reduce this problems:

- special heartbeat features stonith, watchdog and monitoring

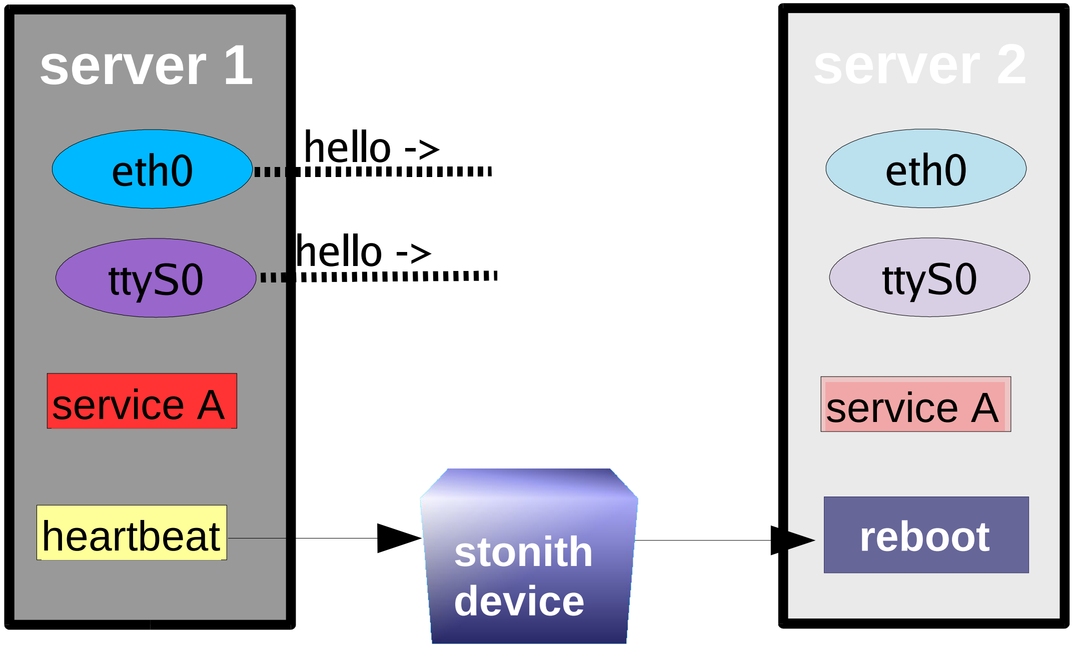
Watchdog

- special heartbeat feature - system reboots as soon as the own “heartbeat” stops



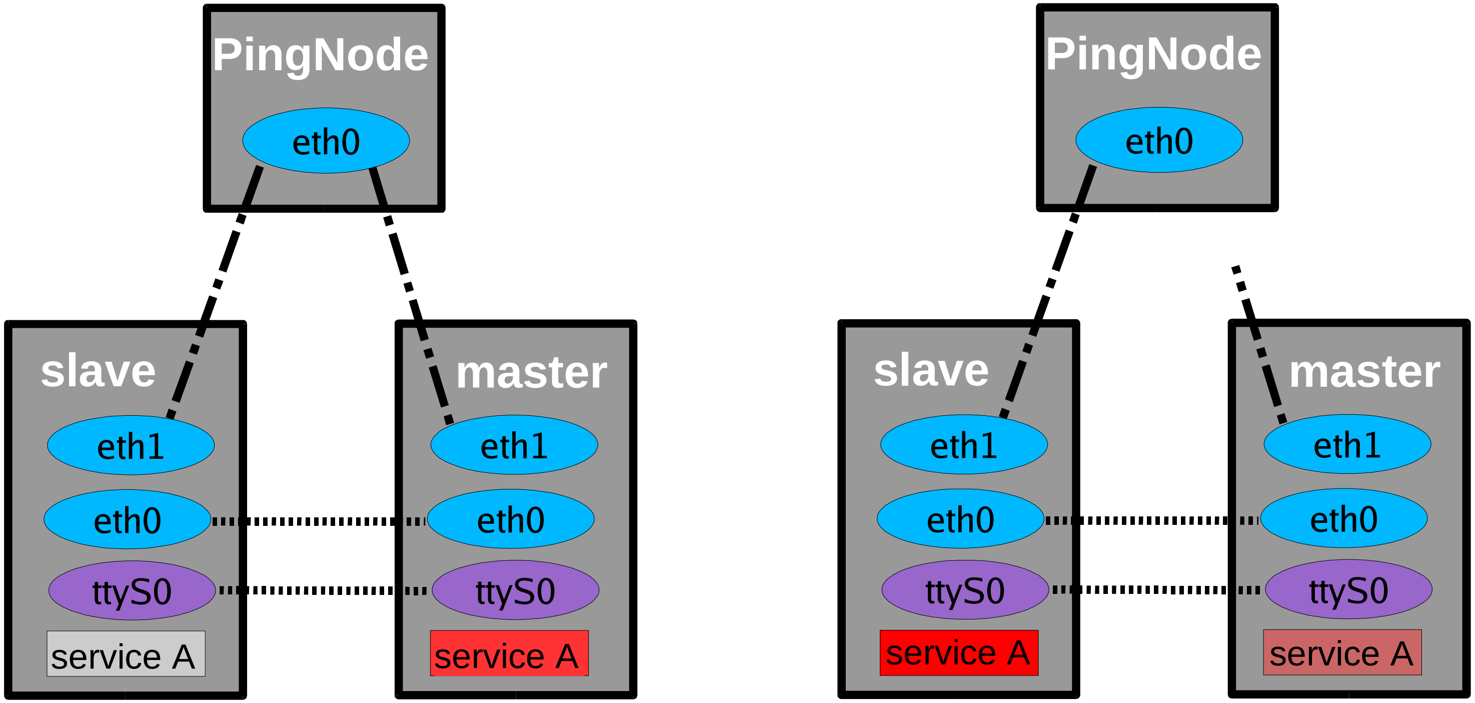
Stonith

- “Shoot the other Node in the Head” - in case a failover happens the slave triggers a reboot of the master node using ssh or special hardware (remotely controlled power switch)



Network Connectivity Check

- ipfail - checks the network connectivity to a certain PingNode
- if the PingNode cannot be reached service is switched to the slave

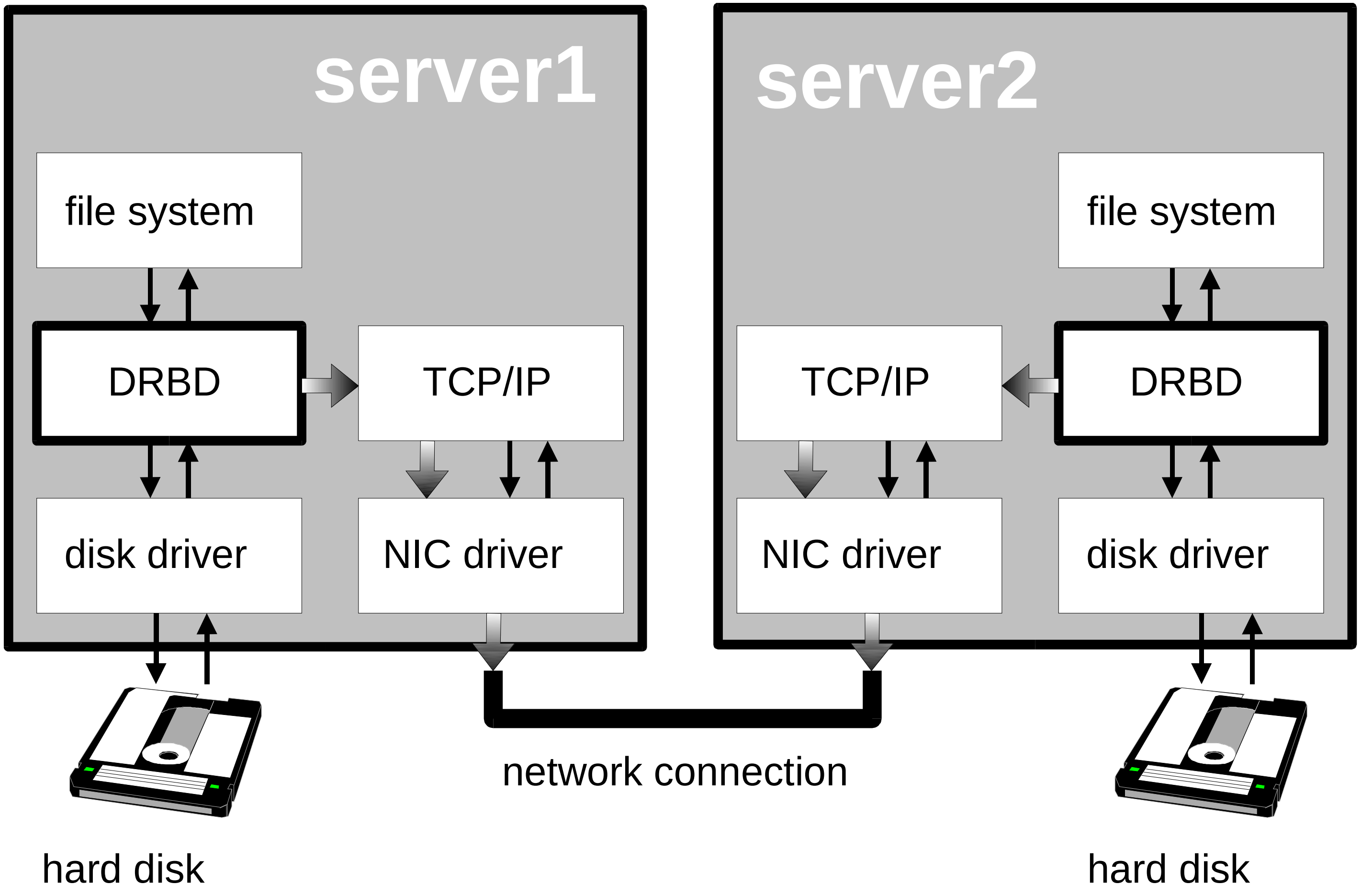


DRDB

- **D**istributed **R**eplicated **B**lock **D**evice
- kernel patch which forms a layer between block device (hard disc) and file system
- over this layer the partitions are mirrored over a network connection
- in principle:

  ➔ RAID-1 over network

DRBD - How it Works



Write Protocols

protocol A:

- write IO is reported as completed, if it has reached local disk and local TCP send buffer

protocol B:

- write IO is reported as completed, if it has reached local disk and remote buffer cache

protocol C:

- write IO is reported as completed, if it has reached both local and remote disk

(Dis-)Advantages of DRBD

- data exist twice
- real time update on slave (--> in opposite to rsync)
- consistency guaranteed by drbd: data access only on master - no load balancing
- fast recovery after failover

overhead of drbd:

  - needs cpu power
  - write performance is reduced (but does not affect read perfomance)

System Monitoring with Mon

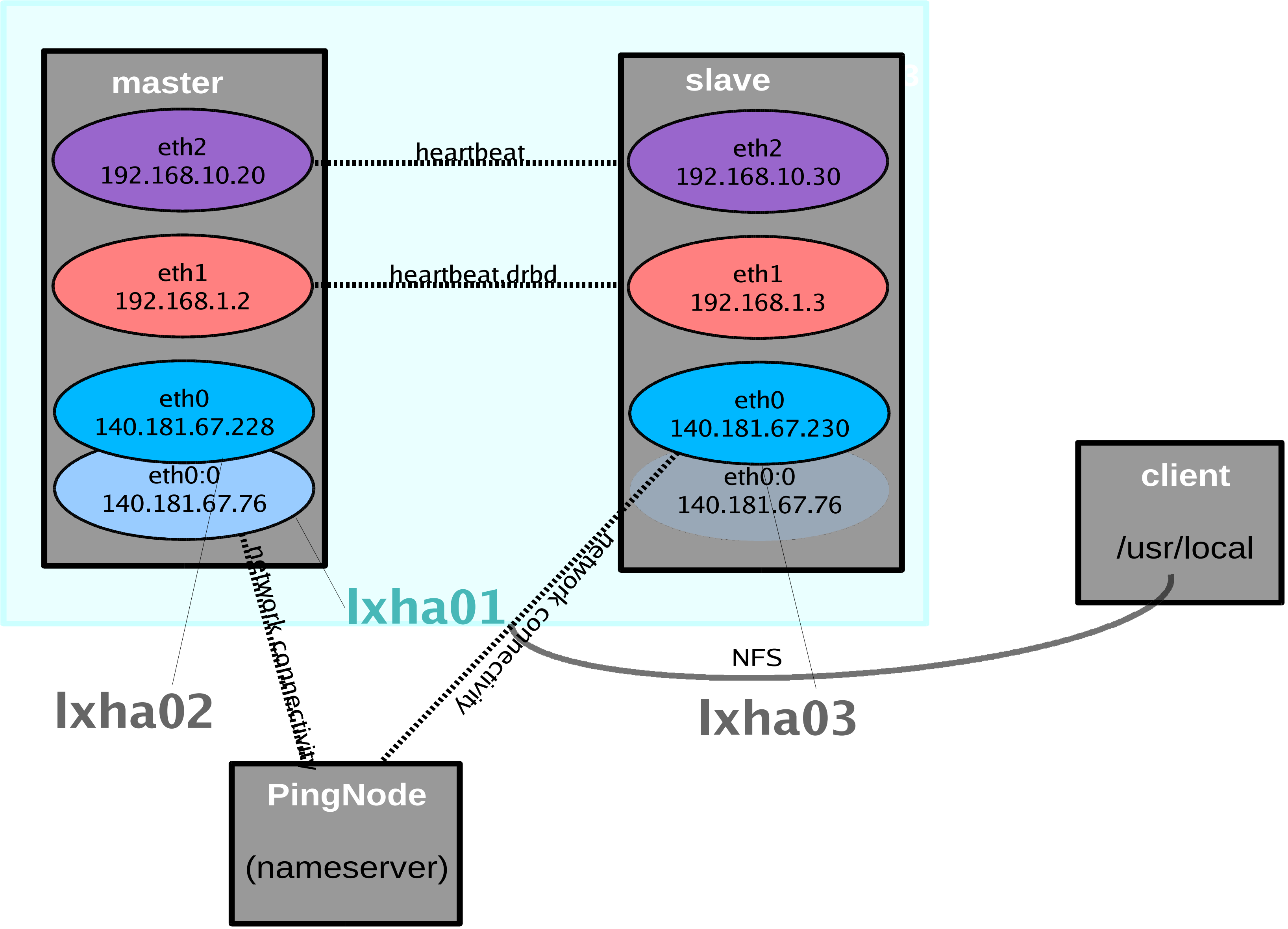
service monitoring daemon:

- monitoring of resources,network, server problems
- monitoring is done with individual scripts
- in case of failure mon triggers an action (e-mail, reboot...

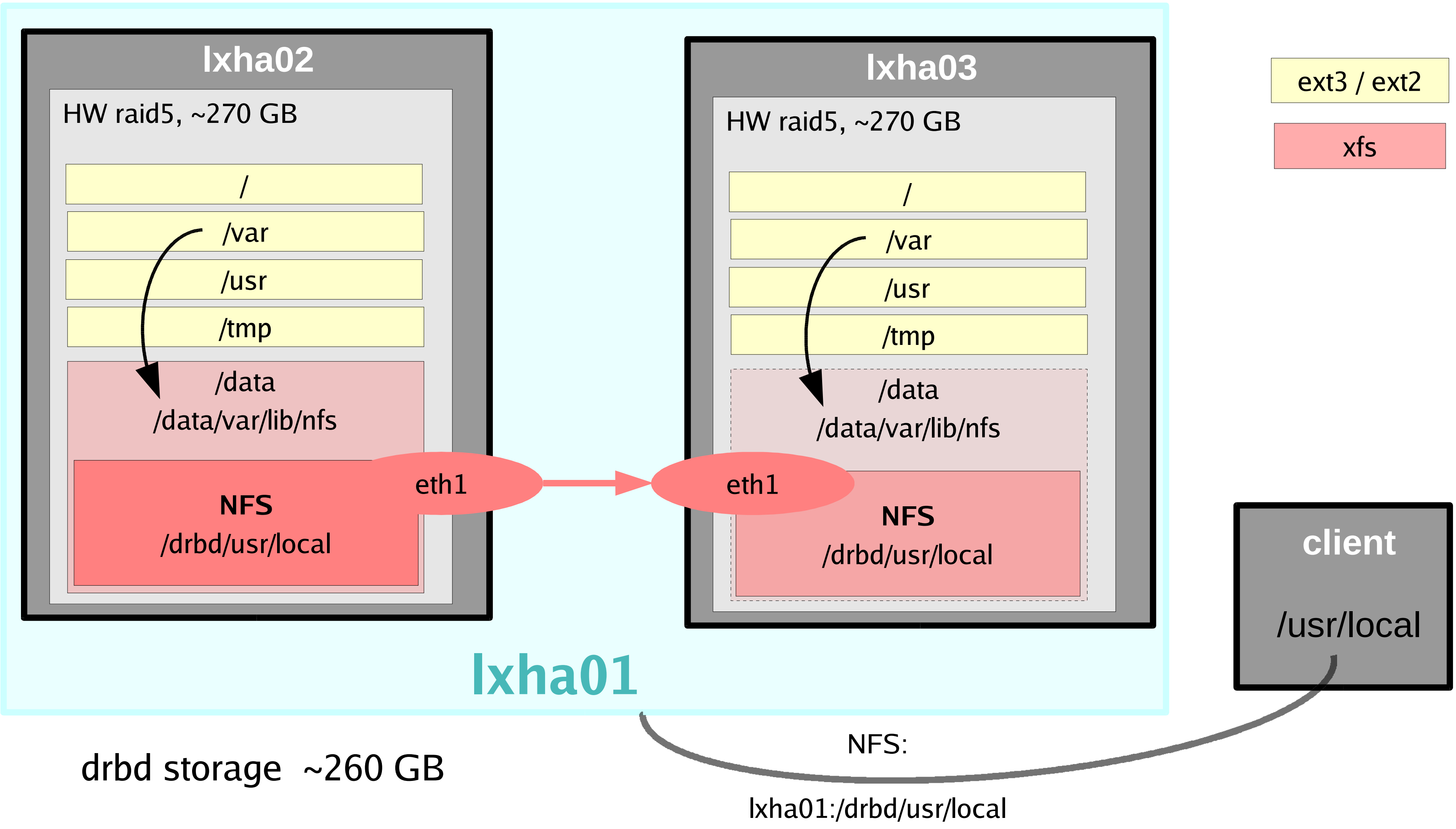
works local and remote (on other node and on a monitoring server):

- drbd, heartbeat running? nfs directory reachable? who is lxha01?
- triggers a reboot and sends information messages

Network Configuration



Configuration Drbd



drbd storage ~260 GB

Experiences in Case of Failure

- in case of failure the nfs service is taken over by the slave server (test -> switch off the master)
- watchdog, stonith (ssh) and ipfail work as designed
- in general clients only see a short interruption and continue to work without disturbance
- down time depends on heartbeat and drbd configuration

example:

- heartbeat 2 s, dead time 10 s = > interruption ~20 s

Replication DRBD

- full sync takes approximately 5 h (for 260 GB)
- only necessary during installation or if a in case of a complete overrun happens
- normal sync duration depends on the change of the file system during down time

example:

- drbd stopped, 1 GB written - sync: 26s until start up, 81s for synchronisation
- 1 GB deleted, 27 s until start up, synchronisation time ~ 0

Write Performance

with iozone, 4GB file size

- xfs file system without drbd, single thread: 28,9 MB/s
- with drbd (connected): 17,4 MB/s --> 60 %
- unconnected: 24,2 MB/s --> 84 %
- 4 threads: 15,0 MB/s
- with drbd (connected), but protocol A: 21,4 MB/s --> 74 %
- unconnected: 24,2 MB/s --> 84 %