

Architetture, Tools e metodologie per lo sviluppo di sistemi di monitoraggio centralizzati per data center distribuiti

Dr Domenico Del Prete
INFN Napoli

MONITORING

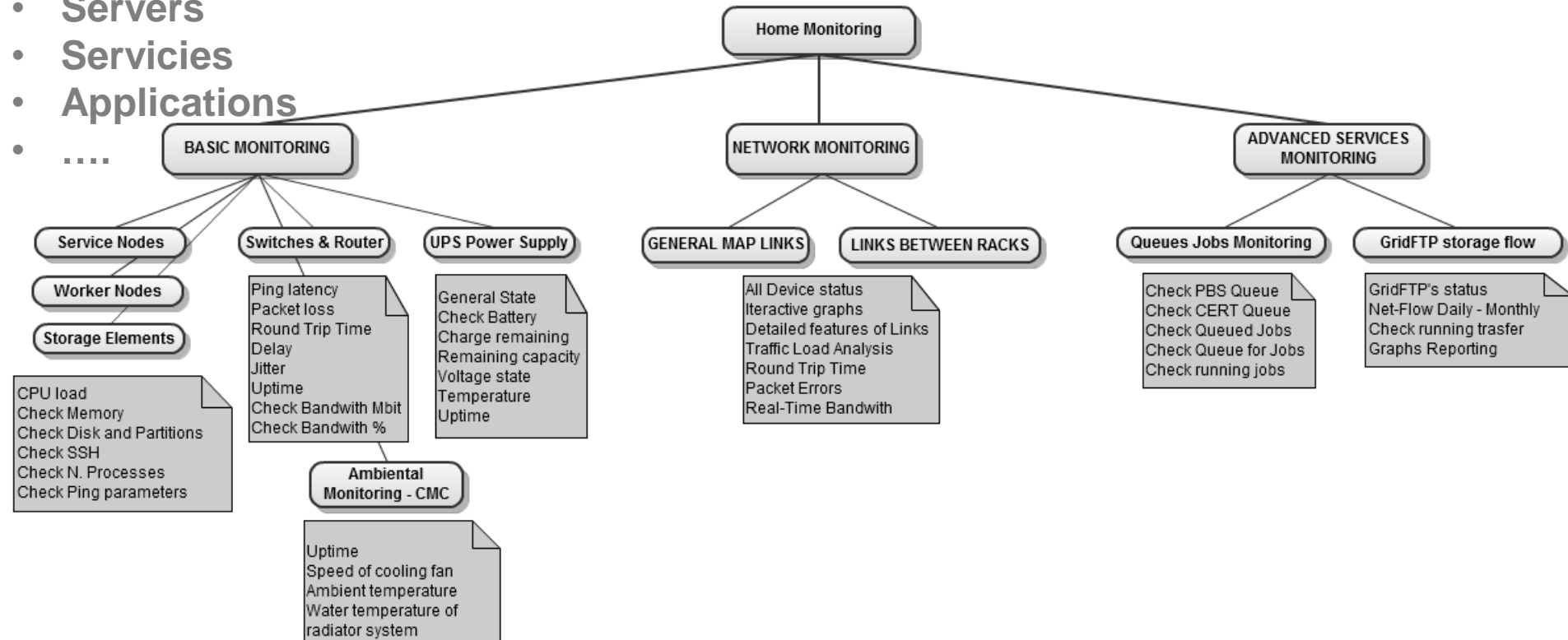
Garantire



Affidabilità
Sicurezza
Performance

Monitoring Multilivello

- Monitoraggio Ambientale
- Power
- Cooling
- Networking
- Servers
- Services
- Applications
-



Sensori di monitoraggio ambientale

Sensore termico

Sensore accesso al rack

Computer Multi Control

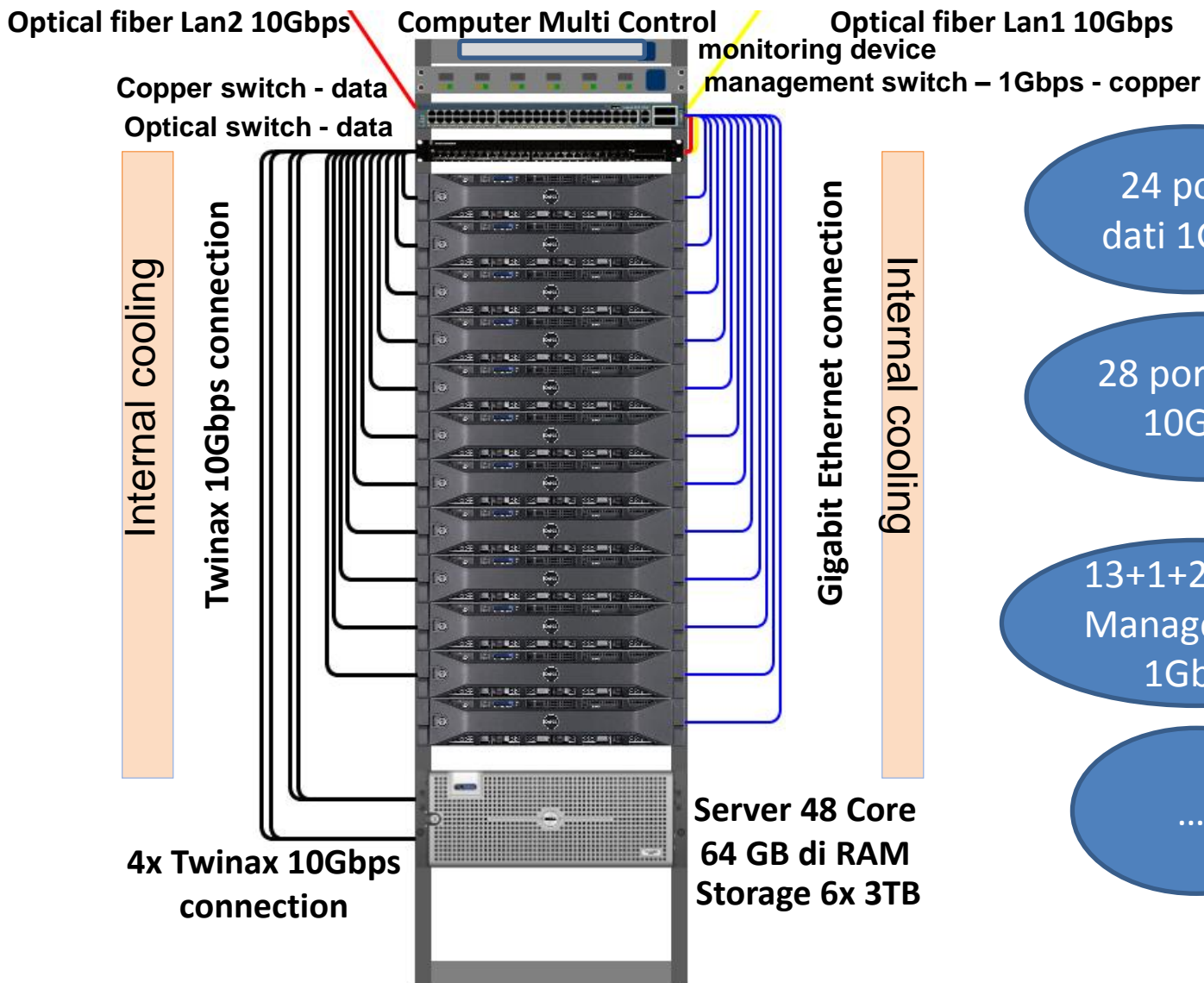
Sensore antifumo

Sensore Allagamento

Power Supply Unit

Liquid Cooling Package





Il protocollo SNMP

(simple network management protocol)

consente la configurazione, la gestione e la supervisione (monitoring) di apparati collegati ad una rete

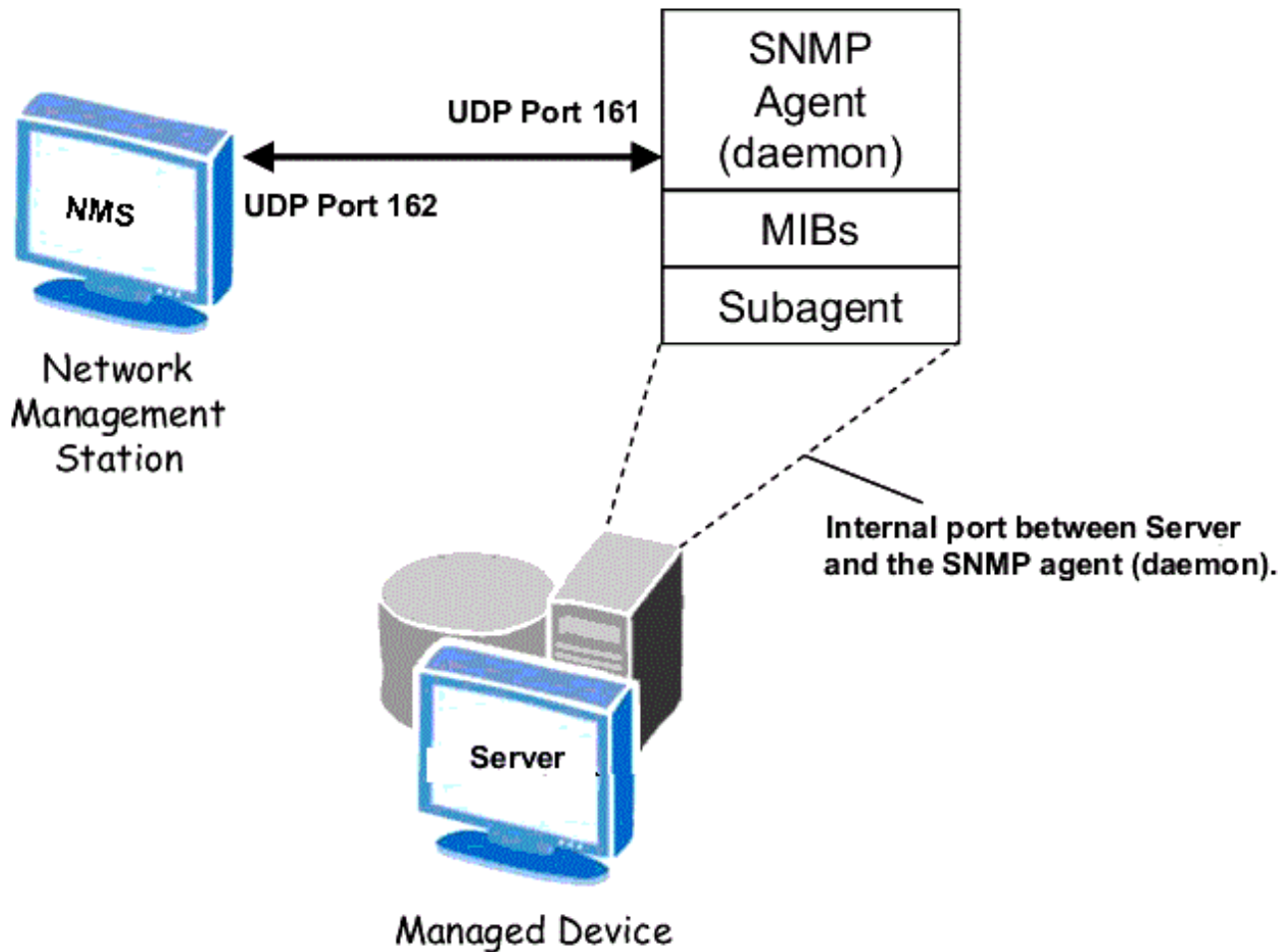


AGENT

- Sistema gestito (nodo cluster, server, switch, router, stampante, pc ...)
 - Agente di gestione (management agent / master agent, subagent)
- Sistema di gestione (Manager) da remoto;

Remote
Manager

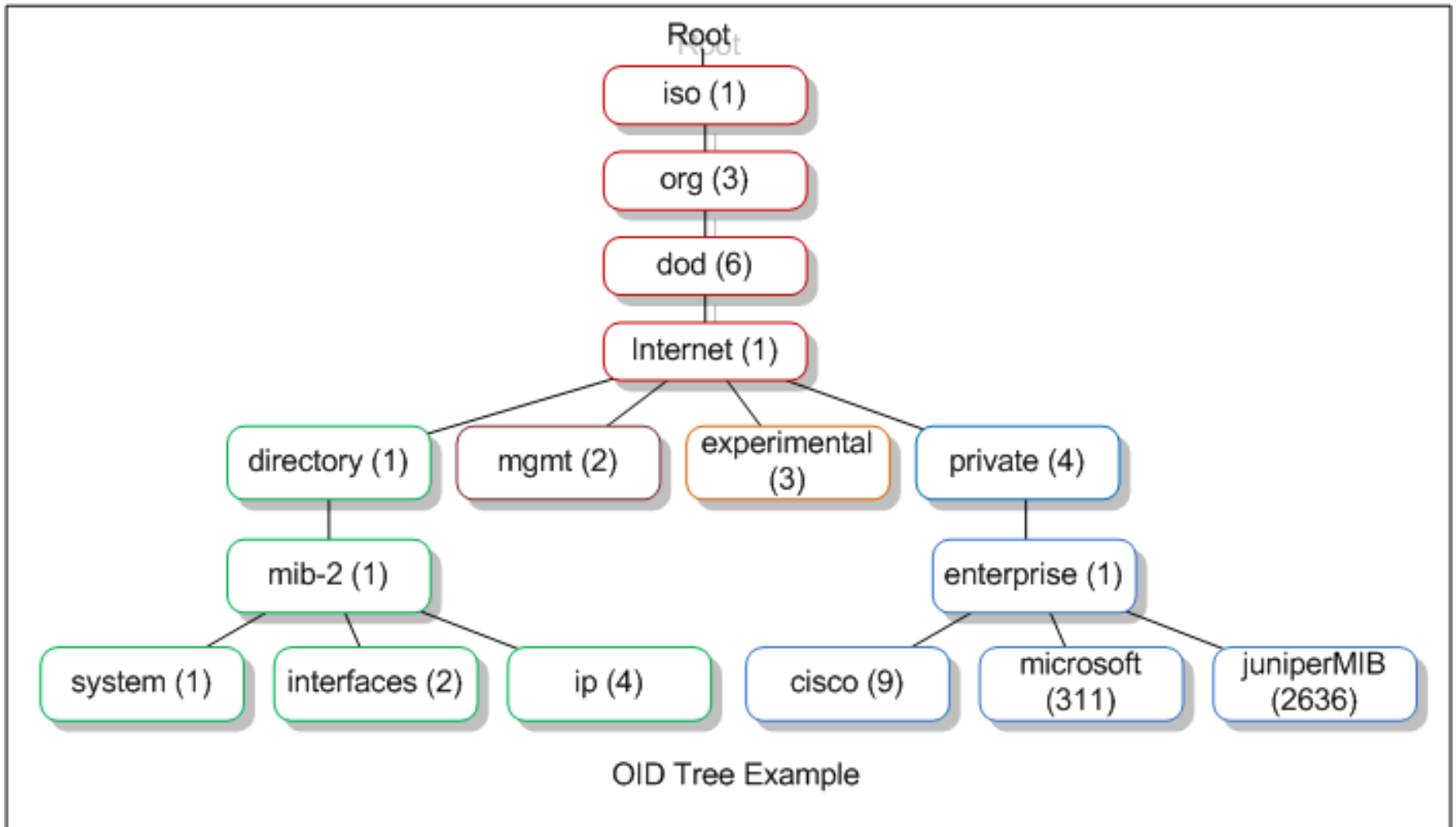
SNMP – le componenti logiche



Management Information Base

- Ogni sottosistema / oggetto gestito è definito da una base di dati detta **MIB**, la quale viene gestita dal subagent di riferimento.
- Database di tipo gerarchico (strutturato ad albero)
 - ogni entry viene indirizzata attraverso un identificatore di oggetto (*object identifier*)
 - Operazioni lettura/scrittura da parte del Manager
- Ogni modifica al database corrisponde un cambiamento di stato del sottosistema e viceversa.

MIB example



Richieste e Notifiche

Richieste alcuni esempi:

- **GET**: usata per leggere uno o più dati di MIB
- **GETNEXT**: usata per leggere iterativamente una sequenza di dati di MIB
- **GETBULK**: usata per leggere con una sola richiesta grandi porzioni di MIB
- **SET**: usata per scrivere (modificare) uno o più dati di MIB

Notifiche:

- Le notifiche sono messaggi inviati dall'agent per segnalare eventi accaduti sul sistema gestito.
 - **Inform** notifiche con previa richiesta dal Manager
 - **trap** notifiche senza alcuna richiesta dal Manager (allarmi in caso di guasti)

Autenticazione ed autorizzazione in SNMP

- Per motivi di sicurezza, i sistemi facenti parte di una rete SNMP vengono raggruppati in *community*
- L'agent SNMP accetta richieste solo da un manager della stessa comunità che si identifica e *autentica* con la stringa della *community*
- L'autorizzazione dei membri di una comunità ad operare su un oggetto può essere di tre tipi:
 - **read**: il manager può interrogare l'agent solo per conoscere lo stato del sistema (solo GET o modalità di sola lettura)
 - **write**: dove il manager può anche variarne l'impostazione (GET e SET, o modalità lettura/scrittura)
 - **trap**: l'agent può inviare trap al manager.

Alcuni Tools di monitoraggio

Nagios®

Per il monitoraggio di tutti i servizi

NagVis

Plug-in di Nagios per la navigazione interattiva

Centreon

Front-end per Nagios – monitoraggio autonomo - configurazione avanzata dei servizi e delle notifiche

CACTI

Monitoraggio dei dispositivi di una rete

Weathermap

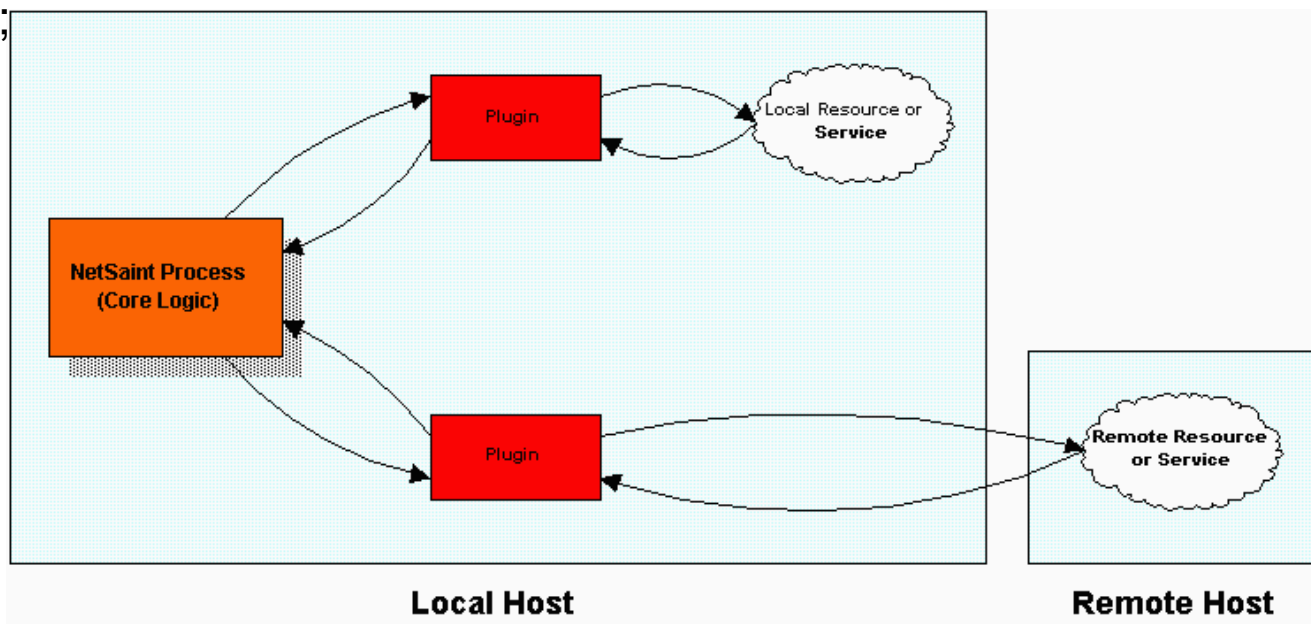
Plug-in di CACTI per la navigazione interattiva

Fully Automated Nagios



Controllo degli stati

- Monitoring delle risorse tramite l'esecuzione di plugin
- Un plugin è un programma (eseguibile o script Perl/sh/...) che può essere eseguito da linea di comando per controllare una risorsa oppure un servizio;
- È possibile specificare (tramite argomenti) delle soglie di allarme (**warning e critical**);



Controllo degli stati

Stati logici di un host

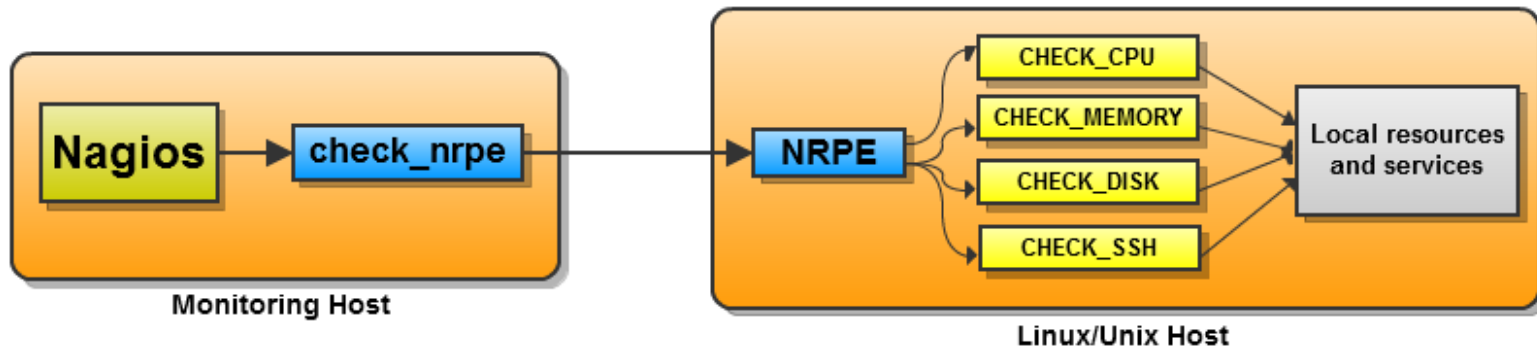
- *UP*
- *PENDING*
- *DOWN*
- *UNREACHABLE*
- *RECOVERED*

Stati logici di un servizio

- *OK*
- *PENDING*
- *WARNING*
- *CRITICAL*
- *UNKNOWN*
- *RECOVERED*

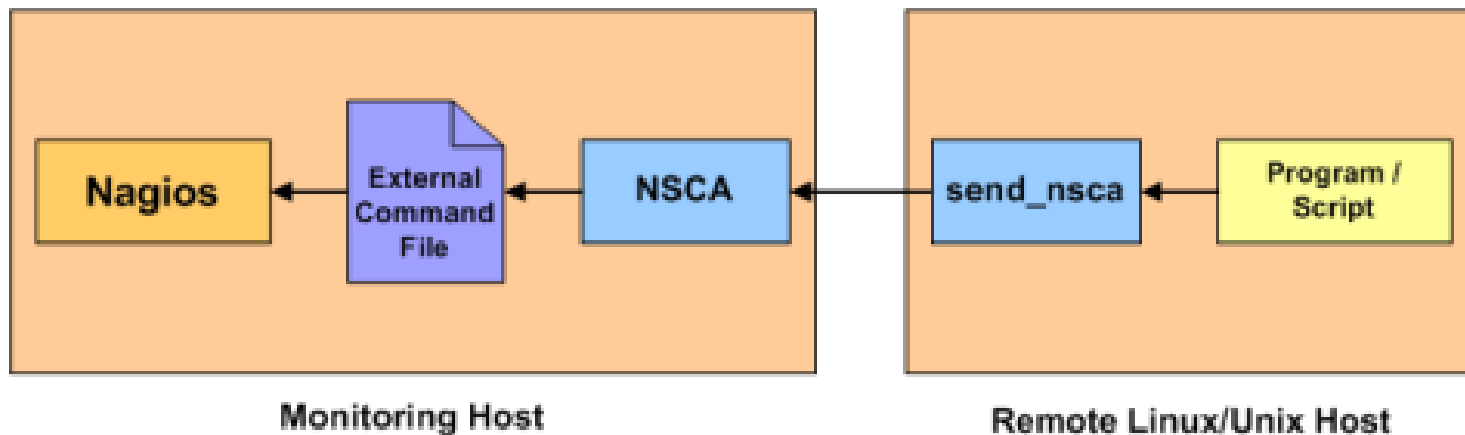
Monitoraggio Remoto – Attivo

- **Remote Active Checks: NRPE (Nagios Remote Plugin Executor)**
Esecuzione su macchine remote di uno o più plugin



Monitoraggio Remoto – Passivo

- **Remote Passive Checks: NSCA (Nagios Service Check Acceptor)**
Esecuzione remota autonoma di plugin e notifica al Server Manager di monitoring

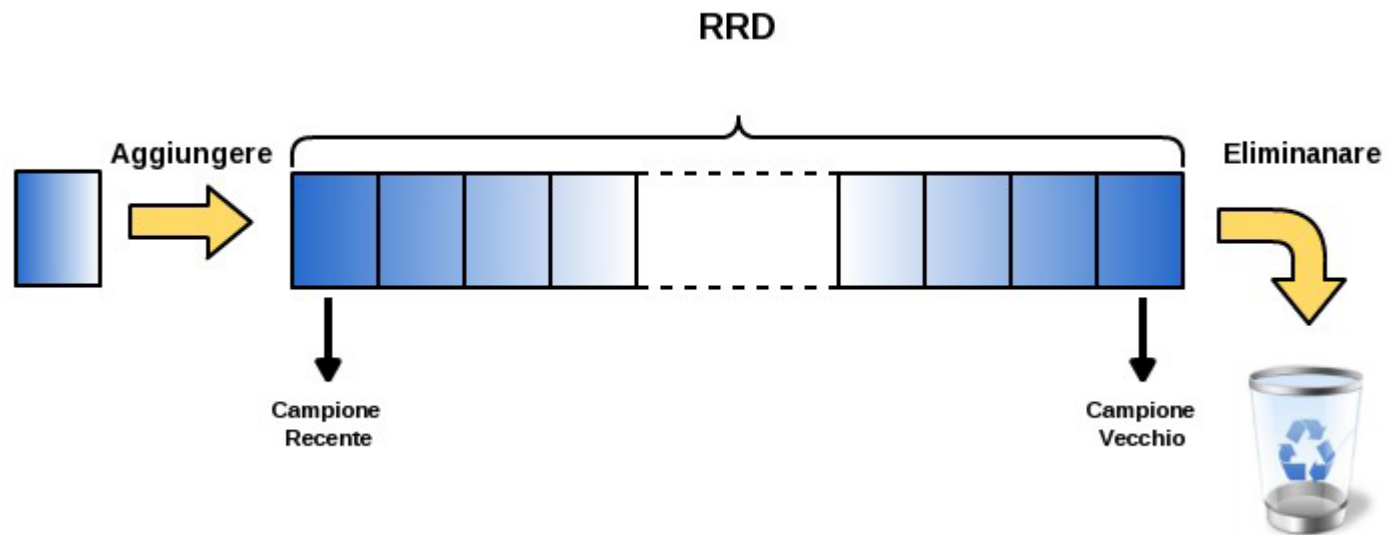


- monitorare servizi e host localizzati dietro un firewall
- monitorare servizi “asincroni” (SNMP traps, security alerts, ecc.);

Nagios/Centreon effettua controlli periodici per verificare che i risultati dei ‘passive check’ siano stati comunicati secondo la frequenza prestabilita

La collezione dei dati: Round Robin Database

- grandezza degli archivi (file RRD) a regime costante
- dump degli archivi in file XML
- prelievo di parziali serie temporali dagli archivi
- rappresentazione dei dati mediante grafici



Esempio di Configurazione

Oggetti

host, host group

contact, contact group

timeperiod, command, service

Host

```
define host {  
    host_name          ns1  
    alias              DNS server #1  
    address            192.168.1.254  
    parents            myrouter [host]  
    check_command      checkhostalive [command]  
    max_check_attempts 5  
    notification_interval 30  
    notification_period 24x7 [timeperiod]  
    notification_options d,u,r  
}
```

Esempio di Configurazione

Host group

```
define hostgroup {  
    hostgroup_name dnsservers  
    alias DNS Servers  
    contact_groups dnsadmins [contactgroup]  
    members ns1,ns2  
}
```

Contact

```
define contact {  
    contact_name  
    alias  
    service_notification_period 24x7 [timeperiod]  
    host_notification_period 24x7 [timeperiod]  
    service_notification_options w,u,c,r  
    host_notification_options d,u,r  
    service_notification_commands notifybyemail [command]  
    host_notification_commands hostnotifybyemail [command]  
    email delprete@na.infn.it  
}
```

Contact group **Esempio di Configurazione**

```
define contactgroup {  
    contactgroup_name    dnsadmins  
    alias                DNS Administrators  
    members              delprete, spardi [contact]  
}
```

Time period

```
define timeperiod {  
    timeperiod_name    nonworkhours  
    alias              NonWork Hours  
    Sunday             00:0024:00  
    monday             00:0009:00,17:0024:00  
    tuesday            00:0009:00,17:0024:00  
    wednesday          00:0009:00,17:0024:00  
    thursday           00:0009:00,17:0024:00  
    friday             00:0009:00,17:0024:00  
    saturday           00:0024:00  
}
```

Esempio di Configurazione

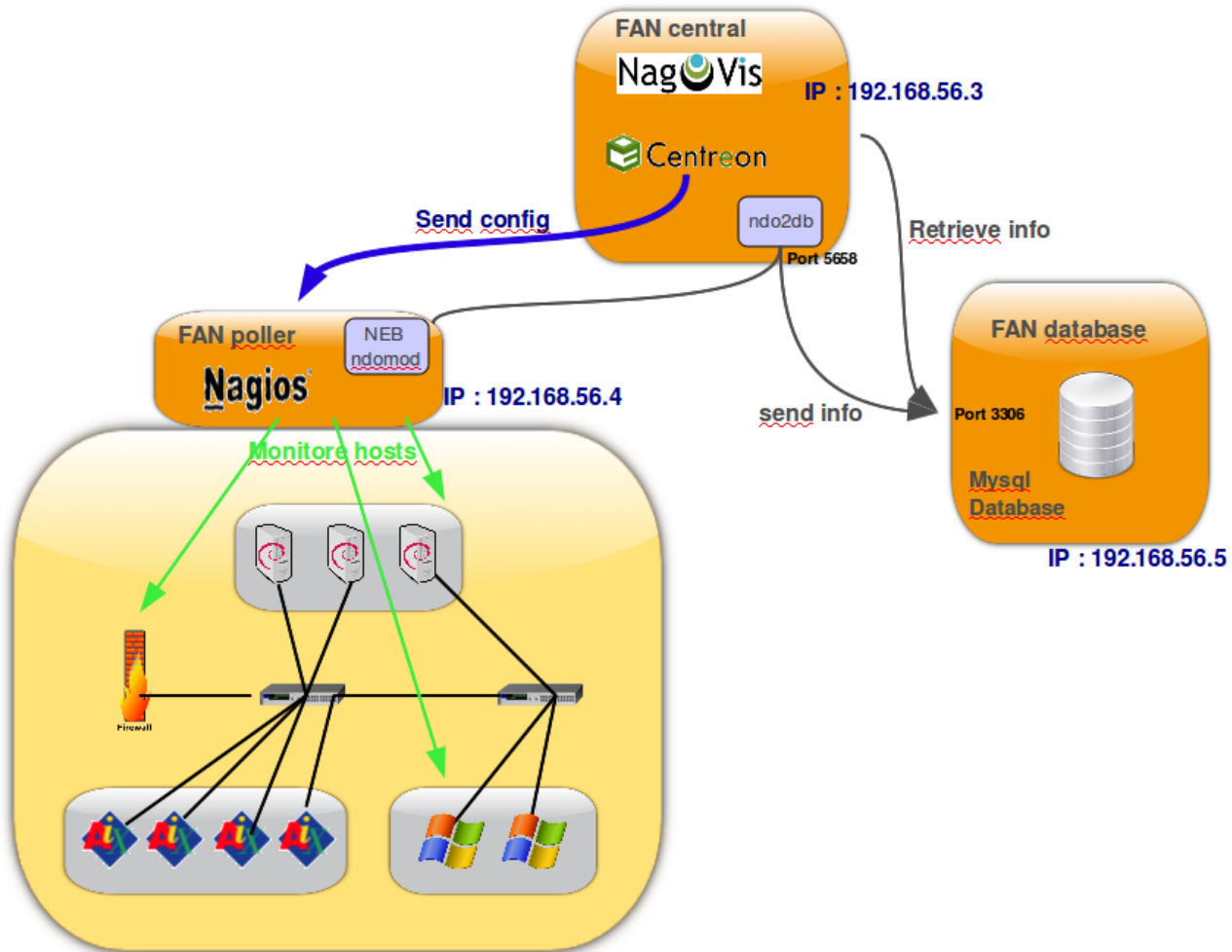
Command

```
define command {  
    command_name check_dns  
    command_line /usr/local/nagios/libexec/check_dns H $HOSTADDRESS$  
.....  
}
```

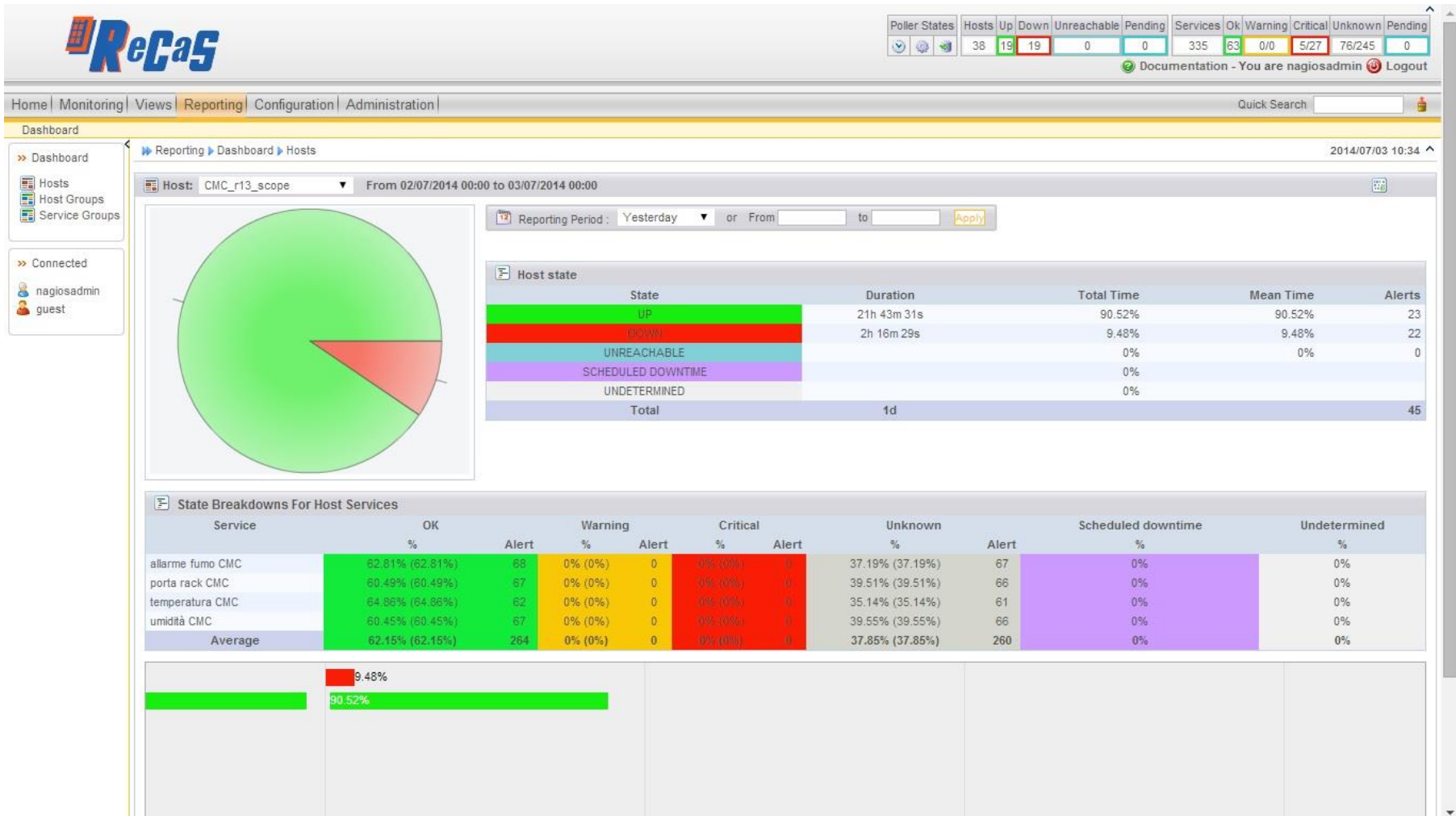
Service

```
define service {  
    host_name ns1 [host]  
    service_description dns  
    check_command check_dns [command]  
    max_check_attempts 5  
    check_period 24x7 [timeperiod]  
    notification_interval 30  
    notification_period 24x7 [timeperiod]  
    notification_options w,c,r  
    contact_groups dnsadmins [contactgroup]  
}
```

Distributed Monitoring Architecture



Fully Automated Nagios (Cetreon)



The screenshot displays the Nagios web interface for host **CMC_r13_scope**. At the top, a status bar shows overall system metrics: 38 Hosts (19 Up, 19 Down, 0 Unreachable, 0 Pending), 335 Services (63 OK, 0 Warning, 5/27 Critical, 76/245 Unknown), and 0 Pending. The interface includes a navigation menu (Home, Monitoring, Views, Reporting, Configuration, Administration) and a search bar.

The main content area shows a pie chart representing the host's state distribution and a table titled "Host state" with the following data:

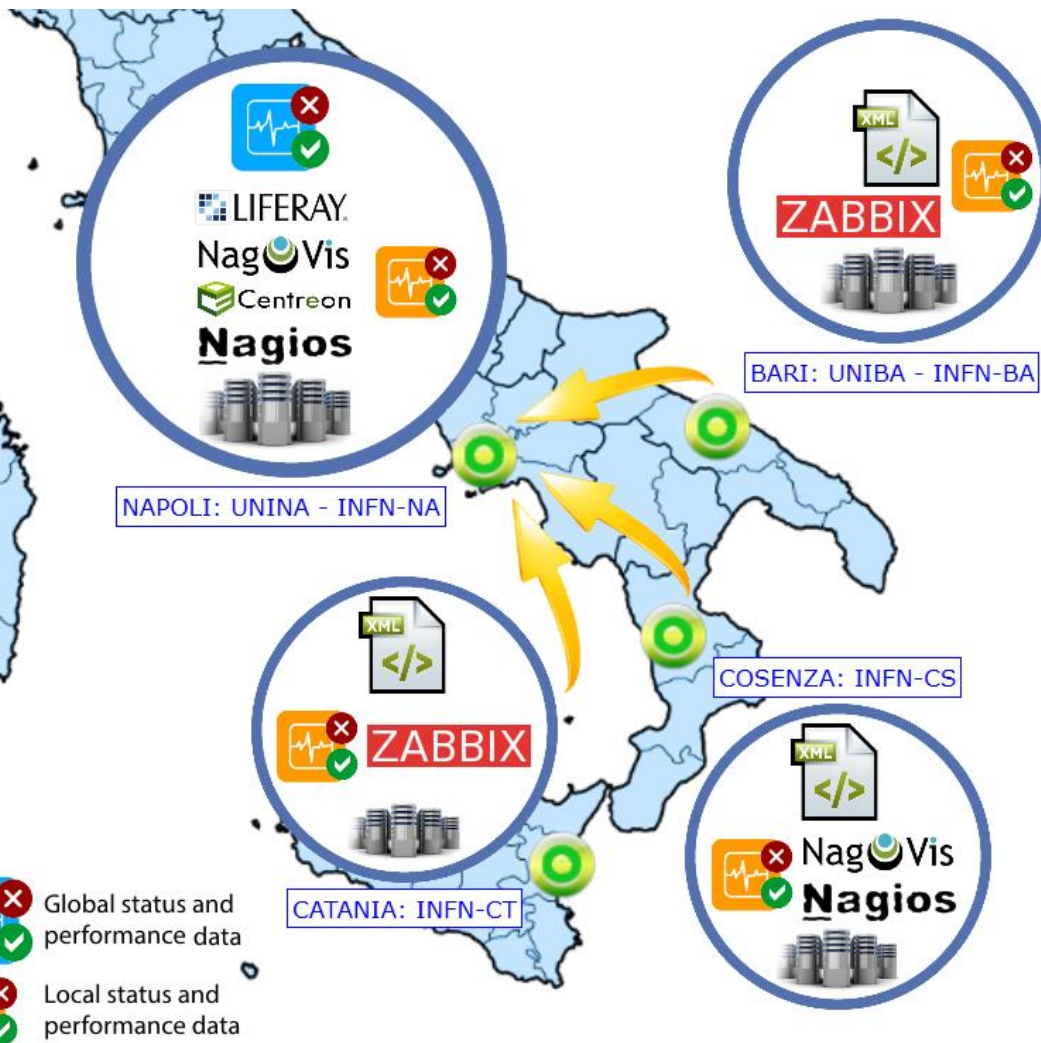
State	Duration	Total Time	Mean Time	Alerts
UP	21h 43m 31s	90.52%	90.52%	23
DOWN	2h 16m 29s	9.48%	9.48%	22
UNREACHABLE		0%	0%	0
SCHEDULED DOWNTIME		0%		
UNDETERMINED		0%		
Total	1d			45

Below this, the "State Breakdowns For Host Services" table provides a detailed view of service health:

Service	OK		Warning		Critical		Unknown	Scheduled downtime		Undetermined
	%	Alert	%	Alert	%	Alert		%	Alert	
allarme fumo CMC	62.81% (62.81%)	68	0% (0%)	0	0% (0%)	0	37.19% (37.19%)	67	0%	0%
porta rack CMC	60.49% (60.49%)	67	0% (0%)	0	0% (0%)	0	39.51% (39.51%)	66	0%	0%
temperatura CMC	64.86% (64.86%)	62	0% (0%)	0	0% (0%)	0	35.14% (35.14%)	61	0%	0%
umidità CMC	60.45% (60.45%)	67	0% (0%)	0	0% (0%)	0	39.55% (39.55%)	66	0%	0%
Average	62.15% (62.15%)	264	0% (0%)	0	0% (0%)	0	37.85% (37.85%)	260	0%	0%

At the bottom, a horizontal bar chart shows the overall state distribution: 9.48% DOWN (red bar) and 90.52% UP (green bar).

Architettura di monitoring ReCaS



- Un sistema di monitoring in ogni sito.
- Un server centrale a Napoli
- Collector delle informazioni dei vari siti e con le mappe generali per la rappresentazione globale.

Singolo sito – alcuni requisiti:

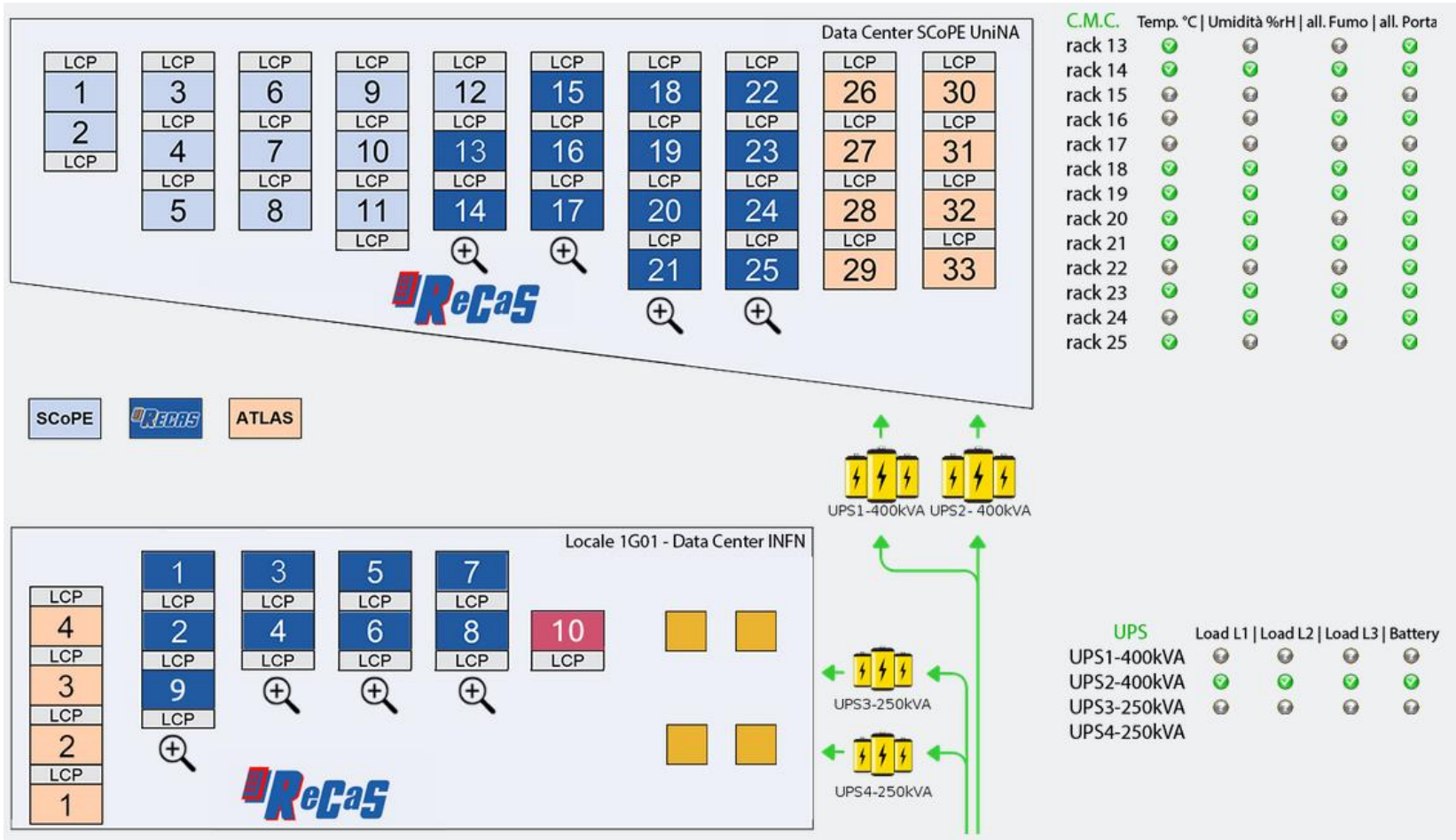
Ogni sito implementa un sistema di monitoraggio locale che andrà a raccogliere le informazioni degli apparati della specifica sede.

Tale sistema potrà essere basato su Nagios, Zabbix o altri tools, e dovrà avere i seguenti requisiti:

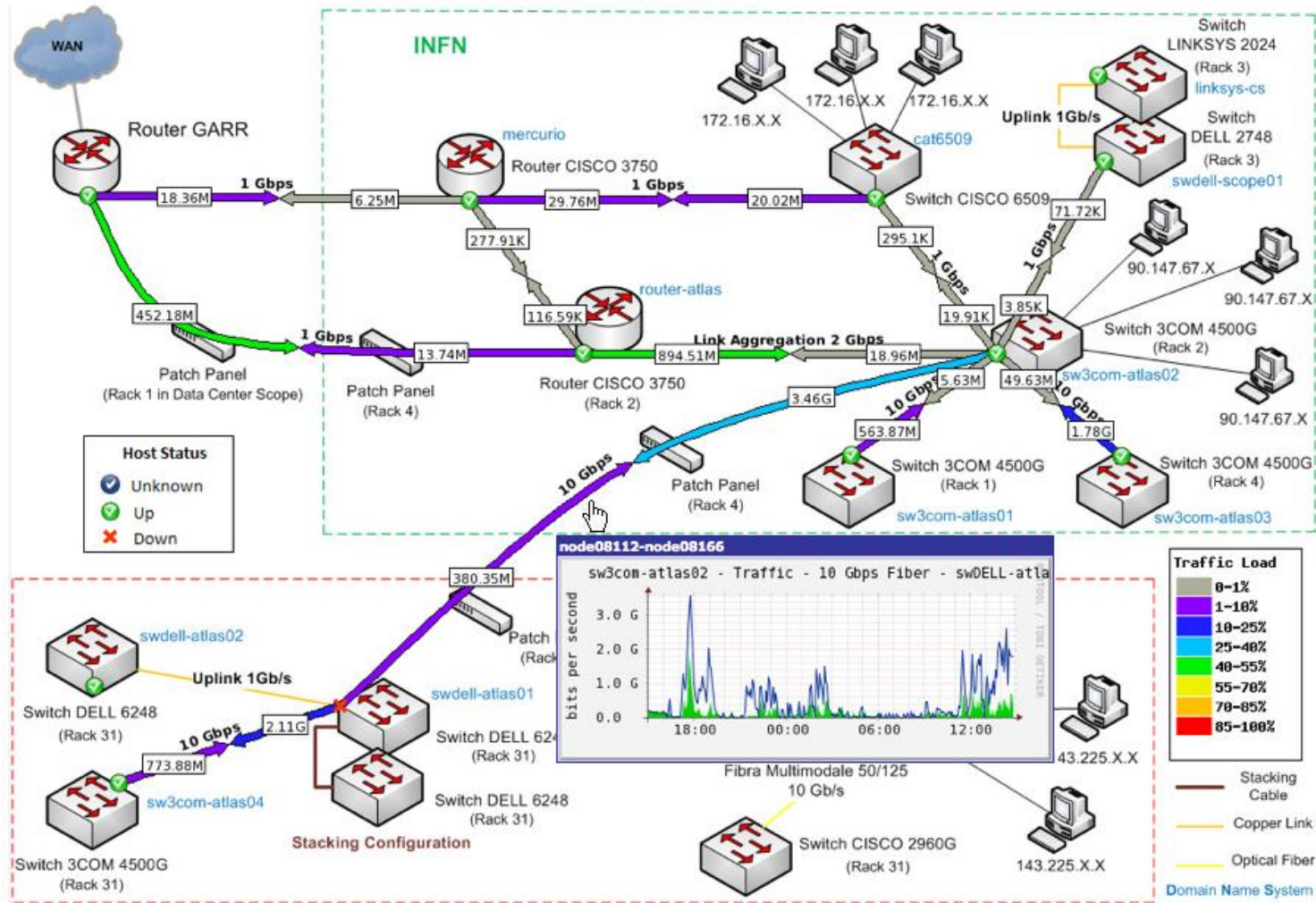
- Un end-point accessibile dall'esterno
- Possibilità di creare mappe grafiche per rappresentare l'infrastruttura
- Un user guest che consente di vedere informazioni di pubblico dominio
- Un superuser che consente di vedere informazioni più dettagliate

Ogni sito crea delle mappe locali per rappresentare in maniera grafica e semplificata lo stato dell'infrastruttura e delle attrezzature.

Example: ReCaS-Napoli services and application monitoring



Example: Tier2-Napoli Network monitoring



- Creare la macchina virtuale con la ISO (FAN-2.4-x86_64.iso) presente della directory monitoring
- Creare 2 macchine virtuali (host) basate su RedHat Linux
- Le 3 VM devono trovarsi sulla stessa rete locale
- Seguire le istruzioni successive

SNMP Configuration

- `yum install net-snmp net-snmp-libs net-snmp-utils`

Package	Provides
net-snmp	The SNMP Agent Daemon and documentation. This package is required for exporting performance data.
net-snmp-libs	The <code>net-snmp</code> library and the bundled management information bases (MIBs). This package is required for exporting performance data.
net-snmp-utils	SNMP clients such as <code>snmpget</code> and <code>snmpwalk</code> . This package is required in order to query a system's performance data over SNMP.
net-snmp-perl	The <code>mib2c</code> utility and the <code>NetSNMP</code> Perl module.
net-snmp-python	An SNMP client library for Python.

- `service snmpd start`

To configure the service to be automatically started at boot time, use the following command:

```
chkconfig snmpd on
```

SNMP Configuration

```
mv /etc/snmp/snmpd.conf /etc/snmp/snmpd.conf.org  
nano /etc/snmp/snmpd.conf
```

Create a new /etc/snmp/snmpd.conf file:

```
rocommunity public  
syslocation RECAS, "INFN DataCenter"  
syscontact vostronome@dominio.it
```

```
service snmpd start  
(check_memory OID)  
snmpwalk -v2c -c public localhost .1.3.6.1.4.1.2021.4.5.0
```

Configurazioni tramite il front-end Centreon

Testare un'istanza di configurazione per il controllo della memoria principale

Sezione *command*:

```
$USER1$/check_snmp -H $HOSTADDRESS$ -C public -o  
.1.3.6.1.4.1.2021.4.5.0 -l 'Total mem' -u 'kB'
```


Centreon Configuration

- **Creare un comando nella sezione COMMANDS con l'esecutore snmp_check**
- **Creare un Template in Hosts**
- **Create un Templates in Services**
- **Creare un Host ed importare il template**
- **Creare un servizio in Services ed associare il comando di pertinenza**
- **Associare il servizio all'Host d'interesse nella sezione Relations**
-

SNMP Checks: CPU Statistics

1 minute Load: .1.3.6.1.4.1.2021.10.1.3.1

5 minute Load: .1.3.6.1.4.1.2021.10.1.3.2

15 minute Load: .1.3.6.1.4.1.2021.10.1.3.3

SNMP Checks: CPU Statistics

Total Swap Size: .1.3.6.1.4.1.2021.4.3.0

Available Swap Space: .1.3.6.1.4.1.2021.4.4.0

Total RAM in machine: .1.3.6.1.4.1.2021.4.5.0

Total RAM used: .1.3.6.1.4.1.2021.4.6.0

Total RAM Free: .1.3.6.1.4.1.2021.4.11.0

Total RAM Shared: .1.3.6.1.4.1.2021.4.13.0

Total RAM Buffered: .1.3.6.1.4.1.2021.4.14.0

Total Cached Memory: .1.3.6.1.4.1.2021.4.15.0