Virtualization using Xen

Álvaro López García [1]

[1] Instituto de Física de Cantabria - CSIC-UC
Grupo de computación avanzada y Grid

Grids & e-Science 2009
UIMP - Palacio de la Magdalena
Santander - Spain

Outline

Virtualization

Definition

Tecnology that enables the creation of several execution environmets –called Virtual Machines– within a given machine –called host, physical machine–, by using a specific software –called Virtual Machine Monitor or hypervisor–.

- Independent.
- Secured and encapsulated.
- Separation between physical and virtual machines.
- Diferent virtualization technologies: OS level, complete, hardware, *paravirtualization*.
- Introduces a performance loss –depending on the technique, negligible or not–.

Benefits

Some of the benefits of the virtualization:

Server consolidation Several VM per host (less noise, less power, less space).

Virtual/Physical isolation Any VM can be executed on any PM, without modifications.

Migration, load balancing A VM can be migrated from a loaded node to a free one.

Cloning, snapshotting Depending sometimes of the storage used, allows the rollback to an older version. Useful for testing purposes, to recover from software failures, etc.

Fast deployment Creation of VMs is a quick procedure.

Benefits

Some of the benefits of the virtualization:

Server consolidation	Several VM per host (less noise, less power, less space).
Virtual/Physical isolation	Any VM can be executed on any PM, without modifications.
Migration, load balancing	A VM can be migrated from a loaded node to a free one.
Cloning, snapshotting	Depending sometimes of the storage used, allows the rollback to an older version. Useful for testing purposes, to recover from software failures, etc.
Fast deployment	Creation of VMs is a quick procedure.

Benefits

Some of the benefits of the virtualization:

Server consolidation	Several VM per host (less noise, less power, less space).
Virtual/Physical isolation	Any VM can be executed on any PM, without modifications.
Migration, load balancing	A VM can be migrated from a loaded node to a free one.
Cloning, snapshotting	Depending sometimes of the storage used, allows the rollback to an older version. Useful for testing purposes, to recover from software failures, etc.
Fast deployment	Creation of VMs is a quick procedure.

Benefits

Some of the benefits of the virtualization:

Server consolidation	Several VM per host (less noise, less power, less space).
Virtual/Physical isolation	Any VM can be executed on any PM, without modifications.
Migration, load balancing	A VM can be migrated from a loaded node to a free one.
Cloning, snapshotting	Depending sometimes of the storage used, allows the rollback to an older version. Useful for testing purposes, to recover from software failures, etc.
Fast deployment	Creation of VMs is a quick procedure.

Benefits

Some of the benefits of the virtualization:

Server consolidation	Several VM per host (less noise, less power, less space).
Virtual/Physical isolation	Any VM can be executed on any PM, without modifications.
Migration, load balancing	A VM can be migrated from a loaded node to a free one.
Cloning, snapshotting	Depending sometimes of the storage used, allows the rollback to an older version. Useful for testing purposes, to recover from software failures, etc.
Fast deployment	Creation of VMs is a quick procedure.

Outline

Xen Overview

- Developed at Cambridge University.
- Free software (GPL License).
- Not only implemets paravirtualization, also supports unmodified VMs (using hardware support).
- Migration, live migration.
- Supported and distributed among several (if not all) the Linux distributions.

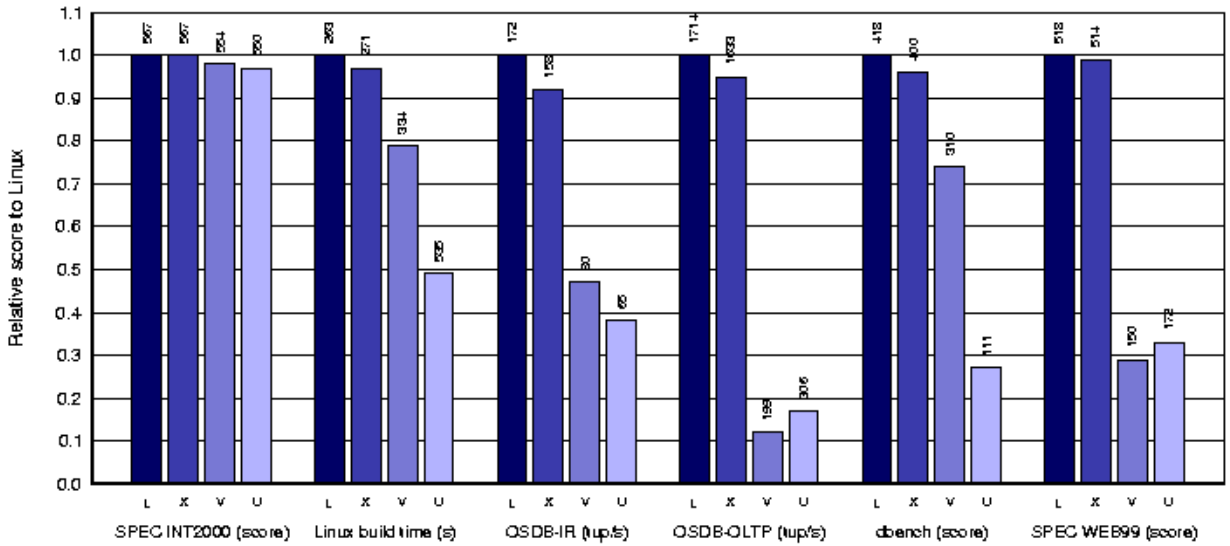
Terminology

dom0 Also called host. System hosting VMs.

domU Also called guest. Virtual Machines.

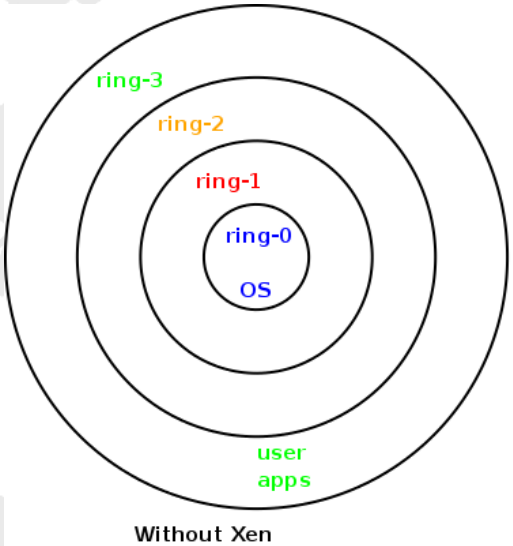
hypervisor Also Virtual Machine Monitor. Virtualization software (i.e. Xen).

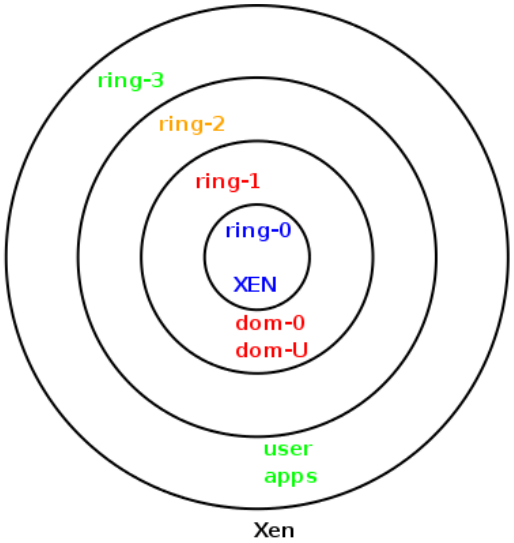
Paravirtualization



- Performance very close to the native OS.
- Modification of the virtualized OS kernel.

Rings and domains





Normal.

ring 0 OS kernel.

ring 1, 2 Rarely used.

ring3 Userspace applications.

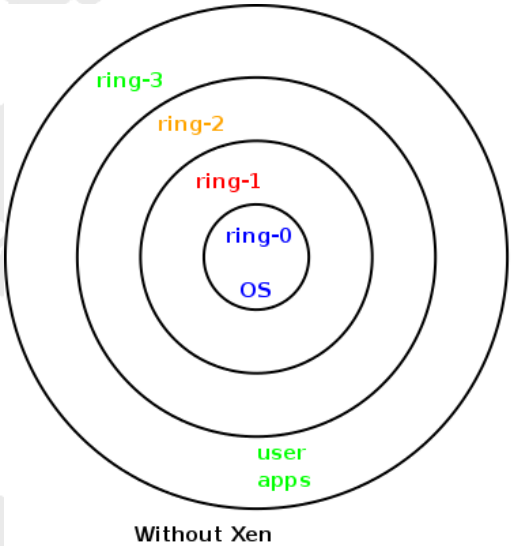
Xen.

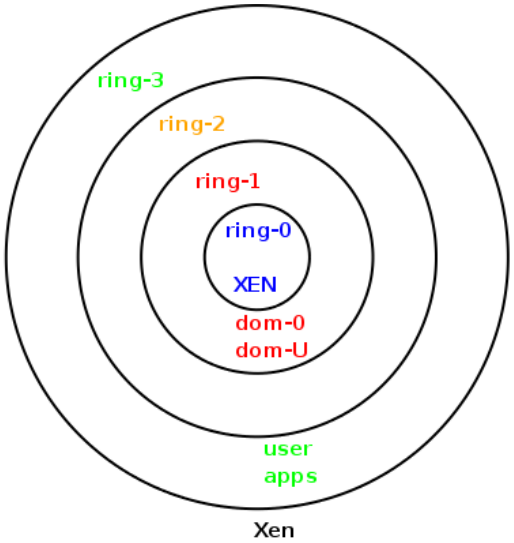
ring 0 Xen hypervisor.

ring 1 dom0 ("physical machine") and domUs. Drivers are on the dom0.

ring3 Userspace applications.

Rings and domains





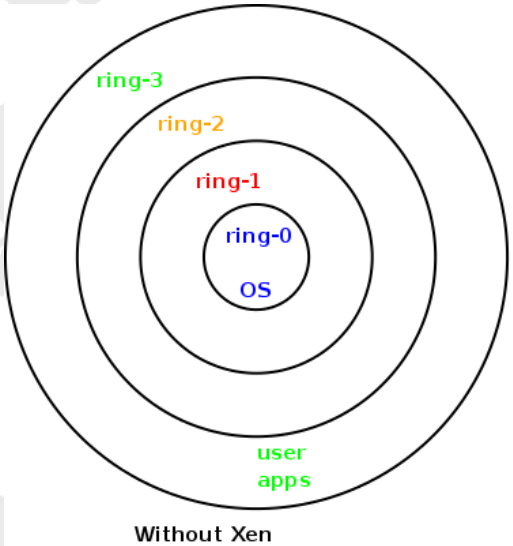
Normal.

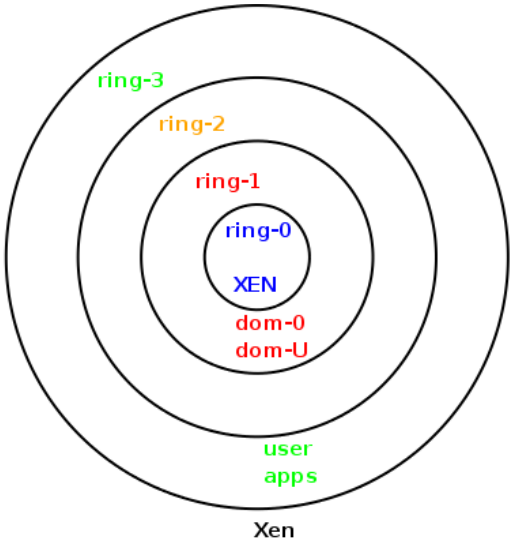
- ring 0 OS kernel.
- ring 1, 2 Rarely used.
- ring3 Userspace applications.

Xen.

- ring 0 Xen hypervisor.
- ring 1 dom0 ("physical machine") and domUs. Drivers are on the dom0.
- ring3 Userspace applications.

Rings and domains





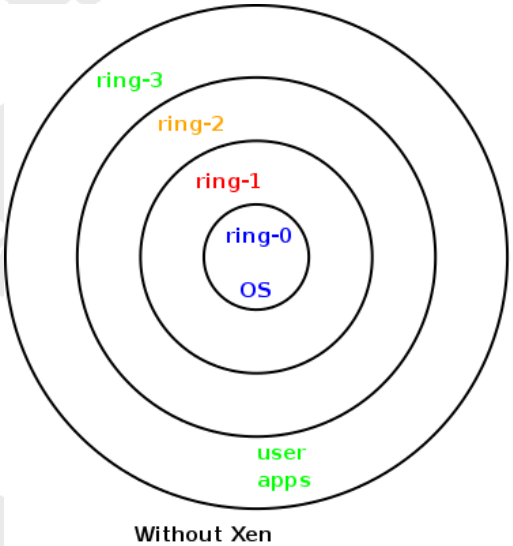
Normal.

- ring 0 OS kernel.
- ring 1, 2 Rarely used.
- ring3 Userspace applications.

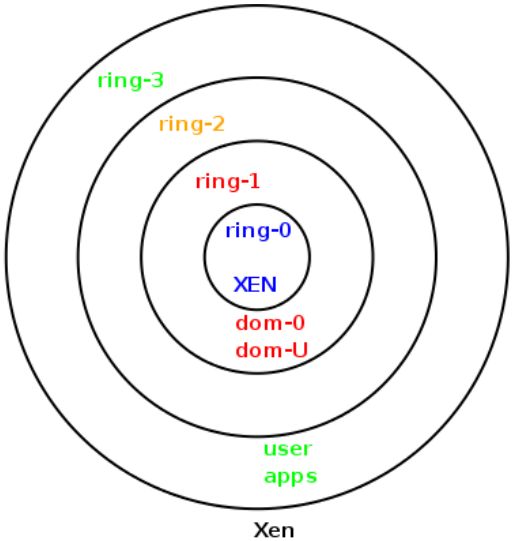
Xen.

- ring 0 Xen hypervisor.
- ring 1 dom0 ("physical machine") and domUs. Drivers are on the dom0.
- ring3 Userspace applications.

Rings and domains



Without Xen



Xen

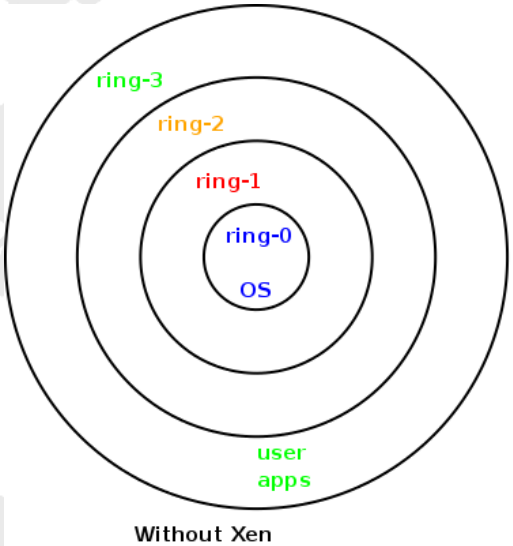
Normal.

- ring 0 — OS kernel.
- ring 1, 2 — Rarely used.
- ring3 — Userspace applications.

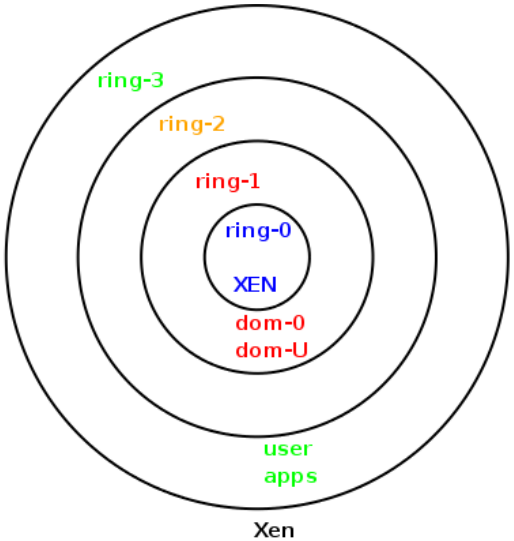
Xen.

- ring 0 — Xen hypervisor.
- ring 1 — dom0 ("physical machine") and domUs. Drivers are on the dom0.
- ring3 — Userspace applications.

Rings and domains



Without Xen



Xen

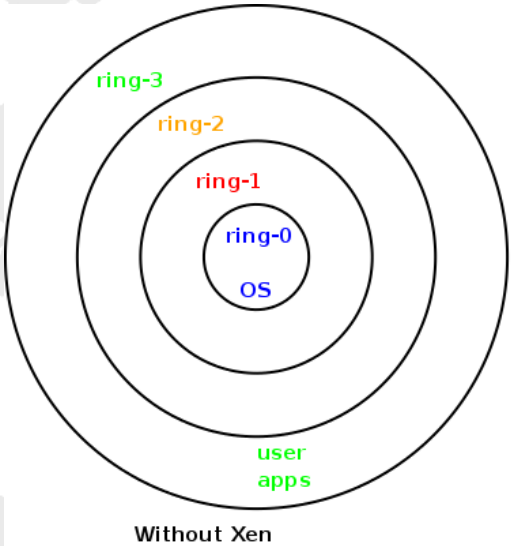
Normal.

- ring 0 — OS kernel.
- ring 1, 2 — Rarely used.
- ring3 — Userspace applications.

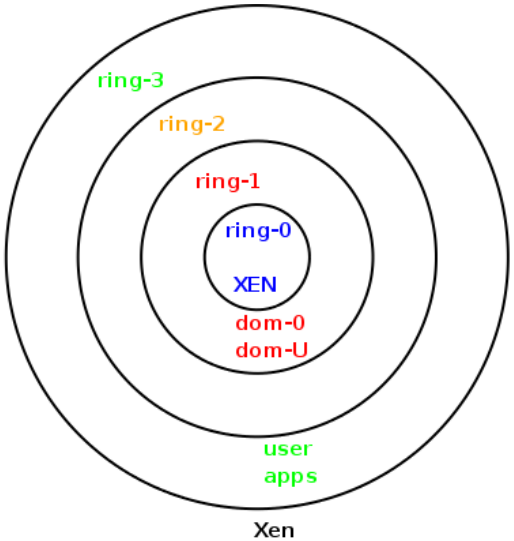
Xen.

- ring 0 — Xen hypervisor.
- ring 1 — dom0 ("physical machine") and domUs. Drivers are on the dom0.
- ring3 — Userspace applications.

Rings and domains



Without Xen



Xen

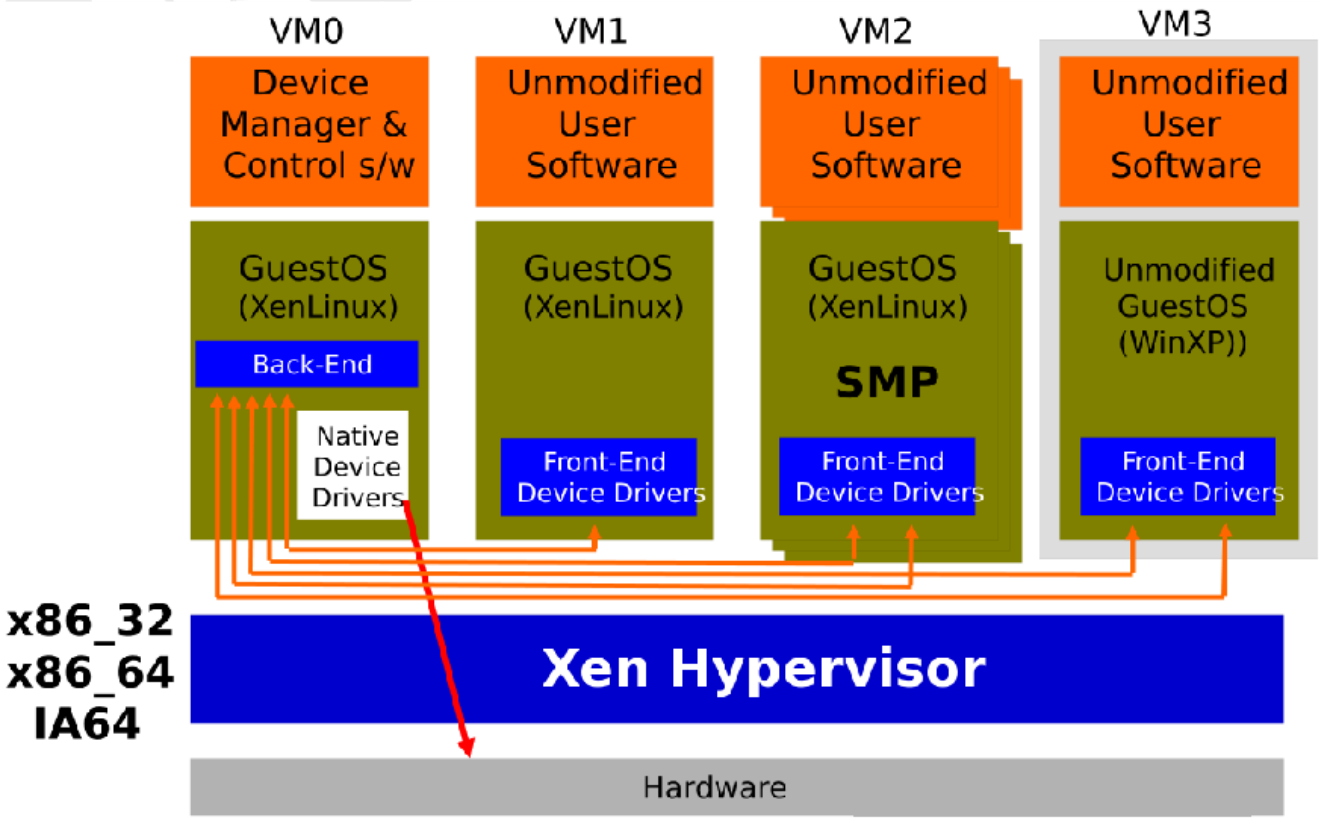
Normal.

- ring 0 — OS kernel.
- ring 1, 2 — Rarely used.
- ring3 — Userspace applications.

Xen.

- ring 0 — Xen hypervisor.
- ring 1 — dom0 ("physical machine") and domUs. Drivers are on the dom0.
- ring3 — Userspace applications.

Drivers and domains



Services on dom0

xend Manipulates dom0.

xendomains Manipulates VMs (auto start/stop, pausing, etc.).

VM management

xm list

```
# xm list
Name                                        ID   Mem VCPUs      State   Time(s)
Domain-0                                     0  6021     8     r-----  27445.5
doriiie01                                    1  1024     1     -b----  13166.2
doriiie02                                    2  1024     1     -b----   3538.6
griddpm01                                   10  1024     1     -b----  23860.9
gridii01                                     4  1024     1     -b----  31701.1
gridiis01                                    5  1024     1     -b----  24291.8
gridlfc01                                    6  1024     1     -b----   7940.1
gridpx01                                     7  1024     1     -b----   8935.3
sams01                                       8  1024     1     -b----   7712.7
voms01                                       9  1024     1     -b----  23889.3
```

VM management

xm top

```
xentop - 12:06:46   Xen 3.2-1
10 domains: 2 running, 8 blocked, 0 paused, 0 crashed, 0 dying, 0 shutdown
Mem: 15728020k total, 15727304k used, 716k free    CPUs: 8 @ 2493MHz
      NAME  STATE    CPU(sec) CPU(%)     MEM(k) MEM(%)  MAXMEM(k) MAXMEM(%) VCPUS NETS NETTX(k) NETRX(k) VB
  Domain-0 -----r       27448    1.0    6165692   39.2   no limit       n/a     8    4        0        0
 doriiie01 -----r       13175    6.8    1048388    6.7    1048576       6.7     1    1   774035   903159
 doriiie02 --b---        3538    0.1    1048320    6.7    1048576       6.7     1    1   153460   460035
 griddpm01 --b---       23866    0.2    1048164    6.7    1048576       6.7     1    2   956369  1226007
  gridii01 --b---       31709    0.7    1048368    6.7    1048576       6.7     1    1  2024982  2447511
 gridiis01 --b---       24297    0.1    1048368    6.7    1048576       6.7     1    1   588537  1538131
 gridlfc01 --b---        7942    0.5    1048332    6.7    1048576       6.7     1    1    40829   380211
  gridpx01 --b---        8936    0.1    1048320    6.7    1048576       6.7     1    1    66159   398708
    sams01 --b---        7713    2.1    1048436    6.7    1048576       6.7     1    1   392356  3793359
    voms01 --b---       23893    0.2    1048348    6.7    1048576       6.7     1    1   332616   407636
```

VM management I

xm overview

`console`	Attach to <Domain>'s console.
`create`	Create a domain based on <ConfigFile>.
`new`	Adds a domain to Xend domain management
`delete`	Remove a domain from Xend domain management.
`destroy`	Terminate a domain immediately.
`dump-core`	Dump core for a specific domain.
`help`	Display this message.
`list`	List information about all/some domains.
`mem-set`	Set the current memory usage for a domain.
`migrate`	Migrate a domain to another machine.
`pause`	Pause execution of a domain.

VM management II

xm overview

`reboot`	Reboot a domain.
`restore`	Restore a domain from a saved state.
`resume`	Resume a Xend managed domain
`save`	Save a domain state to restore later.
`shell`	Launch an interactive shell.
`shutdown`	Shutdown a domain.
`start`	Start a Xend managed domain
`suspend`	Suspend a Xend managed domain
`top`	Monitor a host and the domains in real time.
`unpause`	Unpause a paused domain.
`uptime`	Print uptime for all/some domains.
`vcpu-set`	Set the number of active VCPUs for allowed for the domain.

Overview

Service Availability
Problematic topic

- Inside Grid Computing, there are several critical components.
- A failure could compromise both the rest of the infrastructure and the final users.
- These nodes need to be available almost all the time.
- Several parameters affect availability.

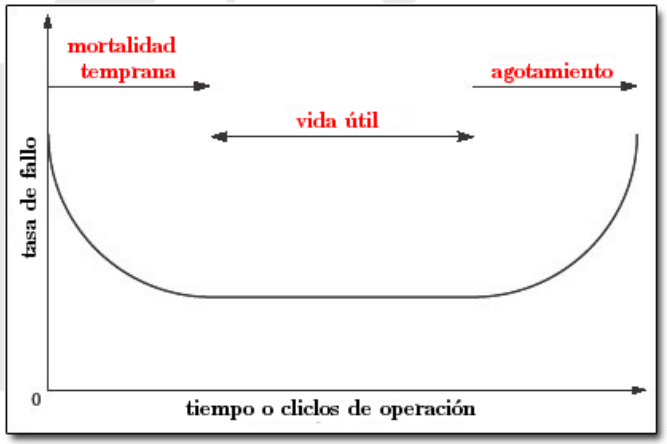
  MTBF Mean Time Between Failres.

  MTTR Mean Time To Recover.

  Availability $d = \frac{MTBF}{MTBF+MTTR}$

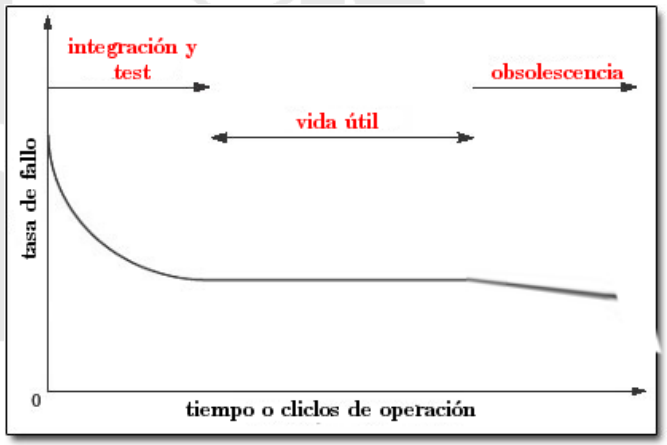
Service availability
Bathtub diagrams

Hardware.



Early failure	Initial failure rate is high until faulty components are identified.
Operating period	Normal operating period. MTBF applies to this period.
Wear out	As the component wears out, the failure starts to increase again.

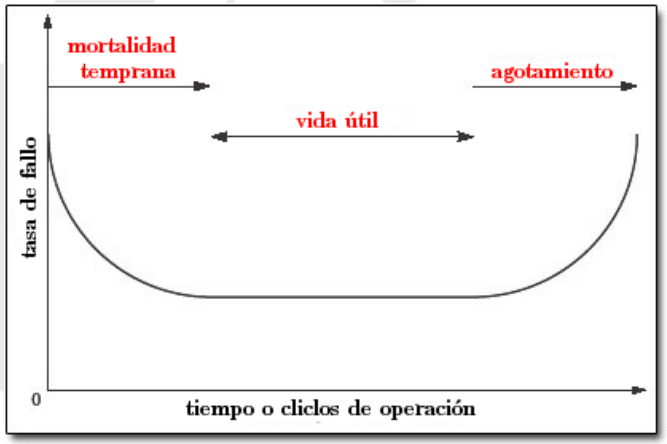
Software.



Test and validation	Early development. Plenty of bugs and errors, which are identified and fixed.
Operating Period	There are still errors, but appear at a slower rate (supposing no new ones are introduced).
Obsolete	The software is obsolete, failure rate continues decreasing.

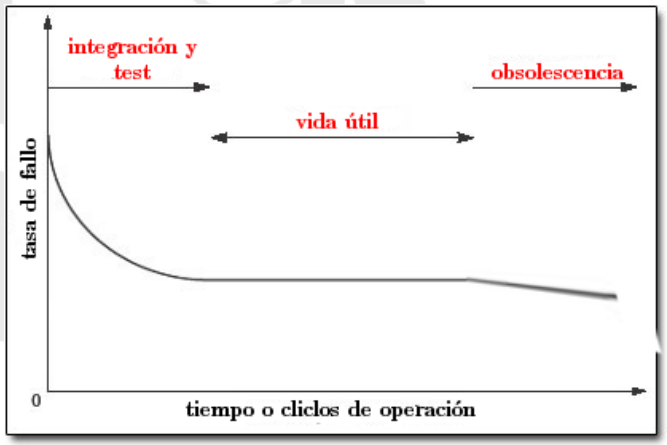
Service availability
Bathtub diagrams

Hardware.



Early failure	Initial failure rate is high until faulty components are identified.
Operating period	Normal operating period. MTBF applies to this period.
Wear out	As the component wears out, the failure starts to increase again.

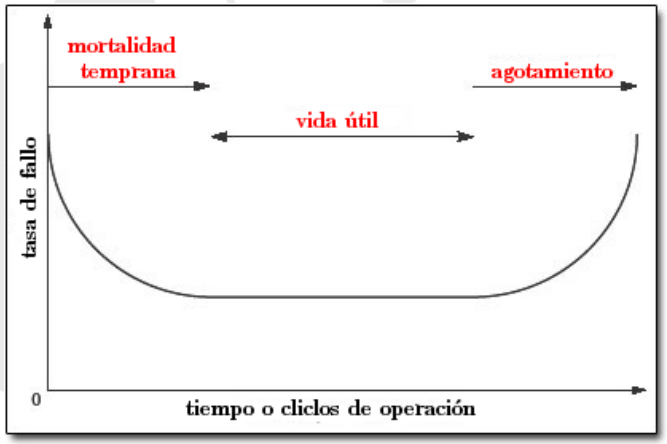
Software.



Test and validation	Early development. Plenty of bugs and errors, which are identified and fixed.
Operating Period	There are still errors, but appear at a slower rate (supposing no new ones are introduced).
Obsolete	The software is obsolete, failure rate continues decreasing.

Service availability
Bathtub diagrams

Hardware.

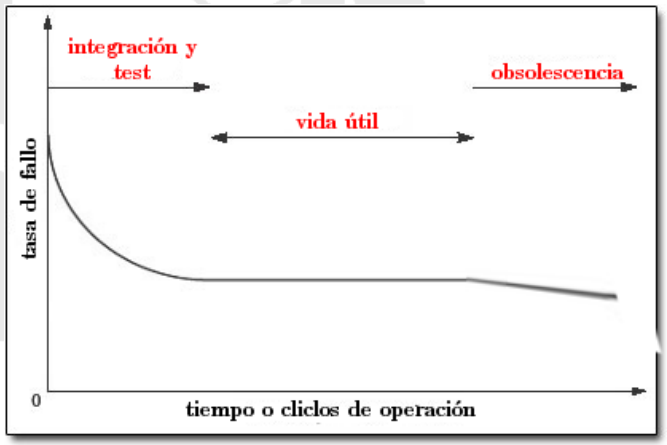


Early failure — Initial failure rate is high until faulty components are identified.

Operating period — Normal operating period. MTBF applies to this period.

Wear out — As the component wears out, the failure starts to increase again.

Software.



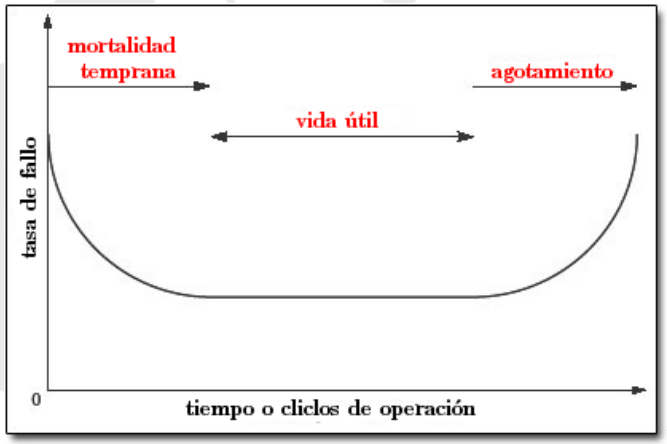
Test and validation — Early development. Plenty of bugs and errors, which are identified and fixed.

Operating Period — There are still errors, but appear at a slower rate (supposing no new ones are introduced).

Obsolete — The software is obsolete, failure rate continues decreasing.

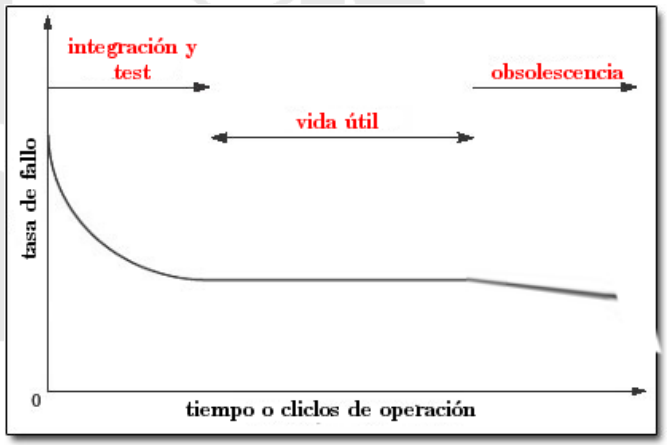
Service availability
Bathtub diagrams

Hardware.



Early failure	Initial failure rate is high until faulty components are identified.
Operating period	Normal operating period. MTBF applies to this period.
Wear out	As the component wears out, the failure starts to increase again.

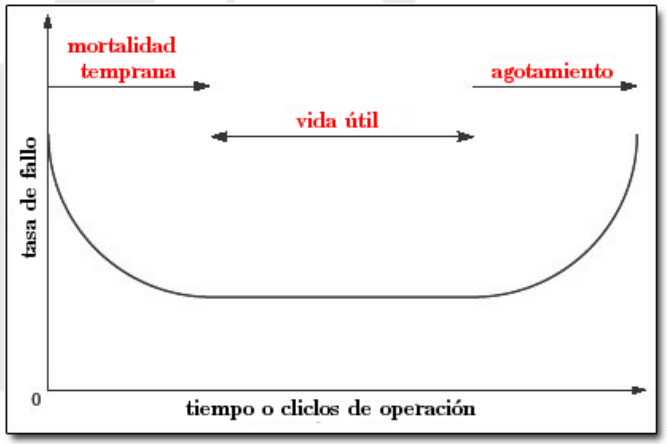
Software.



Test and validation	Early development. Plenty of bugs and errors, which are identified and fixed.
Operating Period	There are still errors, but appear at a slower rate (supposing no new ones are introduced).
Obsolete	The software is obsolete, failure rate continues decreasing.

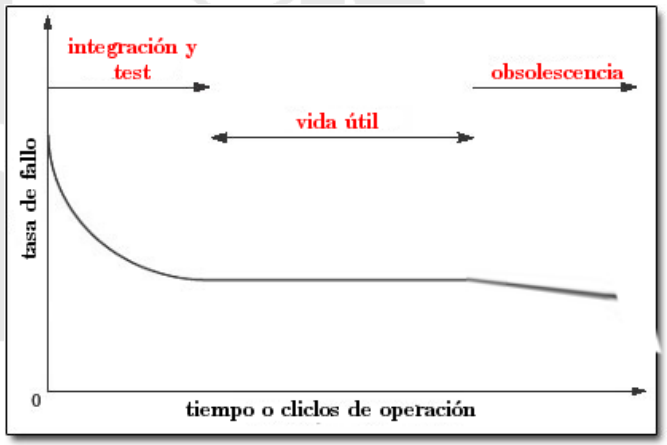
Service availability
Bathtub diagrams

Hardware.



Early failure	Initial failure rate is high until faulty components are identified.
Operating period	Normal operating period. MTBF applies to this period.
Wear out	As the component wears out, the failure starts to increase again.

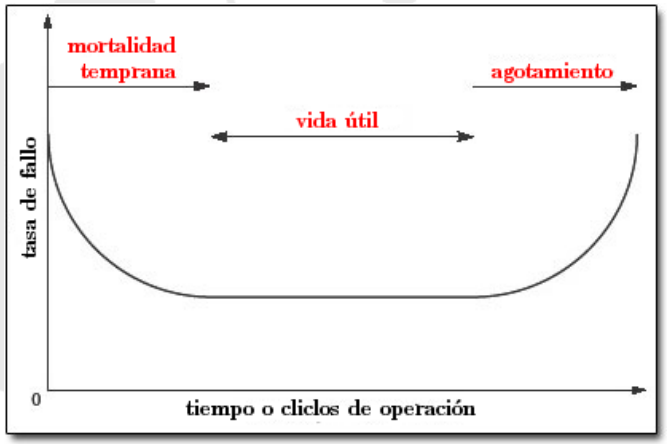
Software.



Test and validation	Early development. Plenty of bugs and errors, which are identified and fixed.
Operating Period	There are still errors, but appear at a slower rate (supposing no new ones are introduced).
Obsolete	The software is obsolete, failure rate continues decreasing.

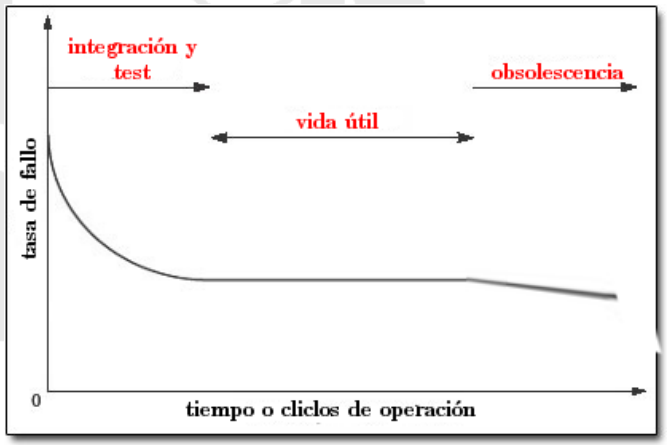
Service availability
Bathtub diagrams

Hardware.



Early failure	Initial failure rate is high until faulty components are identified.
Operating period	Normal operating period. MTBF applies to this period.
Wear out	As the component wears out, the failure starts to increase again.

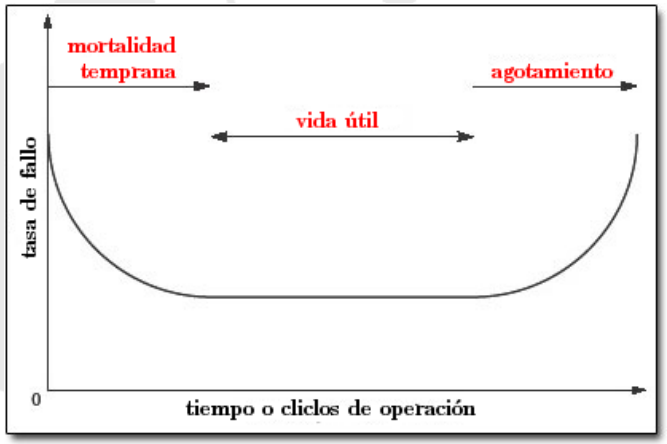
Software.



Test and validation	Early development. Plenty of bugs and errors, which are identified and fixed.
Operating Period	There are still errors, but appear at a slower rate (supposing no new ones are introduced).
Obsolete	The software is obsolete, failure rate continues decreasing.

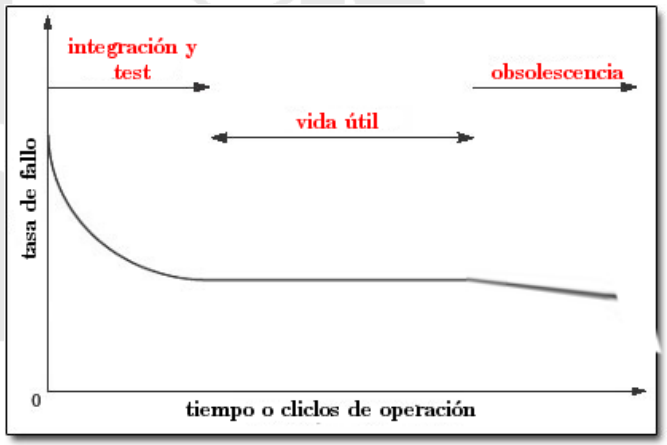
Service availability
Bathtub diagrams

Hardware.



Early failure	Initial failure rate is high until faulty components are identified.
Operating period	Normal operating period. MTBF applies to this period.
Wear out	As the component wears out, the failure starts to increase again.

Software.



Test and validation	Early development. Plenty of bugs and errors, which are identified and fixed.
Operating Period	There are still errors, but appear at a slower rate (supposing no new ones are introduced).
Obsolete	The software is obsolete, failure rate continues decreasing.

Service availability
Recovery

- MTTR: Mean Time to Recover.
- Period of time between the failure (**not its detection!**) and the recovery.
- It's not the same recover from a software or hardware failure
- It can be *easily* lowered.
  - Redundant hardware.
  - Backups, snapshots, etc.

Overview

Motivation

- At IFCA, the number of virtualized resources is growing.
- We are hosting services for several Grid European and National projects, the majority virtualized.
- We have to face not only hardware failures, but also there are several components prone to software failures (constant sw updates, not heavily tested).
- Virtualization is *a* solution for us.

Motivation

- At IFCA, the number of virtualized resources is growing.
- We are hosting services for several Grid European and National projects, the majority virtualized.
- We have to face not only hardware failures, but also there are several components prone to software failures (constant sw updates, not heavily tested).
- Virtualization is *a* solution for us.

Motivation

- At IFCA, the number of virtualized resources is growing.
- We are hosting services for several Grid European and National projects, the majority virtualized.
- We have to face not only hardware failures, but also there are several components prone to software failures (constant sw updates, not heavily tested).
- Virtualization is *a* solution for us.

Motivation

- At IFCA, the number of virtualized resources is growing.
- We are hosting services for several Grid European and National projects, the majority virtualized.
- We have to face not only hardware failures, but also there are several components prone to software failures (constant sw updates, not heavily tested).
- Virtualization is *a* solution for us.

Outline

Solution
Concept

- Virtualization of the service machines.
  - Virtualization component.
- Remote storage of the VM images.
  - Storage component.
- Software that supervises the infrastructure.
  - Monitoring component.

Solution
Concept

- Virtualization of the service machines.
  - Virtualization component.
- Remote storage of the VM images.
  - Storage component.
- Software that supervises the infrastructure.
  - Monitoring component.

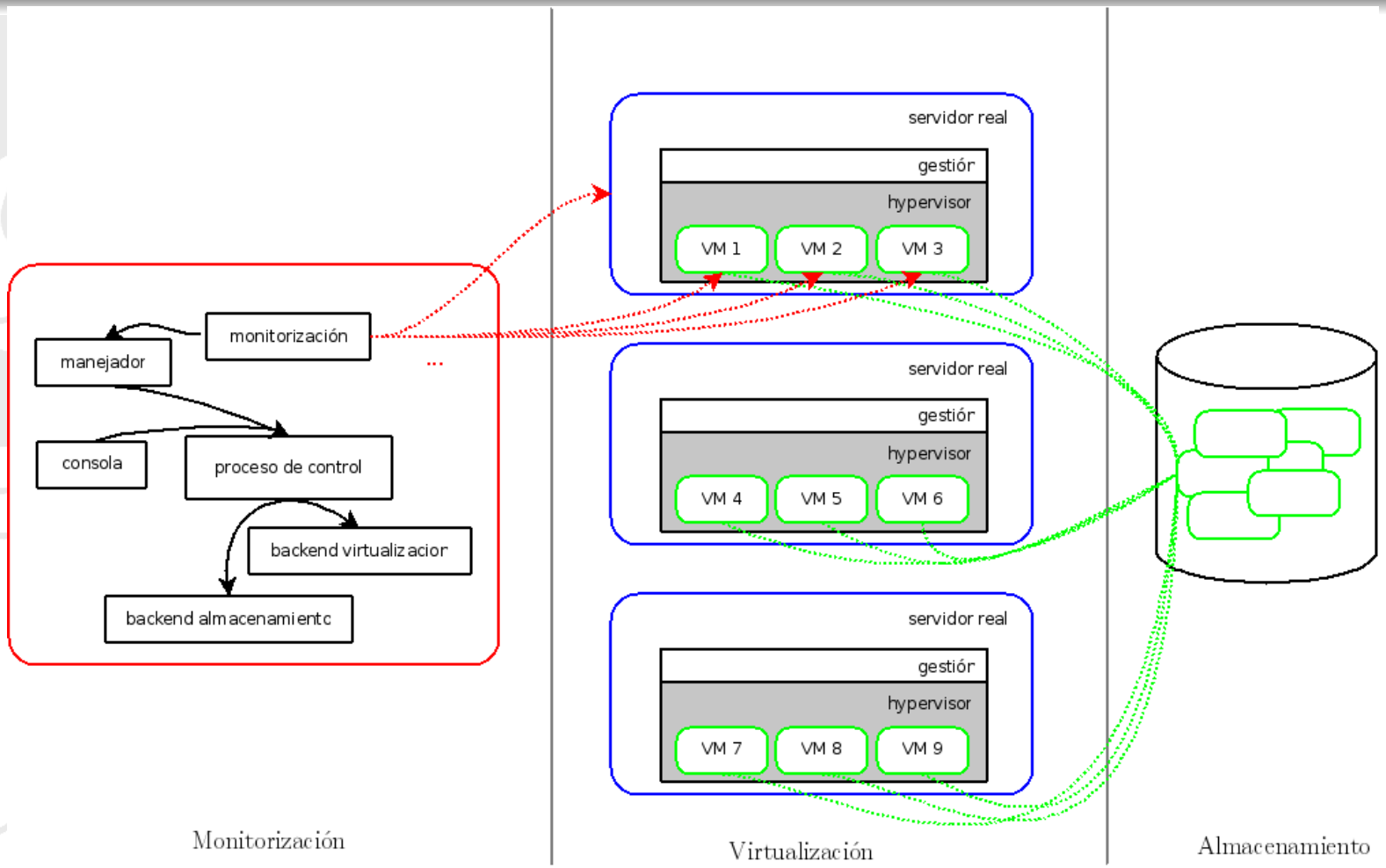
Solution
Concept

- Virtualization of the service machines.
  - Virtualization component.
- Remote storage of the VM images.
  - Storage component.
- Software that supervises the infrastructure.
  - Monitoring component.

Solution
Concept

- Virtualization of the service machines.
  - Virtualization component.
- Remote storage of the VM images.
  - Storage component.
- Software that supervises the infrastructure.
  - Monitoring component.

Solution
Design



Outline

Virtualization

- Virtualization of the service servers.
- Two subcomponents: hypervisor and management.

hypervisor

- Virtualization software.
- Xen so far.

Management

- Abstraction layer, to isolate the virtualization from the final hypervisor.
- Common API.
- Tested libvirt, evaluating OpenNebula.

Storage

- The VM images are stored on a network storage.
- Available for all the virtualization servers across the network.
- Preferably capable of making snapshots and/or cloning.
- Redundant, reliable.
- Two subcomponents:

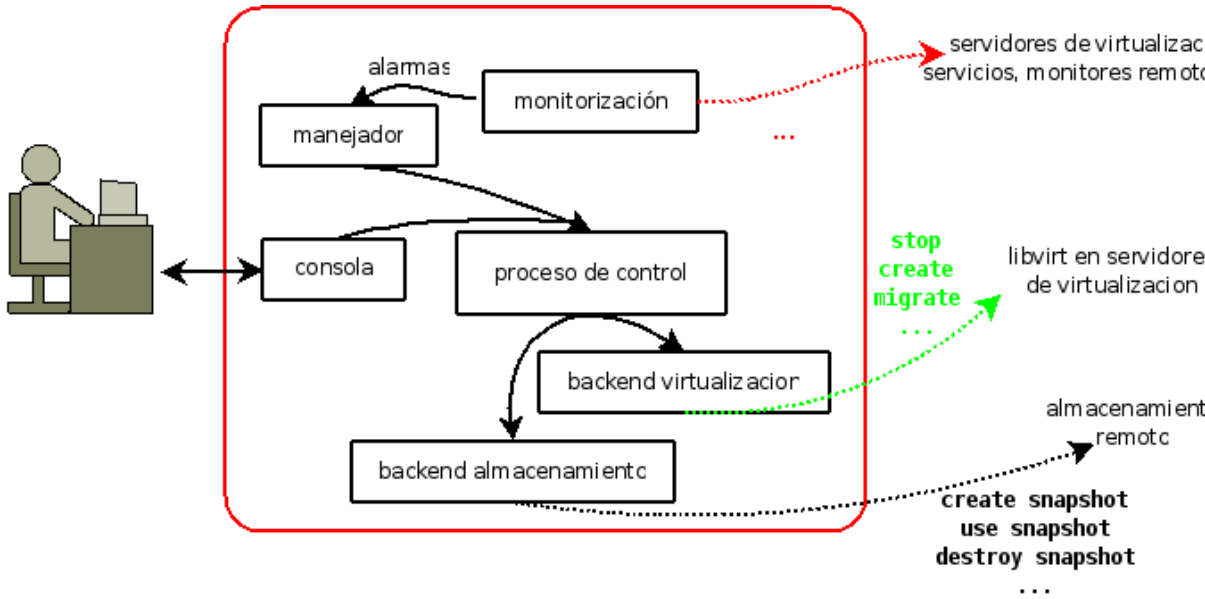
  frontend Exports images to the servers (iSCSI, AoE, FC, shared FS (NFS, GPFS, Lustre, etc), etc.).

  backend Redundancy, snapshotting... (LVM, ZFS, GPFS, etc.).

Monitoring and management

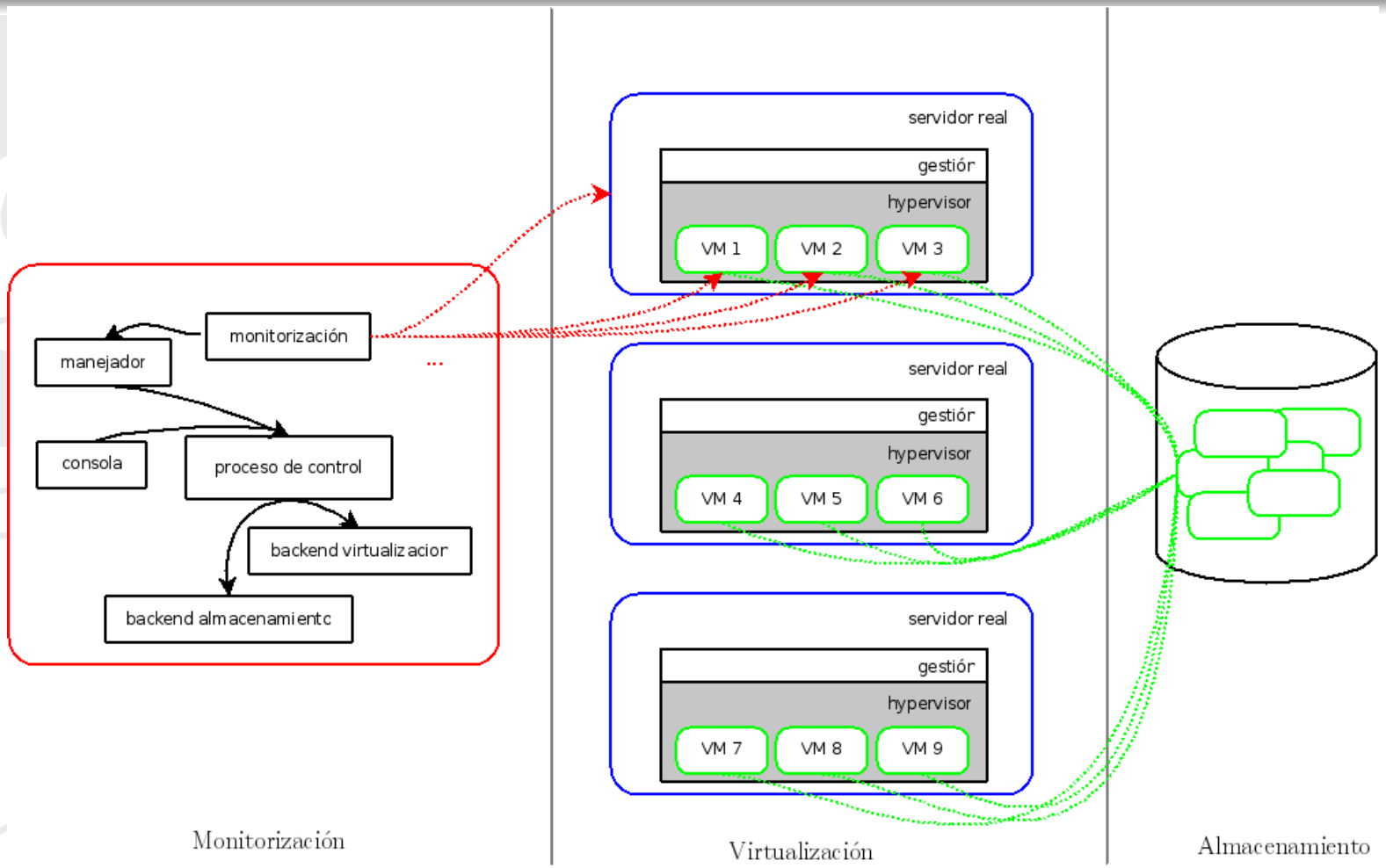
Overview

- Monitoring.
- Alarms.
- Control software.
- Managers
  - Virtualization.
  - Storage.
- Consola.



Outline

Solution
Design



This is the end

Any questions?

¡Muchas gracias!