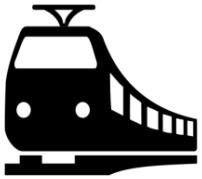
# A tutorial on high availability clusters


© Mário Antunes
mario.antunes@ipleiria.pt

# Fundamentals

**Public businesses**

**Unaceptable outage**

**Uninterrupted services**

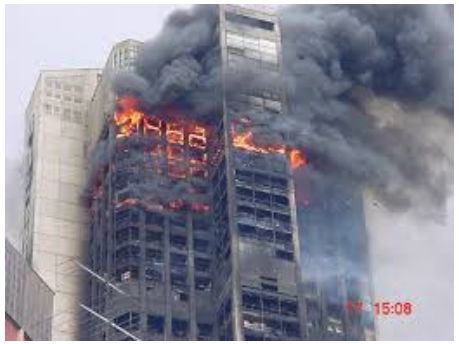

**increasingly based on IT**





mario.antunes@ipleiria.pt

# Fundamentals




**Negative impact in companies' profit and business**

**Unaceptable in E-commerce and E-business companies**

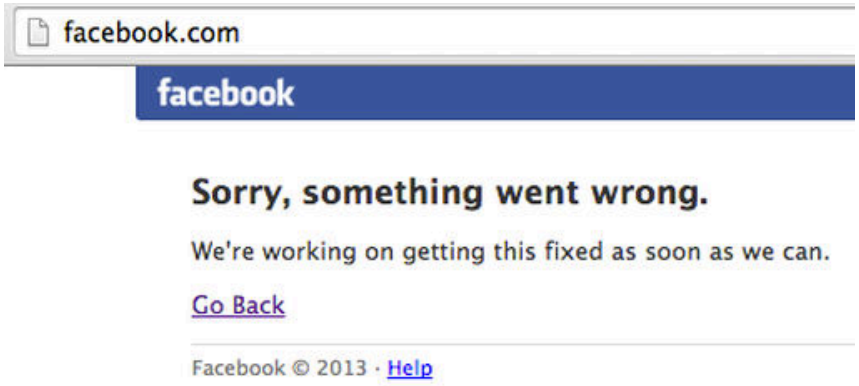
**Colateral damages in global companies and economies**

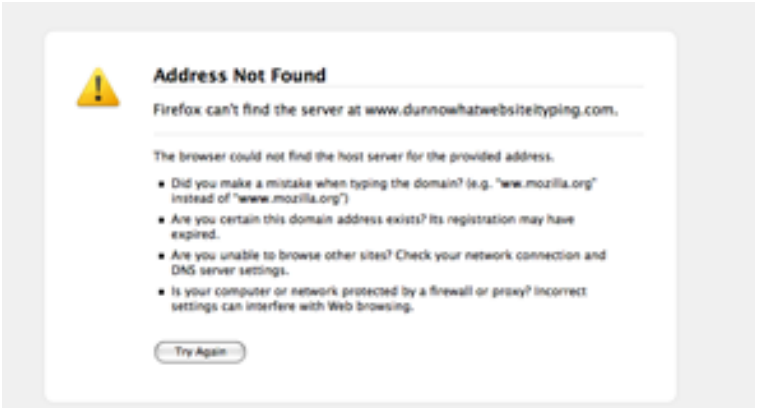
**Business Continuity and Disaster Recovery**



mario.antunes@ipleiria.pt

# Fundamentals

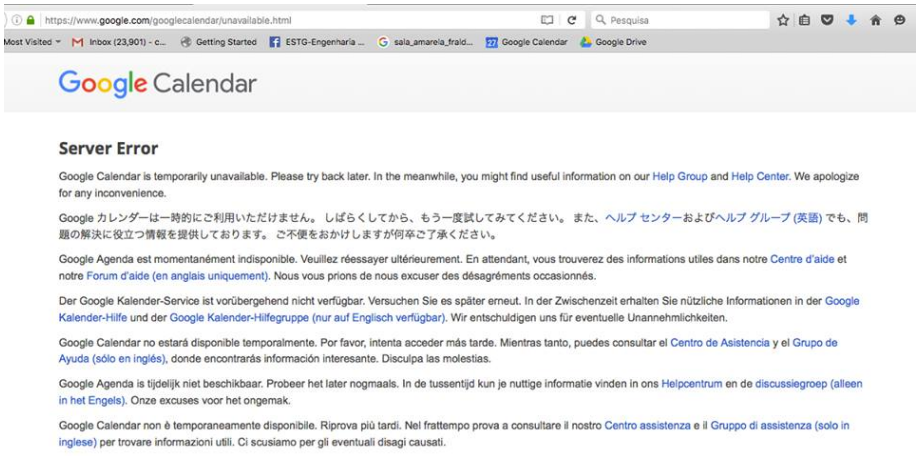

**Downtime**
**Outage**

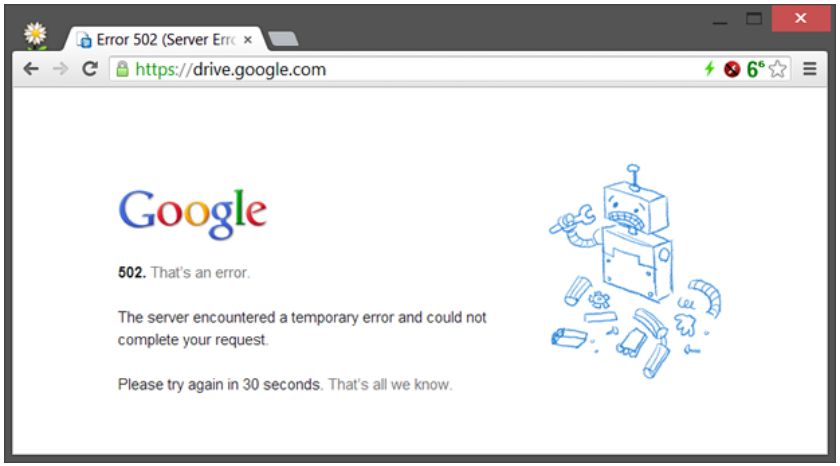
**Uptime**
**Availability**

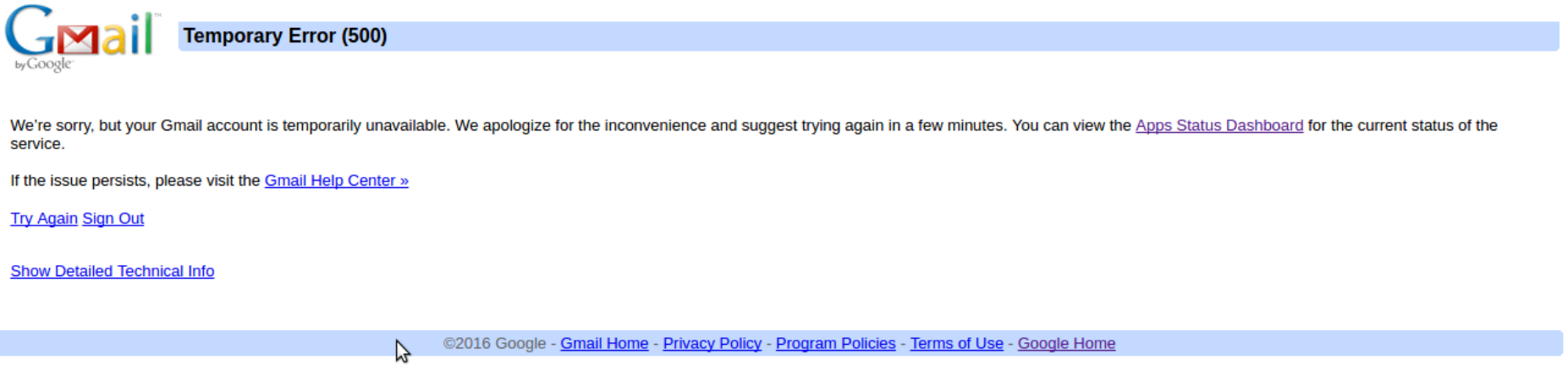


mario.antunes@ipleiria.pt

# Fundamentals




**http://www.google.com/appsstatus**


© Mário Antunes
mario.antunes@ipleiria.pt

# Fundamentals – a definition for availability

- Measure the **percentage** of time (hour, day, month) in which the **service is available** for the end-user.
- Should be **globally analysed** and not by the components separately.
- Should be measured in a **user perspective**.
- Its value has positive (or negative) impact in the business.



mario.antunes@ipleiria.pt

# Fundamentals – a definition for availability




mario.antunes@ipleiria.pt

# Fundamentals – a definition for availability

$$D = \frac{\text{Uptime}}{\text{Operation}} \qquad \text{with } D \in [0,1]$$

Uptime = duration of uptime period (h, min, week, day, month,...)

Operation = duration of total period of operation (h, min, week, ...)

**Example:** Availability in a week (168h) for a uptime of 165h.

$$D = \frac{165}{168} = 0.9821 = 98.21\%$$



mario.antunes@ipleiria.pt

# Fundamentals – a definition for availability

| | Per Hour | Per Day | Per Week | Per Year |
|---|---|---|---|---|
| 99.999% | .0006 | .01 | .10 | 5 |
| 99.98% | .012 | .29 | 2 | 105 |
| 99.95% | .03 | .72 | 5 | 263 |
| 99.90% | .06 | 1.44 | 10 | 526 |
| 99.70% | .18 | 4.32 | 30 | 1577 |

Values for *downtime* (minutes)

**24x60x365 operation:**
- 525600 minutes per year

**24x7x60 operation:**
- 10080 minutes per week

An availability of **99.999%** means 5 minutes of ***downtime*** in a year!

At least three decimal digits should be used (*"The myth of nines"*)

A common metric widely used to assess infrastructure effectiveness



mario.antunes@ipleiria.pt

# Fundamentals – mean availability

- **Measures for availability (mean time)**

$$\textit{Mean Time Between Failures (MTBF)} = \frac{\text{Operation}}{\text{Total failures}}$$

Mean duration of **component (hardware)** *downtime*.

$$\textit{Mean Time to Repair (MTTR)} = \frac{\Sigma\ \text{Repair}}{\text{Total failures}}$$

Mean time to repair a **component**.

When applied to network services:

- *Mean Time Between Service Outage (MTBSO)*
- *Mean Time To Service repair (MTTSR)*



mario.antunes@ipleiria.pt

# Fundamentals – mean availability

$$\textbf{Mean availability} = \frac{\textbf{MTBF}}{\textbf{MTBF+MTTR}}$$

MTBF=1000h
MTTR=1h

$$\longrightarrow \quad D_{mean} = \frac{1000}{1001} = 99{,}90\%$$

$$\textbf{Mean availability} = \frac{\textbf{MTBSO}}{\textbf{MTBSO+MTTSR}}$$

MTBSO=1000 h
MTTSR=3h

$$\longrightarrow \quad D_{mean} = \frac{1000}{1003} = 99{,}70\%$$

*Annualized Failure Rate (AFR)* $= \frac{365x24}{MTBF} = \frac{8760}{MTBF}$



mario.antunes@ipleiria.pt

# Example: Standard S.M.A.R.T for disks

**S.M.A.R.T. = Self-Monitoring, Analysis and Reporting Tech.**

- Available for HDD and SDD devices
- It detects and reports several disks' fiability indicators
- Goal: to antecipate hardware failures

**In Linux:**

```
smartctl -i /dev/sda
smartctl -a /dev/sda
```



mario.antunes@ipleiria.pt

# Fundamentals – common downtime causes

## Planned

**Proactive maintenance**

Hardware replacement

Hardware upgrade

Software upgrade

Users are usually warned.

## Unplanned

**Reactive response**

Hardware, Natural, Power and network failures

**Web and DB server down**

**DNS server down**

Vulnerabilities exploits

No time to warn users.
Backup strategies should start



mario.antunes@ipleiria.pt

# Fundamentals – downtime costs

## Considering:

- 24x7 operation
- $D_{mean}$ = 98.1%
- Cost of dowtime per hour = 10,000€

**Goal**

↑ MTBF

↓ MTTR

## Then:

- Downtime = 3.192 hours per week
- Total cost of downtime ≈ **31,920€** per week

If $D_{mean}$=99,9% then downtime ≈ 10 minutes (0.166h) = **1660€**

Gain = 31.920€ - 1660€ = **30.260€ per week**!



mario.antunes@ipleiria.pt

# Fundamentals – downtime costs

## Direct

Business activities

Productivity

Intranet and internal processes


## Indirect

Clients' dissatisfaction

Stock options

Negative publicity

Legal processes

Company reliability

External reputation

Worker's performance impact



mario.antunes@ipleiria.pt

# Fundamentals – downtime costs

- How does it cost to reduce downtime from 3.2h to 10 minutes?
- What if we reduce  from 3.2h to 1 h?
- The benefits worth the investiment?

**What is the ROI on applying the measures?**

- Reduce the risk (**R**) before (**b**) and after (**a**) the investment
- Assess potential gains (**G**) after the investiment (**I**)

$$G = R_b - R_a$$

$$ROI = \frac{G - I}{I}$$ , being I the cost of preventive measures



mario.antunes@ipleiria.pt

# Fundamentals – calculating the risk

- Probabilility (**P**) of an event
- Duration (**D**) of the downtime event
- Impact (**I**), means the percentage of affected users

The effect of an event ($\mathbf{E_x}$) is measured by its **value** before **(b)** and after **(a)** of applying the preventive measures.

$$E_{ax} = P_{ax} \times D_{ax} \times I_{ax}$$

$$E_{dx} = L_{dx} \times D_{dx} \times I_{dx}$$

Downtime cost

Implementation cost

$$R_{before} = C_D \times (E_{b1} + E_{b2} + \ldots + E_{bx})$$

$$R_{after} = I + (C_D \times (E_{a1} + E_{a2} + \ldots + E_{ax}))$$



mario.antunes@ipleiria.pt

# Fundamentals – calculating the risk

$$G = R_a - R_d \qquad \mathbf{R_a > R_d}$$

$$G = C_D \times (E_{a1} + E_{a2} + \ldots + E_{ax}) - I + (C_D \times (E_{d1} + E_{d2} + \ldots + E_{dx})) \qquad \mathbf{G > 0}$$

$$\mathbf{ROI = \frac{G - I}{I} \%}$$ (annual basis or per cycle)

**Examples:**

Marcus E, Stern H., *"Blueprints for high availability"*; 2003; *Wiley;* ISBN: 0471430269; **pp. 42-46**



mario.antunes@ipleiria.pt

# Fundamentals – calculating the risk

## Calculating the effect before applying the measures:

**Table 3.2** Effects of Outages before Clustering Software Is Installed

| OUTAGE TYPE | BEFORE DURATION (D) | BEFORE LIKELIHOOD (L) | BEFORE IMPACT (I) | BEFORE EFFECT (D × L × I) |
|---|---|---|---|---|
| Crash and reboot | 60 minutes* | 10† | 100% | $E_{B1}$ = 600 (during the day) |
| Crash and reboot (off-hours) | 120 minutes‡ | 10† | 75%§ | $E_{B2}$ = 900 |
| Scheduled reboot | 30 minutes | 60†† | 50%§ | $E_{B3}$ = 900 |
| Motherboard or other major hardware failure | 24 hours (1,440 minutes)¶ | 2 | 100% | $E_{B4}$ = 2,880 |
| Network card failure | 4 hours (240 minutes)** | 2 | 100% | $E_{B5}$ = 480 |
| Application failure | 60 minutes | 20†† | 100% | $E_{B6}$ = 1,200 |
| Scheduled maintenance | 4 hours (240 minutes)‡‡ | 20 | 50% | $E_{B7}$ = 2,400 |
| Failover testing | 0§§ | 0 | 0 | $E_{B8}$ = 0 |
| **Total effect of outages:** | | | | 9,360 minutes (99.644 percent availability over 5 years) |

Marcus E, Stern H., *"Blueprints for high availability";* 2003; *Wiley;* ISBN: 0471430269;



mario.antunes@ipleiria.pt

# Fundamentals – calculating the risk

## Calculating the effect after applying the measures:

**Table 3.3** Effects of Outages after Clustering Software Is Installed

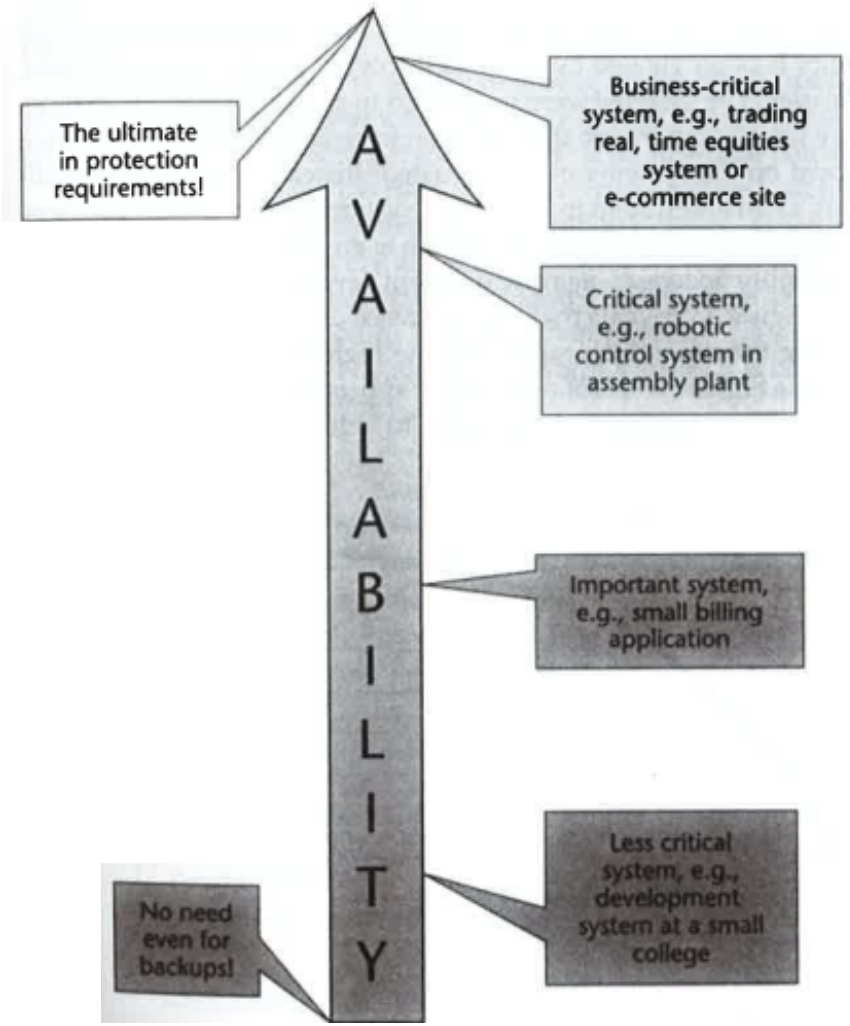
| OUTAGE TYPE | AFTER DURATION (D) | AFTER LIKELIHOOD (L) | AFTER IMPACT (I) | AFTER EFFECT (D × L × I) |
|---|---|---|---|---|
| Crash and reboot | 5 minutes† | 10 | 100% | $E_{A1} = 50$ (during the day) |
| Crash and reboot (off-hours) | 5 minutes† | 10 | 75% | $E_{A2} = 37.5$ |
| Scheduled reboot | 5 minutes | 60 | 50% | $E_{A3} = 150$ |
| Motherboard or other major hardware failure | 5 minutes | 2 | 100% | $E_{A4} = 10$ |
| Network card failure | 2 minutes, then 5 minutes‡ | 2 | 100% then 50% | $E_{A5} = 9$ |
| Application failure | 3 minutes§ | 20 | 100% | $E_{A6} = 60$ |
| Scheduled maintenance | 5 minutes | 20 | 50% | $E_{A7} = 50$ |
| Failover testing | 5 minutes‖ | 20 | 50% | $E_{A8} = 50$ |
| **Total effect of outages:** | | | | 416.5 minutes (99.984 percent availability over 5 years) |

Marcus E, Stern H., *"Blueprints for high availability";* 2003; *Wiley;* ISBN: 0471430269;



mario.antunes@ipleiria.pt

# Fundamentals – calculating the risk




Hospitals

Aviation systems

Other highly critical systems

Web-based businesses

Indistrial-based businesses

Universities and schools

Other less critical businesses

Small businesses

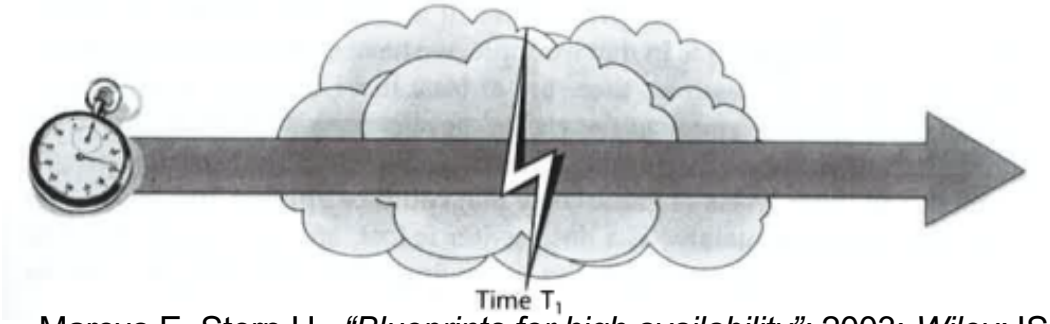
Inactive computers

Marcus E, Stern H., *"Blueprints for high availability";* 2003; *Wiley;* ISBN: 0471430269;



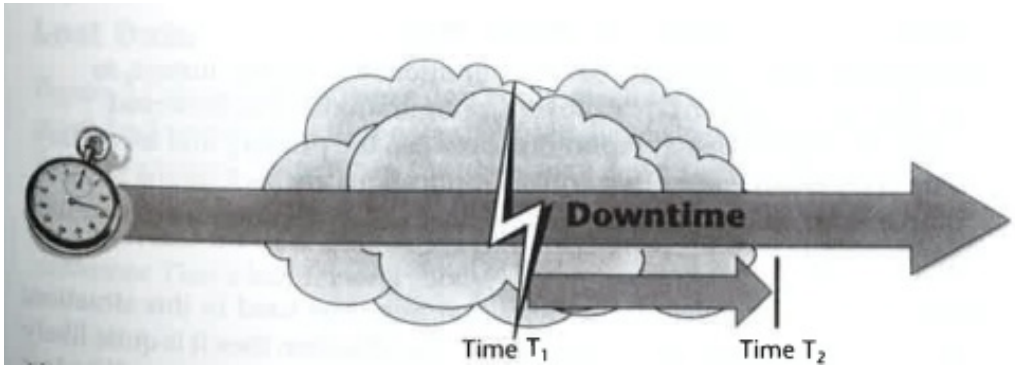
mario.antunes@ipleiria.pt

# Outage lifecycle




Begin of an outage

Marcus E, Stern H., *"Blueprints for high availability";* 2003; *Wiley;* ISBN: 0471430269;



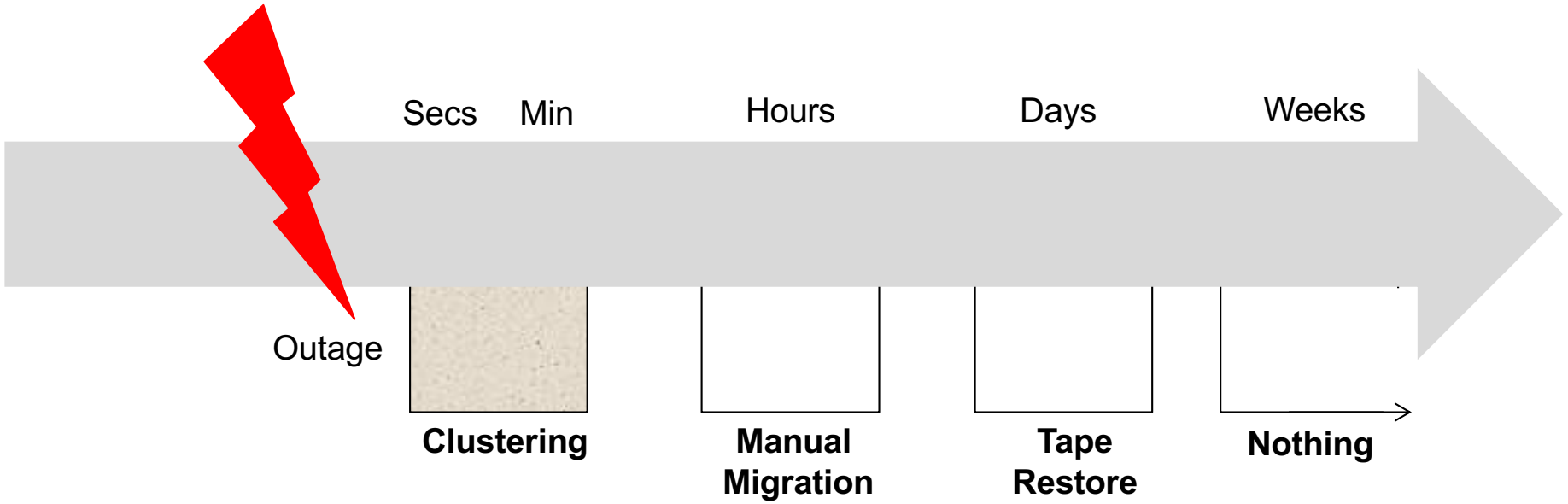

$[T_1; T_2]$ - Downtime

Marcus E, Stern H., *"Blueprints for high availability";* 2003; *Wiley;* ISBN: 0471430269;



mario.antunes@ipleiria.pt

# Outage lifecycle – outage mitigation

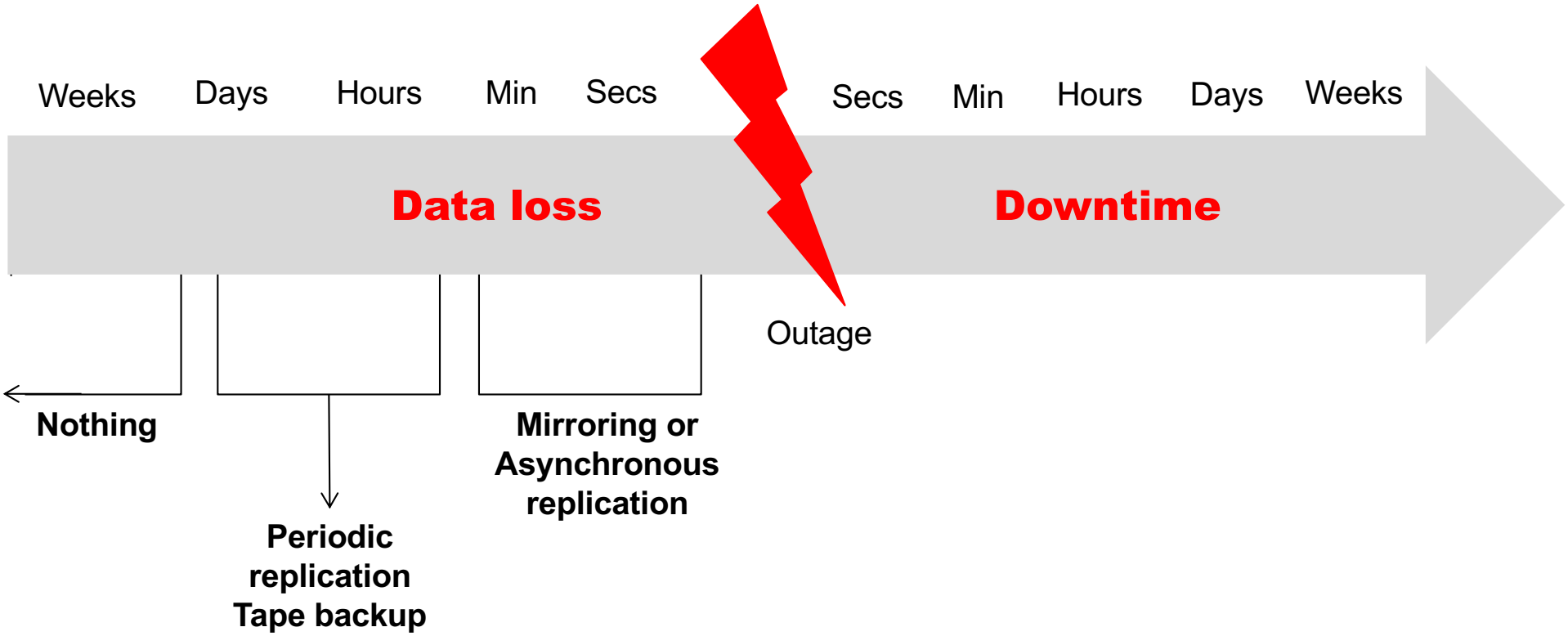



**Time is an important variable**

**Downtime will happen in the presence of an outage**

**Goal: to mitigate outage in a minimim period of time**

# Outage lifecycle – data loss mitigation




**Time is again an important variable**

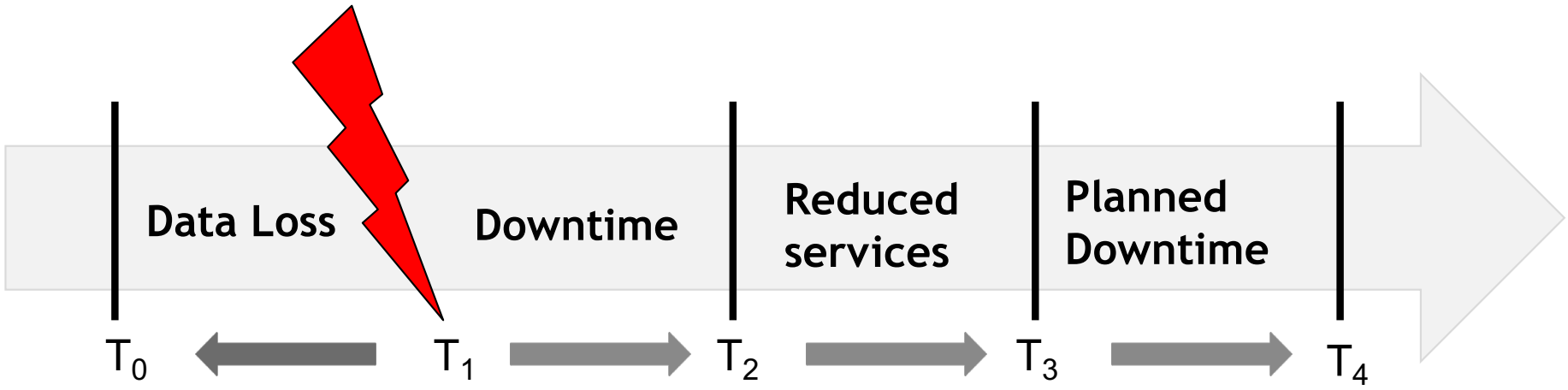
**Outage will cause data loss**

**Goal: to minimize data loss and to assure system consistency**



mario.antunes@ipleiria.pt

# Outage lifecycle




HA implementation is a business decision.

How much does it cost the infrastructure downtime?

How much does it cost to implement HA procedures to mitigate downtime?



mario.antunes@ipleiria.pt

# Impact on the organization

- To define clear and tangible goals
- To define a physical environment
- To automate processes

- To define environment for testing and QA
- To create a stock of physical components
- To adjusts contracts for critical components

- To schedule processes
- To plan catastrophic scenarios
- To train
- To document everything!



mario.antunes@ipleiria.pt

# Impact on the organization - vulnerabilities

| VULNERABILITY | LIKELIHOOD (1-3) | IMPACT (1-3) | LEVEL OF CONCERN (LIKELIHOOD * IMPACT) | COMMENTS |
|---|---|---|---|---|
| Failed disk | 3 | 1 | 3 | Critical systems already have mirrored disks. |
| Blown CPU | 2 | 2 | 6 | Systems are not clustered; a failed CPU will result in major downtime for one server. |
| Database corruption | 2 | 3 | 6 | Corruption in a critical database could shut down the web site for hours. |
| Unreadable backups | 1 | 2 | 2 | Only an issue if we lose data from our disks. Then it could be quite serious. |
| Network component failure | 2 | 2 | 4 | We have a very complex network, with many single points of failure. |
| Data center fire | 1 | 3 | 3 | Could cause the loss of our entire computing infrastructure. |
| Extended power outage | 2 | 2 | 4 | Won't damage data, but could keep us down for a long time. |
| Flooding | 2 | 3 | 6 | This area has a history of flooding. With the data center in a low floor, results could be catastrophic. |
| Chemical spill | 1 | 3 | 3 | An interstate highway goes within 200 yards of the front door. Always a small risk. |
| Tornado | 1 | 2 | 2 | Would have to hit the building to be a major problem. |
| Earthquake | 2 | 3 | 6 | Can, of course, be a major disaster. |
| Bioterrorism | 1 | 3 | 3 | Unlikely, but if it happened, could be serious. |

Marcus E, Stern H., *"Blueprints for high availability";* 2003; *Wiley;* ISBN: 0471430269;



mario.antunes@ipleiria.pt

# HA design – 20 goals

1. Don’t be cheap (€ £ $)
2. Assume nothing from constructors, trades, etc...
3. Remove single points of failure (SPOF)
4. Enforce security (physical, logical)
5. Redefine number of servers
6. Performance measurements
7. Enforce changes to the configuration



mario.antunes@ipleiria.pt

# HA design – 20 goals

8. Document everything
9. Define and apply SLA
10. Proactive planning
11. Test everything
12. Separate environments (production, test, QA, dev,...)
13. Learn from history
14. Design for growth



mario.antunes@ipleiria.pt

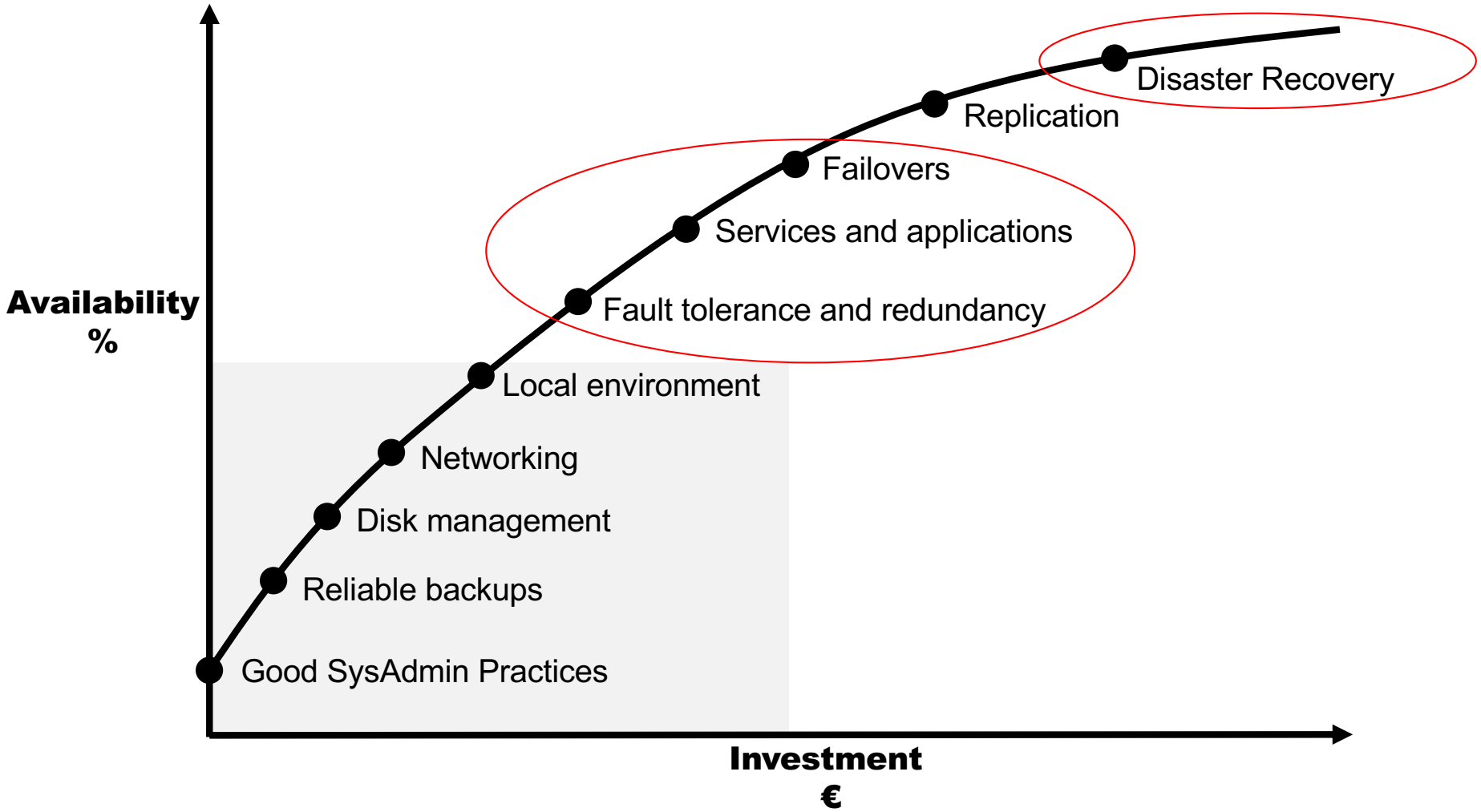
# HA design – 20 goals

15. Choose mature software with support
16. Choose mature and reliable hardware
17. Reuse configurations
18. Resources (papers, BCPs, bibliography, ...)
19. One problem, one solution!
20. “Keep it simple!”



mario.antunes@ipleiria.pt

# Availability index



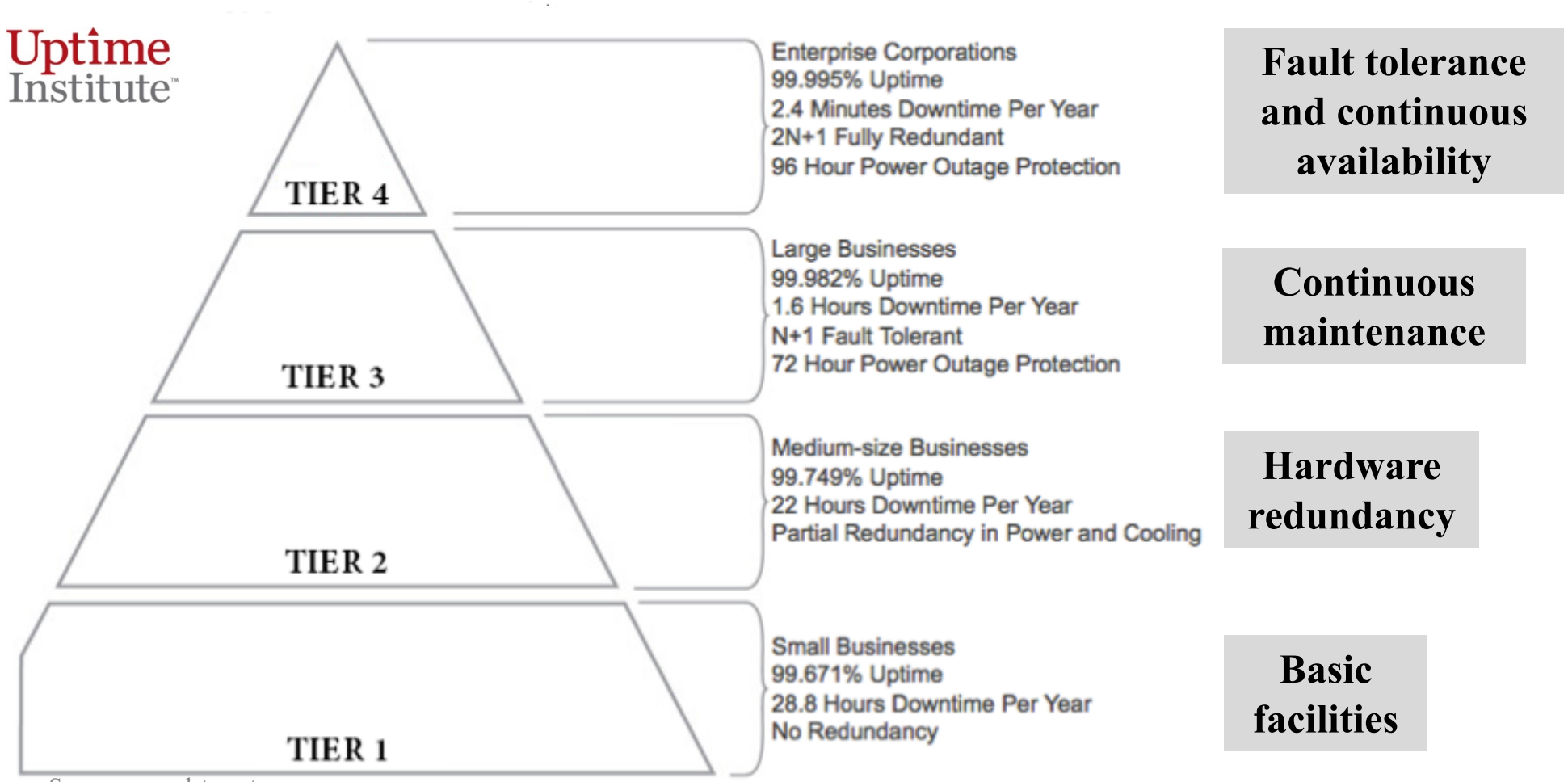

Adapted by Marcus E, Stern H., *"Blueprints for high availability";* 2003; *Wiley;* ISBN: 0471430269;



mario.antunes@ipleiria.pt

# Availability index

## Availability measure is used to evaluate datacenters

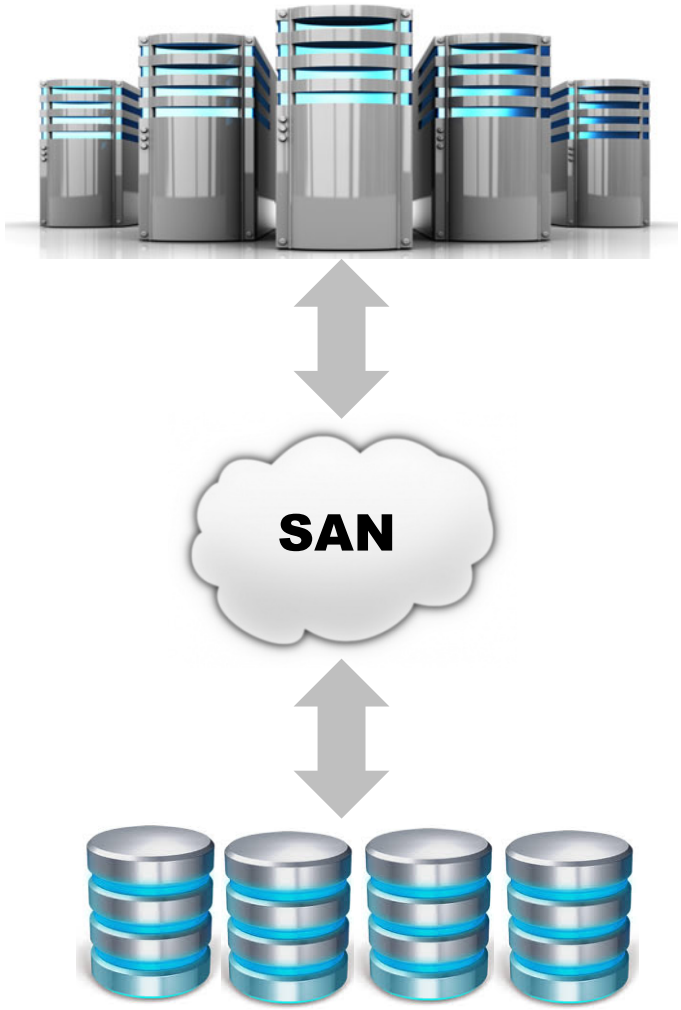



Source: www.datacenters.com



mario.antunes@ipleiria.pt

# Clustering technologies




**Clusters**

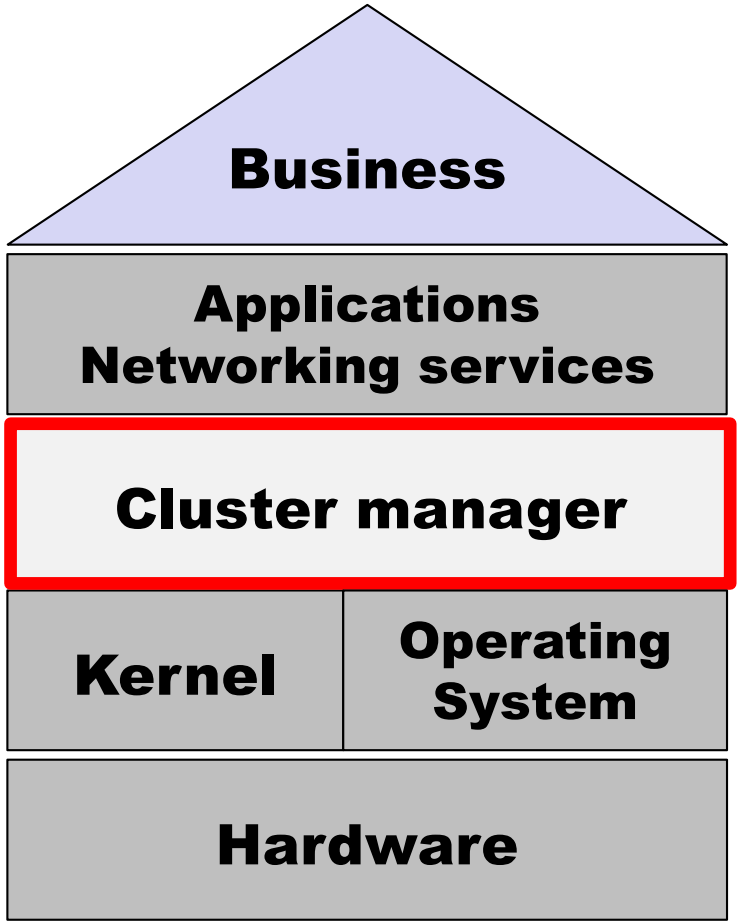
**High Performance (HPC)**

Paralell processing

**High Availability (HAC)**

Failover techniques

# Clustering technologies - HA




## Cluster managers

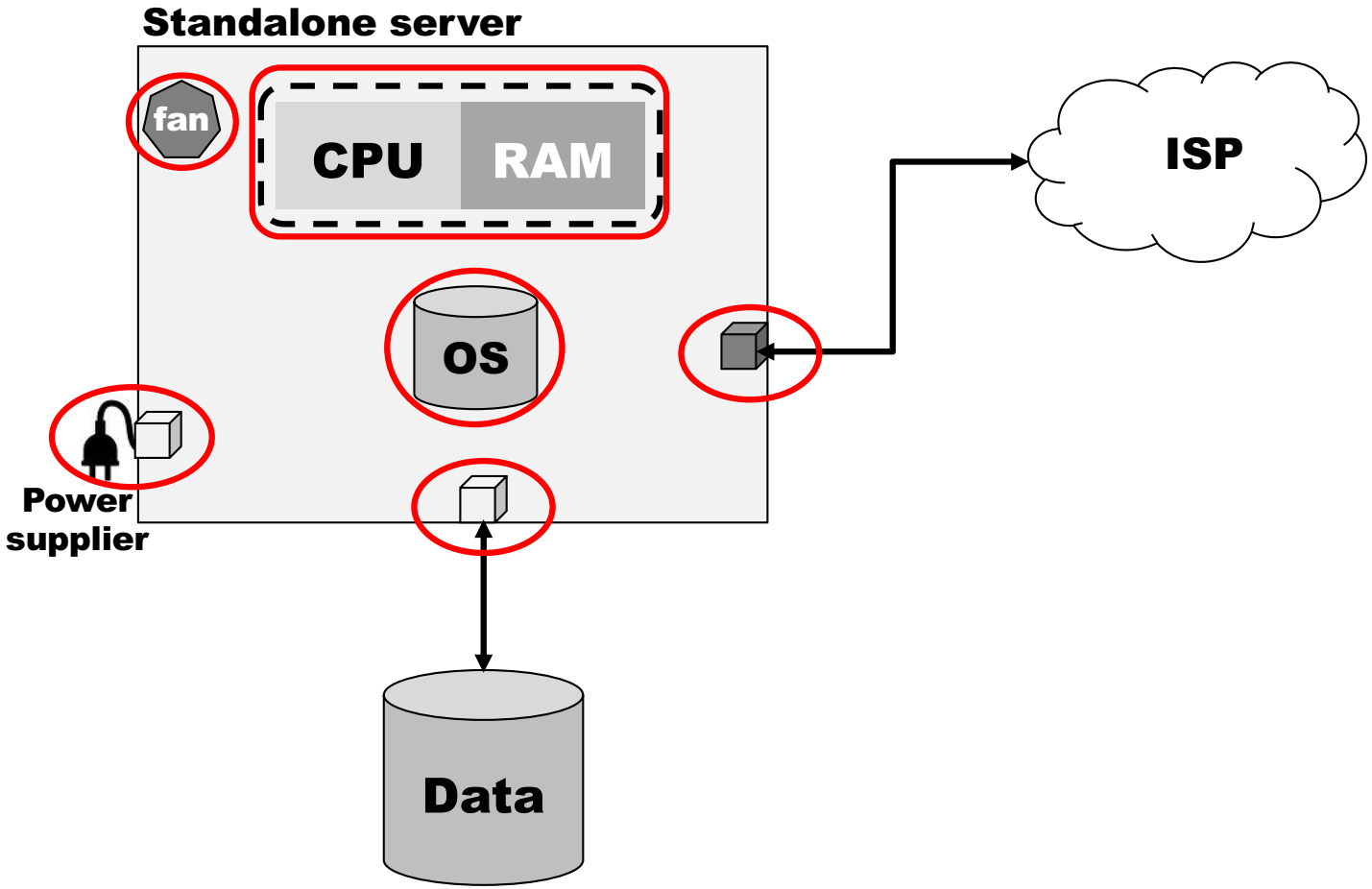
- OS dependent
- Mandatory to be in the market!
- Manufacturers awareness

## Opensource cluster managers

- Mainly for Linux OS
- Linux Virtual Server (LVS)
- Heartbeat: `www.linux-ha.org`

# Clustering technologies – HA

## Identify Single Points of Failure (SPoF)

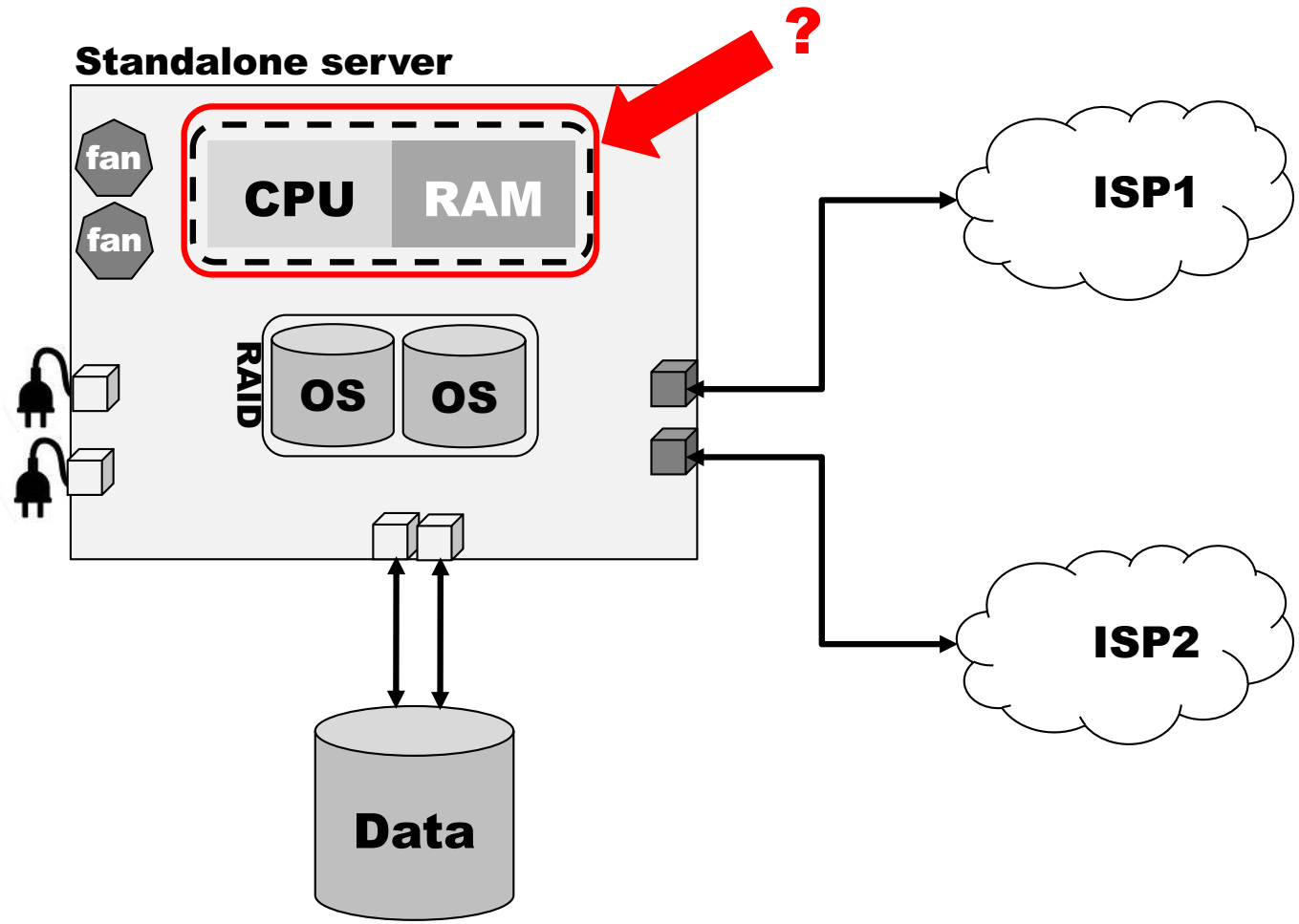





mario.antunes@ipleiria.pt

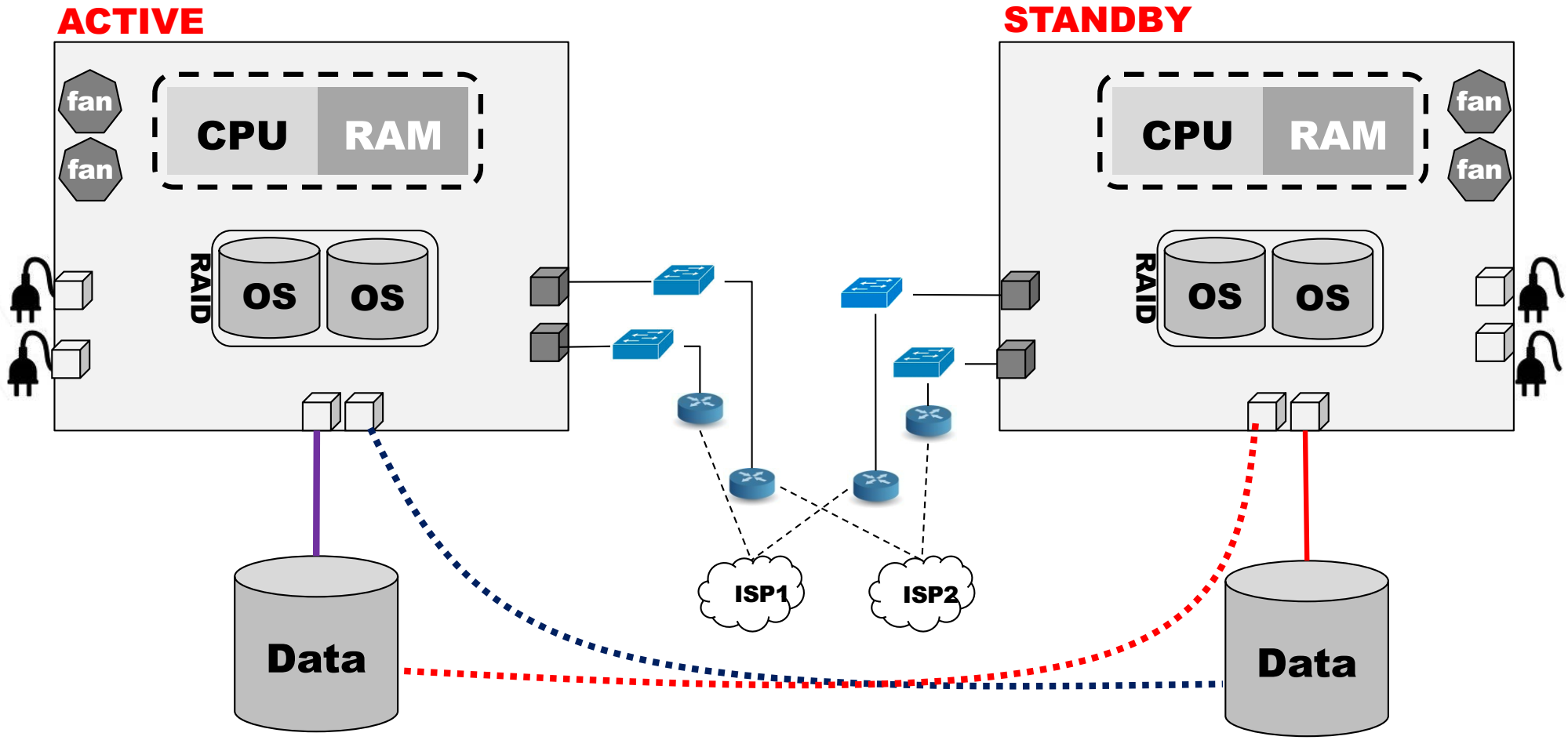
# Clustering technologies - HA

## Eliminate (mitigate) SPoF (1)



# Clustering technologies - HA

## Eliminate (mitigate) SPoF (2)

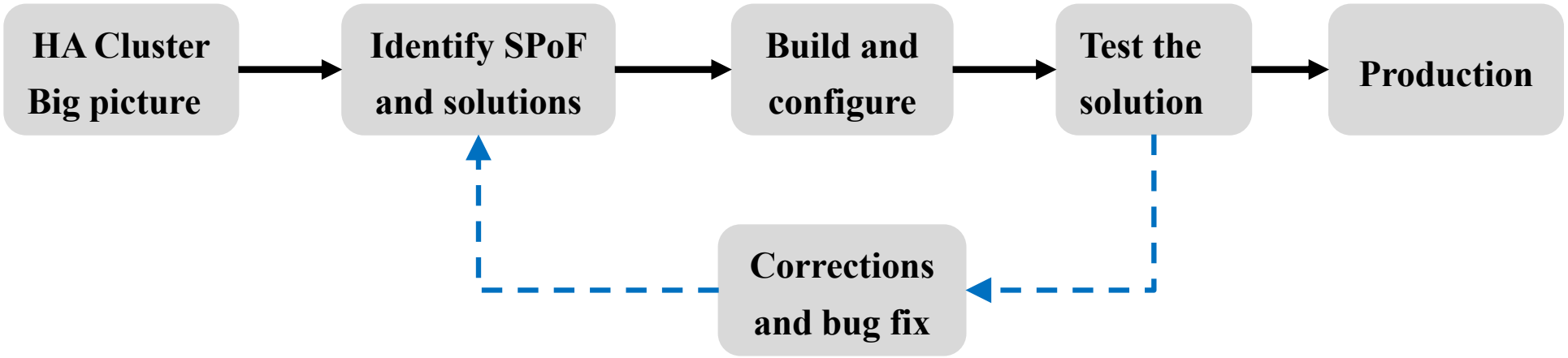





mario.antunes@ipleiria.pt

# Clustering technologies – HA - planning

## General methogology to setup a HA cluster




- **Planning is crucial**
- **Level of availability may required mixed solutions**
- **There is not one solution that fit all problems**
- **Practice ... practice and more practice!**



mario.antunes@ipleiria.pt

# Clustering technologies - HA

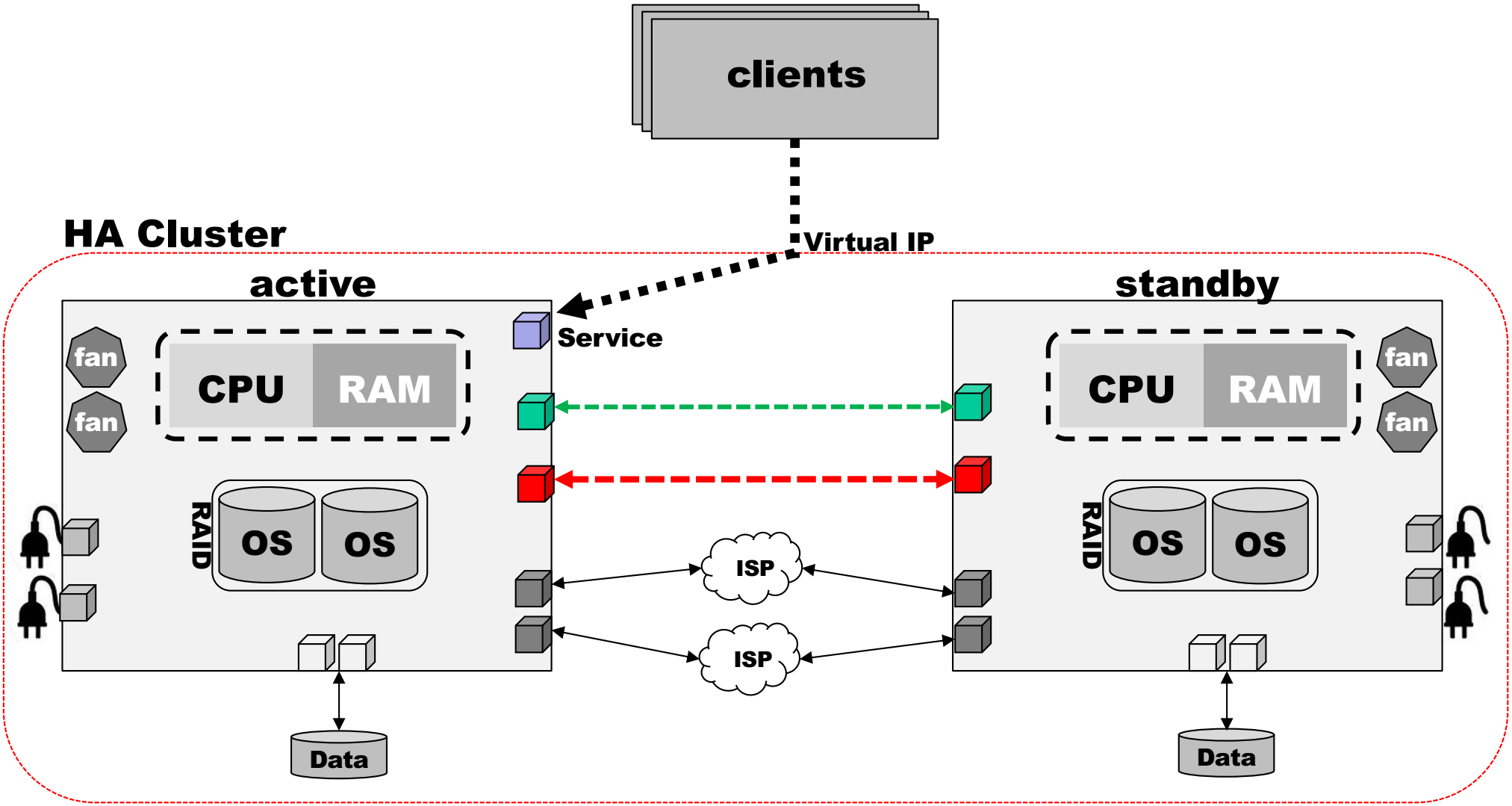
## To have in mind:

- Define attainable goals
- Define and test different types of scenarios
- Different strategies: *"Active-Active"* or *"Active-Standby"*
- Network paths should also be replicated: racks, switches, routers, ISP...
- Applications should be *"HA ready!"*
- Organizations should be *"HA aware"*



mario.antunes@ipleiria.pt

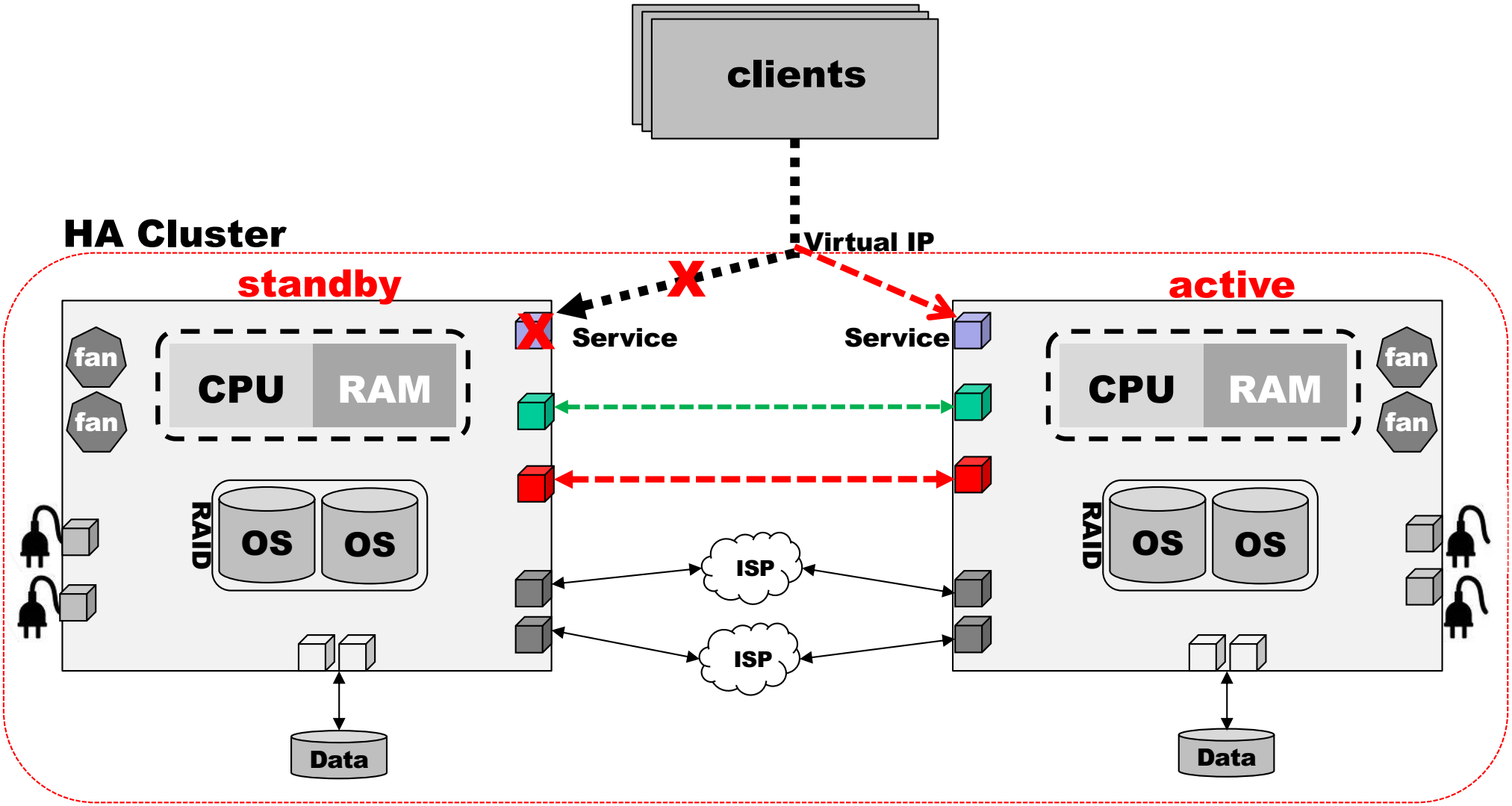
# Cluster Resource Manager






mario.antunes@ipleiria.pt

# Cluster Resource Manager

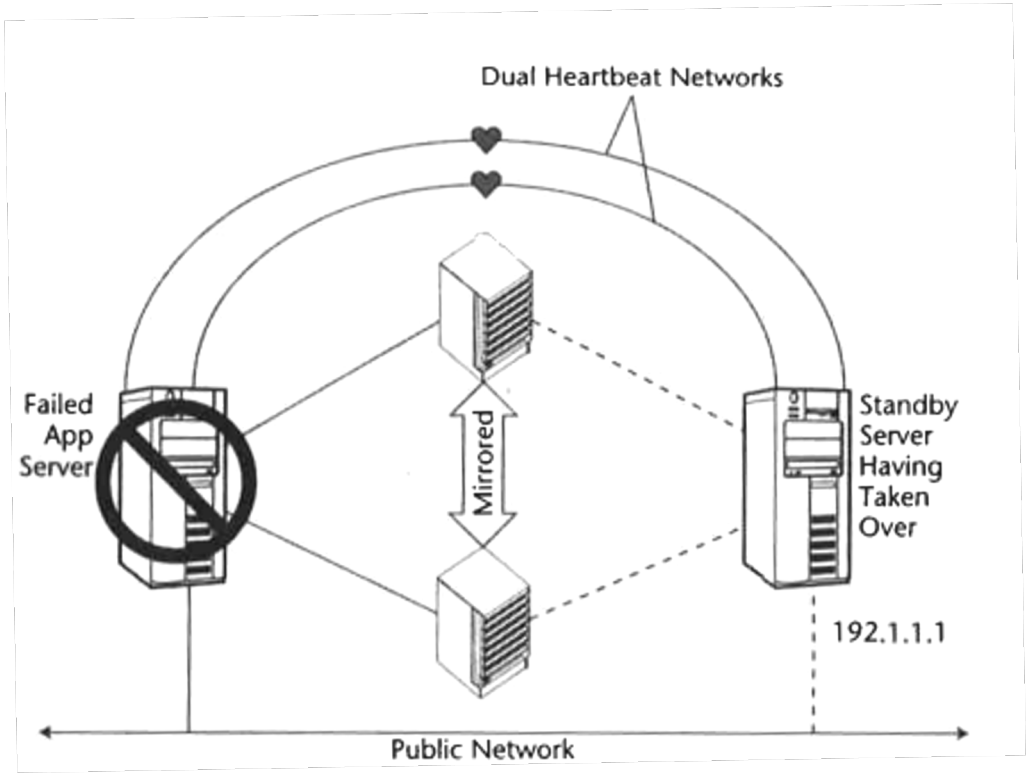





mario.antunes@ipleiria.pt

# Cluster Resource Manager

## "Active-Passive" configuration




- A standby node
- Automatic recover after failure
- heartbeat between two nodes

**How to use standby nodes:**

✓software development
✓Other applications
✓QA and/or test system
✓Solely standby node

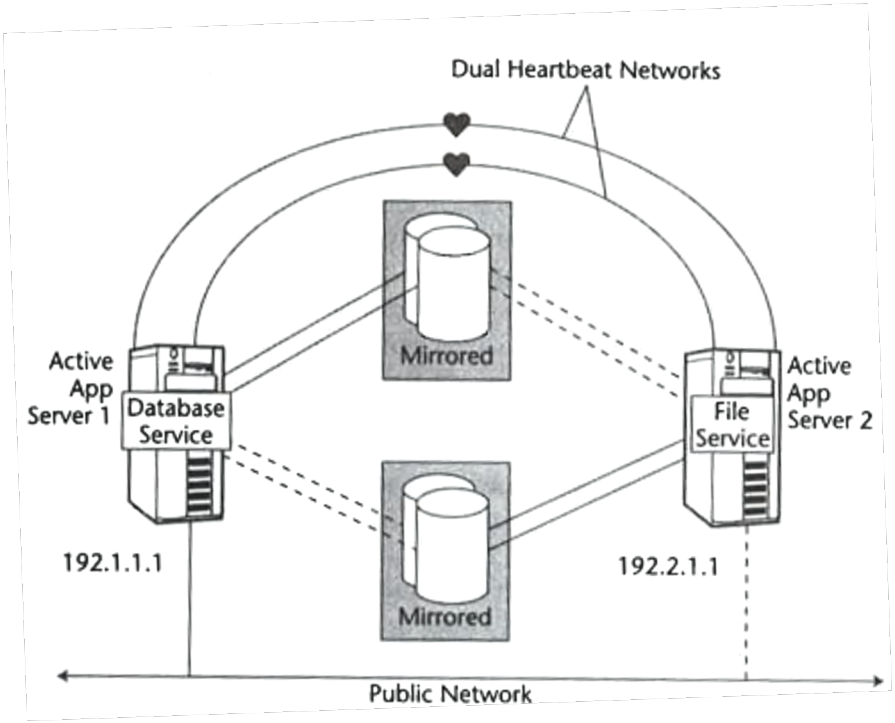
Marcus E, Stern H., *"Blueprints for high availability";* 2003; *Wiley;* ISBN: 0471430269;

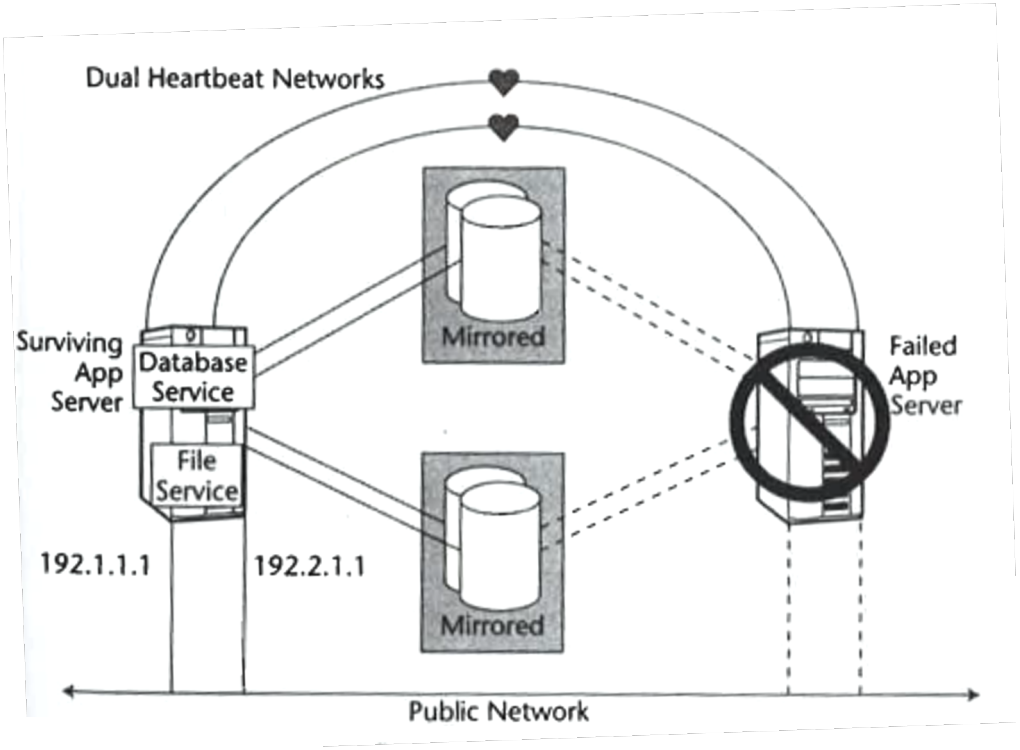


mario.antunes@ipleiria.pt

# Cluster Resource Manager

## "Active-Active" configuration



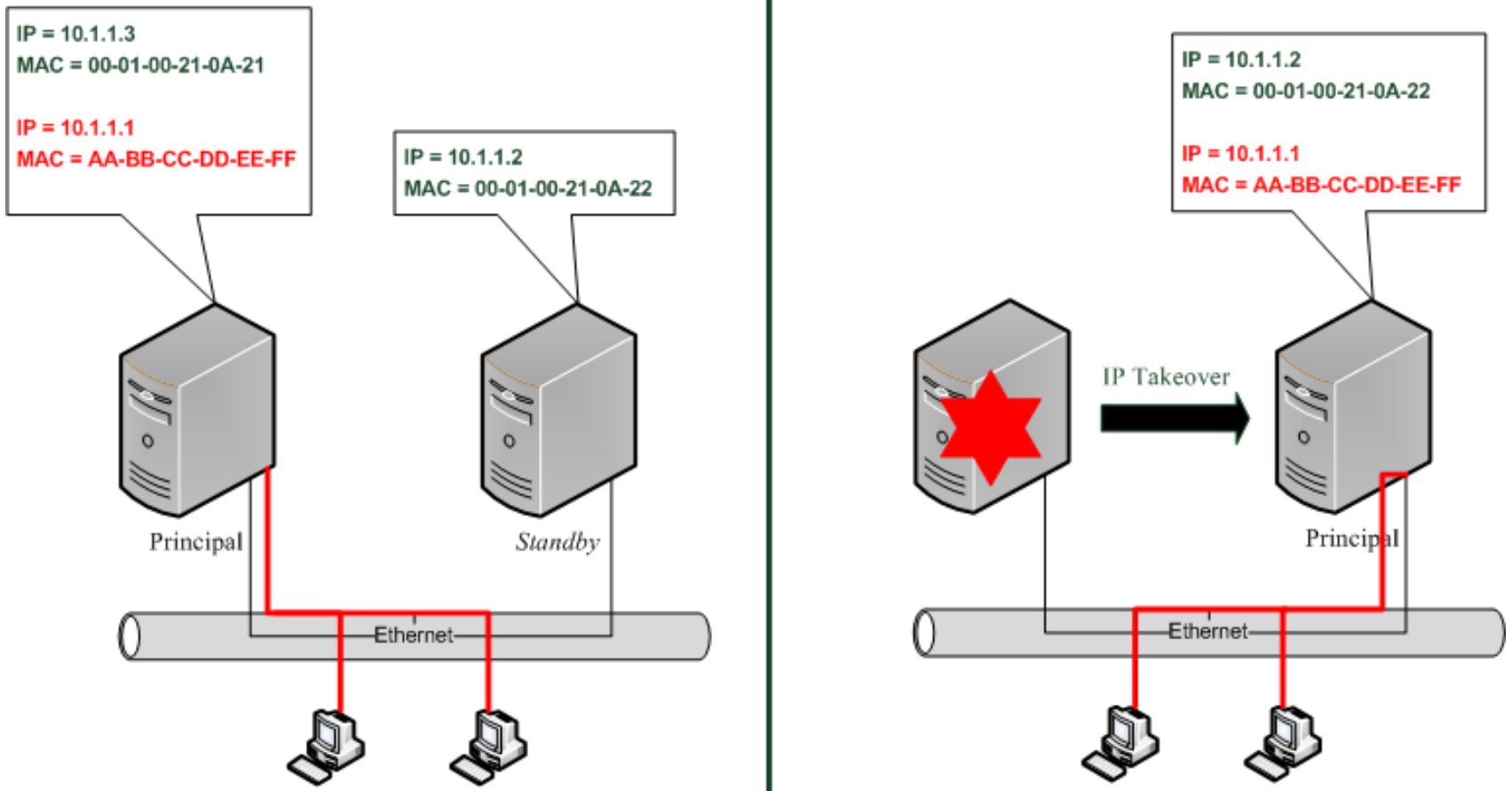

Better use of hardware

Well tested and independent applications
Low performance after takeover
Managament is more complex

Marcus E, Stern H., *"Blueprints for high availability";* 2003; *Wiley;* ISBN: 0471430269;

# Cluster Resource Manager

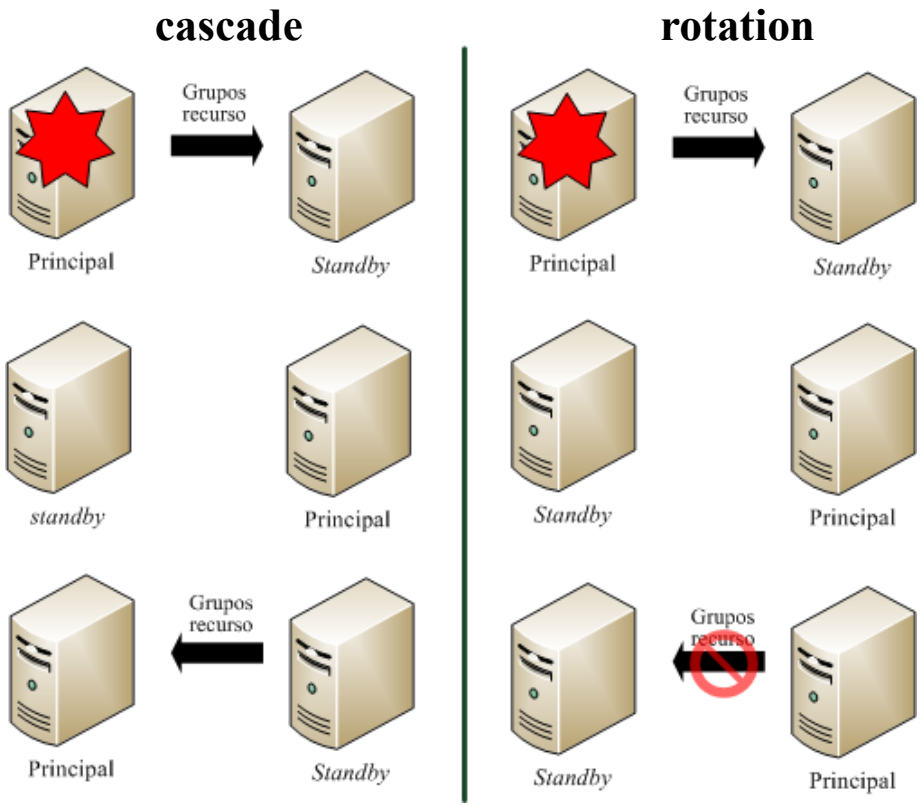
## Address takeover




In *"Gestão de Projetos TI e administração centralizada de sistemas e redes – cenários práticos em contexto empresarial"*; Mário Antunes; IPLeiria; 2010

# Cluster Resource Manager

## Resource groups – takeover strategies




**Resource group**
critical applications
interfaces

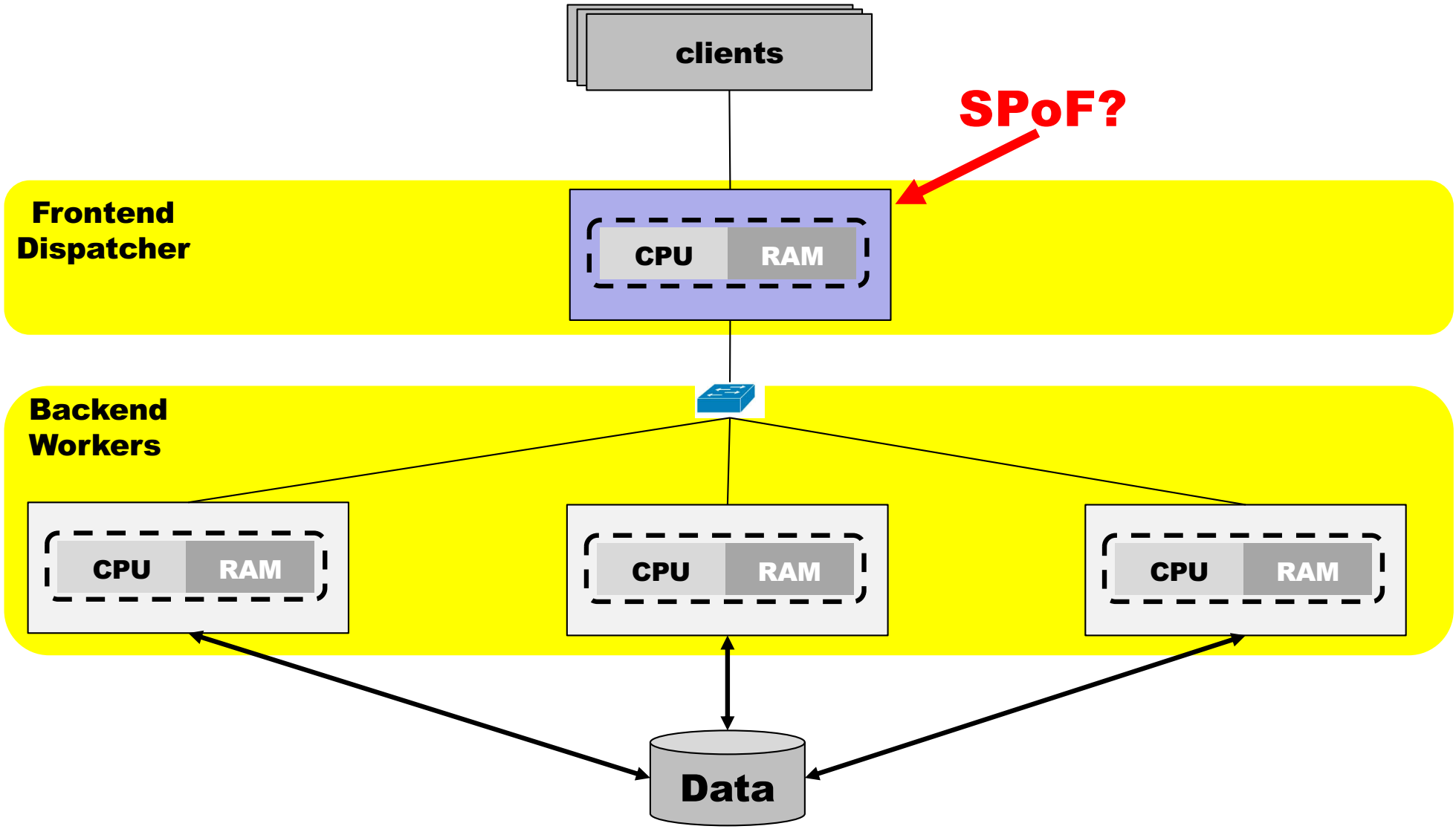
**Which method should we choose?**

In *"Gestão de Projetos TI e administração centralizada de sistemas e redes – cenários práticos em contexto empresarial"*;
Mário Antunes; IPLeiria; 2010

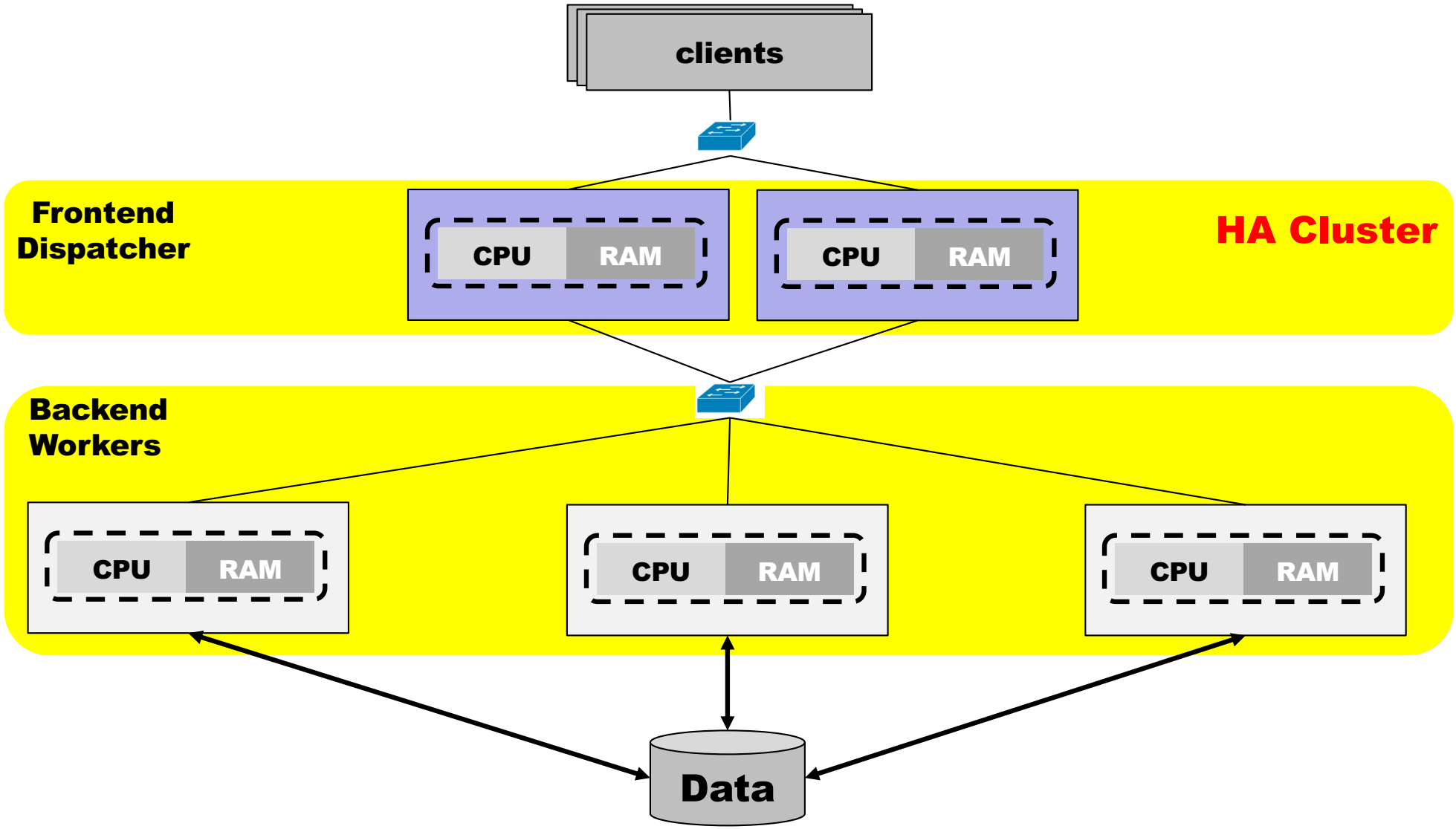


mario.antunes@ipleiria.pt

# Cluster Resource Manager - Load balancing



# Cluster Resource Manager - Load balancing



# Cluster Resource Manager - Challenges

- Geographical distribution of the cluster nodes
- Infrastructure monitoring effectiveness
  - Cluster manager tools
  - Applications monitoring tools
  - Customized monitoring applications
- *"split-brain"* syndrome and how to mitigate it
- How to use backup node during idle period?



mario.antunes@ipleiria.pt

# Final remarks

- Business are highly IT dependent
- Availability is a *must* and a *continuous challenge*
- Companies and businesses are too aware to HA thinking
- Basic concepts to cover core business and IT demands
- Mixed topologies to cover HA and LB features
- Keyword: **investment** ($£€).

IT professionals should be *HA aware*!



mario.antunes@ipleiria.pt

# High availability clusters

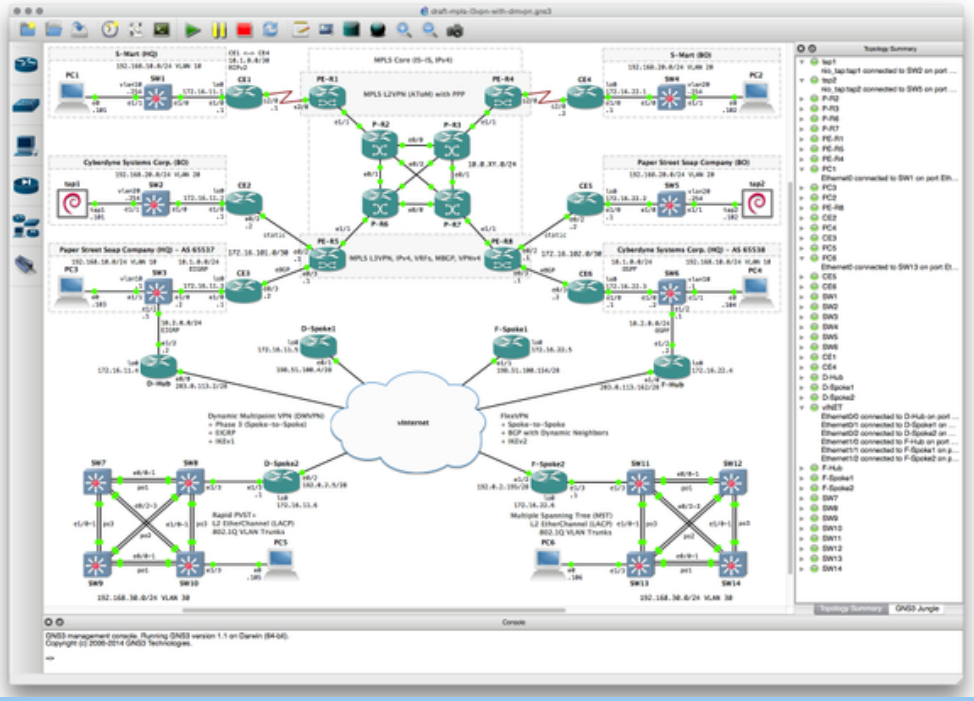
## - case study with Heartbeat -



mario.antunes@ipleiria.pt

# Lab setup – Visualization tool

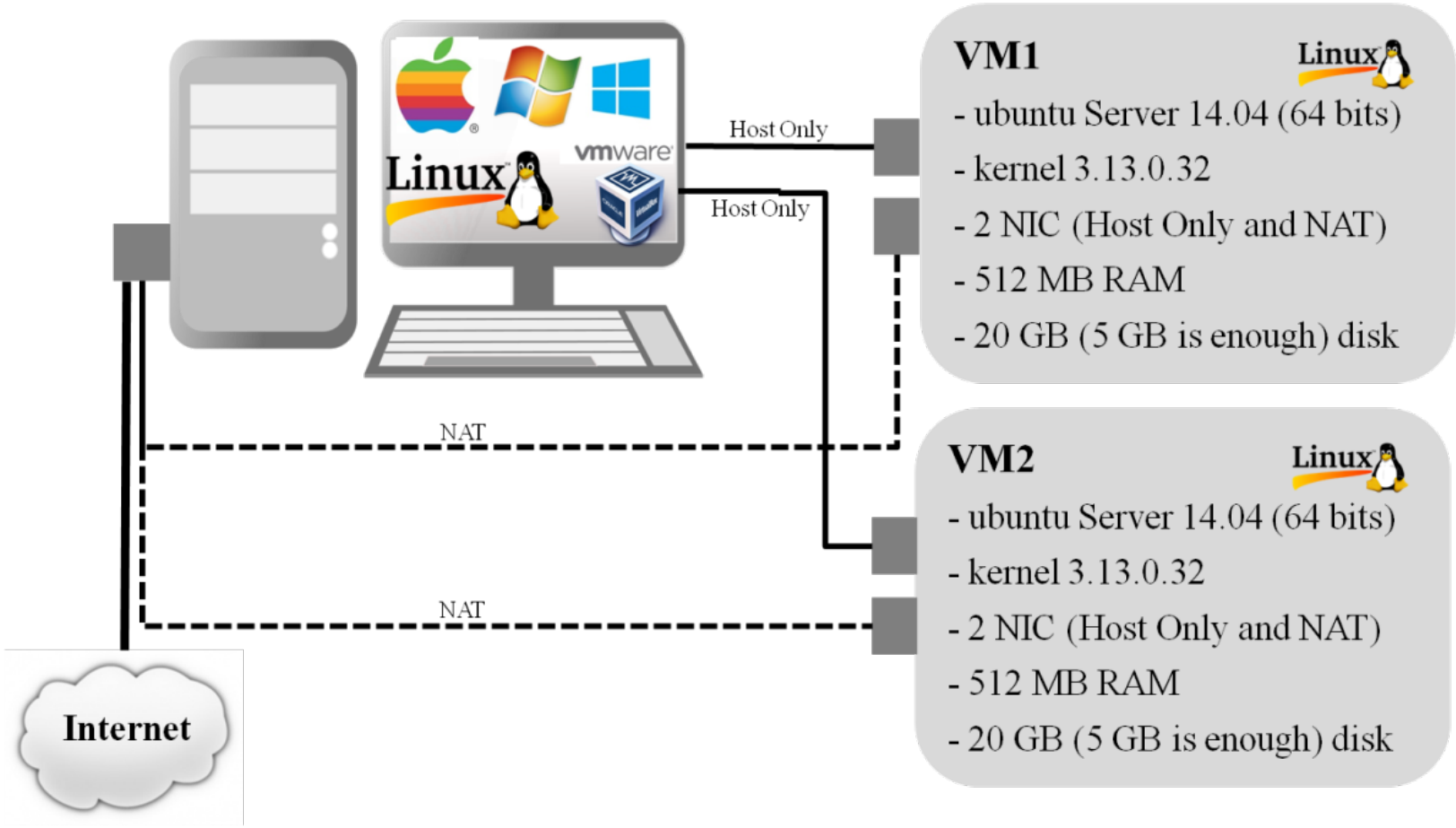
- Virtualization and emulation of network equipments and servers
- Integrates easily with Virtualbox
- Native hypervisor – run `dynamips`
- Graphical and visualization features
- Opensource and free!



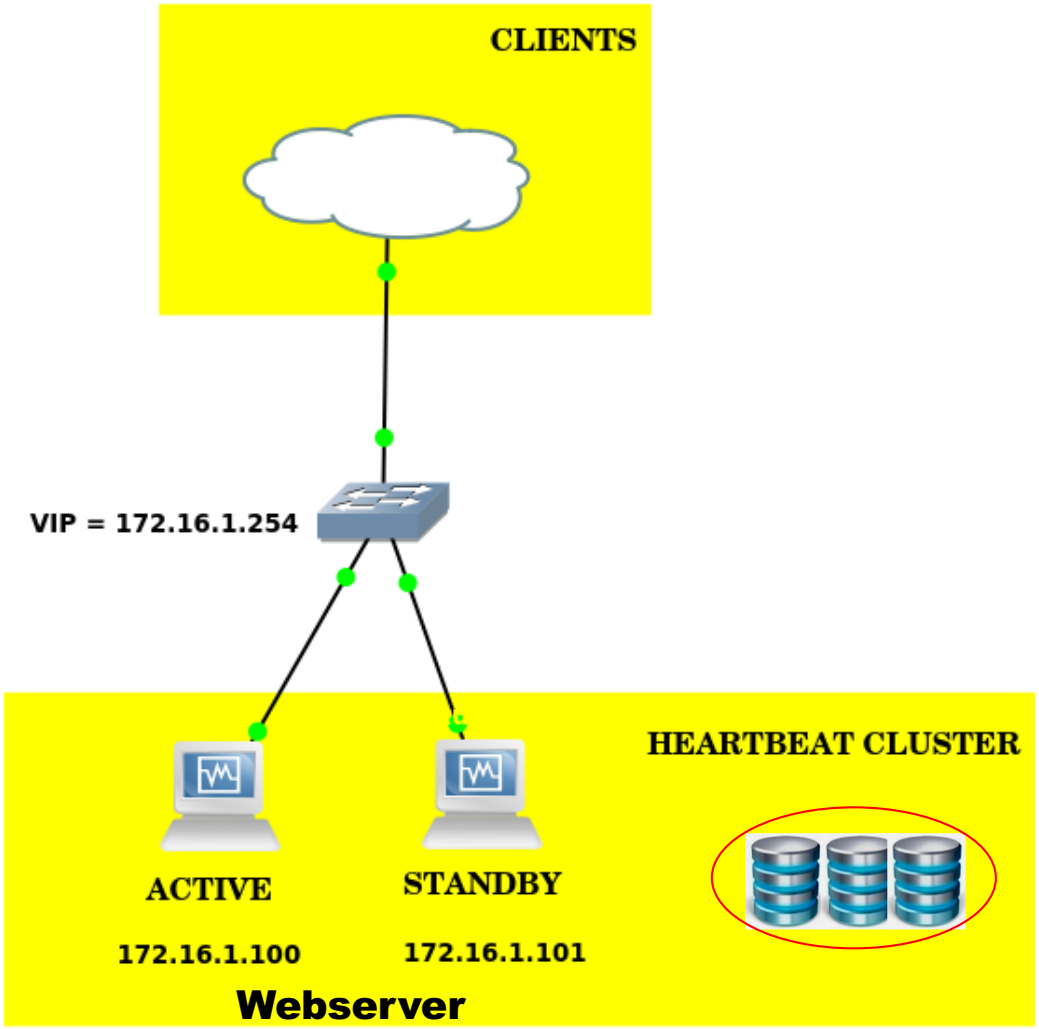
GNS3®
www.gns3.com


© Mário Antunes
mario.antunes@ipleiria.pt

# Lab setup – Visualization tool



# Lab setup - heartbeat




**Network setup**

**Webserver setup**

**Heartbeat setup**

**Logging**

**Config files (`/etc/ha.d/`)**

`authkeys`

`haresources`

`ha.cf`

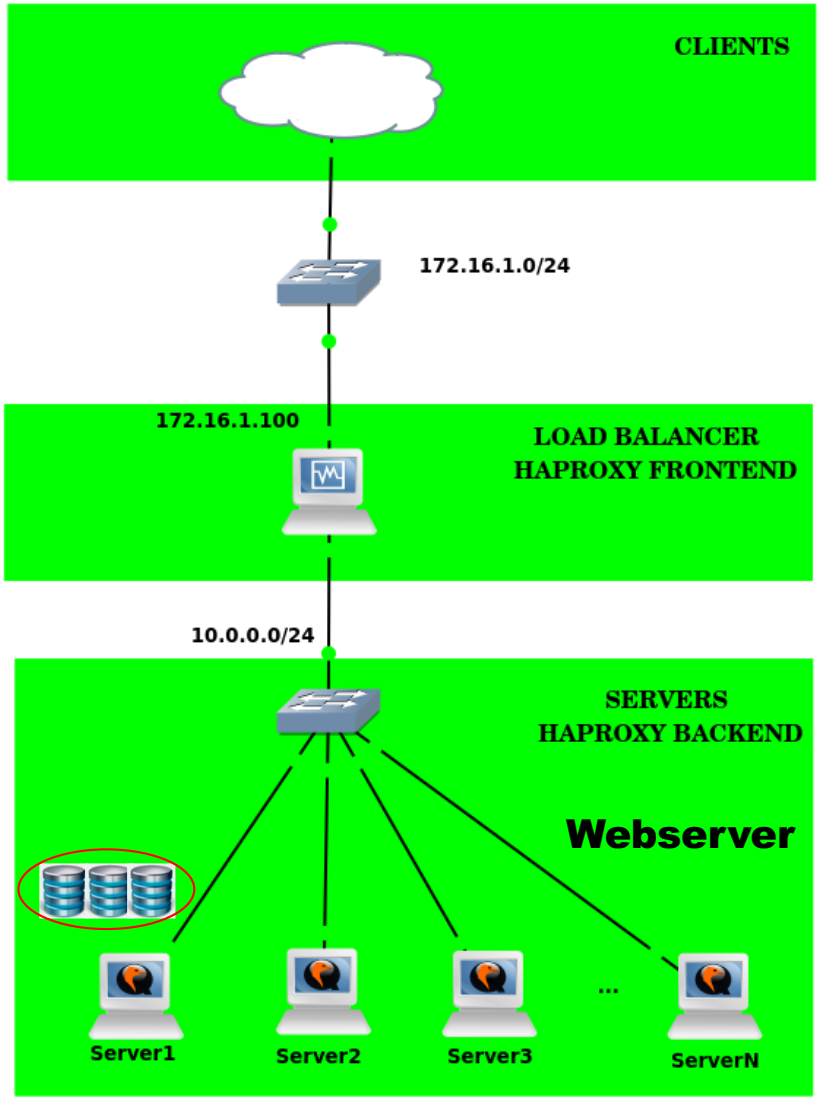
# High availability clusters

## - case study with HAProxy -



mario.antunes@ipleiria.pt

# Lab setup - HAProxy




**Web clients**

- Browser launchs on “host” system

**HAProxy Server**

- Forward Web requests/replies
- Monitoring and configuration rules
- `/etc/haproxy/haproxy.conf`

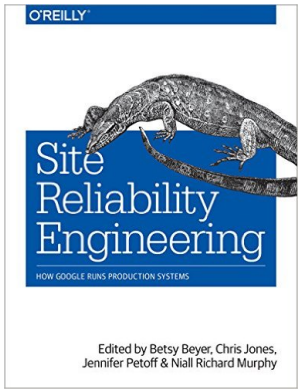
**Web servers**

- Web server configuration rules
- Centralized storage access



mario.antunes@ipleiria.pt

# Final remarks - bibliography




***Site Reliability Engineering***

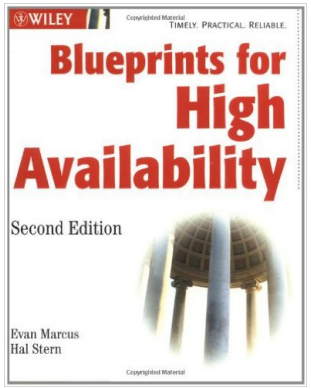
Jennifer Petoff, Niall Richard Murphy, Betsy Beyer, Chris Jones
How Google Runs Production Systems
O'Reilly Media; 1 edition (April 16, 2016)
ISBN-13: 978-1491929124

http://www.wired.com/2016/04/google-ensures-services-almost-never-go/

Uptime Institute™

http://www.uptimeinstitute.com

Reports and research publications


***Blueprints for high availability***

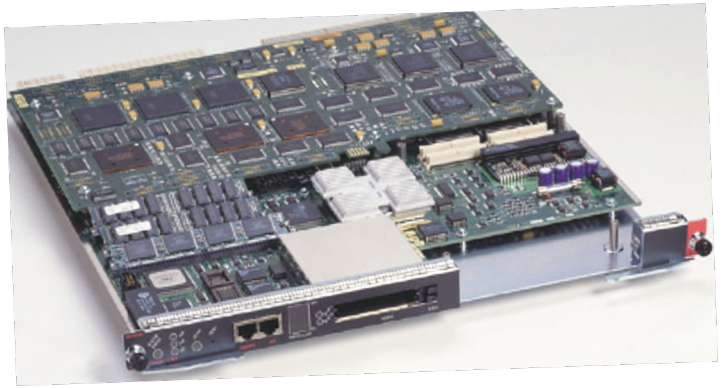
Marcus E, Stern H.
Wiley; 2003
ISBN: 0471430269

# Redundancy
# LB and HA in routing protocols
# Dedicated HA and LB protocols

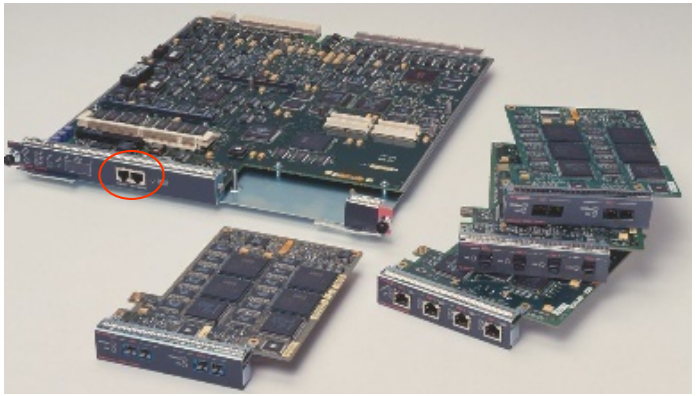
**Mário Antunes**
mario.antunes@ipleiria.pt

# Redundancy

## Hardware level

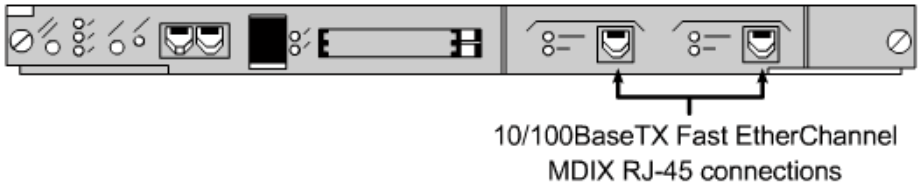


Catalyst 5000,5500, 6000, 6500

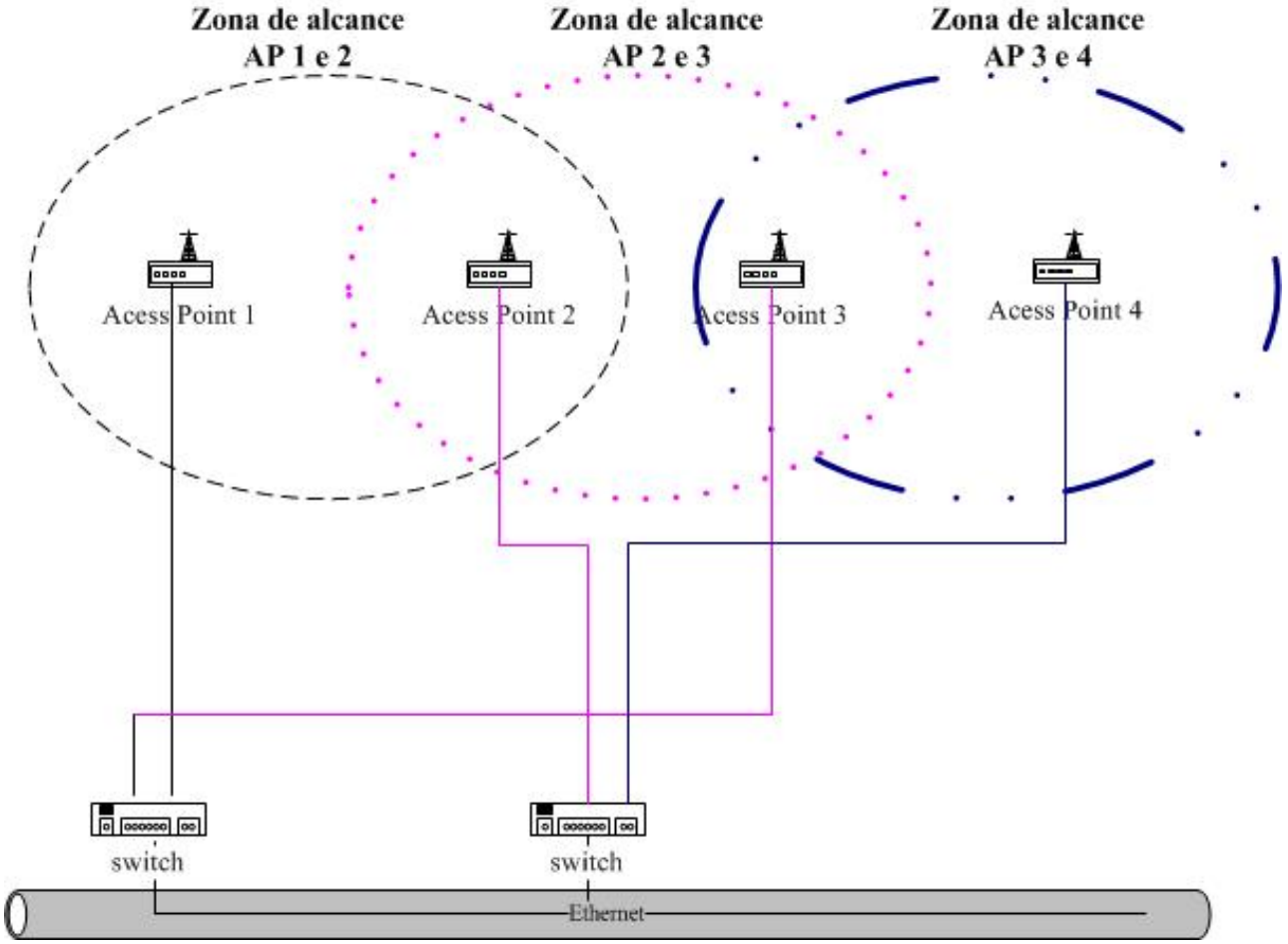


Uplinks redundantes

Power Supply



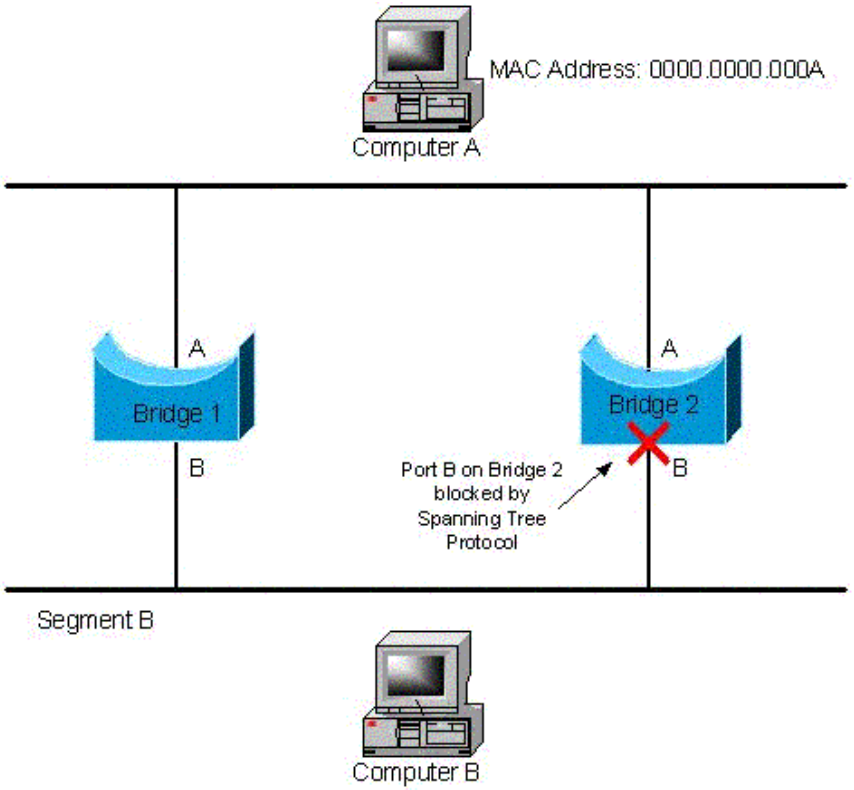
# Redundancy - examples




In *"Gestão de Projetos TI e administração centralizada de sistemas e redes – cenários práticos em contexto empresarial"*; Mário Antunes; IPLeiria; 2010

# Redundancy - examples

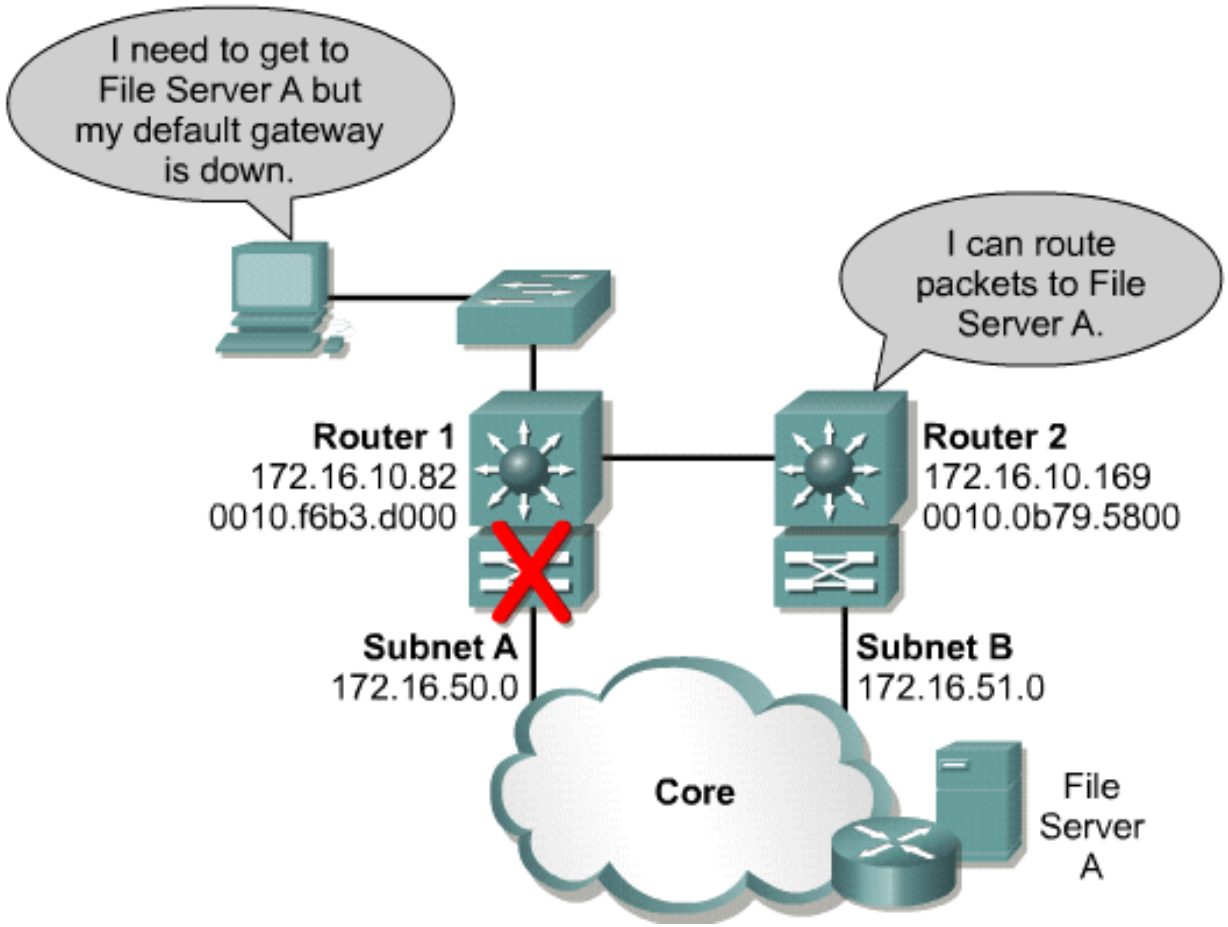
## Redundancy L2 – Spanning Tree Protocol (STP)




http://ipsit.bu.edu

# Case study – L3

## The problem:






mario.antunes@ipleiria.pt

# Case study – L3

**Main topics:**

- Dynamic routing
- ICMP Redirect (IR)
- NAT with TCP Load Distribution
- Hot Standby Router Protocol (HSRP)
- Virtual Router Redundancy Protocol (VRRP)
- Gateway Load Balancing Protocol (GLBP)
- Single Router Mode (SRM)
- Server Load Balancing (SLB)

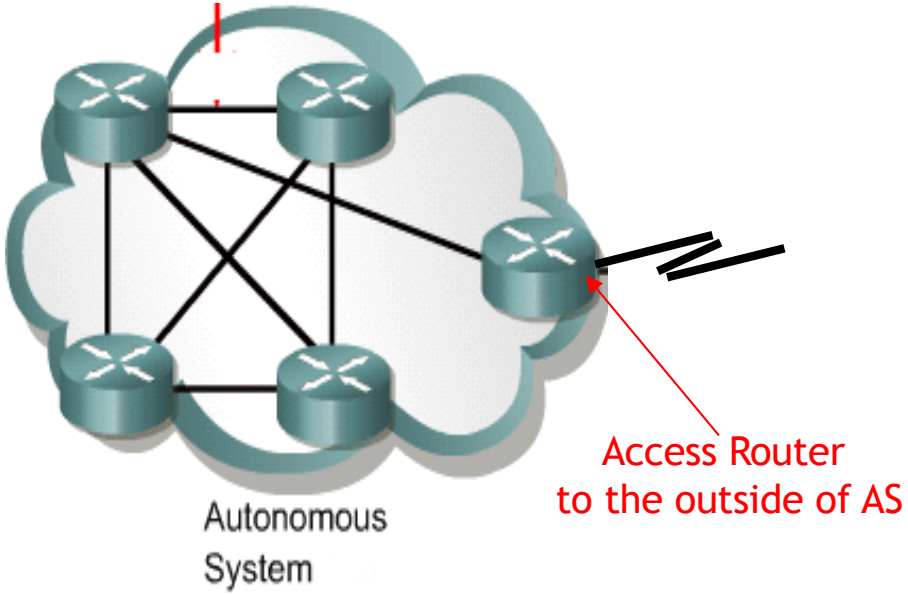


mario.antunes@ipleiria.pt

# Routing fundamentals

## *Autonomous System (AS)*

- Networks that share the same routing politics
- Usually under the same administrative control
- Identified by a unique identifier (AS Id):
  - 32 bits
  - Assigned by RIR






mario.antunes@ipleiria.pt

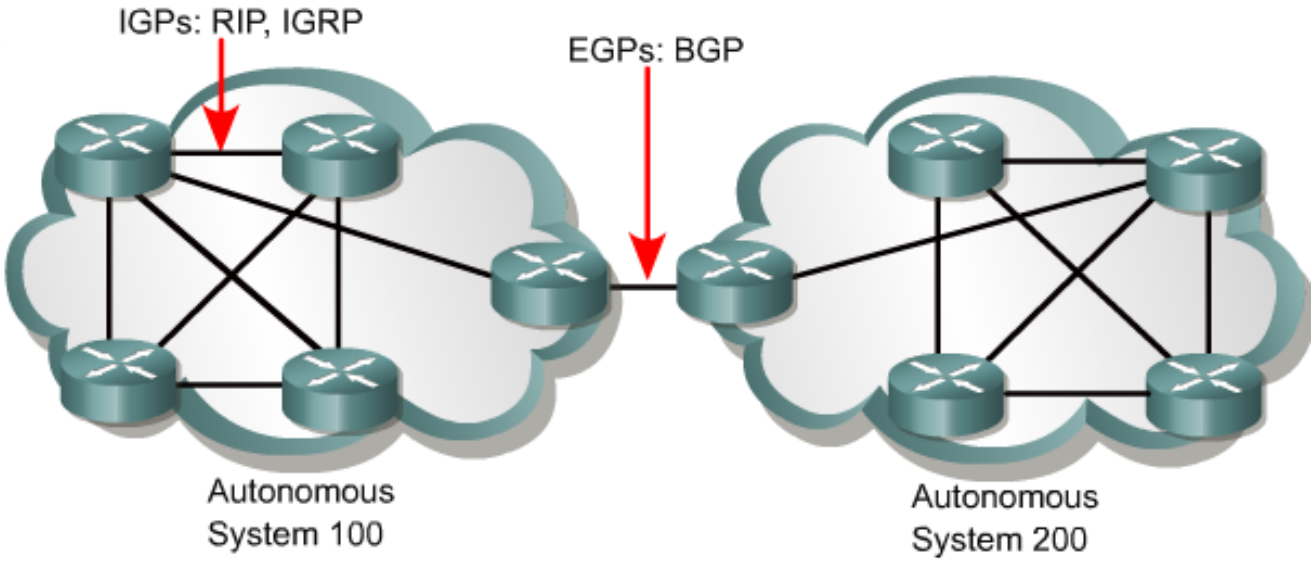
# Routing fundamentals

## IGP

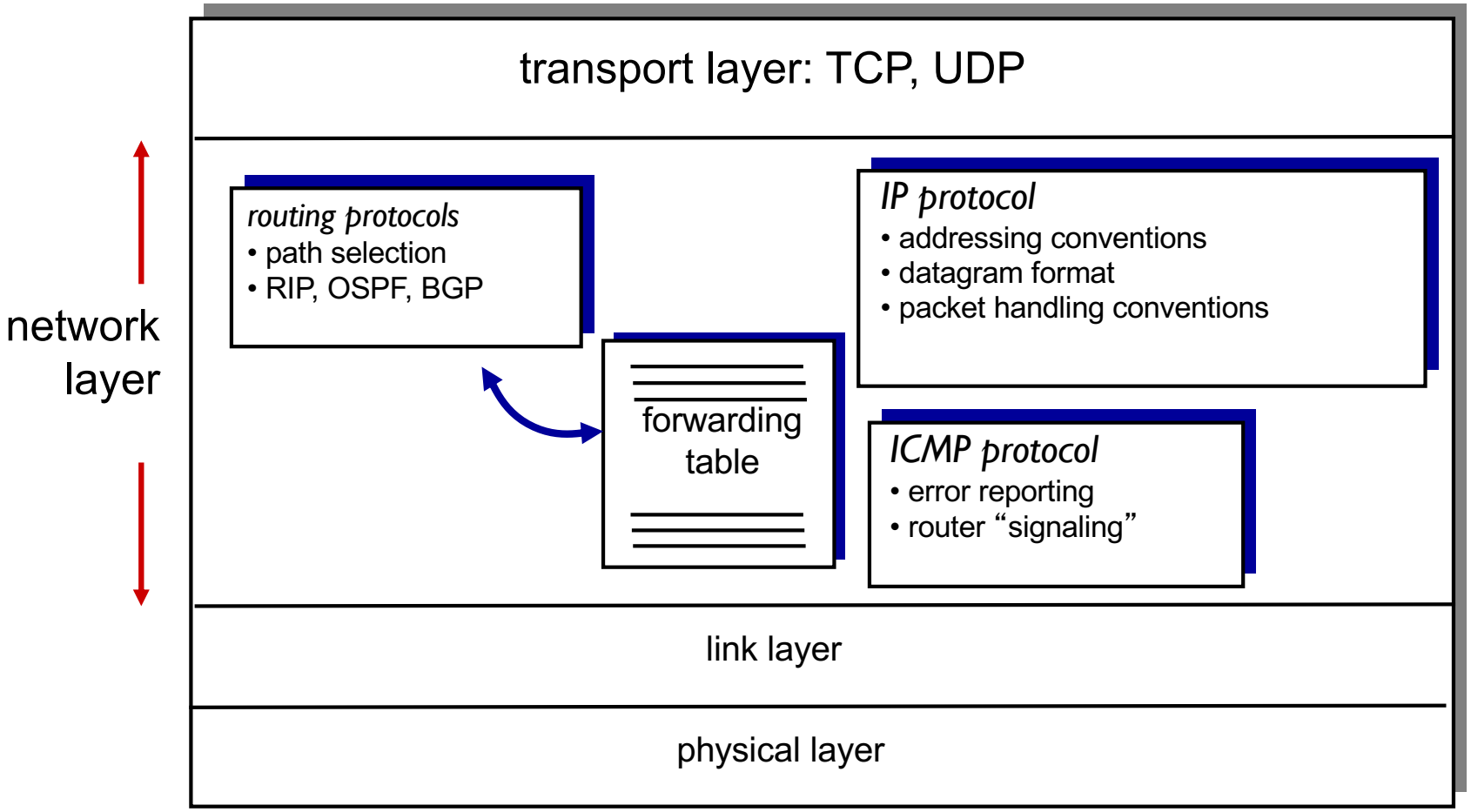
- *Routing* inside AS
- Scalability in the AS
- RIP, IGRP, EIGRP, OSPF

## EGP

- *Routing* between two AS
- Hierarchy in large networks
- Management policies
- BGP



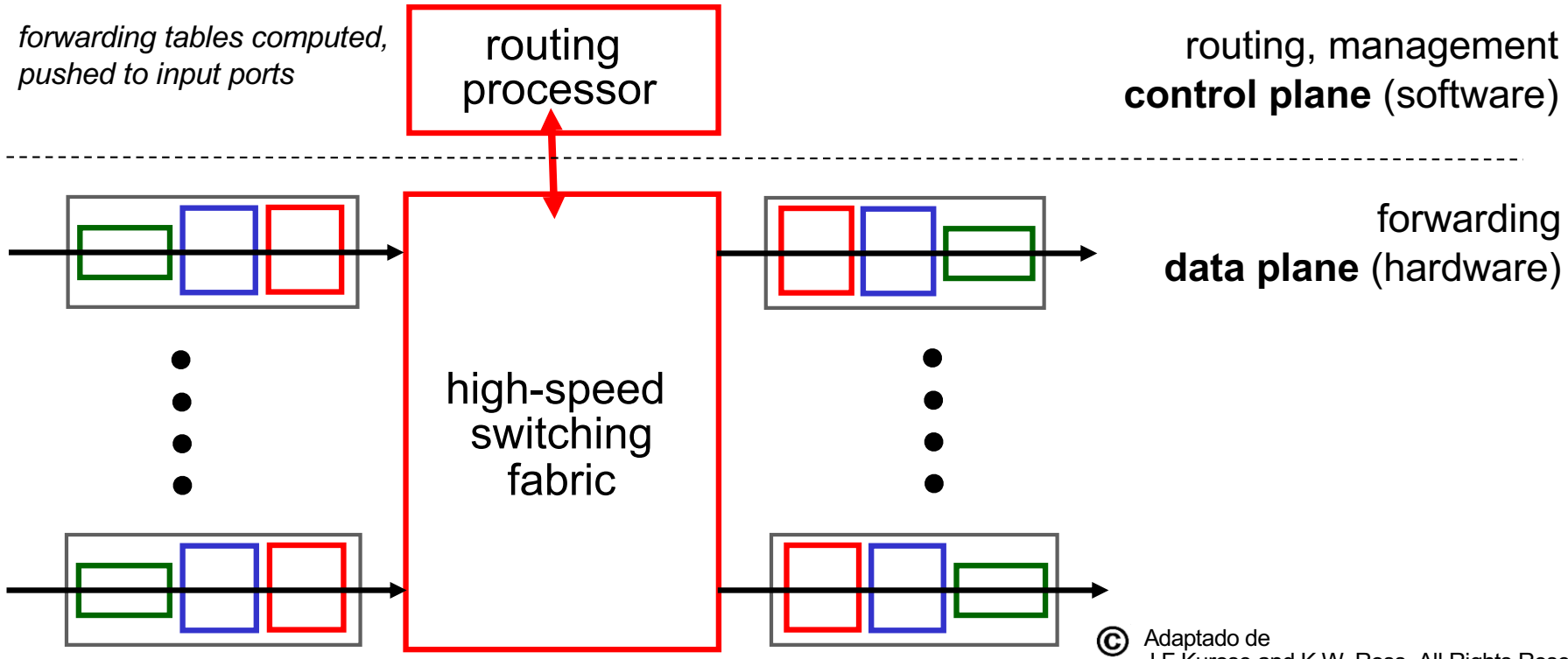
# Routing fundamentals



# Routing fundamentals

## Main functions:

- Execute routing algorithms (RIP, OSPF, BGP)
- Route *(forwarding)* packets to the destination IP by an interface



# Routing fundamentals

- **Non-adaptive (static)**
- **Adaptive (dynamic)**
  - Link State
  - Distance Vector



mario.antunes@ipleiria.pt

# Routing – RIP protocol

## RIP

- Through routes with the **same cost** or alternative routes learned by RIP
- Selective adjust of existing timmers: *update*, *invalid*, *holddown*, *flush*
- Load balance by routes with the same cost
- Maximum number of connections:

```
R1(config)#router rip
R1(config-router)#maximum-path 2
```

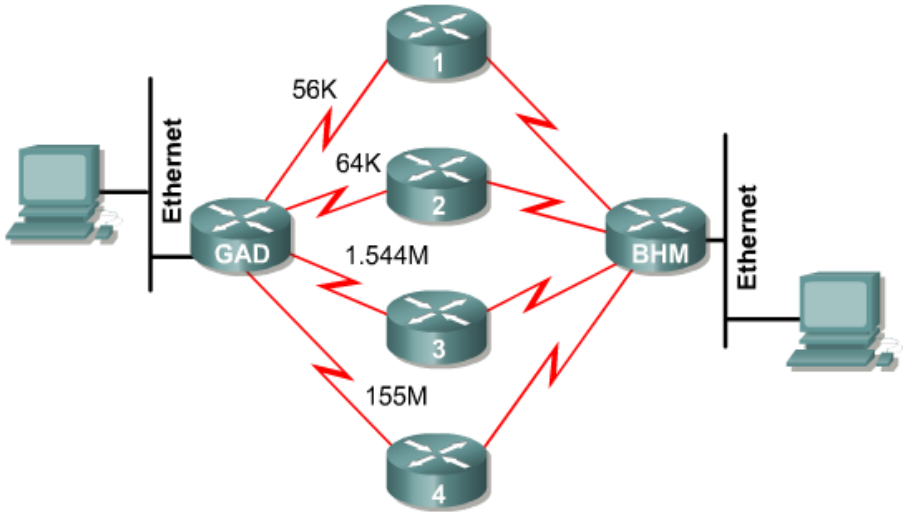


mario.antunes@ipleiria.pt

# Routing – RIP protocol

## *Load balancing*

- Load balancing between connections with the better **equal** cost
- RIP uses up to 32 connections (default=4)
  (`router(config-router)#`**`maximum-paths N`**)
- Method used is *"round-robin"*




```
RouterC#show ip route 192.168.2.0
Routing entry for 192.168.2.0/24
 Known via "rip", distance 120, metric 1
 Redistributing via rip
 Last update from 192.168.4.2 on FastEthernet0/0,
00:00:18 ago
 Routing Descriptor Blocks:
 192.168.4.1, from 192.168.4.1, 00:02:45 ago, via
FastEthernet0/0
 Route metric is 1, traffic share count is 1
 * 192.168.4.2, from 192.168.4.2, 00:00:18 ago,
via FastEthernet0/0
 Route metric is 1, traffic share count is 1
```

# Routing – EIGRP protocol

- Enhanced Interior Gateway Routing Protocol
- Distance vector
- Cisco proprietary
- Periodic updates: 90 seconds
- Main features:
  - Operates well in complex networks
  - Flexibility in networks with links that have distinct characteristics
  - Scalability in large scale networks



mario.antunes@ipleiria.pt

# Routing – EIGRP protocol

## Metric

- Default: BW e DLY
- Composed metric:
  - BW
  - DLY
  - LOAD
  - Fiability
  - MTU

```
Router>show ip protocols
Routing Protocol is igrp 300
  Sending updates every 90 seconds, next due in 55
seconds
  Invalid after 270 seconds, hold down 280, flushed
after 360
  Outgoing update filter list for all interfaces is
not set
  Incoming update filter list for all interfaces is
not set
  Default networks flagged in outgoing updates
  Default networks accepted from incoming updates
  IGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  IGRP maximum hopcount 100
  IGRP maximum metric variance 1
  Redistributing igrp 300
```

Metric = [ K1 * BW +
(K2 * BW)/(256-LOAD) +
K3 * DLY ]

Se K5 ≠ 0 → Metric *= [ K5/(RELIABILITY+K4) ]

Defaullt = BW + DLY

**"K" parameters (default)**

- K1 = BW= 1
- K2 = fiability = 0
- K3 = DLY = 1
- K4 = LOAD= 0
- K5 = MTU = 0



mario.antunes@ipleiria.pt

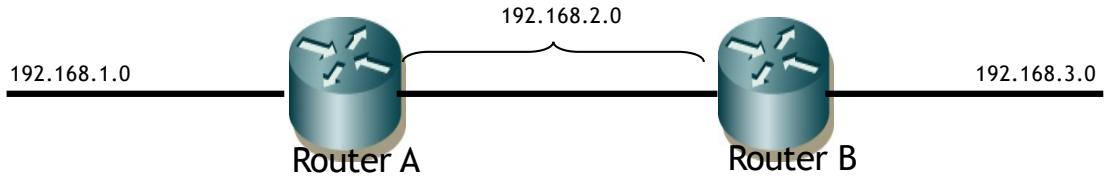
# Routing – EIGRP protocol

## Configuration:

EIGRP on AS 101

Interfaces

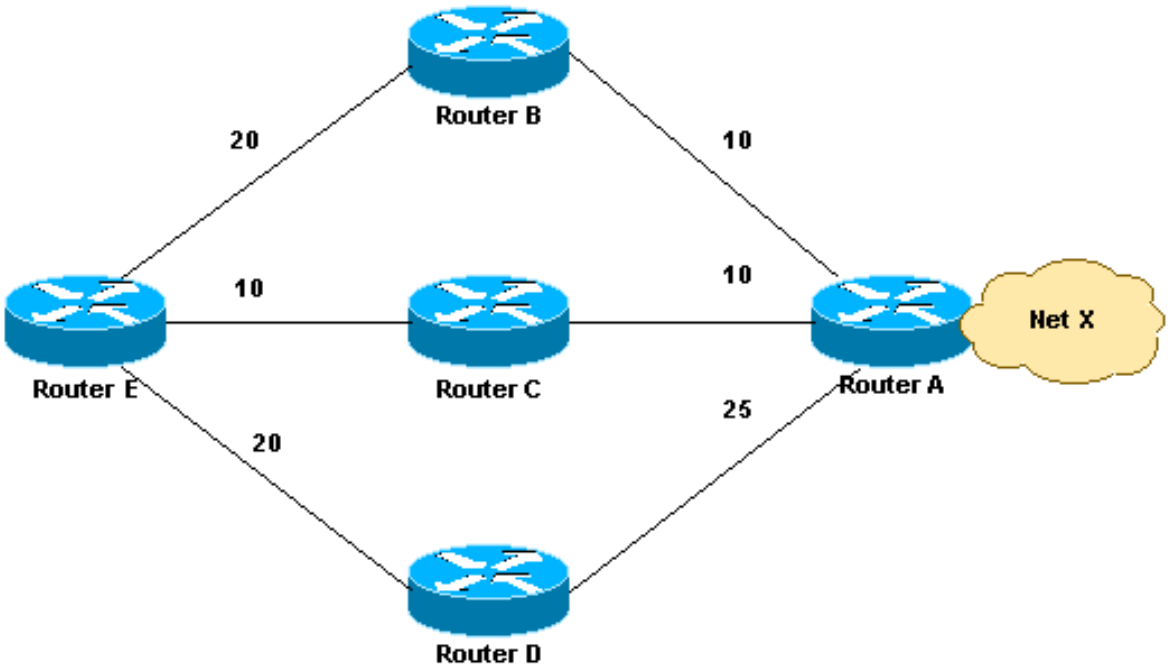
```
RouterA(config)#router igrp 101
RouterA(config-router)#network 192.168.1.0
RouterA(config-router)#network 192.168.2.0

RouterB(config)#router igrp 101
RouterB(config-router)#network 192.168.2.0
RouterB(config-router)#network 192.168.3.0
```



# Routing – EIGRP protocol

## Load balance:

- Routes with equal costs (`maximum-paths`)
- Routes with different costs (`variance`)




E-B-A → 30
E-C-A → 20
E-D-A → 45

router eigrp 1
network *x.x.x.x*
variance 2

E-B-A →30
E-C-A →20

# Routing – other protocols

## OSPF

- Hierarchy with several **areas**
- Designated router (DR) and Backup Designated Router (BDR)
- Faster convergence after failure. Routers learn the whole network topology.

## BGP

- Hierarchical organization
- Use “route reflectors” and confederations.



mario.antunes@ipleiria.pt

# Routing – other protocols

- By default, router caches the routes previously used
- Load balancing → deactivate cache!

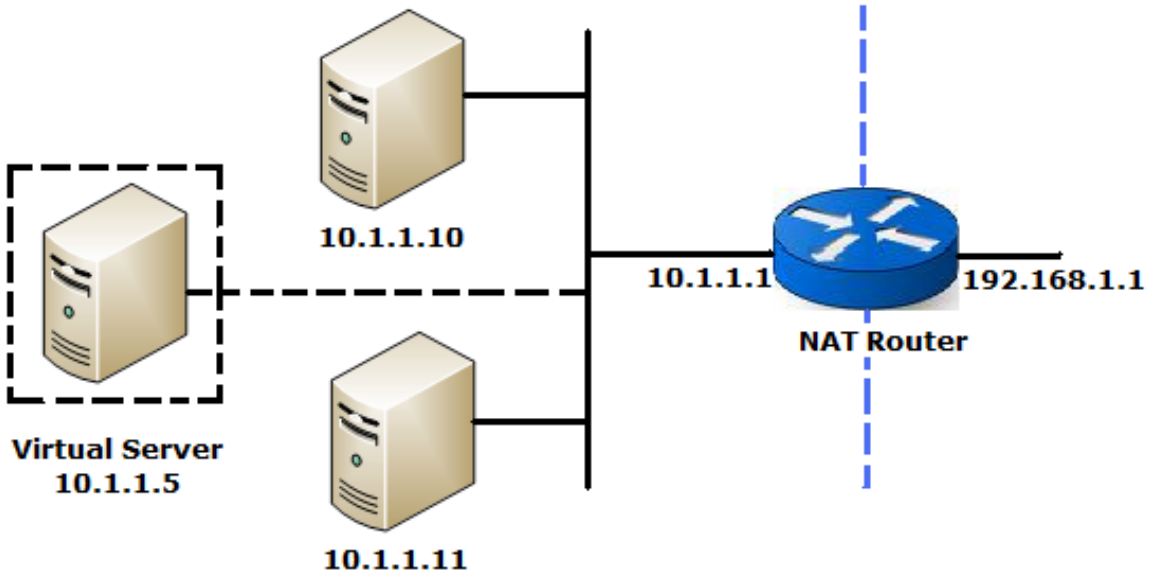
```
[no] ip route-cache [same-interface | flow | distributed | cef | policy]
[no] ip cef
```

- fast switching
- Load balancing by destination



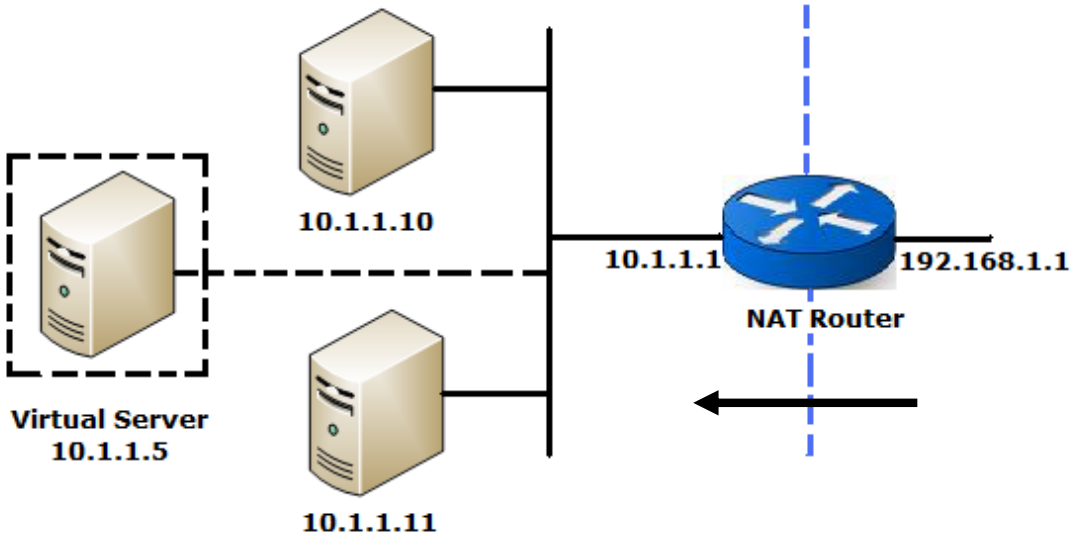
mario.antunes@ipleiria.pt

# NAT with TCP Load Balancing




1. External requests are routed alternately between servers (p.e. `web`, `telnet`)
2. Assure balanced and rotative balancing between servers.

# NAT with TCP Load Balancing




```
ip nat pool teste 10.1.1.10 10.1.1.11 prefix-length 24 type
            rotary
ip nat inside destination list ALL_TCP pool teste
ip alias 10.1.1.5 23
ip alias 10.1.1.5 80
!
```

# NAT with TCP Load Balancing

```
ip access-list extended ALL_TCP
 permit tcp any host 10.1.1.5 eq telnet
 permit tcp any host 10.1.1.5 eq www

ip nat inside
ip nat ouside
```

**Alternative in Cisco IOS:**
To explore Server Load Balancing (SLB) features.



mario.antunes@ipleiria.pt

# First Hop Redundancy Protocols

- Hot Standby Redundancy Protocol (HSRP)
- Virtual Router Redundancy Protocolo (VRRP)
- Global Load Balancing Protocolo (GLBP)



mario.antunes@ipleiria.pt

# Hot Standby Router Protocol (HSRP)


- **Packets are redirected automatically to a *standby* router**
- **Transparently to the end-user**
- **Functioning:**
  - A routers set  share a MAC and IP addresses (Virtual Router)
  - One *router* is elected as active
  - Routers exchange control messages of HSRP state
  - ARP assignes MAC to the  MAC Virtual
  - If active *router* fails, standby router functions as active router.



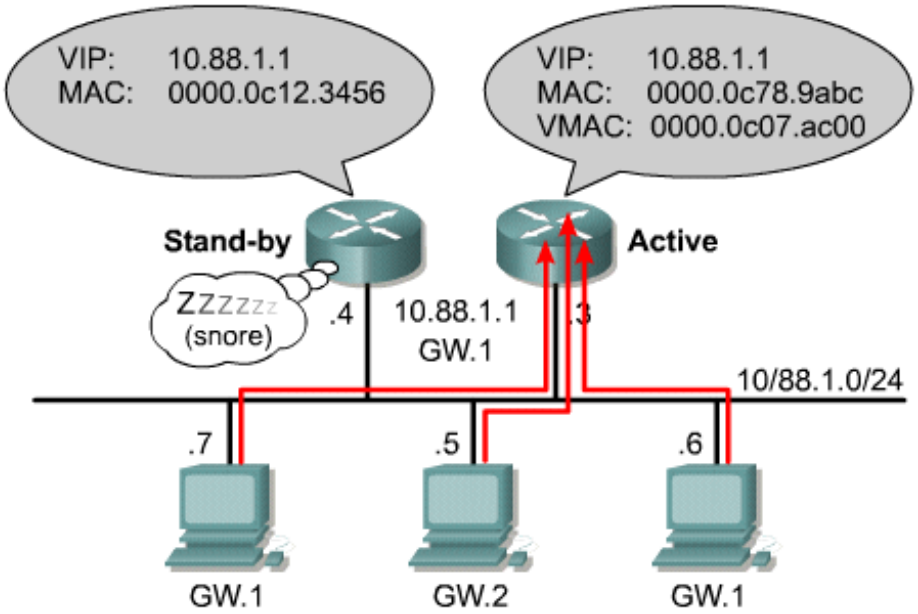
mario.antunes@ipleiria.pt

# Hot Standby Router Protocol (HSRP)

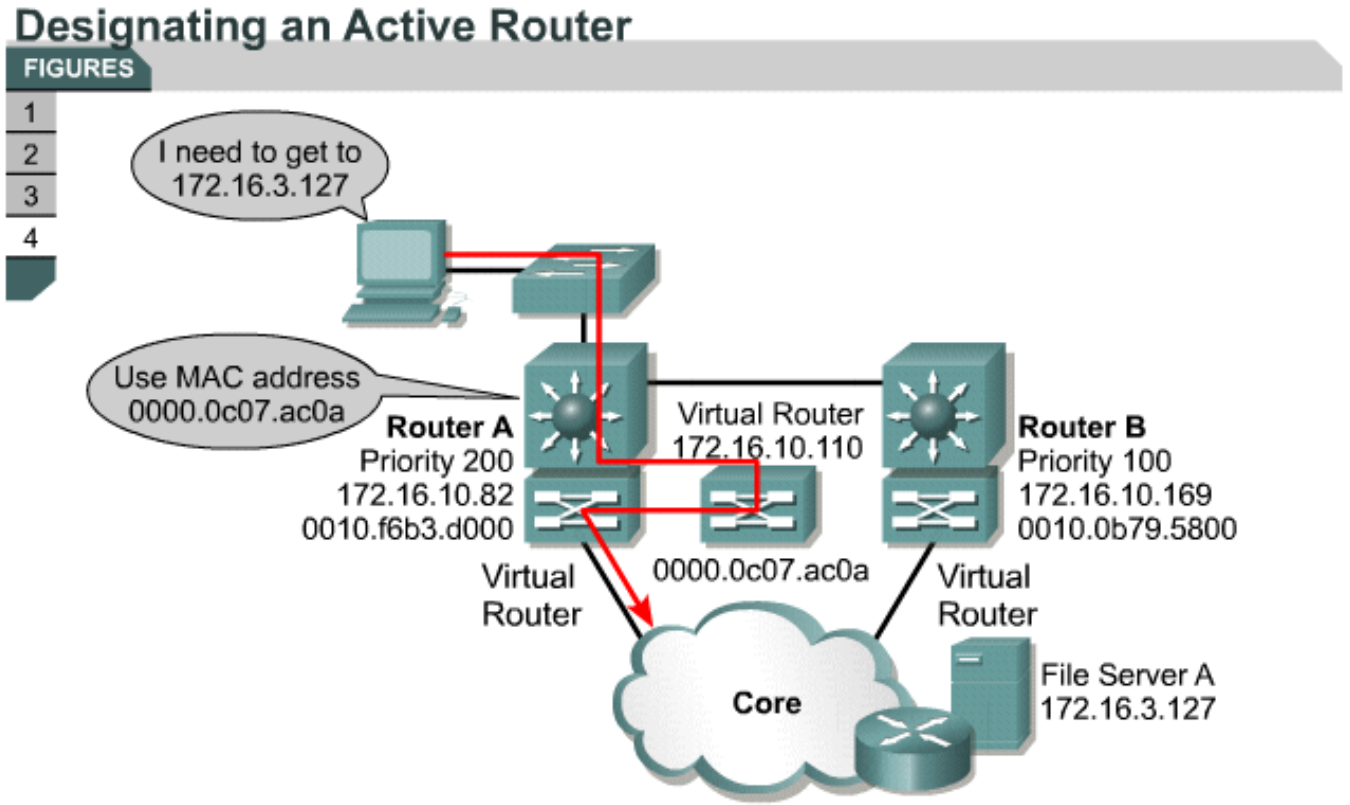
## HSRP Operation

FIGURE

1
2
3
4



# Hot Standby Router Protocol (HSRP)



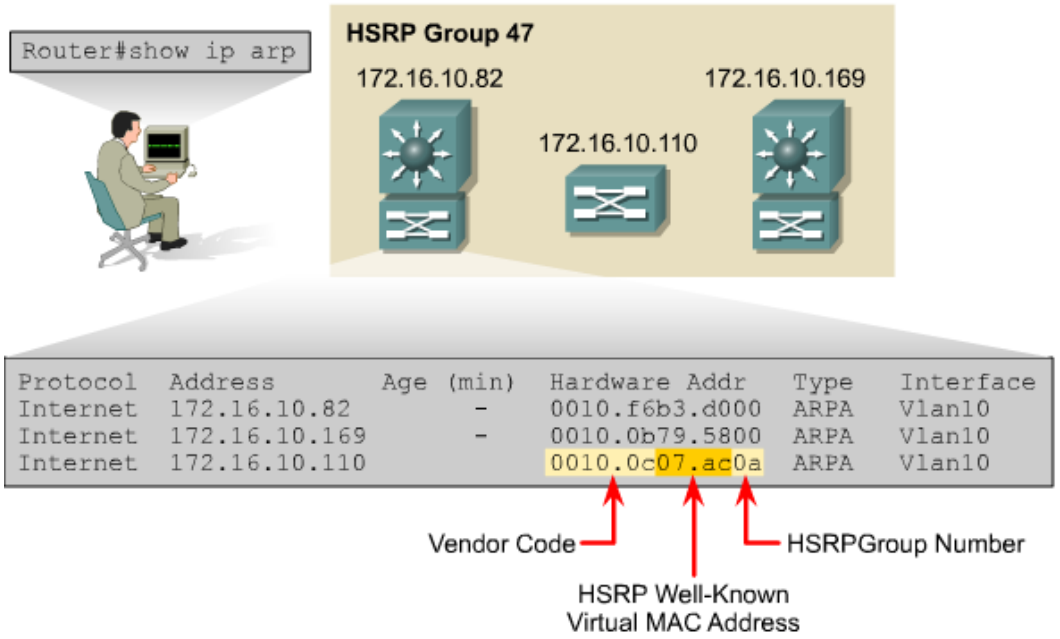


© Mário Antunes
mario.antunes@ipleiria.pt

# Hot Standby Router Protocol (HSRP)

## Virtual MAC address used at HSRP:

- Vendor ID – MAC address (24 bits)
- HSRP Code – 16 bits (“07.ac”)
- Group ID – last 8 bits of MAC endereço

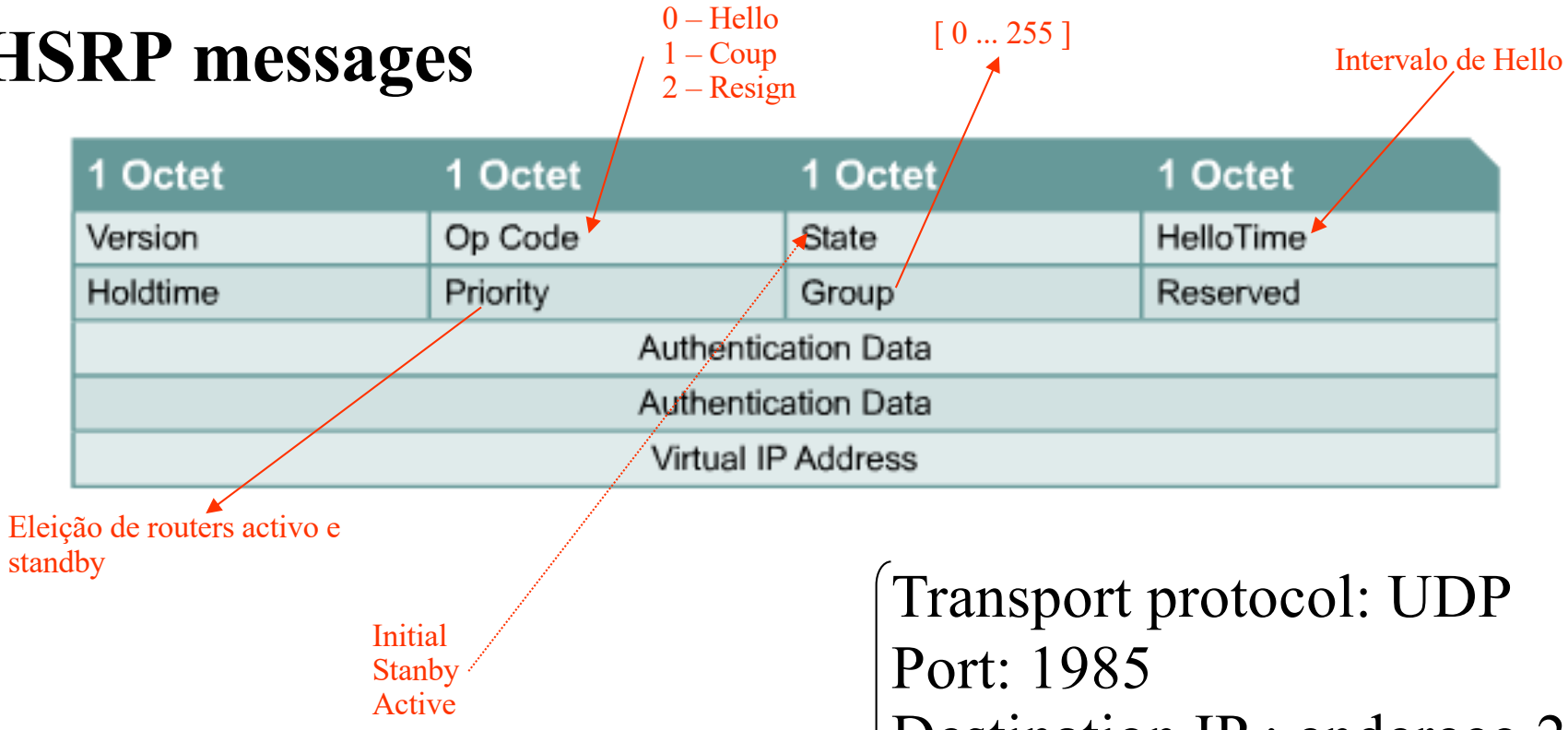





mario.antunes@ipleiria.pt

# Hot Standby Router Protocol (HSRP)

## HSRP messages

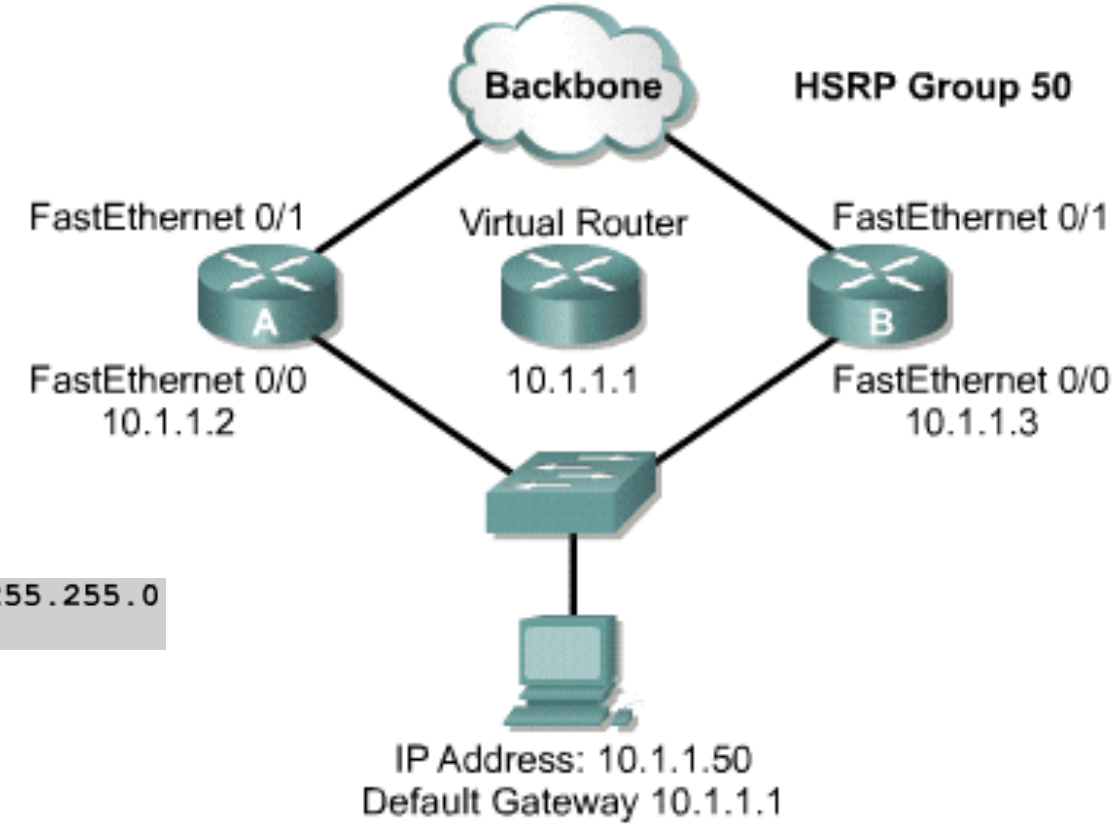



- Transport protocol: UDP
- Port: 1985
- Destination IP : endereço 224.0.0.2
- TTL = 1

Possible states:
- Initial, Learn, Listen, Speak, Stanby, Active

# Hot Standby Router Protocol (HSRP)

## Configuration




## Router A

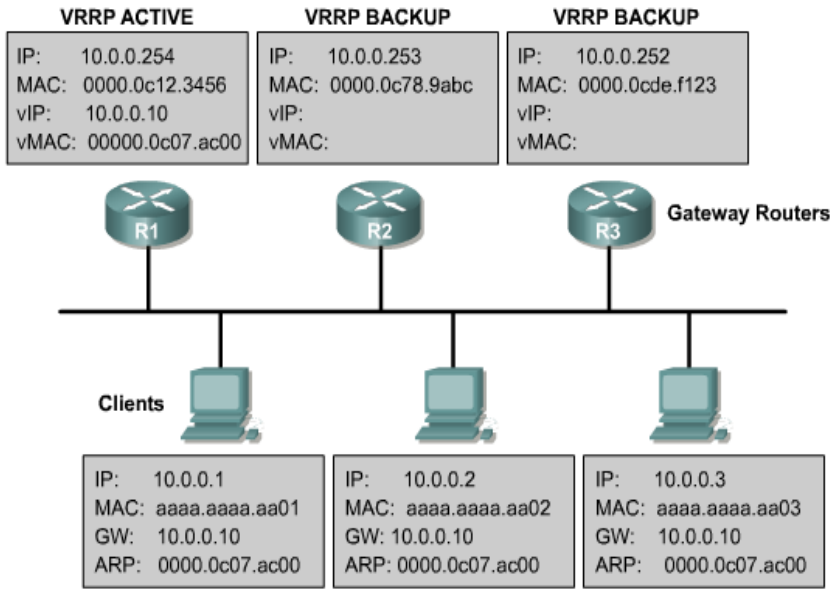
```
A(config-if)#ip address 10.1.1.2 255.255.255.0
A(config-if)#standby 50 ip 10.1.1.1

A(config-if)#standby 50 priority 150

A(config-if)#standby 50 preempt

A(config-if)#standby 50 timers 5 15

A(config-if)#standby 50 track fastethernet 0/1 55
```

# Virtual Router Redundancy Protocol (VRRP)

RFC 2338

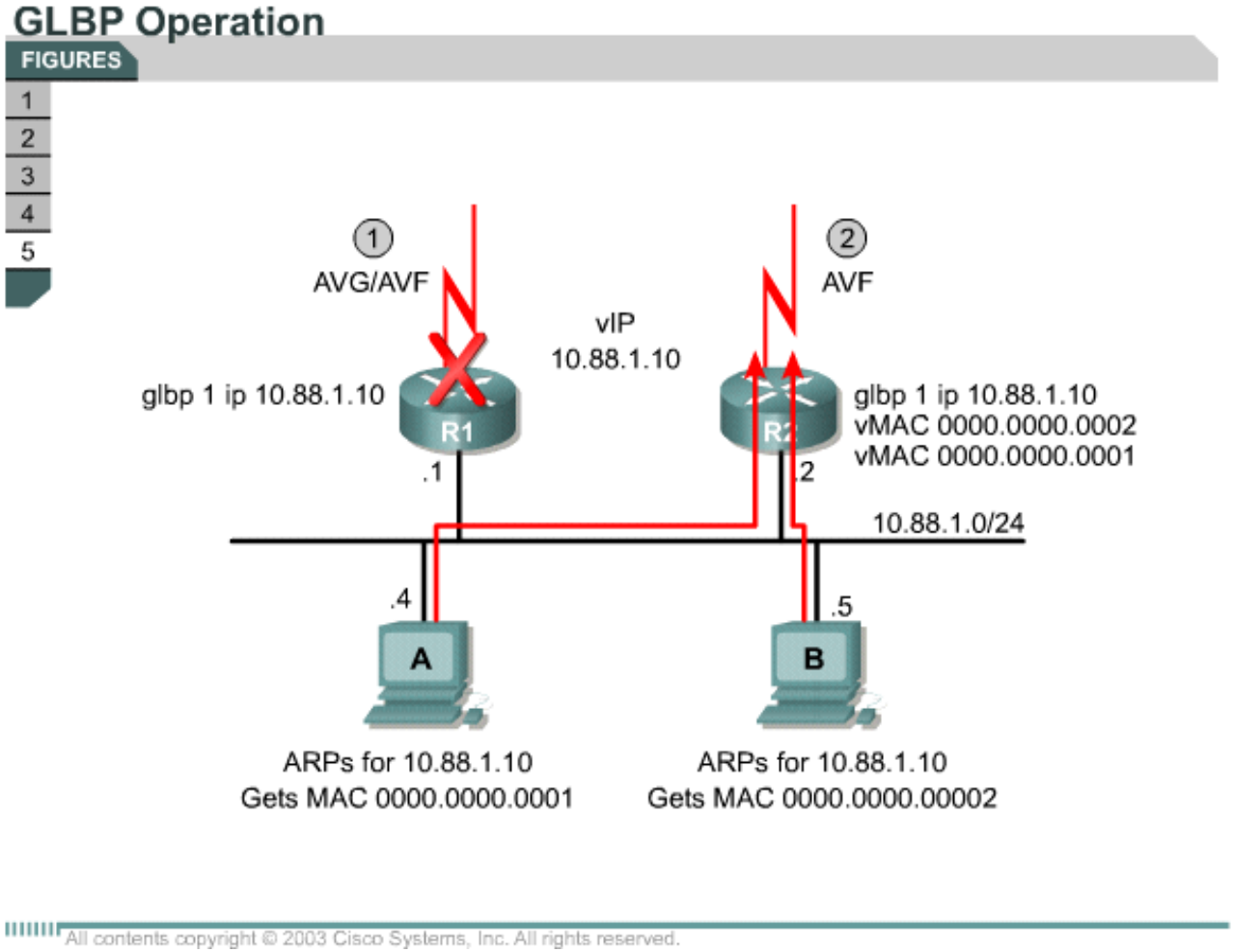
RFC 3768

- Similar to HSRP
- Interoperability between equipments of distinct vendors
- A “Master” *router* and several “Backup” *routers*
- Periodic updates sent by Master *router*
- VRRP should be used when ot all routers are Cisco.



# Gateway Load Balance Protocol (GLBP)

- **Functions: Redundancy + Load Balance**



# Bibliografia


- RFCs 2281, 2378 e 3768

- Mário Antunes; *"Gestão de Projetos TI e administração centralizada de sistemas e redes – cenários práticos em contexto empresarial"*; IPLeiria; 2010




mario.antunes@ipleiria.pt