Event Console Integration

03.05.2018, Sven Panne
Check_MK Conference #4

CONFERENCE
MUNICH 2018/5/2-4
#4

Two Kinds of Monitoring

Status-based

- Pushing/pulling states
- “Classic” Check_MK

Event-based

- Pushing of messages
- Independent of state
- Event Console

Both kinds are useful!

Design Considerations

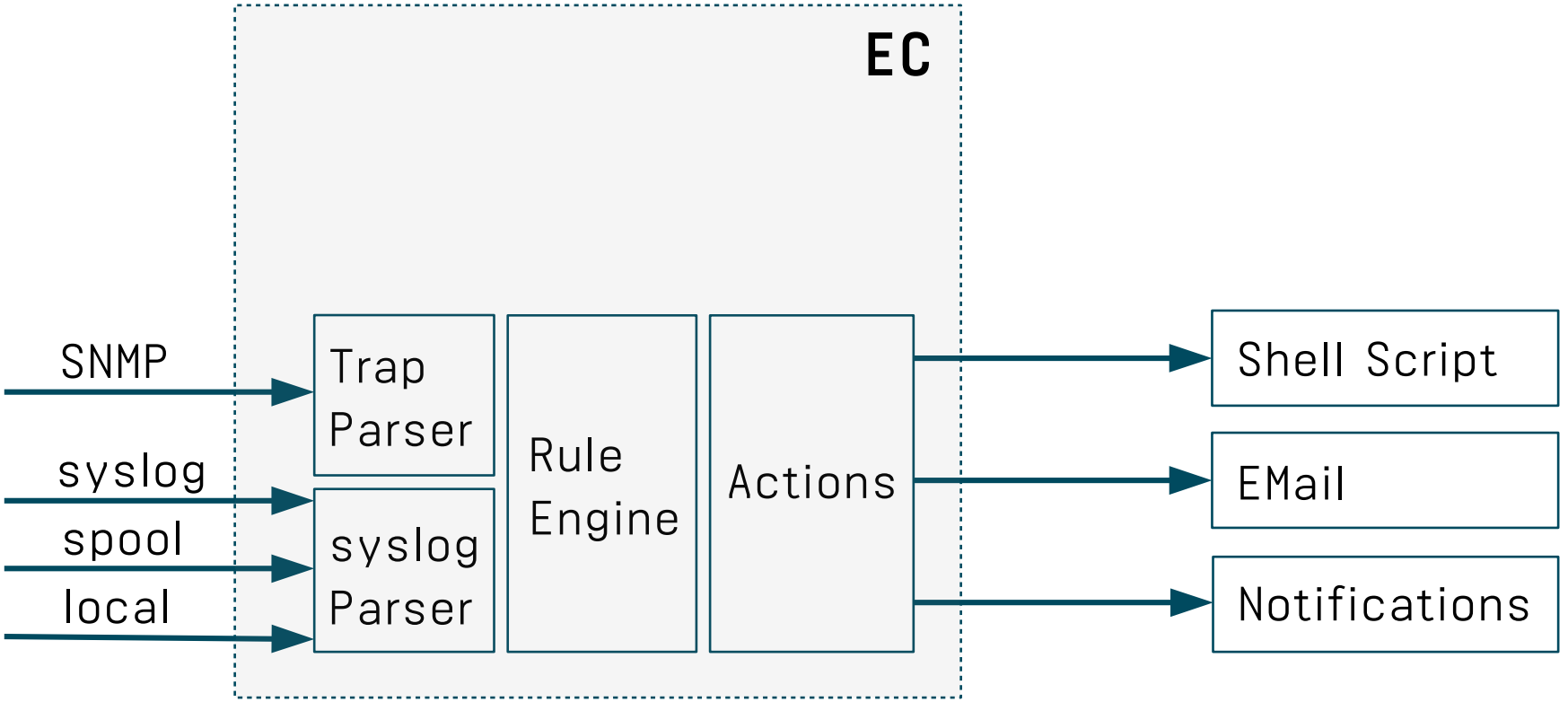
High frequency of incoming messages

Human-manageable amount of actual events

Not a high-performance syslog archive

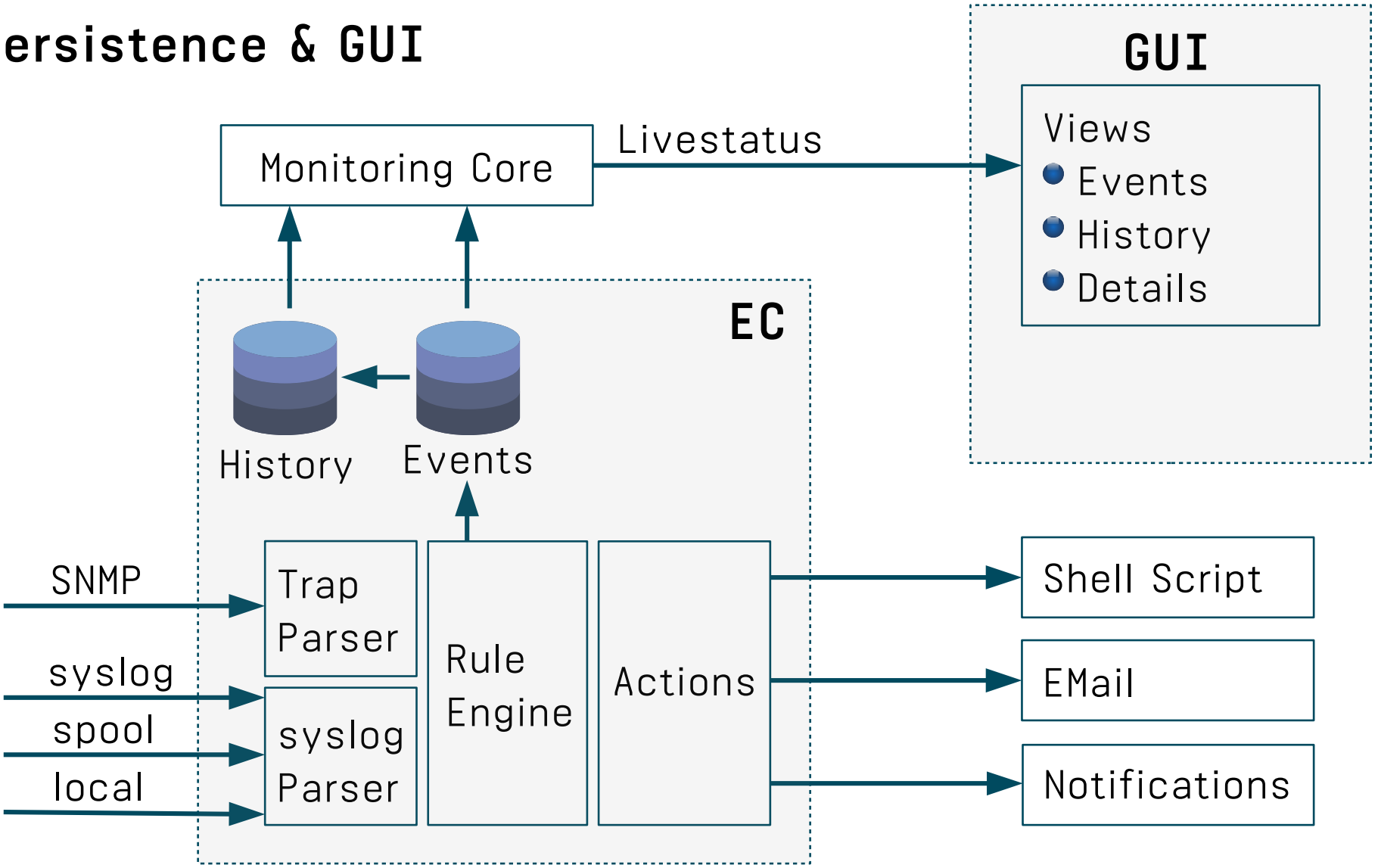
Architectural Overview I

Basic Pipeline



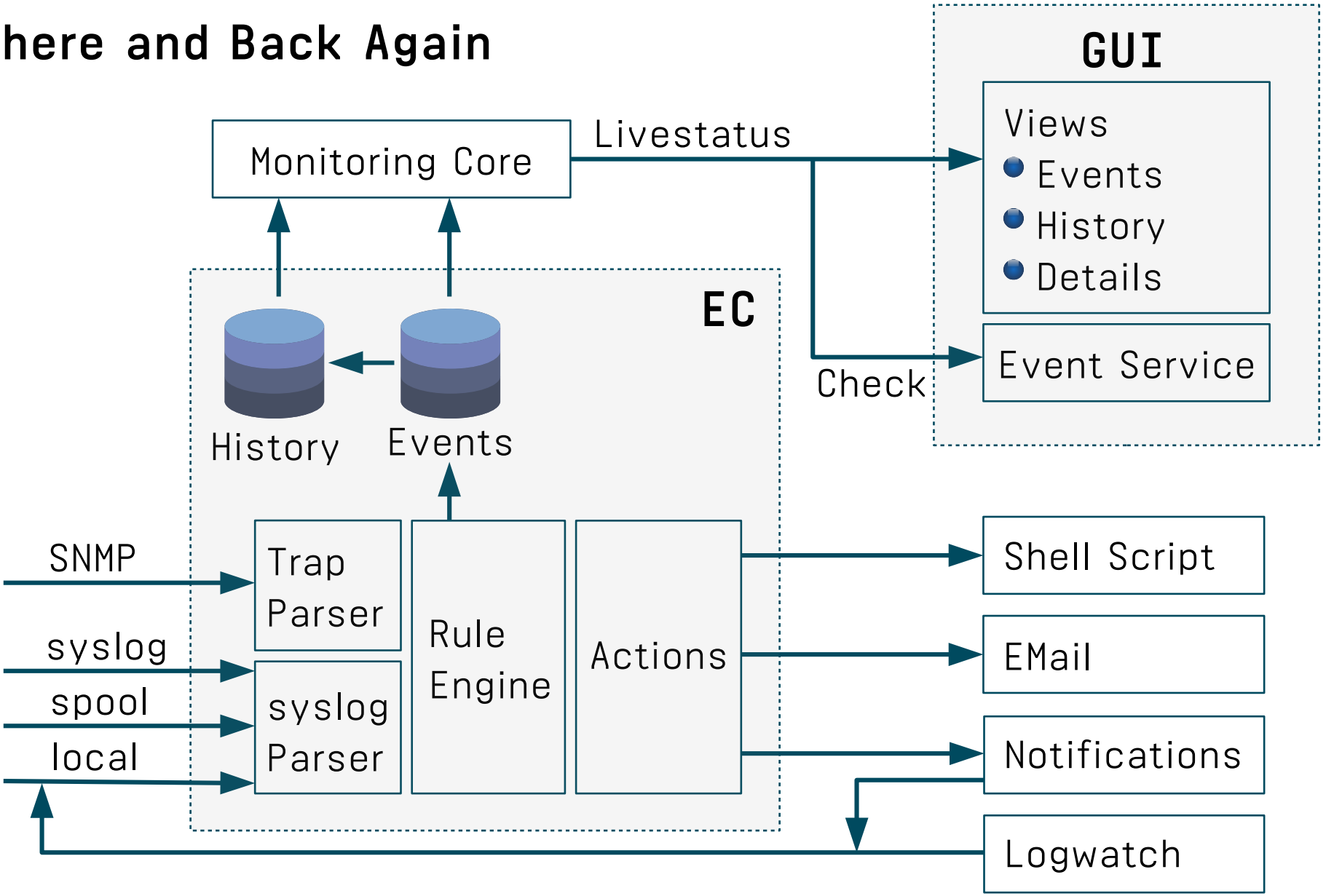
Architectural Overview II

Persistence & GUI



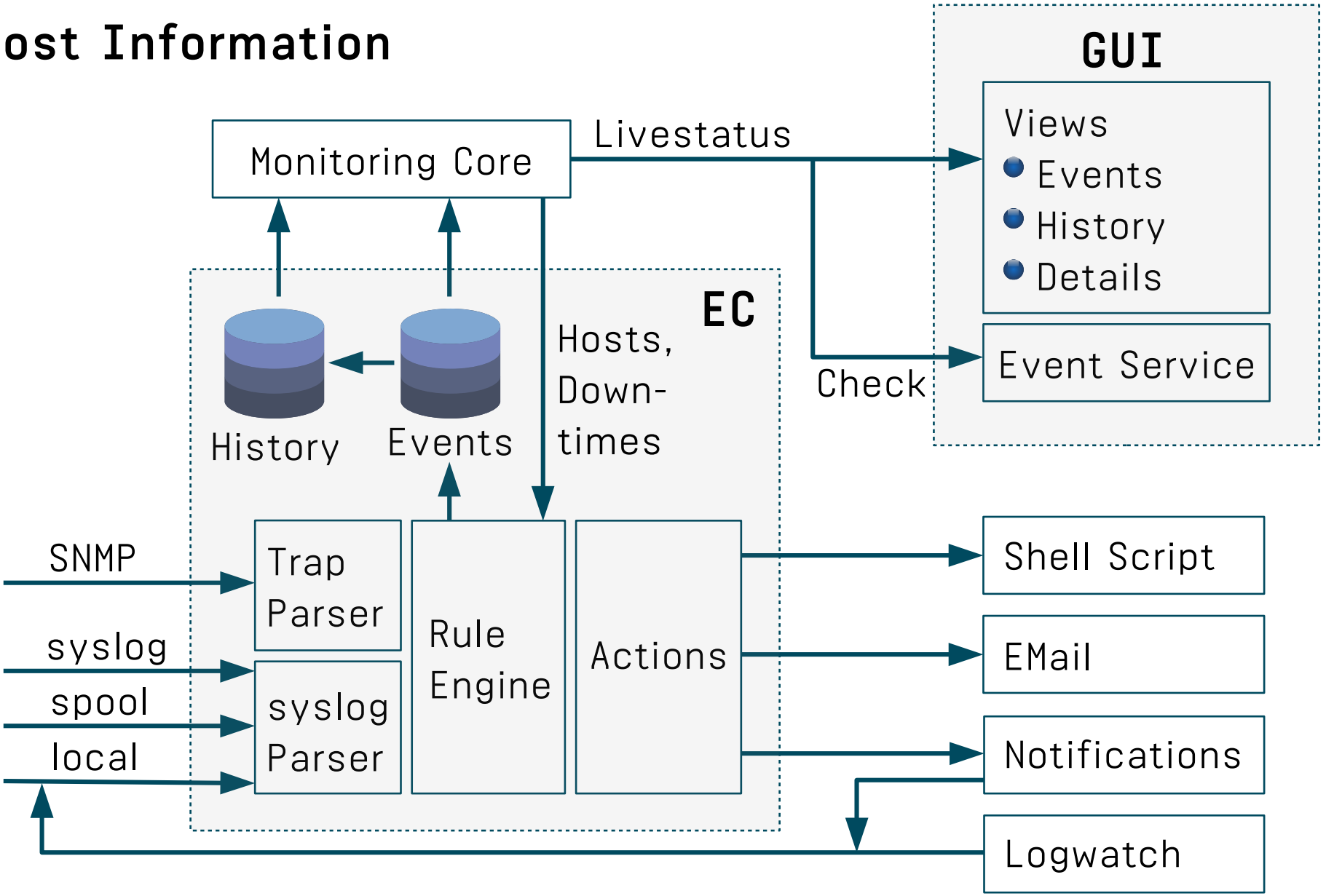
Architectural Overview III

There and Back Again

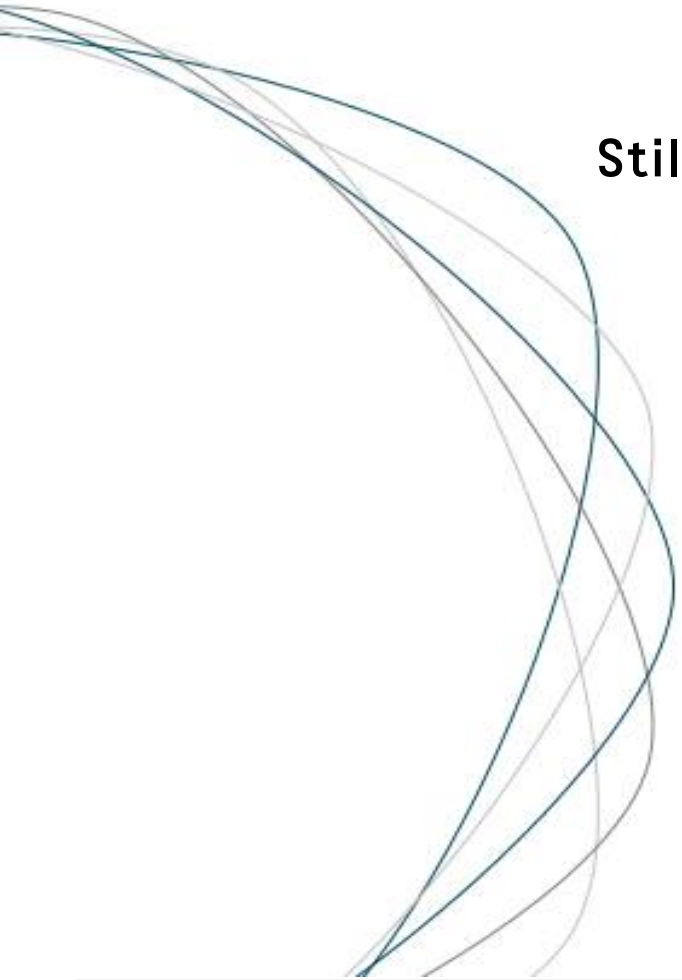


Architectural Overview IV

Host Information



Events are useful, even during host downtimes



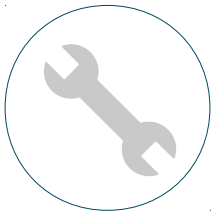
Still generated, but marked specially

Automatically archived after downtime

Tactical overview considers them “handled”

Actions can be skipped

Further Improvements



Finer configuration of limits

More powerful rewriting

Export of rule packs via MKPs

CHECK_MK

CONFERENCE

MUNICH 2018/5/2-4

#4

Background jobs

03.05.2018, Tom Bärwinkel

Check_MK Conference #4

CONFERENCE
MUNICH 2018/5/2-4
#4

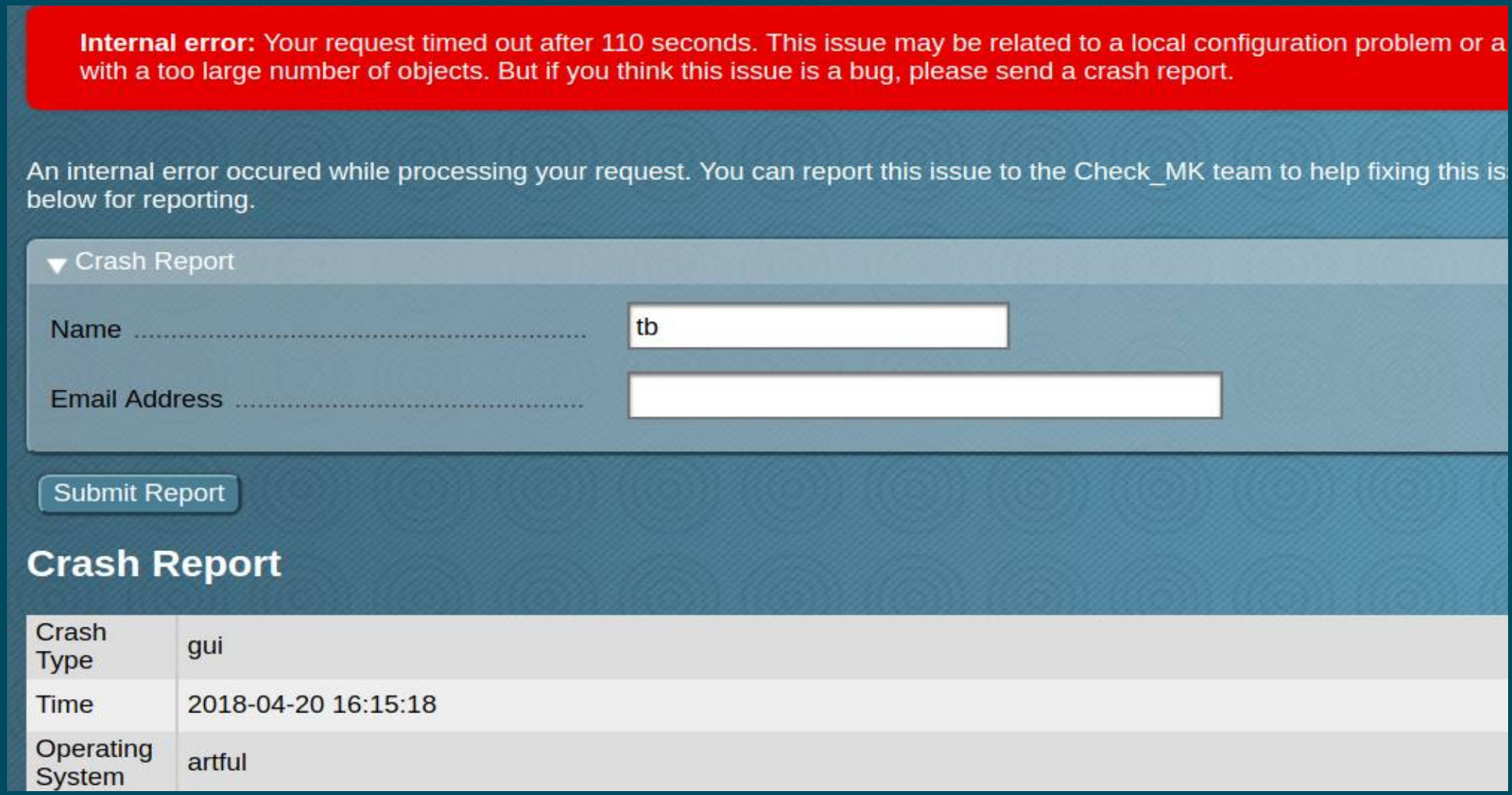
Examples of long running processes

Create Report

Bake Agents

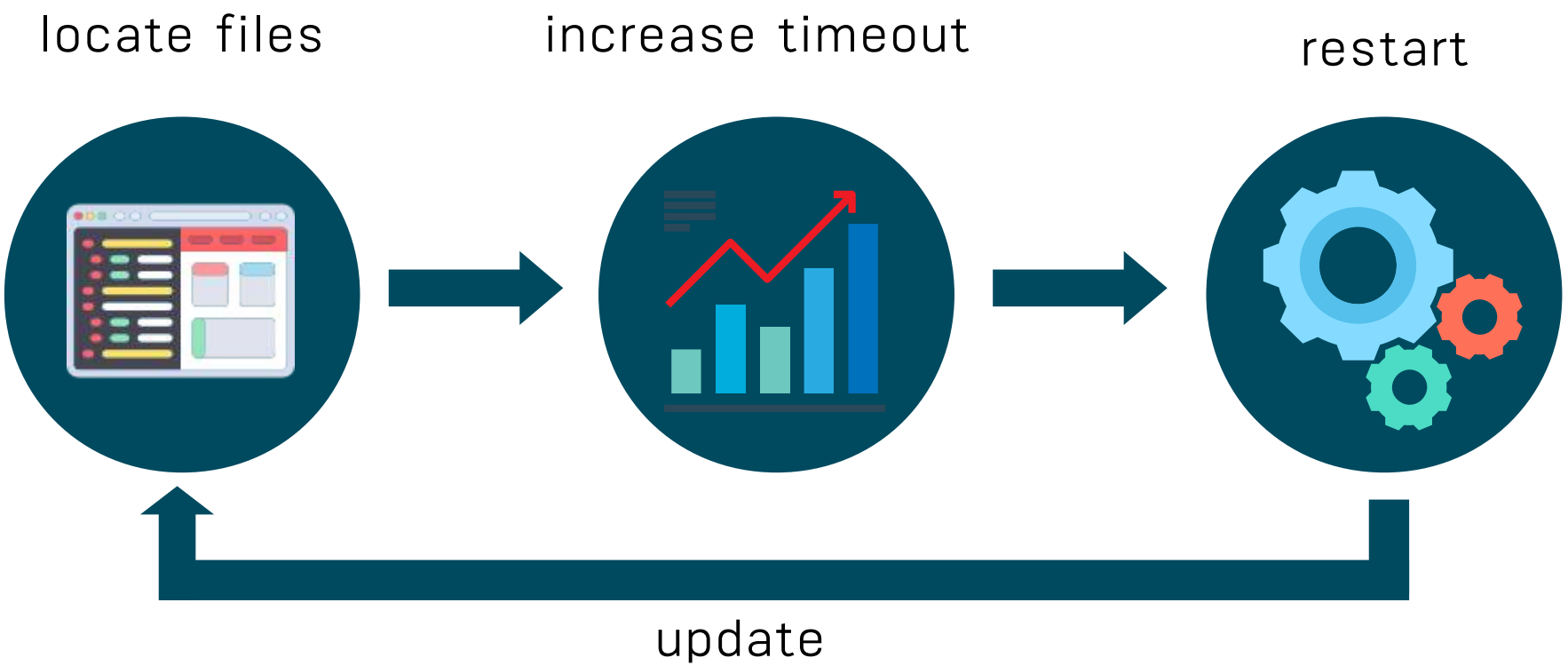
Host renaming

Problem – Apache process timeouts

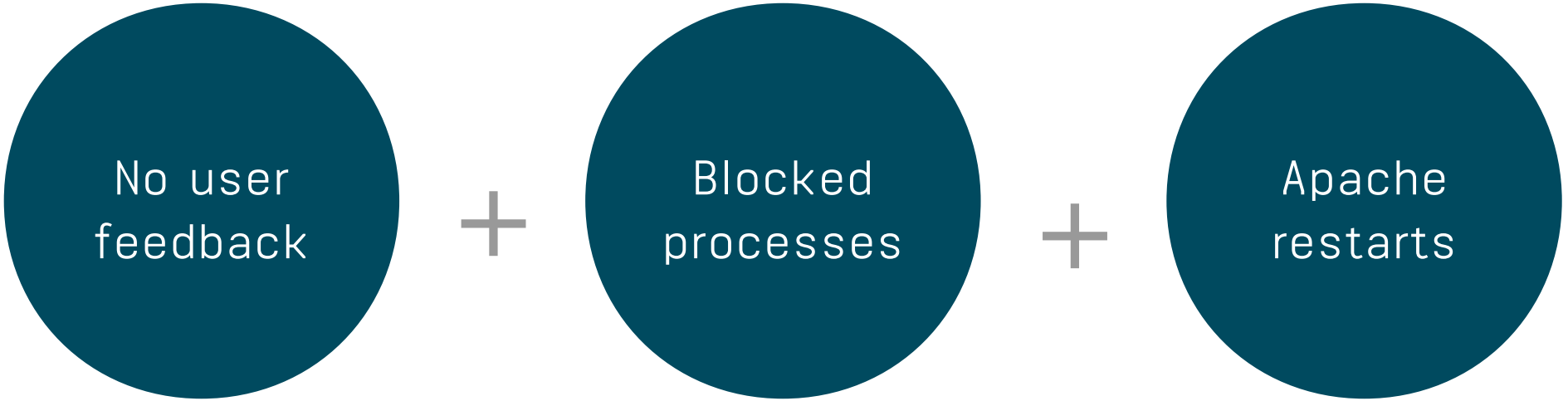


The usual workaround

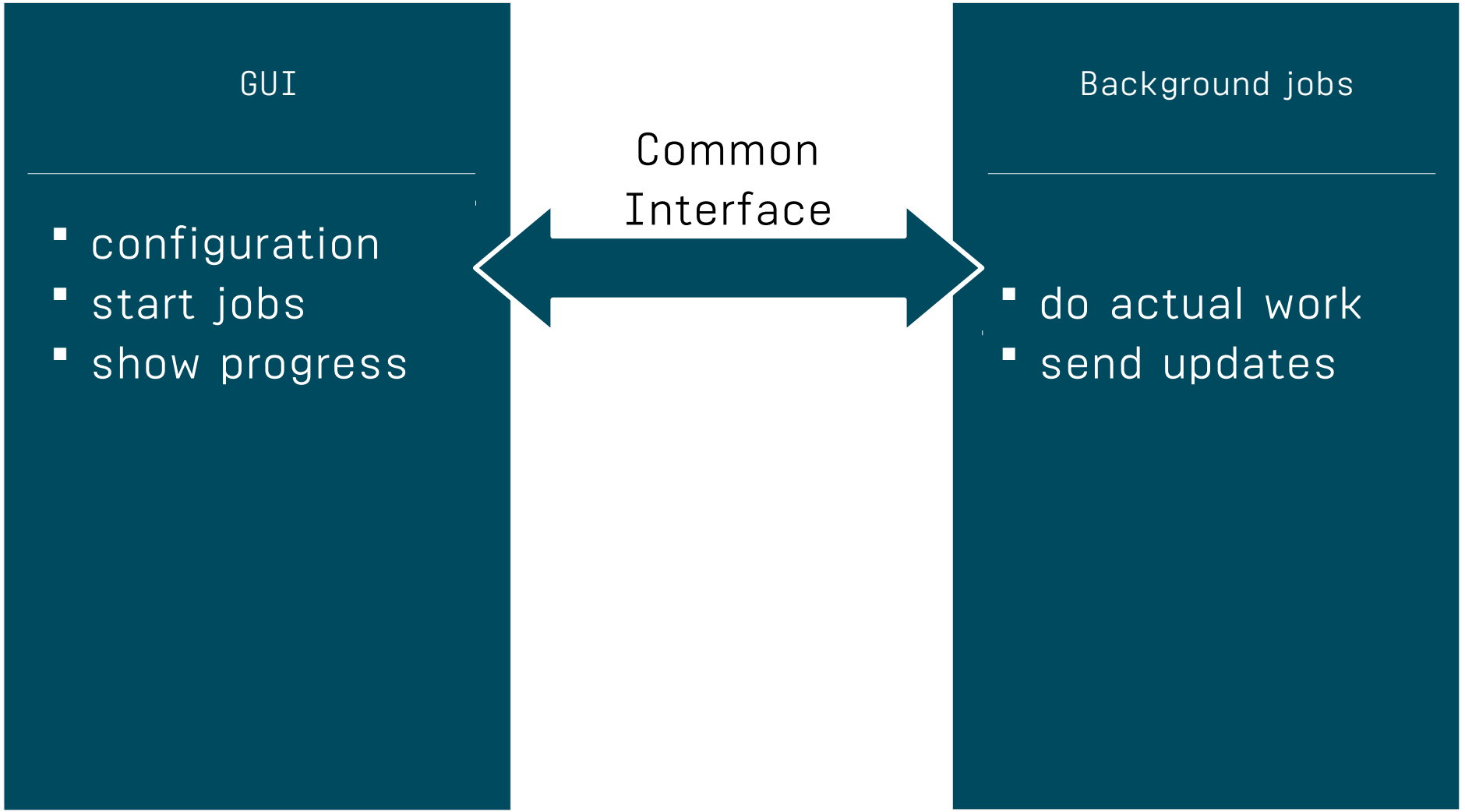
Let’s just increase the timeout



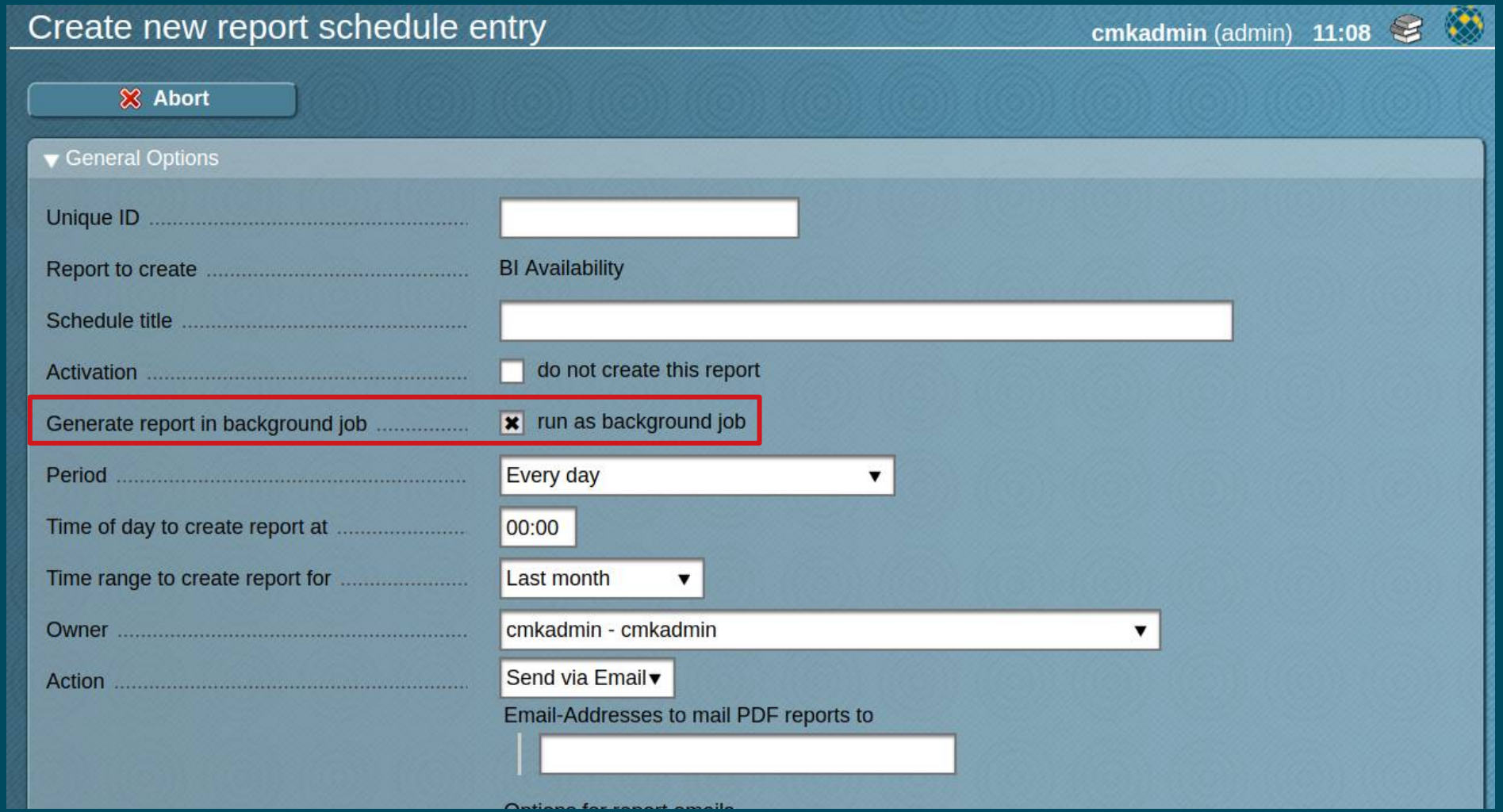
Still a problem - What's the status?



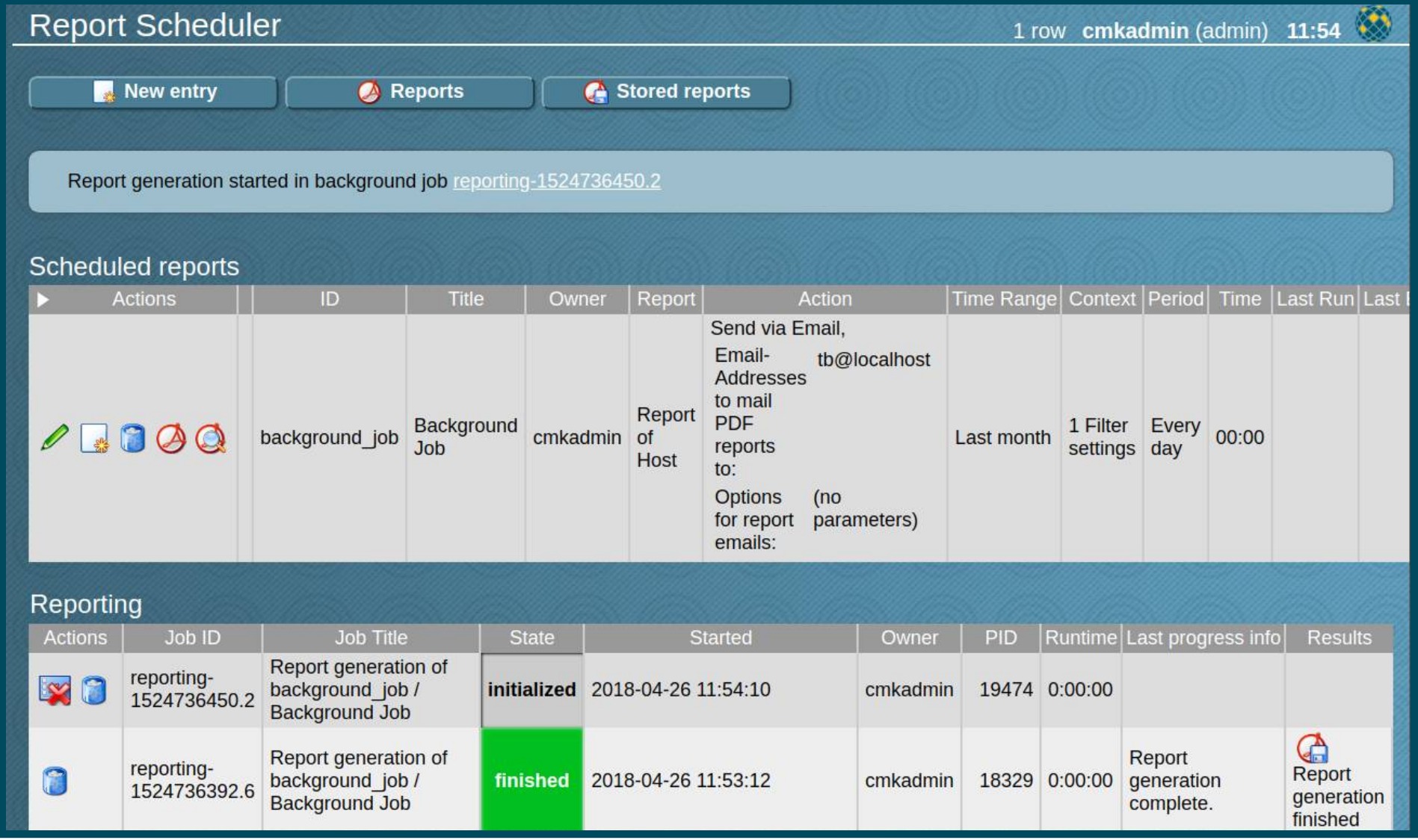
Solution - separation of concerns



How it looks in practice - Reports



How it looks in practice - Reports



Report Scheduler

1 row **cmkadmin** (admin) **11:54**

New entry | Reports | Stored reports

Report generation started in background job reporting-1524736450.2

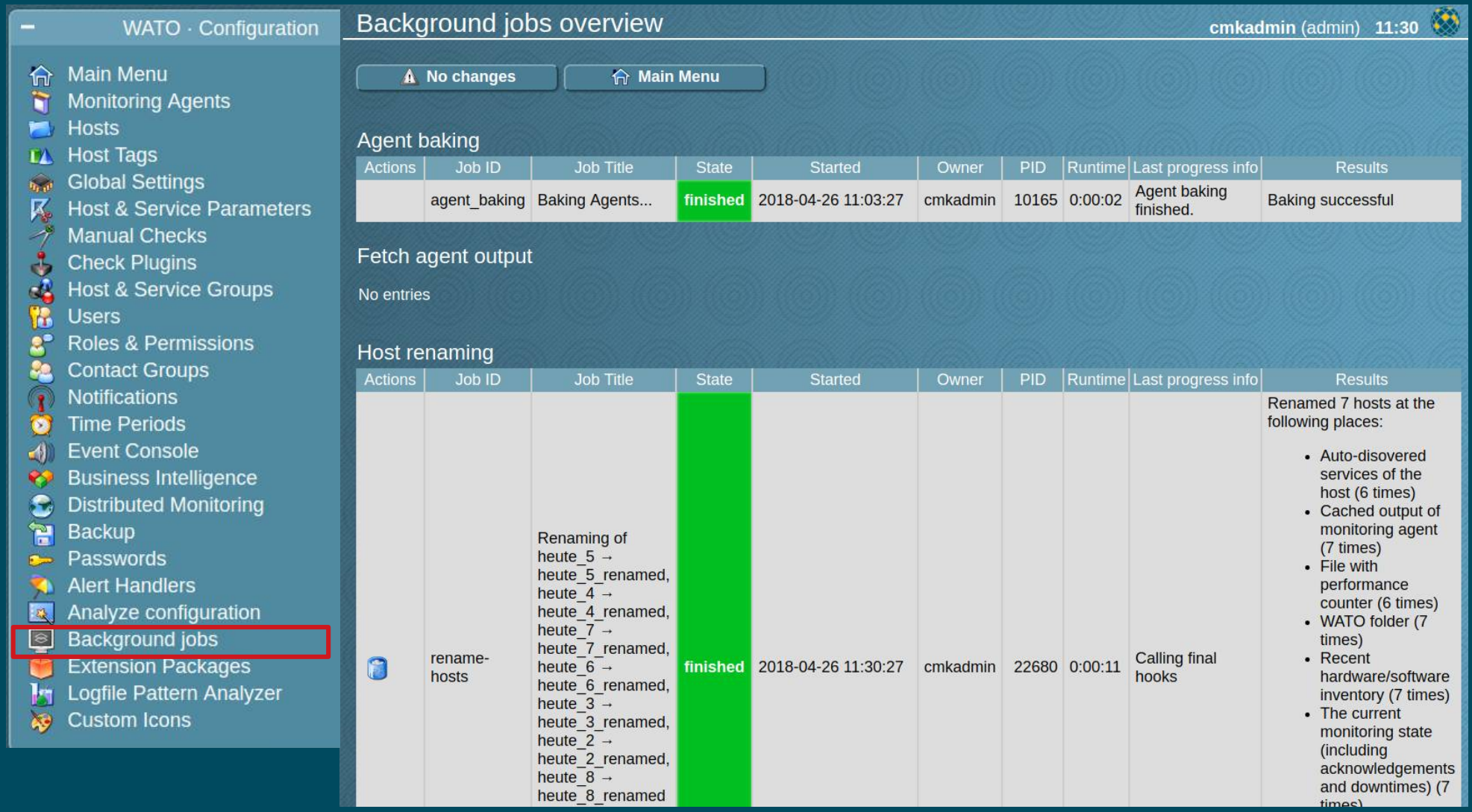
Scheduled reports

Actions	ID	Title	Owner	Report	Action	Time Range	Context	Period	Time	Last Run	Last
	background_job	Background Job	cmkadmin	Report of Host	Send via Email, Email-Addresses to mail PDF reports to: tb@localhost, Options for report emails: (no parameters)	Last month	1 Filter settings	Every day	00:00		

Reporting

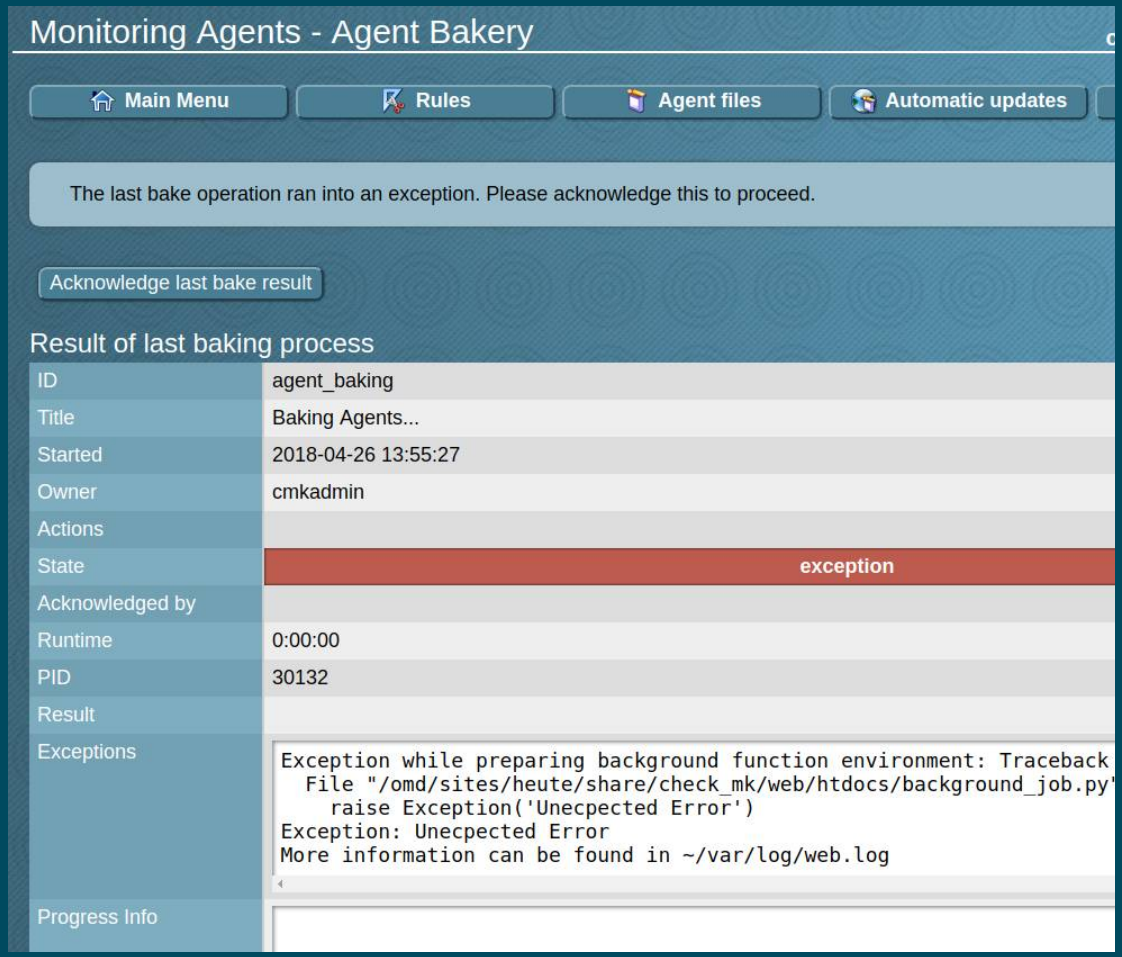
Actions	Job ID	Job Title	State	Started	Owner	PID	Runtime	Last progress info	Results
	reporting-1524736450.2	Report generation of background_job / Background Job	initialized	2018-04-26 11:54:10	cmkadmin	19474	0:00:00		
	reporting-1524736392.6	Report generation of background_job / Background Job	finished	2018-04-26 11:53:12	cmkadmin	18329	0:00:00	Report generation complete.	Report generation finished

The Background jobs overview



Special rules for critical jobs

- Affected jobs:
  - bake agents
  - rename host
- Only one job allowed
- Error page with details
- Require acknowledgement



Upcoming background jobs

Download of agent output

Service discovery

SLA calculations

CHECK_MK

CONFERENCE

MUNICH 2018/5/2-4

#4

Time-specific check parameters

03.05.2018, Konstantin Büttner

Check_MK Conference #4

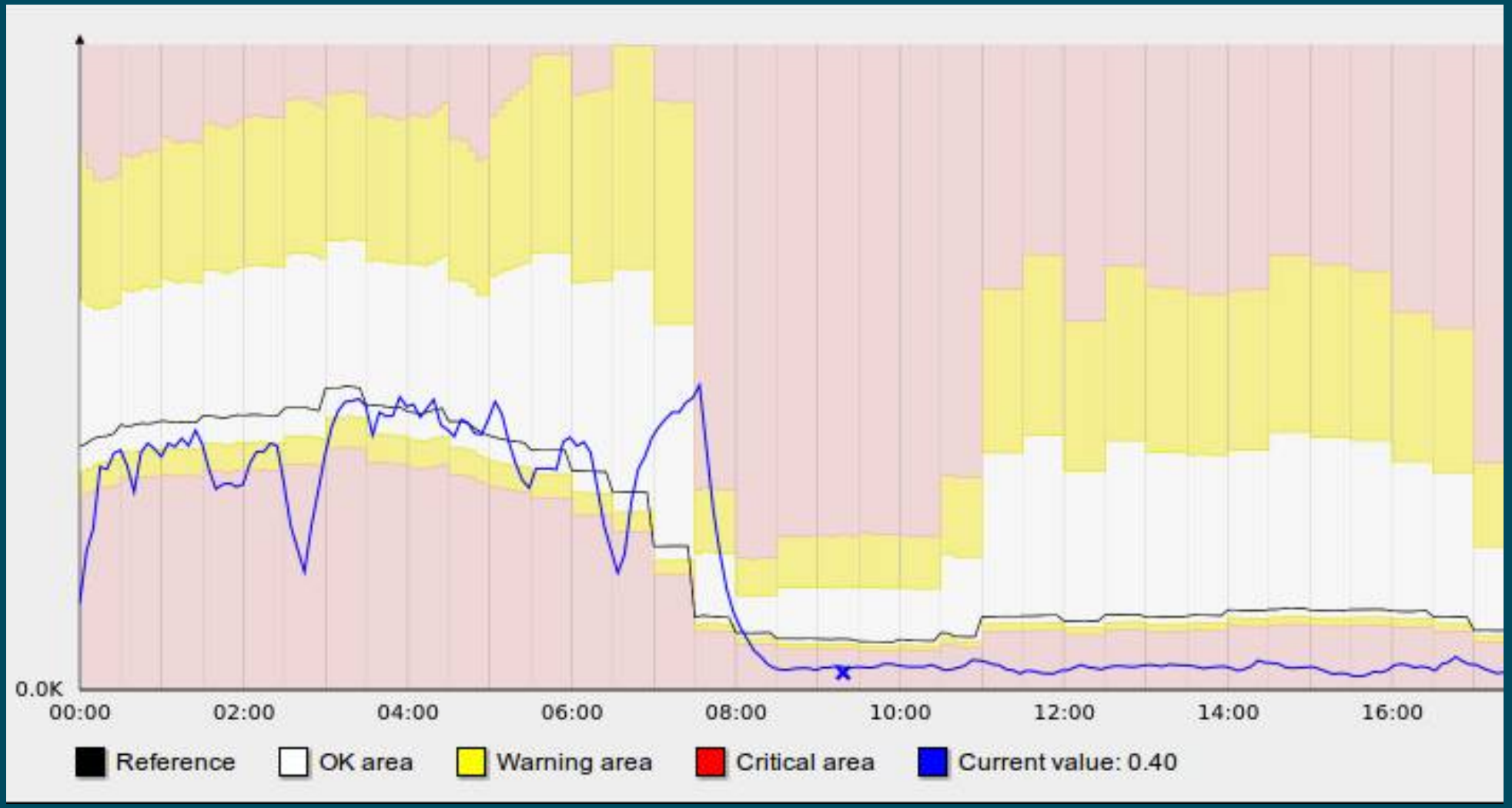
CONFERENCE
MUNICH 2018/5/2-4
#4

Use case

- Recurring events lead to conditions that would be abnormal during other times

- Eg. backup jobs etc.

Predictive Levels?

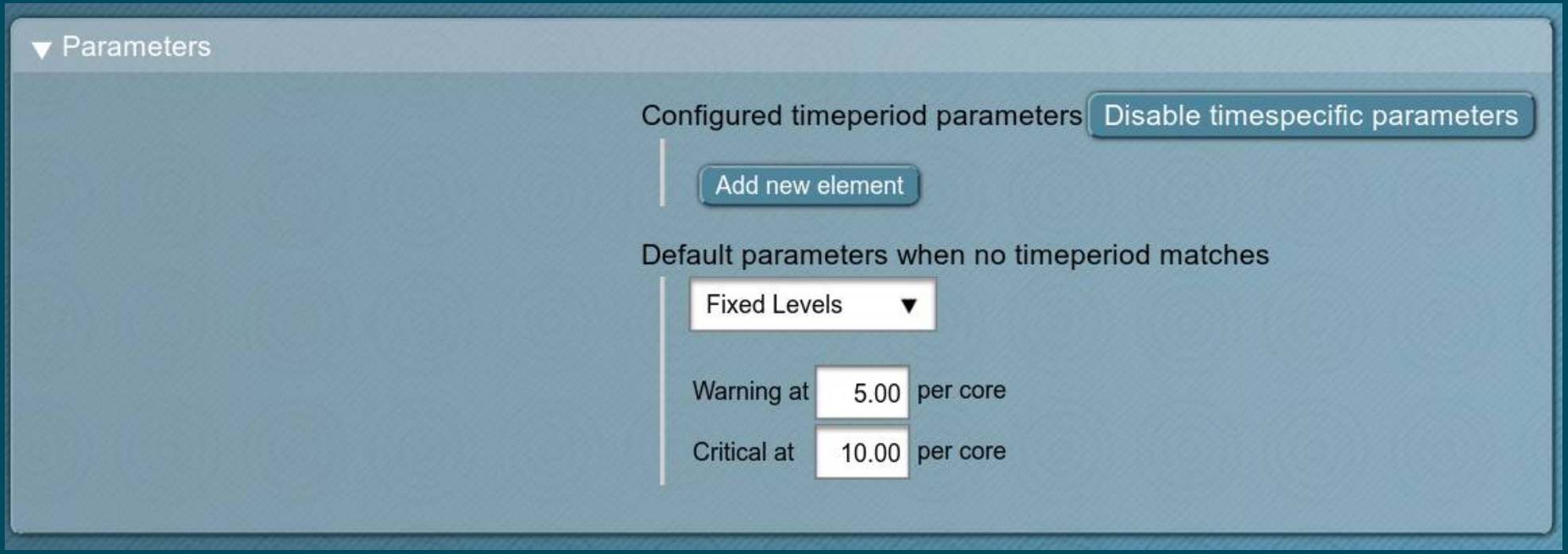


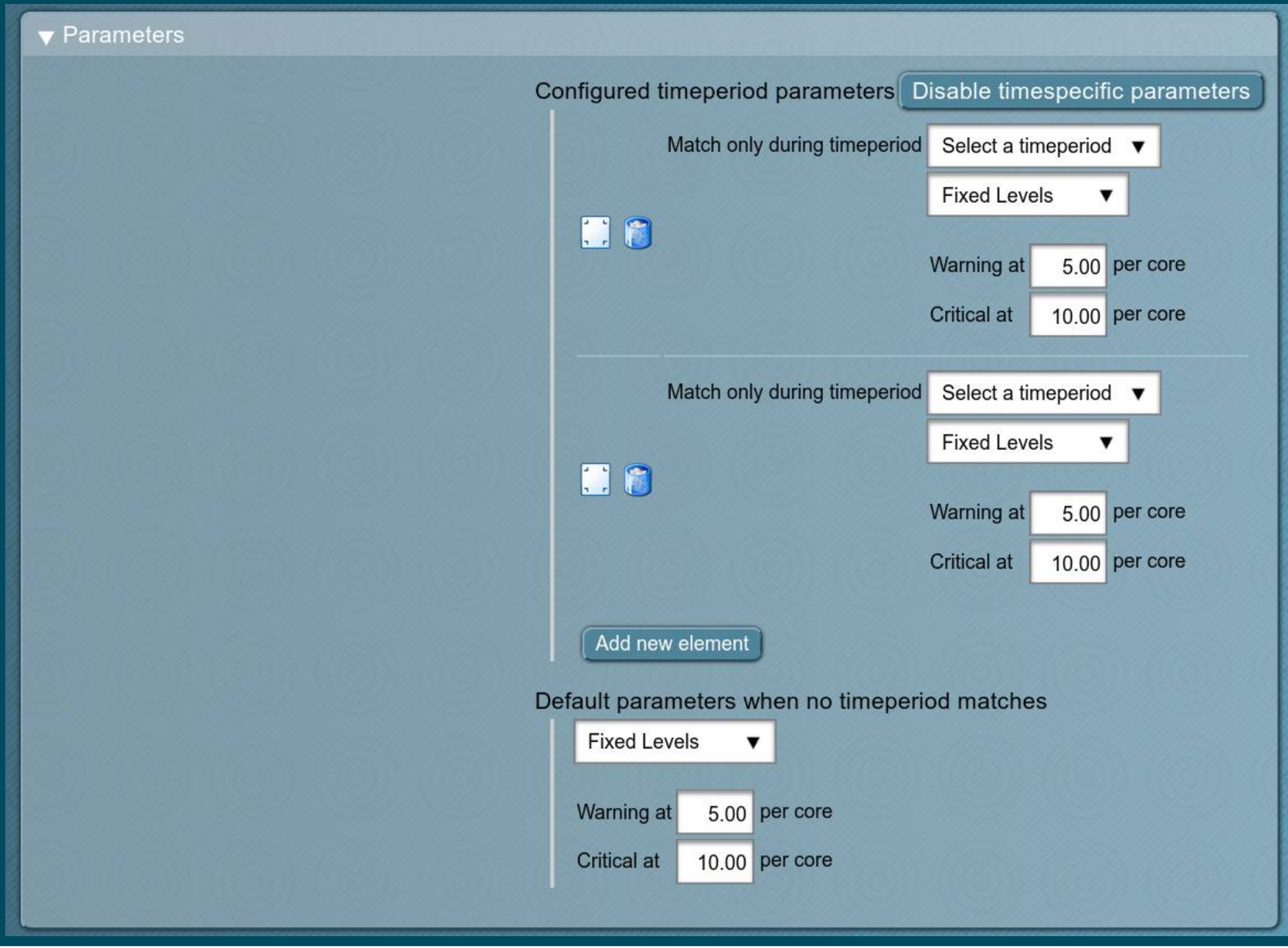
Solution – Timespecific parameters

- Define parameters for time periods
- Generic solution – works for **all** check plugins

How does it work?

How does it work?




Parameters
Configured timeperiod parameters
Disable timespecific parameters
Match only during timeperiod
Select a timeperiod
Fixed Levels
Warning at 5.00 per core
Critical at 10.00 per core
Match only during timeperiod
Select a timeperiod
Fixed Levels
Warning at 5.00 per core
Critical at 10.00 per core
Add new element
Default parameters when no timeperiod matches
Fixed Levels
Warning at 5.00 per core
Critical at 10.00 per core

CHECK_MK

CONFERENCE

MUNICH 2018/5/2-4

#4

Customizable graph layouts

03.05.2018, Konstantin Büttner

Check_MK Conference #4

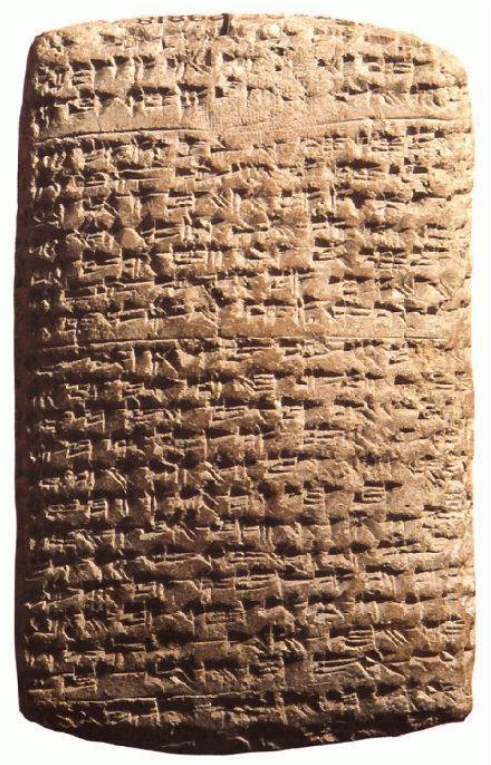
CONFERENCE
MUNICH 2018/5/2-4
#4

Our objective in graphing

- Traditional graphing system are for customizing graphs
- But: day-to-day, we'd rather just **have** graphs
  Tweaking graphs is fun, but time consuming
- Our objective: Make our data useful out of the box

A bit of history - PNP4Nagios

PNP graph



In the Olden Days

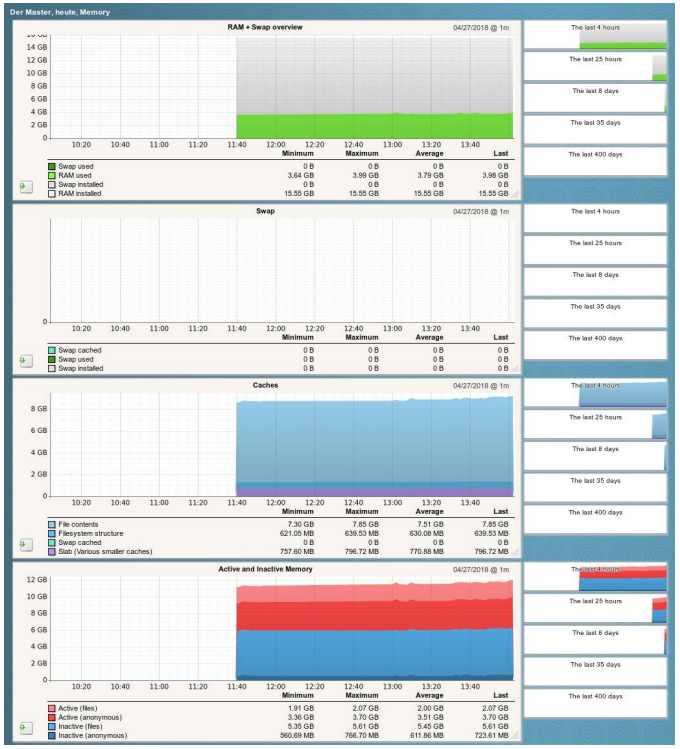
- Graphs were defined for each check type
- Templates in PHP (ノ☉益☉) ノ彡┻━┻
- No semantics, no uniformity*
- No customizability**

*except for checks sharing templates

**unless you wrote your own templates

A bit of history - Metrics

Metrics graph



A new dawn

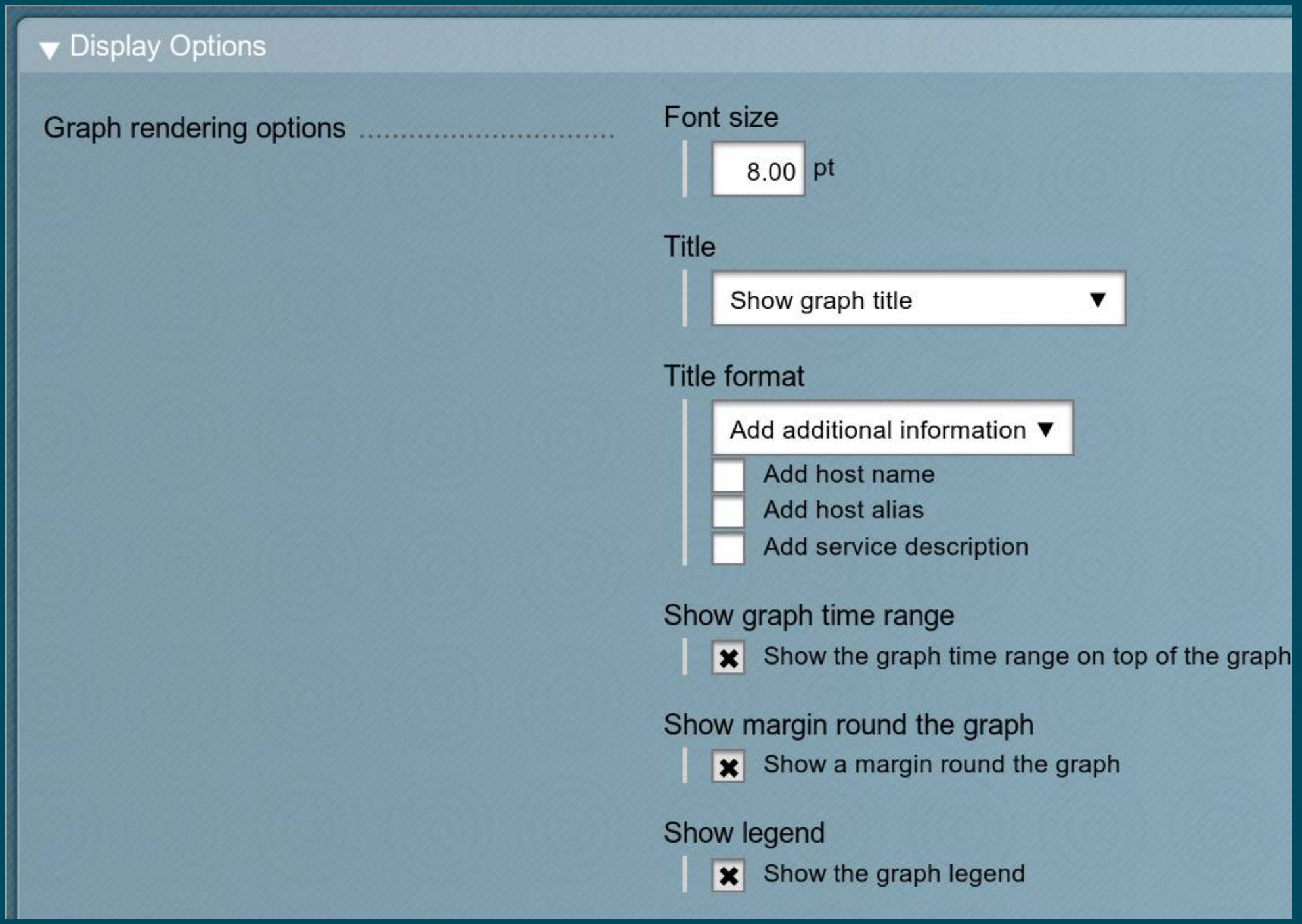
- Most visible: New look
- Most important: Semantics
- Checks no longer have graphs…
- …but values have semantics

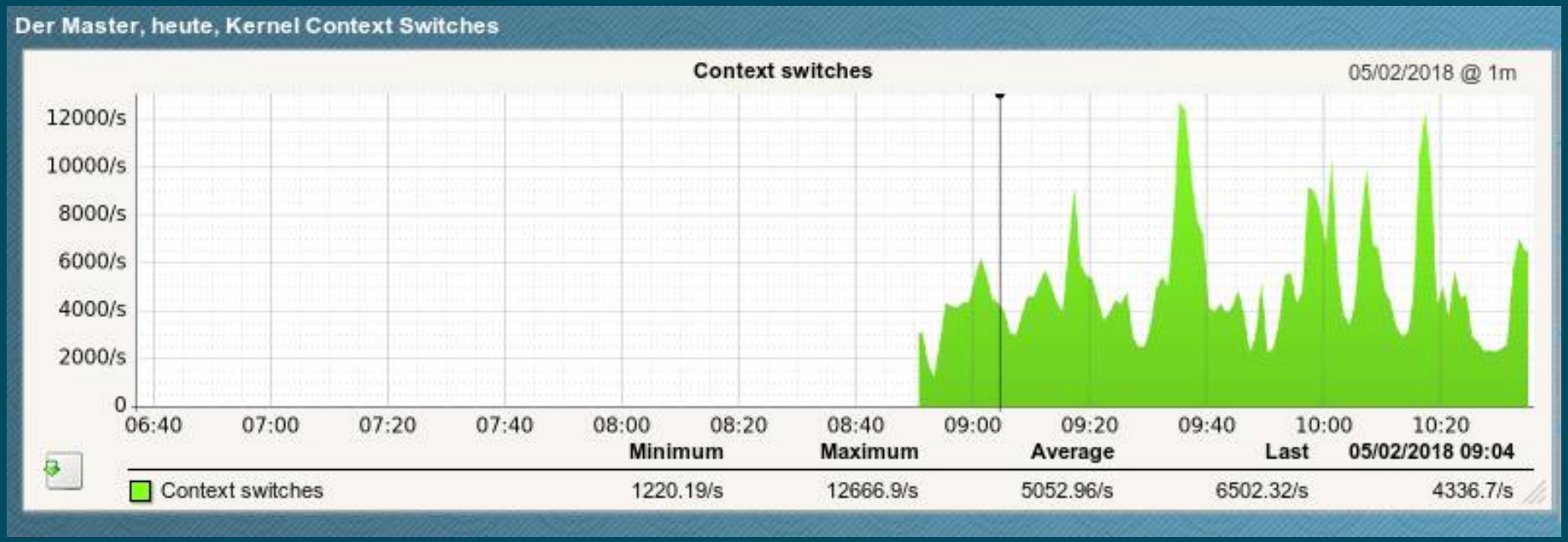
  →Automatic graphs

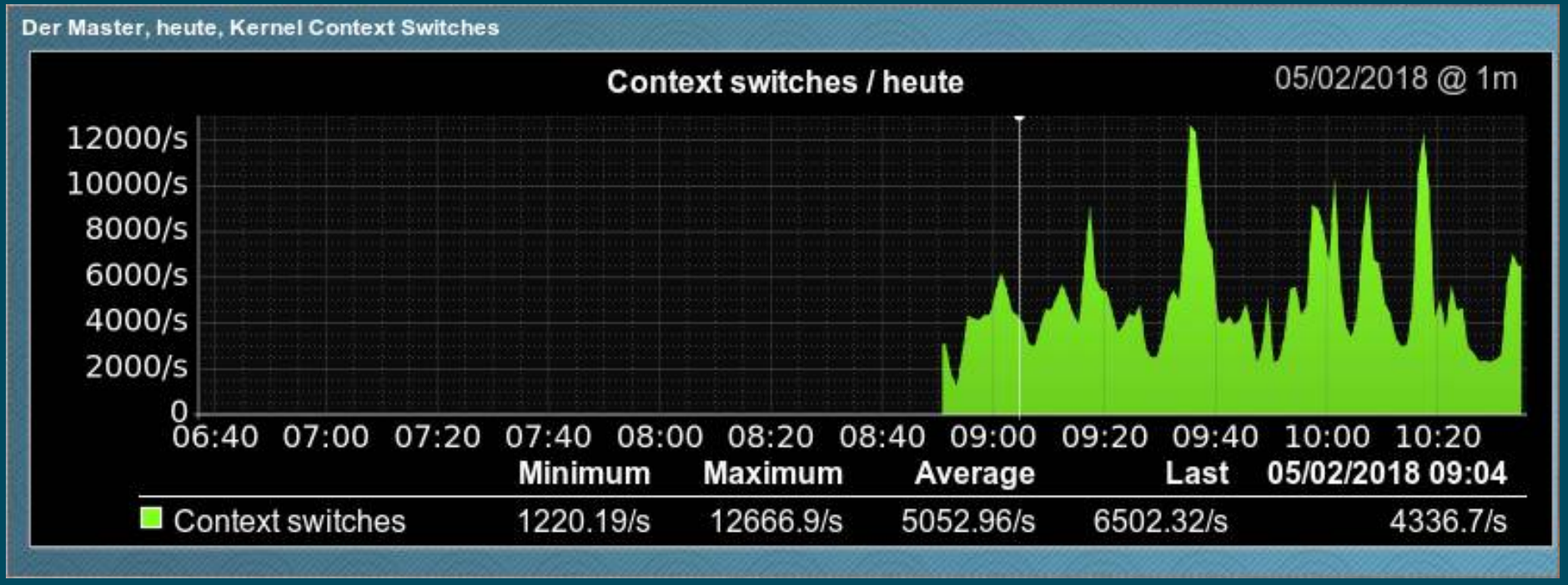
  →Custom graphs

Now: More customizability

- Unified options for dashlets, views, reports (Werk #5569)

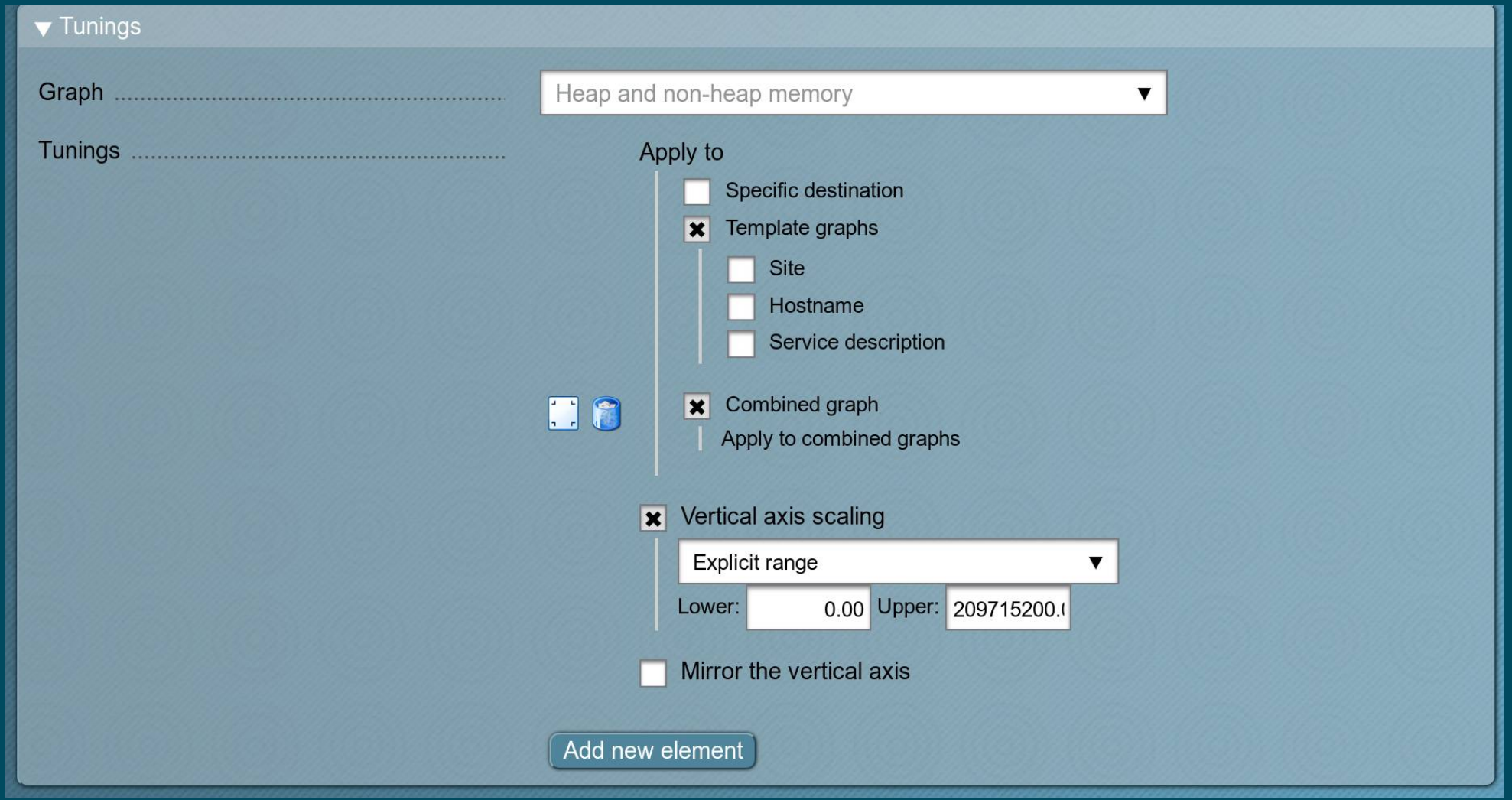

Display Options
Graph rendering options
Font size
8.00
pt
Title
Show graph title
Title format
Add additional information
Add host name
Add host alias
Add service description
Show graph time range
Show the graph time range on top of the graph
Show margin round the graph
Show a margin round the graph
Show legend
Show the graph legend

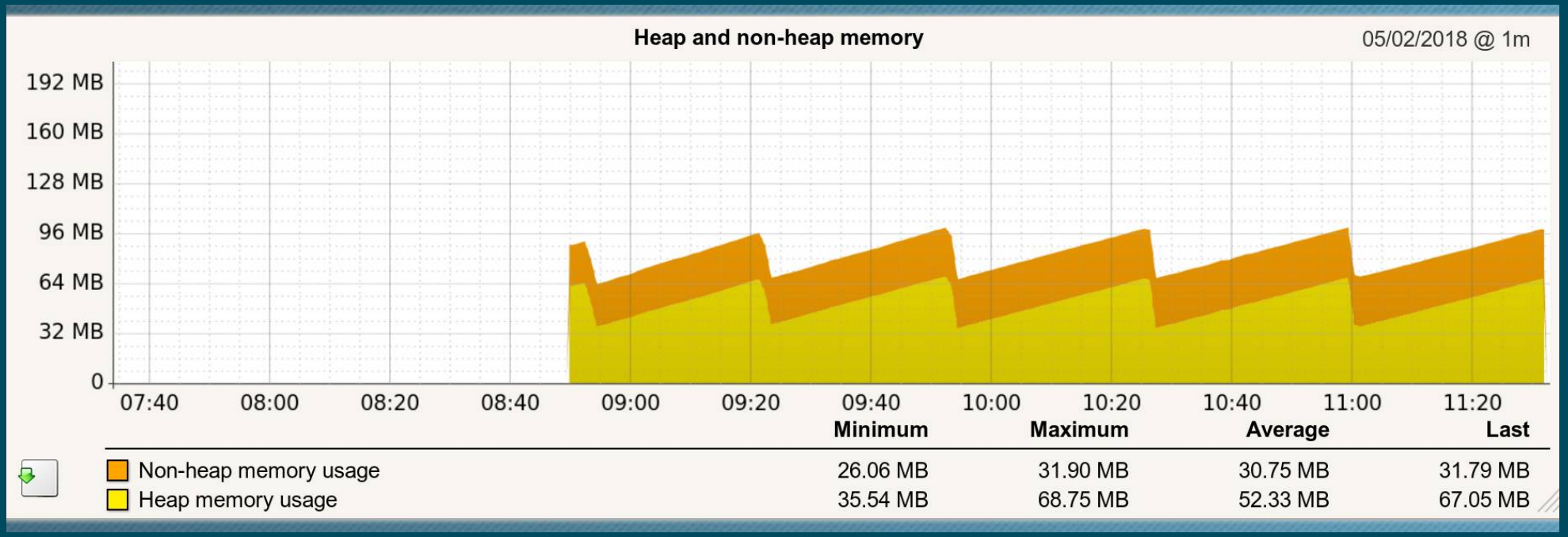

Der Master, heute, Kernel Context Switches
Context switches
05/02/2018 @ 1m
12000/s
10000/s
8000/s
6000/s
4000/s
2000/s
0
06:40
07:00
07:20
07:40
08:00
08:20
08:40
09:00
09:20
09:40
10:00
10:20
Minimum
Maximum
Average
Last
05/02/2018 09:04
Context switches
1220.19/s
12666.9/s
5052.96/s
6502.32/s
4336.7/s


Der Master, heute, Kernel Context Switches
Context switches / heute
05/02/2018 @ 1m
12000/s
10000/s
8000/s
6000/s
4000/s
2000/s
0
06:40
07:00
07:20
07:40
08:00
08:20
08:40
09:00
09:20
09:40
10:00
10:20
Minimum
Maximum
Average
Last
05/02/2018 09:04
Context switches
1220.19/s
12666.9/s
5052.96/s
6502.32/s
4336.7/s

Now: More customizability

- Unified options for dashlets, views, reports (Werk #5569)
- Vertical scaling and mirroring customizable (Werk #3920)


Tunings
Graph
Heap and non-heap memory
Tunings
Apply to
Specific destination
Template graphs
Site
Hostname
Service description
Combined graph
Apply to combined graphs
Vertical axis scaling
Explicit range
Lower:
0.00
Upper:
209715200.(
Mirror the vertical axis
Add new element


Heap and non-heap memory
05/02/2018 @ 1m
192 MB
160 MB
128 MB
96 MB
64 MB
32 MB
0
07:40
08:00
08:20
08:40
09:00
09:20
09:40
10:00
10:20
10:40
11:00
11:20
Minimum
Maximum
Average
Last
Non-heap memory usage
26.06 MB
31.90 MB
30.75 MB
31.79 MB
Heap memory usage
35.54 MB
68.75 MB
52.33 MB
67.05 MB

CHECK_MK

CONFERENCE

MUNICH 2018/5/2-4

#4

Performance improvements

03.05.2018, Konstantin Büttner
Check_MK Conference #4

CONFERENCE
MUNICH 2018/5/2-4
#4

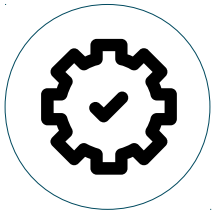
Why is a performance focus critical?

- The typical Check_MK setup keeps growing – more services, more sites, more users

- Better scalability makes new things possible

How to make things go faster

Avoid **unnecessary work**



Optimize

Make things **feel faster**

Avoid unnecessary work - Tabs

- A typical browser: 50 open tabs
- You don’t use them all… but they all make regular livestatus queries, though
- New: Only update active tabs/windows (Werk #4753)
- Benefit: No more wasted data transfers

Avoid unnecessary work – Emails

- Graphs can make emails reasonably large
- Multiply that by the number of recipients …
- What's a mail server for? (Werk #4813)
- Benefit: Notification system becomes more scalable

Avoid unnecessary work – Sites

- Frequent scenario: multi-site setup
- For many users, only few sites are actually relevant
- Yet, GUI in the past contacted all sites for every user
- No more: Now configureable for each user
  (Werk #4921)

Optimize – BI

- Large aggregations now compile faster (Werk #5142)
- Magic - “it’s the algorithm, stupid”
- Further reworking already planned

Optimize – liveproxyd

- Previously: `# The main loop of the daemon goes here`
- One python process – GIL limits scalability
- Now what?

Optimize – liveproxyd

- Now: One process per site, master process to manage
- Process view:

```
OMD[heute]:~$ ps -ef | grep liveproxyd
UID        PID  PPID  C STIME TTY          TIME CMD
heute     9261     1  0 11:40 ?        00:00:00 liveproxyd[master]
heute     9262  9261  0 11:40 ?        00:00:00 liveproxyd[heute_slave_1]
heute     9263  9261  0 11:40 ?        00:00:00 liveproxyd[heute_slave_2]
```

- More resilient, more performant through use of multiple CPUs (cf. Werk #4901)

Make things feel faster - Graphs

- Previously: View with many Graphs? Better wait for **all** the data

- Now: Load the view with placeholders, update graphs asynchronously

Take-home message

Attention to detail pays off

Continuously improving

If you have lots of sites and lots of users, Werk #4921 may help you

CHECK_MK

CONFERENCE

MUNICH 2018/5/2-4

#4