Introduction to Large Scale Machine Management (second part)

DAAD Summer School: Aspects of Large Scale High Speed Computing

15th March 2011

Dr. Dirk von Suchodoletz

Faculty of Engineering, University Freiburg

Albert-Ludwigs-Universität Freiburg

UNI FREIBURG

Structure: Network Part

Network Planning

Network Boot Protocols

Network Booting: Initial Part

- Network booting available for a while
  - Protocols like BOOTP and TFTP pretty old, see the small RFC numbers of them
- Network boot of PC architecture part of the BIOS
- Today: All TCP/IP based focused around protocols like PXE/DHCP/TFTP

Network Booting: Initial Part

- Network boot device different
  - Instead of detecting traditional boot block on a block device (hard drive, optical medium or floppy disk) network adaptor to be initialized
  - Hardware driver and IP / UDP stack loaded
  - DHCP request sent and offers evaluated
  - Special BOOTP/DHCP variables containing next-server for TFTP (and for NFS root) evaluated

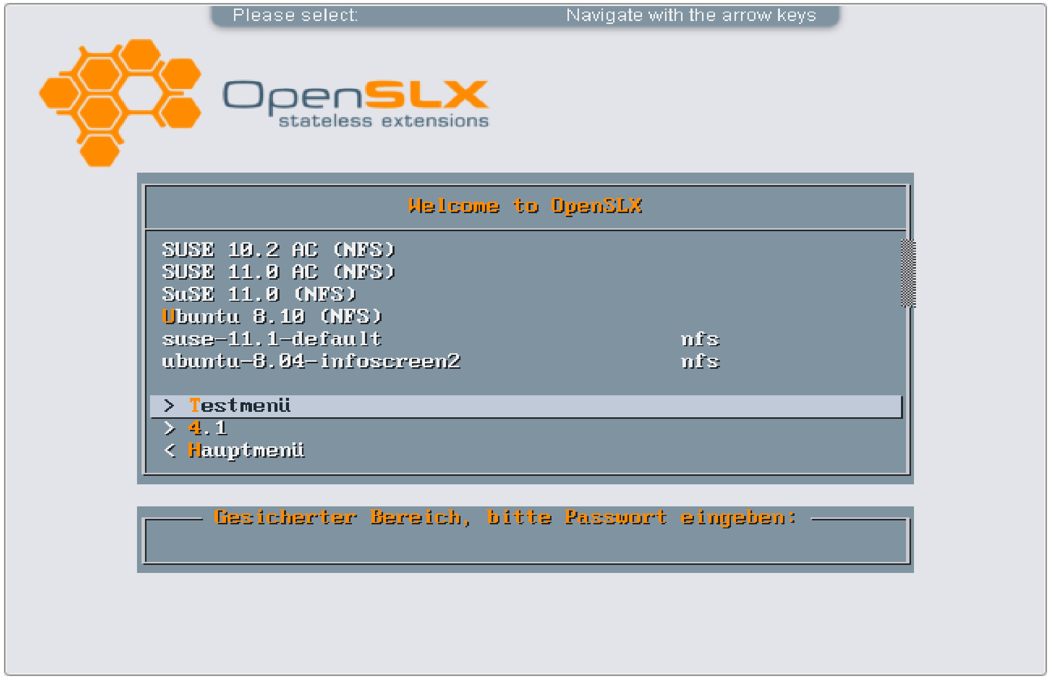
Network Booting: Initial Part

- Typical cluster node or desktop PC offers the capability of PXE booting
- Lots of boot solutions base on PXE
  - RIS for Windows
  - PXE-Linux of the Syslinux suite
  - ...

Flexible Booting

Albert-Ludwigs-Universität Freiburg

- Lots of free and commercial boot products which could even be chained
  - Offering the option of sophisticated boot menus (perfect for flexible test environments)



Welcome to OpenSLX

Gesicherter Bereich, bitte Passwort eingeben:

Structure: Client Filesystem

Client Side Root Filesystem

Readonly Base

Read-Writeable Overlays

Filesystem Challenges

- Filesystems for stateless Linux machines face some challenges
  - One Linux variant/installation to be served to hundreds of different clients
  - All clients “see” same base filesystem
  - Read-only export to avoid any interference and security issues
  - No trivial means to store configuration and run-time system data on local storage (don't personalize nodes!) or on per-client server shares
  - Persistent configuration storage will get complex with rising number of nodes

Filesystem Challenges

- Filesystem for stateless Linux root filesystem – the read-only approach
  - Simplifies matters as e.g. no file locking is required
  - Eases security concerns as modifications are not trivially possible from client side
  - Clients might be made accessible from the Internet, the filesystem server doesn't need to be
  - Approach offers optimizations like using network block devices with special filesystems on-top

Network Filesystem Approaches

- Two general approaches to provide a network based filesystem
- Traditional network filesystems like AFS, NFS, SMB/CIFS
- Linked to the Linux kernel VFS layer
- Common file access implemented in the protocols
- Andrew File System (AFS) implemented, incorporated by IBM, part of the Linux kernel
  - Rather complex, not mainline any more
  - Implements local caching up to 2GByte
  - Comparably slow

Network Filesystem Approaches

- Server Message Block / Common Internet File System
  - SMB, originally invented by IBM end of 1980ies, early 1990ies ontop of NetBIOS protocol
  - Later versions and extensions defined by Microsoft, CIFS solely using TCP/IP
  - Implemented for Linux pretty long
  - Average performance
  - Certain standard file types missing like device nodes or symbolic links
  - Package updates during runtime possible to a certain degree

Traditional Network File System

- Network File System (NFS)
  - Invented, defined by SUN Microsystems in the beginning of 1990ies
  - Made to be root filesystem (all relevant file types and access control mechanisms implemented)
  - Available in fourth version
  - Still prevailing solution for remote root filesystems
  - Okay performance
  - Permanent packet streams generated
  - Root filesystem updateable to a certain degree

Alternative Filesystem Approaches

- Alternative: Cluster filesystems like Lustre or Oracle filesystem
- Distributed approach to span multiple nodes
- Optimized for read-write access across multiple machines
- Often too complex for client root filesystems, used for data provisioning

Alternative: Network Block Device

- Alternatively use block oriented data exchange
  - Server exports block device with (partitioning,) filesystem attached
  - Client imports the block device and mounts the contained filesystem(s) the kernel VFS
- Network Block Devices provide the device layer below filesystems over the net
- Number of different approaches available:
  - iSCSI, ATAoE – putting traditional lower layer hardware protocols onto Ethernet, TCP/IP
  - Number of implementations for Linux present in recent kernels (and for other operating systems)

Linux Network Block Device

- Network Block Device (NBD) present in Linux kernel for more than 10 years
  - Simple implementation using kernel module on client side, providing a file or physical, logical block device as user space process via TCP/IP
  - Read-only and read-write exports (for multiple clients)
  - Read-write creates a block difference file on server side for multiple client access to same block device
  - Good performance in 100 Mbit/s networks, with newer versions in Gigabit too
  - In theory all standard Linux filesystems importable via NBD

Linux Network Block Device

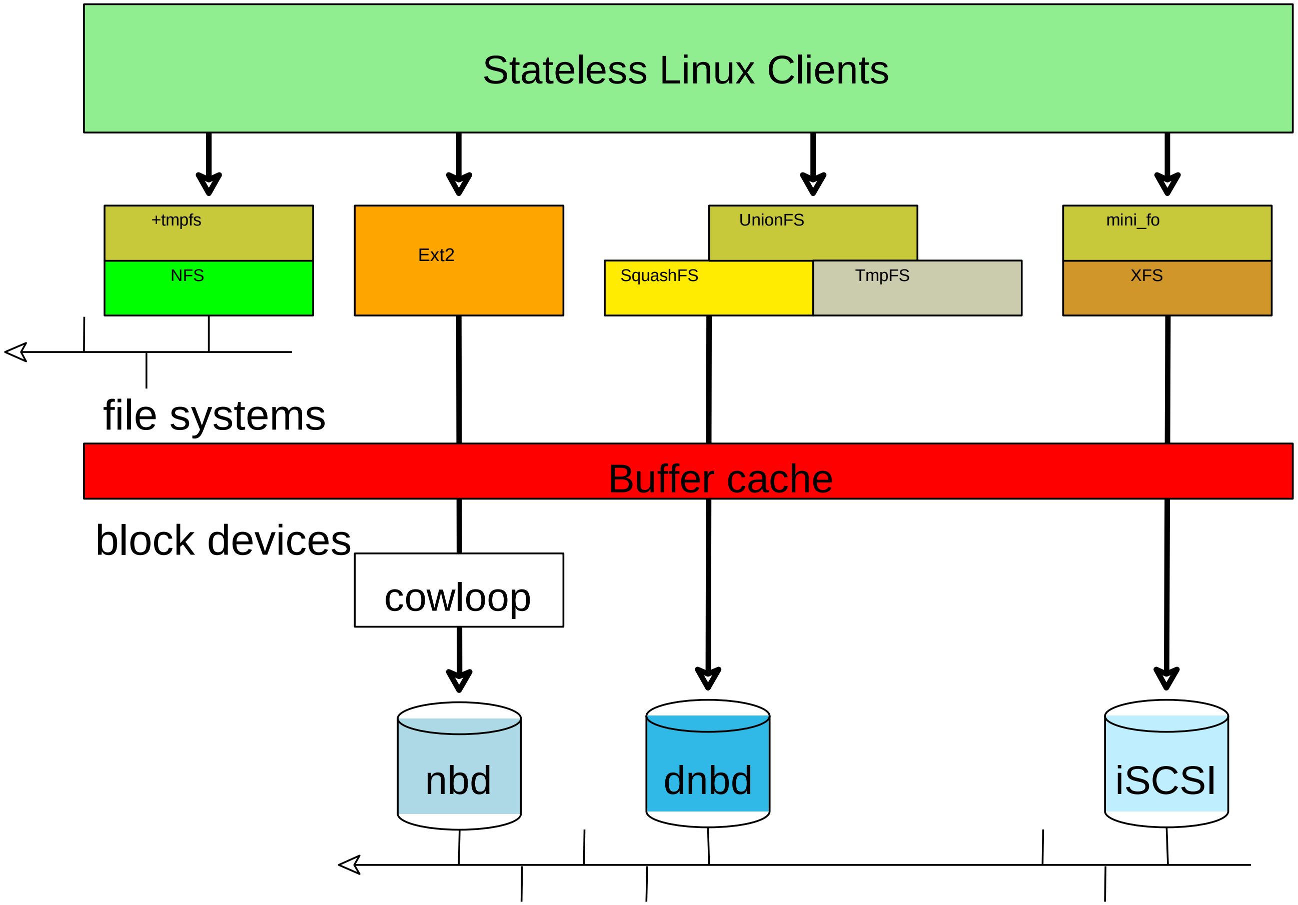
- Using Network Block Devices in netbooting triggered two bachelor theses (2005-7) at our professorship
  - Optimizing NBD for shared media like WLAN (and traditional coax Ethernet)
- Distributed NBD was implementing local client side block caching, UDP based, read-only
  - Using multicast to listen to other client root filesystem block requests
  - Idea: Clients using the same root filesystem on the same block device will request the same data
  - Problem: Not in mainline kernel and not compiling for actual kernels at the moment

Linux Network Block Device

- Next approach: Distributed NBD 2
  - Focusing on fail over
  - Using UDP like the the first DNBD
  - Able to check different servers and attach to the fastest machine, re-checking on a regular base
  - Up to four (with the standard configuration) servers which might fail, switched off during runtime of clients
  - Servers have to provide exactly the same block device content

Comparison

- Different approaches in relation to Linux Kernel virtual filesystem



Add Local Read-Write to Filesystems

- Solutions discussed by now use shared, read-only imports from filesystem, block device server
- For locally generated configuration and run-time data read-writeable parts of root filesystem required
- Two ways: Block wise and file based approaches
- Copy-on-Write-Loop
  - Present in Linux kernel for a while
  - Same concept as used by many virtualization tools
- Translucent/Union filesystems
  - UnionFS / AUFS

Structure: Cloud Monitoring

Client and Server Monitoring

Network Monitoring Tools

Monitoring Challenges

- Traditional approach just to look at machines impossible
  - Compact installations in racks, special systems like CPU blades
  - Sheer number of nodes, KVM not a real solution
  - Different types of hardware
  - Virtual machines out-numbering real hardware
  - Restricted access

Monitoring Challenges

- Challenge to monitor large number of cluster/cloud servers and nodes
  - Generate overviews for administrators
  - Might be used for accounting purposes
  - Different goals: Detect node failures and resource shortages
  - Optimize cloud usage
  - Monitor real and virtual machines
  - Generate different type of short and long term statistics
- Resource planning for optimal usage

Monitoring Approaches

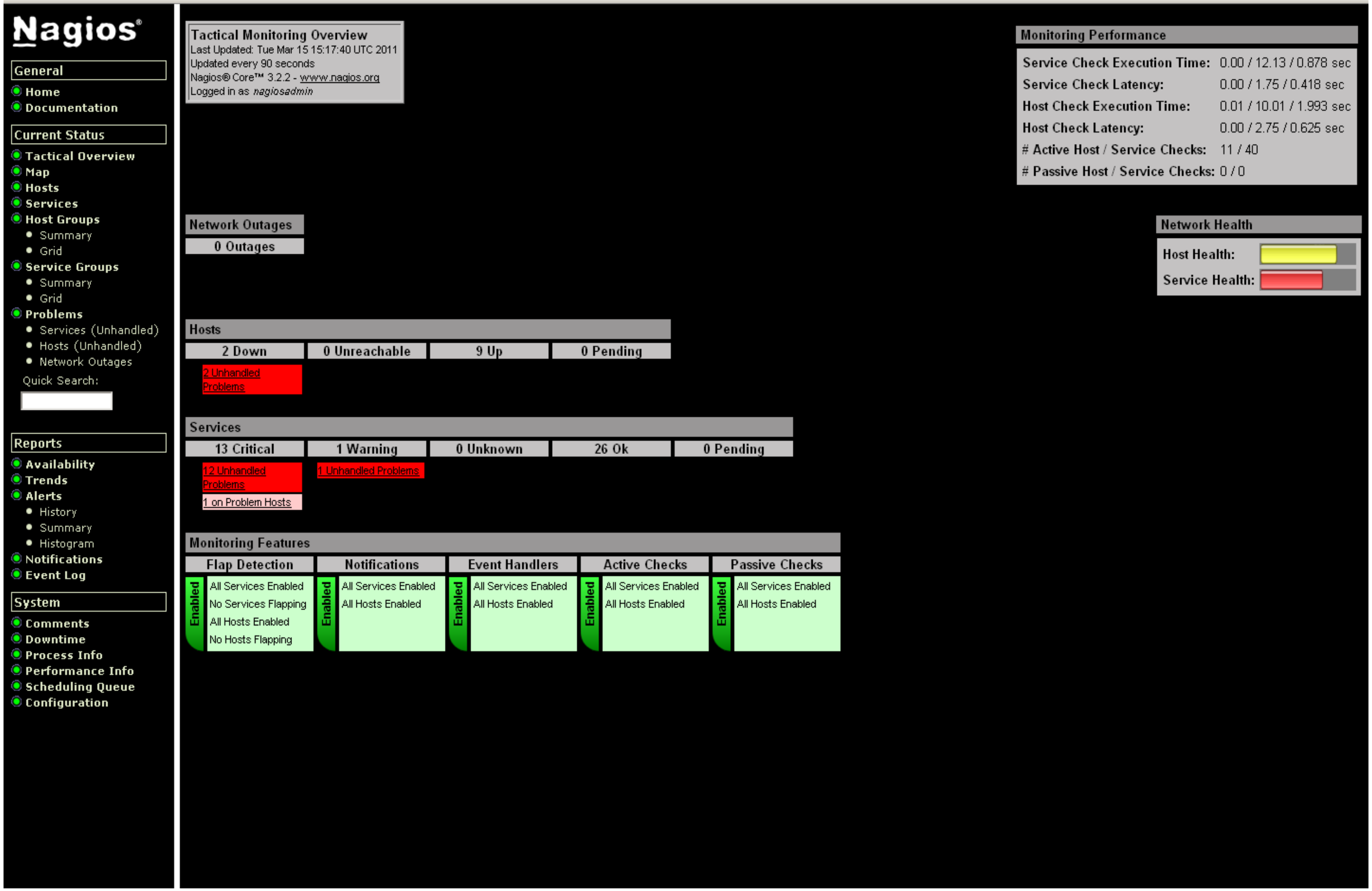
- Different approaches: active / passive
- Passive: Monitor is doing the probes
  - Pinging nodes
  - Trying to request data from services monitored
- Active:
  - Running a small script, daemon or whatever on the monitored targets
  - Deliver data back to the monitoring server/proxy

Tools: Nagios

- One of the first Open Source monitoring frameworks
- One of the oldest tools around, available for over 10 years (first name: Netsaint)
- Short and long term monitoring
- Passive and active monitoring of nodes and services
- Vast range of monitoring applets and remote daemons
- Lots of different views available from very general to very specific level
- Complex alert system on different channels

Tools: Nagios

- Main web browser frontend screen



Tactical Monitoring Overview
Last Updated: Tue Mar 15 15:17:40 UTC 2011
Updated every 90 seconds
Nagios® Core™ 3.2.2 - www.nagios.org
Logged in as *nagiosadmin*

Monitoring Performance

Service Check Execution Time:	0.00 / 12.13 / 0.878 sec
Service Check Latency:	0.00 / 1.75 / 0.418 sec
Host Check Execution Time:	0.01 / 10.01 / 1.993 sec
Host Check Latency:	0.00 / 2.75 / 0.625 sec
# Active Host / Service Checks:	11 / 40
# Passive Host / Service Checks:	0 / 0

Network Outages
0 Outages

Network Health
Host Health:
Service Health:

Hosts

2 Down	0 Unreachable	9 Up	0 Pending
2 Unhandled Problems			

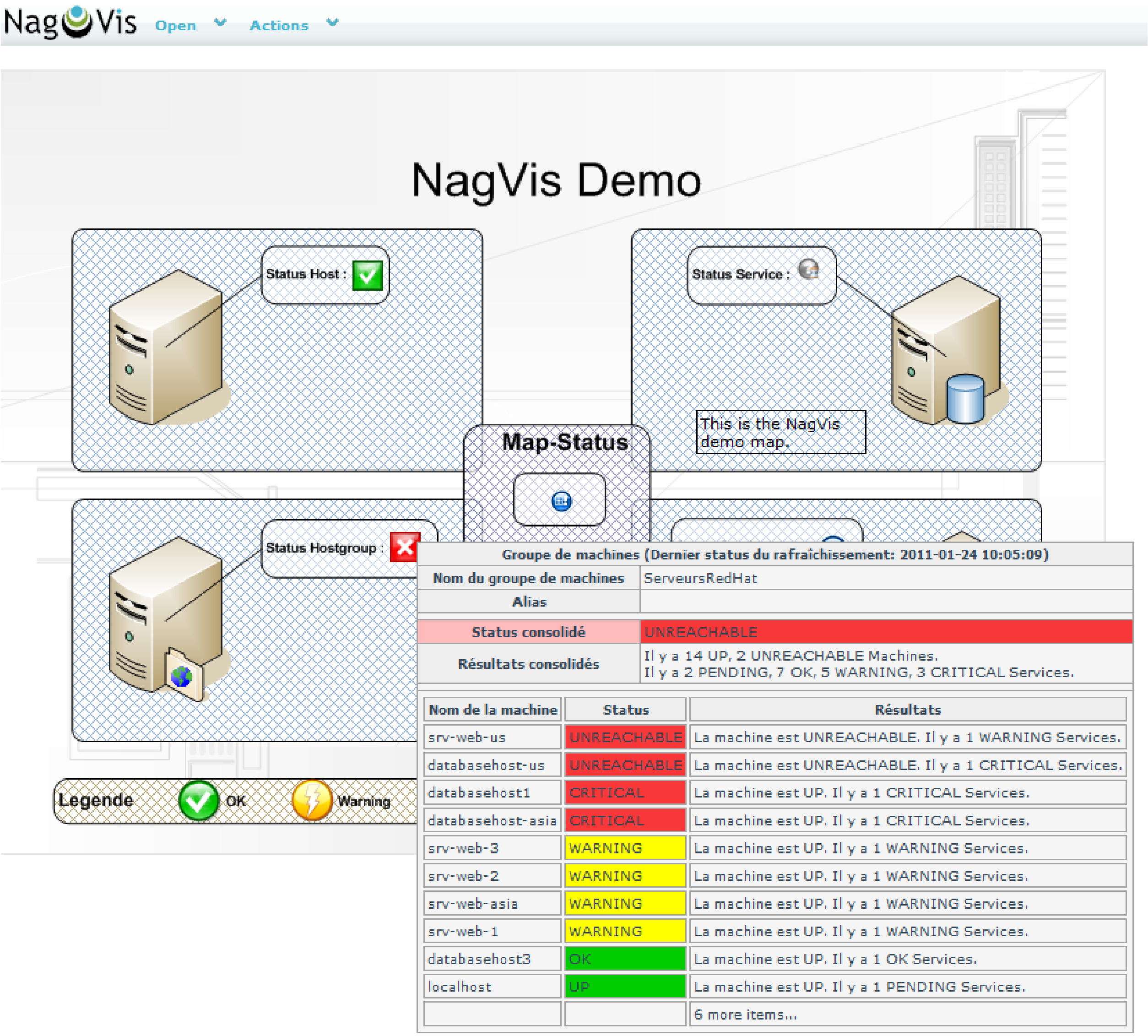
Services

13 Critical	1 Warning	0 Unknown	26 Ok	0 Pending
12 Unhandled Problems 1 on Problem Hosts	1 Unhandled Problems			

Monitoring Features

Flap Detection	Notifications	Event Handlers	Active Checks	Passive Checks
Enabled All Services Enabled No Services Flapping All Hosts Enabled No Hosts Flapping	Enabled All Services Enabled All Hosts Enabled	Enabled All Services Enabled All Hosts Enabled	Enabled All Services Enabled All Hosts Enabled	Enabled All Services Enabled All Hosts Enabled

Tools: Shinken



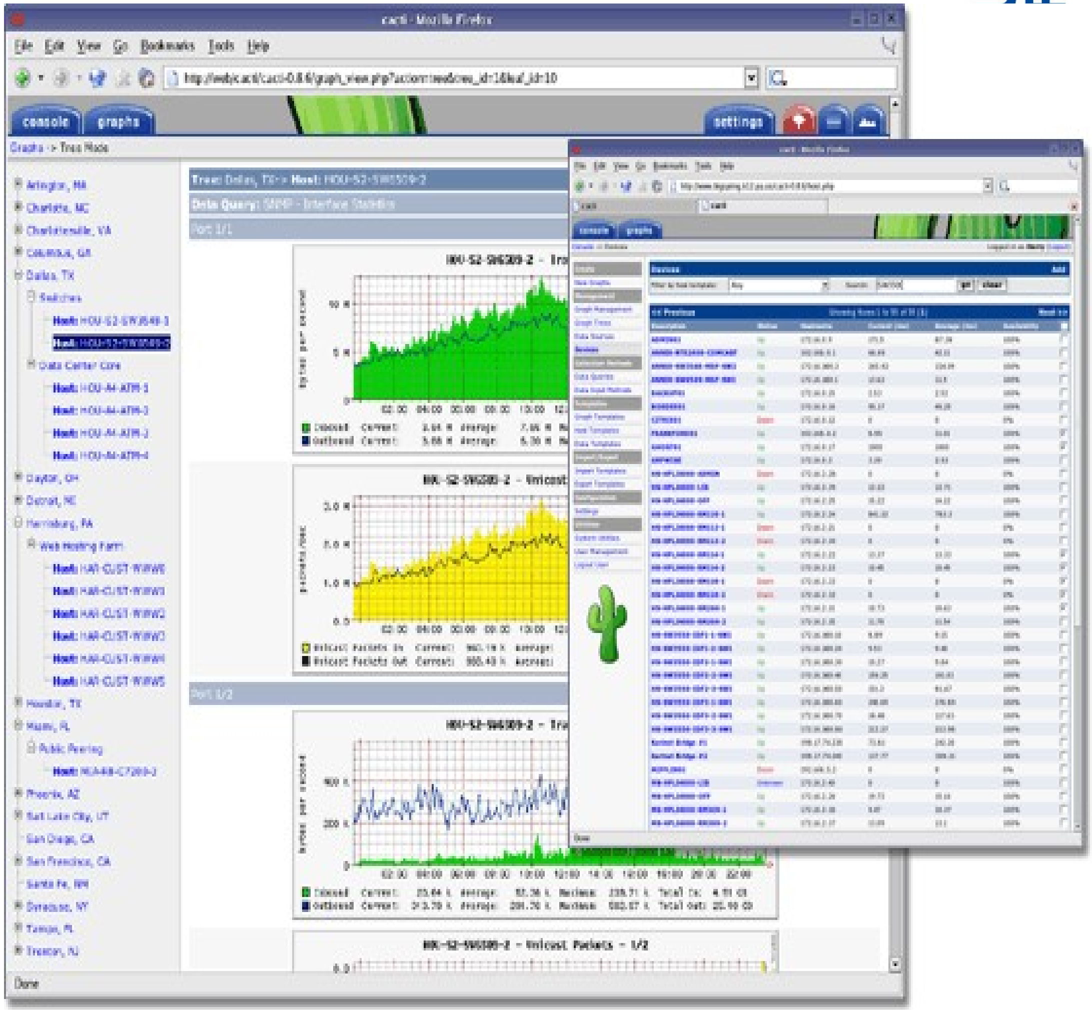
Groupe de machines (Dernier status du rafraîchissement: 2011-01-24 10:05:09)	
Nom du groupe de machines	ServeursRedHat
Alias	
Status consolidé	UNREACHABLE
Résultats consolidés	Il y a 14 UP, 2 UNREACHABLE Machines. Il y a 2 PENDING, 7 OK, 5 WARNING, 3 CRITICAL Services.

Nom de la machine	Status	Résultats
srv-web-us	UNREACHABLE	La machine est UNREACHABLE. Il y a 1 WARNING Services.
databasehost-us	UNREACHABLE	La machine est UNREACHABLE. Il y a 1 CRITICAL Services.
databasehost1	CRITICAL	La machine est UP. Il y a 1 CRITICAL Services.
databasehost-asia	CRITICAL	La machine est UP. Il y a 1 CRITICAL Services.
srv-web-3	WARNING	La machine est UP. Il y a 1 WARNING Services.
srv-web-2	WARNING	La machine est UP. Il y a 1 WARNING Services.
srv-web-asia	WARNING	La machine est UP. Il y a 1 WARNING Services.
srv-web-1	WARNING	La machine est UP. Il y a 1 WARNING Services.
databasehost3	OK	La machine est UP. Il y a 1 OK Services.
localhost	UP	La machine est UP. Il y a 1 PENDING Services.
		6 more items...

- Another monitoring framework, see www.shinken-monitoring.org
- Pretty much Nagios oriented regarding functionality
- More modern user frontend

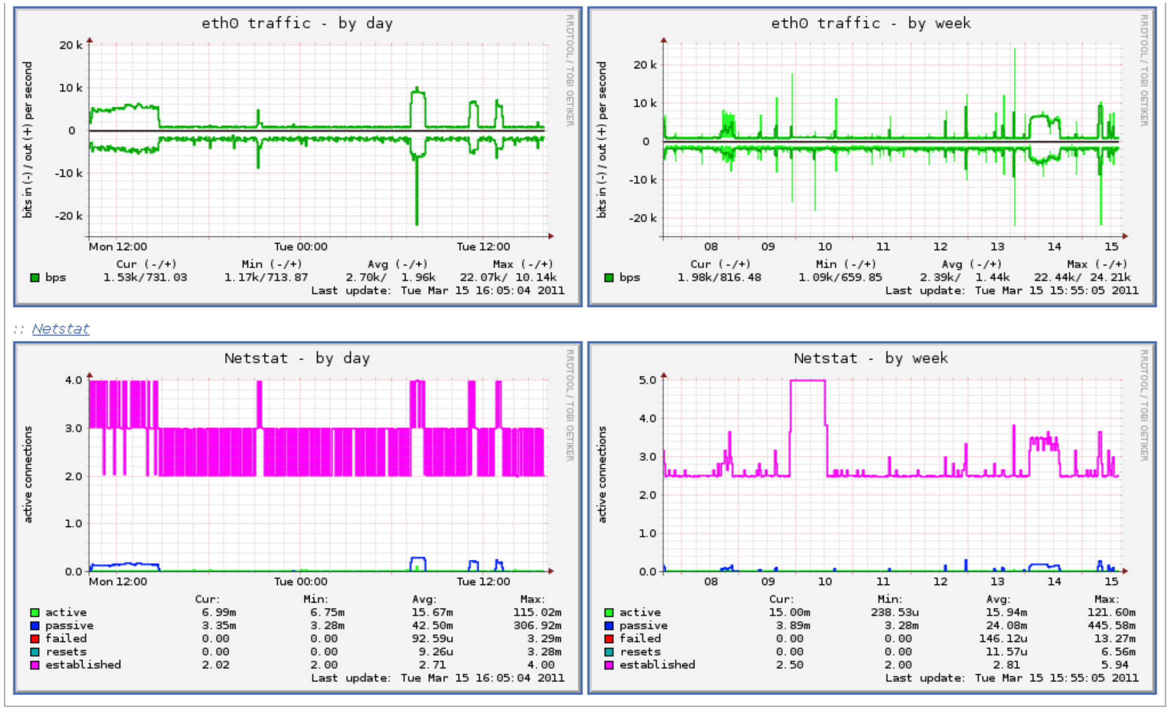
Tools: Cacti

- Inspiration taken from Nagios too, see homepage www.cacti.net
- Using RRD and MySQL as data backends
- Complex long term graphic analysis possible



Tools: Munin

- Open Source, light weight monitoring framework with less features than big counterparts
- Web application just for data presentation
- No ability to analyze syslog data
- No grouping of server, node classes
- No service, node autodiscovery
- Using RRD backend
- IPv 6 capable



```
             Cur (-/+)        Min (-/+)       Avg (-/+)        Max (-/+)
■ bps     1.53k/731.03     1.17k/713.87     2.70k/  1.96k    22.07k/ 10.14k
                                Last update:  Tue Mar 15 16:05:04 2011
```

```
             Cur (-/+)        Min (-/+)       Avg (-/+)        Max (-/+)
■ bps     1.98k/816.48     1.09k/659.85     2.39k/  1.44k    22.44k/ 24.21k
                                Last update:  Tue Mar 15 15:55:05 2011
```

:: Netstat

```
                      Cur:        Min:        Avg:        Max:
■ active             6.99m       6.75m      15.67m     115.02m
■ passive            3.35m       3.28m      42.50m     306.92m
■ failed             0.00        0.00       92.59u       3.29m
■ resets             0.00        0.00        9.26u       3.28m
■ established        2.02        2.00        2.71        4.00
                                Last update:  Tue Mar 15 16:05:04 2011
```

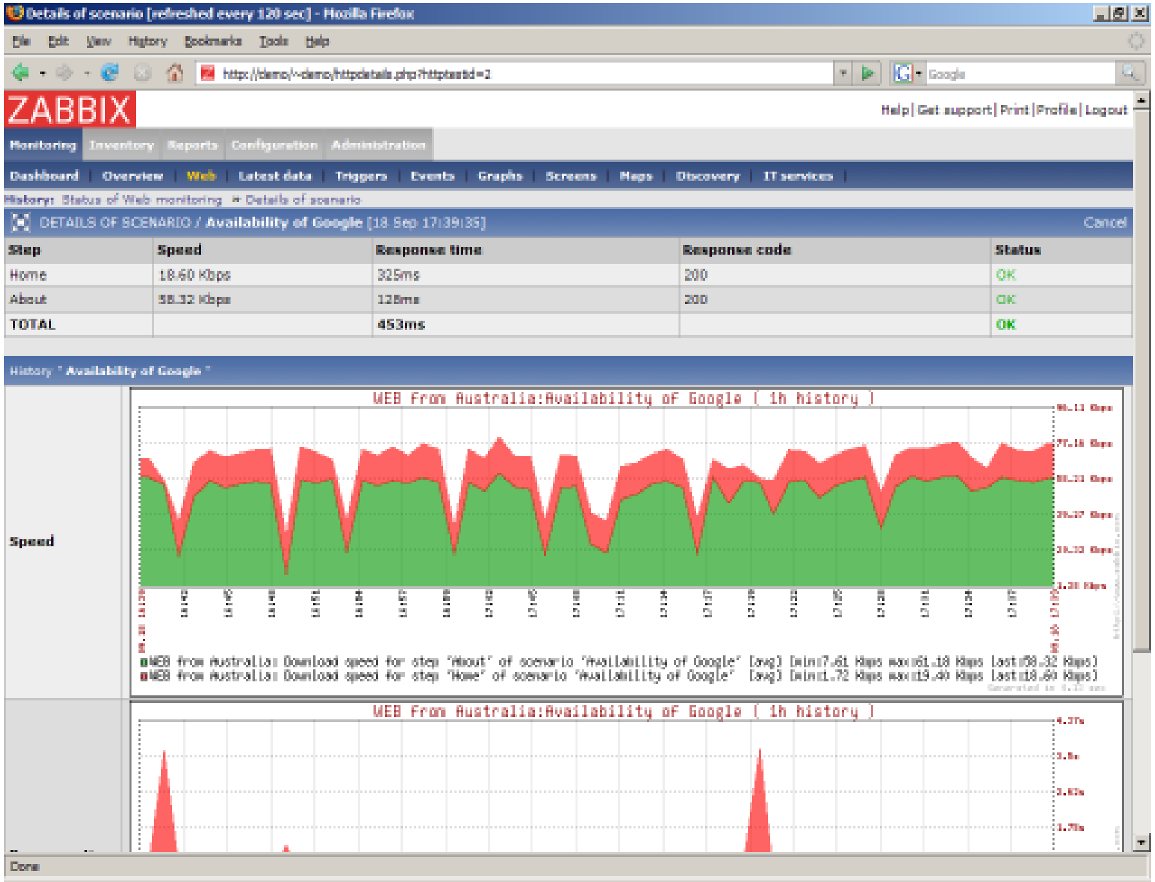
```
                      Cur:        Min:        Avg:        Max:
■ active            15.00m     238.53u      15.94m     121.60m
■ passive            3.89m       3.28m      24.08m     445.58m
■ failed             0.00        0.00      146.12u      13.27m
■ resets             0.00        0.00       11.57u       6.56m
■ established        2.50        2.00        2.81        5.94
                                Last update:  Tue Mar 15 15:55:05 2011
```

Tools: OpenNMS

- Another powerful monitoring framework
- Configurable via web interface
- Could analyze syslog data
- Specialized agents to run certain tests (remotely)
- Autodiscovery
- Jrobin and PostgreSQL
- IPv6 ready to a certain degree

Tools: Zabbix

- Just another rather powerful framework supporting lots of SQL data store backends, www.zabbix.com


Details of scenario [refreshed every 120 sec] - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://demo/~demo/httpdetails.php?httptestid=2
Google
ZABBIX
Help | Get support | Print | Profile | Logout
Monitoring Inventory Reports Configuration Administration
Dashboard | Overview | Web | Latest data | Triggers | Events | Graphs | Screens | Maps | Discovery | IT services |
History: Status of Web monitoring » Details of scenario
Cancel
History " Availability of Google "
Speed
WEB From Australia:Availability of Google (1h history)
WEB From Australia:Availability of Google (1h history)
Done

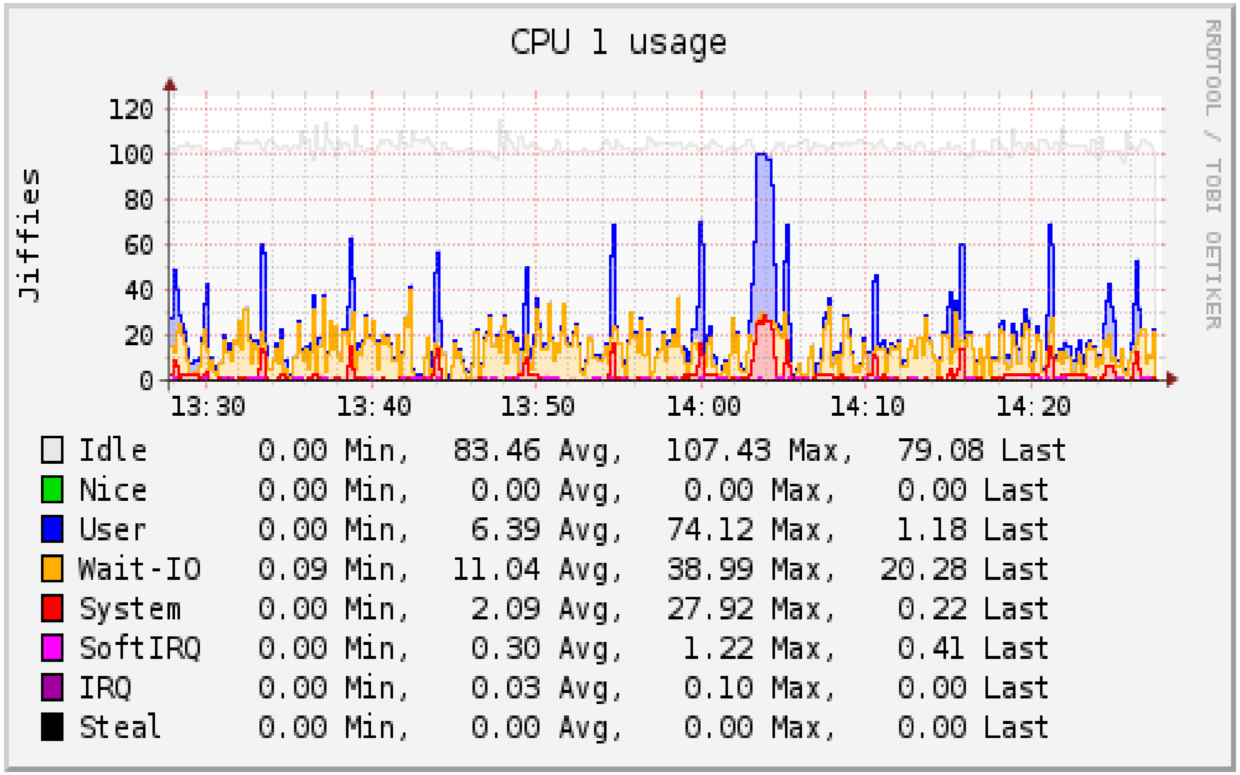
DETAILS OF SCENARIO / Availability of Google [18 Sep 17:39:35]

Step	Speed	Response time	Response code	Status
Home	18.60 Kbps	325ms	200	OK
About	58.32 Kbps	128ms	200	OK
TOTAL		453ms		OK

🟩 WEB from Australia: Download speed for step 'About' of scenario 'Availability of Google' [avg] [min:7.61 Kbps max:61.18 Kbps last:58.32 Kbps]
🟥 WEB from Australia: Download speed for step 'Home' of scenario 'Availability of Google' [avg] [min:1.72 Kbps max:19.40 Kbps last:18.60 Kbps]

Tools: CollectD

- Interesting for larger setups as distributed autodiscovery (not just IP ranges)
- RRD data backend, for more: collected.org



```
☐ Idle       0.00 Min,   83.46 Avg,  107.43 Max,   79.08 Last
🟩 Nice       0.00 Min,    0.00 Avg,    0.00 Max,    0.00 Last
🟦 User       0.00 Min,    6.39 Avg,   74.12 Max,    1.18 Last
🟧 Wait-IO    0.09 Min,   11.04 Avg,   38.99 Max,   20.28 Last
🟥 System     0.00 Min,    2.09 Avg,   27.92 Max,    0.22 Last
🟪 SoftIRQ    0.00 Min,    0.30 Avg,    1.22 Max,    0.41 Last
🟪 IRQ        0.00 Min,    0.03 Avg,    0.10 Max,    0.00 Last
⬛ Steal      0.00 Min,    0.00 Avg,    0.00 Max,    0.00 Last
```

Outlook

- Next lecture, Thursday, same time and venue
- Talking of system virtualization