

BLAUPAUSE

Der selbst gehostete KI-Stack

Private Inferenz, Agenten-Gedächtnis und Assistenten auf dem eigenen Kubernetes. Datensouveränität durch Architektur statt durch Zusicherung - der Stack, den wir selbst betreiben.

Warum KI überhaupt selbst hosten

Managed KI-APIs sind exzellent - bis die Datenfrage kommt. Anfragen mit Kundenangaben, interne Dokumente, die einen Assistenten speisen, Agenten-Gedächtnis mit Geschäftswissen: All das verlässt Ihren Perimeter, sobald der Anbieter extern ist. Self-Hosting lehnt externe KI nicht ab; es zieht eine Linie, die sensible Kontexte nie überschreiten.

Die Referenzarchitektur in unserem Betrieb

- Kubernetes auf Talos Linux - immutable, minimale Angriffsfläche, via GitOps verwaltet, sodass der Clusterzustand in Git überprüfbar ist
- Model Serving für private Inferenz - LLMs und Embeddings laufen im Cluster für Aufgaben, die ihn nicht verlassen dürfen
- Cognee als Agenten-Gedächtnis - eine Wissensgraph-Schicht, die Agenten persistenten, prüfbaren Kontext gibt statt intransparentem Prompt-Füllmaterial
- Typebot für geführte Assistenten - konversationelle Annahme und interne Helfer, selbst gehostet mit eingeschränkten erlaubten Origins
- MinIO für Objektspeicher - Dokumente, Artefakte und modellnahe Daten unter denselben Souveränitätsregeln
- Keycloak für Identität - Assistenten und Werkzeuge authentifizieren sich gegen denselben IdP wie das Team

Das ist kein Verkaufsdiagramm. Es ist der Stack, auf dem die Assistenten, Gedächtnisse und Pipelines dieses Unternehmens heute laufen.

Ehrliche Dimensionierung

Ein rein CPU-basierter Cluster bewältigt Embeddings, Retrieval, Orchestrierung und kleinere Modelle gut. GPU-Knoten werden relevant, wenn schnelle private Inferenz auf grösseren Modellen gefragt ist - sie bringen Kosten und Scheduling-Komplexität. Die ehrliche Antwort lautet: Beginnen Sie mit der Assistenten- und Gedächtnisschicht, messen Sie die reale Nachfrage, dann entscheiden Sie, ob sich dedizierte Inferenz-Hardware lohnt. Für viele Organisationen ist das Muster hybrid - öffentliche APIs für unsensible Aufgaben, private Inferenz für sensible Kontexte.

Was der Betrieb konkret bedeutet

- ☐ Versionierte Deployments via GitOps - jedes Modell, jeder Assistent und jeder Gedächtnisdienst ist eine geprüfte Änderung, kein Klick in einer Konsole
- ☐ Backup- und Restore-Verfahren, die Agenten-Gedächtnis und Objektdaten abdecken, mit dokumentierten Restore-Tests
- ☐ Monitoring, das KI-Dienste wie jede andere Workload behandelt - Verfügbarkeit, Latenz, Ressourcenverbrauch
- ☐ Zugriff über den Unternehmens-IdP - Assistenten sind kein Schatten-Identitätssystem
- ☐ Update-Pfade, die zuerst auf einer Kopie getestet werden - eine kaputte Gedächtnisschicht ist ein Ausfall, kein Experiment

Dieses Dokument ist Teil der UHL-Ressourcen. Direkter Download und verwandte Arbeiten:

<https://uhl.cloud/de/resources/>

