

GSI monitoring amalgamation

Christopher Huhn

GSI Helmholtzzentrum für Schwerionenforschung

C.Huhn@gsi.de

Hepix Bologna

19-04-2013


GSI monitoring showroom

USV values - Iceweasel

File Edit View History Bookmarks Tools Help

www-linux-old.gsi.de/cgi-bin/get_gd09_daten_rz1.pl

Most Visited Search GSI Add2es-f Snipe heise online SPIEGEL ONLINE - S... Fefes Blog Radroutenplaner Hes...

 **RZ-1 ---> USV Values gd09**

Automatic Refresh-Time 30 seconds ! Current Time: Fri Apr 12 13:12:09 2013

Okay
Exceed Warn Level
Exceed Alarm Level
Overload - Shutdown Alarm

USV Host	Output Load L1	Output Load L2	Output Load L3	Output Power L1	Output Power L2	Output Power L3	Batt. Temp	Batt. Hold	Batt. Cap.	Output Volt L1	Output Volt L2	Output Volt L3	Output Current L1	Output Current L2	Output Current L3	Input Volt L1	Input Volt L2	Input Volt L3
	%	%	%	KW	KW	KW	°C	min	%	V	V	V	A	A	A	V	V	V
gd09net1	71	78	79	7.6	8.4	8.5	25	16	100	229	228	229	33.2	36.8	37.1	247	247	249
gd09net2	71	77	77	7.6	8.3	8.3	25	16	100	230	232	231	33.0	35.8	35.9	244	244	245
gd09net3	71	78	80	7.6	8.4	8.6	25	16	100	231	231	232	32.9	36.4	37.1	245	246	247
gd09net4	71	77	79	7.7	8.3	8.5	25	16	100	232	232	232	33.2	35.8	36.6	245	244	244
gd09net5	70	79	75	7.6	8.6	8	25	16	100	229	230	231	33.2	37.4	34.6	243	244	246
gd09net6	71	77	79	7.8	8.3	8.5	25	16	100	232	232	232	33.6	35.8	36.6	246	245	246
Gesamtleistung		75			146.6								199.1	217.9	218.0			

SNMP Script **get_gd09_daten_rz1.pl**, Version 1.6.5 last modified: 24. June 2010 H. Kreiser

Mail Address: H.Kreiser (at) gsi.de

Connecting to lxmon9.gsi.de...

auto-refreshing

☒ auto-refresh

GSI Minicube Monitoring

Help

Overview

Ground Floor

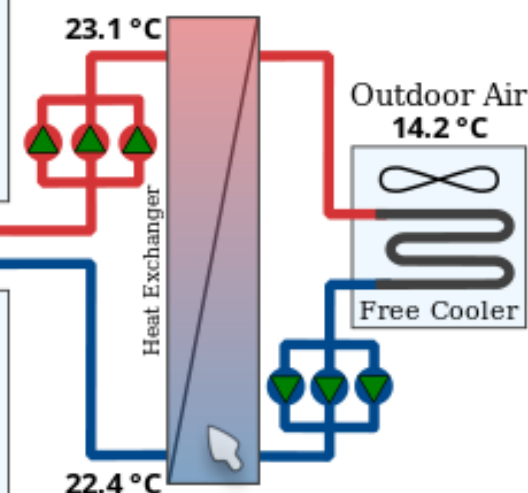
First Floor

First Floor

	Power Outlets	1704 <i>on</i>	24 <i>off</i>	0.99 <i>% on</i>	Cooled Air	Hot Air
Overall	RMS Current (in A)	26.9 <i>max</i>	7.5 <i>avg</i>	0.1 <i>min</i>	27.3 °C <i>max</i>	49.5 °C <i>max</i>
Room Air	Active Power (in W)	15468 <i>max</i>	4135 <i>avg</i>	8 <i>min</i>	24.0 °C <i>avg</i>	34.9 °C <i>avg</i>
Humidity	Power Factor	0.99 <i>max</i>	0.90 <i>avg</i>	0.25 <i>min</i>	22.7 °C <i>min</i>	22.9 °C <i>min</i>
29 %					27 / 31 / 34 % <i>min / avg / max</i>	6 / 17 / 32 % <i>min / avg / max</i>

Ground Floor

	Power Outlets	0 <i>on</i>	1728 <i>off</i>	0.00 <i>% on</i>	Cooled Air	Hot Air
Overall	RMS Current (in A)	0.0 <i>max</i>	0.0 <i>avg</i>	0.0 <i>min</i>	23.3 °C <i>max</i>	24.4 °C <i>max</i>
Room Air	Active Power (in W)	4 <i>max</i>	4 <i>avg</i>	4 <i>min</i>	22.7 °C <i>avg</i>	23.1 °C <i>avg</i>
Humidity	Power Factor	0.33 <i>max</i>	0.32 <i>avg</i>	0.31 <i>min</i>	21.9 °C <i>min</i>	21.8 °C <i>min</i>
35 %					32 / 33 / 35 % <i>min / avg / max</i>	30 / 32 / 34 % <i>min / avg / max</i>



Update requested: 12.4.2013 13:54:06

Time of data: 12.4.2013 13:54:02

Age of data

(dd:HH:MM:SS): 0:00:00:04

Frankfurt Institute for Advanced Studies (FIAS) and GSI Helmholtzzentrum für Schwerionenforschung, 2012 – [Torbjörn Klatt](#). Based on work by Jan de Cuveland and Matthias Bach (FIAS 2011)

Version 0.1.0 Documentation

SGE Farm Monitoring

Queue Status - Icarus

QUEUE	default	RUN	873	RES	0	AVAIL	759	TOTAL	1648
QUEUE	grid	RUN	0	RES	0	AVAIL	1648	TOTAL	1648
QUEUE	highmem	RUN	1	RES	0	AVAIL	407	TOTAL	412
QUEUE	long	RUN	408	RES	0	AVAIL	4	TOTAL	412
QUEUE	short	RUN	9	RES	0	AVAIL	1639	TOTAL	1648

Queue Status - Prometheus

QUEUE	default	RUN	834	RES	0	AVAIL	8100	TOTAL	9144
QUEUE	grid	RUN	0	RES	0	AVAIL	1860	TOTAL	1905
QUEUE	highmem	RUN	494	RES	0	AVAIL	2484	TOTAL	3048
QUEUE	long	RUN	1	RES	0	AVAIL	2975	TOTAL	3048
QUEUE	medium	RUN	3389	RES	0	AVAIL	4091	TOTAL	7620
QUEUE	short	RUN	0	RES	0	AVAIL	8928	TOTAL	9144



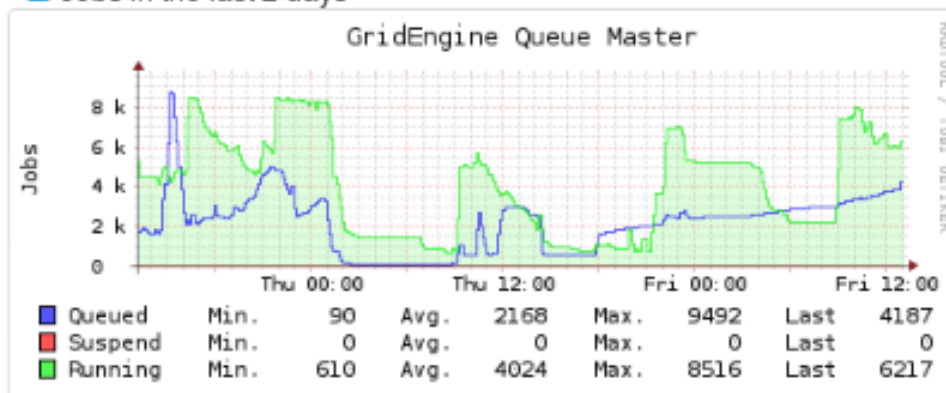
```
~ : lxfsu-gdf.sh
Datei Bearbeiten Ansicht Lesezeichen Einstellungen Hilfe
christo@lxdv42:~$ /usr/local/sbin/lxfsu-gdf.sh 13-04-16 16:33
/u01_1: #####----- (84%)
/u01_2: #####----- (89%)
/u02_1: #####----- (77%)
/u02_2: #####----- (86%)
/u03_1: #####----- (71%)
/u03_2: #####----- (74%)
/u04_1: #####----- (88%)
/u04_2: #####----- (62%)
/u05_1: #####----- (73%)
/u05_2: #####----- (68%)
/u06_1: #####----- (87%)
/u06_2: #####----- (56%)
/u07_1: #####----- (76%)
/u07_2: #####----- (48%)
/u08_1: #####----- (63%)
/u08_2: #####----- (72%)
/u09_1: #####----- (71%)
/u09_2: #####----- (71%)
/u10_1: #####----- (67%)
/u10_2: #####----- (73%)
christo@lxdv42:~$ 13-04-16 16:33
```

...	3627	579	397	3230	0	0	397
...	2	0	0	2	0	0	0
...	402	401	400	2	0	0	400
...	22	0	22	0	0	0	88

Victor Penso: <https://github.com/vpenso/collectd-interface>

...	1184	790	400	784	0	0	400
...	505	422	334	111	0	0	334
...	655	404	399	256	0	0	399
...	2	0	0	2	0	0	0
...	1	0	0	1	0	0	0
...	71	0	0	71	0	71	0
...	10	0	0	10	0	0	0
...	4	1	1	3	0	0	1
...	25	0	0	25	0	25	0
...	2	0	0	2	0	0	0
...	3746	3599	3599	147	0	0	3599
Σ	11065	6568	6374	4691	0	96	6539

[Jobs in the last 2 days](#)



Some quotes

- „That's my private and top secret monitoring script!”
- „In fact, without any email mechanism, the current state might be healthier: if [the service] crashes [...], the affected boxes might soon be in such a bad shape that at least the users will alarm us...”
- „I'm ashamed of my poor coding capabilities therefore I keep it to myself.”

Taming the beast

- Problems
 - Many isolated special purpose monitoring approaches
 - Alerting by looking at web pages
 - Alerting by user complaints
 - Shameful
 - Alerting by Email
 - Overloads admins as well as mail servers
 - Normally no time series collected by these solutions
- Goal: Centralize monitoring of the whole IT infrastructure
 - Oops, IT is not that structured and centralized after all
 - No formal service catalogue
 - And of course no CMDB
 - Separately maintained monitoring configuration outdated most of the time

The genius masters the chaos

- Centralized monitoring for everything effort already started in 2005 (and failed)
- Design goals:
 - Focus on anomaly detection and alerting:
 - Red light == red alert
 - No failed checks to be considered ignorable
 - One incident == one alert – prevent email floods
 - Follow general best practices
 - Integrate with other central IT management systems esp. Trouble Ticket System
 - Stay modular and generic
 - Built what is required for the job and nothing more
 - Don't reinvent the wheel
- Slowly evolved as a „hobby project“ by our former head of IT Mathias Münch
 - cf. talk at Open Source Monitoring Conference 2010:
<http://www.netways.de/index.php?id=2814>

Nagios / Icinga

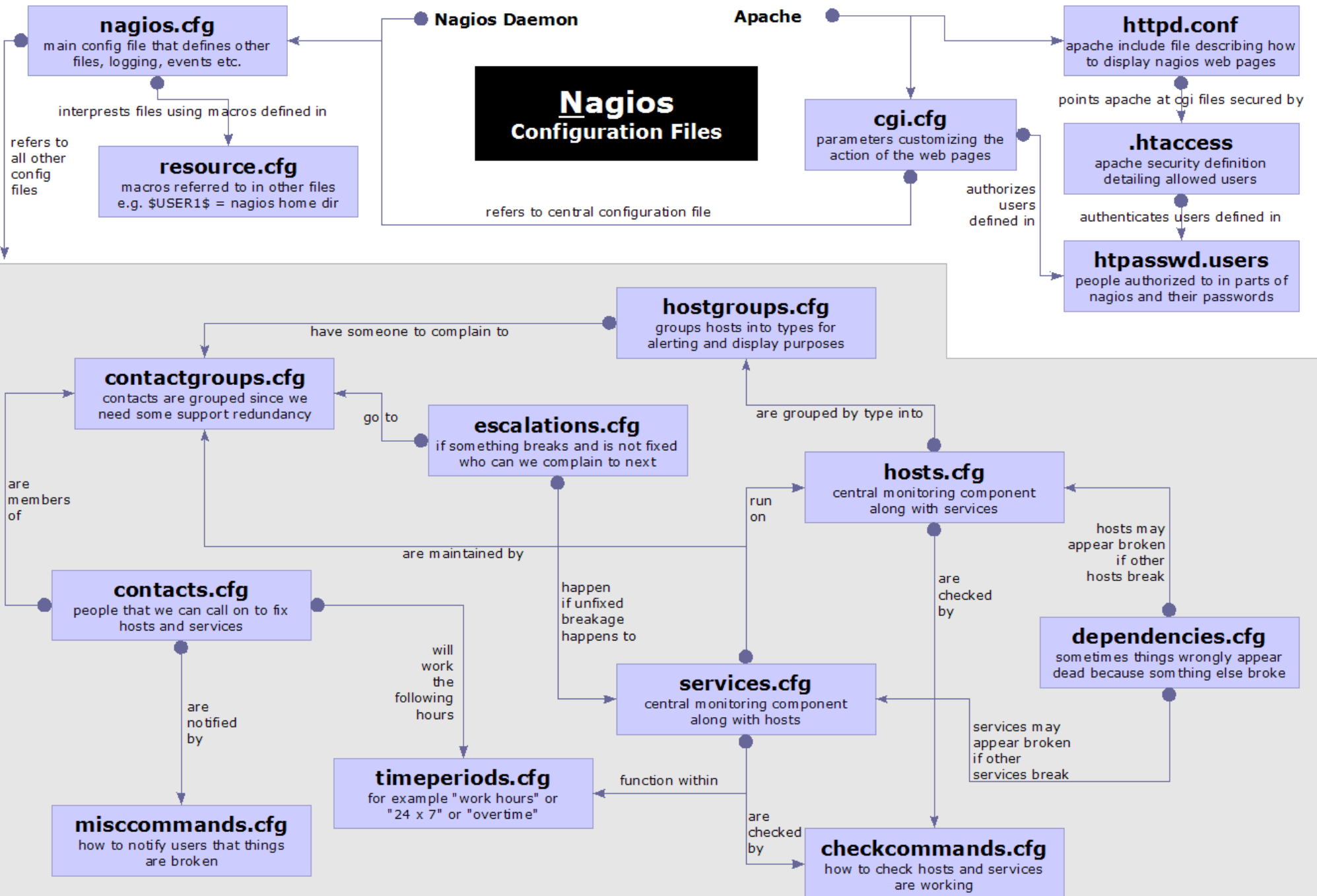
- „Industry standard“
- Hepix presentation by Alf Wachsmann at Karlsruhe 2005
- Nagios configuration is complex and has to be consistent:
Don't maintain it manually – generate it!
- No-red-lights-to-be-ignored-Problem:
 - How to identify the **really relevant** production services?
- Obstacle: No suitable CMDB/Inventory/Service catalogue/Configuration Management Systems available at project start.
 - Are they now?

Nagios / Icinga

- „Industry standard“
- Hepix presentation by Alf Wachsmann at Karlsruhe 2005
- Nagios configuration is complex and has to be consistent:
Don't maintain it manually – generate it!
- No-red-lights-to-be-ignored-Problem:
 - How to identify the **really relevant** production services?
- Obstacle: No suitable CMDB/Inventory/Service catalogue/Configuration Management Systems available at project start.
 - Are they now?

Nagios config generator

- „*Making things complicated is easy, making things easy is hard*“
- Gather the information from the **live service configurations**
- **Servicegroups** are the linchpin of configuration templates
 - *make-orchestrated multi-pass generator:*
 - 2nd pass generator scripts can work on servicegroups defined in the 1st pass etc.
 - Used to add fundamental system health monitors
- Complemented with service templates
- Service, -extinfo, -escalation, host and hostgroup objects are auto-generated
- Contact groups derived from Servicegroup nomenclature
 - e.g. **Linux**.lustre, **Network**.layer2, ...
- *Convention over configuration*



Config generator example

```
#!/bin/sh
#
# define servicegroups for all existing lxi pools by
# quering all possibilities
#
# $Id$
#

. pass0.d/Common/functions.sh

for BITNESS in 32 64; do
    for FLAVOR in lenny squeeze wheezy jessie sid; do
        for CLUSTER in lx${FLAVOR}${BITNESS} ${FLAVOR}lust${BITNESS}; do
            if double_lookup $CLUSTER.gsi.de \
                | servicegroup_for_service Linux.$CLUSTER 'sanity'; then
                # notify TTS about outages of lxi's:
                service_escalation Linux.$CLUSTER sanity 24x7
            fi
        done
    done
done
```


Config generator example

File Edit View History Bookmarks Tools Help

netdisco... /Linux/Monit... Icinga Current N... /Linux/pro_h... OpenNetAd... Home Graphs for /...

Lustre cluster lustig (Linux.lustig)

Host	Status	Services	Actions
lxfs237.gsi.de	UP	1 OK	
lxfs238.gsi.de	UP	1 OK	
lxfs239.gsi.de	UP	1 OK	
lxmids11.gsi.de	UP	1 OK	

Linux.lxlenny32 (Linux.lxlenny32)

Host	Status	Services	Actions
lxi034.gsi.de	UP	1 OK	
lxi038.gsi.de	UP	1 OK	
lxi042.gsi.de	UP	1 OK	
lxi045.gsi.de	UP	1 OK	

Linux.lxlenny64 (Linux.lxlenny64)

Host	Status	Services	Actions
lxi035.gsi.de	UP	1 OK	
lxi039.gsi.de	UP	1 OK	
lxi043.gsi.de	UP	1 OK	
lxi047.gsi.de	UP	1 OK	

Linux.lxsqueeze64 (Linux.lxsqueeze64)

Host	Status	Services	Actions
lxi061.gsi.de	DOWN	1 OK	
lxi062.gsi.de	DOWN	1 OK	

Linux.pro_hpc (Linux.pro_hpc)

Host	Status	Services	Actions
lxsub05.gsi.de	UP	1 OK	
lxsub06.gsi.de	UP	1 OK	
lxsub07.gsi.de	UP	1 OK	
lxsub08.gsi.de	UP	1 OK	
lxsub10.gsi.de	UP	1 OK	
lxsub11.gsi.de	UP	1 OK	
lxsub12.gsi.de	UP	1 OK	
lxsub13.gsi.de	UP	1 OK	
lxsub14.gsi.de	UP	1 OK	

Linux.squeezelust64 (Linux.squeezelust64)

Host	Status	Services	Actions
lxi052.gsi.de	UP	1 OK	
lxi054.gsi.de	UP	1 OK	
lxi055.gsi.de	UP	1 OK	
lxi056.gsi.de	UP	1 OK	
lxi057.gsi.de	UP	1 OK	
lxi058.gsi.de	UP	1 OK	
lxi059.gsi.de	UP	1 OK	
lxi060.gsi.de	UP	1 OK	

WLAN access points (Network.Accesspoints)

Host	Status	Services	Actions
ah-ap-066.nm.gsi.de	UP	1 OK	

Layer2 Network Switches (Network.Layer2)

Host	Status	Services	Actions
gicbcs02.gsi.de	UP	1 OK	
gicbcs03.gsi.de	UP	1 OK	

Network (Network.Layer3)

Host	Status	Services	Actions
10.0.3.1	UP	1 OK	
10.0.4.1	DOWN	1 UNKNOWN	
10.10.16.1	DOWN	1 UNKNOWN	
10.10.20.1	UP	1 OK	

www-oracle.gsi.de/pls/gsi/G_QUERY_V2.geraeteinfo?p_geraete_name=lxsub11.gsi.de

Interfacing with OTRS

Config Einstellungen: SystemMonitoring -> Core::PostMaster

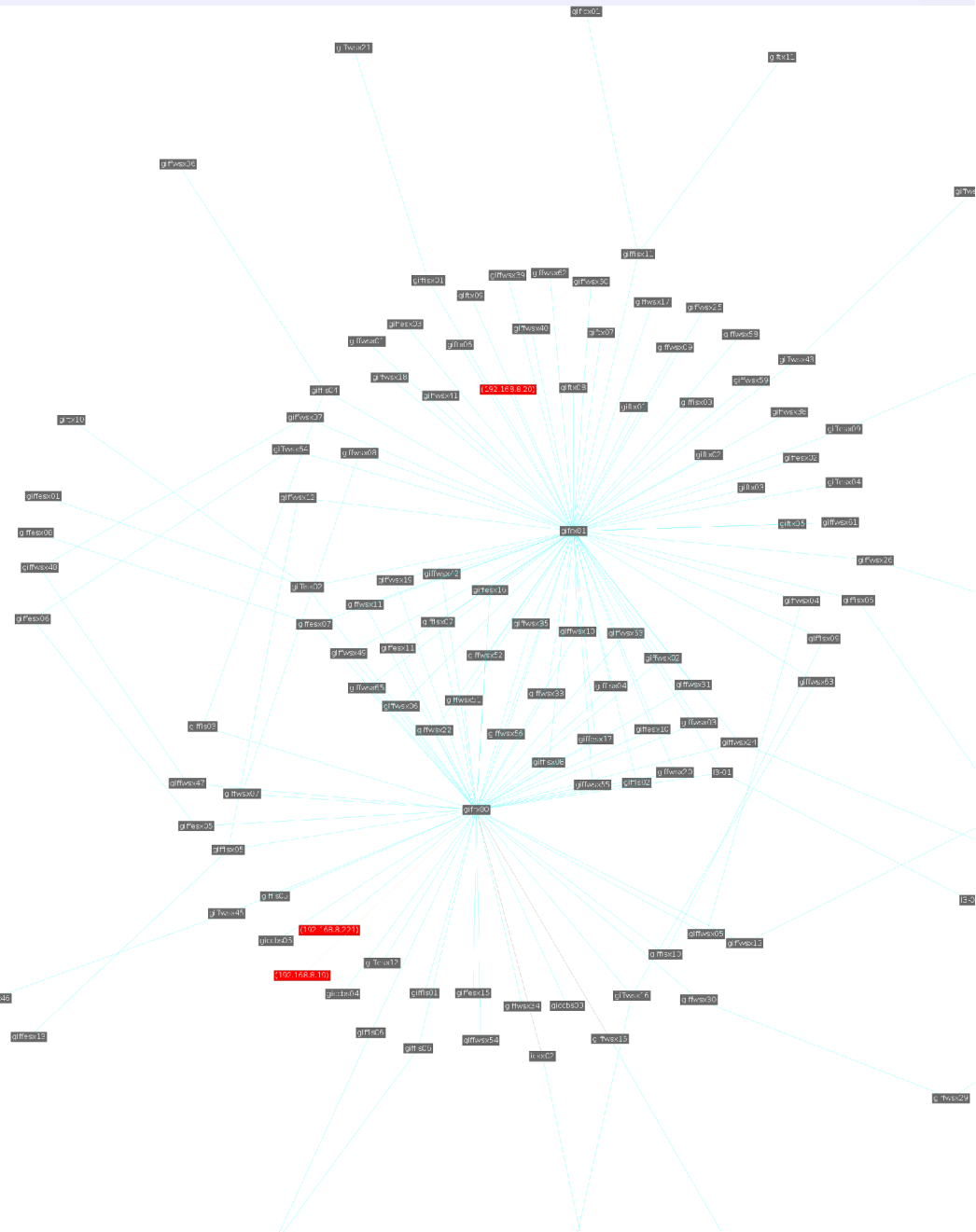
☒ **PostMaster::PreFilterModule###1-SystemMonitoring:**
Einfache Email Schnittstelle zu System Monitoring Suites

- Nagios alerts → OTRS gateway
 - Via service escalations
 - Email notifications sent to the proper Queue addresses for all state changes
 - Ticket opening and closing handled by OTRS filters
- Nagios contacts and contactgroups to be derived from OTRS agents and queue settings
- No back-channel for acknowledgements yet

Schlüssel	Inhalt	
ArticleType	note-internal	Löschen
CloseActionState	closed successful	Löschen
ClosePendingTime	172800	Löschen
CloseTicketRegExp	OK UP	Löschen
DefaultService	Host	Löschen
FreeTextHost	1	Löschen
FreeTextService	2	Löschen
FromAddressRegExp	racle nagios@gsi.de	Löschen
HostRegExp	d+\\.\\d+\\.\\d+\\.\\d+\\s*	Löschen
Module	Kernel::System::Post	Löschen
NewTicketRegExp	CRITICAL DOWN WARN	Löschen
SenderType	system	Löschen
ServiceRegExp	\\s*Service:\\s+(\\.*)\\s*	Löschen
StateRegExp	\\s*State:\\s+(\\S+)	Löschen
Neu		

Network topology: Netdisco

- Introduced by Wolfgang Friebe at last Hepix in Beijing
- Discovers the layer2 network topology via SNMP, CDP etc.
- Data source for network device servicegroups
 - And host dependency information
 - Only for network devices for now
- Data grabbed by direct queries to PostgreSQL DB of Netdisco



Wait, SNMP?

- **Simple?** Network Management Protocol
- Methuselah: Defined **1988** (RFC 1067)
- SNMP natively supported by:
 - Network devices
 - including Infiniband switches. WiFi Aps, ...
 - Infrastructure devices: UPSs, PDUs, temp sensors
 - Printers
 - Linux (net-snmp)
 - Windows

Extending net-snmp

- Pass/pass persist calls an arbitrary program
 - Pass <OID> /path/to/program <params> ...
- AgentX talks to an independent sub-agent
- Existing netsnmp subagents: Lustre, LSI Megaraid, Libvirt, LSF, ...
 - Custom developments at GSI: Cfengine “last seen”, 3ware RAID health check, Postfix mailqueues, ...

Security is Not My Problem

- Version 2c:
 - shared secret „community string“ transferred in plain text
 - Simple Network **Management?** Protocol
 - Disable write-access
 - IP-based access restrictions via *tcpwrappers* + Firewalls
- Version 3:
 - (Too)? complex to setup

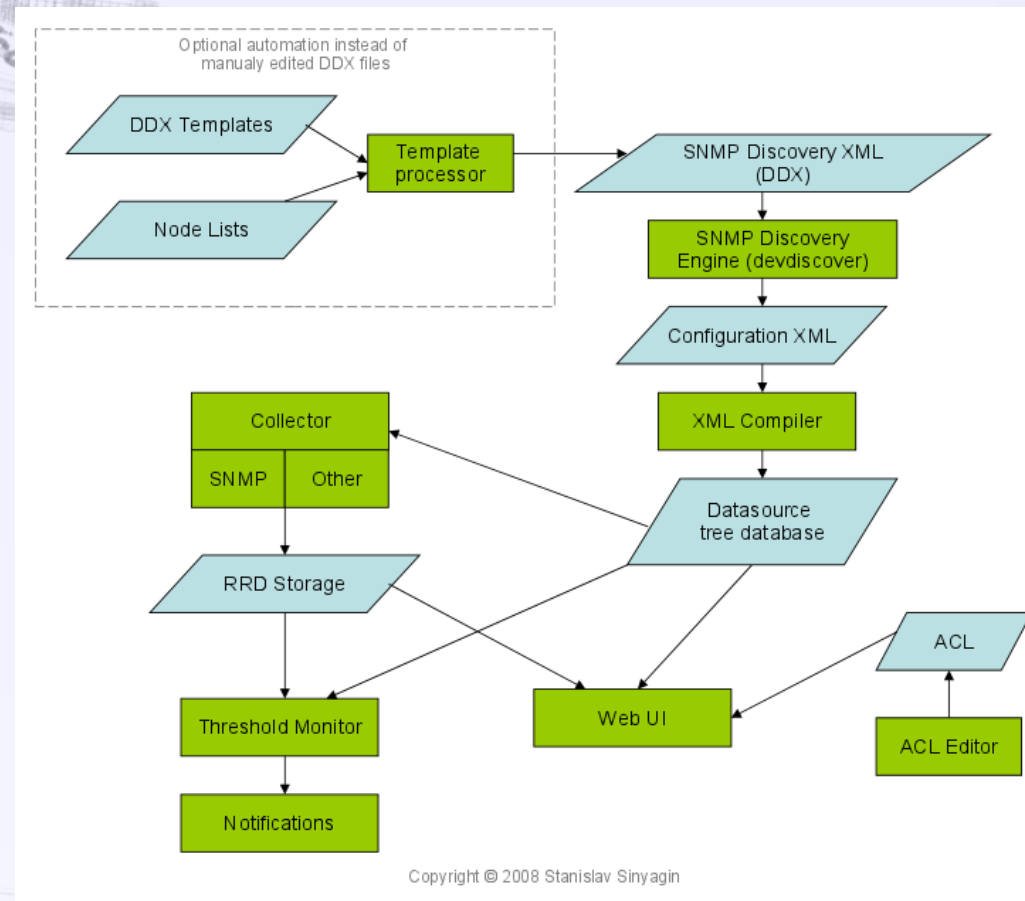
Collecting SNMP data

- **Torrus**

- Efficiently collects data via SNMP and stores the time series in RRD files
 - Used disk space predictable from amount of data sources
 - does not increase over time
- Mature project started 2004
- Claims to collect > 1 million SNMP metrics on a single box
 - Granularity: 5 mins
- Some AJAX interactivity added to RRD graphs by a student
- Currently collecting ~ 55000 values per 5 min without much hassle
- Extensible
 - For adding new OIDs
 - Non-SNMP data sources may be added
 - Even RRD may be replaced
 - But needs (Perl-)coding

Torrus workflow

- **Auto-discovery** of the device capabilities
- Specifying a list of hosts is sufficient to get started
- Converted in a **mind-boggling XML config tree**
- But very **flexible**



Tree: IT

Current path:

/

Linux/

lustig/

lxfs242.gsi.de/

Disk_IO/

sdb/

ReadBps

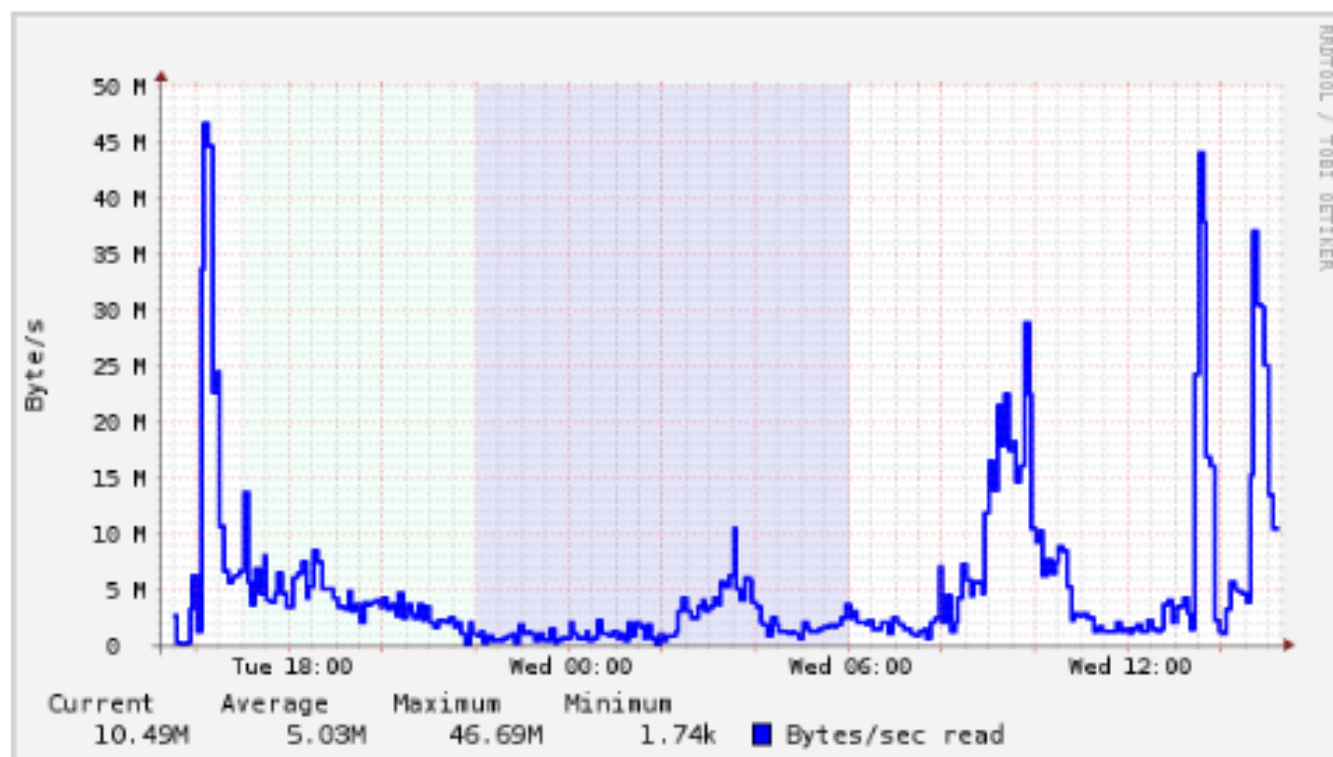
[Top](#) [Up](#)

Graphs for Disk I/O throughput stats

Bytes read

Location: Somewhere at GSI Darmstadt**Contact:** GSI HPC department <hpc@gsi.de>**Uptime:** 146 days since 12-04-2013 14:03**Description:** Linux lxfs242 2.6.32+lustre1.8.7-wc+0.credativ.squeeze.2 #1 SMP Fri Mar 2 02:14:57 CET 2012 x86_64Min: 1.7 k, Avg: 5.0 M, Max: **46.7 M**, Last: 10.5 M

Last day graph



Torrus-Nagios-Integration

- Nagios → Torrus
 - Create discovery lists from Nagios servicegroups
- Torrus → Nagios
 - Torrus monitor: something that is executed when an expression is met
 - not only simple data values possible but complex expressions
 - Nagios monitor writes to Nagios external commands named pipe
 - Nagios can be mis-used as event filter
 - Three monitor definitions: WARNING, CRITICAL and **OK**
 - Automatic name mangling → ugly service names in Nagios.
 - Improvements possible with Torrus v2

Conclusions

- Generating the Nagios config on the fly solves much more problems than it causes.
- SNMP is probably the only universal monitoring agent esp. if other devices than computers have to be queried.
- Knitting together several specialist applications is more versatile than one ACME monitoring solution.
- Choice of tools is not the key factor but their ability to interact and exchangeability

Todo

- Short-term
 - Improve documentation and encourage further utilisation
 - Add Infiniband switches and WAPs
 - Test Torrus reporting
 - Generate network weathermap from Netdisco and Torrus
 - Model on-call duty cycles with Nagios
 - And implement SMS notifications
 - Generate users and groups from TTS queue configurations
 - Add Nagios acknowledgements via TTS ticket locking

Todo II

- Medium-term
 - Further automate configuration generation and add sanity checks
 - Implement Layer3 network topology auto-discovery
- Long-term
 - Abstract the information gathering from the Nagios config generation?
 - Integrate Torrus and collectd?