High Availability using virtualization

Federico Calzolari

Scuola Normale Superiore - INFN Pisa

High availability using virtualization

Outline

- **High Availability** definition and measure
- **Virtualization** definition and features
- **Scenario** Grid data center
- **Infrastructure**
- **Solutions**
  - **High availability using virtualization**
  - Redundancy in virtual environments
  - Physical to Virtual migration
  - Operation in a real crash example
- **Spin-off**
  - Host on demand

High availability using virtualization

Aims

- zero cost High availability service

Requirements

- full exploitation of virtual environment features

High availability using virtualization

High availability definition

- **High Availability:** system design protocol that ensures a certain degree of operational continuity during a given period.
- **Fault Tolerance:** property that enables a system to continue operating properly in the event of the failure of some of its components.
- **Data Reliability / Redundancy:** property of some disk arrays which provides fault tolerance [no data lost in case of disk failure].

supplied by:

- **Load Balancing:** technique to spread work between many computers, processes, disks or other resources.
- **Failover:** capability to automatically switch over to a redundant or standby computer server, system, or network.

High availability using virtualization

High availability features

- User does not have to care about how/where to access services/data
- Reduce downtime to a minimum

High availability measure

- Availability is described in "number of nines"; the number N of nines describes a system available a fraction A of the time
  $N = -\log_{10}(1 - A)$
- Availability is usually expressed as a percentage of uptime in a given year:
  - 99.9% = 8.76 hours /year [my target]
  - 99.99% = 52.6 minutes/year
  - 99.999% = 5.26 minutes/year [telecommunications]

High availability using virtualization

Virtualization definition

- **Virtualization:** abstraction of computer resources.

  Abstraction layer that allows each physical server to run one or more virtual servers, decoupling operating system and applications from the underlying physical server.

Virtualization benefits?

- 1 service/host:

  split a multi processor server into more independent virtual hosts

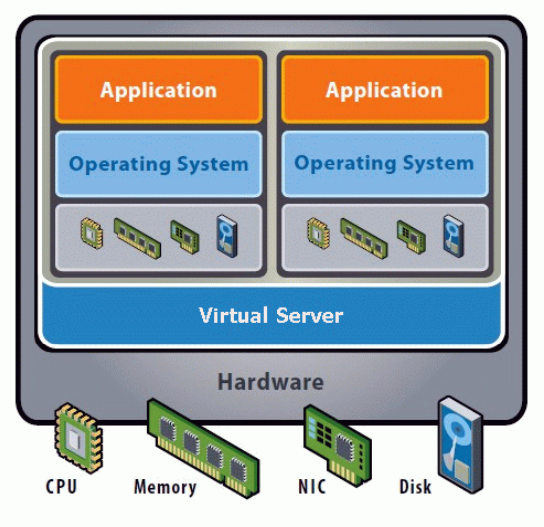
supplied by:

- **VMware:** NOT open source but free version
- **Xen:** open source, free, virtualization and para-virtualization, Kernel patch

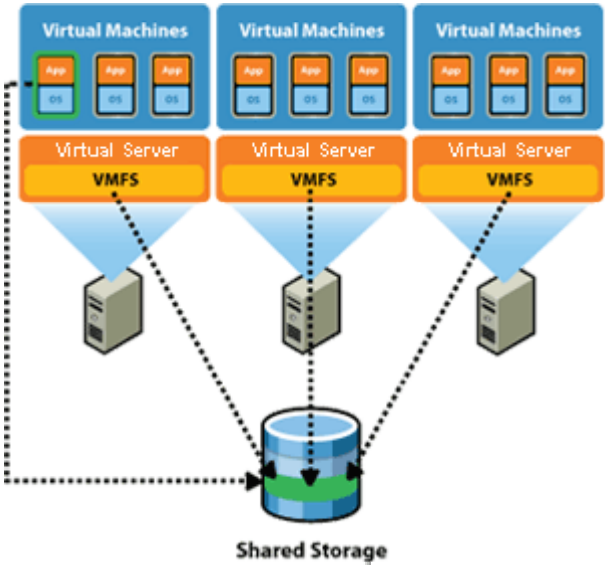
Virtualization features

What can Virtualization do?

- A single server can host multiple Virtual machines, each one providing a specific service.
- More servers can share a common external filesystem to ease virtual disk (VMFS) moving.



Virtualized architecture

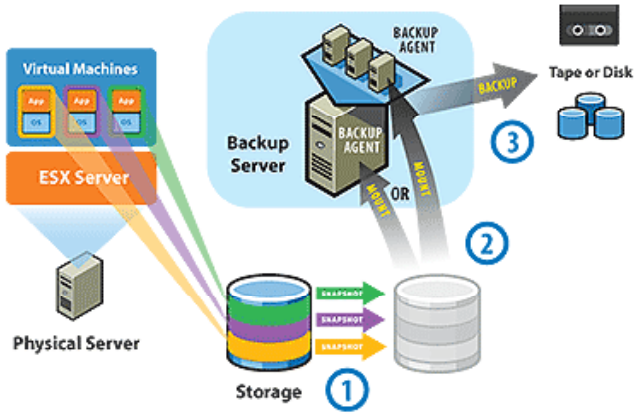


Shared Storage

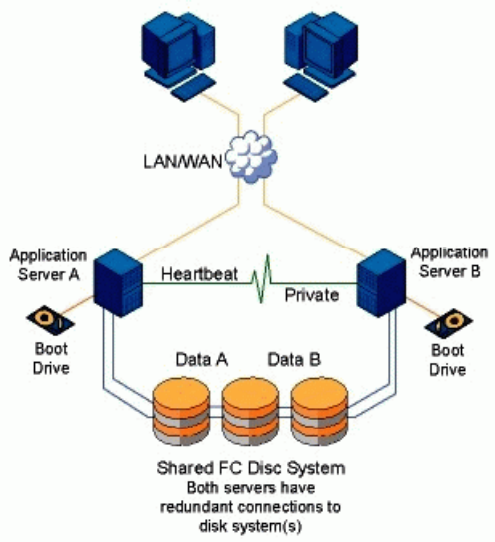
High availability using virtualization

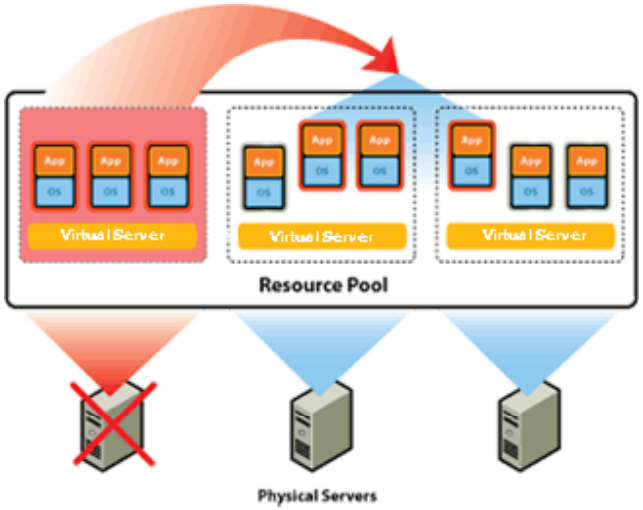
Why Virtualization?

- decouple hardware from software
  - suspend/recover virtual machines
  - virtual machines migration
- increase server density
- better control and manageability



Classical - versus - Virtualized solution





High availability using virtualization

Scenario: GRID data center

What is in a GRID data center?

- 1 + Computing element: communication between farm and external (gateway)
- 1 + Storage element: disk server with SRM features
- 1 Batch Queuing System master
- 1 Monitoring service
- 1 BDII: Berkeley Database Information Index (Information provider)
- 5 Services: specific Virtual Organization applications
- 1 + User Interface: user access to Grid
- 1 Cache proxy server: Squid
- N Worker nodes: computational nodes

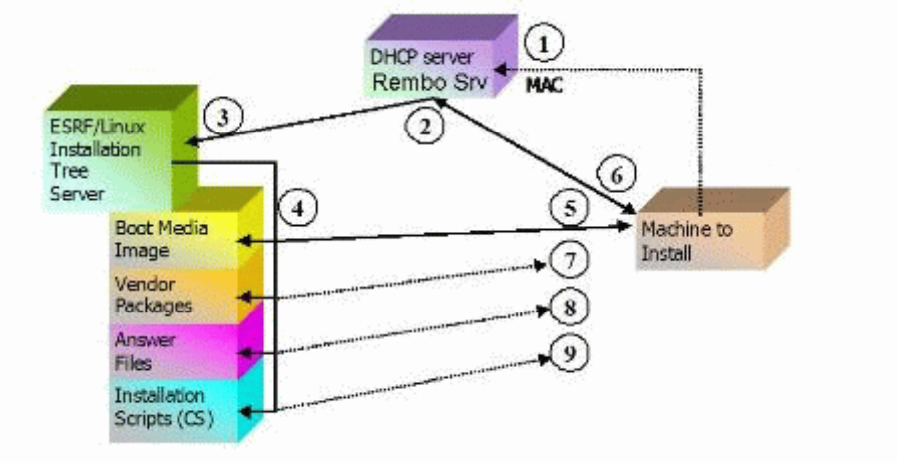
What is necessary to grant service?

- ALL but Worker nodes (~ 20 hosts)

Infrastructure - I

How to provide an automatic host installation?

- DHCP
- DNS with HINFO (Host Info) = host_type
- PXE Preboot eXecution Environment
- TFTP
- HTTP



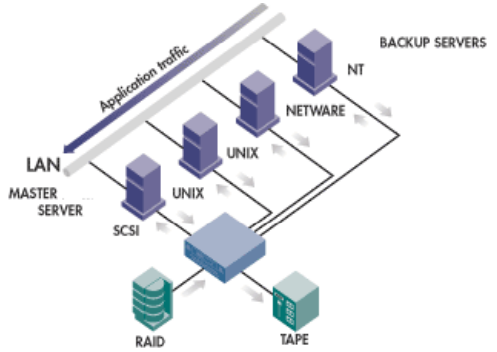
PXE architecture

High availability using virtualization

Infrastructure - II

Storage solutions

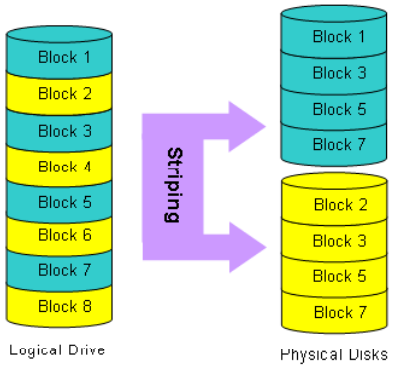
- DAS — Direct Attached Storage
- NAS — Network Attached Storage
- SAN — Storage Area Network



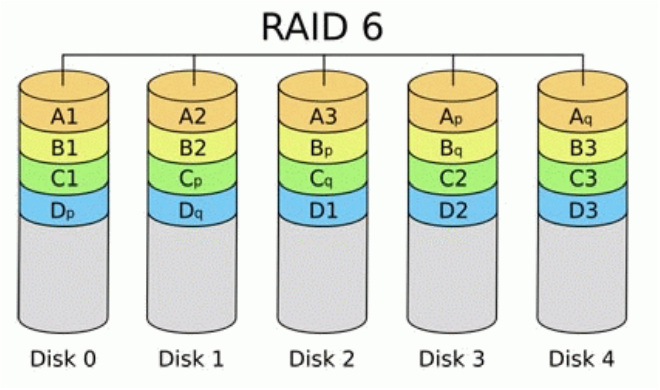
Storage architecture

Requirement: reliable storage

- RAID — Redundant Array of Independent Disks
- DRBD — Distributed Replicated Block Device - Mirror over Network



Data Striping



RAID 6

High availability using virtualization

Infrastructure - III

- INFN-PISA EGEE Grid node: 2000 CPU, 500 TB disk
- SNS-PISA EGEE Grid node: ~small, testbed
- CNR-ISTI EGEE Grid node: Pre Production Service

- centralized installation via PXE, DNS, DHCP, TFTP, HTTP
- manage up to 2000 virtual machines/disks simultaneously: 16 Gb/s aggregate bandwidth

High availability using virtualization

A new approach to High availability

- **RELAXED** High availability service: a system able to restore any previously running application in less than ten minutes from the crash time.
- A relaxed system may ensure the application redundancy required in the greater part of cases.

How can a High availability service be achieved?

- Virtual machines are highly portable between computers.
- A virtual machine can pause operation, be moved or copied to another physical computer, and there resume execution exactly where it left off.

Hysteresis

Tendency of a system
to respond differently to the same stimulus
depending on the initial state of the system

definition by Claudia Guida
Molecular Biologist @IEO Milan

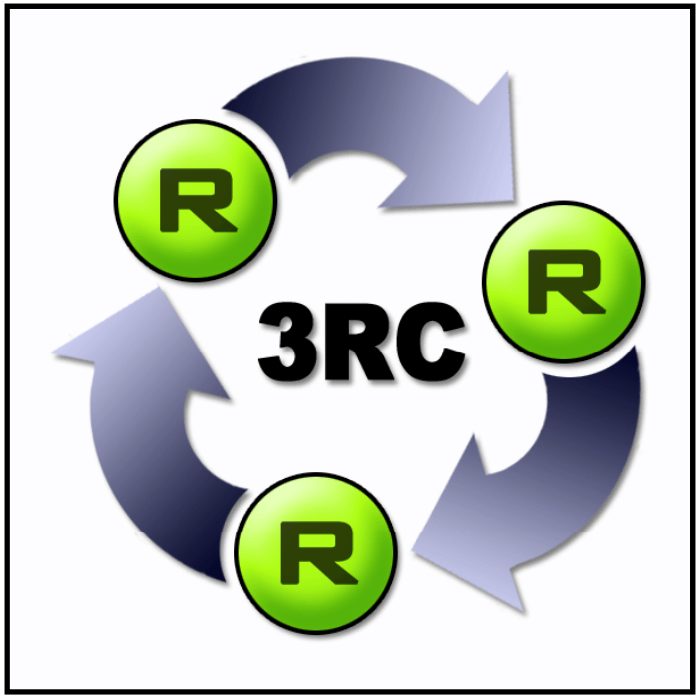
High availability using virtualization

Research topics

- Monitor service to check the physical/virtual hosts health status
- Remote controller able to perform actions over physical/virtual hosts
  - reboot
  - restart virtual machine
  - restart virtual layer
  - move virtual machine to another host
  - reinstall from scratch - via Preboot eXecution Environment PXE
- Infrastructure: DHCP, DNS, HTTP, PXE, TFTP
- Storage architecture
- Procedures: physical to virtual migration

High availability using virtualization

Project 3RC: 3 Re Cycle



Finite state machine with hysteresis

- Ⓡ Reboot
- Ⓡ Restart
- Ⓡ Reinstall

Requirements

- N physical hosts
  - each **ONE** can backup **ALL** others
- 1 **controller** [shared]
- reliable storage
  - SAN or NAS via FC or NFS
  - RAID over network DRBD

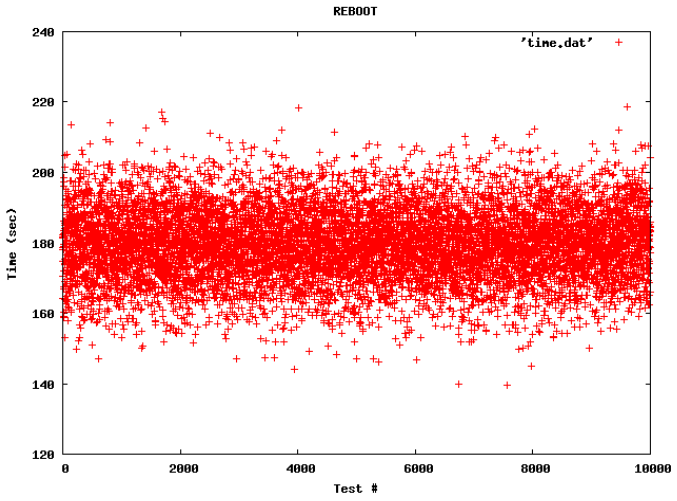
Goals

- **relaxed** High Availability **< 10 min**
- backup **ONLY** @disaster_time

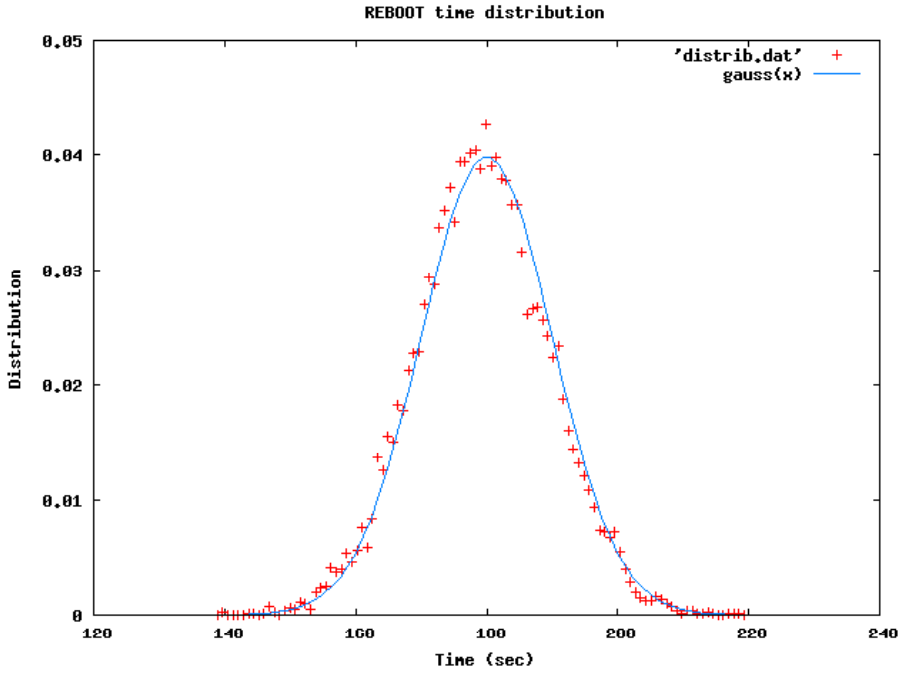
High availability using virtualization

Experimental data - I

- NON Destructive test



Recovery time - 10.000 crash test



Recovery time distribution - 10.000 crash test

NON Destructive test:

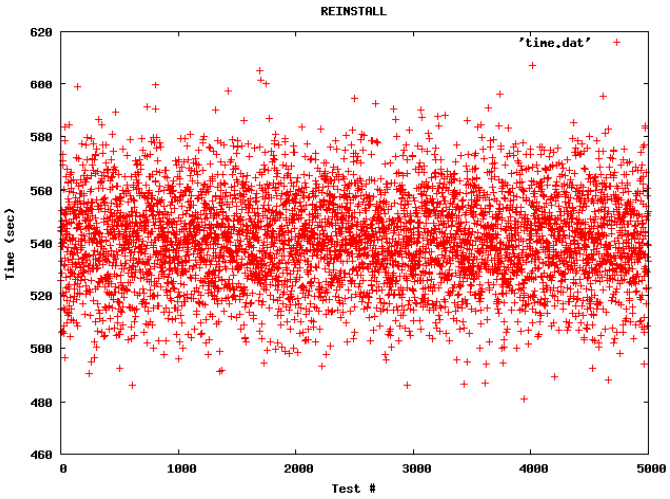
- overload
- shutdown

Gaussian: mean 181 sec
sigma 10 sec

High availability using virtualization

Experimental data - II

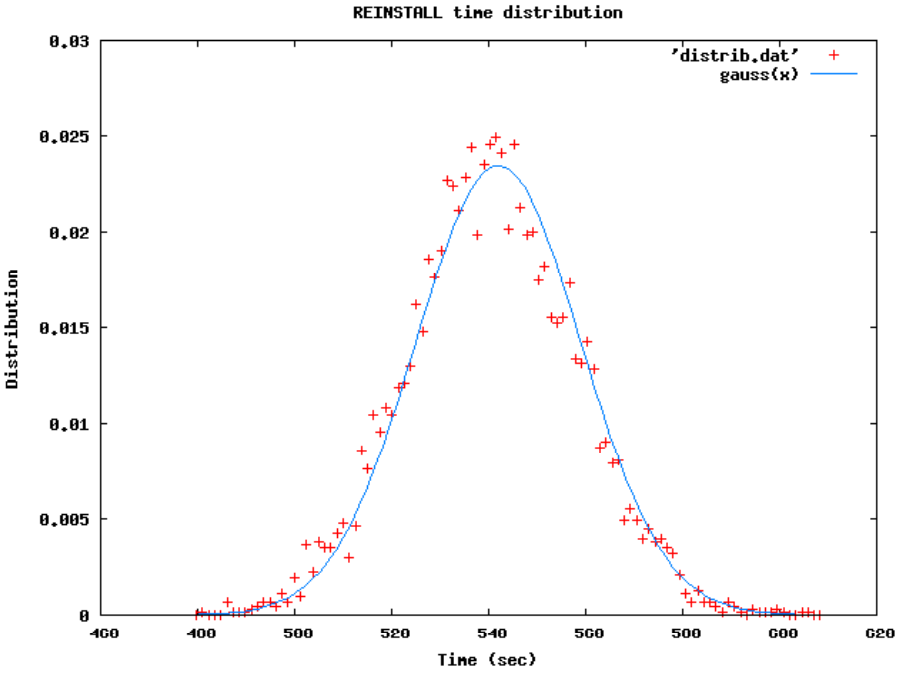
- DESTRUCTIVE test



Reinstall time - 5.000 crash test

DESTRUCTIVE test:

- rm /boot; reboot



Reinstall time distribution - 5.000 crash test

Gaussian: mean 542 sec
sigma 17 sec

High availability using virtualization

Redundancy in virtual environments

Several redundancy strategies ⇒ several availability levels

- Virtual machines/disks on external storage
  - problems if software crashes
- Scheduled virtual machines dump: disk, ram, registers
  - dump at scheduled times ⇒ recovery at time T_{n-1}
- Virtual machines/disks with operating system and middleware ready to be mounted
  - virgin machine from disk copy
- Install from scratch: operating system and middleware
  - virgin machine from real installation via PXE

High availability using virtualization

Physical to Virtual

How to migrate a physical machine to a virtual machine

- physical machine RUNNING
  - create virtual disk
  - mount virtual disk with Linux live distro or Virtualization-tools
  - rsync <real> to <virtual>
  - untar <special path> [/dev]
  - grub install

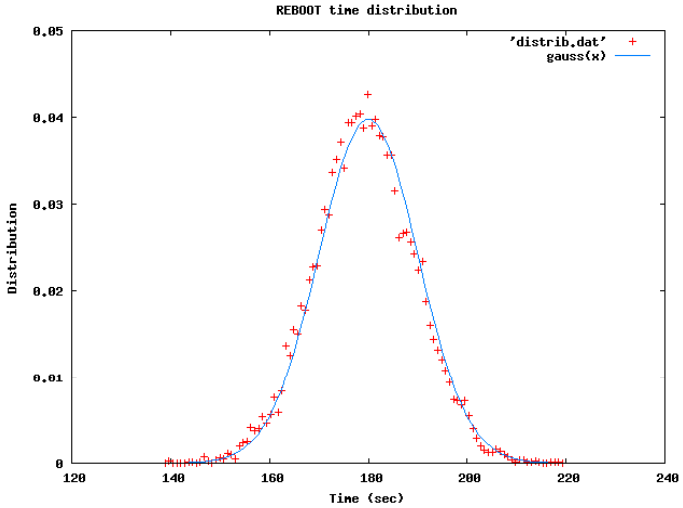
  < 20 sec downtime for switch real to virtual

- physical machine STOPPED
  - create virtual disk
  - mount virtual disk with Linux live distro or Virtualization-tools
  - dd <real> to <virtual>
  - grub install

High availability using virtualization

Outcomes

- RECOVER crashed machine in 3 min
- REINSTALL broken machine in 9 min



RECOVERY TIME

- SNS-PISA is the first EGEE/LCG Grid node
  - fully virtualized (services + WN)
  - highly available

⇨ NO downtime after service crash

High availability using virtualization

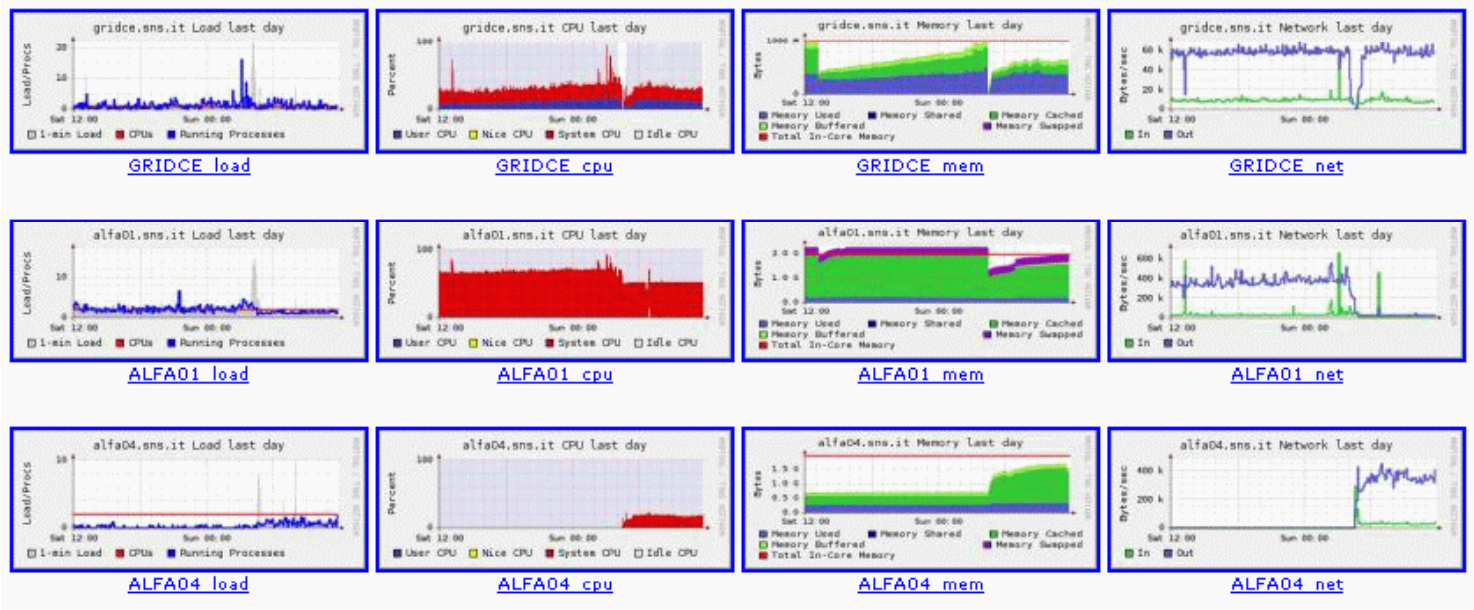
What 3RC High availability project is for

- All the environments satisfied by a Relaxed High availability solution
  - computing
  - information
  - monitoring
  - users management
  - GRID data center services

High availability using virtualization

Operation in a real crash example

- gridce.sns.it [SNS-PISA Grid node Computing Element] CRASH for an electrical power glitch @4:00 AM



GRIDCE crashed virtual machine

ALFA01 primary physical host
ALFA04 secondary physical host

High availability using virtualization

Note

It is important to know what a theorem states,
but it is probably more important
to know what a theorem does not state

statement by Luigi Picasso
Theoretical Physics Professor @University of Pisa

High availability using virtualization

What 3RC High availability project is NOT for

- Mission critical applications
  - financial transactions
  - security certificates management
  - real time controllers
  - human health related applications
- miracles [at least in the current release]

High availability using virtualization

Spin-off: Host on-demand

Host on-demand: basic concepts

- Virtualization and PXE architecture allows to bring up a server in a few minutes

Possibility to offer host on-demand:

- CPU n core
- RAM n GB
- DISK n TB
- Operating System Linux [several distro, Windows]
- Middleware and Applications Grid Globus/LCG
- for T time

- at the end of time T hosts will be erased!!!

Thanks