

# GARR WS9

OpenSource per l'erogazione di servizi in alta disponibilità

Roma 17 giugno 2009

Mario Di Ture  
Università degli Studi di Cassino  
Centro di Ateneo per i Servizi Informatici

# Programma

- **Cluster Linux per l'alta disponibilita'**
- **DRBD**
- **Linux-HA Heartbeat**

# Tipi di cluster

Un cluster è un gruppo di computer collegati che, lavorando in stretta relazione sembrano, sotto molti aspetti, un unico computer. Esistono diversi tipi di cluster:

- **High performance**
- **Load balancing**
- **High availability**



# Alta disponibilità e Clustering

- Tra i computer (**detti anche nodi**) del **Cluster** intercorre una relazione di trust
- Quando un computer si rompe, altri si fanno carico (**take over**) del suo lavoro
- L'operazione di take over normalmente riguarderà **l'indirizzo IP, i servizi**, ecc.
- Il clustering esaminato non riguarda le elevate performance ma la **continuità del servizio**
- E' impossibile raggiungere una disponibilità del **100%**

# Misurare il fermo macchina (downtime\*)

% Disponibilit�	Downtime per anno	Downtime per mese	Downtime per settimana
90%	36.5 giorni	72 ore	16.8 ore
95%	18.25 giorni	36 ore	8.4 ore
98%	7.30 giorni	14.4 ore	3.36 ore
99%	3.65 giorni	7.20 ore	1.68 ore
99.5%	1.83 giorni	3.60 ore	50.4 min
99.8%	17.52 ore	86.23 min	20.16 min
99.9% ("tre nove")	8.76 ore	43.2 min	10.1 min
99.95%	4.38 ore	21.56 min	5.04 min
99.99% ("quattro nove")	52.6 min	4.32 min	1.01 min
99.999% ("cinque nove")	5.26 min	25.9 sec	6.05 sec
99.9999% ("sei nove")	31.5 sec	2.59 sec	0.605 sec

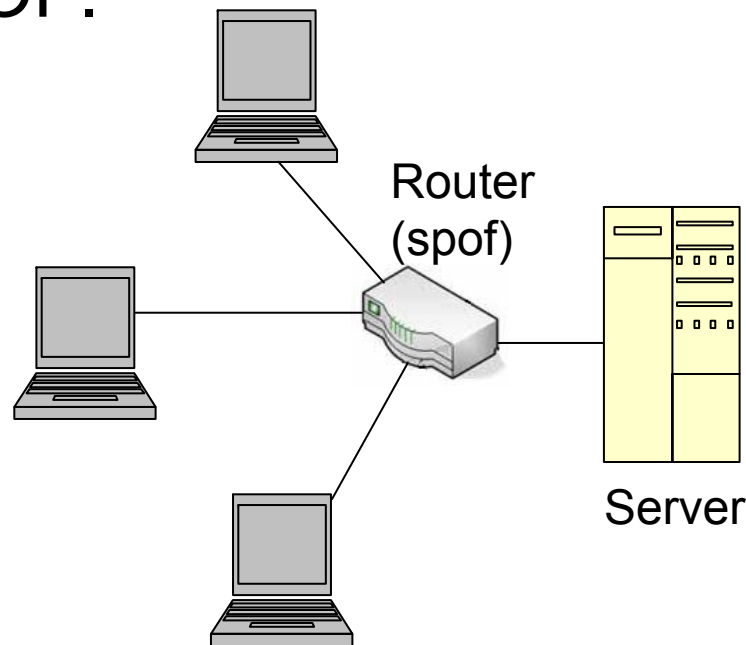
\* I downtime possono essere **pianificati** (ad es. per l'applicazione di patch) o **non pianificati** (ad es. per la rottura di hardware)



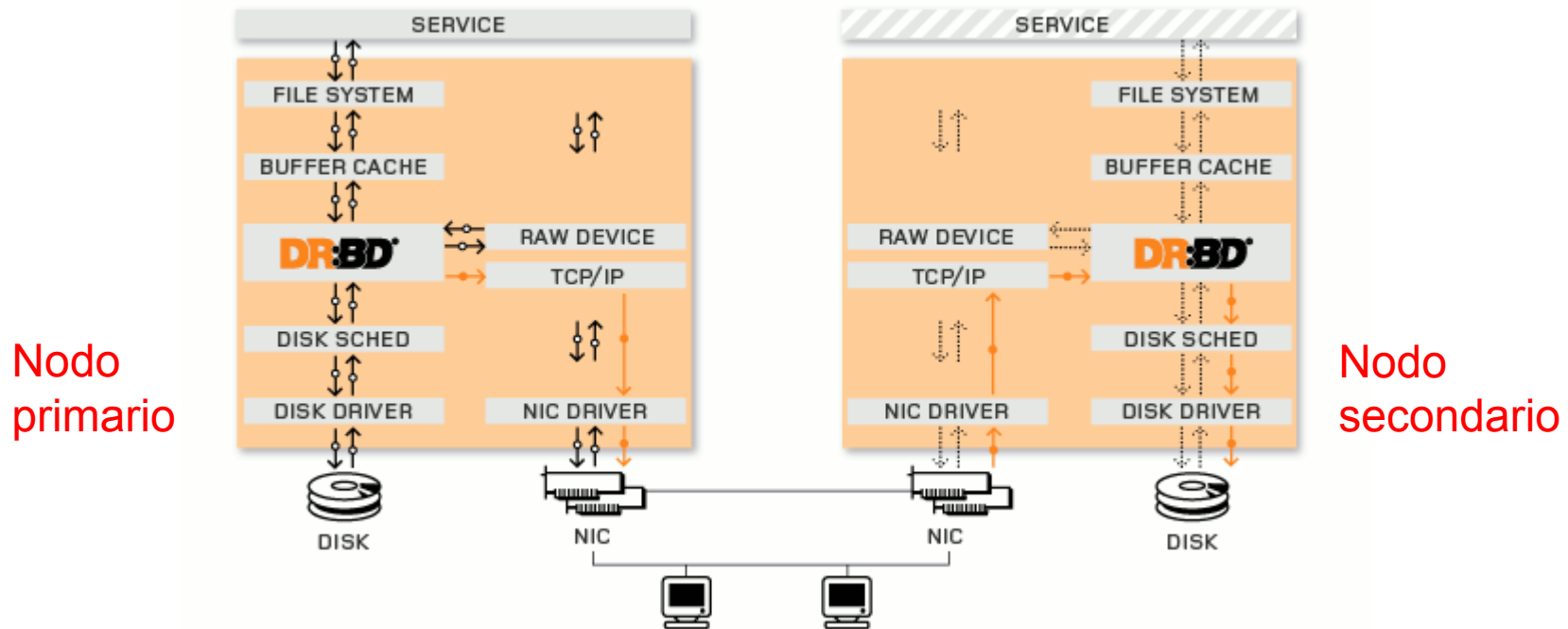
# Single Point of Failure (SPOF)

E' importante eliminare o ridurre al minimo i single point of failure.

Per fare ciò si deve tendere a **ridondare** tutti gli attuali SPOF.



# DRBD



DRBD sta per Distributed Replicated Block Device ed è, tipicamente, un componente di un cluster per la high availability (**modulo del kernel linux**). Effettua il mirror di un intero device a blocchi attraverso una connessione di rete. DRBD può essere visto come un raid-1 via rete.

# DRBD – Configurazione e avvio

```
# Configurazione - file /etc/drbd.conf
# identico sui due host
global {
    usage-count yes;
}
common {
    protocol C;
    syncer { rate 5M; }
}
resource r0 {
    on nodo-a {
        device /dev/drbd1;
        disk /dev/hda5;
        address 192.168.0.1:7789;
        meta-disk internal;
    }
    on nodo-b {
        device /dev/drbd1;
        disk /dev/hda5;
        address 192.168.0.2:7789;
        meta-disk internal;
    }
}
```

```
# Su entrambi i nodi:
# Avvio drbd:
/etc/init.d/drbd start
# Elimino il fs preesistente (se necessario):
dd if=/dev/zero bs=1M count=1 of=/dev/hda5; sync
# Creo e collego la risorsa /dev/drbd1 (disk state: Diskless):
drbdadm create-md r0
# Collego il device e set della sincr. (disk state: Inconsistent):
drbdadm attach r0
drbdadm syncer r0
# Apro sul firewall dei nodi la porta tcp 7789.
# Peer connect (ds stato: Inconsistent/Inconsistent)
drbdadm connect r0
# Disattivo i servizi al boot: (esempio di un web server)
chkconfig httpd off
chkconfig mysqld off
# Attivo il servizio al boot:
chkconfig drbd on
```

```
# Solo sul nodo PRIMARIO:
drbdadm -- --overwrite-data-of-peer primary r0
# Ora DRBD è operativo
# Creo il File System
mkfs.ext3 /dev/drbd1
mkdir /ha
mount /dev/drbd1 /ha
```





# DRBD - meta-dati

DRBD memorizza alcune informazioni relative ai dati-utente replicati in un'area dedicata. Tali informazioni prendono il nome di meta-dati e possono essere allocate sullo stesso device o su un device esterno.

DRBD Metadata Size

Block device size	DRBD metadata
1 GB	2 MB
100 GB	5 MB
1 TB	33 MB
4 TB	128 MB

Calcolo approssimato dello spazio occupato dai meta-dati:

( $C_{MB}$  = dimensione del device)

$$M_{MB} < \frac{C_{MB}}{32768} + 1$$

# DRBD – Utilizzo \*

I 10 nodi con i device più grandi:\*\*

Device size [GB]	Data creazione
21.419,6953	2008-09-15
20.362,9516	2008-11-21
20.352,9548	2008-11-20
20.275,2031	2008-12-04
20.275,2031	2008-11-30
17.602,4477	2008-08-14
17.602,4477	2008-08-14
16.384,0000	2009-04-07
16.384,0000	2009-03-13
16.000,0000	2007-11-20

Nodi e risorse:

Numero di nodi installati	47.968
Numero di risorse configurate	75.362
Numero massimo di risorse su un nodo	185

\* Fonte: [http://usage.drbd.org/cgi-bin/show\\_usage.pl](http://usage.drbd.org/cgi-bin/show_usage.pl) - Dati al 7/5/2009.

\*\* Per le versioni open source antecedenti la 8.3 vengono supportati complessivamente device fino a 4 TB per nodo. La versione 8.3 supporta device fino a 16TB.

# Spostare i dati sul device DRBD

## *Esempio: Web server Apache*

```
mkdir /ha/http
cp /etc/httpd/conf/httpd.conf /ha/http/
/etc/init.d/httpd stop
ln -s -f /ha/http/httpd.conf /etc/httpd/conf/
cp -R /var/www/html /ha/http/htdocs
vi /etc/httpd/conf/httpd.conf
    DocumentRoot "/ha/http/htdocs"
chown -R apache.apache /ha/http
/etc/init.d/httpd start
```

Comandi da eseguire sul nodo primario

Comandi da eseguire su entrambi i nodi

# Spostare i dati sul device DRBD

## *Esempio: Database server Mysql*

```
/etc/init.d/mysqld stop
```

```
mkdir /ha/mysql
```

```
cp /etc/my.cnf /ha/mysql
```

```
ln -s -f /ha/mysql/my.cnf  
/etc/my.cnf
```

```
cp -R /var/lib/mysql
```

```
/ha/mysql/data
```

```
vi /etc/my.cnf
```

```
    datadir = /ha/mysql/data
```

```
chown -R mysql:mysql
```

```
/ha/mysql/data
```

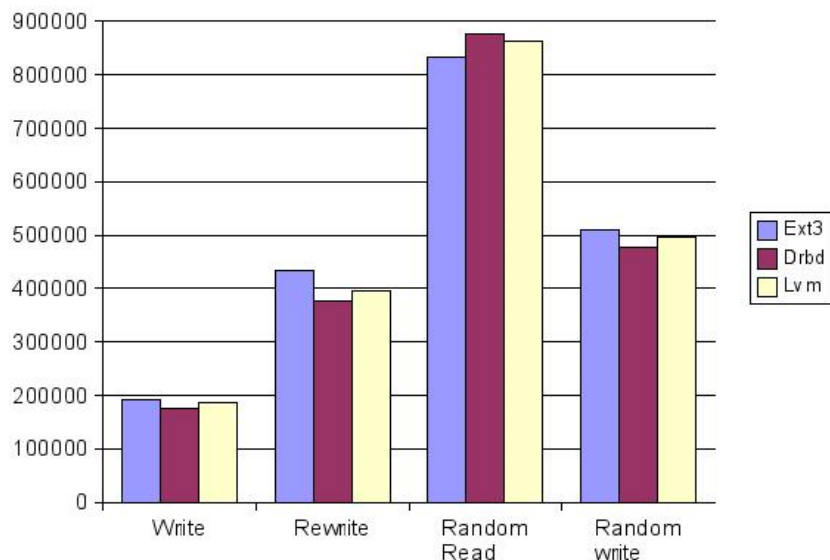
```
/etc/init.d/mysqld start
```

Comandi da eseguire sul nodo primario

Comandi da eseguire su entrambi i nodi

# Performance dei device - iozone

## Prestazioni in cache

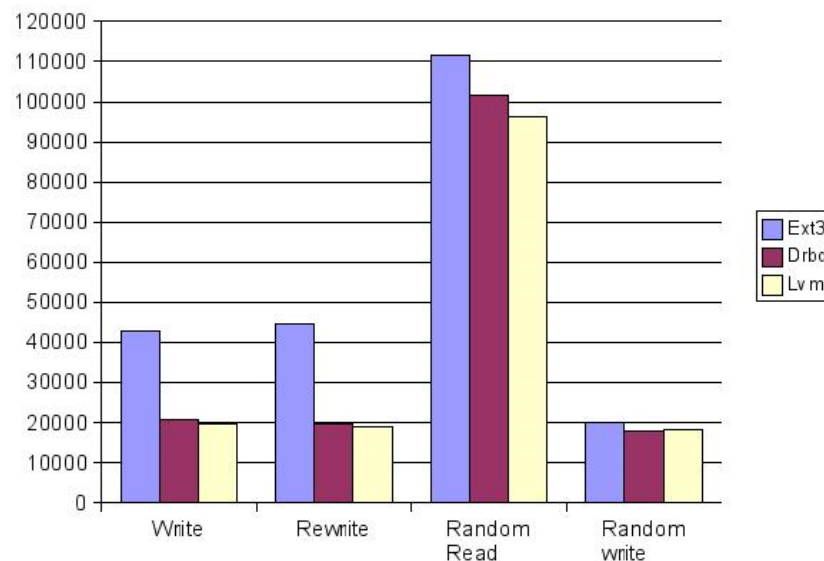


Ext3

Ext3+DRBD

Ext3+DRBD+LVM

## Prestazioni fuori cache

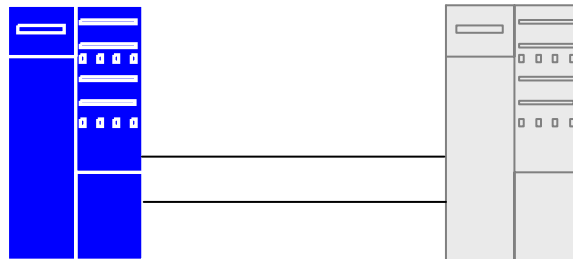


Fonte: <http://www.ba.infn.it/calcolo/documenti/Cluster.html>

# Linux HA High Availability

Il ruolo normale di Linux HA è quello di:

- monitorare i nodi costantemente
- monitorare, tipicamente, utilizzando connessioni ridondanti
- attivare, ad esempio, un nodo inattivo quando un nodo che dovrebbe essere attivo non è più raggiungibile



# Linux HA – Configurazione R1 (Obsoleta)

Dopo aver installato il sw creare sui due nodi:

1. l'utente hacluster e relativo gruppo haclient
2. il file /etc/ha.d/ha.cf
3. il file /etc/ha.d/authkeys (chmod 600)
4. il file /etc/ha.d/haresources

```
# /etc/ha.d/ha.cf
udpport 694
bcast eth0
node nodo-a nodo-b
ping 10.10.0.1 10.10.0.2
keepalive 4
deadtime 10
initdead 15
```

```
# /etc/ha.d/authkeys
auth 2
1 crc
2 sha1 MySecret
3 md5 MySecret
```

```
# /etc/ha.d/haresources
nodo-a 192.168.0.3 drbddisk::r0 Filesystem::/dev/drbd1::/ha httpd mysqld
```

5. Avviare il servizio: /etc/init.d/heartbeat start

# Linux HA – Configurazione R2

1. Attivare crm in ha.cf (aggiungere direttiva **crm yes**).
2. Attivare l'interfaccia grafica e avviare il programma **hb\_gui** (gui)
3. In hb\_gui creare un **gruppo di risorse**, aggiungere al gruppo **IPAddr, Drbddisk, FileSystem, Mysqld e Httpd**. Avviare il gruppo di risorse
4. Creare un constraint **location** per il nodo destinato ad ospitare il gruppo di risorse
5. Associare alla location una espressione in cui **uname=nodo-a**
6. Tutte le impostazioni sulle risorse e sul constraint vengono salvate nel file **cib.xml**



# Linux-HA Gui – Esempio 1

The screenshot shows the Linux HA Management Client interface. The main window displays a tree view of the configuration on the left and a detailed view of the selected resource on the right.

**Tree View (Left Panel):**

Name	Status
linux-ha	with quorum
Nodes	
10.10.0.2	ping node
10.10.0.1	ping node
nodo-b	running(dc)
nodo-a	running
Resources	
group_web	group
resource_ip	running on [nodo-a]
resource_drbd	running on [nodo-a]
resource_filesystem	running on [nodo-a]
resource_mysql	running on [nodo-a]
resource_http	running on [nodo-a]
Constraints	
Locations	
location_nodo_preferito	
Orders	
Colocations	

**Attributes (Right Panel):**

ID: location\_nodo\_preferito    Score: 100

Resource: group\_web    Boolean OP: [ ]

**Expressions:**

Attribute	Operation	Value
#uname	eq	nodo-a

Buttons: Add Expression, Delete Expression, Apply, Reset

Connected to 127.0.0.1

# Linux-HA Gui - Es. 1- Monitoring

The screenshot displays the Linux HA Management Client interface. The main window shows a tree view of the configuration, including Nodes (10.10.0.2, 10.10.0.1, nodo-b, nodo-a) and Resources (group\_web, resource\_ip, resource\_drbd, resource\_filesystem, resource\_mysql, resource\_http). A dialog box titled "Add Operation" is open, allowing the user to configure a new operation named "monitor".

**Add Operation Dialog Configuration:**

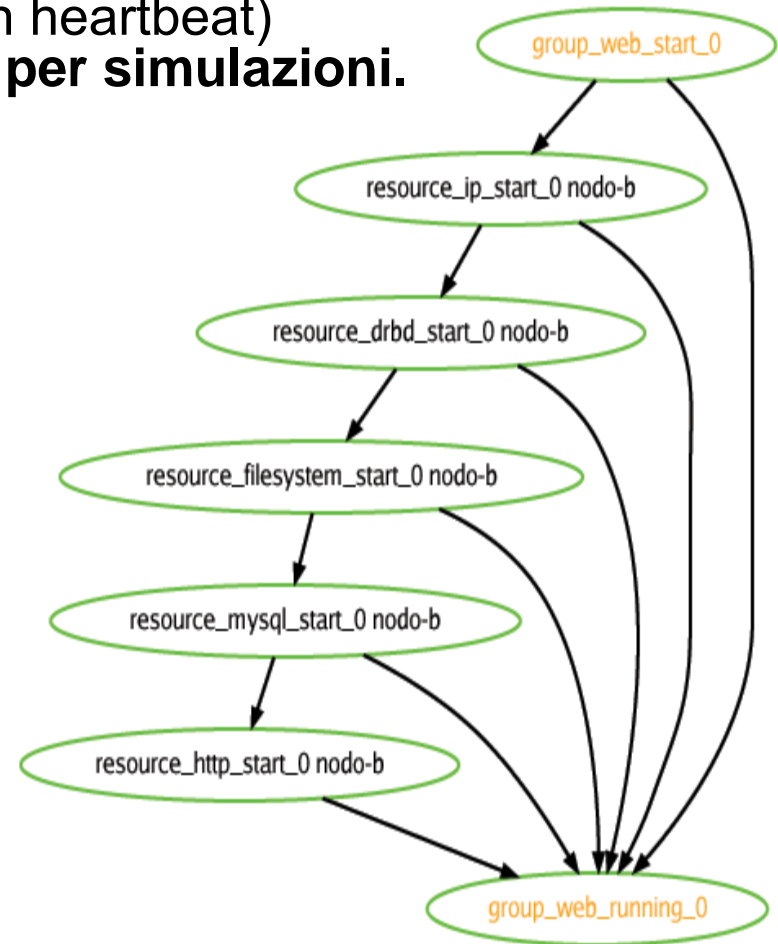
- Name: monitor
- Description: Monitoring-Http
- Interval: 15
- Timeout: 15
- Start Delay: 15
- Disabled: false
- Role: Started
- Prereq: nothing
- On Fail: restart

The main window also shows a table for "Current Running on [nodo-a]" with columns: Name, Description, Interval, Timeout, Start Delay, Disabled, Role, Prereq, C.

# Linux-HA – Schema transizioni

Viene utilizzato il comando **ptest** (incluso in heartbeat) e graphviz per l'output grafico. Utile **anche per simulazioni**.

- Le frecce indicano l'ordine delle dipendenze
- Le azioni con bordo verde e testo nero sono transizioni effettive
- Le azioni con bordo verde e testo arancio sono pseudo azioni col solo scopo di semplificare il grafico
- Le azioni con testo nero vengono inviate al LRM
- I loop dipendono da errori di implementazione del cluster o bug di heartbeat.



# In pratica:

```

cibadmin -Q | grep -v lrm > /tmp/hostname.xml
ptest -D /tmp/hostname.dot -X /tmp/hostname.xml
dot -Tpdf -o /tmp/hostname.pdf /tmp/hostname.dot
  
```

# Linux-HA Gui

## Mail Server

The screenshot shows the Linux HA Management Client window. The main area is divided into two panes. The left pane shows a tree view of the configuration:

- linux-ha (with quorum)
  - Nodes
    - 10.10.0.5 (ping node)
    - nodo-b (running(dc))
    - nodo-a (running)
      - resource\_drbdisk (running on [nodo-a])
      - resource\_network\_ha (running on [nodo-a])
      - resource\_postfix (running on [nodo-a])
      - resource\_mail (running on [nodo-a])
      - resource\_filesystem (running on [nodo-a])
      - resource\_cyrus (running on [nodo-a])
  - Resources
    - gruppo\_posta\_unicas (group)
      - resource\_network\_ha (running on [nodo-a])
      - resource\_drbdisk (running on [nodo-a])
      - resource\_filesystem (running on [nodo-a])
      - resource\_cyrus (running on [nodo-a])
      - resource\_postfix (running on [nodo-a])
      - resource\_mail (running on [nodo-a])
  - Constraints
    - Locations
      - location\_posta\_unicas
    - Orders
    - Colocations

The right pane shows the configuration details for the selected resource (resource\_mail). The configuration parameters are:

- Version: 2.1.3
- Debug Level: 0
- UDP Port: 694
- Keep Alive: 4
- Warning Alive: 15000ms
- Dead Time: 30
- Initial Dead Time: 60

At the bottom of the right pane, there are buttons for 'Apply', 'Reset', and 'Default'. The status bar at the bottom of the window indicates 'Connected to 127.0.0.1'.

# Linux-HA e Fencing

- Ci possono essere situazioni in cui il software di gestione non è in grado di definire correttamente lo stato del cluster
- Il fencing è un metodo adottato per riportare lo stato del cluster alla “normalità”
- Il fencing ha lo scopo principale di evitare danni ai dati
- In pratica, nega l'accesso ad una risorsa (ad esempio ad un hard disk) ad un nodo se quest'ultimo perde il contatto con gli altri nodi
- Il fencing può avvenire a livello di risorsa o a livello di nodo
- **Stonith** (shoot the other node in the head) è il sistema di fencing di Heartbeat ed opera a livello di nodo facendo uso di device come iLO, power switch etc.

# Grazie per l'attenzione

*info: m.diture@unicas.it*

*Riferimenti:*

[www.linux-ha.org](http://www.linux-ha.org)

[www.drbd.org](http://www.drbd.org)

[www.ba.infn.it](http://www.ba.infn.it)

[www.clusterlabs.org](http://www.clusterlabs.org)