

---

# Integration von Nagios und OTRS

Nürnberg, 2008-09-12

# Agenda

---

- OTRS – (ganz) kurze Einführung
- Workflow
- Nagios-Mechanismen
- OTRS-Modul SystemMonitoring (v1.1.1)

# OTRS – Intro (1)

- E-Mail basiertes Ticketsystem
- Mit E-Mails kann man vieles steuern:
  - Queue zuordnen
  - Priorität setzen
  - Ticket-Status ändern (z.B. schließen)
  - zusätzliche Informationen mitgeben
    - > z.B. Host + Servicebezeichnung
- Nagios-OTRS-Interaktion: per E-Mail

# OTRS – Intro (2)

---

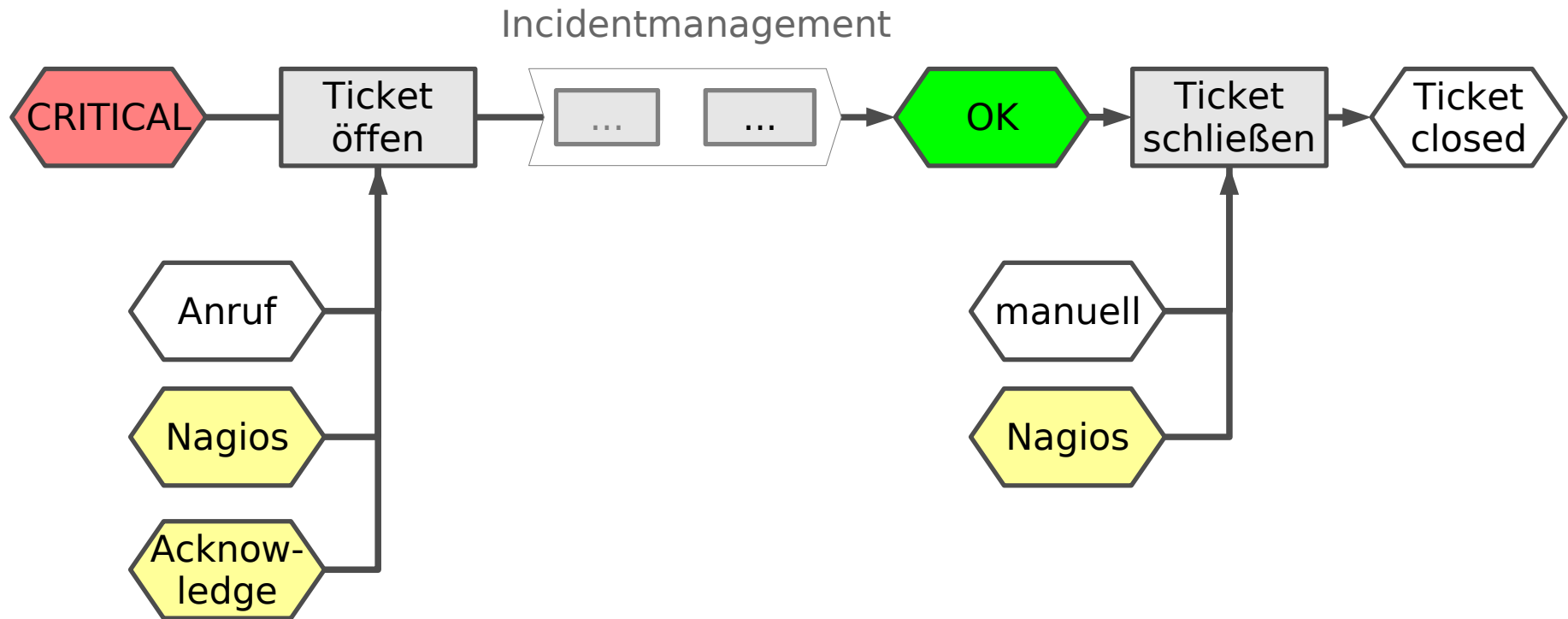
- E-Mail => !Security!
- Perl, Apache, Datenbank
- OpenSource
- ursprünglich entstanden bei SuSE
- Installation:
  - best fit: RPMs für SUSE, RedHat + MySQL
  - (k)ubuntu 8.04: aptitude install otrs2 (2.2.4)
  - Sourcen: tar.gz
- Homepage: [www.otrs.org](http://www.otrs.org)

# OTRS-Demo

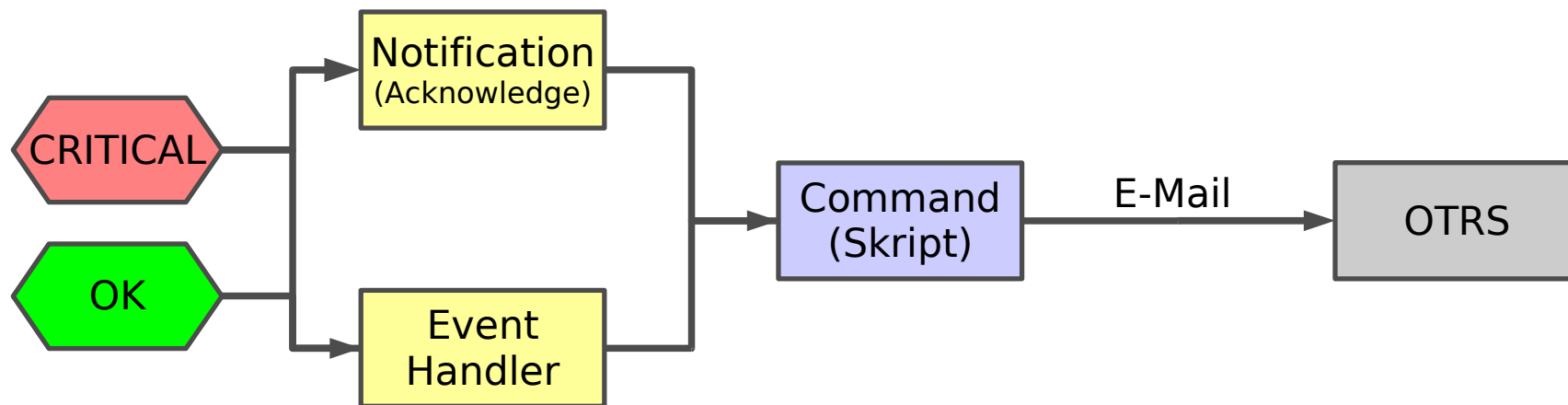
---

- Login
- Queue-Ansicht
- Ticket-Ansicht
- Telefon-Ticket/E-Mail-Ticket
- Administration
  - Paketverwaltung -> SystemMonitoring
  - Systemkonfiguration

# Workflow

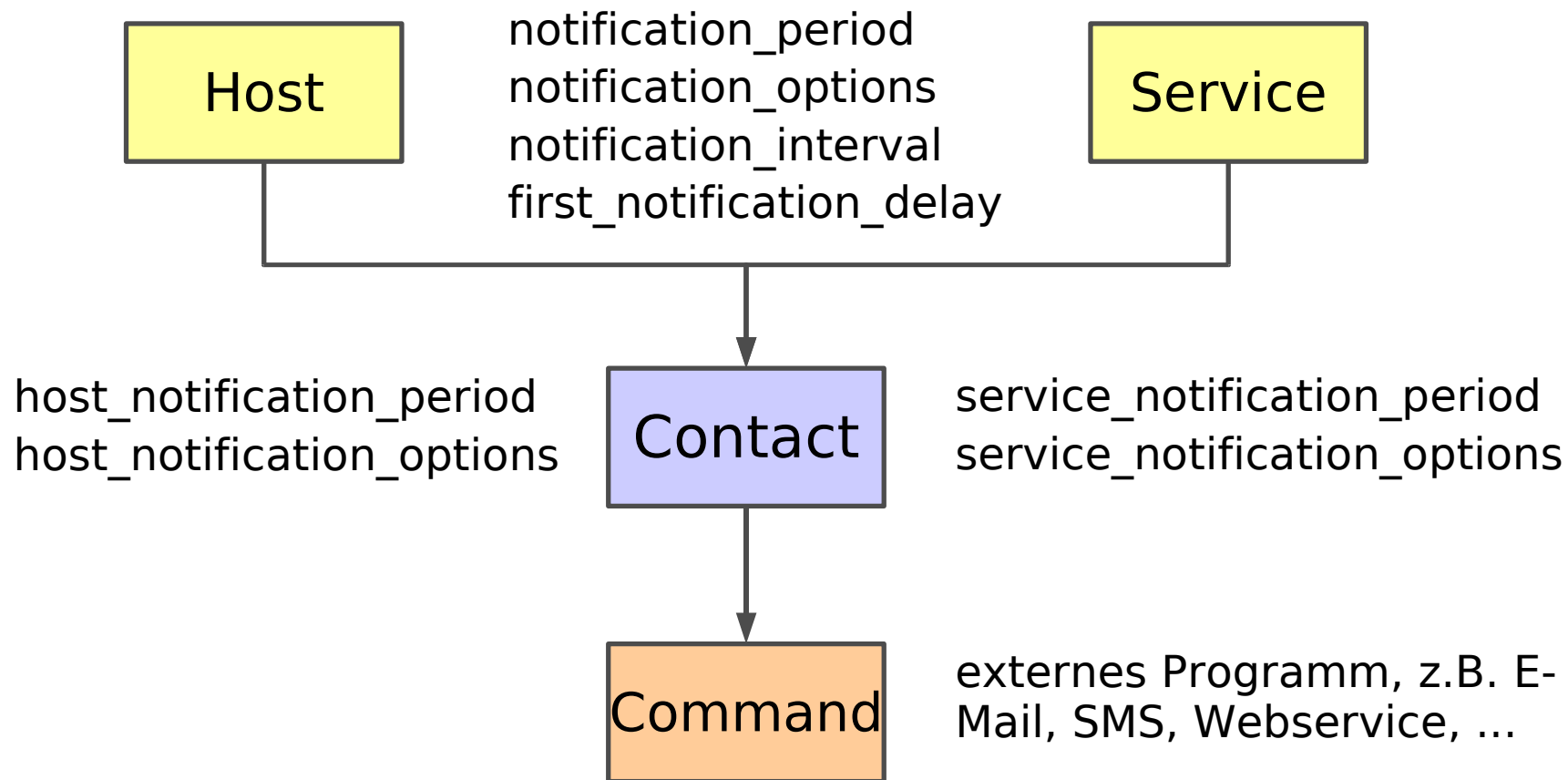


# Nagios ruft OTRS



- Notification -> abhängig vom definierten Kontakt
- Event Handler -> unbedingt

# Notifications: Parameter



# Notifications: Stolperstellen

---

- nur wer eine Problem-Notification bekommt, erhält später eine Recovery-Notification
- Verschiedene Mechanismen unterdrücken Benachrichtigungen:
  - Dependencies
  - Acknowledgements
  - Scheduled Downtime
  - Flapping

# Notification: Ticket erzeugen

- Steuerung über Kontakt
- automatisch:
  - notification\_period = 7x24
    - > kein zeitabhängiges Queuing
    - > Andere Benachrichtigungsarten nicht frei konfigurierbar
- Acknowledgement:
  - Idee: Ticket erst bei ACK erzeugen (nicht bei jedem Problem)
  - kein \_options-Flag für ACK => Skriptabfrage

# Notification: Ticket schließen

- Problem-Nachricht muss versandt worden sein, sonst gibt es keine Recovery-Nachricht:
  - first\_notification\_delay problematisch
  - period muss 7x24h laufen!
  - Filterung Recovery wie Problem State!
- **Notifications sind für Ticket schließen nicht wirklich gut geeignet!**

# Event Handler

- Werden ausgeführt bei:
  - jedem Soft-State
  - neuem Problem-Hard-State
  - jedem Recovery (Soft- und Hard-Recovery)
- Steuerung über:
  - `$SERVICESTATE$` (OK, WARNING, ...)
  - `$SERVICESTATETYPE$` (SOFT, HARD)
  - `$SERVICEATTEMPT$`
  - dito für Hosts
- Finetuning über das aufgerufene Skript (nicht über Nagios-Konfiguration)

# Event Handler: Ticket erzeugen

---

- unabhängig von Benachrichtigungen
- muss bei Host/Service aktiviert werden
  - keine Steuerung über Contact
- evtl. Kollision mit anderen Event-Handler-Aufgaben
  - z.B. Reset eines Dienstes

# Event Handler: Ticket schließen

---

- Information über Recovery kommt immer an
- Ideal für automatisches Schließen

# Ticket manuell in OTRS schließen ?

- grundsätzlich möglich
- **Problem offen**, Ticket schließen:
  - anschließender Wechsel z.B. von WARNING auf CRITICAL -> neues Ticket
  - mehrere Tickets zum gleichen Vorgang müssen manuell zusammengefasst werden (in OTRS möglich)
- **Problem erledigt**, Ticket schließen:
  - manuelle Endkontrolle erforderlich
  - sinnvoll u.U. für bestimmte Services (z.B. Filesystem-Auslastung)

# Menü a la wob

---

- Services explizit auswählen
  - nicht alles immer ins Ticketsystem schicken
- Ticket erzeugen via Acknowledgement
  - damit man das Problem nicht vergisst
- Ticket schließen per Event-Handler
- Einige Service-Tickets nur manuell schließen:
  - "Filesystem full"

# ACK -> Ticket öffnen

```
#!/bin/bash
# notify-otrs.sh
...
case "$NAGIOS_NOTIFICATIONTYPE" in
    ACKNOWLEDGEMENT*)
        # mailtext zusammenbauen
        ...
        printf "%s\n" $body | /usr/bin/mail ...
        ;;
    *)
        exit 0
        ;;
esac
```

# Event -> Ticket schließen

```
#!/bin/bash
# otrs-event-ok.sh
...
case "$NAGIOS_SERVICESTATE" in
  OK)
    case "$NAGIOS_SERVICESTATETYPE" in
      HARD)
        # mailtext zusammenbauen
        printf "%s\n" $body | /usr/bin/mail ...
        ;;
      *) ;;
    esac
  *) ;;
esac
```

**ACHTUNG:**  
Mailadresse  $\neq$  CONTACTEMAIL

# OTRS::SystemMonitoring

[ Paket Verwaltung ]

**Installieren:**

Paket:

**Online Repository:**

Quelle: [-Master-]

**Online Repository:**

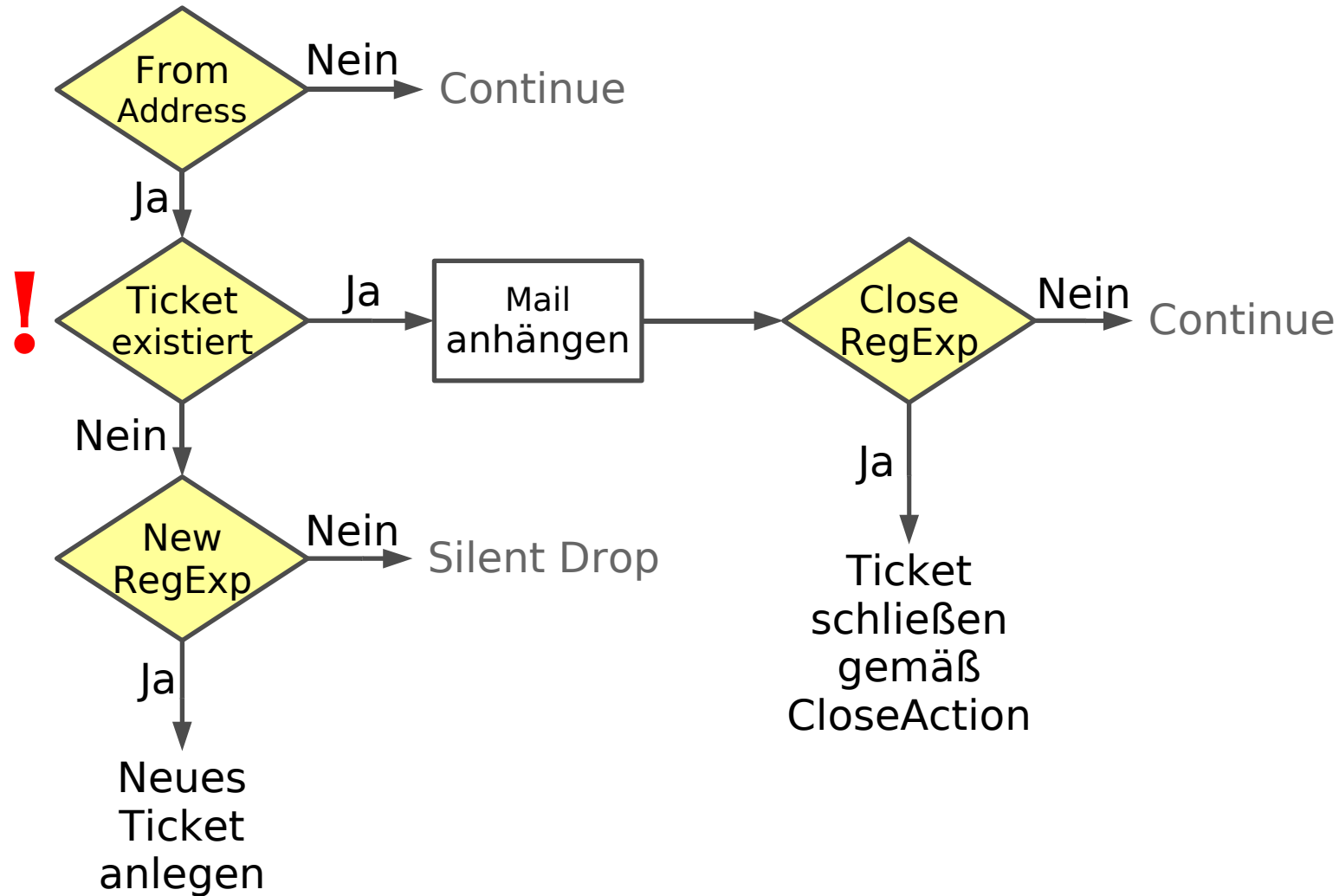
Name	Version	Anbieter	Beschreibung	Aktion
<a href="#">Benchmark</a>	1.2.5	<a href="#">OTRS AG</a>	Ein einfache Benchmark Applikation.	<a href="#">Installieren</a>
<a href="#">Calendar</a>	1.7.2	<a href="#">OTRS GmbH</a>	Ein Web Kalender.	<a href="#">Installieren</a>
<a href="#">FAQ</a>	1.3.1	<a href="#">OTRS AG</a>	Eine FAQ-/Wissensdatenbank.	<a href="#">Installieren</a>
<a href="#">FileManager</a>	1.3.1	<a href="#">OTRS GmbH</a>	Ein Web Datei Manager mit download/upload moeglichkeit[..]	<a href="#">Installieren</a>
<a href="#">Survey</a>	1.0.17	<a href="#">OTRS GmbH</a>	Ein Kundenumfragen Werkzeug.	<a href="#">Installieren</a>
<a href="#">WebMail</a>	0.11.3	<a href="#">OTRS AG</a>	Ein Web Email (IMAP) Programm.	<a href="#">Installieren</a>

**Lokales Repository:**

Name	Version	Anbieter	Beschreibung	Status	Aktion
<a href="#">LinkObject2</a>	1.0.5	<a href="#">OTRS AG</a>	Das LinkObject2 Paket.	installiert	<a href="#">Deinstallieren</a>
<a href="#">SystemMonitoring</a>	1.0.2	<a href="#">OTRS GmbH</a>	Einfache Email Schnittstelle zur System Monitoring Suit[..]	installiert	<a href="#">Deinstallieren</a>
<a href="#">SystemStatus</a>	1.5.1	<a href="#">OTRS GmbH</a>	Ein Web System-Status-Monitor.	installiert	<a href="#">Deinstallieren</a>

- ftp-Download: ftp://ftp.otrs.org/pub/otrs/packages
- lokale Installation oder via Online-Repository

# OTRS::SystemMonitoring Logik



# OTRS::SystemMonitoring Konfiguration

Config Einstellungen: SystemMonitoring -> Core::PostMaster



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

- ArticleType
- FreeTextHost
- FreeTextService
- Regexp aus E-Mail:
  - Host, Service
  - State
  - FromAddress
- Regexp aus State:
  - NewTicket, CloseTicket

Schlüssel	Inhalt	
ArticleType	email-external	Löschen
CloseActionState	closed successful	Löschen
ClosePendingTime	172800	Löschen
CloseTicketRegExp	OK UP	Löschen
DefaultService	Host	Löschen
FreeTextHost	6	Löschen
FreeTextService	7	Löschen
FromAddressRegExp	nagios@(elix01.eli mhkna	Löschen
HostRegExp	s*Host: s+(\ S+)\ s*	Löschen
Module	Kernel::System::PostMast	Löschen
NewTicketRegExp	CRITICAL DOWN WARNI	Löschen
SenderType	system	Löschen
ServiceRegExp	s*Service: s+(\ .\ )\ s*	Löschen
StateRegExp	s*State: s+(\ S+)\ s*	Löschen
Neu		

# OTRS::PostMaster

- Filter für Queue-Zuordnung:

Bearbeiten:

Filtername:

Treffer:

Kopf 1:  Wert:

Kopf 2:

Wert:

Kopf 3:

Wert:

Kopf 4:

Wert:

Setzen:

Kopf 1:  Wert:

Kopf 2:

Wert:

Kopf 3:

Wert:

Kopf 4:

Wert:

Übermitteln

---

# Live Demo

**(falls die Zeit noch reicht ;-)**

