

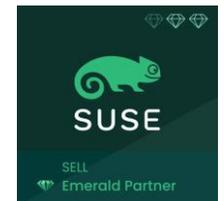
System Monitoring vs. Observability

Christian Anton, Mai 2025

Wer sind wir?

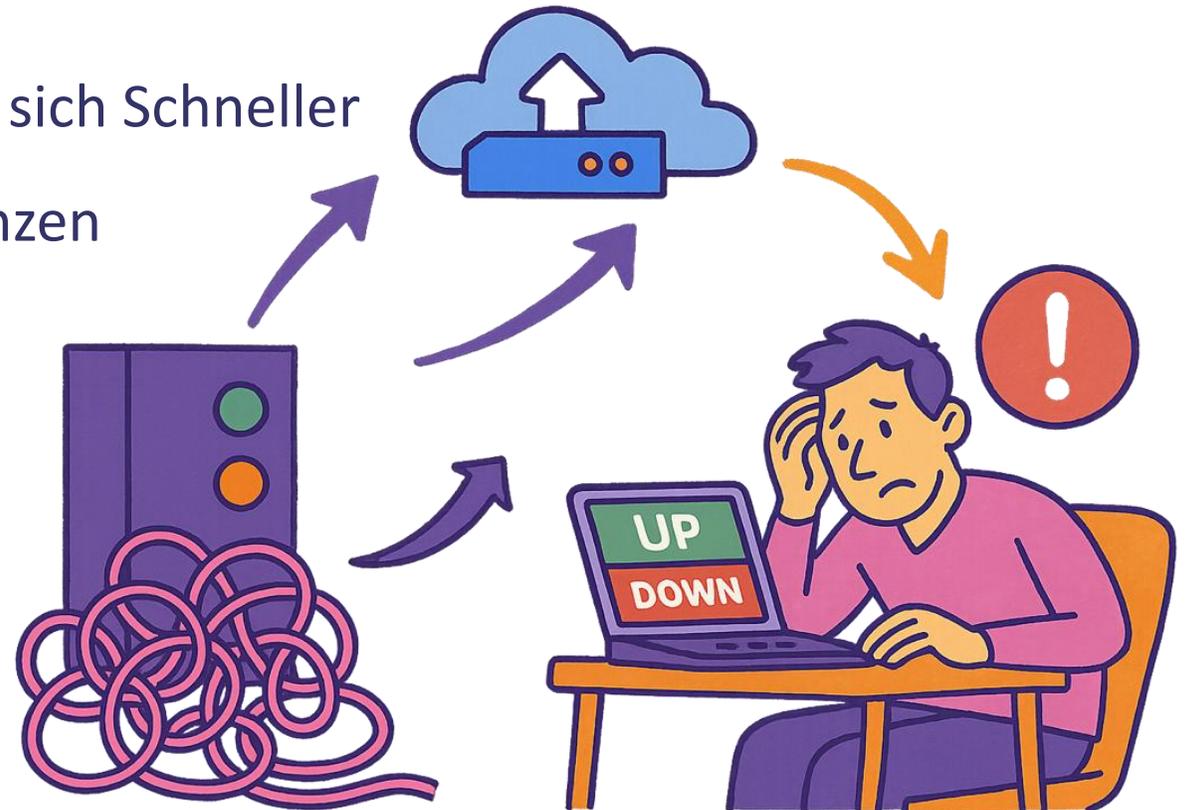
Inqbeo ist ein spezialisiertes IT Consulting Unternehmen mit Fokus auf Automatisierung, Monitoring und Open Source.

- Consulting / Projekte
- Konzipierung und Infrastruktur Design
- Architekturberatung
- Support & Services
- Entwicklung von Microservices / Integrationen
- Schulungen, Workshops & Knowledge Transfer
- Containerisierung v. Legacy Anwendungen
- Kubernetes Plattformen u. Container Orchestration

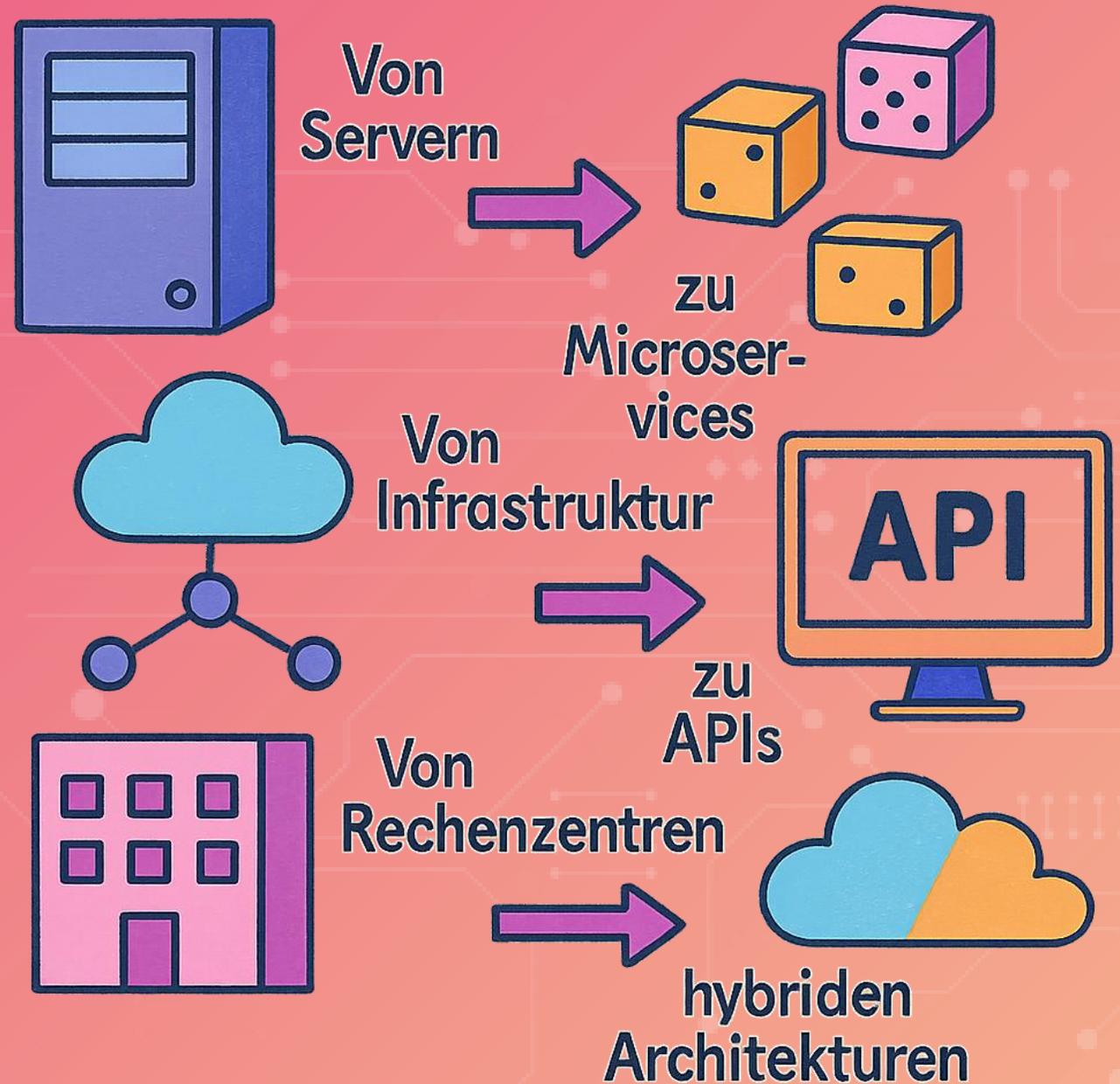


Warum überhaupt neu denken?

- IT Umgebungen sind komplexer geworden
- Dynamik, Skalierung, Cloud – alles bewegt sich Schneller
- Klassisches Monitoring stößt an seine Grenzen



IT-Landschaften im Wandel





Monitoring

Klassisches IT Monitoring

- Statusorientiert
- Fokus: Verfügbarkeit und Alarmierung
- "nur die wirklich wichtigen Daten sammeln"

Current Network Status
Last Updated: Tue Jul 19 16:09:38 CEST 2005
Updated every 60 seconds
Nagios® - www.nagios.org
Logged in as *nagiosadmin*

Host Status Totals

Up	Down	Unreachable	P
3	0	0	

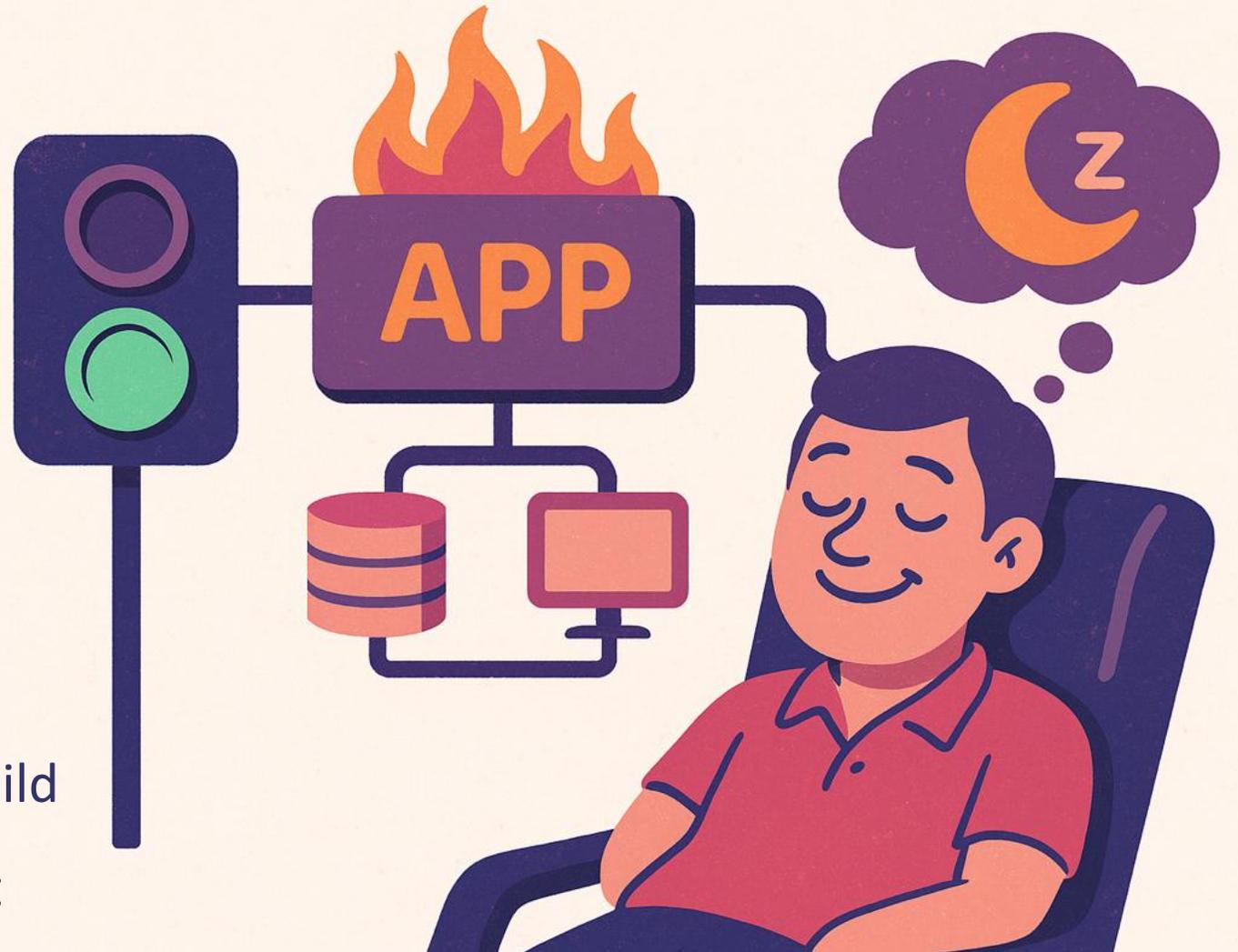
[View History For all hosts](#)
[View Notifications For All Hosts](#)
[View Host Status Detail For All Hosts](#)

Service Status Details Hosts

Host	Service	Status	Last Check	Duration	Attempt
aladin	ABEND	OK	19-07-2005 16:09:02	3d 8h 3m 21s	1/3
	CACHE	OK	19-07-2005 16:04:42	20d 20h 3m 58s	1/3
	CPU-LOAD1	OK	19-07-2005 16:08:12	3d 8h 5m 1s	1/3
	CPU-LOAD5	OK	19-07-2005 16:05:42	3d 8h 3m 1s	1/3
	DNS	OK	19-07-2005 16:06:12	5d 5h 0m 9s	1/3
	FTP	OK	19-07-2005 16:06:52	3d 8h 4m 51s	1/3
	GWA	OK	19-07-2005 16:07:32	3d 7h 59m 51s	1/3
	MTA	OK	19-07-2005 16:08:22	3d 8h 2m 31s	1/3
	PING	OK	19-07-2005 16:04:22	36d 3h 30m 56s	1/3
	POA	OK	19-07-2005 16:05:02	3d 8h 4m 31s	1/3
	PROC_CN	OK	19-07-2005 16:05:22	1d 7h 26m 7s	1/3
	SMTTP	OK	19-07-2005 16:08:32	3d 8h 4m 11s	1/3
	SYS	OK	19-07-2005 16:06:32	15d 9h 47m 47s	1/3
	V_DAT1	OK	19-07-2005 16:07:13	3d 8h 4m 41s	1/3
	V_DAT2	OK	19-07-2005 16:07:52	3d 8h 2m 20s	1/3
	V_DAT3	OK	19-07-2005 16:08:43	15d 9h 49m 47s	1/3
V_DAT4	OK	19-07-2005 16:04:32	20d 20h 5m 58s	1/3	
V_DAT5	OK	19-07-2005 16:09:12	0d 0h 15m 46s	1/3	
V_DATA	OK	19-07-2005 16:05:32	20d 22h 26m 28s	1/3	

Die Grenzen dieser Methode

- Alles grün – und trotzdem Probleme
- Keine Metriken-Trends, kein Gesamtbild
- Reaktion nur nachdem etwas passiert





Observability ist die Fähigkeit, den inneren Zustand eines Systems allein durch von außen beobachtbare Metriken und deren gezielte Aufbereitung zu verstehen – mit dem Ziel, frühzeitig Abweichungen, Muster und Ursachen zu erkennen.

Observability

Was steckt dahinter?

Zustand von Systemen

verstehen

nicht nur „ob es geht“

Intuition

statt Ampelanzeige

Nicht nur

Status

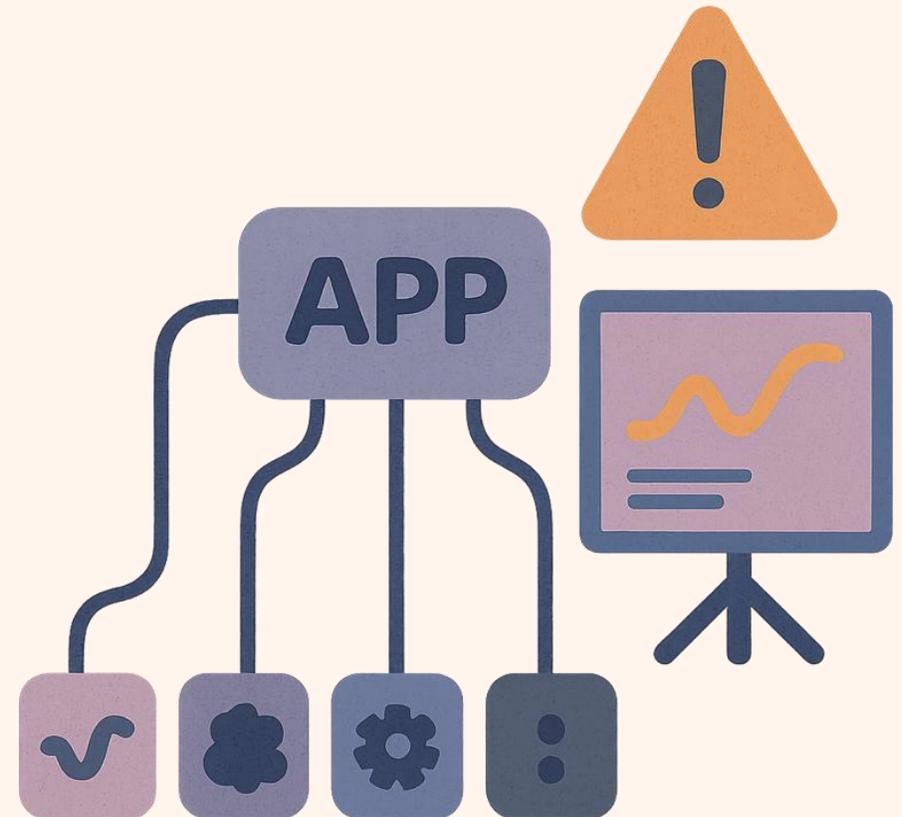
kontinuierliche **Messung** und **Kontext**

mehr als

Monitoring

Wie entsteht Observability?

- Applikations- statt Serverbezogen denken
- Anwendungen / Umgebungen in Komponenten aufspalten
- Pro Komponente passende Metriken definieren
- Metriken strukturieren, korrelieren, sichtbar machen
- Relevante Informationen oben platzieren
- Wenige, aber aussagekräftige Alarme



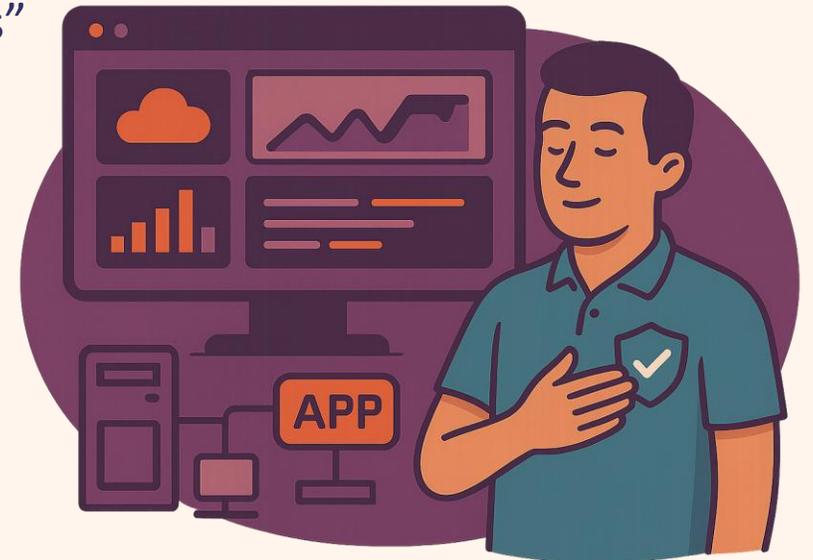
Observability in Cloud-native Umgebungen

- Plattformüberwachung zunehmend automatisiert (Kubernetes, PaaS)
- Immutable Infrastructure. **“Threat servers like cattle, not pets”**
- Self—Healing, Auto-Restart, Skalierung statt manueller Eingriffe
- Weniger “moving parts” → weniger Komplexität -> weniger Monitoring-Aufwand
- Fokus verschiebt sich auf Applikationen und Nutzererlebnis
- Observability wird zur dominanten Disziplin für DevOps Teams
- In klassischen, silo-orientierten Betriebsmodellen bleibt Systemmonitoring zentral



Von Visualisierung zu Verantwortung

- Der Weg zu Observability beginnt nicht mit dem Tool, sondern mit der Arbeitsweise und dem Plattformmodell
- Dashboards sind Arbeitswerkzeug und nicht nur “beautiful colors”
- Wer versteht, kann Verantwortung übernehmen
- DevOps bedeutet End-to-End Verantwortung
- Observability ist dafür die zentrale Voraussetzung



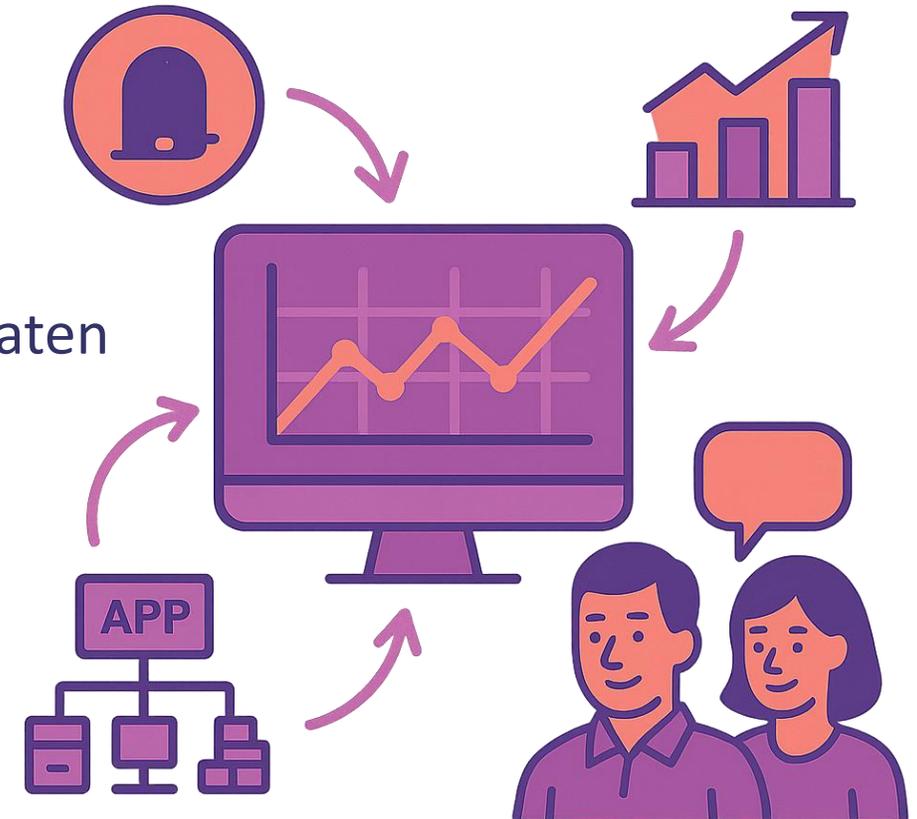
Zabbix

Teil einer modernen
Observability-Strategie



Was sich durch Observability verändert

- Weg vom Schwellwertdenken
- Hin zu Systemverständnis, Trends, vielfältigen Live-Daten
- Frühwarnung statt Eskalation
- Mehr Zusammenarbeit, weniger Silos



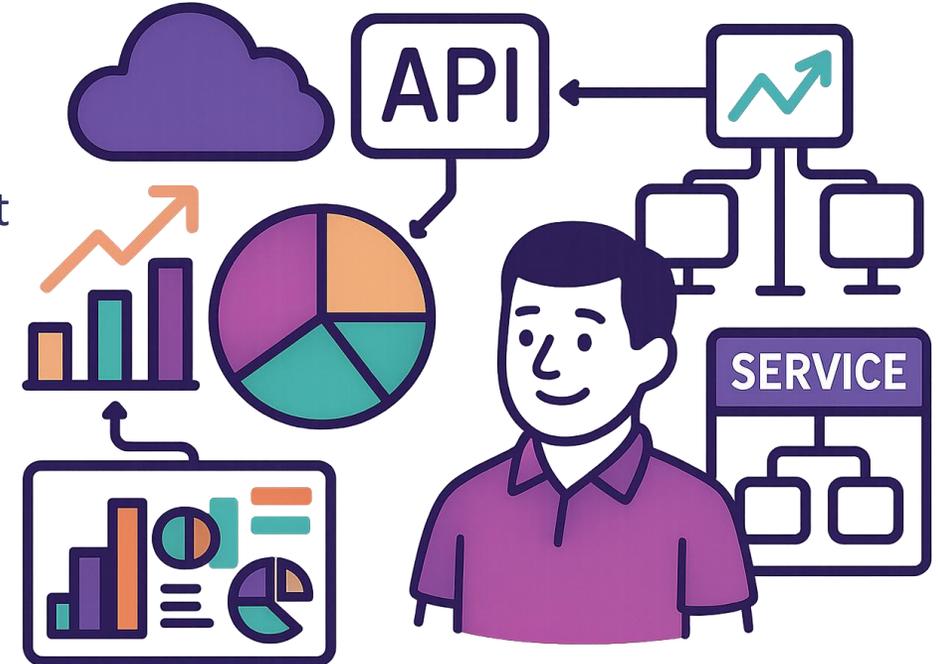
Praxisbeispiel: Klassisches Rechenzentrum

- Überwachung von Hosts, Storage, Netzwerk
- Agent/SNMP/Checks
- SLA-Dashboards, Alarmierung,
Kapazitätsplanung
- Monitoring & Betrieb meist zentral organisiert
- Silo-Strukturen: Fachabteilungen betreiben
Systeme getrennt



Praxisbeispiel: DevOps & Microservices

- Telemetrie direkt aus Applikationen (instrumentierte Anwendungen)
- Exporter oder APIs als Metrikenquelle
- Technische + Business KPIs kombinieren
- Applikationsteams übernehmen Verantwortung für ihre Services
- Monitoring ist Teil der täglichen Entwicklungs- und Betriebsarbeit



Erkennen statt Reagieren – wenn Observability wirkt

Dashboards zeigen Trends & Entwicklungen über Zeit

Kontext-Metriken machen Anomalien sichtbar, bevor sie kritisch sind

Teams interpretieren statt nur alarmieren

Geteiltes Verständnis = schnelleres Handeln

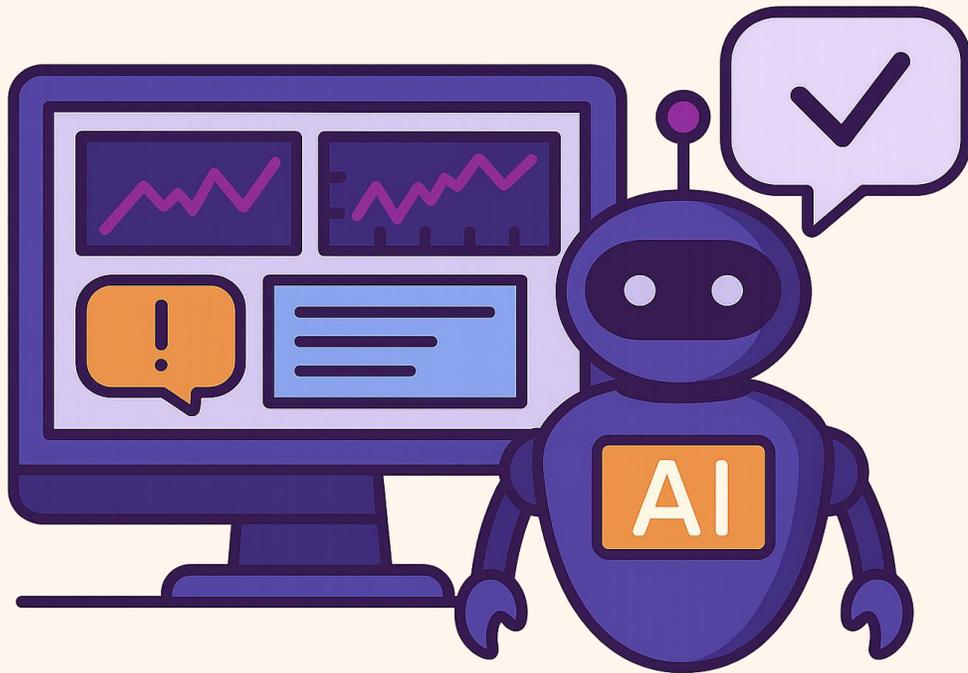
Klassische Tools hätten das nicht gezeigt





Zukunft

AIOps



- Dashboards für Menschen – AIOps erkennt Muster
- KI analysiert Metriken und schlägt Handlungen vor
- Erste Erweiterungen existieren
- Zabbix als Datenbasis für smarte Systeme

Monitoring, Observability, oder beides?



- Observability ist nicht nur ein neues Buzzword!
- Klassisches Monitoring bleibt wichtig bei Plattformverantwortung
- Observability ist ideal für moderne, cloudnative Umgebungen
- Zabbix kann beides – stabil und zukunftsorientiert
- Es gibt selten *ein einziges Tool* für alles
- Monitoring-Strategien müssen fallweise entwickelt werden



Vielen Dank!

Christian Anton

christian.anton@inqbeo.de