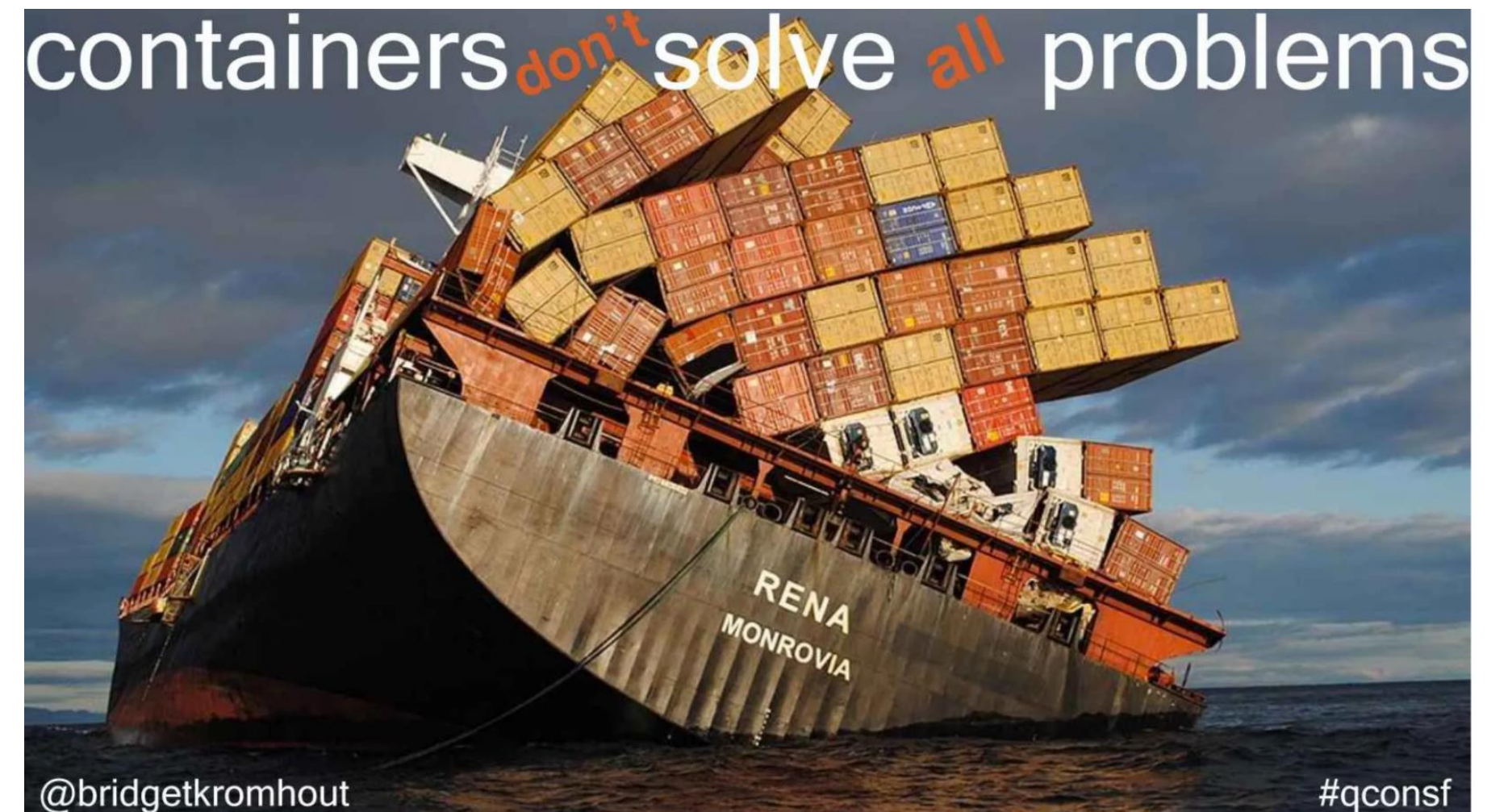


# Warum ein Admin Agent

## Motivation & Problemstellung

- Komplexität moderner Cloud-Native Infrastruktur
- Vielzahl an Quellen:
  - Logs, Metriken, YAML-Konfiguration,....
- Zeitintensive Manuelle Analyse bei Störungen
- Einarbeitungszeit



# Was soll und kann ein Agent leisten

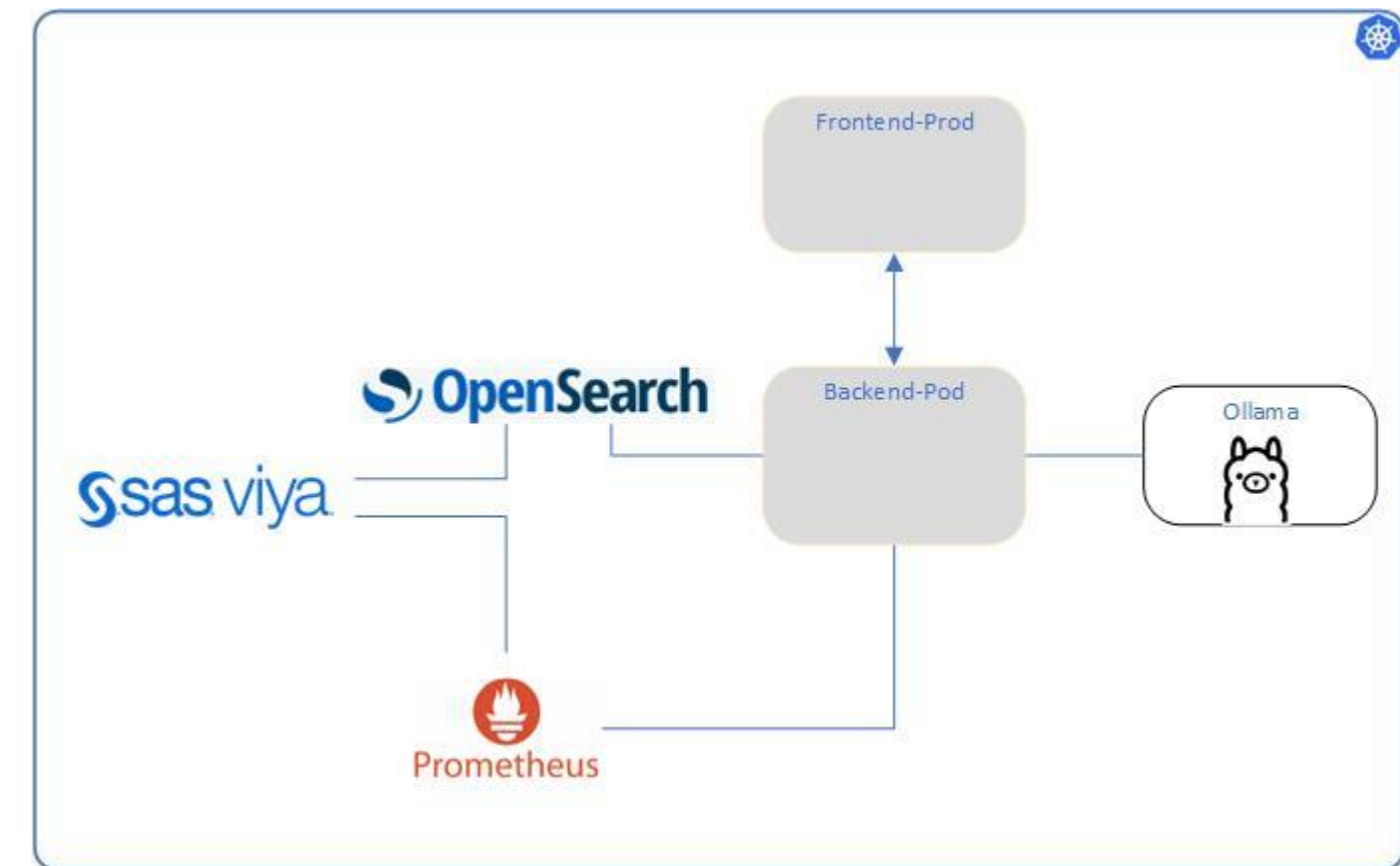
## Zielsetzung & Funktionsweise

- Anfragen über natürlich Sprache beantworten wie z.B.: Was war gestern in der Zeit von ... der Grund für Probleme mit ...
- Automatisierte Abfragen in:
  - Opensearch (Logs)
  - Prometheus (Metriken)
  - Vektor-Datenbank (Dokumentationen, Deployment YAMLs, Jira, ...)
- Interpretation durch ein LLM

# Architekturüberblick

## Architektur & Komponenten

- Frontend:
  - Streamlit, FastAPI
- Backend:
  - FastAPI – (API, LLM Anbindung, Vektor-Suche)
- LLM
  - Ollama ([ollama.com](https://ollama.com))
- Datenquellen:
  - Opensearch für Logs ([SAS Viya Monitoring for Kubernetes](#))
  - Prometheus für Metriken ([SAS Viya Monitoring for Kubernetes](#))
  - Vektor DB (YAMLs, PDFs, ...)
- Kubernetes-Deployment



# Use Case – Viya Status

## Demo & Use Case

- Anfrage:
  - Hat es am 14.05.2025 Probleme gegeben, die sich auf die Performance der Viya Web-Anwendungen negative ausgewirkt haben sollte?
- Schritte des Agenten:
  - Zeitfenster analysieren
    - LLM zur Ermittlung des angefragten Zeitfensters befragen
  - OpenSearch Logs abfragen
    - < LLM auf Basis der User-Anfrage eine DQL Queries generieren und abschicken >
  - Prometheus Logs abfragen
    - < LLM auf Basis der User-Anfrage eine PromQL Queries generieren und abschicken >
  - Übergabe der gesamten Ergebnisse an das LLM und Ausgabe eines detaillierten Berichts



# Demo

## Beispielausgabe

### Kubernetes System Status & Optimizations LLM

Stelle eine Frage zu deinem Cluster (z. B. Status, Optimierungsempfehlungen, potenzielle Engpässe):

Deine Frage:

Lief gestern am 12.05.2025 zwischen 13 und 14 Uhr alles gut?

Antwort generieren

### Antwort

### Log Auswertung:

PodName	Log Message	Log Count
sas-decisions-framework	"[CUSTOM_FUNCTIONS_SERVICE_ERROR]"	369
KUBE_EVENT	"invalid metrics (1 invalid out of 1)"	91
sas-bootstrap-tool	"Error bootstrapping resources"	37
cas	"Successfully retrieved List of cluster nodes [error: number nodes:10]"	17

### Analyse:

Die große Zahl von Fehlern in den Logs zeigt eine Störung im Verhalten des **sas-decisions-framework** und der **KUBE\_EVENT**. Der Fehler "503 : "[{"errorCode":1200,"message":"The SAS license for product ID "1,046" was not found." ...]" aus dem sas-decision framework kann dazu führen, dass nicht richtige Business Rules berechnet werden. Der Fehler in den KUBE\_EVENT Logs ist mit "invalid metrics (1 invalid out of 1)" beschrieben und kann die Messung der Ressourcen verzerren, was zu falscher Resource-Management und Performanceproblemen führen kann. Die weiteren Error Nachrichten im kube-event deuten an, dass es Probleme mit Image Pulls gibt. Dies kann dazu führen, dass Docker-Container nicht korrekt inititalisiert oder beendet werden können, was zu Performanceproblemen und unberechenbarer Verhalten führen kann.

Der Warnung aus dem sas-microanalytic-score steht deutlich auf, dass "spring.jpa.open-in-view is enabled by default". Dies kann dazu führen, dass bei Vielbenutzung der Anwendungen, die das Framework verwenden, mehr Database-Abfragen durchgeführt werden können, was zu Performanceproblemen führen kann.

Die OAuth-Tool Fehler in sas-oauth-tool können dazu führen, dass es Probleme mit der Authentifizierung gibt oder die Erstellung von OAuth Klienten fehlschlägt - was zu Sicherheitslücken führen kann.

### Zusammenfassung:

Es gab ein hohes an Fehler im sas-decisions-framework, wahrscheinlich aufgrund eines fehlenden SAS Lizenzes für produkt ID 1,046 vorhanden war. Die Fehler in den KUBE\_EVENT Logs sind mit den Image Pull Errors zusammenhängend und könnten dazu führen, dass Docker-Container nicht korrekt initialisiert werden oder beendet werden können, was sich negativ auf Performance und Bereitstellbarkeit auswirken kann. Die Anweisung in sas-microanalytic-score könnte dazu führen, dass mehr Database Abfragen durchgeführt werden können und daher zu Performanceproblemen führt. Die OAuth-Tool Fehler zeigen auf Probleme mit der Authentifizierung oder der Erstellung von OAuth Klienten, dies kann sich negativ auf Anwendungssicherheit auswirken.

# Was bringt der Agent konkret

## Vorteile, Ausblick & Fazit

- Weniger Zeitaufwand bei der Fehleranalyse
- Probleme frühzeitig erkennen
- Benötigte Erweiterungen:
  - Retrieval-Augmented Generation (RAG)
    - Nutzung von OpenSearch Funktionalitäten aus dem SAS-Monitoring Stack
    - Weitere Test mit Informationen in einer Vektor-DB wie SAS und K8s Dokumentationen, Deployment-Files, ...
    - Ggf. bestimmte Logs im Batch in die Vektor DB laden
    - Erneute Tests zur automatisierten Log- und Metrik-Bestimmung für passende Abfragen
- Mögliche Erweiterungen
  - Automatisierte Gegenmaßnahmen (Self-Healing)
    - Wie erfolgreich könnte man den Agenten dazu bringen z.B. nach Bedarf selbstständig zu Skalieren oder kleinere Fehler zu beheben?

# Danke

