DOS AND DON'TS IN ITSI

BSIDES SPL

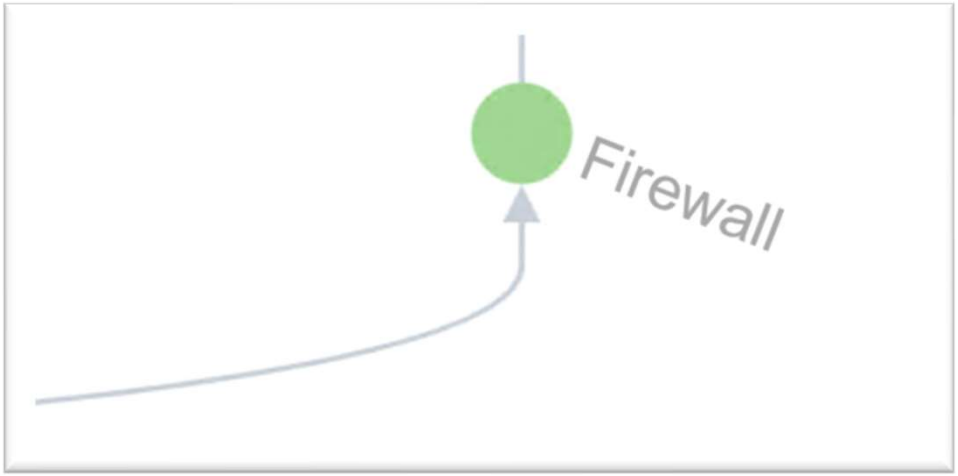

POW!
BASTI
(SEBASTIAN KRAMP)

Many companies have already implemented ITSI and use it for a single pane of glass. But it could become a **resource** hungry monster, hard to handle and give **misinformation** if you don't know how to set it up correctly. I'll talk about some dos and don'ts and also explain what happens if you disregard them. After this step the basic concept is clear and we'll discover how existing systems can *provide* **valuable** data. Out of this data you can construct service dependencies automatically or keep them up to date. As an example we'll take data from a Service Bus, VMware and a production line.

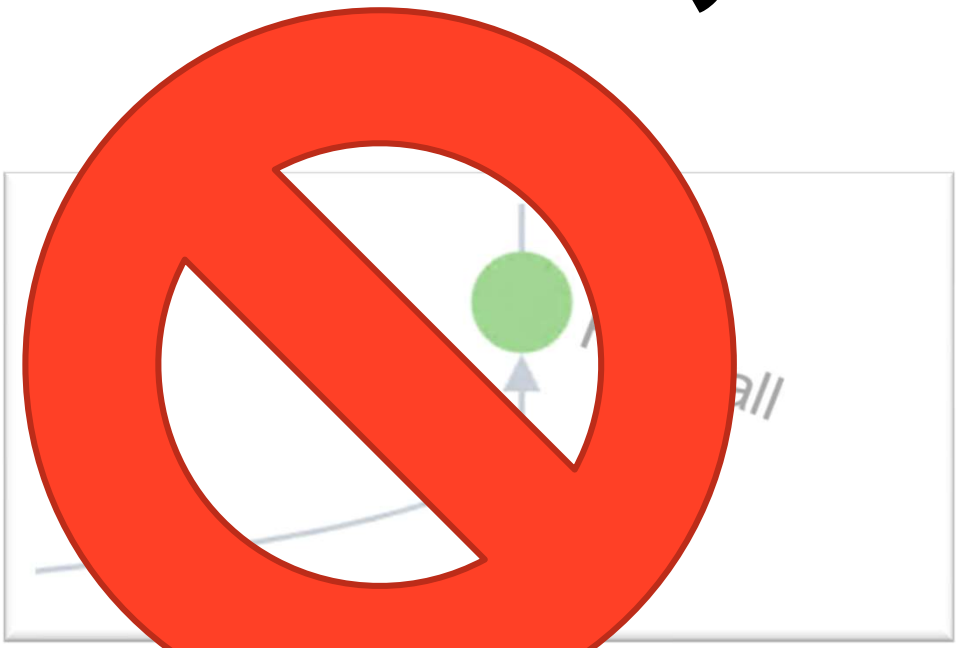
DOS AND DON'TS (IN THE CONCEPT)

Don‘t think too much in technics!

DOS AND DON'TS (IN THE CONCEPT)



DOS AND DON'TS (IN THE CONCEPT)



DOs and DON'Ts (in the concept)

It is a „internet access (service)“ which can be measured i.e. by:

- Relation between blocked and allowed traffic
- Sources outgoing
- Destinations (by devices)

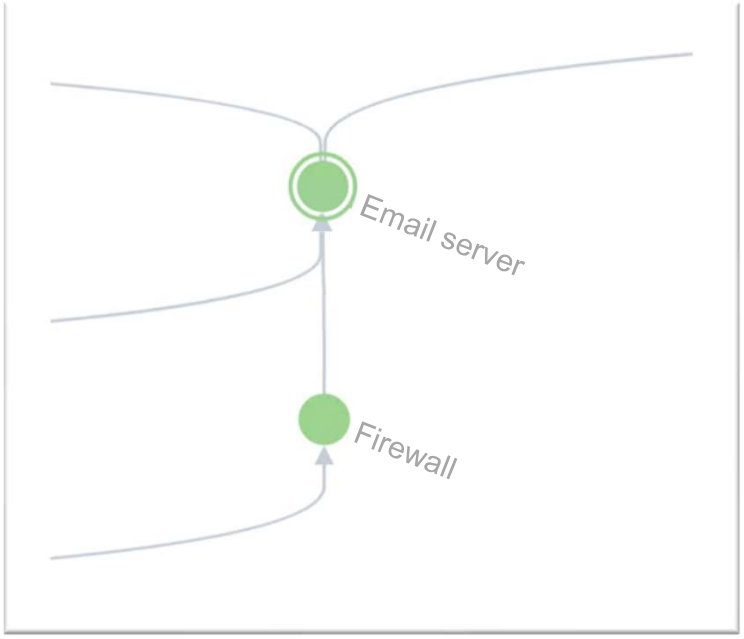
So a backup-system can be added in the same service as an additional entity

DOS AND DON'TS (IN THE CONCEPT)

Don‘t display a wire map!

You get wrong information. If a firewall fails, your mail system isn‘t destroyed

DOS AND DON'TS (IN THE CONCEPT)



DOS AND DON'TS (IN THE CONCEPT)

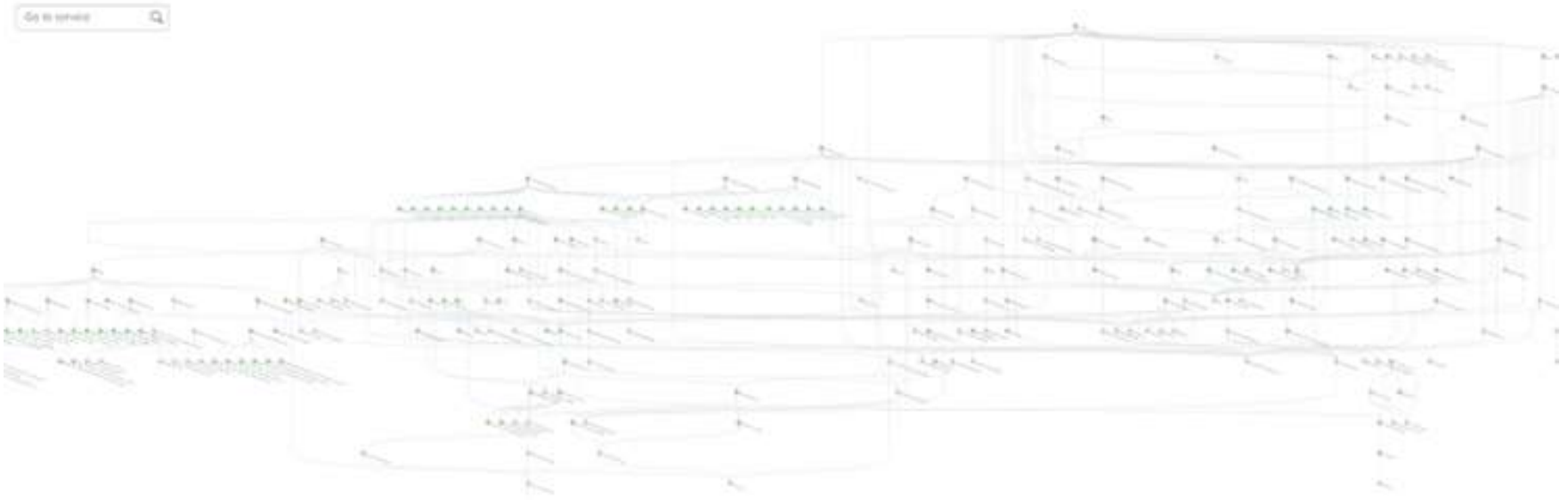
DOS AND DON'TS (IN THE CONCEPT)

The message is wrong! If the firewall fails, your mail service is not destroyed. Yes, it is not accessable by external systems, but the service itself is working fine!

DOS AND DON'TS (IN THE CONCEPT)

Don't try to display everything in one dependency tree

DOS AND DON'TS (IN THE CONCEPT)



DOS AND DON'TS (IN THE CONCEPT)

DOS AND DON'TS (IN THE CONCEPT)

- × You can't figure out root causes vizual
- × You have trouble to bundle dependet Noteables to an Episode
- × You can't work with Teams (Permissions)

OTHERWISE YOUR PROJECT WILL FAIL!

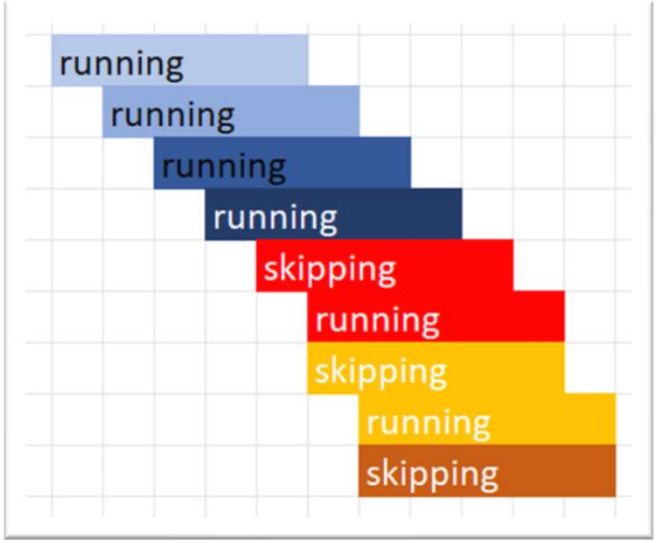
DOs and DON'Ts (in technics)

Don‘t write too long running searches recurring in very small time windows.

Keep in mind, your cpu cores are not limitless!

DOS AND DON'TS (IN TECHNICS)

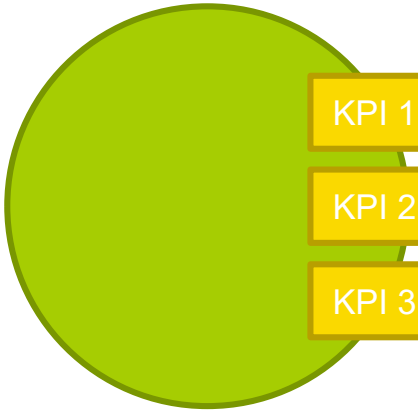
- Every minute
- runs 5 minutes
- 4 CPU cores



DOS AND DON'TS (IN TECHNICS)

Understand the technical concept of ITSI to need less performance

DOS AND DON'TS (IN TECHNICS)

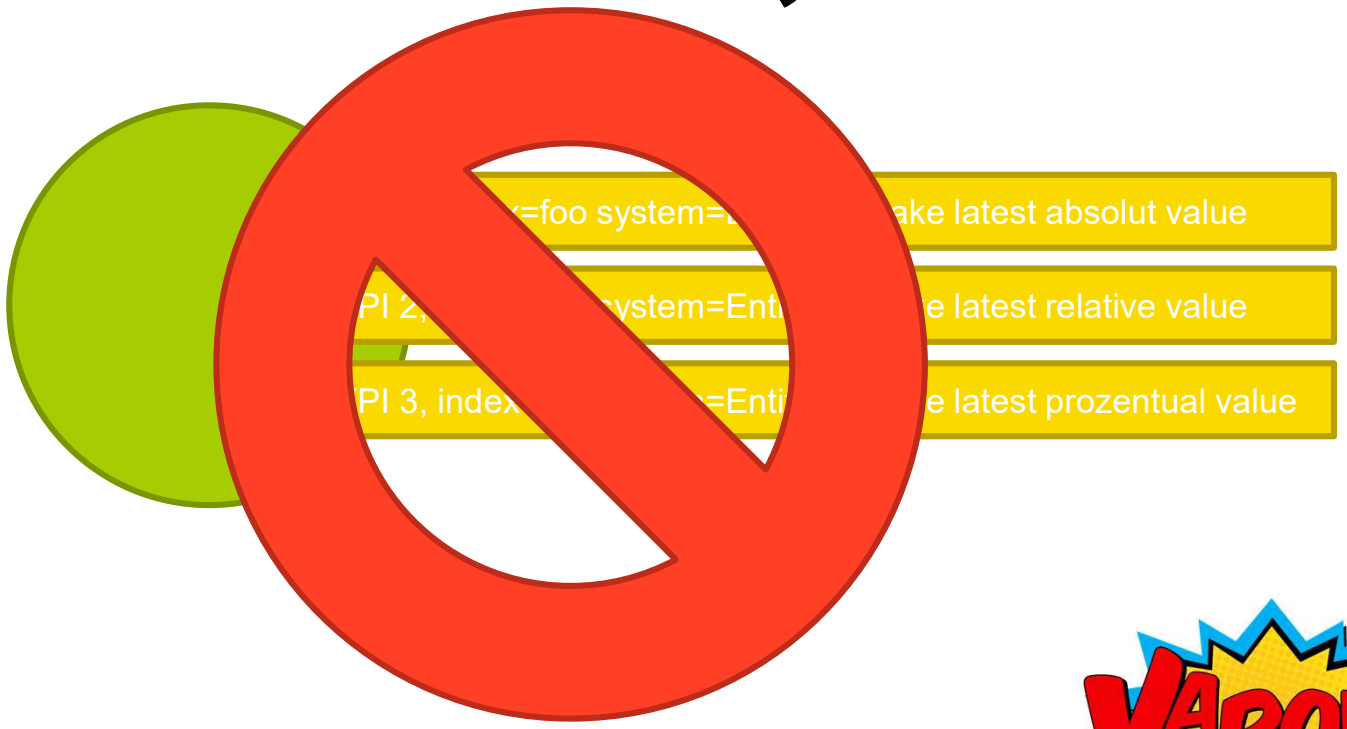


KPI 1, index=foo system=Entity1 -> take latest absolut value

KPI 2, index=foo system=Entity1 -> take latest relative value

KPI 3, index=foo system=Entity1 -> take latest prozentual value

DOS AND DON'TS (IN TECHNICS)



DOS AND DON'TS (IN TECHNICS)

Don‘t try to work without KPI

```
index=foo system=*
                -> take latest absolut value
                -> take latest relative value
                -> take latest prozentual value
```

DOS AND DON'TS (IN TECHNICS)

Don‘t try to work without Service Templates

index=foo system=*
-> take latest absolut value
-> take latest relative value
-> take latest prozentual value

Service Template

KPI1: absolut value
KPI2: relative value
KPI3: prozentual value

DOS AND DON'TS (IN TECHNICS)

Don‘t try to work without Service Templates

index=foo system=*
-> take latest absolut value
-> take latest relative value
-> take latest prozentual value

Service Template

KPI1: absolut value
KPI2: relative value
KPI3: prozentual value

Entity1

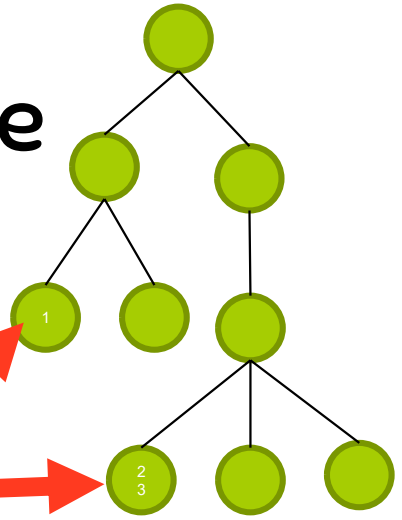
DOS AND DON'TS (IN TECHNICS)

Don‘t try to work without Service Templates

index=foo system=*
-> take latest absolut value
-> take latest relative value
-> take latest prozentual value

Service Template

KPI1: absolut value
KPI2: relative value
KPI3: prozentual value



DOS AND DON'TS (IN TECHNICS)

- Easy to build
- Save performance
- Easy to update

DOS AND DON'TS (IN TECHNICS)

Don‘t try to work without Service Templates

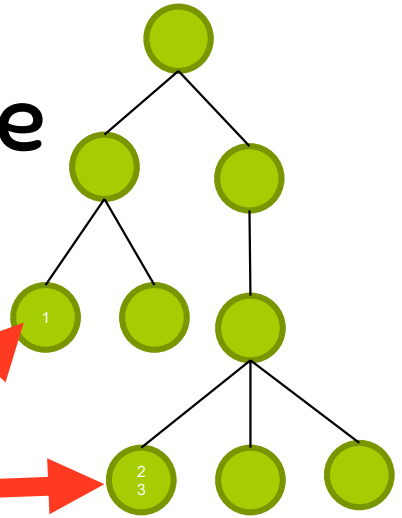
index=foo system=*
-> take latest absolut value
-> take latest relative value
-> take latest prozentual value

index=foo2 system=*
-> take latest absolut value
-> take latest relative value
-> take latest prozentual value

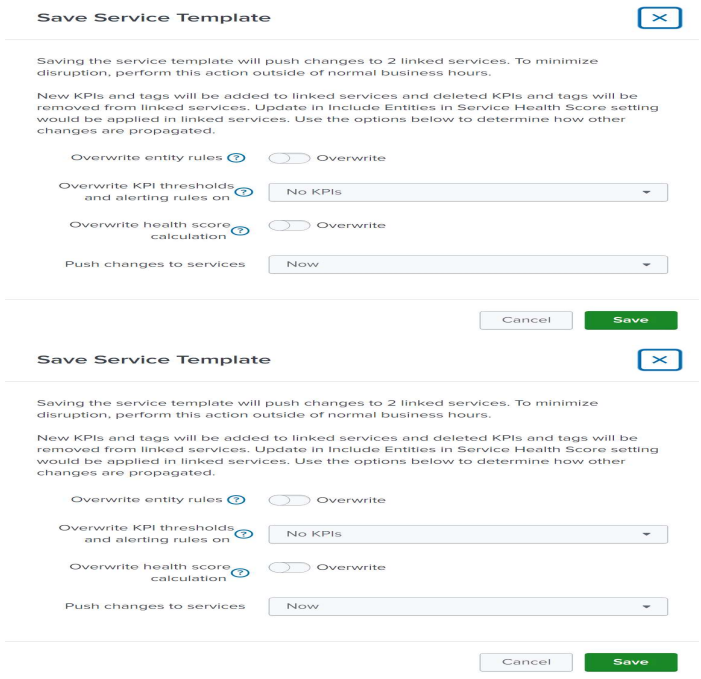
Service Template

KPI1: absolut value
KPI2: relative value
KPI3: prozentual value

KPI 4: …



DOS AND DON'TS (IN TECHNICS)



DOS AND DON'TS (IN TECHNICS)

DOS AND DON'TS (IN TECHNICS)

write searches which

- × just collect data
- × don‘t end with an aggregation when it‘s not needed

DOS AND DON'TS (IN TECHNICS)

BSides example KPI Base Search

KPI Base Search description

Search Properties | Dependent KPIs

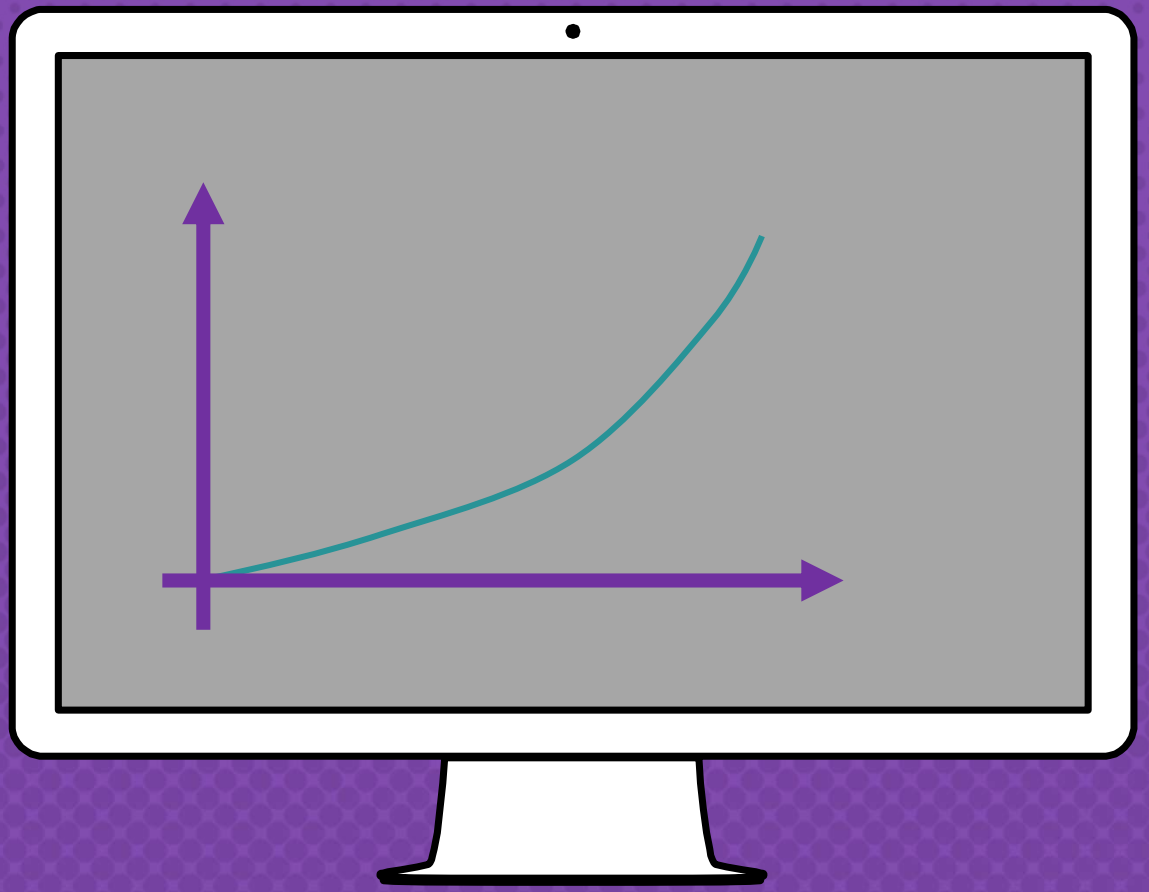
Team (?) **Global**

Search (?) Ad hoc Search | Metrics Search

```
index=bsides sourcetype="mysourcetype"
```

Run Search

Otherwise your workload will be extremely high!

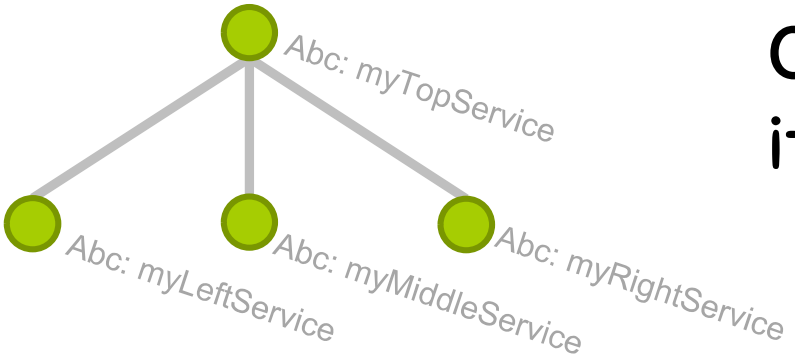


DOs and DON'Ts (in operating)

Think about how to group Notable Events in Episodes

I.e. by a prefix in service names of a service tree

DOS AND DON'TS (IN OPERATING)



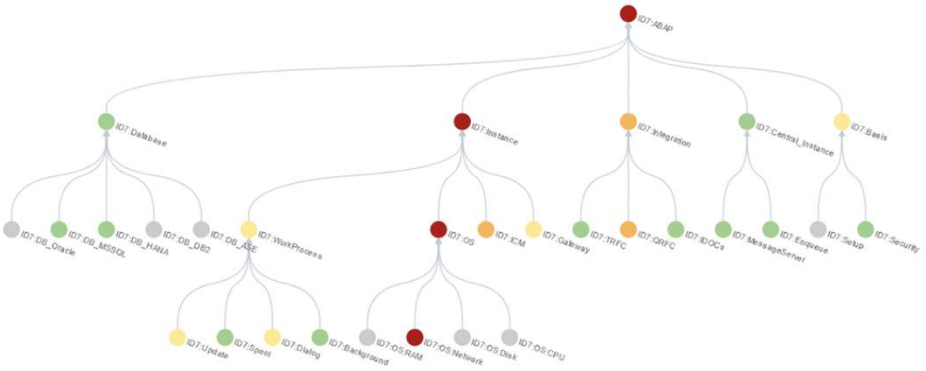
Configure your itsi_kpi_attributes.csv

DOs and DON'Ts (in operating)

Use Prebuild dependency trees i.e.

- out of content packs
- out of specific apps

DOS AND DON'TS (IN OPERATING)



Additional trees will be generated for each SAP system

Automatic Service Tree Generation

Sometimes automatc tree updates are a must have

If services change, the ITSI must be updated automatically

Automatic Service Tree Generation

Out of data, ie

- Virtualization-Service
  - VM, esxi, host, vcenter, storage, management-systems

DOS AND DON'TS (IN OPERATING)

New cluster will be added automatically

OTHERWISE YOUR EVENTS WILL BE OUTDATED AND USELESS

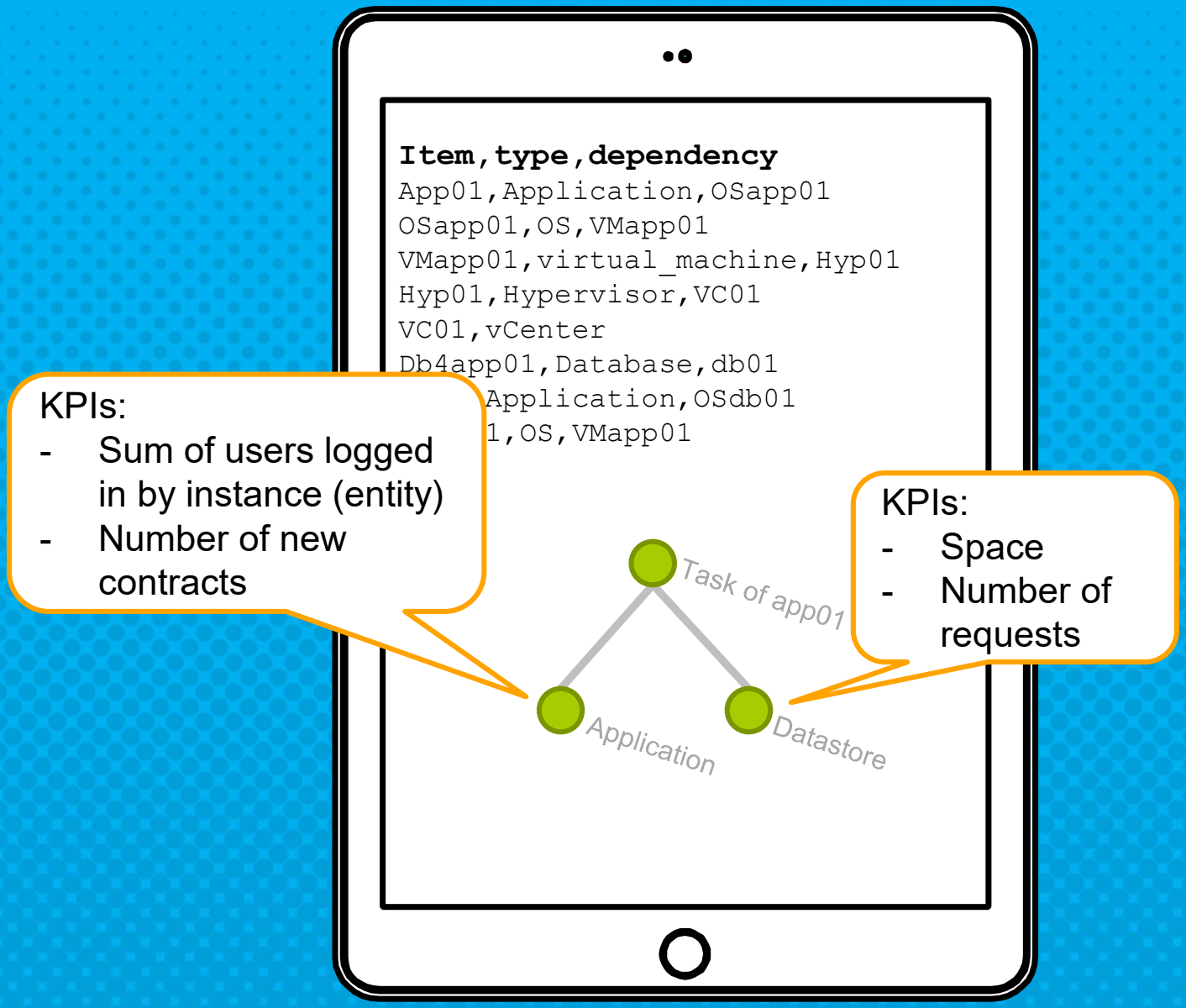
Automatic Service Tree Generation

Out 3rd party tool like a Service Bus or a CMDB

- DB, Hostsystem, VM, ESX, Network

DATA IS AVAILABLE

Must be combined with additional information to get the value

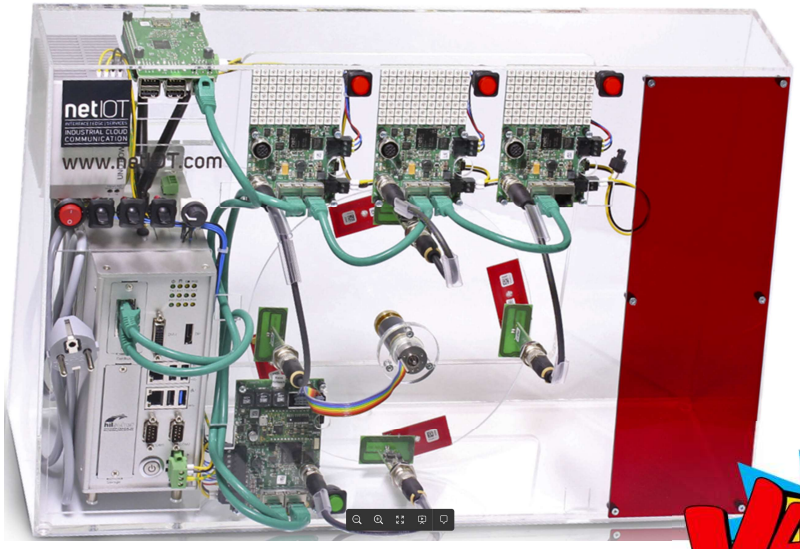


Automatic Service Tree Generation

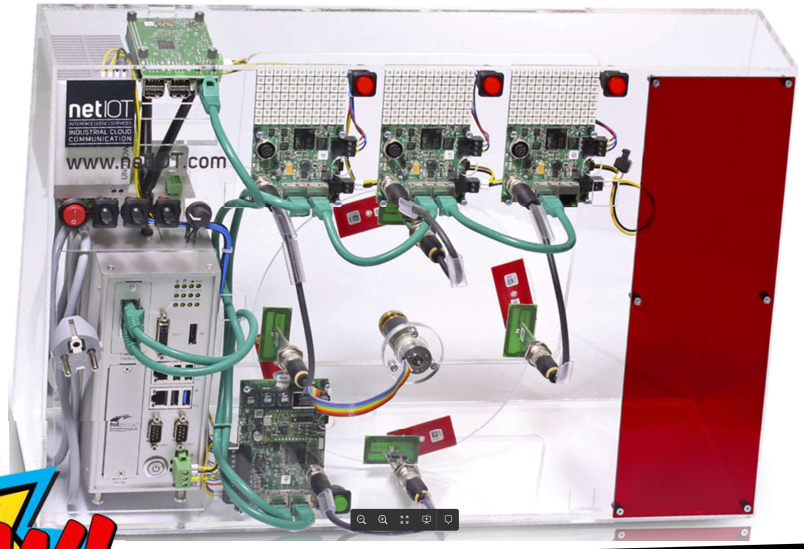
Out of production line data

AUTOMATIC SERVICE TREE GENERATION

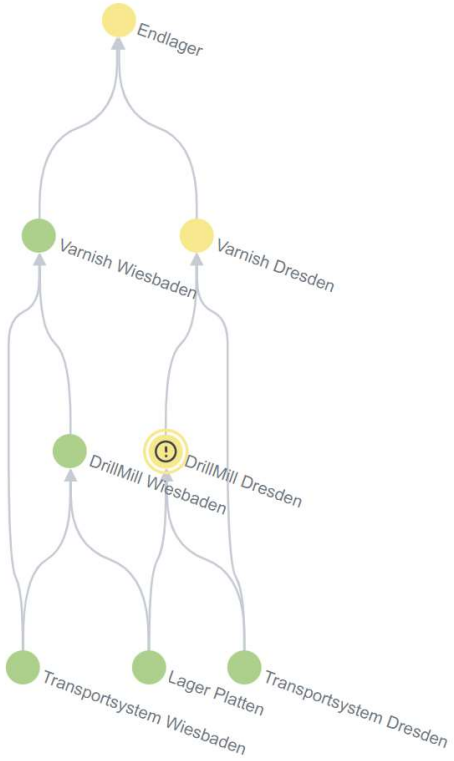
Wiesbaden



Dresden



AUTOMATIC SERVICE TREE GENERATION



DrillMill Dresden

65.1

5 KPIs Open all in Deep Dive

Severity	KPI Name	Value
Critical	workpiece_drill	4
Medium	workload	19.52 %
Low	workpiece_load	4
Low	workpiece_mill	23
Low	workpiece_unload	4

LET'S REVIEW SOME STUFF

Keep load in mind

Use more time and write some small, easy and valueable KPIs. KPIs are running cyclic, so use Base Searches and Entities to reduce load, create templates if your set of KPIs is used in many services.

Stay in the service view

Always think in services, what's useful for the end user who's normally a business guy

Use infos from already existing sources

Don't try to develop everything by yourself, keep it dynamically

THANKS!