



Gestión de Redes

NAGIOS



These materials are licensed under the Creative Commons *Attribution-Noncommercial 3.0 Unported* license
(<http://creativecommons.org/licenses/by-nc/3.0/>)

Introduccción

Herramientas de Monitoreo de Redes

- Disponibilidad
- Confiabilidad
- Rendimiento

*Nagios activamente monitorea la
disponibilidad de dispositivos y servicios*

Introducción

- Una herramienta clave de medir en forma activa la disponibilidad de dispositivos y servicios
- Posiblemente el software de fuente abierto mas usado para gestion y monitoreo de redes
- Tiene un interfaz de web
 - Usa CGIs escrito en C para responder mas rapido y apoyar crecimiento
- Apoya hasta miles de dispositivos y servicios.

Instalación

En Debian/Ubuntu

```
# apt-get install nagios3
```

Directorios importantes

```
/etc/nagios3
```

```
/etc/nagios3/conf.d
```

```
/etc/nagios-plugins/conf
```

```
/usr/lib/nagios/plugins
```

```
/usr/share/nagios3/htdocs/images/logos
```

El interfaz web de Nagios esta aquí:

<http://pcN.ws.nsrc.org/nagios3/>

Plugins

Plugins estan usados para verificar servicios y dispositivos/equipos:

- La arquitectura de Nagios es bastante simple que escribir plugins nuevos es fácil y en el lenguaje de su eleccion.
- Hay ***muchos*** plugins disponibles
 - ✓ <http://exchange.nagios.org/>
 - ✓ <http://nagiosplugins.org/>



Características

- Configuración esta hecha en archivos de texto basado en plantillas.
- Nagios lee su configuración de un directorio – Ud. decide como dividir los archivos de configuración en ese directorio.
- Usa chequeos en paralelo y bifurcación (forking) por escalabilidad.

Características

- Utiliza información topológica para determinar dependencias
 - Diferenciación entre lo que está 'caído' y lo que está 'inalcanzable' – así no se tratar de hacer chequeos de los máquinas no 'inalcanzable'
- Permite definir políticas de notificación, basadas en combinaciones de:
 - contactos y listas de contactos
 - dispositivos y grupos de dispositivos
 - servicios y grupos de servicios.
 - horarios definidos por grupos o personas.
 - El estado de servicio:

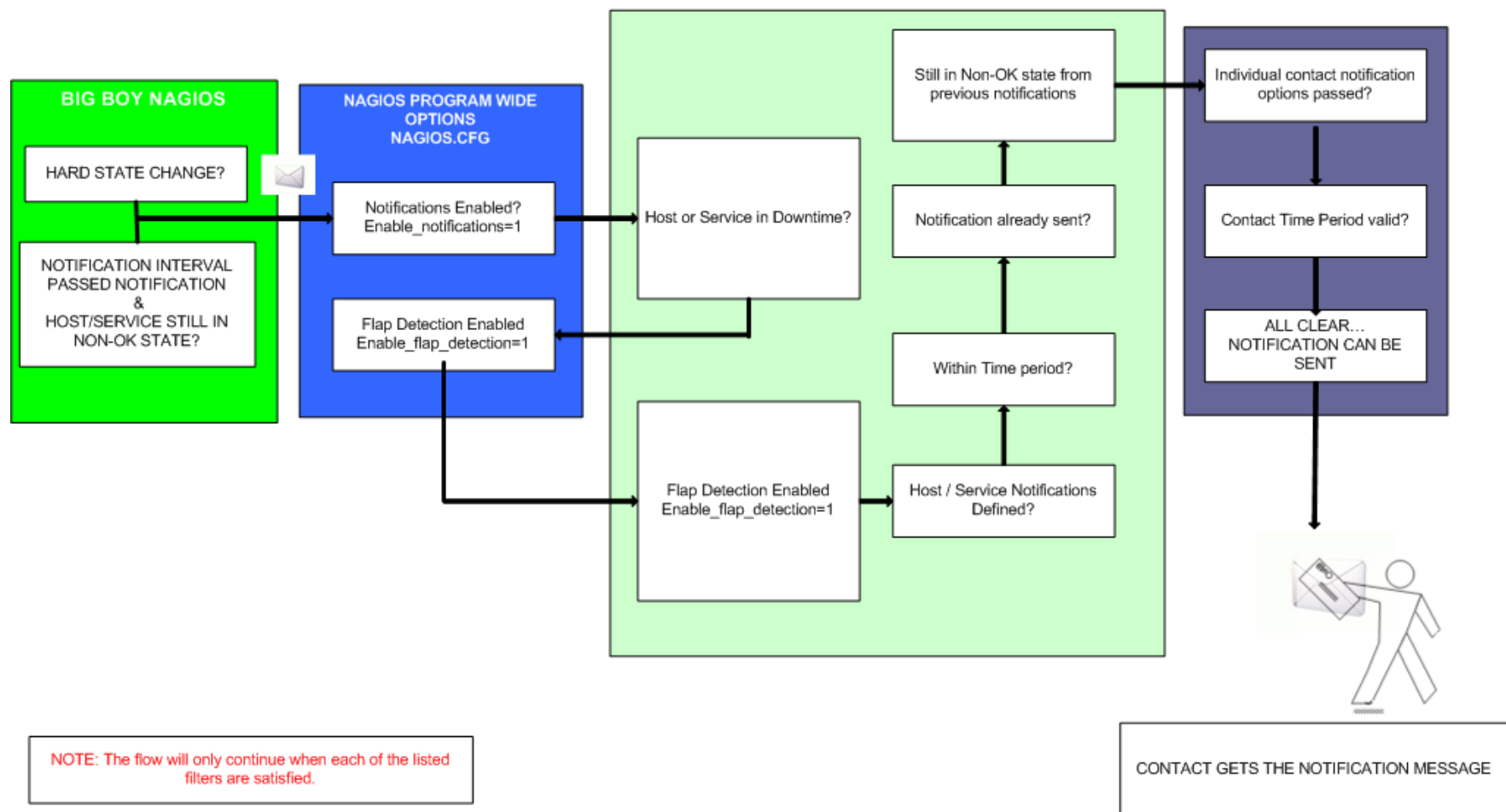
Opciones de Notificacion (Host)

Estado de Servicio:

Cuando uno configura un servicio tiene los siguiente opciones:

- d:** DOWN: El servicio esta caido (no disponible)
- u:** UNREACHABLE: Dispositivo no esta visible
- r:** RECOVERY: (OK) Dispositivo recuperando
- f:** FLAPPING: La primera vez con un dispositivo sube, bajo o esta en un estado indeterminado
- n:** NONE: No manda ningun notificacion

NAGIOS - NOTIFICATION FLOW DIAGRAM



Mecánica de chequeos

Uno nodo o dispositivo (host) consta de uno o más servicios a chequear (PING, HTTP, MYSQL, SSH, etc)

Nagios chequea periódicamente cada servicio de cada nodo y determina si ha habido algún cambio de estado:

- CRITICAL
- WARNING
- UNKNOWN

A cada cambio de estado, se le puede asignar:

- Opciones de notificación (como vimos antes)
- Operaciones de manejo de eventos (event handlers)

Mecánica de chequeos cont.

Parámetros

- Intervalo de chequeo normal
- Intervalo de re-chequeo
- Número máximo de chequeos
- Período de chequeo

Los chequeos de nodo (host) sólo se ejecutan cuando ninguno de los servicios responde

- Uno nodo (host) puede estar:
 - DOWN
 - UNREACHABLE

How checks work continued

- Por defecto Nagios se lo hace un chequeo de nodo 3 veces antes que se cambia el estado de un nodo a *down*.
- El estado de no respuesta va de *warning* a *critical*

Concepto de “Padres”

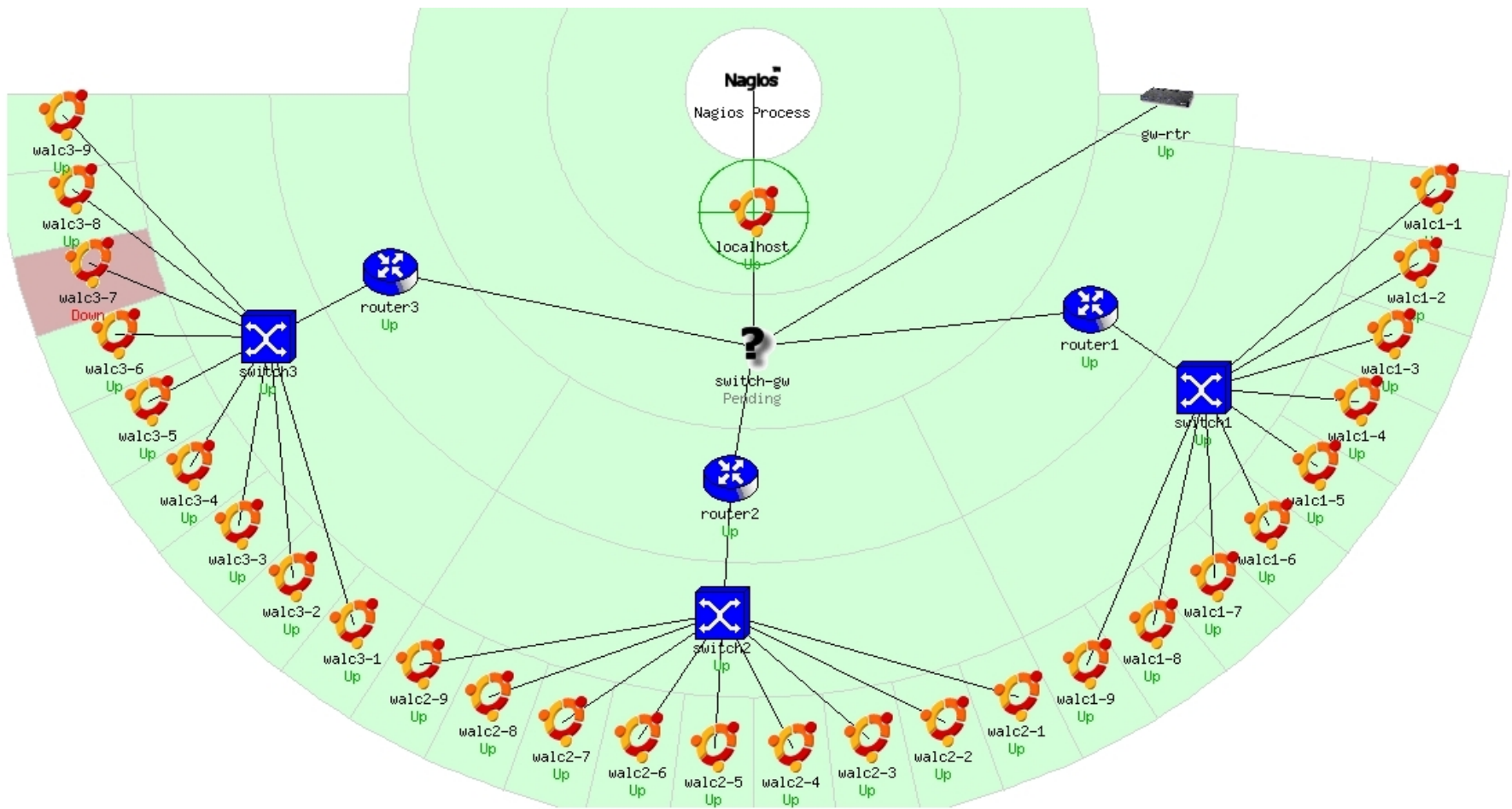
- El padre de un PC conectado a un switch seria el switch.
- Nos permite especificar las dependencias entre dispositivos (PCs, switches, enrutadores)
- Evita mandar multiples alarmas cuando el padre no responde. Es un sistema jerarquico.
- Un nodo puede tener multiple padres (dual homes).



Punto de Vista de la Red

- Donde pones Nagios va a determinar su punto de vista de la red.
- El servidor de Nagios se pone como el “raiz” de su arbol de dependencias.

Punto de Vista de la Red



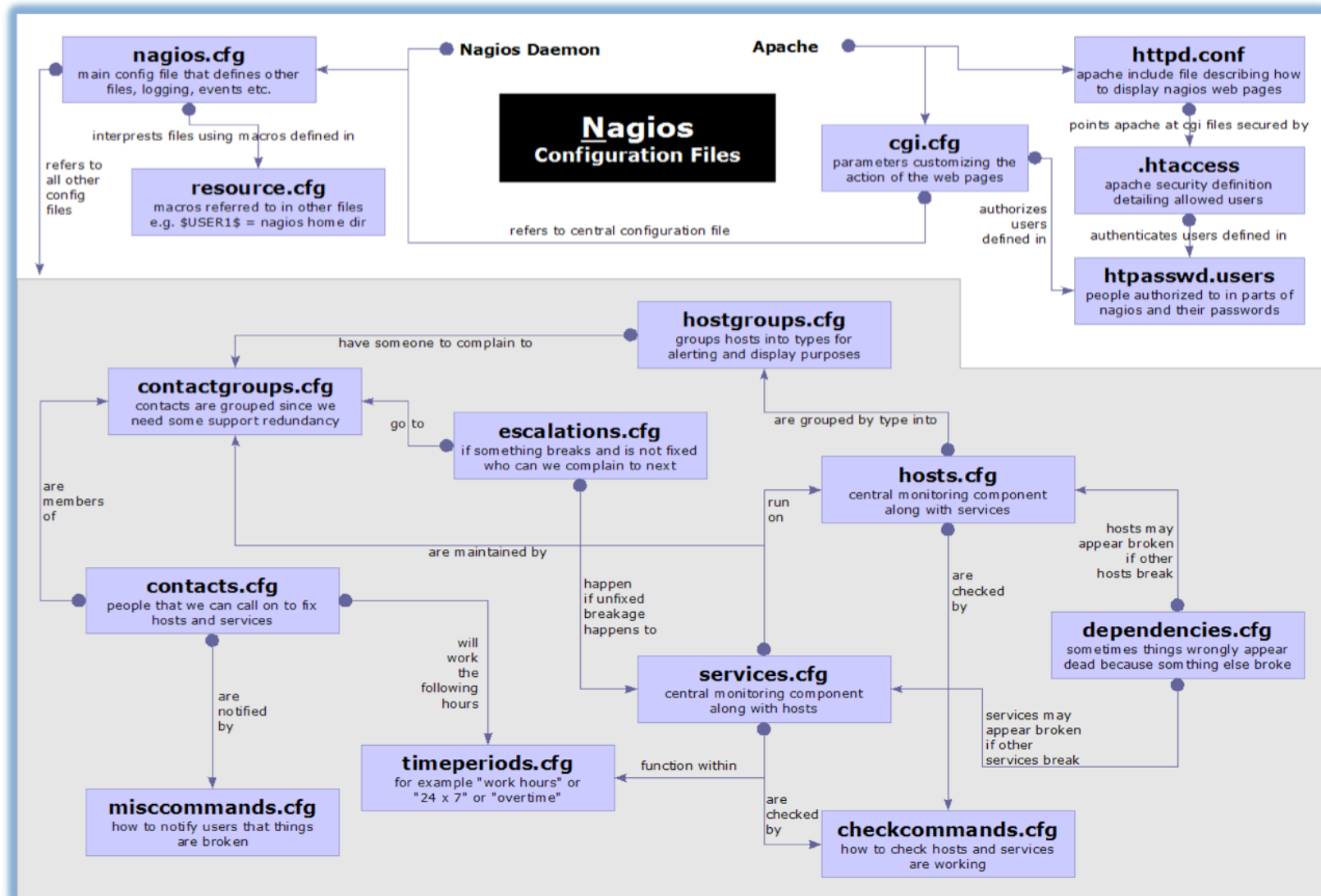
A thick, dark red vertical bar is positioned on the left side of the slide, extending from the top header area down to the bottom.

Demonstracion de Nagios

Archivos de Configuracion

- Muchos! Parece complejo al principio
- **Orientada a objetos**
 - Objetos (dispositivos o servicios) heredan los atributos
 - Permite aplicar funcionalidad a grupos de dispositivos o servicios.
 - No aplica funcionalidad a objetos individuales. No se escala!
 - Despues que entiende los archivos de configuracion de Nagios el resto es facil.

Archivos de Configuración (Oficial)



Archivos de Configuración

Ubicados en `/etc/nagios3/`

Archivos importantes incluyen:

- `cgi.cfg` Controla el interfaz de Web y las opciones de seguridad.
- `commands.cfg` Los comandos que usa Nagios para notificaciones.
- `nagios.cfg` El archivo principal de configuración
- `conf.d/*` El resto de los archivos de configuración por servicios, grupos, nodos, etc.

Archivos de Configuracion

Bajo conf.d/*

- `contacts_nagios2.cfg` Usuarios y grupos
- `extinfo_nagios2.cfg` Haz to UI bonito
- `generic-host_nagios2.cfg` Plantilla de host por defecto
- `generic-service_nagios2.cfg` Plantilla de servicio p/defecto
- `host-gateway_nagios3.cfg` Definicion enlace de puerto
- `hostgroups_nagios2.cfg` Engrupacion de nodos
- `localhost_nagios2.cfg` Definicion de localhost
- `services_nagios2.cfg` Que servicios a chequear
- `timeperiods_nagios2.cfg` Cuanda hacer chequeo
Quien a notificar

Archivos de Configuracion

Bajo /etc/nagios3/conf.d puede hacer (por ejemplo):

- [servicegroups.cfg](#) Engrupacion de servicios y nodos
- [pcs.cfg](#) Definicion de los PCs en su red
- [switches.cfg](#) Definicion de los switches
- [routers.cfg](#) Definicion de los enrutadores

PCs, switches y enrutadores son (hosts). Se defina servicios a monitorear por los hosts (si quieres).

Los plugins instalado por Ubuntu

/usr/lib/nagios/plugins

check_apt	check_file_age	check_jabber	check_nttp	check_procs	check_swap
check_bgpstate	check_flexlm	check_ldap	check_nttps	check_radius	check_tcp
check_breeze	check_ftp	check_ldaps	check_nt	check_real	check_time
check_by_ssh	check_host	check_linux_raid	check_ntp	check_rpc	check_udp
check_cload	check_hppjd	check_load	check_ntp_peer	check_rta_multi	check_ups
check_cluster	check_http	check_log	check_ntp_time	check_sensors	check_users
check_dhcp	check_icmp	check_mailq	check_nwstat	check_simap	check_wave
check_dig	check_idc_smart	check_mrtg	check_oracle	check_smtp	negate
check_disk	check_ifoperstatus	check_mrtgtraf	check_overcr	check_snmp	urlize
check_disk_smb	check_ifstatus	check_mysql	check_pgsql	check_spop	utils.pm
check_dns	check_imap	check_mysql_query	check_ping	check_ssh	utils.sh
check_dummy	check_ircd	check_nagios	check_pop	check_ssmtp	

/etc/nagios-plugins/config

apt.cfg	disk-smb.cfg	ftp.cfg	ldap.cfg	mysql.cfg	ntp.cfg	radius.cfg	ssh.cfg
breeze.cfg	dns.cfg	hppjd.cfg	load.cfg	netware.cfg	pgsql.cfg	real.cfg	tcp_udp.cfg
dhcp.cfg	dummy.cfg	http.cfg	mail.cfg	news.cfg	ping.cfg	rpc-nfs.cfg	telnet.cfg
disk.cfg	flexlm.cfg	ifstatus.cfg	mrtg.cfg	nt.cfg	procs.cfg	snmp.cfg	users.cfg

Configuración de Nodos y Servicios

Basado en Plantillas

- Ahorra tiempo evitando repetición
- Similar a la programación orientada a objetos

Nagios viene con una plantilla por defecto con parámetros por un:

- *nodo generico* (generic-host_nagios2.cfg)
- *servicio generico* (generic-service_nagios2.cfg)
- contacto generico (contacts_nagios2.cfg)

Plantilla de nodo generico

generic-host_nagios2.cfg

```
define host{
    name                generic-host    ; The name of this host template
    notifications_enabled 1            ; Host notifications are enabled
    event_handler_enabled 1            ; Host event handler is enabled
    flap_detection_enabled 1           ; Flap detection is enabled
    failure_prediction_enabled 1       ; Failure prediction is enabled
    process_perf_data     1            ; Process performance data
    retain_status_information 1        ; Retain status information across program restarts
    retain_nonstatus_information 1     ; Retain non-status information across program restarts
    check_command          check-host-alive
    max_check_attempts     10
    notification_interval  0
    notification_period     24x7
    notification_options    d,u,r
    contact_groups          admins
    register                0          ; DONT REGISTER THIS DEFINITION - ITS NOT A REAL HOST, JUST A TEMPLATE!
}
```

Configuracion de un nodo individual

```
define host{
    use                generic-host
    host_name          gw-rtr
    alias              Enrutador principal de taller
    address            10.10.0.254
    contact_groups     router_group
}
```

Plantilla de servicio generico

generic-service_nagios2.cfg

```
define service{
    name                                generic-service
    active_checks_enabled               1
    passive_checks_enabled              1
    parallelize_check                   1
    obsess_over_service                 0
    check_freshness                     1
    notifications_enabled               1
    event_handler_enabled               1
    flap_detection_enabled              1
    process_perf_data                   1
    retain_status_information            1
    retain_nonstatus_information        1
    is_volatile                         0
    check_period                        24x7
    max_check_attempts                  5
    normal_check_interval                5
    retry_check_interval                 1
    notification_interval                60
    notification_period                  24x7
    notification_options                 c,r
    register                            0
}
```

Configuracion un servicio individual

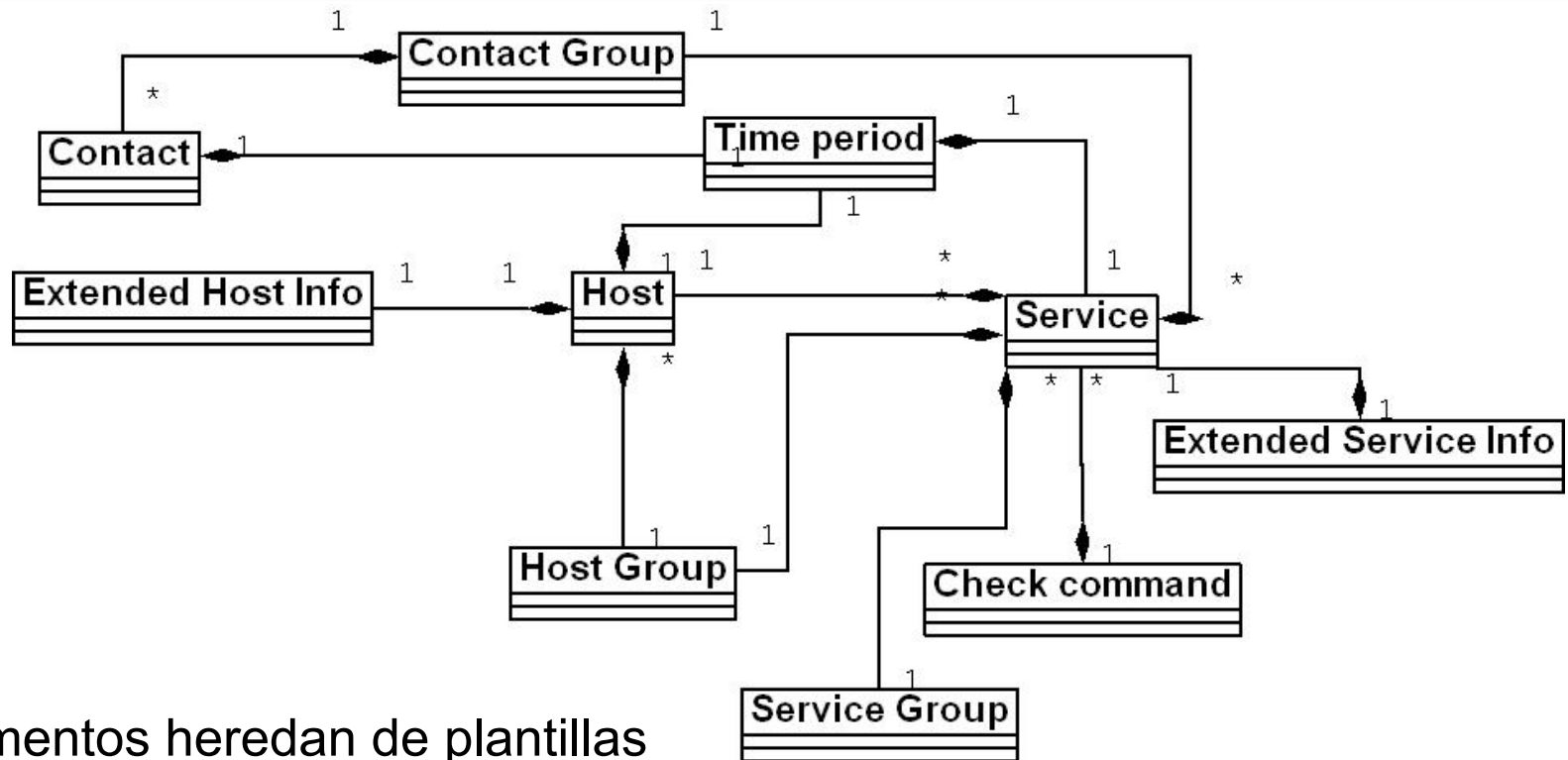
```
define service{
    hostgroup_name      servers
    service_description  PING
    check_command        check-host-alive
    use                 generic-service
    max_check_attempts  5
    normal_check_interval 5
    notification_options c,r,f
    notification_interval 0 ; set > 0 if you want to be renotified
}
```

c: Critico

r: Recuperando

f: Aleteo (Flapping)

Flujo de configuracion

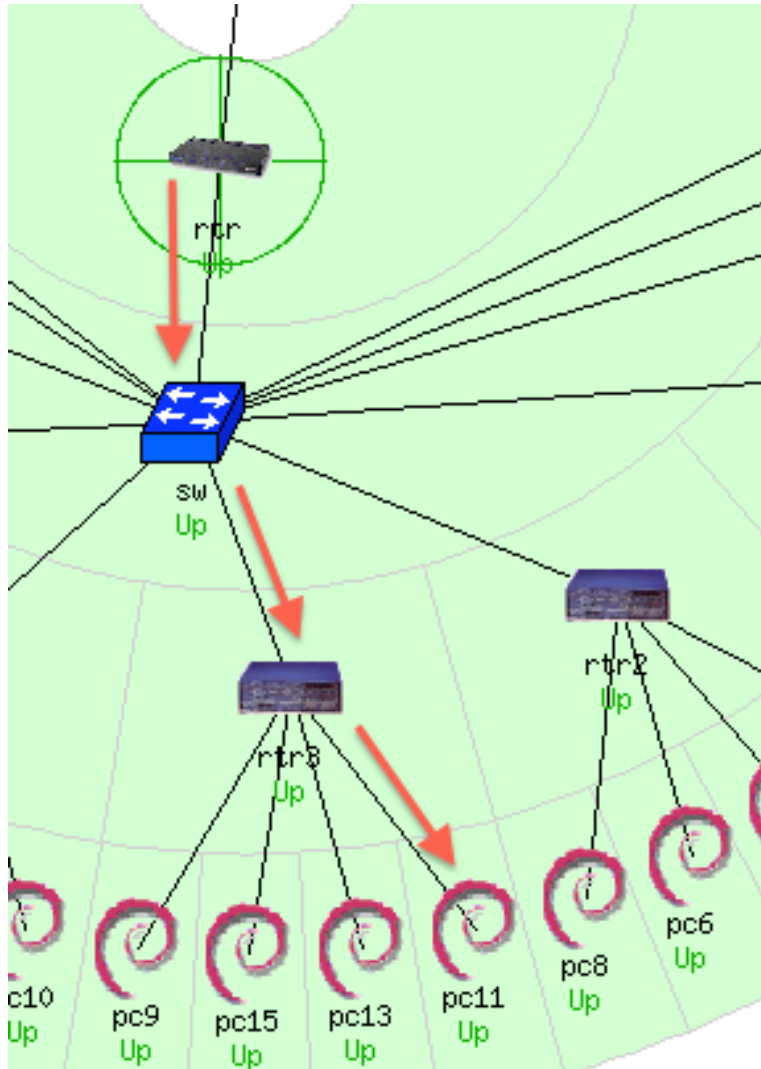


Elementos heredan de plantillas

Empezmos con los hosts (PCs, switches y enrutadores)

- Poner multiple hosts en un grupo
- Definir los padres de cada host
- Agrega un chequeo de servicio al grupo (no a los hosts individuales)
- Agrega informacion extendida, si haya

Otra vista de flujo de configuracion



RTR

```
define host {  
  use  
  host_name  
  alias  
  address
```

```
generic-host  
rtr  
Enrutador GW  
10.10.0.254 }
```

SW

```
define host {  
  use  
  host_name  
  alias  
  address  
  parents
```

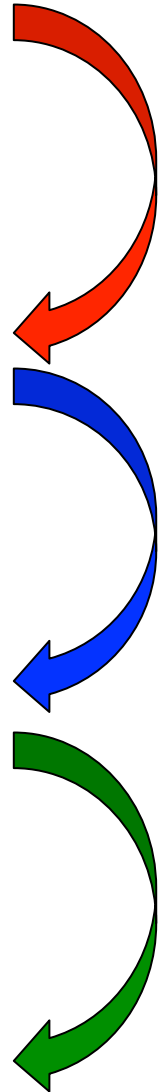
```
generic-host  
sw  
Switch Dorsal  
10.10.0.253  
rtr }
```

RTR3

```
define host {  
  use  
  host_name  
  alias  
  address  
  parents
```

```
generic-host  
rtr3  
router 3  
10.10.3.254  
sw }
```

PC11...



Notificaciones de OoB

Es importante recordar un sistema de mensaje de texto o mensajes que es independiente de su red.

- Puede utilizar un celular conectado a su servidor de Nagios
- Puede usar software como:
 - gnokii:** <http://www.gnokii.org/>
 - qpage:** <http://www.qpage.org/>
 - sendpage:** <http://www.sendpage.org/>

Referencias

- **Sitio de web Nagios**
<http://www.nagios.org/>
- **Sitio de web de Plugins por Nagios**
<http://www.nagiosplugins.org/>
- *Nagios System and Network Monitoring*, por Wolfgang Barth. Un buen libro sobre Nagios.
- **Sitio no oficial de Plugins por Nagios**
<http://nagios.exchange.org/>
- **Una enseñanza de Debian sobre Nagios**
<http://www.debianhelp.co.uk/nagios.htm>
- **Aporte Comercial por Nagios**
<http://www.nagios.com/>

Preguntas?

?