



Dateien zum Herunterladen
Einwählen ins WLAN ,mhg'; Passwort ist ,salzburg'.
Netz-Volume mounten unter:
WIN: [\\192.168.77.1\](http://192.168.77.1)
Mac: smb://192.168.77.1
Laufwerk ,H:' bzw. ,H' → Ordner ,Claris RAG Server'

RAG-Workshop · FileMaker 2026 & Claris AI Server

RAG mit Claris AI Server

Retrieval-Augmented Generation: aus Gesetzes-PDFs wird durchsuchbares Wissen

Vortrag & Mini-Workshop



RAG-Workshop · FileMaker 2026 & Claris AI Server

RAG mit Claris AI Server

Retrieval-Augmented Generation: aus Gesetzes-PDFs wird durchsuchbares Wissen

Vortrag & Mini-Workshop



Was ist RAG?

RAG (Retrieval-Augmented Generation) verbindet ein Sprachmodell mit einer Wissensquelle:

Relevante Dokumente werden gesucht und der KI vor der Antwort als Kontext mitgegeben.



Über mich — Dipl.-Ing. Michael Heider

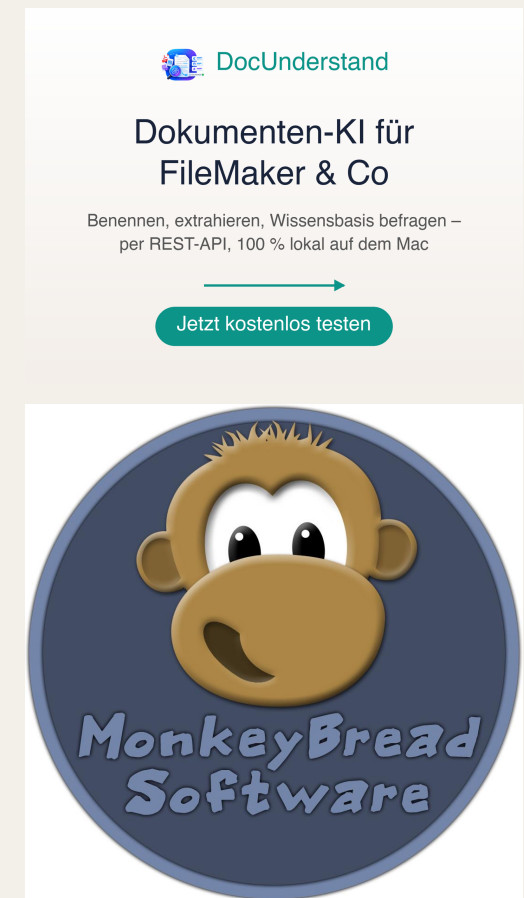
Dipl.-Ing. Michael Heider

- Studium der Verfahrenstechnik mit Schwerpunkt auf Optimierungsstrategien & Bio-Verfahrenstechnik
- Seit 1984 Datenbankentwicklung
- Seit 1988 Datenbankentwicklung mit FileMaker Pro
- 1995 Buch „FileMaker Pro – Good Programming Practice“
- 2002 Gründung der Michael Heider GmbH
- Spezialisierung rund um individuelle Softwarelösungen für Unternehmen
- Passionierter (Frei-)Taucher, Behindertentauchlehrer, Gründungsmitglied des 1. Berliner Inklusions-Tauchklub e.V.





Wir danken unseren Sponsoren!





Rechtlicher Hinweis

⚠ Ich bin kein Rechtsanwalt!

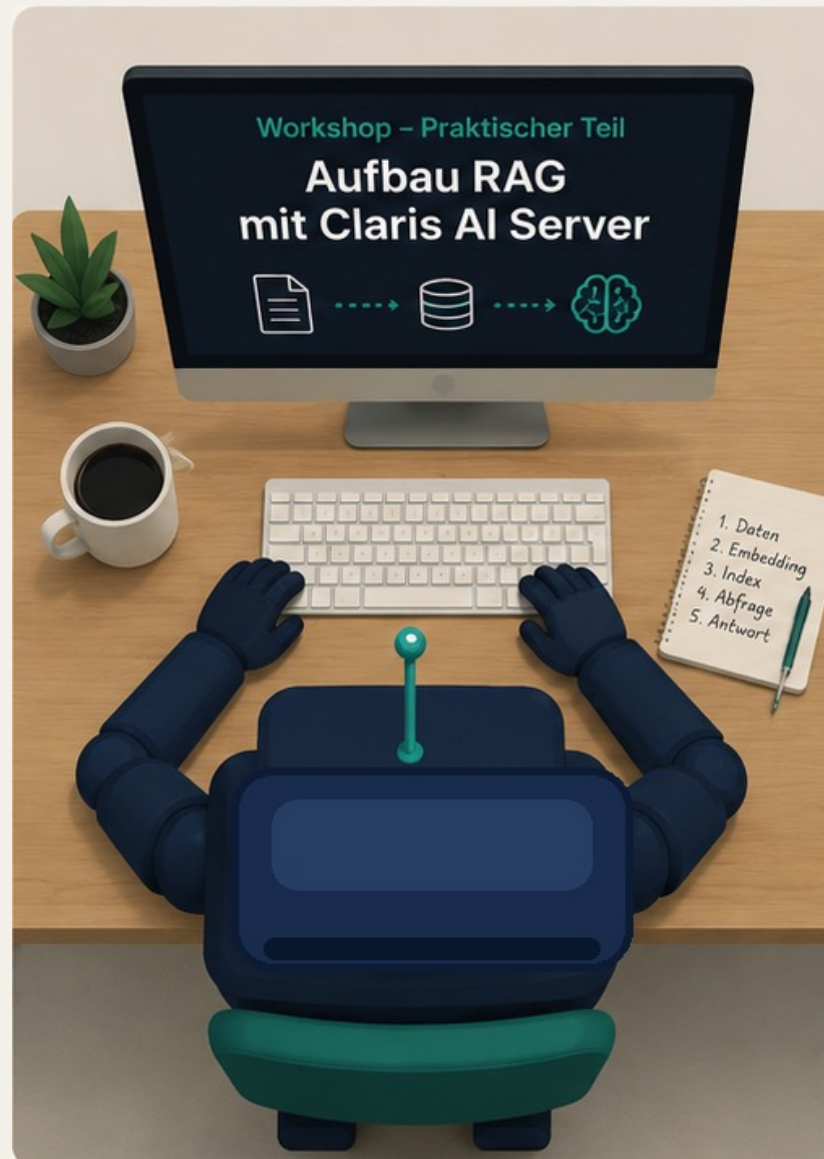
Alle Inhalte dieses Vortrags sind nach bestem Wissen und Gewissen recherchiert — jedoch ohne Gewähr!
Das erstellte RAG ist keine rechtssichere Auskunft. Für verbindliche Aussagen Fachanwalt und / oder Datenschutzbeauftragte konsultieren.
Jede/r muss für seine Anwendungen selbst recherchieren und die Schlussfolgerungen eigenverantwortlich abschätzen.



Diese Folien und der Workshop wurden unter Zuhilfenahme einer Künstliche Intelligenz erstellt!

⚠ **Alle Angaben in diesem Vortrag sind ohne Gewähr!**

⚠ **Alle Angaben des RAG sind ohne Gewähr!**





Glossar

LLM

Large Language Model – ein KI-Sprachmodell, das auf riesigen Textmengen trainiert wurde (z. B. Apertus, Teuken, Google Gemma, GPT).

RAG

Retrieval-Augmented Generation – das LLM wird vor der Antwort mit relevantem Kontext aus einer Wissensdatenbank versorgt.

Chunk

Textabschnitt, in den ein Dokument zerlegt wird. Im Workshop: genau ein § bzw. Artikel = ein Chunk.

Embedding / Vektor

Numerische Darstellung eines Texts als Zahlenvektor. Ähnliche Texte liegen im Vektorraum nah beieinander.

Kosinusähnlichkeit

Maß für die Ähnlichkeit zweier Vektoren: 1 = identisch, 0 = völlig verschieden. Grundlage der semantischen Suche.

RAG Space

Persistente Vektor-Datenbank auf dem Claris AI Model Server, die alle Chunk-Embeddings speichert.

Prompt

Eingabetext an das LLM – enthält die Nutzerfrage sowie den vom RAG Space gelieferten Kontext.

Halluzination

Wenn ein LLM plausibel klingende, aber erfundene Antworten liefert – besonders bei Themen außerhalb des RAG Space möglich.





Was ist RAG?

Ein klassisches Large Language Model (LLM) kennt nur das, was in seinen Trainingsdaten stand – mit festem Stichtag, ohne Zugriff auf Ihre Datenbankinhalte und mit der bekannten Neigung, fehlende Fakten plausibel zu „erfinden“ (Halluzination). Für betriebliche Anwendungen ist das ein Problem: Sie wollen Antworten auf Basis *Ihrer* Dokumente, Verträge, Handbücher oder Stammdaten – und das nachvollziehbar.

Genau hier setzt **RAG** an. Statt das Modell aufwändig nachzutrainieren (Fine-Tuning), reichert man jede Anfrage zur **Laufzeit** mit **passendem Kontext** an. Das geschieht in zwei Phasen.



Was ist RAG?

Phase 1 – Indizierung (vorbereitend): Ihre Dokumente werden in kleinere Abschnitte zerlegt („Chunks“). Ein Embedding-Modell wandelt jeden Chunk in einen Vektor um – eine Zahlenrepräsentation seiner Bedeutung. Diese Vektoren landen in einem durchsuchbaren Vektorspeicher (der „Wissensbasis“).

Phase 2 – Abfrage (zur Laufzeit): Die Frage des Nutzers wird ebenfalls in einen Vektor umgewandelt. Per Ähnlichkeitssuche (typischerweise Kosinus-Ähnlichkeit) holt das System die thematisch nächstliegenden Chunks (die „Top-K“). Diese werden zusammen mit der ursprünglichen Frage zu einem Prompt kombiniert und an das LLM geschickt. Das Modell formuliert die Antwort also *auf Basis der gelieferten Fakten*, nicht aus dem Gedächtnis.

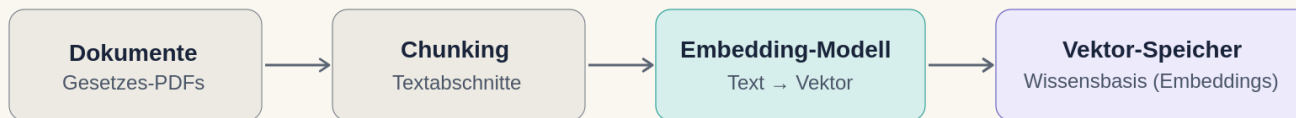
Der Nutzen für Sie als Entwickler: aktuelle und unternehmenseigene Daten ohne Neutraining, deutlich weniger Halluzinationen, Belegbarkeit (man weiß, welche Quelle zur Antwort geführt hat) und einfache Aktualisierung – neue Dokumente werden einfach in die Wissensbasis aufgenommen, nicht ins Modell eintrainiert. RAG ist damit das Standard-Muster, um ein generisches Sprachmodell an eine konkrete Fachdomäne anzubinden.



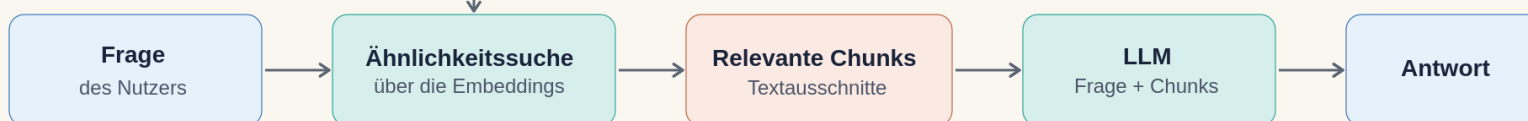
Was ist RAG?

Retrieval-Augmented Generation – erst relevante Dokumente suchen, dann das Sprachmodell damit antworten lassen.

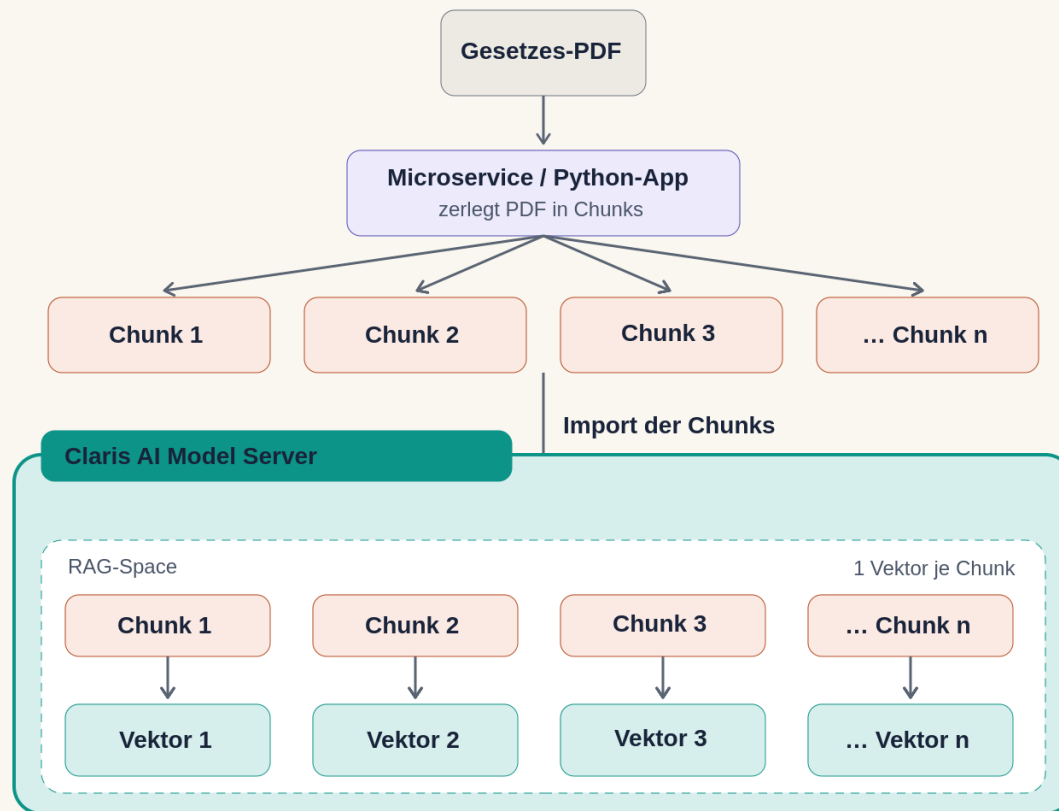
Phase 1 · Indizierung (vorbereitend)



Phase 2 · Abfrage (zur Laufzeit)



Phase 1: Aus Gesetzestext werden Einbettungen



1 · Aufteilen – Ein Microservice (Python-App) zerlegt den Gesetzestext in Chunks.

2 · Import – Die Chunks gehen über die Systemgrenze in den RAG-Space.

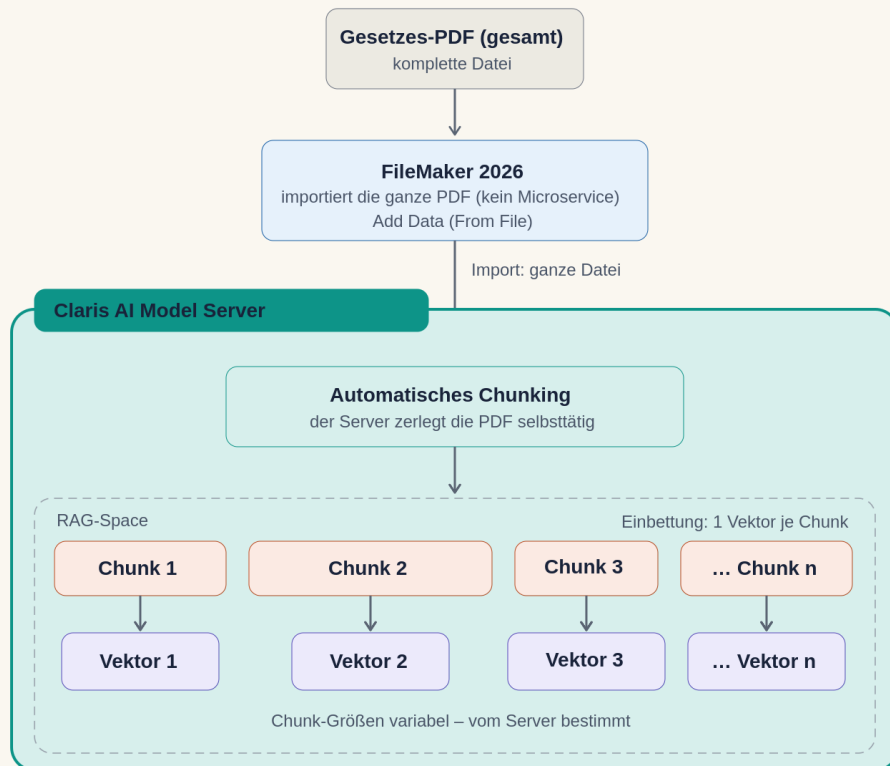
3 · Einbettung – Der Claris AI Server erzeugt je Chunk genau einen Vektor.

Chunkgröße:

Im Workshop ist jeder Gesetzes-Paragraph genau ein Chunk.



Phase 1: Einbettungen – Alternative



Nachteil: keine Kontrolle über Chunk-Größe und -Aufteilung

Das Chunking läuft automatisch – eigene Vorgaben (z. B. pro Paragraph) sind nicht möglich.

Unterschied zur Microservice-Variante

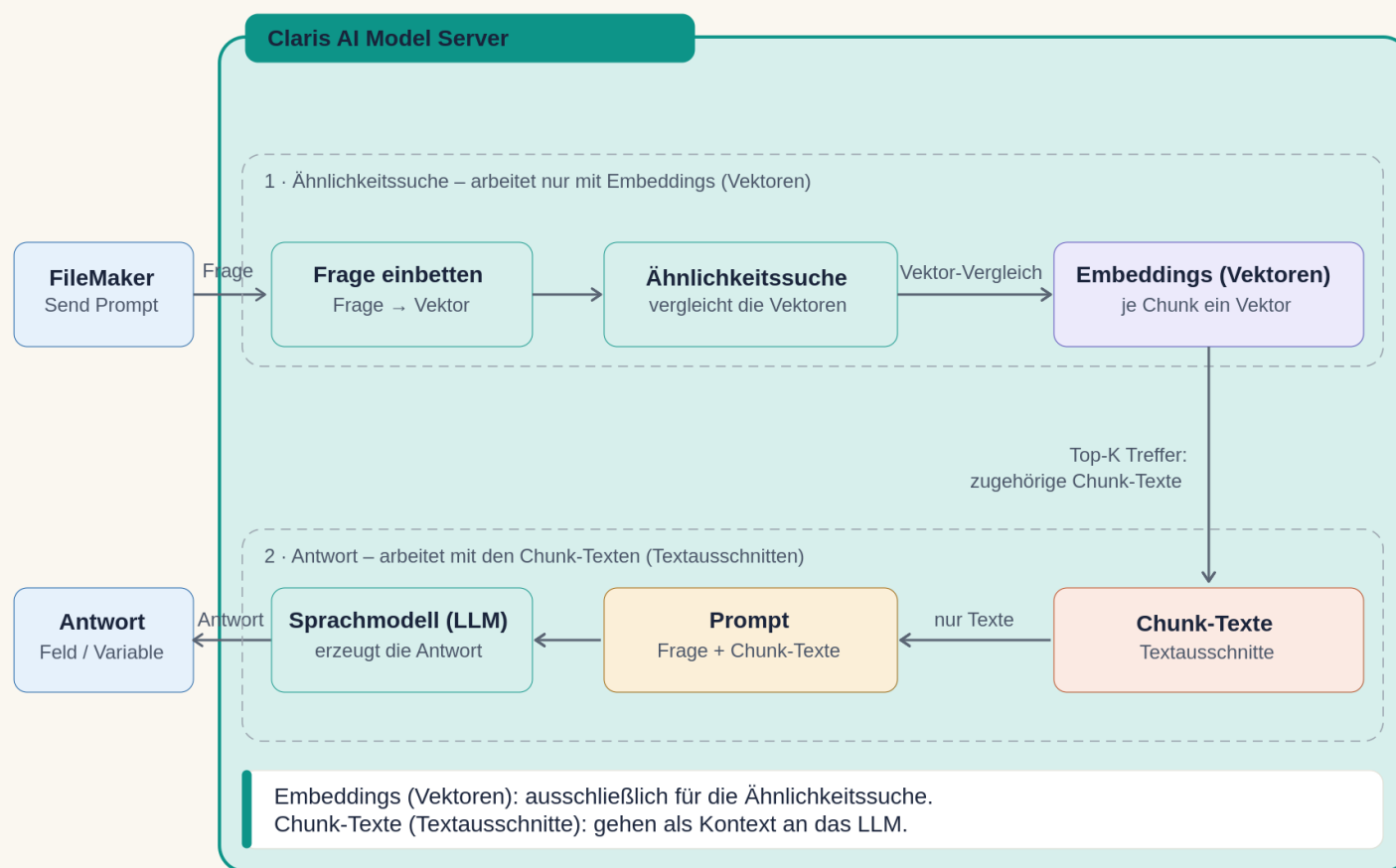
Kein Microservice – FileMaker übergibt die ganze PDF (Add Data, From File).

Automatisch – Chunking und Einbettung erledigt der Clarif AI Server selbst.

Nachteil – keine Kontrolle über Chunk-Größe und -Aufteilung.



Phase 2: Der Abfrageweg



Embeddings nur für die Ähnlichkeitssuche – an das LLM gehen ausschließlich die Chunk-Texte.





Workshop- Praxisteil





☐	swiss-ai/Apertus-8B-Instruct-2509	Textgenerierung	Geladen	Ja	Entladen
☐	paraphrase-multilingual-MiniLM-L12-v2	Einbettung	Geladen	Ja	Entladen



[Erweiterte Einstellungen ausblenden](#)

RAG-Einstellungen [?](#)

Kosinus-Ähnlichkeitsschwelle für RAG-Ergebnisse verwenden	Deaktiviert	☐
Anzahl der bestplatzierten Ergebnisse für die RAG-Zusammenfassung	20	Ändern
RAG-Cache-Pfad	/Library/FileMaker Server/Data/AIServices/RAG/cache/rag.txt	Ändern
Lokales Einbettungsmodell für RAG	multi-qa-MiniLM-L6-cos-v1	Ändern
OpenAI-Einbettungen verwenden	Deaktiviert	☐
Batch-Größe für das Erstellen von RAG-Einbettungen	16	Ändern
Anzahl der Token pro Textblock ?	200	Ändern

[RAG-Spaces importieren](#)

Feinabstimmungseinstellungen [?](#)

System-Prompt zur Feinabstimmung	Nicht konfiguriert	Ändern
----------------------------------	--------------------	------------------------



[Erweiterte Einstellungen ausblenden](#)

RAG-Einstellungen [?](#)

Kosinus-Ähnlichkeitsschwelle für RAG-Ergebnisse verwenden	Deaktiviert	☐
Anzahl der bestplatzierten Ergebnisse für die RAG-Zusammenfassung	20	Ändern
RAG-Cache-Pfad	/Library/FileMaker Server/Data/AIServices/RAG/cache/rag.txt	Ändern
Lokales Einbettungsmodell für RAG	paraphrase-multilingual-MiniLM-L12-v2	Ändern
OpenAI-Einbettungen verwenden	Deaktiviert	☐
Batch-Größe für das Erstellen von RAG-Einbettungen	16	Ändern
Anzahl der Token pro Textblock ?	200	Ändern

[RAG-Spaces importieren](#)

Feinabstimmungseinstellungen [?](#)

System-Prompt zur Feinabstimmung	Nicht konfiguriert	Ändern
----------------------------------	--------------------	------------------------



[Erweiterte Einstellungen ausblenden](#)

RAG-Einstellungen [?](#)

Kosinus-Ähnlichkeitsschwelle für RAG-Ergebnisse verwenden	Aktiviert	☒
Kosinus-Ähnlichkeitsschwelle für RAG-Ergebnisse	0.7	Ändern
Anzahl der bestplatzierten Ergebnisse für die RAG-Zusammenfassung	20	Ändern
RAG-Cache-Pfad	/Library/FileMaker Server/Data/AIServices/RAG/cache/rag.txt	Ändern
Lokales Einbettungsmodell für RAG	paraphrase-multilingual-MiniLM-L12-v2	Ändern
OpenAI-Einbettungen verwenden	Deaktiviert	☐
Batch-Größe für das Erstellen von RAG-Einbettungen	16	Ändern
Anzahl der Token pro Textblock ?	200	Ändern

[RAG-Spaces importieren](#)

Feinabstimmungseinstellungen [?](#)

System-Prompt zur Feinabstimmung	Nicht konfiguriert	Ändern
----------------------------------	--------------------	------------------------



192.168.6.74/admin-console

Claris FileMaker AI Model Serve

Neuer Tab

Nicht sicher

192.168.6.80/admin-console/app/llm/llmkeys

☆

↓

M

🔖

⚙

☰

Claris AI Model Server

Mittwoch, 9. September 2026 um 19:40

KI-SERVER

🔔

Abmelden

Konfiguration

KI-Dienste

Administration

Protokolle

KI-Dienste

KI-Dienste verwalten das Herunterladen von Open-Source-Modellen und deren Laden in den Arbeitsspeicher, führen Inferenz mit geladenen Modellen durch, führen Retrieval-Augmented Generation (RAG) durch, optimieren Basismodelle mit Ihren eigenen Training-Daten und verwalten API-Schlüssel, um den Zugriff auf den Modellserver zu steuern.

Modellserver

Einstellungen

Schlüssel

Modelle

Anfragen

Chat

Schlüssel

Erstellen Sie API-Schlüssel, um den Zugriff auf KI-Dienste zu kontrollieren. Verwenden Sie in FileMaker-Scripts diese Schlüssel zusammen mit dem Endpunkt, um Konten für den Zugriff auf den Modellserver als benutzerdefinierten Modellanbieter zu konfigurieren.

Schlüssel erstellen

Löschen

☐	Name	Erstellt	Läuft ab
☐	Neuer Schlüssel	09.09.2026, 19:40	Einstellungen

Name

Maximal 31 Zeichen

Zugriff auf Dienste zulassen

☐ Einbettungsgenerierung

☐ Text- und Abfragegenerierung

☐ Feinabstimmung

☐ Bildbeschriftung

☒ Retrieval-Augmented Generation (RAG)

☐ Berechtigungen auf ausgewählte RAG-Spaces anwenden

☐ Zugriff auf Serverkonfiguration

RAG Space ID

rechtsinfo

Löschen

Berechtigungen

☐ Prompt senden

☐ Daten hinzufügen

☐ Daten entfernen

☐ Daten importieren aus

Hinzufügen

Läuft ab

Läuft nicht ab

Speichern

Abbrechen

fm-meets-mozart.net





Frage stellen	★	RAGspace_Befüllung	LLM	Configure Accounts	★	RAGspace_Info	👁
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							

```
# LLM
Set Variable [ $endpoint ; Value: "https://10.20.20.61/llm/v1/" ]
Set Variable [ $key ; Value: "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJyY2Nlc3MiOiJZWlZWRkaW5nIjp0cnVlLCJmaW5lX3R1bmluZyI6ZmFsc..." ]
Configure AI Account [ Account Name: "ai" ; Model Provider: Custom ; Endpoint: $endpoint ; API key: $key ]

# RAG
Configure RAG Account [ RAG Account Name: "rag" ; Endpoint: $endpoint ; API key: $key ]

Set Variable [ $$registeredThisSession ; Value: 1 ]

Configure Prompt Template [ Template Name: "rechtsratgeber_prompt" ; Model Provider: Custom ; Template Type: RAG Prompt ; RAG Prompt:
"D Du bist ein juristischer Informationsassistent für deutsches Recht. Beantworte die Frage AUSSCHLIESSLICH auf Basis der bereit..." ]
```

Specify Calculation

mined at runtime.

+

||

"Du bist ein juristischer Informationsassistent für deutsches Recht. Beantworte die Frage AUSSCHLIESSLICH auf Basis der bereitgestellten Gesetzes-Auszüge im RAG-Space (Kontext). Regeln:

1. Bevor du die Frage recherchierst, ermittle zunächst aus welchem Rechtsgebiet die Frage stammt. Dies kann sein 'Zivilrecht', 'Strafrecht', 'Datenschutz', 'Verkehrsrecht' etc.pp. Ordne die Frage eindeutig einem Rechtsgebiet zu. Schreibe dieses Rechtsgebiet in die erste Zeile deiner Antwort.
3. Im RAG-Space sind vorhanden Quellen für die Gesetztestexte zu BGB, GG und DSGVO. Leite hieraus ab, welche Rechtsgebiete mit diesen Quellen abgedeckt sind.
4. Vergleiche das Rechtsgebiet der Frage mit den Rechtsgebieten aus den Quellen und beantworte Fragen zu den zugehörigen Rechtsgebieten aus den vorhandenen Quellen im RAG-Space. Gebe bei Fragen zu anderen Rechtsgebieten zu Beginn der Antwort diesen Satz aus: \"Zu diesem Rechtsgebiet ist leider kein Material hinterlegt, meine Spekulation hierzu lautet: \".
5. Erfinde KEINE Antworten aus eigenen Quellen zu Themen, zu denen kein Material im RAG-Space steht!
6. Antworte auf Deutsch, sachlich und knapp.
7. Zitiere die einschlägige Norm mit Fundstelle (z.B. \"§ 433 BGB\").
8. Beende JEDE Antwort mit: \"Unverbindliche Information – keine Rechtsberatung.\"

Frage: :question:"

""

()

&

&

=

±

>

<

≥

≤

+

-

/

*

not

and

Search

> Text

> Text forr

> Number

> Date

> Time

> Timestar

> Containe

> Japanes

> JSON

> Aggrega

> Repeatir

> Financia

