Check_MK Roadmap

Stuff we already work on

Check_MK Package Manager

Package Manager

Current way to work with MKPs:

```
OMD[mysite]:~ $ mkp install foo-1.2.mkp
```

Future way: all operations via WATO

- install, remove, list, show details
- create packages!
- Maybe even:
  access to **Check_MK Exchange**!

Automatic Agent Updates

Automatic Agent Updates

- Agent Bakery bakes agents
- Admin signs und publishes them
- Agents poll for updates...
- ...check the signature....
- ...and update themselves
- Will be available for Linux and Windows

And finally...

Hell...

has **frozen over!**


WHAM!

Check_MK supports

IP v6 !!!

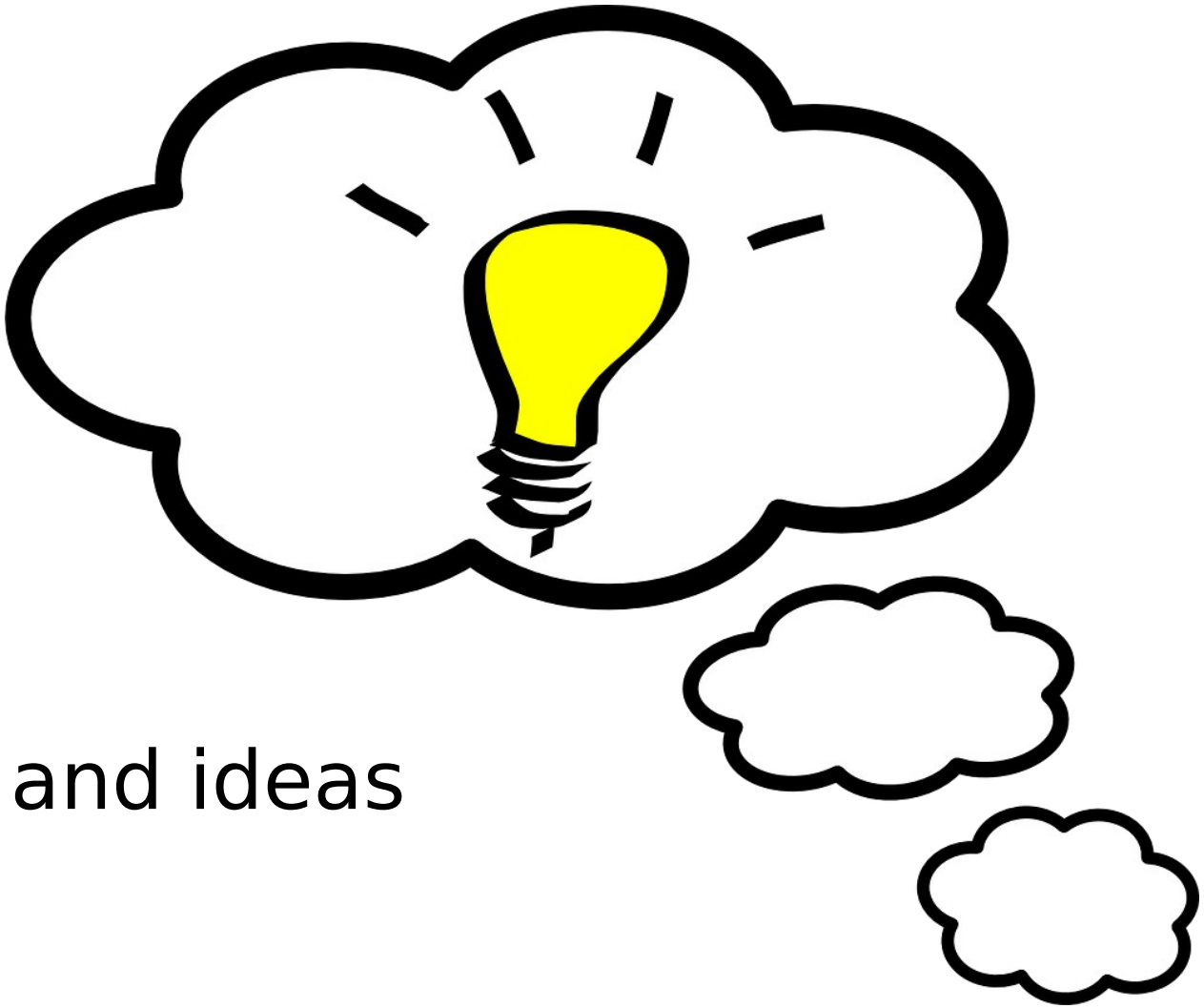
IPv6

- Hosts can have v4, v6 or both addresses
- Real dual-stack monitoring
- „Primary address“:
  - is used for accessing the agent
- „Secondary address“:
  - is monitored by extra PING service
- Works with Check_MK-Agent, SNMP, PING and some active checks

One second resultion for metrics

One-second RRD updates

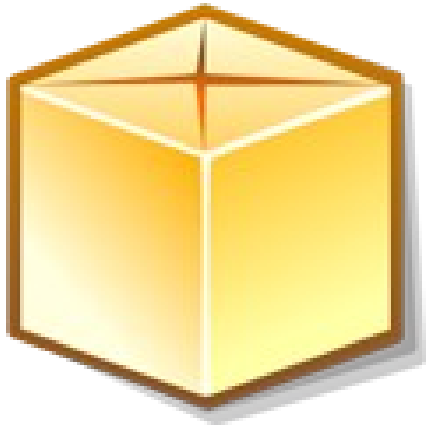
- Linux and Windows agent get new plugin
- active updates of **some** crucial metrics:
  - CPU load, utilization
  - Windows performance counters
  - etc.
- Updates are done by agent via UDP
- New daemon on Check_MK receives these...
- ...and updates RRDs of **existing** services



Plans and ideas for...

...the **future!**

Configuration in MKPs



Configuration in MKPs

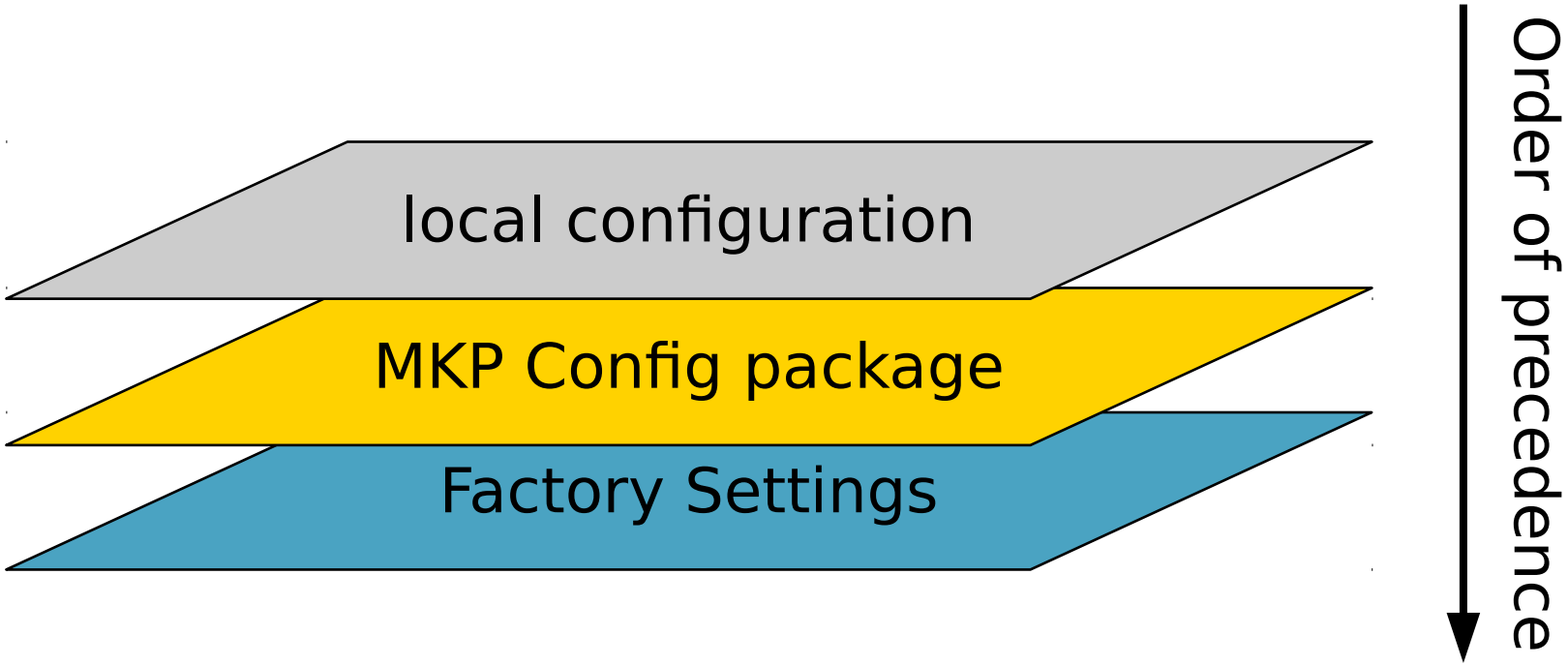
Imagine you could package:

- Event Console rule packs
- Collections of active checks
- Tuned global settings
- Predefined host tag groups

→ **A monitoring template**

Configuration in MKPs

Organized in layers:



→ Configuration MKPs could be updated without damage in the local configuration


• New style of central configuration
• Without central Multisite!
• Interesting for loosely connected sites
• or very large setups

Ultrafast™ config generation

Fast Config Generation

- `cmk -0` or „Activate Changes“ can take a long time...
- Especially when you monitor **many** hosts
- Reason:
  - Configuration for CMC needs to be created
  - This is one big file

Possible solution:

- Split up into one file per folder
- After changes to a host just update that file
- ... and do this **immediately**
- shouldn't take **too** long
- → „Activate Changes“ would take **no** time

Setup with Distributed WATO

- After saving a change immediately replicate to according remote site

Global configuration changes

- Assumed to be < 5% of the cases
- Fall back to current behaviour

Managed Services Edition
CME

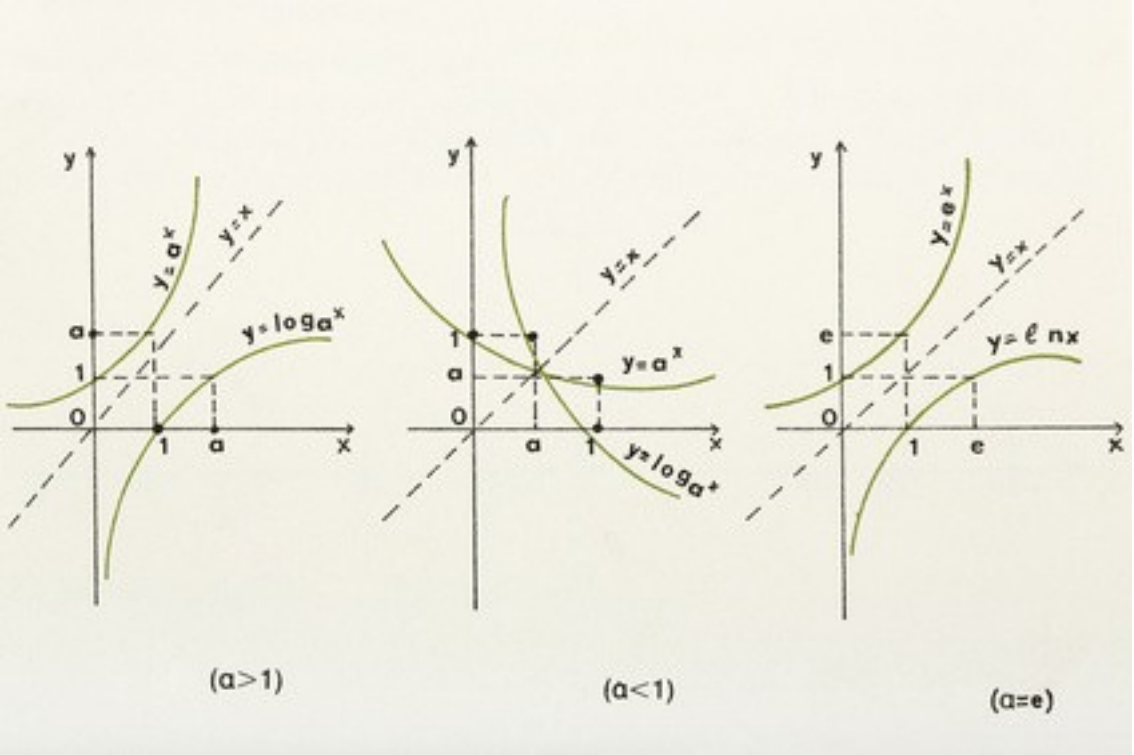
Managed Services Edition

Special Check_MK Edition for users that monitor in behalf of their customers

Features:

- Manage customers (German: Mandanten)
- Assign hosts, users, sites to customers
- Upload your own logo for the GUI
- Adjusted license and pricing

More flexible graphs



Current situation:

- graph templates are hard coded
- and can only show one service

Plan:

- interactive graph editor
- graphs spanning multiple services

More flexible graphs

Step 1

Interactive editing of graph templates

- Add/remove metrics from service
- Select MAX / MIN / AVERAGE
- Layout-style: LINE / AREA / STACK
- Computations (used = total - free)
- Derived scalars (90% percentile)

Customization similar to views:

- Each user can do own customizations
- Admin users can publish their templates

Step 2

Freeform graphs

- Use metrics from **any** host or service
- These are kind of **global** graphs
- Can be put into reports or dashboards

Step 3

Multi-Service-Graphs

1. Select multiple **similar** services
   (e.g. HTTP checks of hosts from a pool)
2. Klick on **Graphs**
3. Get all curves in **one** graph

Multi-Service-Graphs

Possible operations:

- Sum
- Stack (using different colors)
- Lines
- Average

Multi-Service-Graphs

Usages:

- Which of the servers has the worst response time?
- How grows the **combined** space of all selected file systems?
- Show switch port statistics for port 1+2 (in case of trunking)

Numeric access to historic data

Current situation:

- metric data is stored for years
- only access via time series graphs

Plan:

- derive scalar numbers...
- ... and show these in GUI tables
- ... also available for export via CSV / JSON

Example

- I have 150 ESX hosts
- Which of them are the least in use?

Solution

- Use service „CPU utilization“ from ESX-Monitoring
- Create column „Average over last month“
- Create view and sort by this column

Some Questions to the public

How much would you like to see....

- A Check_MK rack1 with more CPU, Mem, IO
- A small/cheap Check_MK Appliance for setups with many (hundreds or thousands) of sites
- Check_MK virt1 in Amazon-Cloud or similar clouds
- IPv6 Support in the Appliance
- Central Management of Check_MK Appliances
- Netflow support

The End

See you again next year!