System Monitoring

OMD // Check_MK // Nagios

about

Jörg Wiemann

* 1986

- Consultant bei Kite Consult
  joerg.wiemann@kite-consult.de
- Kernthemen: Monitoring, Netzwerkdesign/Security, Virtualisierung
- Beginn mit Nagios in 2009 | KMU
  Später Check_MK und OMD
- ca. 1,5 Jahre Microsoft SCOM | Enterprise Umgebungen

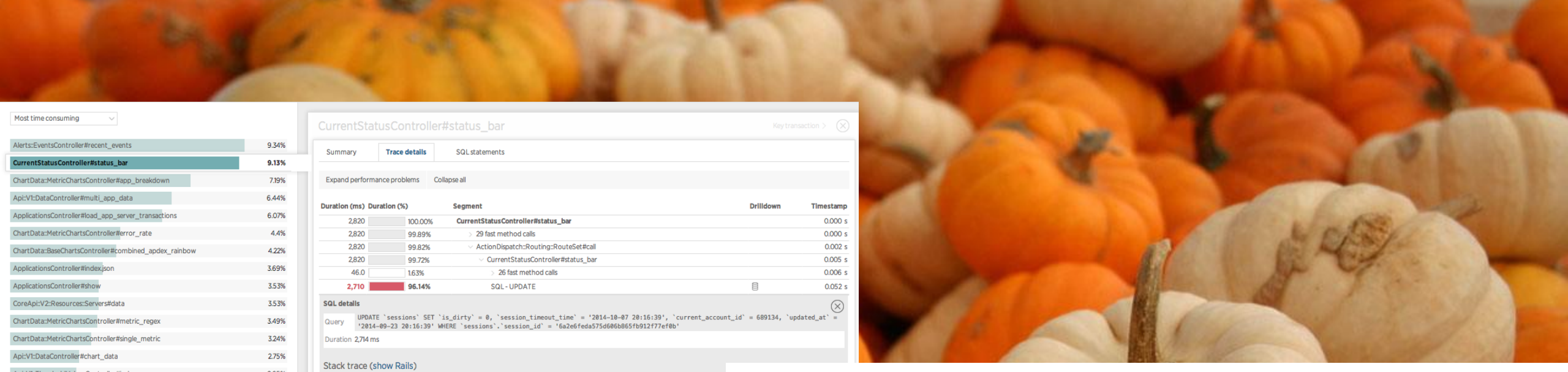
motivation

Abhängigkeit von IT

- Automatische Überwachung aller relevanten IT Komponenten (24x7)
- Benachrichtigung bei Ausfällen
- Vermeidung von Ausfällen durch vorzeitige Informierung
- Troubleshooting
- Metriken / SLAs

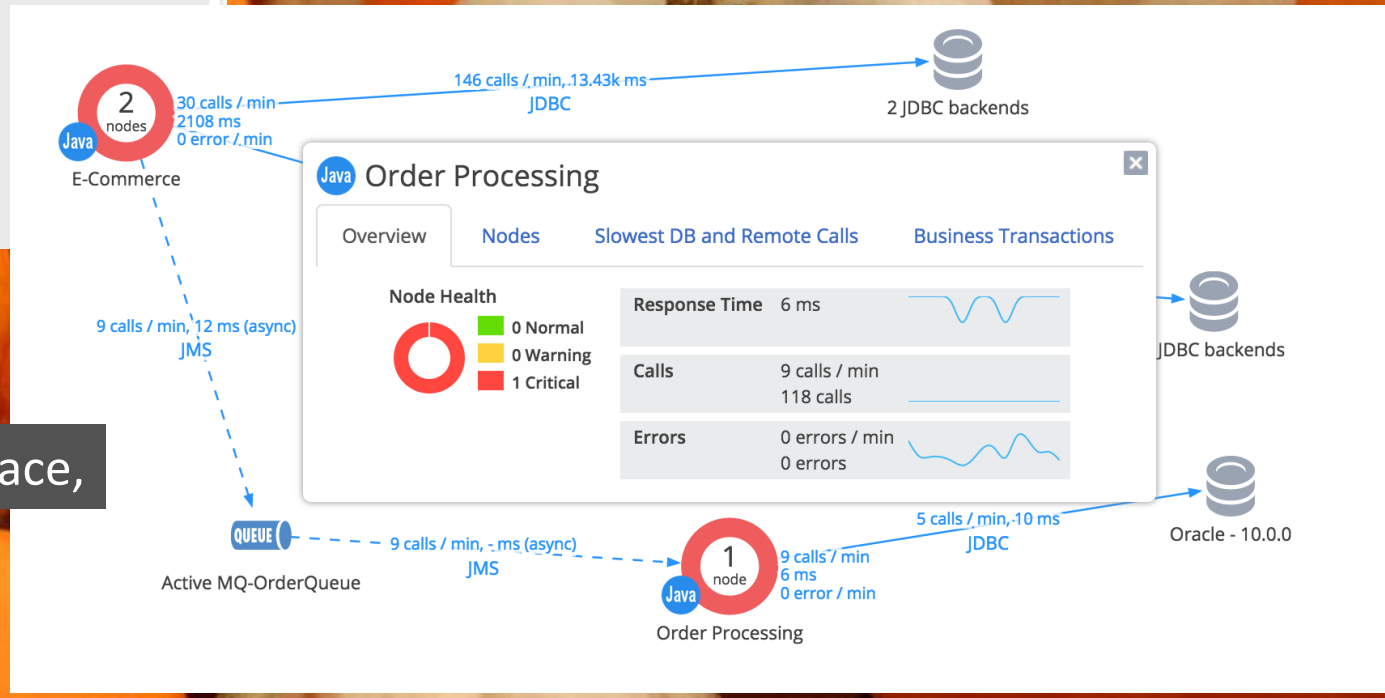
✓ disk space, cpu/mem, backups, mailing, ups, internet connectivity, services, processes, business applications, network devices, printers


was darf‘s denn sein?
Nagios (Core / XI),
Opsview,
Icinga,
GroundWork,
OP5,
Shinken,
NetEye,
SM-BOX,
Zenoss,
Zabixx,
OpenNMS,
Sensu,
Spiceworks,
Paessler PRTG,
servereye,
monitis,
WhatsUpGold,
SolarWinds,
Microsoft SCOM,
HP OpenView,
vRealize Operations / Hyperic,
ca Opscenter,
AppDynamics,
boundary,
New Relic,
Dynatrace,
etc…



src: http://newrelic.com/application-monitoring/features

AppDynamics, boundary, New Relic, Dynatrace,



src: http://www.appdynamics.com/product/application-performance-management/

the beginning is joyous

- Sie können und möchten alles überwachen
  - E-Mail Benachrichtigungen funktionieren schnell und zuverlässig
- Wenige Systeme zu Beginn halten den Aufwand in Grenzen
  - Viele Erweiterungsmöglichkeiten durch Plugins

…but then

- Mit steigender Anzahl von Systemen steigt auch Ihr Konfigurationsaufwand
  - Manuelle Konfiguration (Text/GUI)
  - Die Anzahl von Fehlalarmen nimmt zu
    - *» Sie ignorieren Ihre Benachrichtigungen (besonders die Warnungen)*
- Ihr Monitoring System ist statisch
  - Host-Orientiert

```
    contact_name                    jdoe
    alias                           John Doe
    host_notifications_enabled      1
    service_notifications_enabled   1
    service_notification_period     24x7
    host_notification_period        24x7
    service_notification_options    w,u,c,r
    host_notification_options       d,u,r
    service_notification_commands   notify-by-email
    host_notification_commands      host-notify-by-email
    email                           jdoe@localhost.localdomain
    pager                           555-5555@pagergateway.localhost.localdomain
    address1                        xxxxx.xyyy@icq.com
    address2                        555-555-5555
    can_submit_commands             1
    }


define contactgroup{
    contactgroup_name               novell-admins
    alias                           Novell Administrators
    members                         jdoe,rtobert,tzach
    }

define timeperiod{
    timeperiod_name                 nonworkhours
    alias                           Non-Work Hours
    sunday                          00:00-24:00
    monday                          00:00-09:00,17:00-24:00   ; Every Monday of every week
    tuesday                         00:00-09:00,17:00-24:00   ; Every Tuesday of every week
    wednesday                       00:00-09:00,17:00-24:00   ; Every Wednesday of every week
    thursday                        00:00-09:00,17:00-24:00   ; Every Thursday of every week
    friday                          00:00-09:00,17:00-24:00   ; Every Friday of every week
    saturday                        00:00-24:00               ; Every Saturday of every week
    }

define host{
    host_name                       host_name
    alias                           alias
    display_name                    display_name
    address                         address
    parents                         host_names
    hostgroups                      hostgroup_names
    check_command                   command_name
    initial_state                   [o,d,u]
    max_check_attempts              #
    check_interval                  #
    retry_interval                  #
    active_checks_enabled           [0/1]
    passive_checks_enabled          [0/1]
    check_period                    timeperiod_name
    obsess_over_host                [0/1]
    check_freshness                 [0/1]
    freshness_threshold             #
    event_handler                   command_name
```

```
    host_name                       bogus-router
    alias                           Bogus Router #1
    address                         192.168.1.254
    parents                         server-backbone
    check_command                   check-host-alive
    check_interval                  5
    retry_interval                  1
    max_check_attempts              5
    check_period                    24x7
    process_perf_data               0
    retain_nonstatus_information    0
    contact_groups                  router-admins
    notification_interval           30
    notification_period             24x7
    notification_options            d,u,r
    }


define hostgroup{
    hostgroup_name                  novell-servers
    alias                           Novell Servers
    members                         netware1,netware2,netware3,netware4
    }

define service{
    host_name                       linux-server
    service_description             check-disk-sda1
    check_command                   check-disk!/dev/sda1
    max_check_attempts              5
    check_interval                  5
    retry_interval                  3
    check_period                    24x7
    notification_interval           30
    notification_period             24x7
    notification_options            w,c,r
    contact_groups                  linux-admins
    }

define servicegroup{
    servicegroup_name               dbservices
    alias                           Database Services
    members                         ms1,SQL Server,ms1,SQL Server Agent,ms1,SQL DTC
    }

define command{
    command_name                    check_pop
    command_line                    /usr/local/nagios/libexec/check_pop -H $HOSTADDRESS$
    }

define servicedependency{
    host_name                       WWW1
    service_description             Apache Web Server
```

..but then

- Mit steigender Anzahl von Systemen steigt auch Ihr Konfigurationsaufwand
- Manuelle Konfiguration (Text/GUI)
- Die Anzahl von Fehlalarmen nimmt zu

» Sie ignorieren Ihre Benachrichtigungen (besonders die Warnungen)

- Ihr Monitoring System ist statisch
- Host-Orientiert

…but then

- Es wird doch nicht „alles“ überwacht
- Im Monitoring werden „Schuldige“ gesucht
- Die Monitoring-Abfragen sind unzureichend


was darf‘s denn sein?
OMD + Check_MK
(Nagios Core)

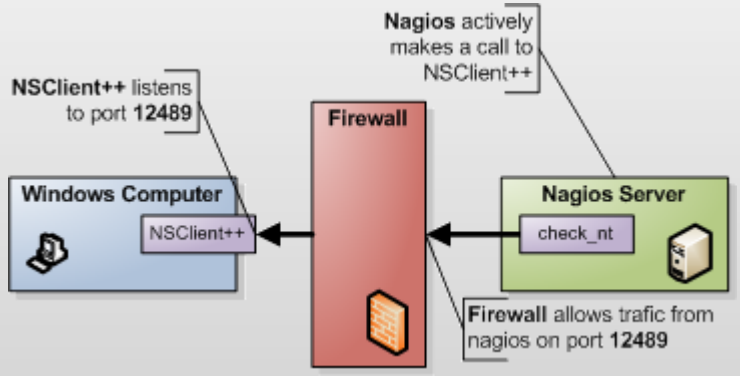
nagios

Framework / scheduling engine

- Release: 1999
- OS: Linux
- Input/Output: Textdateien (Konfiguration / status.dat)

Hosts:	Server/Netzwerkgeräte o.ä. („Container“)
Service:	Auf Hosts zu überwachende Elemente
Plugins:	Programme/Skripte, die Abfragen ausführen (check_http, check_printer)

Aktives + Passives Monitoring



src: http://nsclient.org/nscp/wiki/doc/usage/nagios

nagios

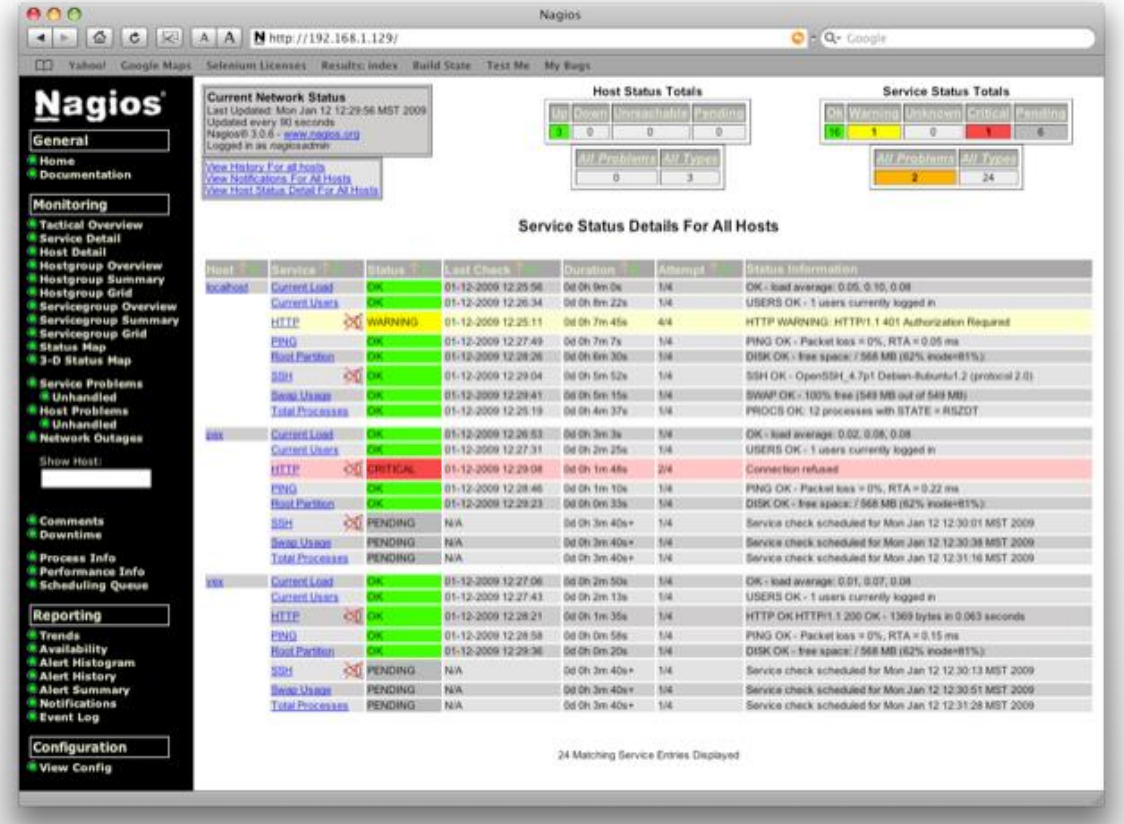
Host checks (ping) alle n Minuten

- UP
- UNREACHABLE
- DOWN

Service checks (plugin) alle n Minuten

- OK
- WARNING
- UNKNOWN
- CRITICAL

Auslösung eines Events/Kommandos bei Statuswechsel

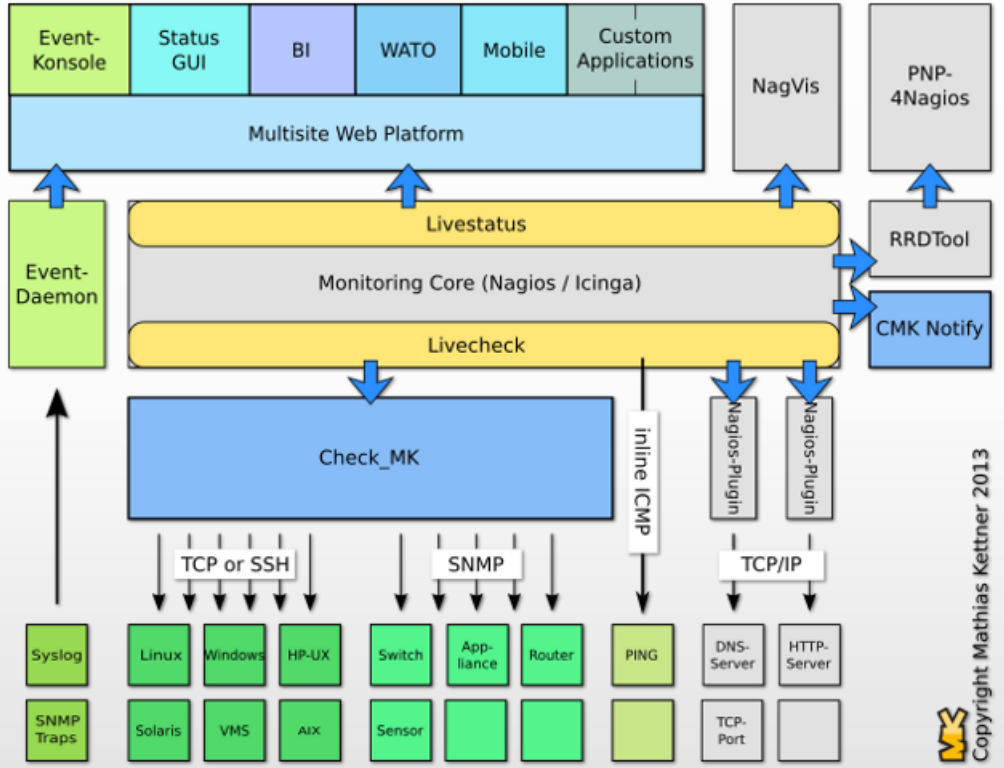


check_mk

CMK Projekt: Sammlung von Erweiterungen zum Nagios Monitoring-Kern

Kernbestandteile (Auszug):

- Configuration & Check Engine (+ check_mk_agent)
  Automatische Serviceerkennung und Konfigurationserzeugung
- Livestatus
  Nagios-Broker-Modul für Zugang zu Statusdaten von Hosts und Services
- Multisite
  Web-GUI: verteiltes Monitoring durch mehrere Instanzen, Integration von NagVis/PNP4Nagios, LDAP-Anbindung
- WATO
  Check_MK Web Administration Tool



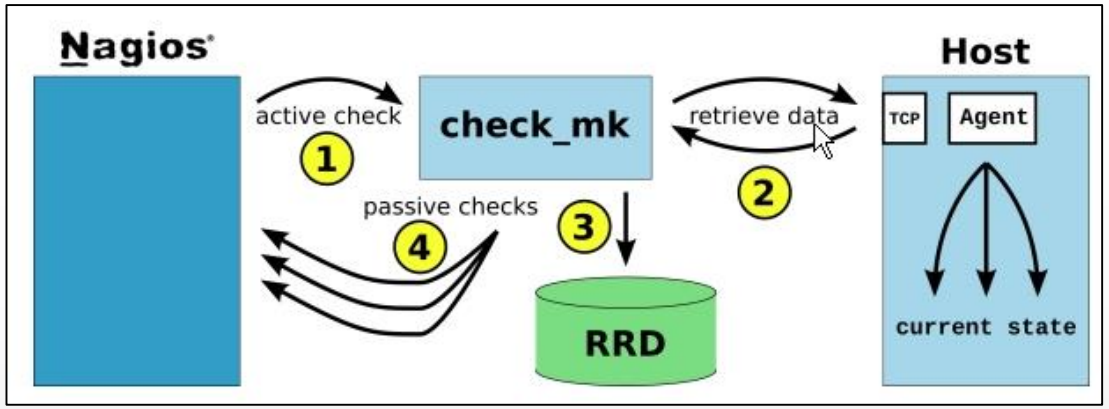
src: https://mathias-kettner.de/check_mk.html

check_mk agent

Merkmale

- Serviceerkennung durch den Check_MK Agenten
- Passive checks
  Alle Hostdaten werden in einem Aufruf übermittelt
  Performancedaten werden in eine RRD gespeichert

- Automatische Inventarisierung
- Automatische Konfigurationserzeugung



main.mk

```
inventory_services = ['TSMListener', 'Httpd', 'TapiSrv' ]
```

src: https://mathias-kettner.de/check_mk.html

check_mk agent

Windows Agent

- Installation als Windows Service
- Größe < 140 KB
- Konfigurationsfrei
- Abfrage über 6556/TCP

Installation

C:\some\directory\> check_mk_agent.exe install

C:\some\directory\> net start check_mk_agent

Linux Agent

- Installation über Repository oder Monitoring via SSH

State	Service	Icons	Status detail	Age	Checked	Perf-O-Meter
OK	Check_MK		OK - Agent version 1.1.13i3, execution time 0.2 sec	2 min	2 sec	0.2s
OK	CPU utilization		OK - 11% used / 2 CPUs (in last 3 secs)	2 min	2 sec	12%
OK	Disk IO SUMMARY		OK - 0.00B/sec read, 38.00KB/sec write	2 min	2 sec	0.00M/s 0.04M/s
OK	fs_C:/		OK - 30.0% used (17.99 of 60.0 GB), (levels at 80.0/90.0%), trend: +27.28MB / 24 hours	2 min	2 sec	29%
OK	LOG Application		OK - no error messages	2 min	2 sec	
OK	LOG Internet Explorer		OK - no error messages	2 min	2 sec	
OK	LOG Security		OK - no error messages	2 min	2 sec	
OK	LOG System		OK - no error messages	2 min	2 sec	
OK	Memory and pagefile		OK - Memory usage: 25.8% (0.3/1.0 GB), Page file usage: 6.8% (0.2/2.4 GB)	2 min	2 sec	25%
OK	System Time		OK - Offset is +0.7 sec (levels at 30/60 sec)	2 min	2 sec	0.7 s
OK	Uptime		OK - up since Wed Jun 27 14:23:24 2012 (0d 00:05:43)	2 min	2 sec	00d 00h 05m

```
user@host> telnet windowshost 6556
<<<check_mk>>>
Version: 1.1.13i1
AgentOS: windows
WorkingDirectory: C:\some\directory
ConfigFile: C:\some\directory\check_mk.ini
AgentDirectory: C:\some\directory
PluginsDirectory: C:\some\directory\plugins
LocalDirectory: C:\some\directory\local
OnlyFrom: 0.0.0.0/0
<<<uptime>>>
12227
<<<df>>>
C:\        NTFS     62902472 18363880 44538592  30% C:\
```

src: https://mathias-kettner.de/check_mk.html

check_mk config

Config Syntax

main.mk

```
all_hosts = [
 'smucsrv05|muc',
 'smucsrv06|muc',
 'smucsrv08|muc|test',
 'smucsrv11|muc|test',
 'sparsrv04|par|sap',
 'sparsrv06|par|test|sap|RX_04',
 'smat01|ping',
 'smat02|ping',
]
```

main.mk

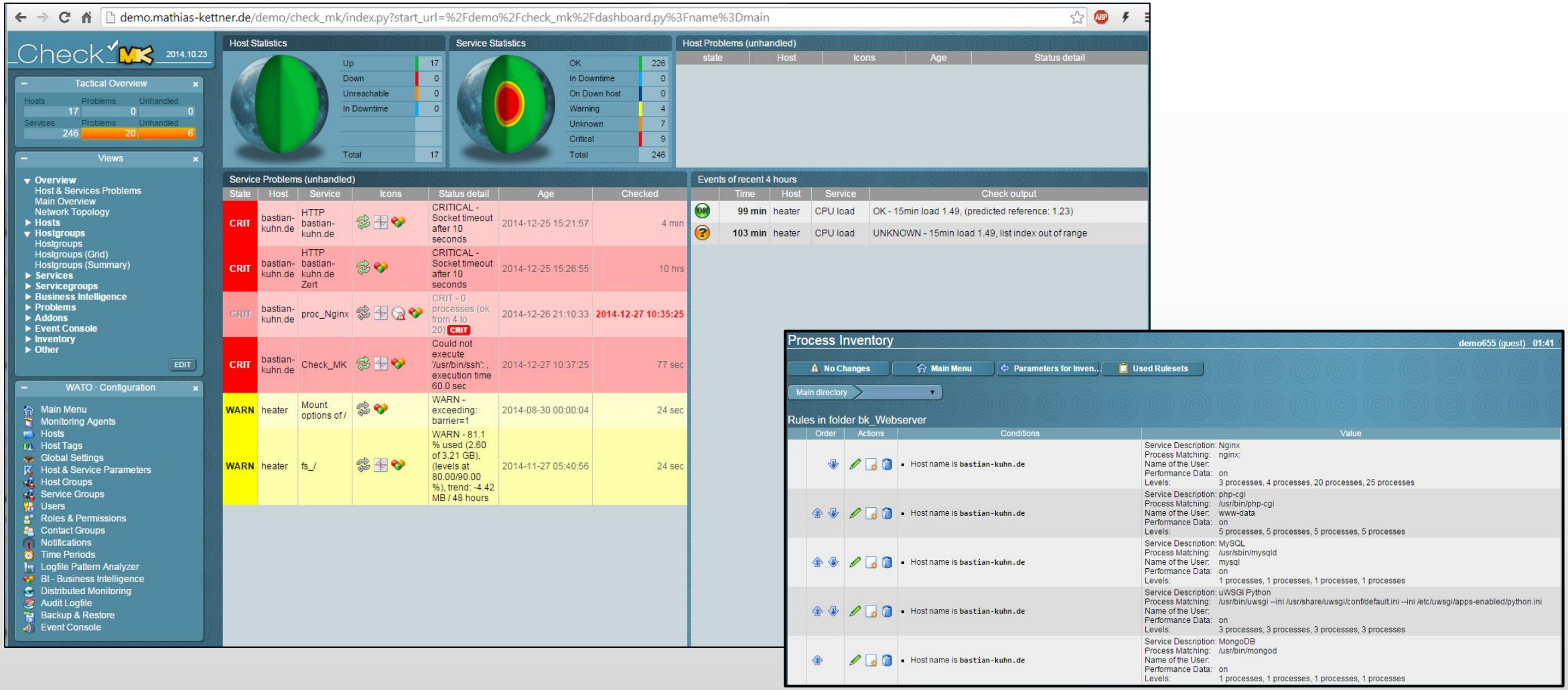
```
host_groups = [
 ( 'munich', [ 'muc' ],         ALL_HOSTS ), # all hosts with tag muc
 ( 'saptest',[ 'sap', 'test' ], ALL_HOSTS ), # hosts with tags muc and test
 ( 'mathias',[ 'smat01', 'smat02' ] ),       # the hosts smat01 and smat02
]
```

main.mk

```
check_parameters += [
 ( { "levels" : (95.0, 98.0)} , ALL_HOSTS, [ "fs_/var" ] ),
 ( { "levels" : (75.0, 85.0)}, [ "win" ], ALL_HOSTS, [ "fs_" ] ),
]
```

src: https://mathias-kettner.de/check_mk.html

multisite


demo.mathias-kettner.de/demo/check_mk/index.py?start_url=%2Fdemo%2Fcheck_mk%2Fdashboard.py%3Fname%3Dmain
Check_MK 2014.10.23
Tactical Overview
Views
WATO - Configuration
Host Statistics
Service Statistics
Host Problems (unhandled)
Service Problems (unhandled)
Events of recent 4 hours
Process Inventory
Rules in folder bk_Webserver

OMD – Open Monitoring Distribution

Software Bundle aus Nagios und vielen zugehörigen Addons

- Integriert, vorkonfiguriert
- Als RPM/DEB Pakete zu beziehen oder über Repositories

Zusätzliche Features

- Mehrere Instanzen pro Host („sites“)
- Einfache Erstellung von sites
- Performanceoptimiert
  - tmpfs
  - rrdcached

Software inkludiert:

- **Nagios**
  - Monitoring Plugins
  - nsca
  - check_nrpe
- Icinga
- Shinken
- **NagVis**
- **pnp4nagios**
- rrdtool/rrdcached
- **Check_MK**
- **MK Livestatus**
- **Multisite**
- Dokuwiki
- Thruk
- Mod-Gearman
- check_logfiles
- check_oracle_health
- check_mysql_health
- jmx4perl
- check_webinject
- check_multi

src: http://omdistro.org/

OMD Installation

Beispiel Ubuntu Server 14.04 LTS

Download der DEB Datei und Installation:

```
# wget http://files.omdistro.org/releases/debian_ubuntu/omd-1.20.trusty.amd64.deb
# dpkg -i omd-1.20.trusty.amd64.deb
# apt-get -f install
```

Siteerstellung:

```
# omd create kunde1
```

Start der Site (Nagios und alle zugehörigen Prozesse):

```
# omd start kunde1
```

Konfiguration über http://localhost/kunde1

src: http://omdistro.org/

OMD Config

demo

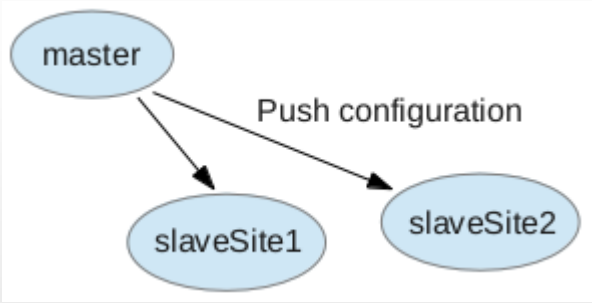
OMD distributed monitoring

Anbindung über multisite

multisite.mk

```
sites = {
   # connect to local Nagios
   "local" : {
        "alias" : "Munich"
   },

   # connect to remote site
   "paris": {
        "alias":          "Paris",
        "socket":         "tcp:10.0.0.2:6557",
        "persist":        True,
    },
}
```



MS Exchange monitoring

Beispiel lokale Checks

Vielen Dank!

Referenzen

OMD: http://omdistro.org

Check_MK: https://mathias-kettner.de/check_mk.html