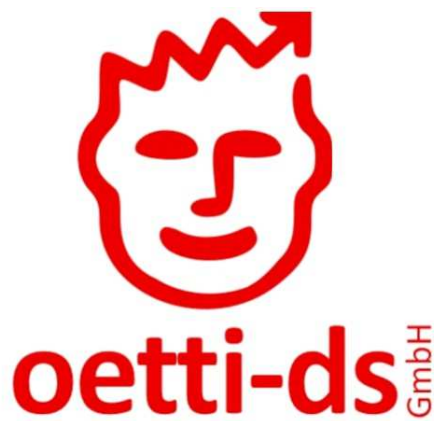




Der AI-Agenten-Kompass

Marktüberblick und Bewertung der führenden
Software-Plattformen für die Erstellung von
Enterprise AI-Agenten

November 2025

oetti-ds - AI for Excellence

oetti-ds - AI for Excellence

oetti-ds -





Wir begleiten Sie auf dem Weg in die Ära der **Agentic AI**

Warum dieser Kompass für AI-Agenten Software

Künstliche Intelligenz hat sich in den vergangenen Jahren von einer futuristischen Vision zu einer geschäftskritischen Realität entwickelt. Während große Sprachmodelle wie ChatGPT, Claude oder Gemini bereits den ersten Schritt in eine neue Ära der KI-Anwendungen markierten, erleben wir heute den nächsten evolutionären Sprung: die Entstehung autonomer AI-Agenten. Diese intelligenten Systeme gehen weit über einfache Chatbots hinaus – sie können komplexe Aufgaben eigenständig planen, durchführen und dabei auf verschiedene Tools, Datenquellen und Schnittstellen zugreifen.

Die Revolution der autonomen AI-Agenten

AI-Agenten repräsentieren einen Paradigmenwechsel in der Art, wie Unternehmen mit Technologie interagieren. Statt passiver Antworten auf Anfragen können diese Systeme proaktiv handeln, Entscheidungen treffen und ganze Geschäftsprozesse automatisieren. Von der automatisierten Kundenbetreuung über intelligente Datenanalyse bis hin zur strategischen Unternehmensplanung – AI-Agenten versprechen, die Produktivität und Effizienz in Organisationen grundlegend zu transformieren.

Diese Entwicklung hat nicht nur die Aufmerksamkeit von Technologieenthusiasten geweckt. Vorstände, IT-Entscheider und Fachbereichsleiter weltweit diskutieren intensiv über die Möglichkeiten und Herausforderungen, die AI-Agenten für ihre Unternehmen bedeuten. Die Frage ist längst nicht mehr, ob AI-Agenten Teil der digitalen Transformation werden, sondern wie schnell und in welcher Form sie implementiert werden sollten.

*„Wir möchten einen AI-Agenten erstellen.
Welche Software sollen wir dafür verwenden?“*

Ein Markt im Aufbruch – und im Chaos

So vielversprechend die Technologie auch ist, so chaotisch präsentiert sich derzeit der Markt für Software(-Plattformen) für AI-Agenten. In einem Tempo, das selbst für die schnelllebige Tech-Branche außergewöhnlich ist, entstehen nahezu täglich neue Anbieter, Plattformen und Lösungsansätze. Etablierte Technologiekonzerne wie Microsoft, Google und Amazon erweitern ihre bestehenden KI-Services um Agenten-Funktionalitäten, während gleichzeitig hunderte von Start-ups mit innovativen Speziallösungen auf den Markt drängen.

Diese Dynamik führt zu einer paradoxen Situation: Gerade in dem Moment, in dem Unternehmen dringend Orientierung benötigen, um fundierte Investitionsentscheidungen zu treffen, wird der Überblick über verfügbare Optionen nahezu unmöglich. Wer heute nach "AI-Agenten-Plattformen" sucht – sei es über Google, in Fachpublikationen oder durch Befragung von Large Language Models – erhält ein verwirrend uneinheitliches Bild.

Das Problem der fehlenden Orientierung

Verschiedene Recherchequellen liefern völlig unterschiedliche Anbieterempfehlungen. Was bei einer Google-Suche als "Top 10 AI-Agent Platforms" erscheint, hat oft wenig Überschneidung mit den Ergebnissen einer ähnlichen Suche am nächsten Tag oder auf einer anderen Plattform.

Die meisten verfügbaren Übersichten folgen keiner erkennbaren Systematik. Plattformen werden wahllos nebeneinandergestellt, ohne Berücksichtigung ihrer unterschiedlichen Zielgruppen, Funktionsumfänge oder Reifegrade.

Viele scheinbar objektive Marktübersichten sind tatsächlich von kommerziellen Interessen geprägt. Beratungsunternehmen bevorzugen Partner-Lösungen, Anbieter finanzieren gesponserte Inhalte, und Affiliate-Marketing führt zu einseitigen Empfehlungen.

Anders als in etablierten Technologiebereichen existieren bislang wenige umfassenden Bewertungen renommierter Analystenhäuser wie Gartner, Forrester oder IDC. Diese würden normalerweise durch ihre Magic Quadrants oder Wave-Reports Orientierung bieten, befinden sich aber selbst noch im Lernprozess bei diesem neuen Marktfeld. Bestehende Reports sind meist auf Teilaspekte beschränkt und umfassen nicht den Bereich „*Software, mit der ich AI-Agenten erstellen kann*“.

Zielsetzung dieses Dokuments

Genau hier setzt "Der AI-Agenten-Kompass" an. Dieses Dokument wurde entwickelt, um Entscheidungsträgern eine systematische, objektive und umfassende Orientierung im Markt für AI-Agenten-Plattformen zu bieten. Unser Ziel ist es, die bestehende Informationslücke zu schließen und eine strukturierte Grundlage für strategische Technologieentscheidungen zu schaffen.

Wir analysieren und bewerten hier die führenden Plattformen nicht nach willkürlichen Kriterien oder kommerziellen Interessen, sondern anhand klar definierter, praxisrelevanter Parameter.

Welche Anbieter - Gliederung

Um in der Vielfalt der verfügbaren Lösungen Orientierung zu schaffen, haben wir den Markt für AI-Agenten-Plattformen in vier klar abgrenzbare Kategorien strukturiert. Diese Gruppierung basiert auf den unterschiedlichen technologischen Ansätzen, Zielgruppen und Geschäftsmodellen der Anbieter. Die vier Hauptkategorien im Überblick:



Programmier-Plattformen: Open-Source Frameworks

Den technischen Kern der AI-Agenten-Entwicklung bilden spezialisierte Open-Source Frameworks, die auf etablierten Programmiersprachen wie Python basieren. Diese Plattformen ermöglichen es Entwicklern, hochkomplexe AI-Anwendungen und autonome Agenten auf Basis von Large Language Models (LLMs) zu erstellen.

Der entscheidende Vorteil dieser Kategorie liegt in ihrer Herstellerunabhängigkeit und Flexibilität. Open-Source Frameworks bieten maximale Gestaltungsfreiheit und vermeiden Vendor Lock-in-Situationen. Gleichzeitig haben sich um viele dieser Projekte kommerzielle Ökosysteme entwickelt, die professionellen Support, gehostete Versionen und Enterprise-Features anbieten, ohne die grundsätzliche Offenheit der Plattform zu beeinträchtigen.

Workflow-Agentic Software: Spezialisierte Anwendungen

Eine dynamisch wachsende Kategorie bilden Software-Anwendungen, die explizit für die Erstellung und Verwaltung von AI-Agenten entwickelt bzw. erweitert wurden. Dieser Bereich zeigt eine interessante Zweiteilung:

Einerseits haben etablierte Anbieter von Workflow-Management-Systemen und Chatbot-Plattformen ihre bestehenden Lösungen um AI-Agenten-Funktionalitäten erweitert. Diese Evolution bestehender Tools bringt den Vorteil bereits erprobter Infrastrukturen und Nutzerbasen mit sich.

Andererseits entstehen in rasantem Tempo völlig neue Produkte von Start-ups, die ausschließlich für den AI-Agenten-Markt konzipiert wurden. Diese "Native AI-Agent Plattformen" versprechen oft innovativere Ansätze und modernere Architekturen, müssen aber ihre Marktreife und Stabilität erst unter Beweis stellen. Nicht in die Betrachtung einbezogen wurden Softwarelösungen, die auf ein beschränktes Einsatzgebiet spezialisiert sind (Rechnungswesen, HR-Prozesse, Übersetzungen o.ä.) wie z.B. paradox.ai, smartcat oder vic.ai.

LLM-integrierte Lösungen: AI-Agenten aus einer Hand

Die führenden Large Language Model-Anbieter haben erkannt, dass AI-Agenten eine natürliche Erweiterung ihrer Kernkompetenz darstellen. Entsprechend haben OpenAI, Anthropic, Google und andere ihre Plattformen um mehr oder weniger umfangreiche Agenten-Funktionalitäten erweitert.

Diese Kategorie zeichnet sich durch die nahtlose Integration zwischen dem zugrundeliegenden Sprachmodell und den Agenten-Capabilities aus. Nutzer profitieren von optimierter Performance, vereinfachter Architektur und oft auch von Kostenvorteilen. Gleichzeitig entstehen aber auch Abhängigkeiten von einzelnen Modell-Anbietern.

Cloud- und Enterprise-Software-Ökosysteme: Integration in bestehende Landschaften

Die etablierten Cloud-Hyperscaler (Amazon Web Services, Microsoft Azure, Google Cloud) sowie führende Enterprise-Software-Anbieter wie z.B. SAP und Salesforce haben AI-Agenten-Komponenten in ihre umfassenden Plattform-Ökosysteme integriert.

Diese Ansätze zielen primär auf Unternehmen ab, die bereits stark in entsprechende Technologie-Stacks investiert haben. Der Hauptvorteil liegt in der nahtlosen Integration mit bestehenden Daten, Prozessen und Benutzeroberflächen. Gleichzeitig profitieren Nutzer von der Enterprise-grade Sicherheit, Compliance und Support-Strukturen dieser etablierten Anbieter.

Auswahl Software / Anbieter

Für diese umfassende Bewertung haben wir die marktführenden und strategisch relevantesten Plattformen in jeder der vier Kategorien einer detaillierten Analyse unterzogen und unsere Erfahrung aus der konkreten Anwendung diverser Lösungen bei der Entwicklung von AI-Agenten genutzt. Während eine vollständige Erfassung aller verfügbaren Lösungen angesichts der Marktdynamik praktisch unmöglich wäre, konzentriert sich unsere Untersuchung auf jene Anbieter, die wir aufgrund ihrer Marktposition, technologischen Innovation oder strategischen Bedeutung als repräsentativ für ihre jeweilige Kategorie eingeschätzt haben.

Diese Auswahl gewährleistet, dass alle wesentlichen Lösungsansätze und Geschäftsmodelle im AI-Agenten-Markt abgebildet werden und Entscheidungsträger eine fundierte Grundlage für ihre Technologiestrategie erhalten.



Programmier-Plattformen

Open -Source

- Langchain
- Langraph
- Langflow
- CrewAI
- AutoGPT

Workflow-Agentic Software

- N8N
- BeamAI
- Zapier
- Make

LLM-intern

- OpenAI – ChatGPT Agent builder
- Microsoft Copilot
- Google ADK Agent Development Kit
- Mistral Agents API
- Antropic – Claude Agent SDK

Cloud- and Enterprise Software-Eco-Systems

- Amazon Web Services
- Microsoft Azure
- Google Cloud
- IBM Watsonx
- Salesforce – Agentforce
- SAP - Joule
- Databricks - Mosaic AI

Schlüsselkompetenzen von Agentic AI Software

Viele Softwarelösungen werben heute mit dem Begriff „AI-Agenten“. Doch nicht überall, wo „Agent“ draufsteht, steckt auch echte Agentic AI dahinter. Damit eine Lösung wirklich als Agentic AI Software gelten kann, muss sie bestimmte Schlüsselfähigkeiten besitzen, die sie deutlich von klassischen Workflow- oder Automatisierungssystemen unterscheiden.

Um diese Unterschiede greifbar zu machen, haben wir drei zentrale Themenfelder definiert:

- **Was bedeutet Agentic AI eigentlich?** Wir klären, was ein System „agentisch“ macht – und woran sich echte agentische Intelligenz erkennen lässt.
- **Unterstützung von MCP-Servern (Model Context Protocol):** Wir betrachten, welche Rolle MCP-Unterstützung spielt, um Agenten mit Tools, APIs und Kontextdaten zu verbinden.
- **Integration von RAG und Vektordatenbanken:** Wir zeigen, warum der Zugriff auf externe Wissensquellen über Retrieval-Augmented Generation (RAG) und Vektordatenbanken entscheidend ist, um Kontextverständnis und Handlungsfähigkeit zu erweitern.

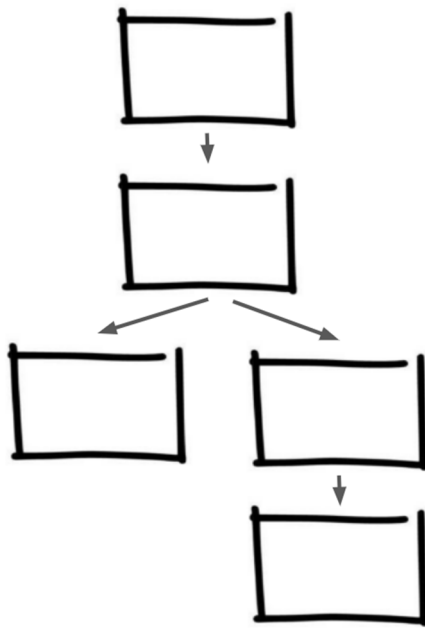
Agenten und Agentic AI

Software-Agenten gibt es schon seit vielen Jahren. Dabei handelt es sich in der Regel um fest programmierte Workflows, die Aufgaben automatisiert ausführen. Mit dem Aufkommen von Agentic AI – also KI-Systemen, die eigenständig planen, handeln und Entscheidungen treffen – verändert sich jedoch das Verständnis von „Agenten“ grundlegend.

Um den Unterschied zwischen klassischen Automatisierungen und modernen, selbststeuernden KI-Systemen zu verdeutlichen, lassen sich **vier Entwicklungsstufen** unterscheiden:

1. Klassischer Software-Agent mit fest programmiertem Workflow
2. Fest programmierter Workflow mit AI-Komponenten
3. Agentic AI Workflow
4. Agentic AI mit hierarchischer Agenten-Struktur

1. Klassischer Software-Agent mit fest programmiertem Workflow



Ein klassischer Software-Agent führt einen vordefinierten Ablauf (Workflow) Schritt für Schritt aus. Es besteht keine Entscheidungsfreiheit: Reihenfolge und eingesetzte Tools oder Methoden sind vollständig im Code festgelegt.

Zwar können Verzweigungen oder Bedingungen existieren, diese sind jedoch deterministisch programmiert und folgen stets denselben Regeln.

Beispiel:

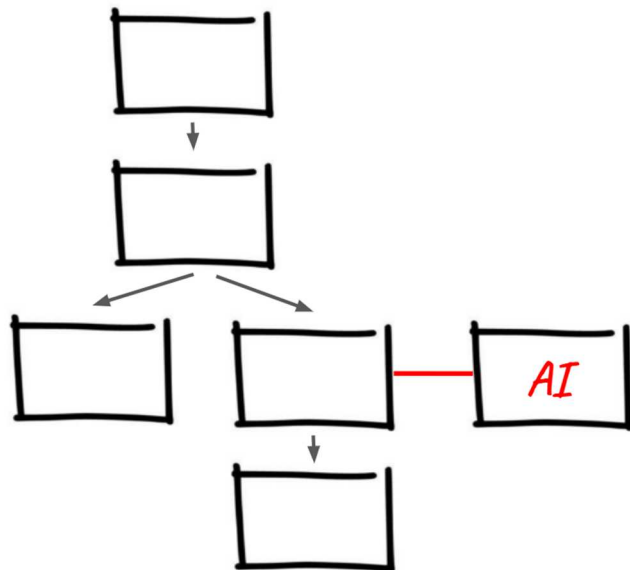
Eine Webanwendung generiert auf Basis eines Formulars automatisch ein Antragsdokument und leitet dieses an die zuständige Abteilung weiter. Der Ablauf bleibt dabei immer gleich.

2. Fest programmierter Workflow mit AI-Komponenten

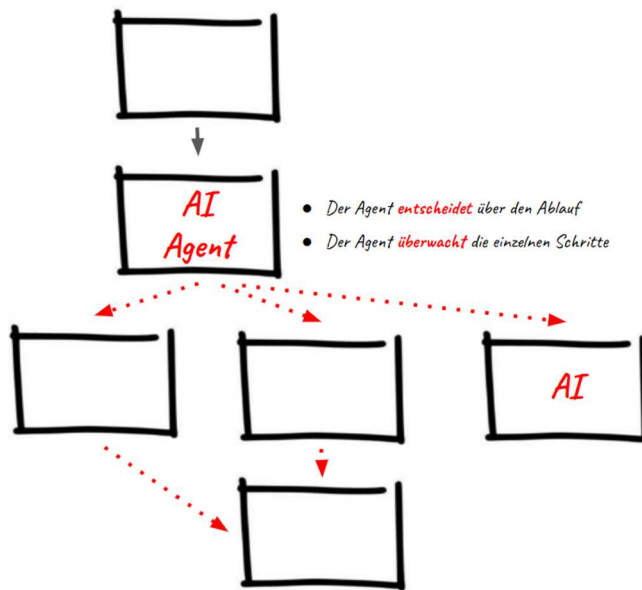
Der Ablauf bleibt auch hier fest definiert – allerdings werden einzelne Schritte durch KI-Module unterstützt. Diese Module übernehmen Teilaufgaben wie Klassifikation, Bilderkennung oder Textgenerierung. Die KI beeinflusst jedoch nicht die Prozesslogik und entscheidet nicht selbstständig, wie es weitergeht.

Beispiel:

Ein Workflow zur E-Mail-Analyse ruft in einem Arbeitsschritt eine KI auf, die eingehende Nachrichten klassifiziert. Danach läuft der Prozess – etwa Weiterleitung, Speicherung oder Antwortvorlage – nach festem Muster weiter.



3. Agentic AI Workflow



Hier beginnt der eigentliche Paradigmenwechsel: Ein oder mehrere KI-Agenten steuern den Prozess eigenständig. Anstatt einem starren Skript zu folgen, definiert der Agent auf Basis eines übergeordneten Ziels (Goal) dynamisch die notwendigen Zwischenschritte, wählt passende Tools (z. B. Suchmaschinen, Code-Interpreter, APIs) und überprüft kontinuierlich die Ergebnisse.

Der Agent überwacht den Ablauf, bewertet Teilergebnisse und passt die Strategie situativ an. Zielorientierung, Adaptivität und Reflexionsfähigkeit sind die zentralen Merkmale dieser Stufe.

Beispiel:

Ein AI-Agent analysiert eine Aufgabe, erkennt ein Problem, entscheidet eigenständig, ob er Informationen recherchiert, Text generiert oder eine Datenbankabfrage ausführt – und kombiniert diese Methoden flexibel.

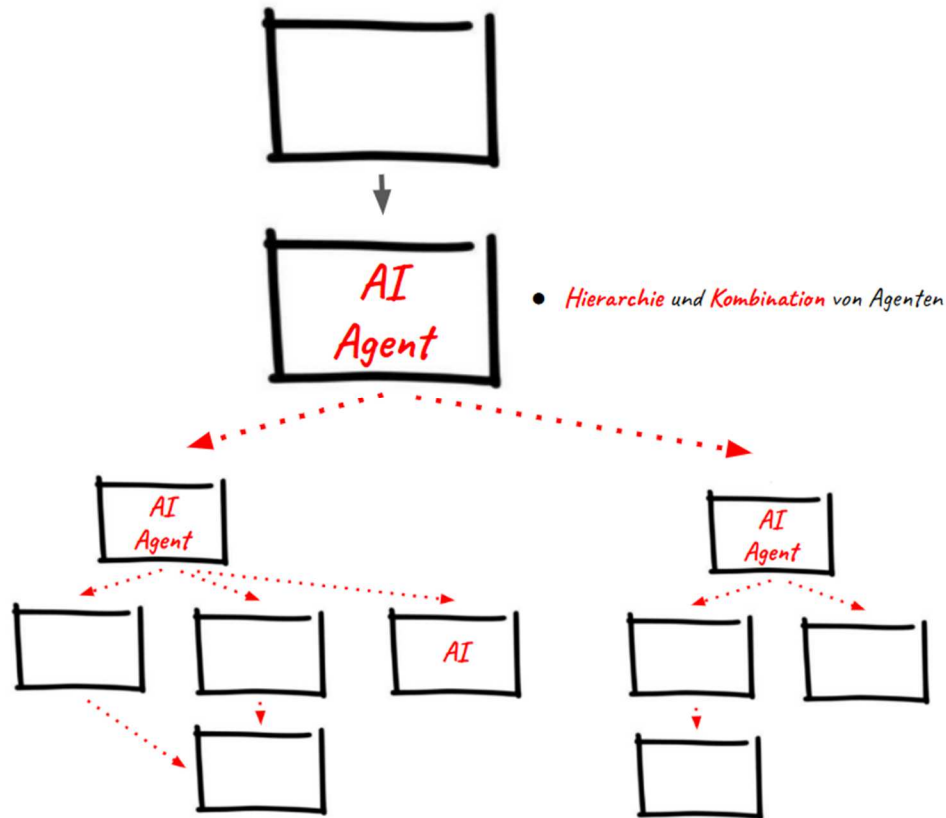
4. Agentic AI mit hierarchischer Agenten-Struktur

In dieser erweiterten Form arbeiten mehrere spezialisierte Agenten in einer hierarchischen Struktur zusammen. Ein übergeordneter Manager-Agent plant den Gesamtprozess, teilt Aufgaben in Teilaufgaben auf und delegiert sie an Subagenten mit klarer Spezialisierung (z. B. Recherche, Analyse, Schreiben, Qualitätssicherung). Diese Subagenten handeln wiederum autonom, bewerten ihre Ergebnisse und liefern sie an den Manager zurück.

So entsteht ein koordiniertes, arbeitsteiliges System, das komplexe Projekte bearbeiten kann – vergleichbar mit einem Team aus Fachkräften, die gemeinsam auf ein Ziel hinarbeiten.

Beispiel:

Ein Leit-Agent entwickelt eine Strategie, delegiert Recherche an einen Subagenten und Textproduktion an einen anderen. Die Ergebnisse werden überprüft, integriert und in die nächsten Schritte überführt.



Bewertungskriterien im Software-Kompass

Bei der Analyse und Bewertung der verschiedenen Softwarelösungen berücksichtigen wir insbesondere:

- Inwieweit **echte agentische Workflows** unterstützt werden (Planung, Tool-Nutzung, Reflexion)
- Ob und wie **hierarchische Agentenstrukturen** mit Rollen, Kommunikation und Koordination möglich sind

Diese Unterscheidung ist entscheidend, um den Reifegrad und die Autonomie verschiedener AI-Agenten Softwarelösungen richtig einzuordnen.

Unterstützung von MCP-Servern (Model Context Protocol)

Das Model Context Protocol (MCP) ist ein offener Standard, der darauf abzielt, die **Einbindung von Kontext, Tools und Datenquellen** in KI-Agenten systematisch und standardisiert zu ermöglichen. Analog zu einem „USB-C-Port“ für KI-Anwendungen: So wie USB-C es erlaubt, unterschiedlichste Hardware einfach anzuschließen, erlaubt MCP es KI-Agenten, heterogene Tools und Datenquellen auf einheitliche Weise anzubinden.

Das Model Context Protocol wurde ursprünglich von Anthropic als internes Tool entwickelt, um die Fähigkeiten des eigenen KI-Modells Claude zur Interaktion mit externen Systemen zu verbessern. Die Veröffentlichung als Open-Source-Projekt erfolgte strategisch, um eine breitere Akzeptanz zu fördern und die Kommunikation zwischen KI-Systemen und Tools zu standardisieren.

Bereits kurz nach der Ankündigung wurde das Protokoll von führenden KI-Anbietern wie OpenAI und Google DeepMind übernommen. Entwicklungstools-Unternehmen wie Zed, Replit, Codeium und Sourcegraph integrierten MCP in ihre Plattformen.

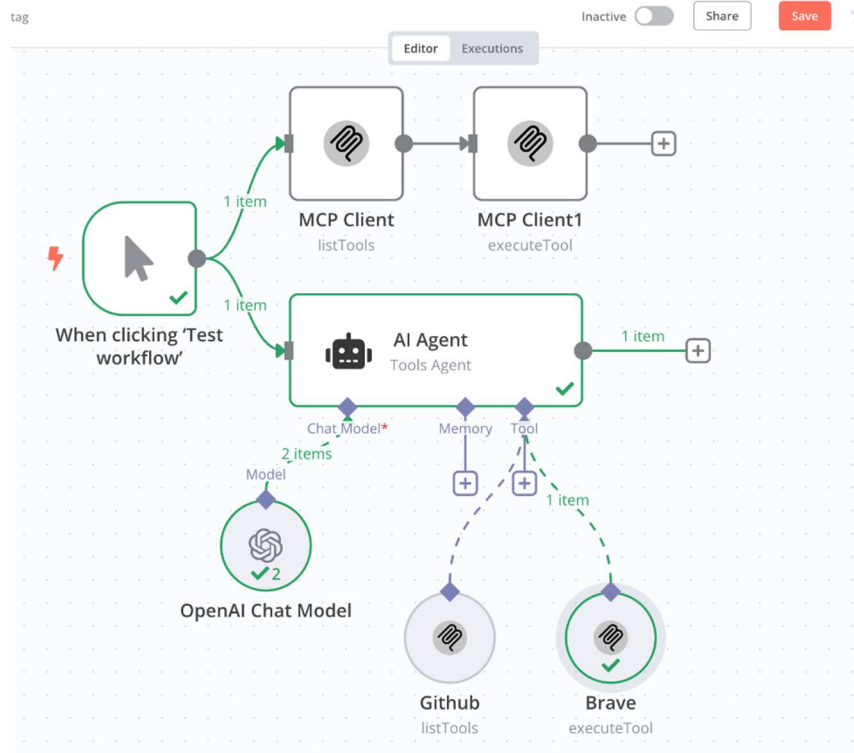
Vor MCP war es üblich, für jede Kombination aus Modell, Datenquelle oder Tool einen eigenen Connector zu schreiben – das sogenannte „ $M \times N$ Integrationsproblem“. MCP ersetzt dieses durch ein „ $M + N$ “ Modell: Werkzeuganbieter bauen N Server; Agentenentwickler bauen M Clients.

2. Rolle der MCP-Unterstützung im Agentic AI Kontext

In einem echten Agentic AI-System spielt MCP eine zentrale Rolle, da sie folgende Funktionen übernimmt:

- **Tool- und Datenquellen-Integration:** Ein MCP-Server kann Funktionen („Tools“), Datenressourcen („Resources“) und Prompt-Templates („Prompts“) bereitstellen, die ein Agent bei Bedarf nutzt.
- **Dynamische Entdeckung:** Der Agent (bzw. der MCP-Client) kann zur Laufzeit ermitteln, welche Tools und Ressourcen verfügbar sind, anstatt sie hart im Code aufzunehmen.
- **Verbindung von Agent und Umgebung:** Durch den Betrieb eines MCP-Servers hat das Agentensystem Zugriff auf externe APIs, Datenbanken, lokale Dateien oder Workflows – dies ermöglicht den Schritt von einem fixen Workflow hin zu einem selbststeuernden Agenten.
- **Skalierung & Modularität:** Weil Tools über den MCP-Standard angebunden werden, ist es einfacher, bestehende und neue Funktionalitäten einzubinden, ohne tiefgreifend die Agentenlogik anzupassen.

Somit ist die Unterstützung von MCP ein **Schlüsselmerkmal** für Agentic AI-Software: Ohne standardisierte Schnittstelle zur Außenwelt bleiben Agenten eher isoliert.



Beispiel MCP-Server Einbindung mit n8n

3. Bewertungsmaßstab bei Software-Paketen

Bei der Bewertung von Softwarelösungen, die Agentic AI-Fähigkeiten versprechen, haben wir MCP-Unterstützung in zwei Dimensionen betrachtet:

1. **Anbindung von externen MCP-Servern:** Kann die Software als Client existierende MCP-Server nutzen? Unterstützt sie die Verbindung zu verschiedenen MCP-Servern und damit Tools / Datenquellen? Diese Fähigkeit ist wichtig, damit Agenten Zugriff auf ein breites Spektrum von Werkzeugen haben.
2. **Bereitstellung eines eigenen MCP-Servers:** Ermöglicht die Lösung, selbst einen MCP-Server zu betreiben (z. B. um eigene Tools oder Datenquellen als Services anzubieten)? Diese Fähigkeit zeigt eine höhere Autonomie und Flexibilität: Die Lösung kann ihr eigenes Ökosystem bereitstellen. Zudem ist damit eine bessere Kontrolle über Sicherheits-, Governance- und Integrationsbelange möglich.

Wir berücksichtigen also sowohl die **Client-Seite** (Nutzen von MCP-Servern) als auch die **Server-Seite** (Betrieb eigener MCP-Server) als Bewertungskriterium. Softwarepakete, die beide Dimensionen gut abdecken, gelten als reifer im Hinblick auf echte Agentic AI-Fähigkeit.

Retrieval-Augmented Generation und Vektordatenbank-Integration

Für Enterprise-taugliche Agentic-AI-Systeme ist der Zugriff auf unternehmensspezifisches Wissen entscheidend. Nur wenn ein Agent die internen Dokumente, Datenquellen und Prozesse versteht, kann er qualitativ hochwertige, kontextbezogene Ergebnisse liefern. Hier kommt Retrieval-Augmented Generation (**RAG**) ins Spiel – ein Ansatz, der die Grenzen reiner Sprachmodelle überwindet.

1. Konzept von RAG

RAG kombiniert zwei Kernkomponenten:

- **Retrieval (Abruf):** Ein System durchsucht externe Wissensquellen, meist Vektordatenbanken, um relevante Informationen zum aktuellen Kontext oder zur Anfrage zu finden.
- **Generation (Erzeugung):** Ein Large Language Model (LLM) nutzt diese abgerufenen Informationen, um eine fundierte, aktuelle und kontextgenaue Antwort zu generieren.

Im Gegensatz zu rein generativen Modellen, die nur auf ihr Trainingswissen zurückgreifen, kann ein RAG-System dynamisch auf aktuelle, interne und vertrauliche Unternehmensdaten zugreifen – ohne das Modell selbst neu trainieren zu müssen.

2. Bedeutung für Agentic AI

In agentischen Systemen spielt RAG eine zentrale Rolle für Kontextualisierung und Entscheidungsfähigkeit:

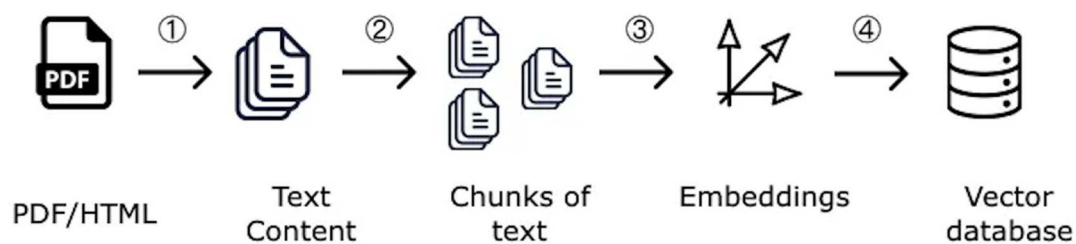
- **Wissensintegration:** Agenten können situationsabhängig Informationen abrufen, bewerten und in ihre Handlungsplanung einbeziehen.
- **Aktualität:** Entscheidungen basieren nicht nur auf Trainingsdaten, sondern auch auf tagesaktuellen, externen oder internen Quellen.
- **Transparenz:** RAG ermöglicht es, Quellen und Belege nachzuvollziehen – ein wichtiger Aspekt für Nachvollziehbarkeit und Compliance.
- **Adaptivität:** Der Agent kann selbstständig neue Informationen beschaffen, wenn er feststellt, dass sein aktuelles Wissen unzureichend ist.

Damit bildet RAG den Wissensmotor eines echten Agentic-AI-Systems – es macht aus einem sprachfähigen Modell einen wissensfähigen, anpassungsfähigen Agenten.

3. Technische Integration: Vektordatenbanken und Kontextmanagement

Die praktische Umsetzung von RAG setzt eine enge Integration mit Vektordatenbanken voraus. Diese speichern Dokumente, Code oder Daten in Form von Vektoren, sodass semantisch ähnliche Inhalte schnell gefunden werden können. Wichtige Aspekte dabei sind:

- **Einbettungsmodelle (Embeddings):** Sie wandeln Text in numerische Vektoren um, die in semantischen Räumen verglichen werden können.
- **Intelligente Retrieval-Strategien:** Der Agent muss entscheiden, wann und wie er welche Datenquellen abfragt – z. B. basierend auf Confidence Scores, Kosten oder Sicherheitsrichtlinien.
- **Kontext-Management:** Gefundene Informationen müssen in den aktiven Kontext des Agenten integriert werden, ohne den Prompt zu überladen oder irrelevante Daten einzubeziehen.



4. Bewertungskriterien im Software-Kompass

Bei der Bewertung von Softwarelösungen im Bereich Agentic AI haben wir daher untersucht:

- Unterstützt die Software die Anbindung an Vektordatenbanken (z. B. Pinecone, Weaviate, Milvus, FAISS, Chroma)?
- Gibt es integrierte Mechanismen für semantische Suche, Kontextverwaltung und Prompt-Optimierung?
- Wie flexibel ist das System bei der Kombination verschiedener Wissensquellen und Datenformate?

Software, die RAG-Mechanismen nativ integriert und eine intelligente Verwaltung des Wissenskontexts erlaubt, bietet die Grundlage für skalierbare, verlässliche und nachvollziehbare Agentic-AI-Anwendungen im Unternehmensumfeld.

Wie haben wir bewertet

Zur objektiven Vergleichbarkeit der analysierten Plattformen haben wir ein systematisches Bewertungsframework entwickelt, das sieben zentrale Evaluationsdimensionen umfasst. Jede Lösung wurde in allen Kategorien auf einer Skala von 1 bis 10 bewertet, wobei 10 die bestmögliche Bewertung darstellt.



Die Kategorien sind:

- **Funktionsumfang** – Wie vollständig und vielseitig sind die angebotenen Funktionen für den vorgesehenen Einsatzzweck?
- **Offenheit / Open Source** – Ist man auf ein bestimmtes Ecosystem beschränkt oder ist die Lösung offen für andere Umgebungen? In welchem Maß ist der Quellcode offen, lizenzfrei nutzbar und transparent?
- **Anpassbarkeit** – Wie gut lässt sich die Software konfigurieren, erweitern oder individualisieren? Kann man eigenen Programmcode (z.B. Python) verwenden.
- **Verbreitung & Community** – Wie groß und aktiv ist die Nutzerbasis und wie gut ist die Unterstützung durch eine Community?
- **Erlernbarkeit** – Wie schnell und intuitiv können neue Anwender die Software produktiv nutzen (Lernkurve)?
- **Enterprise-Tauglichkeit** – Wie gut erfüllt die Software Anforderungen großer Organisationen in Bezug auf Skalierbarkeit und Leistung?
- **Sicherheit & Governance** – Wie zuverlässig sind Sicherheitsfunktionen, Compliance und administrative Steuerungsmöglichkeiten?

Funktionsumfang

Bewertung des Funktionsumfangs: Die entscheidende Dimension

Als zentrale Bewertungskategorie verdient der Funktionsumfang eine detaillierte Betrachtung. Dabei liegt der Fokus nicht auf beliebigen Features, sondern spezifisch auf jenen Capabilities, die echte AI-Agenten von traditionellen, statisch programmierten Workflows oder einfachen AI-unterstützten Anwendungen wie Chatbots unterscheiden.

Abgrenzung: Echte AI-Agenten vs. traditionelle Automatisierung

Ein wesentlicher Aspekt unserer Bewertung liegt in der klaren Unterscheidung zwischen verschiedenen Automatisierungsansätzen. Während klassische Workflows festen, vordefinierten Pfaden folgen und AI-unterstützte Tools wie Chatbots primär reaktiv agieren, zeichnen sich echte AI-Agenten durch autonome Entscheidungsfindung, dynamische Anpassung und proaktive Problemlösung aus.

Die acht Kernkriterien für AI-Agenten-Funktionalität



1. Anzahl unterstützter LLMs: Die Fähigkeit, verschiedene Large Language Models einzubinden und je nach Anwendungsfall das optimale Modell zu wählen. Dies umfasst sowohl die technische Integration unterschiedlicher API-Provider als auch intelligente Model-Routing-Mechanismen, die basierend auf Aufgabentyp, Kosten oder Performance-Anforderungen das geeignete LLM auswählen.

2. Multi-Agent-Architekturen und Hierarchien: Moderne AI-Agenten-Systeme bestehen selten aus isolierten Einzelagenten. Vielmehr erfordern komplexe Anwendungsfälle koordinierte Agent-Ensembles mit definierten Rollen, Verantwortlichkeiten und Kommunikationsprotokollen. Dies schließt sowohl horizontale Kollaboration als auch hierarchische Strukturen mit Supervisor- und Worker-Agenten ein.

3. Klassische Workflow-Funktionalität: Obwohl AI-Agenten über traditionelle Workflows hinausgehen, müssen sie dennoch nahtlos mit bestehenden, deterministischen Prozessen interagieren können. Dies umfasst die Einbettung von Regel-basierten Logiken, Conditional Branching und strukturierten Datenverarbeitungsschritten in ansonsten autonome Agent-Prozesse.

4. Anzahl API und Schnittstellen: Die Stärke von AI-Agenten liegt in ihrer Fähigkeit, als Orchestratoren verschiedener Services und Datenquellen zu fungieren. Daher ist die Unterstützung vielfältiger APIs, Webhooks und Integrationsprotokollen essentiell für praktische Anwendbarkeit.

5. MCP Unterstützung: Als emerging Standard für Client-Server-Kommunikation in AI-Systemen gewinnt MCP zunehmend an Bedeutung. Plattformen, die MCP unterstützen, können flexibler mit externen Ressourcen kommunizieren und sind besser auf zukünftige Standardisierungen vorbereitet. Ein weiterer Punkt ist die Frage, ob die Plattform nur **als MCP-Client** fungieren und somit Tools von angebundenen **MCP-Servern** nutzen kann, oder geht sie noch darüber hinaus und ermöglicht es den erstellten AI-Agenten, selbst als MCP-Server zu agieren und ihre Capabilities anderen MCP-Clients zur Verfügung zu stellen? Das können übrigens die wenigsten der untersuchten Software-Lösungen (siehe Abschnitt oben).

6. Retrieval-Augmented Generation (RAG) und Vektordatenbank-Integration: Für Enterprise-taugliche AI-Agenten ist der Zugriff auf unternehmensspezifisches Wissen durch RAG-Systeme unverzichtbar. Dies erfordert nicht nur die technische Integration von Vektordatenbanken, sondern auch intelligente Retrieval-Strategien und Kontext-Management (siehe Abschnitt oben).

7. Session-übergreifendes Gedächtnis: Anders als stateless Interactions sollten AI-Agenten Kontext und Lernfortschritt über einzelne Interaktionen hinaus bewahren. Dies umfasst sowohl kurzfristiges Arbeitsgedächtnis als auch langfristige Präferenz- und Verhaltensspeicherung.

8. Persistente Workflow-Steuerung Komplexe Agent-Prozesse erstrecken sich oft über längere Zeiträume und müssen Unterbrechungen, System-Restarts und asynchrone Operationen überdauern. Die Fähigkeit zur session-übergreifenden Prozesssteuerung ist daher ein Qualitätsmerkmal professioneller AI-Agenten-Plattformen.

Jede Softwarelösung wurde in allen acht Kriterien einzeln bewertet und in einer Gesamtbewertung der Kategorie „Funktionsumfang“ zusammengefasst.

Offenheit / Open Source

Die Offenheit einer AI-Agenten-Plattform ist ein vielschichtiges Kriterium, das mehrere Dimensionen umfasst. Zunächst ist die **Ökosystem-Neutralität** zu bewerten: Ist die Lösung auf ein spezifisches Technologie-Ökosystem beschränkt oder kann sie flexibel in verschiedenen Umgebungen deployed werden? Dies betrifft sowohl Cloud-Provider-Abhängigkeiten als auch Integrationsmöglichkeiten mit unterschiedlichen Tech-Stacks.

Darüber hinaus spielt der **Grad der Quellcode-Offenheit** eine entscheidende Rolle. Hier ist zwischen verschiedenen Stufen zu differenzieren: vollständig proprietäre Lösungen, Open-Core-Modelle mit gemischten Lizenzierungsansätzen, und vollständig Open-Source-Projekte. Besonders relevant sind dabei die spezifischen Lizenzmodelle (MIT, Apache 2.0, GPL, etc.) und deren Auswirkungen auf kommerzielle Nutzung und Modifikationsrechte.

Die **Transparenz und Community-Beteiligung** bildet eine weitere wichtige Facette. Können Nutzer den Code einsehen, Bugs selbständig beheben, Features beitragen oder sind sie auf die Roadmap und Reaktionszeiten des Anbieters angewiesen? Diese Faktoren haben direkte Auswirkungen auf Vendor Lock-in-Risiken, Anpassbarkeit und langfristige strategische Kontrolle über die gewählte Technologie.

Anpassbarkeit

„Die Anpassbarkeit einer AI-Agenten-Plattform bestimmt maßgeblich, inwieweit sie an spezifische Unternehmensanforderungen adaptiert werden kann.“

Dieses Kriterium umfasst mehrere kritische Aspekte der Flexibilität und Erweiterbarkeit

Konfigurationstiefe bildet die erste Bewertungsdimension: Können grundlegende Verhaltensweisen, Workflows und Benutzeroberflächen durch administrative Einstellungen angepasst werden, oder ist man auf vordefinierte Templates beschränkt? Hochwertige Plattformen bieten granulare Konfigurationsmöglichkeiten für Agent-Verhalten, Entscheidungslogiken und Interaktionsparadigmen.

Die **Integration eigener Programmlogik** stellt einen entscheidenden Differenzierungsfaktor dar. Können Entwickler benutzerdefinierten Code in Sprachen wie Python, JavaScript oder anderen einbetten, um spezifische Geschäftslogik zu implementieren? Dies umfasst sowohl die Möglichkeit, Custom Functions zu definieren, als auch die nahtlose Integration externer Libraries und Frameworks.

Erweiterbarkeit durch Plugins und Module erweitert diese Flexibilität auf Architekturebene. Unterstützt die Plattform modulare Erweiterungen, Third-Party-Plugins oder Custom Components? Wie ausgereift sind die verfügbaren APIs und SDKs für die Entwicklung eigener Erweiterungen?

Darüber hinaus ist die **Templating- und White-Label-Fähigkeit** relevant: Können Benutzeroberflächen, Branding und User Experience an Corporate Design Guidelines angepasst werden? Für Enterprise-Deployment ist oft auch die Möglichkeit wichtig, die Plattform als Teil eigener Produktangebote zu integrieren.

Schließlich spielt die **Datenmodell-Flexibilität** eine wichtige Rolle: Können eigene Datenstrukturen definiert, bestehende Schemas erweitert und komplexe Datenverarbeitungslogiken implementiert werden, ohne die Kernfunktionalität der Plattform zu beeinträchtigen?

Verbreitung & Community

Die Größe und Aktivität der Nutzerbasis sowie die Qualität der Community-Unterstützung sind oft unterschätzte, aber strategisch wichtige Faktoren bei der Plattform-Auswahl. Eine starke Community kann den Unterschied zwischen einer erfolgreichen Implementierung und kostspieligen Entwicklungsengpässen bedeuten.

Nutzerbasengröße und Marktdurchdringung bilden die quantitative Grundlage dieser Bewertung. Größere Nutzerbasen signalisieren nicht nur Marktvalidierung und Stabilität des Anbieters, sondern auch eine höhere Wahrscheinlichkeit für kontinuierliche Weiterentwicklung und langfristige Unterstützung. Gleichzeitig reduziert eine breite Adoption das Risiko von Vendor Lock-in-Situationen und erhöht die Verfügbarkeit qualifizierter Entwickler am Arbeitsmarkt.

Die **Community-Aktivität** lässt sich an verschiedenen Indikatoren messen: Häufigkeit und Qualität von Beiträgen in Foren und Diskussionsplattformen, Anzahl und Aktivität von GitHub-Repositories, Frequenz von Community-Events und Meetups sowie die Reaktionsgeschwindigkeit bei Support-Anfragen. Aktive Communities entwickeln oft ein Eigenleben, das wertvolle Ressourcen, Best Practices und innovative Anwendungsfälle hervorbringt.

„Die Größe und Aktivität der Nutzerbasis sind oft unterschätzte, aber strategisch wichtige Faktoren bei der Plattform-Auswahl.“

Verfügbarkeit von Lernressourcen ist ein weiterer kritischer Aspekt: Existieren umfassende Dokumentationen, Video-Tutorials, Online-Kurse und praktische Beispiele? Werden diese Ressourcen sowohl vom Anbieter als auch von der Community gepflegt und aktuell gehalten? Die Qualität und Quantität verfügbarer Lernmaterialien beeinflusst direkt die Einarbeitungszeit und Implementierungskosten.

Die **Ökosystem-Entwicklung** zeigt sich in der Verfügbarkeit von Third-Party-Tools, Integrationen, Templates und vorgefertigten Lösungen. Starke Communities entwickeln oft umfangreiche

Marktplätze für Erweiterungen, was die Entwicklungszeit für spezifische Anwendungsfälle erheblich reduzieren kann.

Schließlich ist die **Qualität des Community-Supports** zu bewerten: Wie hilfsbereit und kompetent sind Community-Mitglieder bei der Problemlösung? Existieren etablierte Mentoring-Programme oder Support-Strukturen? In vielen Fällen kann qualitativ hochwertiger Community-Support sogar den offiziellen Vendor-Support übertreffen und zu schnelleren Problemlösungen führen.

„Eine steile Lernkurve kann selbst technisch überlegene Lösungen zu kostspieligen Investitionen werden lassen.“

Erlernbarkeit

Die Erlernbarkeit einer AI-Agenten-Plattform ist ein kritischer Erfolgsfaktor, der direkte Auswirkungen auf Implementierungsgeschwindigkeit, Adoptionsrate und Total Cost of Ownership hat. Eine steile Lernkurve kann selbst technisch überlegene Lösungen zu kostspieligen Investitionen werden lassen.

Intuitivität der Benutzeroberfläche bildet die erste Bewertungsebene: Folgt das Interface etablierten Design-Patterns und Usability-Standards? Können neue Nutzer ohne umfangreiche Schulungen grundlegende Funktionen verstehen und ausführen? Besonders relevant ist hier die Balance zwischen funktionaler Vollständigkeit und übersichtlicher Darstellung – komplexe AI-Agenten-Systeme neigen dazu, Benutzer durch Feature-Überladung zu überfordern.

Die **Verfügbarkeit und Qualität von Onboarding-Prozessen** ist entscheidend für den ersten Eindruck: Existieren interaktive Tutorials, Guided Tours oder Wizard-basierte Setup-Prozesse? Wie effektiv führen diese neuen Nutzer von der ersten Anmeldung zu produktiven Anwendungsfällen? Hochwertige Plattformen bieten oft rollenbezogene Onboarding-Pfade für verschiedene Nutzertypen (Entwickler, Business User, Administratoren).

Dokumentationsqualität und Lernressourcen-Strukturierung spielen eine zentrale Rolle bei der langfristigen Produktivität: Sind Dokumentationen aktuell, vollständig und verständlich geschrieben? Existieren praktische Beispiele, Step-by-Step-Anleitungen und Use-Case-spezifische Tutorials? Die beste Plattform ist nur so gut wie ihre Dokumentation, wenn Nutzer komplexere Funktionen erschließen möchten.

Die **Abstraktionsebene und Komplexitätsgraduierung** beeinflusst maßgeblich die Zugänglichkeit für verschiedene Skill-Level: Bietet die Plattform sowohl Low-Code/No-Code-Ansätze für Business User als auch programmatische Interfaces für erfahrene Entwickler? Können Nutzer mit einfachen Anwendungsfällen beginnen und schrittweise zu komplexeren Implementierungen fortschreiten?

Fehlererkennung und Debugging-Unterstützung sind oft unterschätzte Aspekte der Erlernbarkeit: Wie verständlich sind Fehlermeldungen und Warnungen? Existieren integrierte Debugging-Tools, Validierungshelfer oder Syntax-Checker? Gute Plattformen verwandeln potenzielle Frustrationserlebnisse in Lernmöglichkeiten durch konstruktive Fehlerbehandlung.

Schließlich ist die **Skill-Transferierbarkeit** zu berücksichtigen: Können bestehende Kenntnisse in verwandten Technologien (Workflow-Tools, Programmiersprachen, Cloud-Plattformen) genutzt werden, oder erfordert die Plattform das Erlernen völlig neuer Konzepte und Paradigmen? Plattformen, die auf bekannten Standards und Methoden aufbauen, reduzieren die Einarbeitungszeit erheblich.

„Die Enterprise-Tauglichkeit einer AI-Agenten-Plattform entscheidet darüber, ob sie den spezifischen Anforderungen großer Organisationen gerecht werden kann.“

Enterprise-Tauglichkeit

Die Enterprise-Tauglichkeit einer AI-Agenten-Plattform entscheidet darüber, ob sie den spezifischen Anforderungen großer Organisationen gerecht werden kann. Diese Anforderungen gehen weit über die reine Funktionalität hinaus und umfassen kritische Aspekte der Skalierbarkeit, Performance und organisatorischen Integration.

Skalierbarkeit und Lastverteilung bilden das technische Rückgrat enterprise-tauglicher Lösungen: Kann die Plattform horizontal skalieren, um wachsende Benutzeranzahlen und Workloads zu bewältigen? Existieren automatische Load-Balancing-Mechanismen und elastische Ressourcenzuteilung? Besonders kritisch ist die Fähigkeit, tausende gleichzeitiger Agent-Instanzen zu orchestrieren, ohne Performance-Degradation zu erleiden.

Die **Multi-Tenancy-Architektur** ist für große Organisationen essentiell: Können verschiedene Geschäftsbereiche, Tochtergesellschaften oder Projektgruppen isoliert auf derselben Plattform-Instanz arbeiten? Wie granular sind Ressourcen-Isolation, Daten-Trennung und Zugriffskontrollen? Enterprise-grade Lösungen bieten oft hierarchische Mandanten-Strukturen mit vererbten Policies und zentraler Governance.

Integration in bestehende IT-Landschaften stellt einen weiteren kritischen Faktor dar: Unterstützt die Plattform Enterprise-Standards wie SAML, LDAP, Active Directory oder moderne Identity Provider? Können AI-Agenten nahtlos mit ERP-Systemen, CRM-Plattformen, Data Warehouses

und anderen Kernsystemen interagieren? Die Qualität und Vollständigkeit der API-Landschaft entscheidet oft über Erfolg oder Scheitern von Enterprise-Implementierungen.

Monitoring, Logging und Observability sind für den produktiven Betrieb unverzichtbar: Bietet die Plattform umfassende Metriken zu Agent-Performance, Ressourcenverbrauch und Geschäftskennzahlen? Existieren integrierte Dashboards, Alerting-Systeme und die Möglichkeit zur Integration in bestehende Monitoring-Infrastrukturen wie Prometheus, Grafana oder enterprise SIEM-Systeme?

Die **Hochverfügbarkeit und Disaster Recovery**-Fähigkeiten runden die Enterprise-Anforderungen ab: Unterstützt die Plattform Multi-Region-Deployments, automatische Failover-Mechanismen und konsistente Backup-Strategien? Wie schnell können kritische Agent-Workflows nach Systemausfällen wiederhergestellt werden?

Sicherheit & Governance

Sicherheit und Governance sind nicht optional, sondern existentielle Anforderungen für Enterprise-AI-Implementierungen. In einer Zeit zunehmender Cyber-Bedrohungen und strenger Compliance-Vorgaben können Sicherheitslücken oder Governance-Defizite katastrophale Folgen haben.

Datenverarbeitung und -schutz stehen im Zentrum sicherheitskritischer Bewertungen: Wo und wie werden sensible Daten verarbeitet und gespeichert? Unterstützt die Plattform End-to-End-Verschlüsselung, sowohl im Transit als auch im Ruhezustand? Können AI-Agenten so konfiguriert werden, dass sie ausschließlich on-premise oder in spezifischen geografischen Regionen operieren, um Datenlokalisierungsanforderungen zu erfüllen?

Identitäts- und Zugriffsmanagement (IAM) bildet die Grundlage sicherer AI-Agent-Deployments: Bietet die Plattform granulare Rollen- und Rechtesysteme? Können Zugriffe auf Agent-Funktionen, Datenquellen und administrative Bereiche präzise gesteuert werden? Besonders wichtig ist die Unterstützung für Zero-Trust-Architekturen und die Integration mit Enterprise-IAM-Systemen.

„Sicherheit und Governance sind nicht optional, sondern existentielle Anforderungen für Enterprise-AI-Implementierungen.“

Compliance und Zertifizierungen sind in regulierten Branchen kritisch: Erfüllt die Plattform relevante Standards wie GDPR, HIPAA, SOX, ISO 27001 oder branchenspezifische Vorgaben? Existieren entsprechende Auditierungsberichte und Compliance-Dokumentationen? Können Compliance-Anforderungen durch konfigurierbare Policies automatisch durchgesetzt werden?

Die **Auditierbarkeit und Nachverfolgbarkeit** aller Agent-Aktivitäten ist sowohl für Security als auch für Compliance essentiell: Werden alle Agent-Entscheidungen, Datenverarbeitungsschritte und Systeminteraktionen lückenlos protokolliert? Können diese Logs für forensische Analysen und Compliance-Nachweise genutzt werden? Wie transparent sind die Entscheidungsprozesse der AI-Agenten für menschliche Auditoren?

Governance-Frameworks und Policy-Management ermöglichen die zentrale Steuerung komplexer AI-Agent-Landschaften: Können organisationsweite Richtlinien für Agent-Verhalten, Datennutzung und Interaktionsmuster definiert und automatisch durchgesetzt werden? Existieren Approval-Workflows für kritische Agent-Aktionen und Change-Management-Prozesse für Policy-Updates?

Schließlich ist die **Incident Response und Security Operations**-Integration zu bewerten: Wie schnell können Sicherheitsvorfälle erkannt, isoliert und behoben werden? Unterstützt die Plattform die Integration in bestehende Security Operations Center (SOC) und kann sie bei Verdacht auf Kompromittierung automatisch Schutzmaßnahmen einleiten?

Ergebnis

Wir haben die untersuchten Softwarelösungen nach bestem Wissen bewertet, wobei auch eigene praktische Erfahrungen mit vielen der Lösungen eingeflossen sind. Eine vollkommen objektive und abschließende Bewertung war angesichts der zeitlichen und personellen Rahmenbedingungen nicht möglich – und auch nicht das Ziel. Vielmehr ging es darum, einen umfassenden Überblick sowie eine strukturierte Einordnung in der unübersichtlichen Landschaft der verfügbaren Plattformen zu bieten. Einzelbewertungen stehen daher nicht im Mittelpunkt; der Bewertungsprozess ist jedoch transparent dokumentiert und kann jederzeit durch eigene Gewichtungen oder Punktevergaben angepasst werden.

Gewichtung der Bewertungskriterien

Da nicht alle Bewertungsdimensionen gleichermaßen entscheidungsrelevant sind, haben wir den **Funktionsumfang** mit dem Faktor 4 übergewichtet, da die verfügbaren Features und Capabilities maßgeblich über die praktische Einsetzbarkeit einer Plattform entscheiden.

Die Gewichtungsfaktoren spiegeln typische Priorisierungen in Unternehmensentscheidungen wider, können jedoch je nach spezifischen Anforderungen und strategischen Schwerpunkten angepasst werden. Der Gesamtscore jeder Plattform errechnet sich aus der Multiplikation von Einzelbewertung und jeweiligem Gewichtungsfaktor.

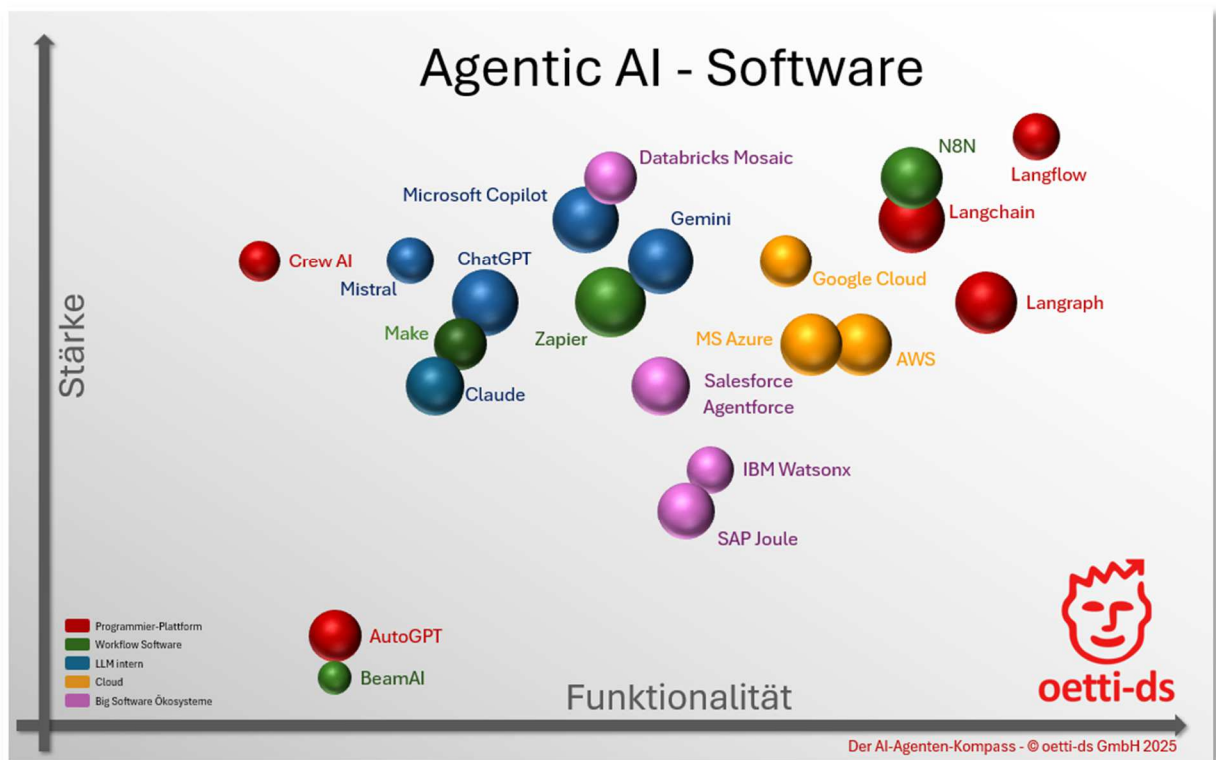
	Gesamt Score	Funktionsumfang (x 4)	Multi-LLM-Integration	Multi-Agent-Architekturen	Workflow-Integration	API- und Schnittstellen	MCP Client / Server	RAG-Integration	Session-übergf. Gedächtnis	Persistente Workflow-Steuerung	Offenheit Open Source	Anpassbarkeit	Verbreit. & Community	Leichte Erlernbarkeit	Enterprise tauglich	Sicherheit Governan.
Programmier-Plattformen																
Langchain	77	7,5	10	9	7	7	7	7	7	6	10	10	8	5	7	7
Langgraph	75,5	7,875	9	10	8	7	7	4	9	9	10	10	7	3	7	7
Langflow	77,5	8,125	10	9	7	8	10	9	5	7	9	9	4	9	7	7
Crew AI	58	4,25	4	8	7	4	1	1	2	7	10	8	3	6	7	7
AutoGPT	52,5	4,625	5	4	7	4	1	2	7	7	10	7	5	5	4	3
Workflow-Agentic Software																
N8N	77	7,5	8	7	8	9	8	7	7	6	6	9	7	7	9	9
BeamAI	48,5	4,625	2	8	3	4	4	1	7	8	1	2	2	7	9	9
Zapier	70	6	3	3	10	10	8	1	5	8	1	10	9	8	9	9
Make	62	5,25	6	2	10	10	2	1	4	7	2	9	5	9	8	8
LLM-Intern																
ChatGPT	66,5	5,375	3	5	6	4	5	6	6	8	4	6	8	7	10	10
Claude	61,5	5,125	3	4	4	4	8	4	6	8	2	8	6	5	10	10
Microsoft Copilot	70,5	5,875	5	8	5	4	6	4	7	8	4	8	8	7	10	10
Google Gemini	71	6,25	5	8	3	4	7	7	8	8	4	7	8	7	10	10
Mistral Agents API	62	5	2	8	4	4	5	4	6	7	8	6	4	6	9	9
Cloud- and Enterprise																
Amazon Web Services	72	7,25	8	7	7	7	7	5	8	9	4	7	7	5	10	10
Microsoft Azure	71	7	9	8	7	4	7	4	8	9	1	8	7	7	10	10
Google Cloud	70,5	6,875	4	9	7	5	7	6	8	9	5	7	5	6	10	10
IBM Watsonx	63	6,5	5	5	4	5	8	8	8	9	1	7	4	5	10	10
Salesforce	66	6,25	2	6	5	6	7	8	8	8	3	6	6	6	10	10
SAP Joule	63,5	6,375	5	5	5	5	7	7	9	8	2	5	6	5	10	10
Databricks	69	6	4	5	5	4	7	7	8	8	8	7	5	5	10	10

Visualisierung der Ergebnisse

Zur übersichtlichen Darstellung der Bewertungsergebnisse nutzen wir ein zweidimensionales Matrix-Format:

- **X-Achse:** Funktionsumfang (aufgrund der höchsten strategischen Relevanz)
- **Y-Achse:** Durchschnittswert aller übrigen Bewertungskategorien (exkl. Verbreitung & Community)
- **Punktgröße:** Repräsentiert die Nutzeranzahl bzw. Marktdurchdringung der jeweiligen Plattform – bezogen auf die jeweilige AI-Agenten Komponente. (Verbreitung & Community)

Diese Darstellung ermöglicht es, sowohl die funktionalen Stärken als auch die Gesamtpformance der Plattformen auf einen Blick zu erfassen, während gleichzeitig die Marktrelevanz durch die Visualisierung der Nutzerbasis berücksichtigt wird.



Zwar ergibt die Bewertung einen „Testsieger“ mit der höchsten Gesamt-Punktzahl, eine generelle Empfehlung lässt sich daraus aber nicht ableiten. Die Wahl der passenden Lösung hängt stets von den individuellen Anforderungen und dem Ausgangskontext ab. In der Praxis ist zudem wahrscheinlich, dass unterschiedliche Plattformen parallel genutzt werden.

Unsere differenzierte Empfehlung lautet daher:

- **Unabhängige, generische Plattform für AI-Agenten:** Langflow bietet sich als vielseitige Low-Code-Lösung mit vollständiger Python-Basis, hoher Flexibilität und zahlreichen Schnittstellen an. Für Teams mit tiefem Entwickler-Know-how, die Low-Code-Ansätze bewusst vermeiden möchten, sind auch die Kombinationen aus LangChain, LangGraph und LangSmith eine solide Wahl. Die in Deutschland entwickelte Software n8n erfreut sich steigender Beliebtheit und bietet einen sehr großen Funktionsumfang bei gleichzeitig vergleichsweise einfacher Benutzung.
- **Workflows über verschiedene Anwendungen hinweg:** Für die schnelle und stabile Automatisierung bieten sich etablierte Tools wie Zapier oder make an. Dabei entstehen jedoch keine „echten“ hierarchischen AI-Agenten, sondern vor allem Software-Workflows – maximal mit *AI-Unterstützung*. Wenn dies für den Anwendungsfall ausreicht, ist diese Lösung aufgrund ihrer Einfachheit und Stabilität oft sogar vorzuziehen.
- **Agenten-Funktionalitäten innerhalb großer LLMs:** Diese eignen sich weniger für systemübergreifende Aufgaben. Wenn jedoch ein Enterprise-Zugang zu einem Pro-Modell bereits besteht und die Aufgabe innerhalb des Chatmodells lösbar ist, können AI-Agenten hier sehr einfach und benutzerfreundlich eingerichtet werden, zumal die Funktionalität laufend ausgebaut wird.
- **Spezialisierte Lösungen innerhalb von Softwaresystemen:** Wenn der Einsatz ausschließlich in einem bestimmten Ökosystem erfolgt (z. B. ERP/SAP oder CRM/Salesforce), sind die dort angebotenen Lösungen wie SAP Joule oder Agentforce die erste Wahl.
- **Cloud-Infrastrukturanbieter und Databricks:** Diese Plattformen stellen umfassende Funktionalitäten bereit und punkten insbesondere durch Enterprise-Stabilität und Sicherheitsstandards. Allerdings sind die Angebote oft schwer zu überblicken, da regelmäßig neue Frameworks und Tools mit teils überschneidenden Funktionen erscheinen. Zudem ist ein gewisser Vendor-Lock-in unvermeidlich, sobald man sich in einer solchen Plattform bewegt. Wenn jedoch bereits Know-how für die jeweilige Umgebung vorhanden ist, lohnt sich die Beschäftigung mit diesen Lösungen in jedem Fall.

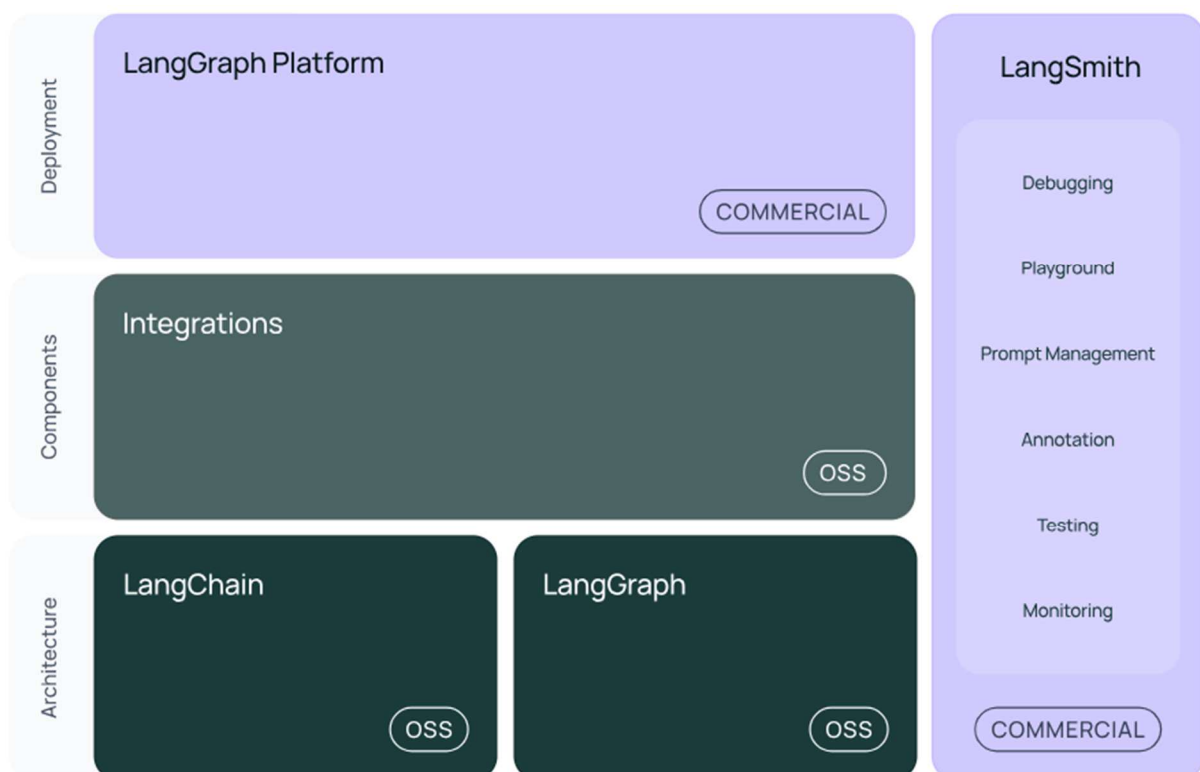
Einzelbewertungen:

 LangChain	 n8n	 ChatGPT AGENT	 Amazon Bedrock	 watsonx IBM
 LangGraph	 beam	 Gemini Vertex AI	 VERTEX AI	 Joule SAP
 Langflow	 zapier	 Copilot Semantic Kernel	 Azure AI Agent Service	 Agentforce Salesforce
 AUTOgpt	 make	 MISTRAL AI		 mosaic ^{ML} databricks
 crewai		 Claude Agent SDK		



Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

LangChain ist ein führendes Open-Source-Framework zur Entwicklung von AI-Agenten, bei dem LLMs (Large Language Models) die Kontrolle über den Ablauf von Anwendungen übernehmen (z.B. Tool-Aufrufe zur Lösung von Aufgaben). Es bietet eine dedizierte “Agents”-Modularität, mit der Entwickler Tools, Prompts und Aktionen kombinieren können, um Agenten zu erstellen, die eigenständig handeln (rootstrap.com, LangChain). Mit visuellen Entwicklungs-IDE-Features, Vorlagen und Unterstützung für menschliche Eingriffe (“human-in-the-loop”) ermöglicht LangChain schnelle und kontrollierte Agentenentwicklung. Es ist besonders beliebt: 100 000+ Entwickler*innen, über 5 Mio. Downloads pro Monat und 50 000+ erstellte Apps zeugen vom breiten Einsatz und der Community-Unterstützung.



1. Funktionsumfang 7,5/10

- **Anzahl unterstützter LLMs:** Unterstützt eine Vielzahl von Modellen (OpenAI, Anthropic, Google Gemini, Hugging Face, lokale Modelle. 10/10
- **Multi-Agent-Architekturen und Hierarchien:** In Kombination mit LangGraph ermöglicht stateful Multi-Agent-Anwendungen mit Kontrolle über Flüsse, Speicher, menschliche Einbindung. 9/10
- **Workflow-Funktionalität:** Chains, Agents, Tools, RAG, Dokument-Loader, vector stores etc. bieten komplette Workflow-Pipelines. 7/10
- **Anzahl API und Schnittstellen:** Integration mit REST-APIs, SQL-Datenbanken, Cloud usw. 7/10
- **MCP Unterstützung:** Mit dem Paket langchain-mcp-adapters können MCP-Server eingebunden werden. 7/10
- **RAG und Vektordatenbank-Integration:** Umfangreiche Unterstützung für RAG via FAISS, Pinecone, RetrievalQA, Embeddings etc. 7/10
- **Session-übergreifendes Gedächtnis:** Memory Module, ConversationBufferMemory, Einbindung LangGraph-Persistenz für Gedächtnis über Sessions. 7/10
- **Persistente Workflow-Steuerung:** Nur in Kombination mit LangGraph + LangSmith (Monitoring) + LangServe, Workflow-Persistenz, Steuerung und Observability. 6/10

2. Offenheit / Open Source 10/10

Kernbibliothek ist MIT-lizenziert und offen verfügbar. Die ergänzenden Services (LangSmith etc.) sind teilweise geschlossen.

3. Anpassbarkeit 10/10

LangChain ist modular, erweiterbar, unterstützt eigene Python (und JS) Integrationen, Third-Party Tools, eigene Chains

4. Verbreitung & Community 8/10

Über 100k Entwickler, mehr als 5 Mio Downloads pro Monat, über 50k Apps gebaut, lebendige Discord, GitHub mit hunderten Beiträgen.

5. Erlernbarkeit 5/10

Modularität hilft, aber viele Abstraktionen, Komplexität, schwierige Dokumentation – insbesondere bei größeren Setups.

6. Enterprise-Tauglichkeit 7/10

Hohe Modularität, RAG, Sicherheit, Private Deployment, Observability, Skalierbarkeit – eingesetzt u.a. bei Fortune-500, Telcos, Finanzdienstleistern.

7. Sicherheit & Governance 7/10

Private Cloud/Hybrid Unterstützung, Compliance, Rollen, Audit Logging möglich (siehe Implementierungen). Observability: LangSmith aber nicht open source.

LangChain ist eine äußerst leistungsfähige und vielseitige Open-Source-Plattform für die Entwicklung von LLM-basierten Anwendungen. Mit umfassender Multi-LLM-Unterstützung und modularen Bausteinen für Chains, Agents, Tools und RAG bietet sie nahezu alles, was moderne KI-Workflows benötigen. Besonders stark ist sie im Bereich RAG und Vektordatenbankintegration sowie in der Orchestrierung persistenter, kontextreicher Agentensysteme über LangGraph und Observability durch LangSmith.

Zusammengefasst ist LangChain eine exzellente Wahl für technisch versierte Teams, die skalierbare, KI-gestützte Applikationen bauen – vorausgesetzt, man ist bereit, in Lernaufwand und Komplexitätsmanagement zu investieren.



LangGraph

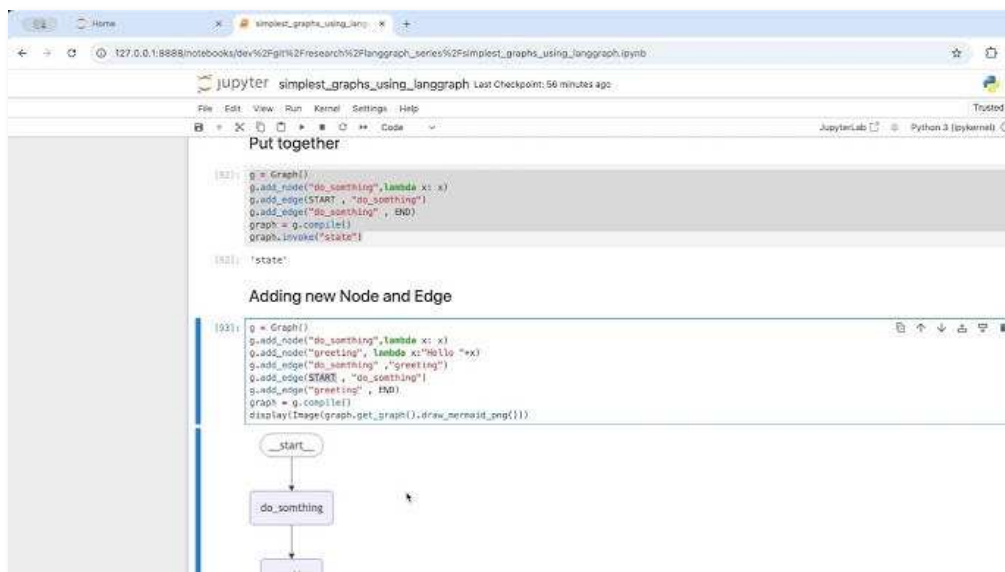
Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

LangGraph ist eine Erweiterung von LangChain, die speziell für die Erstellung von zustandsbasierten, mehrstufigen AI-Anwendungen entwickelt wurde. Es ermöglicht die Definition von AI-Workflows als gerichtete Graphen, wodurch komplexe Entscheidungsbäume und bedingte Verzweigungen möglich werden.

Hauptmerkmale:

- **Graph-basierte Workflows:** Visualisierung und Definition von AI-Prozessen als Graphen
- **Zustandsverwaltung:** Persistente Speicherung des Anwendungszustands zwischen Schritten
- **Bedingte Logik:** Dynamische Pfadwahl basierend auf Zwischenergebnissen
- **Parallelisierung:** Gleichzeitige Ausführung unabhängiger Workflow-Schritte
- **Debugging:** Eingebaute Tools zur Verfolgung der Ausführung

LangGraph eignet sich besonders für komplexe AI-Anwendungen wie Multi-Agent-Systeme, iterative Problemlösung oder Anwendungen mit mehreren Validierungsschritten. Die visuelle Darstellung der Workflows erleichtert das Verständnis und die Wartung komplexer Logik. Entwickler können verschiedene LLMs für unterschiedliche Schritte verwenden und Human-in-the-Loop-Szenarien implementieren. Die Integration in bestehende LangChain-Projekte ist nahtlos möglich.



1. Funktionsumfang 7,9/10

- **Anzahl unterstützter LLMs:** LangGraph basiert auf LangChain und unterstützt zahlreiche Modelle über LangChain-Integrationen (z. B. Anthropic, OpenAI etc.). 9/10
- **Multi-Agent-Architekturen und Hierarchien:** Ermöglicht graph-basierte Multi-Agent-Workflows mit Knoten (Agenten) und Kanten (Logik), unterstützt Schleifen, Bedingungen sowie komplexe topologische Strukturen. 10/10
- **Workflow-Funktionalität:** Bietet graphbasierte Workflows mit Debugging, Checkpoints, Branching, menschlicher Einbindung und flexibler Zustandspassage. 8/10
- **Anzahl API und Schnittstellen:** Als Framework nutzbar mit Python/JS APIs; Plattform-Option bietet HTTP-Endpoints und visuelle IDE (LangGraph Studio). 7/10
- **MCP Unterstützung:** Mit dem Paket langchain-mcp-adapters können MCP-Server eingebunden werden. 7/10
- **RAG und Vektordatenbank-Integration:** Nur über Kombination mit LangChain. 4/10
- **Session-übergreifendes Gedächtnis:** Unterstützt Short- und Long-Term Memory via persistentem Zustand, Checkpointing und Überblick durch LangSmith. 9/10
- **Persistente Workflow-Steuerung:** Bietet dauerhafte Ausführung, automatische Wiederaufnahme, Human-in-the-Loop, Streaming, Logs und Hintergrundlaufzeiten. 9/10

2. Offenheit / Open Source 10/10

LangGraph ist Open Source unter MIT-Lizenz, aktiv entwickelt auf GitHub (GitHub, LangChain). Die Cloud-basierte LangGraph Plattform ist hingegen proprietär (LangChain).

3. Anpassbarkeit 10/10

Sehr flexibel: eigene Knoten, State-Objekte, graph-lastige Logik, Python/JS-Support, eng verbunden mit LangChain-Ökosystem

4. Verbreitung & Community 7/10

Wird eingesetzt von namhaften Unternehmen wie Klarna, Replit, Elastic sowie in Forschung (u. a. automatisiertes Bugfixing, maschinelle Übersetzung) (LangChain AI, arXiv). Entwicklerberichte zeigen starke Akzeptanz in komplexen Anwendungsfällen

5. Erlernbarkeit 3/10

Visualisierung durch graphische Struktur erleichtert Einstieg für Entwickler, aber Nutzer beklagen mangelhafte Dokumentation und steile Lernkurve

6. Enterprise-Tauglichkeit 7/10

LangGraph Plattform bietet starke Produktionsfeatures (Streaming, Scheduling, Skalierung, Monitoring, Human-in-Loop) (LangChain, LangChain Dokumentation). Das Framework selbst ist robust und ausgelegt für langlebige Workflows.

7. Sicherheit & Governance 7/10

Features wie Checkpoints, menschliche Unterbrechung, vollständige Zustandstransparenz, Debugging (z. B. via LangSmith), plus Open-Source-Transparenz

LangGraph ist ein hochmodernes, graph-basiertes Orchestrierungssystem für autonome, langlaufende KI-Agenten. Während LangChain lineare Chains nutzt, erlaubt LangGraph flexible, nicht-lineare Graphen mit Schleifen, Bedingungen und komplexer Kontrolllogik – ideal für anspruchsvolle Anwendungen wie dynamische Workflows, Agenten-Kollaboration und Wiederaufnahme nach Fehlern.

Insgesamt bietet LangGraph eine exzellente Grundlage für den Aufbau skalierbarer, robust gesteuerter KI-Workflows – vorausgesetzt, man ist bereit, in Wissen und technische Integration zu investieren.



Langflow

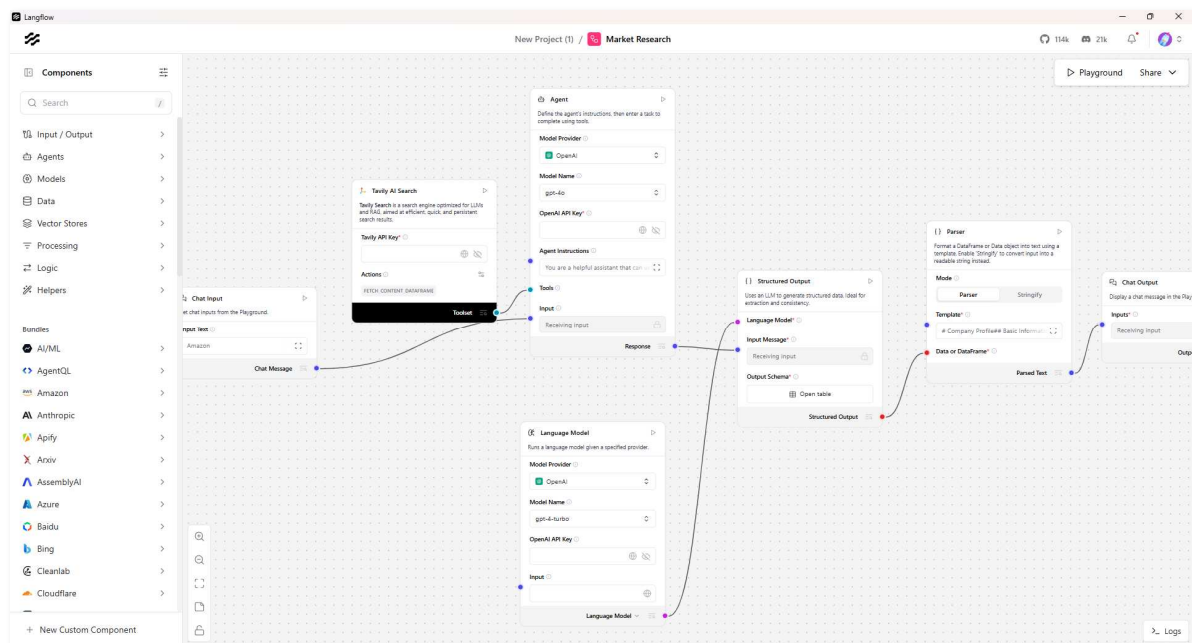
Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

Langflow ist eine Low-Code-Plattform mit grafischer Benutzeroberfläche für die Erstellung von AI-Workflows und -Anwendungen. Es basiert auf LangChain und ermöglicht es auch nicht-technischen Benutzern, komplexe AI-Pipelines durch Drag-and-Drop zu erstellen.

Zentrale Features:

- **Visuelle Pipeline-Erstellung:** Drag-and-Drop-Interface für Workflow-Design
- **Vorgefertigte Komponenten:** Umfangreiche Bibliothek von AI-Bausteinen
- **Real-time Testing:** Sofortige Ausführung und Testen von Workflows
- **Code-Export:** Generierung von ausführbarem Python-Code
- **Template-Bibliothek:** Vorgefertigte Lösungen für häufige Anwendungsfälle

Die Plattform richtet sich sowohl an Entwickler als auch an Business-Analysten. Langflow unterstützt verschiedene Datenquellen, APIs und LLM-Anbieter. Besonders nützlich ist die Möglichkeit, Prototypen schnell zu erstellen und zu iterieren. Die generierten Workflows können als APIs bereitgestellt oder in andere Anwendungen integriert werden. Die Community-Edition ist kostenlos verfügbar, während Enterprise-Features zusätzliche Funktionalitäten für Unternehmen bieten.



1. Funktionsumfang 8,1/10

- **Anzahl unterstützter LLMs:** Langflow ist Azure modellagnostisch, unterstützt zahlreiche LLM-Provider wie OpenAI, Hugging Face, Anthropic oder Vertex AI und verschiedene Vektordatenpeicher. 10/10
- **Multi-Agent-Architekturen und Hierarchien:** Die Version 1.1 bringt umfassende Multi-Agent-Orchestrierung: Agent-Komponenten, Tool-Mode, Agents können andere Agents aufrufen – für komplexe, mehrstufige Workflows. 9/10
- **Workflow-Funktionalität:** Visual Editor erlaubt Drag-and-Drop-Flows mit Komponenten, Playground zum Testen, Export/Deployment über API oder als Server möglich. 7/10
- **Anzahl API und Schnittstellen:** Flows lassen sich via API, Python-/JS-Snippets, MCP-Server einbinden – ideal für Integration in bestehende Anwendungen. 8/10
- **MCP Unterstützung:** Langflow kann sowohl MCP-Server als auch MCP-Client sein, erlaubt Anwendung von Flows als Tools im MCP-Ökosystem. 10/10
- **RAG und Vektordatenbank-Integration:** RAG-Anwendungen werden explizit unterstützt: Integration mit Pinecone, FAISS, Weaviate, ChromaDB; visuelle Datenabläufe für Retrieval. 9/10
- **Session-übergreifendes Gedächtnis:** Langflow bietet konversationsbasiertes Agenten-Memory (z. B. Chat-Verläufe) als Teil der Multi-Agent-Orchestrierung; erinnert sich an Zwischenschritte. 5/10
- **Persistente Workflow-Steuerung:** Workflows können als APIs oder Server bereitgestellt werden, mit Playground, Logs und Wiederaufnahme möglich. Visual Editor mit Debugging-Support. 7/10

2. Offenheit / Open Source 10/10

Langflow ist vollständig Open Source, unter MIT-Lizenz verfügbar, und auf GitHub aktiv – mit Security-Policy, Adressierung von CVEs.

3. Anpassbarkeit 9/10

Extrem flexibel durch Python-basierte Architektur: Custom-Components, eigene Python-Logik, Teilen von Komponenten

4. Verbreitung & Community 4/10

Gute Verbreitung: über 114k GitHub-Stars, aktive Weiterentwicklung. Community-Feedback auf Reddit zeigt wachsende Nutzung und Austausch.

5. Erlernbarkeit 9/10

Visual Interface erleichtert Einstieg – Drag & Drop, Playground, Templates. Desktopversion hat die vorher vorhandene komplexe Installationsprobleme behoben.

6. Enterprise-Tauglichkeit 7/10

Unterstützt Deployment als API oder Server, Visual Developer Experience, Multi-Agent, RAG, MCP sowie Security-Advisories. Skalierungsprobleme früher Versionen deutlich verbessert.

7. Sicherheit & Governance 7/10

Open-Source, MIT-Lizenz, aktive Security-Fixes (z. B. CVE-Updates), Logs und Monitoring über LangSmith.

Langflow ist ein äußerst leistungsstarkes, visuelles Framework zur Entwicklung und Orchestrierung von AI-Flows und Agenten. Dank umfassender Unterstützung für Anzahl unterstützter LLMs, RAG, Multi-Agent-Strukturen und das Model Context Protocol (MCP) eignet es sich hervorragend für flexible, erweiterbare Projekte. Die visuelle Drag-and-Drop-Oberfläche in Kombination mit einem interaktiven Playground, API-Deployment-Möglichkeiten und Python-Erweiterbarkeit macht die Integration in bestehende Systeme schnell und intuitiv. Langflow ist unser „Liebling“ und Testsieger. Der Installationsprozess ist mittlerweile einfacher geworden und die Kinderkrankheiten werden von Version zu Version weniger. Die Übernahme (von DataStax) durch IBM lässt für die Zukunft auf strategische „Stabilität“ hoffen.



AutoGPT war einer der ersten autonomen AI-Agenten, der darauf ausgelegt ist, komplexe Aufgaben selbständig zu planen und auszuführen, indem er die Nutzungsziele in subtasks zerlegt und eigenständig automatisiert ausführt. Es verwendet GPT-4 als Kern-Engine und kann iterativ an Problemen arbeiten, bis es eine Lösung findet.

Besondere Eigenschaften:

- **Autonome Ausführung:** Selbständige Planung und Durchführung von mehrstufigen Aufgaben
- **Selbstreflexion:** Bewertung der eigenen Arbeit und Anpassung der Strategie
- **Tool-Integration:** Verwendung verschiedener Tools wie Web-Browser, Dateisystem, APIs
- **Persistent Memory:** Speicherung von Zwischenergebnissen und gelernten Informationen
- **Goal-oriented Behavior:** Fokussierung auf das Erreichen definierter Ziele

Die Software erfordert sorgfältige Konfiguration der Sicherheitsparameter, da sie weitreichende Aktionen ausführen kann. Die Open-Source-Natur ermöglicht Anpassungen und Erweiterungen für spezifische Anwendungsfälle.

```
edxd@bytexd: ~/Projects/Auto-GPT/Auto-GPT
File Edit View Search Terminal Help
Resolving deltas: 100% (931/931), done.
edxd@bytexd:~/Projects/Auto-GPT$ workon autogpt
(autoGPT) edxd@bytexd:~/Projects/Auto-GPT$ cd Auto-GPT/
(autoGPT) edxd@bytexd:~/Projects/Auto-GPT/Auto-GPT$ ls
ai_settings.yaml  Dockerfile  outputs  requirements.txt  tests
CONTRIBUTING.md  LICENSE    README.md  scripts
(autoGPT) edxd@bytexd:~/Projects/Auto-GPT/Auto-GPT$ pip install -r requirements.txt
Collecting beautifulsoup4
  Downloading beautifulsoup4-4.12.2-py3-none-any.whl (142 kB)
    143.0/143.0 kB 4.6 MB/s eta 0:00:00
Collecting colorama==0.4.6
  Downloading colorama-0.4.6-py2.py3-none-any.whl (25 kB)
Collecting openai==0.27.2
  Downloading openai-0.27.2-py3-none-any.whl (70 kB)
    70.1/70.1 kB 4.4 MB/s eta 0:00:00
Collecting playsound==1.2.2
  Downloading playsound-1.2.2-py2.py3-none-any.whl (6.0 kB)
Collecting python-dotenv==1.0.0
  Downloading python_dotenv-1.0.0-py3-none-any.whl (19 kB)
Collecting pyyaml==6.0
  Downloading PyYAML-6.0-cp310-cp310-manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_2_12_x86_64.manylinux2010_x86_64.whl (682 kB)
    0.0/682.2 kB eta -1:--:--
```

1. Funktionsumfang 4,6/10

- **Anzahl unterstützter LLMs:** Unterstützt GPT-4 (OpenAI), teils auch Llama, Anthropic, Groq. 5/10
- **Multi-Agent-Architekturen und Hierarchien:** Arbeitet modular mit Task-Creation, Prioritization und Execution Agents. Keine echten hierarchischen Agenten. 4/10
- **Workflow-Funktionalität:** Ermöglicht Internetzugriff, Datei- und Codeoperationen, Plattform mit Server/Frontend, Low-Code Workflow Builder 7/10
- **Anzahl API und Schnittstellen:** API-Integrationen, externe Tool-Zugriffe, REST APIs vorhanden, aber wenige „fertige“ Schnittstellen 4/10
- **MCP Unterstützung:** Keine direkte MCP-Unterstützung 1/10
- **RAG und Vektordatenbank-Integration:** Kein RAG-Support. 2/10
- **Session-übergreifendes Gedächtnis:** Unterstützt Short- und Long-Term Memory, persistente Vektor-basierte Speicherung. 7/10
- **Persistente Workflow-Steuerung:** Frontend/Server Architektur, Scheduler, Monitoring und Visual Builder vorhanden. 7/10

2. Offenheit / Open Source 10/10

Kernprojekt unter MIT-Lizenz, offen auf GitHub, hohe Popularität.

3. Anpassbarkeit 7/10

Modular, Plugin-System, offene Erweiterbarkeit, Agent Protocol

4. Verbreitung & Community 5/10

Sehr populär auf GitHub (100k+ Sterne rasch erreicht) (Reddit, Wikipedia); Discord-Community mit 50 000+ Mitgliedern.

5. Erlernbarkeit 5/10

Komplexe Einrichtung, technische Hürde für Nicht-Entwickler.

6. Enterprise-Tauglichkeit 4/10

Innovativer Ansatz, aber Produktionstauglichkeit ist fraglich bzw. es sind Anpassung nötig.

7. Sicherheit & Governance 3/10

Teilnahme am GitHub Secure Open Source Fund, automatisierte Sicherheitspolicies, Dependabot, CodeQL. Allerdings Risiken (z. B. Codeausführung) bekannt.

AutoGPT ist ein visionärer Open-Source-Agenten-Ansatz, der autonome KI-Agenten auf Basis moderner LLMs wie GPT-4 ermöglicht. Der Kern ist modular aufgebaut: Agenten zerlegen komplexe Ziele in Teilaufgaben (Task Creation, Prioritization, Execution), können über das Web recherchieren, Dateien managen, Code generieren und ausführen. Es bietet eine Plattform mit Server-Frontend-Architektur, visuellen Low-Code-Workflows, API-Zugriff und Monitoring-Tools.

Die Quelle ist offen (MIT-Lizenz) und das Projekt hat in kurzer Zeit enorme Popularität erlangt – 100 000+ GitHub-Sterne, große Discord-Community – was die Verbreitung und Dynamik unterstreicht. Anpassung über Plugins und offene Protokolle ist möglich, birgt aber technisch eine Herausforderung: Setup und effektive Nutzung erfordern solides Entwicklerwissen und sind für Nicht-Techniker oft schwer zugänglich.

Für Unternehmen ist AutoGPT eher ein experimentelles Basiswerkzeug denn ein sofort einsetzbares Produkt. Sicherheit wurde erst jüngst durch Teilnahme am GitHub SOSF-Programm gestärkt, doch Risiken bleiben bestehen – insbesondere durch autonome Codeausführung.

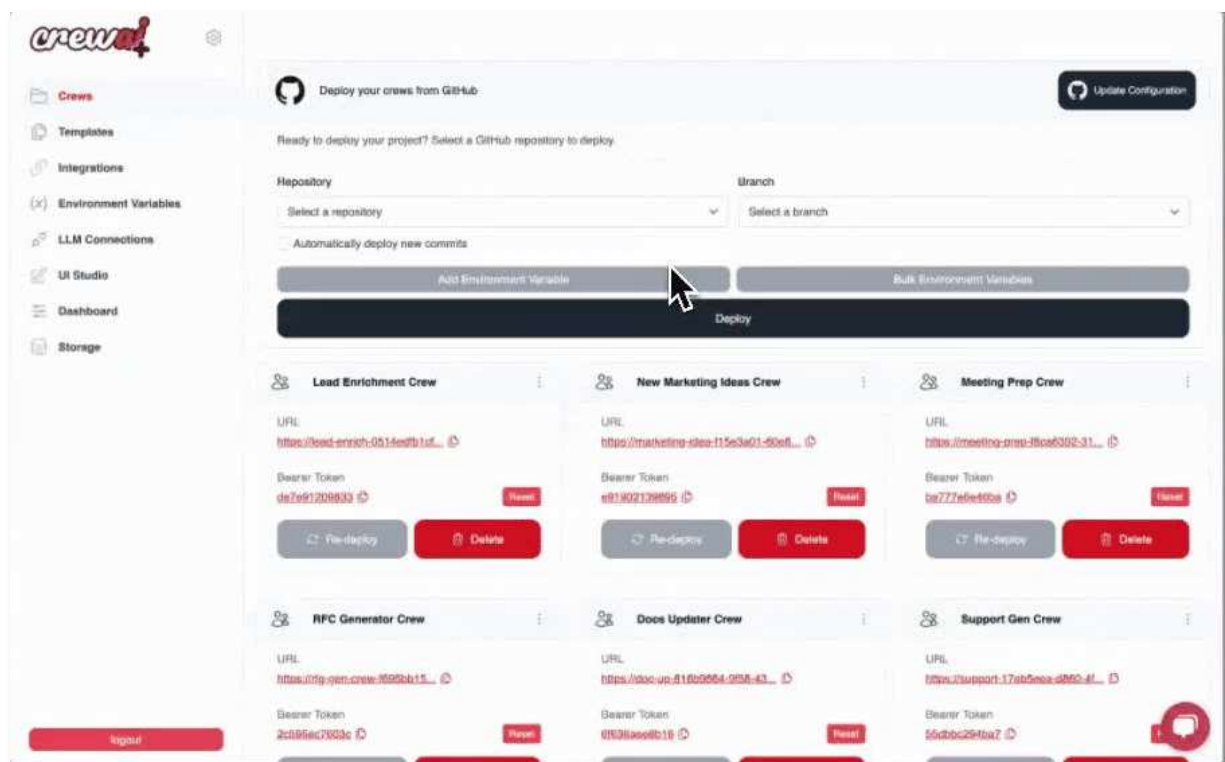


Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

CrewAI ist ein Python-Framework zur Orchestrierung von multi-agenten-basierten KI-Systemen – also „Crews“ – in denen mehrere spezialisierte KI-Agenten gemeinsam komplexe Aufgaben lösen. Hier sind die wichtigsten Punkte:

Kernmerkmale

- **Offenes, leichtgewichtiges Framework:** CrewAI ist in Python geschrieben, von Grund auf neu konzipiert und unabhängig von LangChain oder anderen Agent-frameworks. Es wurde auf Geschwindigkeit und geringe Abhängigkeiten optimiert
- **Multi-Agenten in „Crews“:** Eine Crew besteht aus mehreren KI-Agenten mit definierten Rollen (z. B. Forscher, Analytiker, Autor), die Aufgaben übernehmen und sich untereinander abstimmen.
- **Flows für Ereignis-basierte Abläufe:** Zusätzlich zu Crews gibt es Flows — eventgesteuerte Automationsabläufe mit strikter Steuerung, die sich nahtlos mit Crews integrieren lassen.



1. Funktionsumfang 4,3/10

- **Anzahl unterstützter LLMs:** CrewAI erlaubt die Nutzung eigener Modelle von OpenAI, Google, Azure und HuggingFace sowie vorkonfigurierter Modelle und Templates. 4/10
- **Multi-Agent-Architekturen und Hierarchien:** CrewAI nutzt ein Crew-Konzept, bei dem spezialisierte Agenten (z. B. Forscher, Autor etc.) als Team (Crew) zusammenarbeiten, ergänzt durch Flows für strukturierte Workflows. 8/10
- **Workflow-Funktionalität:** Mit Flows lassen sich ereignisgesteuerte Workflows (inkl. Zustandsmanagement, Branching) definieren, die Crews ergänzen. 7/10
- **Anzahl API und Schnittstellen:** CrewAI bietet API-Integration, Tool-Anbindung und die Möglichkeit, Crews via CrewAI+ als API einzubinden. 4/10
- **MCP Unterstützung:** Keine explizite Unterstützung von MCP. 1/10
- **RAG und Vektordatenbank-Integration:** Keine direkte RAG Unterstützung 1/10
- **Session-übergreifendes Gedächtnis:** Keine eindeutigen Hinweise auf Memory-Funktionalität oder Persistenz über Sessions hinweg. 2/10
- **Persistente Workflow-Steuerung:** Flows unterstützen Zustandsmanagement, deterministische Ausführung, Resumability und Kontrolle. 7/10

2. Offenheit / Open Source 10/10

CrewAI ist als Open-Source-Framework verfügbar (z. B. auf GitHub) und erlaubt tiefgreifende Individualisierung.

3. Anpassbarkeit 9/10

Sehr flexibel: Crews, Flows, Tools, APIs, Rollen, und selbst komplette Flows lassen sich anpassen – von Low-Code bis hin zu Python-Code

4. Verbreitung & Community 4/10

CrewAI hat über 30k GitHub-Stars, wird von rund 40–60 % der Fortune-500-Firmen eingesetzt und ist in über 60 Ländern aktiv.

5. Erlernbarkeit 6/10

Viele Nutzer berichten von minimalem Einstiegscode und einfacher Integration („sehr einfach einzubinden“. Andere finden es komplex – insbesondere, wenn man über die „Happy Path“-Szenarios hinausgeht

6. Enterprise-Tauglichkeit 8/10

CrewAI ist für Unternehmen ausgerichtet – es unterstützt Self-Hosting, Cloud, Monitoring, Security sowie eine intuitive UI zur Steuerung und ROI-Übersicht.

7. Sicherheit & Governance 8/10

Jeder Crew läuft isoliert (VPCs), plus visuelle Kontrolle und Monitoring; gezielte Security-Maßnahmen sind Teil des Angebots.

CrewAI ist eine moderne, flexibel einsetzbare Open-Source-Plattform zur Orchestrierung autonomer KI-Agenten. Sie kombiniert die Agenten-basierten Teams ("Crews") mit strukturierten Workflows ("Flows") und bietet so eine starke Mischung aus Autonomie und Kontrolle. Die Plattform ist modular, integrationsbereit mit diversen LLM-Anbietern und unterstützt flexible Deployments (Cloud, Self-Hosted, lokal).

Für einfache Anwendungsfälle lässt sich CrewAI mit wenigen Zeilen Code nutzen. Für fortgeschrittene oder maßgeschneiderte Einsätze benötigt man allerdings Erfahrung und etwas Zeit, die Lernkurve ist also moderat. Die Plattform bringt robuste Enterprise-Features mit, einschließlich Monitoring, Sicherheit und ROI-Einblicken – ideal für produktive Nutzung.



Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

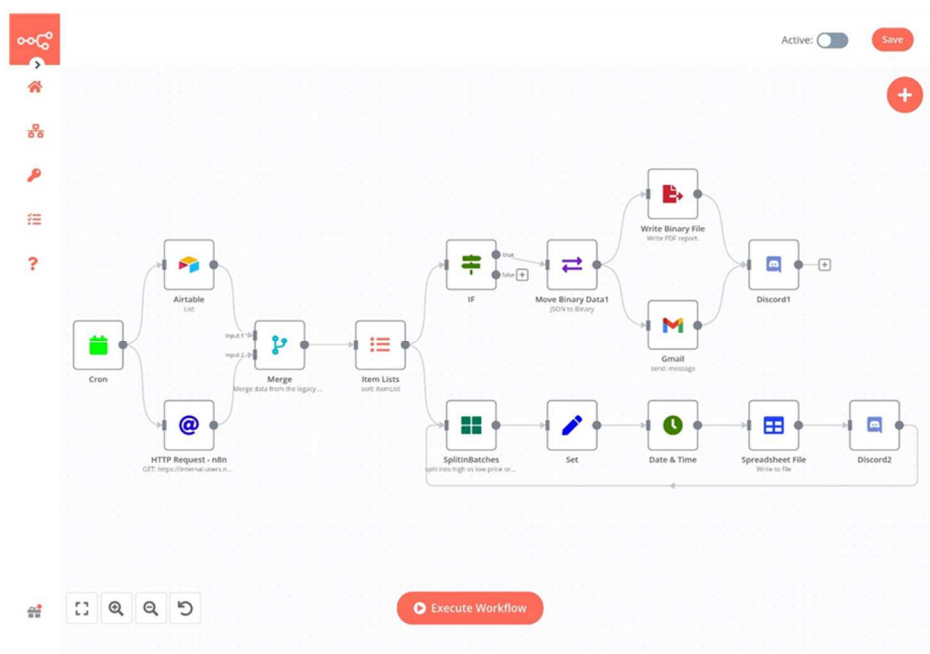
n8n („nodemation“) ist eine Open-Source-Automatisierungsplattform, die den Aufbau komplexer Workflows über eine visuelle Oberfläche ermöglicht.

Seit der Integration von KI- und LangChain-Komponenten hat sich n8n zunehmend zu einer Plattform entwickelt, die nicht nur klassische Automatisierungen, sondern auch agentische KI-Anwendungen abbilden kann. Über dedizierte AI-Nodes lassen sich Large Language Models (LLMs) wie OpenAI, Anthropic oder Hugging Face anbinden und mit Speicherkomponenten („Memory Nodes“) kombinieren.

Hauptfunktionen:

- **Visual Workflow Builder:** Grafische Erstellung von Automatisierungs-Workflows
- **400+ Integrationen:** Vorgefertigte Konnektoren für populäre Services und APIs
- **AI-Service-Integration:** Native Unterstützung für OpenAI, Google AI, und andere AI-Anbieter
- **Conditional Logic:** Komplexe Verzweigungslogik und Entscheidungsbäume
- **Self-hosted Option:** Vollständige Kontrolle über Daten und Ausführungsumgebung

N8N eignet sich besonders für die Integration von AI-Funktionalitäten in bestehende Geschäftsprozesse. Es kann als Orchestrierungsschicht für AI-Agenten fungieren, die mit CRM-Systemen, E-Mail-Plattformen, Datenbanken und anderen Unternehmensanwendungen interagieren.



1. Funktionsumfang 7,5/10

- **Anzahl unterstützter LLMs:** Über dedizierte AI-Nodes lassen sich Large Language Models (LLMs) wie OpenAI, Anthropic oder Hugging Face anbinden 8/10
- **Multi-Agent-Architekturen und Hierarchien:** n8n unterstützt verschiedene Architektur-Muster für Multi-Agent-Systeme: Einfache Agent-Ketten für sequentielle Task-Verarbeitung, Single-Agent-Systeme mit spezialisiertem Tool-Zugriff, Gatekeeper-Patterns mit übergeordneter Koordination von Spezialist-Agenten, sowie verteilte Multi-Agent-Teams mit Mesh- oder hierarchischen Kommunikationsmustern. 7/10
- **Workflow-Funktionalität:** Starker visueller Node-basierter Editor: Drag & Drop, IF-Bedingungen, Loops, JavaScript-/Python-Knoten, Workflow-Passing, Logging 8/10
- **Anzahl API und Schnittstellen:** Über 400 vorbereitete Integrationen, plus vollständige API-Flexibilität via HTTP Request und eigene Custom-Nodes. 9/10
- **MCP Unterstützung:** Unterstützung von MCP-Server. 8/10
- **RAG und Vektordatenbank-Integration:** Agenten können mit Window Buffer Memory kurzfristige Interaktionshistorien speichern, und mit vektordatenbank-basierten RAG-Systemen (z.B. Qdrant, Supabase) langfristige kontextuelle Informationen abrufen. 7/10
- **Session-übergreifendes Gedächtnis:** Agenten können mit Window Buffer Memory kurzfristige Interaktionshistorien speichern 7/10
- **Persistente Workflow-Steuerung:** Selbsthosting oder Cloud-Betrieb, Logging, Fehlerabwicklung, Wiederholungsoptionen. Cloud-Enterprise unterstützt Skalierung (200+ Exekutionen/s) mit Monitoring und SLA-Support. 6/10

2. Offenheit / Open Source 6/10

n8n ist quelloffen und frei für Selbsthoster, aber nicht uneingeschränkt Open Source im klassischen Sinn. Self-hosting ist möglich, doch kommerzielle Nutzung bzw. SaaS-Angebote erfordern Lizenzierung.

3. Anpassbarkeit 9/10

Node-basierte Architektur, eigene JavaScript-/Python-Knoten, Custom Nodes, Templates, API, Code-Trigger möglich.

4. Verbreitung & Community 7/10

Große Nutzerbasis, aktives Forum, viele GitHub-Stars (Community aktiv). Bewertungen positiv; viele Firmen nutzen n8n als Self-Host-Plattform

5. Erlernbarkeit 7/10

Einsteigerfreundliche Visualisierung, dennoch technische Anforderungen bei komplexen Workflows.

6. Enterprise-Tauglichkeit 9/10

Selbsthosting mit vollständiger Kontrolle, Cloud-Angebot mit SLA, Skalierungsmöglichkeiten, Support-consulting.

7. Sicherheit & Governance 9/10

Self-hosting ermöglicht maximale Datenkontrolle. SSL, OAuth2 werden unterstützt; Cloud-Version hat enterprise-grade Policies

n8n zeichnet sich als ideale Plattform für Agentic AI aus, weil es die Komplexität echter Systemintegration mit der Leistung moderner LLMs verbindet, dabei aber niedrige Einstiegsbarrieren schafft. Die Plattform integriert LangChain direkt als Kernelement. Viele der Standard-AI-Nodes sind tatsächlich auf LangChain aufgebaut, was Agenten mit erweiterten Fähigkeiten wie Tool-Calling, Memory Management und Reasoning ermöglicht. Für maximale Flexibilität bietet n8n auch einen LangChain Code Node, der vollständige Kontrolle über Agent-Logik, Speicherung und Tool-Integration gewährt.

Das Tool ermöglicht es Teams schnell produktive Agentic-Systeme zu entwickeln, ohne dabei auf Enterprise-Grade-Features wie Monitoring, Sicherheit oder Skalierbarkeit zu verzichten.

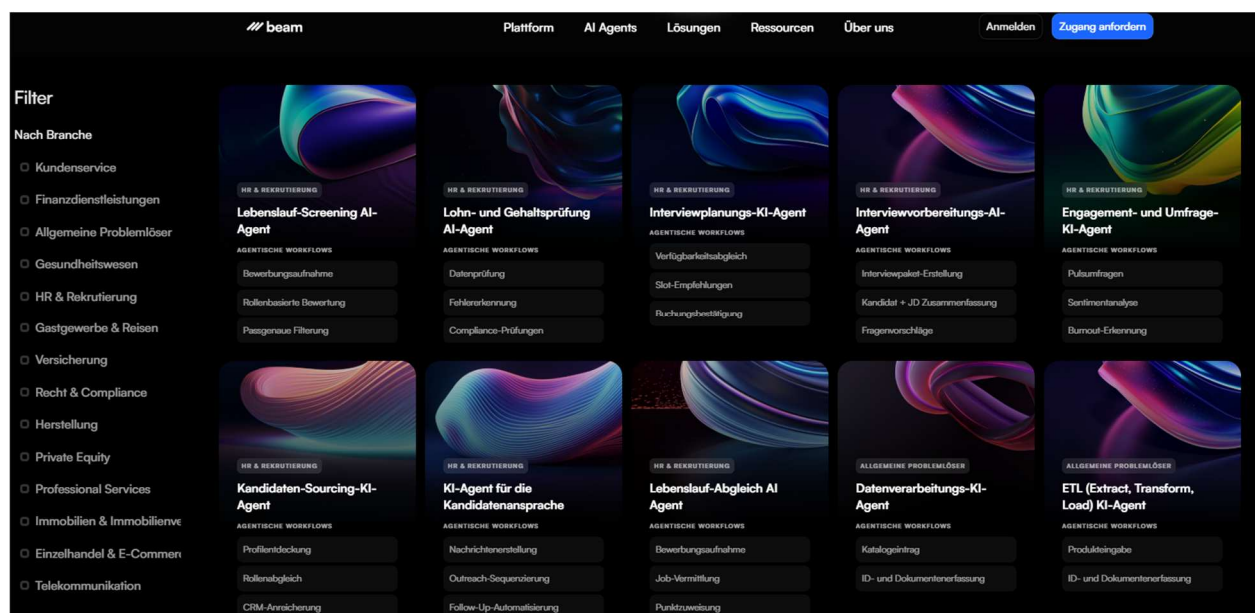


Beam ist eine spezialisierte Plattform für die Bereitstellung und Orchestrierung von AI-Workloads in der Cloud. Es fokussiert sich auf die skalierbare Ausführung von AI-Modellen und -Workflows mit besonderem Schwerpunkt auf Performance und Kosteneffizienz.

Kernmerkmale:

- **Serverless AI Compute:** Automatische Skalierung von AI-Workloads ohne Server-Management
- **Multi-Cloud Support:** Ausführung auf verschiedenen Cloud-Providern
- **Container-basierte Deployments:** Einfache Paketierung und Bereitstellung von AI-Anwendungen
- **GPU-Optimierung:** Effiziente Nutzung von GPU-Ressourcen für ML-Workloads
- **API-first Approach:** RESTful APIs für Integration in bestehende Systeme

BeamAI richtet sich an Entwickler und Unternehmen, die AI-Modelle in Produktionsumgebungen betreiben möchten. Die Plattform abstrahiert die Komplexität der Infrastruktur-Verwaltung und ermöglicht es, sich auf die Entwicklung der AI-Logik zu konzentrieren. Besonders wertvoll ist die automatische Kostenoptimierung durch intelligente Ressourcenzuteilung. Die Integration verschiedener ML-Frameworks und die Unterstützung für Custom-Container machen es flexibel für verschiedene AI-Anwendungsfälle.



1. Funktionsumfang 4,8/10

- **Anzahl unterstützter LLMs:** Beam AI stellt keinen Fokus auf Sprachmodelle, sondern darauf, spezialisierte Agenten zur Automatisierung von Geschäftsprozessen einzusetzen – keine direkte Anzahl unterstützter LLMs. 2/10
- **Multi-Agent-Architekturen und Hierarchien:** Die Plattform basiert auf einem Agenten-Paradigma, bei dem mehrere KI-Agenten in komplexen Workflows zusammenarbeiten. 8/10
- **Workflow-Funktionalität:** Unterstützung für agentenbasierte Workflows, interaktive Dashboards, Automatisierungsaufgaben über CRM, ERP, Projektmanagementsysteme usw. Kein Workflowcreator. 3/10
- **Anzahl API und Schnittstellen:** Vielfältige Integrationen (Airtable, ServiceNow, Asana u.v.m.) sowie eine offene API zur Verbindung mit anderen Tools aber weniger fertige Schnittstellen. 4/10
- **MCP Unterstützung:** Integrierte MCP-Server Unterstützung aber keine freie Definition. 4/10
- **RAG und Vektordatenbank-Integration:** Beam AI adressiert primär Automatisierung von Geschäftsprozessen, nicht Retrieval-Augmented Generation. 1/10
- **Session-übergreifendes Gedächtnis:** Beam AI unterstützt Gedächtnis über Session- und Ausführungsgrenzen hinweg. Agenten behalten Informationen mithilfe von Memory-Datenbanken, Tools und dokumentierten Flows im Hintergrund. 8/10
- **Persistente Workflow-Steuerung:** Die Plattform bietet Dashboards zur historischen Übersicht, Performance-Monitoring und Planung zukünftiger Aufgaben – unterstützt dadurch persistente Steuerung. 8/10

2. Offenheit / Open Source 1/10

Beam AI ist eine proprietäre Plattform

3. Anpassbarkeit 2/10

Beam AI bietet vorgefertigte Agenten-Templates und Anpassung über API-Integration, aber keine Angabe darüber, ob komplette individuelle Komponenten oder Programmierung möglich sind.

4. Verbreitung & Community 2/10

Beam AI wird in Unternehmen (inkl. Fortune-500) eingesetzt und deckt Branchen wie Healthcare, Insurance, Manufacturing ab (vivociti.com). Die Community im öffentlichen Raum (z. B. Reddit) ist eher spärlich vertreten.

5. Erlernbarkeit 7/10

Die Plattform ist benutzerfreundlich aufgebaut (Dashboards, Templates), jedoch fehlt öffentliche Dokumentation, Nutzerfeedback oder Tutorials.

6. Enterprise-Tauglichkeit 9/10

Beam AI richtet sich klar an Unternehmen – mit robusten Integrationen, Agentenabläufen für HR, Compliance, Back-Office, 24/7-Betrieb und Skalierung.

7. Sicherheit & Governance 9/10

Beam AI ist SOC 2 Typ 2- und ISO 27001-zertifiziert – damit bietet die Plattform ein hohes Maß an Informationssicherheit

Beam AI ist eine leistungsstarke Plattform zur *agentic process automation* für mittelständische und große Unternehmen. Die Kernidee: selbstlernende KI-Agenten übernehmen komplexe, repetitive Aufgaben. Dank Multi-Agent-Architektur und umfassender Integrationsfähigkeit verbindet Beam AI bestehende Systeme wie CRM, ERP oder Projektmanagement-Plattformen effizient und intelligent. Weniger stark ist Beam AI bei LLM-Funktionen oder im RAG-Bereich – dort bleibt die Plattform klassisch datengetrieben, weniger modellorientiert.

Beam AI eignet sich hervorragend für Unternehmen, die wiederkehrende Operationen automatisieren und optimieren möchten. Für KI-spezifische Gedächtnis- oder Retrieval-Anwendungen ist eine Ergänzung mit spezialisierten Tools zu empfehlen.

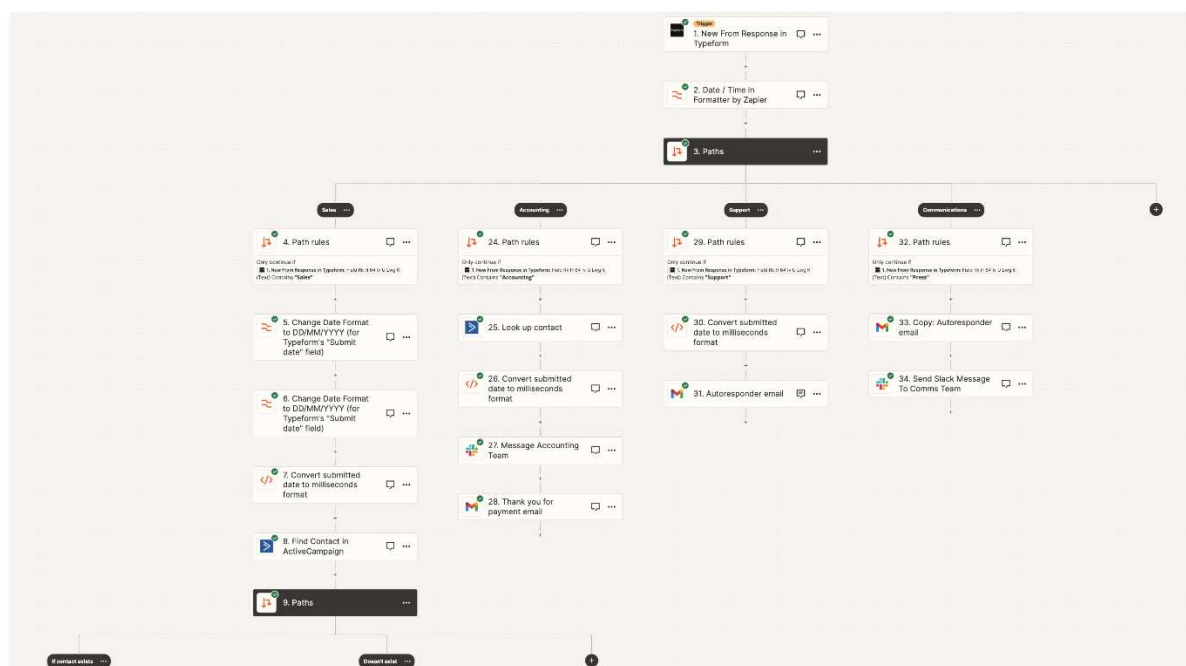


Zapier ist eine etablierte Workflow-Automatisierungsplattform, die in den letzten Jahren umfangreiche AI-Funktionalitäten integriert hat. Es ermöglicht die Verbindung von über 6000 Apps und Services mit AI-Modellen für intelligente Automatisierung.

Zentrale Funktionen:

- **Zap-basierte Workflows:** Trigger-Action-Automatisierungen zwischen verschiedenen Apps
- **AI-Integration:** Native Unterstützung für ChatGPT, Claude, und andere AI-Services
- **No-Code Approach:** Erstellung komplexer Automatisierungen ohne Programmierung
- **Extensive App Ecosystem:** Verbindung mit praktisch allen populären Business-Tools
- **Conditional Logic:** Intelligente Verzweigungen basierend auf Dateninhalt

Zapier eignet sich besonders für die Integration von AI-Funktionalitäten in bestehende Geschäftsprozesse ohne technische Komplexität. Typische Anwendungsfälle umfassen automatische Content-Generierung, intelligente E-Mail-Verarbeitung, Sentiment-Analyse von Kundenfeedback oder automatische Datenextraktion. Die Plattform bietet sowohl kostenlose als auch kostenpflichtige Pläne, wobei Enterprise-Features erweiterte Sicherheit und Compliance-Funktionen umfassen. Die große Benutzergemeinschaft und umfangreiche Template-Bibliothek erleichtern den Einstieg erheblich.



1. Funktionsumfang 6/10

- **Anzahl unterstützter LLMs:** Chat GPT-4o-mini ist das Standardmodell für Zapier Chatbots, Api Verbindung zu OpenAI und Claude. 3/10
- **Multi-Agent-Architekturen und Hierarchien:** Workflows bestehen aus „Zaps“ mit Triggern und Aktionen; komplexere Abläufe mit Bedingungen, Pfaden und Logikstrukturen sind möglich. Agenten-orientierte Architekturen sind in einem Beta-Zustand vorhanden. 3/10
- **Workflow-Funktionalität:** Starker visueller Workflow-Builder mit branched logic, Filter, Pfaden, Webhooks, API-Anbindung. Der sweet spot von Zapier. 10/10
- **Anzahl API und Schnittstellen:** Zapier bietet über 8.000 Integrationen, plus Webhooks, Developer Platform, APIs und Services wie Zapier Tables, Interfaces, Canvas. 10/10
- **MCP Unterstützung:** Zapier unterstützt das MCP-Protokoll – sowohl als Host (MCP-Server) als auch Client. Einrichtung aufwendig 8/10
- **RAG und Vektordatenbank-Integration:** Kein Fokus auf Retrieval-Augmented Generation oder explizite Vektordatenbank-Unterstützung. 1/10
- **Session-übergreifendes Gedächtnis:** Kein natives persistentes Gedächtnis über Sessions, Workflows sind stateless, nur über externe Memories. 5/10
- **Persistente Workflow-Steuerung:** Workflows können versioniert, geprüft, getriggert, analysiert und genehmigt werden. Enterprise-Features wie Alerts, Analysen, Genehmigungen oder Zap Runs API sind verfügbar. 8/10

2. Offenheit / Open Source 1/10

Zapier ist proprietär – kein Open Source oder Source-Available.

3. Anpassbarkeit 10/10

Leistungsstarke Plattform mit Developer Platform für eigene Integrationen, Custom Code, Webhooks, Trigger – hochgradig anpassbar.

4. Verbreitung & Community 9/10

Weltweit führende Automatisierungsplattform, über 2,2 Millionen Unternehmen nutzen Zapier; 87 % der Forbes Cloud 100 setzen auf Zapier.

5. Erlernbarkeit 8/10

Visuelle Oberfläche ist einfach zugänglich; Templates, Copilot und umfangreiche Dokumentation erleichtern den Einstieg. Komplexere Zaps erfordern jedoch etwas Einarbeitungszeit.

6. Enterprise-Tauglichkeit 9/10

Enterprise-Pläne bieten SSO (SAML), SCIM Provisionierung, 2FA, Domain Insights, App-Berechtigungen, Audit Logs, Alerts, Performance Analytics, Dedizierter Support.

7. Sicherheit & Governance 9/10

Umfassende Sicherheitsmaßnahmen: SOC 2 & SOC 3 Compliance, GDPR/CCPA, Verschlüsselung (AES-256, TLS), Bug-Bounty, regelmäßige PenTests, Audit-Logs, Zugriffsverwaltung. Enterprise-Data-Retention individuell konfigurierbar.

Zapier ist eine führende No-Code-Plattform für Workflow-Automatisierung und Integration, mit über 8.000 (!) vorgefertigten App-Konnektoren. Nutzer können komplexe Automatisierungen über Softwaregrenzen hinweg visualisieren und erstellen. Die Plattform unterstützt moderne KI-Automatisierungen einschließlich LLM-gesteuerter Tools und MCP-Orchestrierung. Dank intuitiver Oberfläche und unterstützenden Features wie Copilot ist der Einstieg leicht, auch wenn komplexe Workflows mehr Wissen erfordern.

Für die Erstellung hierarchische Agentenstrukturen oder spezialisierter KI-Funktionen wie RAG oder Gedächtnis-Management empfiehlt sich aber die Nutzung anderer Lösungen.



Make bietet eine leistungsstarke No-Code-Automatisierungsplattform, mit der schnell Workflows und Automatisierungen rund um KI-gestützte Prozesse erstellt werden können.

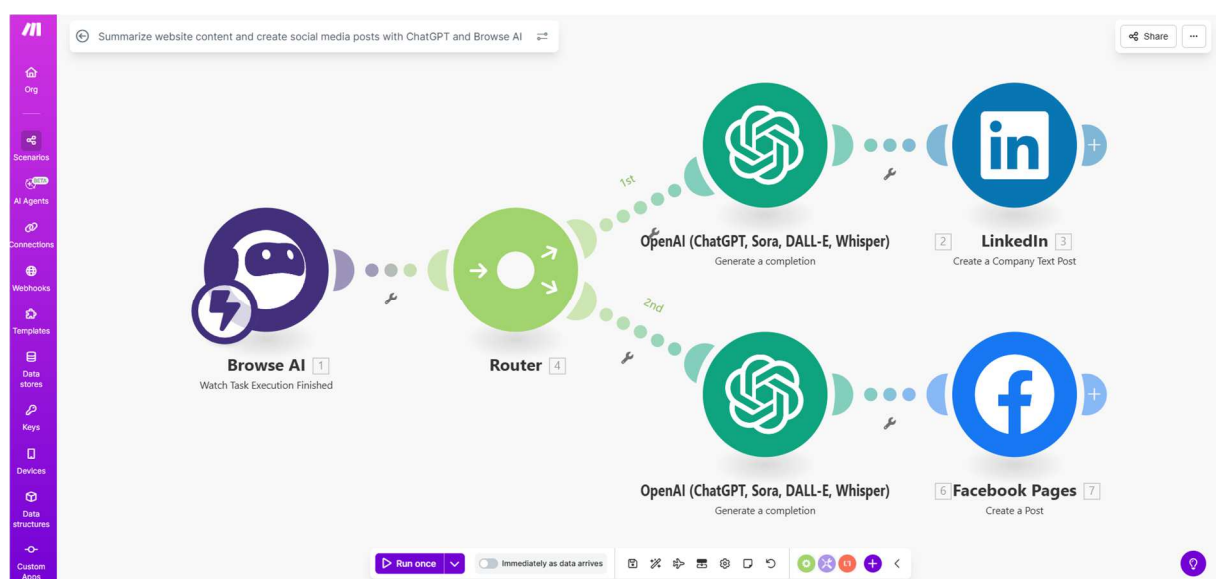
KI-Module und Integration

Make integriert über 2.000 Apps und bietet native Unterstützung für zahlreiche KI-Dienste wie OpenAI, Gemini, Vapi (Voice Agents), HeyGen (Video KI) und viele mehr. Dies umfasst gängige Funktionen wie:

- Textgenerierung und Bilderzeugung
- Sprach-zu-Text-Umwandlung
- Automatische Datenextraktion und Scraping
- Voice Agents und automatische Videoproduktion

Eigene Features wie der KI-Assistent und „Fill by AI“ automatisieren die Erstellung und Dokumentation von Workflows, erklären Szenarien, helfen beim Troubleshooting und machen die Bedienung noch zugänglicher.

Make ist kein "AI-native" Framework im Sinne von vollständigem Agentic Reasoning oder Multi-Agent-Orchestrierung (wie spezialisierte AI-Agenten-Frameworks). Komplexe Logik oder Zustandsmanagement über längere Zeiträume erfordern oft Zusatzmodule oder externe Tools; der Fokus liegt auf "Task-Automation" im Workflow-Stil, nicht auf tiefgehender Agenten-Architektur.



1. Funktionsumfang 5,3/10

- **Anzahl unterstützter LLMs:** Make AI unterstützt zahlreiche KI-Modelle wie Claude, Gemini, GPT-4, Mistral AI u.a. 6/10
- **Multi-Agent-Architekturen und Hierarchien:** Make bietet visuelle Automatisierung mit Agent-ähnlichen Workflows, aber keine echte Multi-Agenten Orchestrierung. 2/10
- **Workflow-Funktionalität:** Make ist primär eine Automation/Workflow-Plattform und bietet Drag-&-Drop Szenarien, Verknüpfung vieler Apps, Trigger-Action-Logik. 10/10
- **Anzahl API und Schnittstellen:** Make verfügt über hunderte (400+ KI-Apps) vordefinierte Integrationen und unterstützt freie API-Aufrufe/Custom Modules. 10/10
- **MCP Unterstützung:** Keine explizite native Unterstützung für Anthropic's MCP-Standard. 2/10
- **RAG und Vektordatenbank-Integration:** RAG-Funktionen (Kontextanreicherung) sind auf der Roadmap. Derzeit ist eine Anbindung an Vektordatenbanken nur über manuelle API-Module möglich. 1/10
- **Session-übergreifendes Gedächtnis:** Kein natives persistentes Gedächtnis über Sessions, Workflows sind stateless, nur über externe Memories. 4/10
- **Persistente Workflow-Steuerung:** Workflows können versioniert, geprüft, getriggert, analysiert und genehmigt werden. Enterprise-Features wie Alerts, Analysen, Genehmigungen oder Zap Runs API sind verfügbar. 7/10

2. Offenheit / Open Source 2/10

Make.com ist eine geschlossene, proprietäre SaaS-Plattform. Der Quellcode ist nicht offen zugänglich, was die Datensouveränität einschränkt.

3. Anpassbarkeit 9/10

Sehr hoch. Durch die HTTP-Module und die Möglichkeit, Python/Node.js-Code über externe Services oder Code-Module zu integrieren, ist die Software extrem erweiterbar.

4. Verbreitung & Community 5/10

Sehr große und aktive Nutzerbasis und Community (als Nachfolger von Integromat), unterstützt durch umfangreiche Dokumentation, Foren und Partnernetzwerke.

5. Erlernbarkeit 9/10

Die visuelle Drag-and-Drop-Oberfläche ist intuitiv. Der AI Assistant zur Szenario-Erstellung senkt die Lernkurve im Vergleich zu Code-Frameworks deutlich.

6. Enterprise-Tauglichkeit 8/10

Bietet hohe Skalierbarkeit, SLA-Garantien, umfassende Audit-Logs, Team-Kollaborationsfunktionen und SSO-Unterstützung in den höheren Tarifen.

7. Sicherheit & Governance 8/10

Solide durch **E2E-Verschlüsselung**, **IP-Whitelisting** und robuste Fehlerbehandlungsmechanismen; entspricht gängigen Compliance-Anforderungen (Audit-Logs).

Make AI – als Teil der Make Plattform – ist eine sehr leistungsfähige Lösung für die Integration von KI-Modellen in Automatisierungs- und Workflow-Szenarien. Besonders stark zeigt sich die Plattform bei klassischen Workflow-Integrationen, der breiten API- und Schnittstellenlandschaft sowie der Unterstützung vieler LLMs. Auch die Anpassbarkeit und Community-Verbreitung sind solide.

Im Bereich echter Agenten-Architekturen mit Multi-Agent-Strukturen, persistentem Gedächtnis und tiefgreifender RAG/Vektordatenbank-Integration ist die Funktionalität weit geringer, als bei den dedizierten Agentic-AI Plattformen.



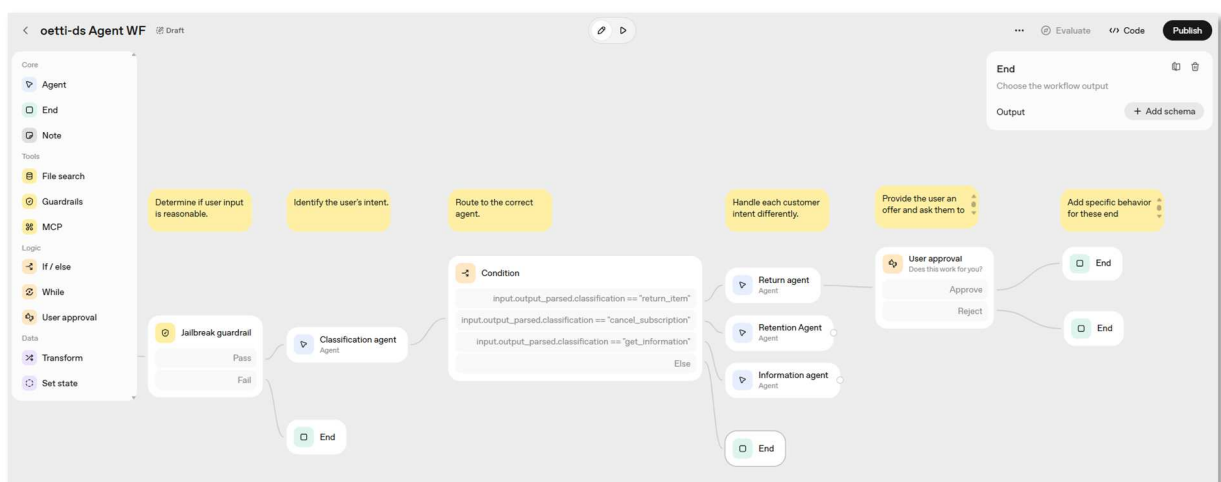
ChatGPT hat mit dem sogenannten „**Agent Builder**“ eine Reihe von Erweiterungen eingeführt, die das System von einem reinen Chatbot hin zu einem aktionsfähigen Agenten weiterentwickeln. Die Plattform ermöglicht Drag-and-Drop-Design von mehrstufigen Agentenworkflows mit versionierter Entwicklung, Sandbox-Previews und Live-Kollaboration.

Das System bietet vier Node-Kategorien:

- **Core:** Agent-Nodes (mit Modelauswahl, Reasoning-Levels, System-Prompts), End-Nodes, Note-Blocks
- **Tools:** File Search (RAG), Guardrail-Nodes (PII-Detektion, Moderation, Halluzination-Checks), MCP-Server-Integration
- **Logic:** If/Else-Verzweigungen, While-Loops, User-Approval-Steps
- **Data:** Transform-Nodes, State-Variable-Management

Die Plattform unterstützt MCP (Model Context Protocol) mit vorinstallierten Servern für Gmail, Drive, Outlook sowie die Möglichkeit, Custom MCP-Server zu hosten. Vector Stores können für RAG-Anwendungen eingebunden werden, Chat History ist optional konfigurierbar, und die Modelauswahl umfasst GPT-5, GPT-4o und kostengünstigere Varianten.

Hierarchische Agenten können erstellt werden, aber es ist kein Delegation-Pattern mit Context-Preservation möglich. Der Agent Builder unterstützt nur permanente Handoffs, jedoch keine "Task-and-Return"-Delegation.



1. Funktionsumfang 4/10

- **Anzahl unterstützter LLMs:** Integration der ChatGPT Modelle. 3/10
- **Multi Agent Architekturen und Hierarchien:** Hierarchische Agenten können erstellt werden, aber es ist kein Delegation-Pattern mit Context-Preservation möglich. 5/10
- **Workflow-Funktionalität:** ChatGPT Agent Builder kann Workflows erstellen, aber große Workflows werden unübersichtlich. 6/10
- **Umfassende API und Schnittstellenkapazitäten:** Der Agent bietet Schnittstellen zu verschiedenen Anwendungen und Plattformen. 4/10
- **MCP Unterstützung:** Die Plattform unterstützt MCP mit vorinstallierten Servern für Gmail, Drive, Outlook sowie die Möglichkeit, Custom MCP-Server zu hosten. 5/10
- **RAG und Vektordatenbank Integration:** File Search unterstützt RAG-Anwendungen. 6/10
- **Session übergreifendes Gedächtnis:** Session-übergreifendes Gedächtnis mit Einschränkungen möglich. 6/10
- **Persistente Workflow Steuerung:** Der Agent kann Aufgaben über längere Zeiträume hinweg ausführen, wobei der Nutzer die Kontrolle behält. 8/10

2. Offenheit / Open Source 4/10

OpenAI verfolgt einen offenen Ansatz, indem es Entwicklern Tools zur Anpassung und Erweiterung von ChatGPT Agent bietet. Dennoch sind die Kernlösungen proprietär.

3. Anpassbarkeit 6/10

ChatGPT Agent ermöglicht es Nutzern, Aufgaben nach ihren spezifischen Bedürfnissen zu gestalten, indem sie benutzerdefinierte Befehle und Workflows erstellen. Aber keine Code Unterstützung.

4. Verbreitung & Community 8/10

OpenAI hat eine breite Nutzerbasis und eine aktive Entwicklergemeinschaft, die ChatGPT Agent nutzt und weiterentwickelt.

5. Erlernbarkeit 7/10

OpenAI bietet umfangreiche Schulungsressourcen, Dokumentationen und Support, um Nutzern den Einstieg in ChatGPT Agent zu erleichtern. Prompting in natürlicher Sprache.

6. Enterprise-Tauglichkeit 10/10

ChatGPT Agent ist für den Unternehmenseinsatz konzipiert und bietet Skalierbarkeit, Sicherheit und Compliance, die für große Organisationen erforderlich sind.

7. Sicherheit & Governance 10/10

OpenAI legt großen Wert auf Sicherheits- und Governance-Aspekte, einschließlich Datenschutz, Compliance und ethische KI-Praktiken, um den verantwortungsvollen Einsatz von ChatGPT Agent zu gewährleisten.

Für agentische KI-Anwendungen bietet der ChatGPT Agent Builder eine sehr nützliche Plattform, die viele wesentliche Funktionen bereitstellt: Handlung, Tool-Zugriff, Automatisierung, Multi-Step-Flows.

Wenn es aber um hierarchische Multi-Agenten-Systeme geht – also mehrere Agenten mit Delegation, Koordination, Überwachung und persistierendem Gedächtnis – dann ist die Plattform momentan eher als Einstieg zu sehen und weniger als vollwertiger Agenten-Orchestrator.

In Summe: Der OpenAI Agent Builder eignet sich hervorragend für Rapid Prototyping und MVP-Entwicklung, Anwendungsfälle mit moderater Komplexität (Single-Agent- oder einfache Multi-Agent-Szenarien mit 2-5 spezialisierten Agents) und Organisationen, die im OpenAI-Ökosystem committed sind.



Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

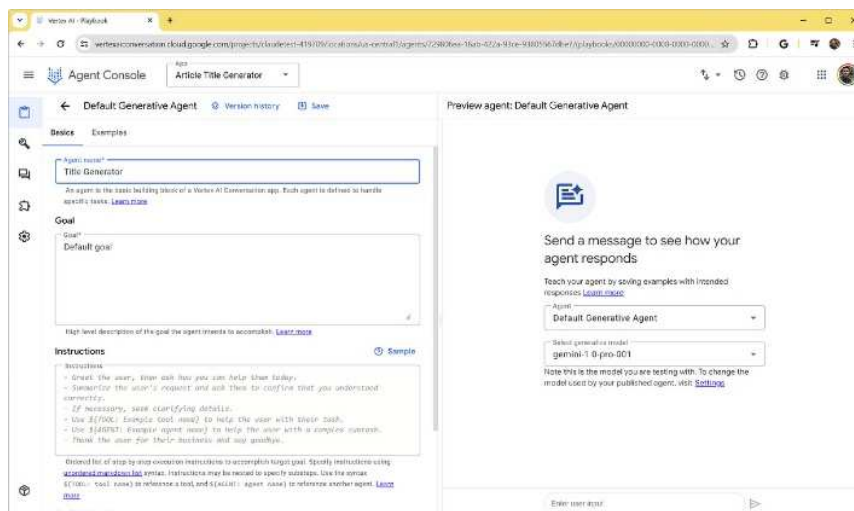
Google Gemini bietet verschiedene Ansätze für die Agent-Erstellung:

Gemini API und Google AI Studio: Google AI Studio bietet eine webbasierte Umgebung für die Erstellung und das Fine-Tuning von Gemini-basierten Agenten. Entwickler können hier Prompts optimieren, verschiedene Modellversionen testen und Agenten für spezifische Anwendungsfälle konfigurieren.

Vertex AI Agent Builder: Vertex AI bietet Enterprise-grade Tools für die Erstellung komplexer AI-Agenten. Diese können mit Unternehmensdaten trainiert, in bestehende Google Cloud-Services integriert und mit robusten Sicherheits- und Compliance-Features ausgestattet werden. Die Plattform unterstützt RAG-Systeme, Custom Knowledge Bases und die Integration mit Google Workspace-Anwendungen.

Extensions und Function Calling: Gemini-Agenten können mit Google-Services wie Gmail, Google Drive, Google Maps und YouTube integriert werden. Dies ermöglicht die Erstellung von Agenten, die nahtlos mit dem Google-Ökosystem arbeiten. Die Function-Calling-Fähigkeiten erlauben es, externe APIs einzubinden und komplexe Workflows zu orchestrieren.

Das **Agent Development Kit (ADK)** von Google bietet ein leistungsstarkes und flexibles Framework für die Entwicklung von Agentic-AI-Lösungen, insbesondere in Verbindung mit den Google Gemini-Modellen. Es ermöglicht die Erstellung komplexer Multi-Agent-Systeme, die in der Lage sind, spezialisierte Aufgaben zu übernehmen und miteinander zu kommunizieren. Durch die Unterstützung verschiedener Large Language Models (LLMs) und die Integration in Google Cloud-Dienste wie Vertex AI bietet das ADK eine skalierbare und sichere Plattform für den Unternehmenseinsatz.



1. Funktionsumfang 6,3/10

- **Anzahl unterstützter LLMs:** Unterstützt Google Gemini-Modelle sowie Integrationen mit Frameworks wie LangChain und CrewAI. ADK unterstützt weitere Modelle. 5/10
- **Multi Agent Architekturen und Hierarchien:** Ermöglicht die Erstellung von Multi-Agenten-Systemen mit Subagenten und Agent2Agent-Kommunikation. 8/10
- **Workflow-Funktionalität:** Bietet Integration mit Google Cloud-Diensten wie Apigee, Vertex AI Search und AlloyDB. 3/10
- **Anzahl API und Schnittstellen:** Verfügt über REST APIs, Pub/Sub-Integration und Unterstützung für das Model Context Protocol (MCP) 4/10
- **MCP Unterstützung:** Vertex und ADK unterstützen MCP für die Kommunikation zwischen Agenten und externen Systemen. 7/10
- **RAG und Vektordatenbank Integration:** Integriert Retrieval-Augmented Generation (RAG) mit Tools wie Vertex AI Search und AlloyDB. 7/10
- **Session übergreifendes Gedächtnis:** Bietet Sitzungsmanagement und Memory Bank für persistente Konversationen. 8/10
- **Persistente Workflow Steuerung:** Unterstützt Observability, Logging und Tracing für Agenten-Workflows. 8/10

2. Offenheit / Open Source 4/10

Die Sprachmodelle Gemini und Vertex sind proprietär, ADK sind Open Source.

3. Anpassbarkeit 7/10

Ermöglicht die Erstellung benutzerdefinierter Agenten mit Unterstützung für eigene Modelle und Frameworks

4. Verbreitung & Community 8/10

Gemini hat riesige Nutzerbasis, Vertex und ADK sind naturgemäß nur ein Bruchteil davon.

5. Erlernbarkeit 7/10

Bietet Low-Code-Optionen und umfassende Dokumentation für Entwickler. Das ADK bietet eine strukturierte Entwicklungsumgebung mit einer interaktiven Benutzeroberfläche, die den Lernprozess erleichtert. Dennoch erfordert die Komplexität der Multi-Agent-Architekturen Einarbeitungszeit.

6. Enterprise-Tauglichkeit 10/10

Integriert in die Google Infrastruktur mit Sicherheits- und Compliance-Funktionen

7. Sicherheit & Governance 10/10

Bietet Sicherheitsfunktionen wie IAM, Verschlüsselung und Compliance-Tools.

Die **Agentic-AI-Lösungen von Gemini**, insbesondere der **Vertex AI Agent Builder**, **ADK** und **Agent Assist**, bieten eine leistungsstarke und skalierbare Infrastruktur für die Entwicklung und Bereitstellung von KI-Agenten. Mit Unterstützung für Google Gemini-Modelle, Integration von Frameworks wie LangChain und CrewAI sowie Tools wie AlloyDB und Vertex AI Search ermöglichen sie die Erstellung komplexer Multi-Agenten-Systeme.

Die Plattform bietet sowohl Entwicklern als auch Citizen-Developern Low-Code-Optionen und umfassende APIs für die Anpassung und Integration in bestehende Systeme. Obwohl die Lösungen proprietär sind, bieten sie durch die Unterstützung von Open-Source-Tools Flexibilität bei der Entwicklung. Mit Sicherheitsfunktionen wie IAM, Verschlüsselung und Compliance-Tools ist die Plattform für den Unternehmenseinsatz geeignet. Es handelt sich aber um mehrere verschiedene Tools (Vertex, SDK, Copilot Agent Assist) die nicht immer integriert sind und unterschiedliche Schwerpunkte bieten.



Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

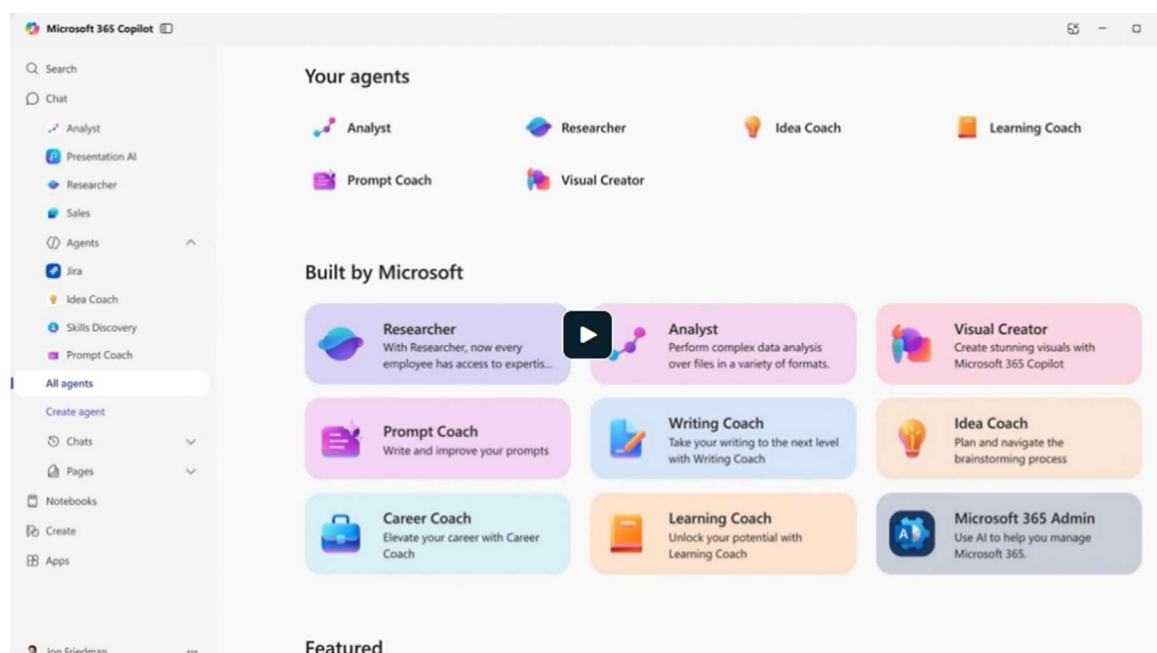
Microsoft Copilot repräsentiert eine tiefe Integration von AI-Agent-Funktionalitäten in das Microsoft 365-Ökosystem und Azure-Plattform.

Copilot Studio: Copilot Studio ist Microsofts Low-Code-Plattform für die Erstellung benutzerdefinierter Copilots. Die visuelle Entwicklungsumgebung ermöglicht es auch nicht-technischen Nutzern, spezialisierte Agenten zu erstellen. Diese können mit Microsoft 365-Daten trainiert, in Teams oder SharePoint integriert und mit Power Platform-Konnektoren erweitert werden. Die Plattform bietet vorgefertigte Templates für häufige Geschäftsszenarien.

Der **Semantic Kernel** ist ein Open-Source-Framework von Microsoft, das speziell dafür entwickelt wurde, KI-gestützte Anwendungen effizient und flexibel zu erstellen. Es dient als eine Art „Middleware“, die moderne KI-Modelle wie Large Language Models (LLMs) mit deinem eigenen Code verbindet – und das auf eine sehr modulare und skalierbare Weise

Microsoft 365 Integration: Copilot-Agenten können nahtlos in Microsoft 365-Anwendungen wie Outlook, Word, Excel, PowerPoint und Teams integriert werden.

Power Platform Konnektivität: Die Integration mit Power Platform ermöglicht es Copilot-Agenten, mit Hunderten von externen Services und Datenquellen zu interagieren.



1. Funktionsumfang 5,9/10

- **Anzahl unterstützter LLMs:** Der Semantic Kernel ermöglicht die Integration verschiedener Large Language Models (LLMs) und spezialisierter APIs, um komplexe Aufgaben zu bewältigen. 5/10
- **Multi Agent Architekturen und Hierarchien:** Copilot Studio unterstützt die Erstellung von Multi-Agent-Systemen, bei denen spezialisierte Agenten miteinander kommunizieren und Aufgaben delegieren können. 8/10
- **Workflow-Funktionalität:** Die Lösungen lassen sich nahtlos in bestehende Microsoft 365-Anwendungen integrieren, um Arbeitsabläufe zu automatisieren und zu optimieren. 5/10
- **Anzahl API und Schnittstellen:** Der Semantic Kernel bietet APIs zur Integration von Agenten in verschiedene Anwendungen und Systeme, einschließlich der Nutzung von Drittanbieter-Bibliotheken. Keine Interface-Kataloge. Einbindung in MS-365-Kosmos. 4/10
- **MCP Unterstützung:** Die Unterstützung des Model Context Protocol (MCP) ermöglicht eine erweiterte Interoperabilität zwischen Agenten und externen Systemen. 6/10
- **RAG und Vektordatenbank Integration:** Die Lösungen unterstützen Retrieval-Augmented Generation (RAG) und die Integration von Vektordatenbanken, um Agenten mit externem Wissen zu versorgen. Keine fertigen RAG Interfaces. 4/10
- **Session übergreifendes Gedächtnis:** Die Implementierung eines sessionübergreifenden Gedächtnisses ermöglicht es Agenten, sich an frühere Interaktionen zu erinnern und diese Informationen für zukünftige Aufgaben zu nutzen. 7/10
- **Persistente Workflow Steuerung:** Die Lösungen bieten Mechanismen zur Überwachung und Steuerung von Agenten-Workflows, einschließlich der Möglichkeit, den Status von Aufgaben zu verfolgen und bei Bedarf einzugreifen. 8/10

2. Offenheit / Open Source 4/10

Der Semantic Kernel und Copilot Studio sind ein Open-Source-Framework. Andere Komponenten sind proprietär.

3. Anpassbarkeit 8/10

Copilot Studio ermöglicht es Nutzern, Agenten mit benutzerdefinierten Funktionen und Tools auszustatten, einschließlich der Integration von Drittanbieter-Bibliotheken und eigenen APIs.

4. Verbreitung & Community 8/10

Die Lösungen haben eine wachsende Nutzerbasis und eine aktive Entwicklergemeinschaft, die zur Weiterentwicklung beiträgt.

5. Erlernbarkeit 7/10

Die Bereitstellung von umfangreichen Schulungsressourcen, Dokumentationen und Support erleichtert den Einstieg in die Nutzung der Lösungen. Copilot Studio bietet eine Low-Code/No-Code-Umgebung. Semantic Kernel ist komplexer.

6. Enterprise-Tauglichkeit 10/10

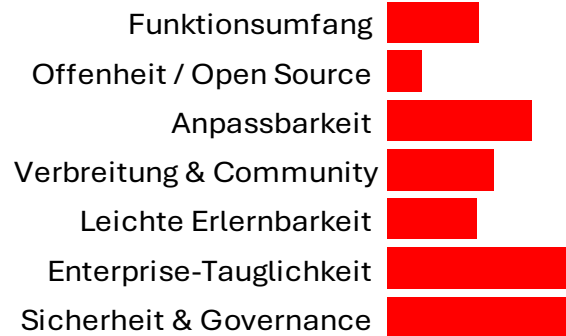
Die Lösungen sind für den Unternehmenseinsatz konzipiert und bieten Skalierbarkeit, Sicherheit und Compliance, die für große Organisationen erforderlich sind.

7. Sicherheit & Governance 10/10

Die Implementierung von Sicherheits- und Governance-Aspekten gewährleistet den verantwortungsvollen Einsatz der Lösungen in verschiedenen Umgebungen.

Die **Agentic-AI-Lösungen innerhalb von Microsoft Copilot**, insbesondere der **Semantic Kernel** und **Copilot Studio**, bieten eine leistungsstarke und flexible Plattform für die Entwicklung und Verwaltung von KI-gesteuerten Agenten. Der Semantic Kernel ermöglicht die Integration verschiedener Large Language Models (LLMs) und spezialisierter APIs. Copilot Studio bietet eine Low-Code/No-Code-Umgebung. Beide Lösungen unterstützen Multi-Agent-Architekturen, (RAG) und die Unterstützung von MCP.

Claude Agent SDK



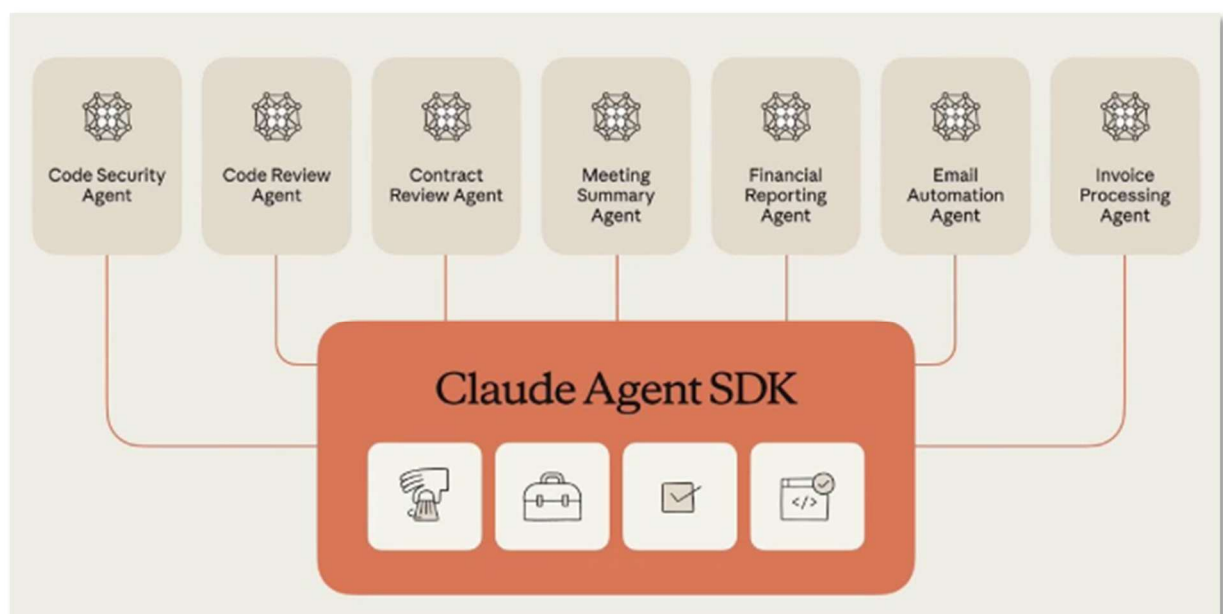
Das **Claude Agent SDK** ist ein von Anthropic bereitgestelltes Toolkit, das es Entwicklern ermöglicht, leistungsstarke **KI-Agenten** auf Basis des Claude-Modells (insbesondere Claude Sonnet) zu erstellen.

Es wurde aus dem ursprünglichen **Claude Code SDK** entwickelt und umbenannt, um seine breiteren Fähigkeiten über das bloße Codieren hinaus widerzuspiegeln – es dient dem Bau von Agenten für alle Arten von Aufgaben.

Hauptfunktionen & technische Eckdaten

- Es gibt eine Python- und auch eine TypeScript/Node-SDK.
- Entwickelt, damit Claude-Agenten „auf einem Computer“ arbeiten können: Dateien schreiben, Befehle ausführen, Iterationszyklen durchlaufen („agent loop“) – etwa Kontext sammeln, Aktion durchführen, Arbeit verifizieren.
- Unterstützt sogenannte „Agent Skills“: spezialisierte Tools oder Fähigkeiten, die Agenten hinzugefügt werden können, um realweltliche Aufgaben zu lösen.
- Integration mit dem open-Standard Model Context Protocol (MCP), wodurch der Agent Zugriff auf externe Datenquellen oder Tools bekommt.

Das System implementiert intelligente Context Compaction – dynamische Zusammenfassung und Verdichtung von Konversationshistorien über viele Turns hinweg.



1. Funktionsumfang 5,1/10

- **Anzahl unterstützter LLMs:** Die SDK baut exklusiv auf Claude-Modellen (z. B. Sonnet 4.5) auf. 3/10
- **Multi Agent Architekturen und Hierarchien:** das SDK unterstützt Agenten mit Tools, Skill-Sets, Sub-agenten und hierarchischen Steuerungen. 4/10
- **Workflow-Funktionalität:** Das SDK erlaubt die Einbindung von Workflows wie Codeausführung, Tools, Dateien, Web-Suche etc. 4/10
- **Anzahl API und Schnittstellen:** Integration über Python Code und API's 4/10
- **MCP Unterstützung:** Anthropic war Mitinitiator des MCP und die SDK erwähnt explizit MCP als Feature. 8/10
- **RAG und Vektordatenbank Integration:** Das SDK unterstützt Tools und Zugriff auf externe Datenquellen, aber nicht so explizit wie bei spezialisierten RAG-Frameworks. 4/10
- **Session übergreifendes Gedächtnis:** Es gibt Funktionen wie Memory, Kontextfenster und Langzeitkontext-Features („memory tool“). 6/10
- **Persistente Workflow Steuerung:** Agenten lassen sich überwachen, steuern, Logs erzeugen, Tools aufrufen. 8/10

2. Offenheit / Open Source 2/10

Das SDK ist öffentlich verfügbar (z. B. via GitHub) und unterstützt offene Standards wie MCP. Dennoch bleibt das Kernmodell proprietär.

3. Anpassbarkeit 8/10

Sehr hohe Anpassbarkeit: SDK-Bibliotheken in Python/TS, eigene Tools, Hooks, eigene Code-Integrationen.

4. Verbreitung & Community 6/10

Die Nutzer- und Entwicklerbasis wächst schnell, insbesondere im Agenten- und Coding-Agenten-Bereich, aber im Vergleich zu sehr etablierten Plattformen noch kleiner.

5. Erlernbarkeit 5/10

Die Dokumentation und Tutorials existieren, Quickstarts sind verfügbar. Dennoch erfordert die Nutzung eher Entwickler-Know-how, vor allem bei komplexen Agenten.

6. Enterprise-Tauglichkeit 10/10

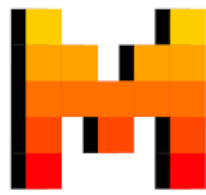
Enterprise-Anforderungen wie Sicherheit, Tools, Skalierung, Memory, Tool-Ausführung sind berücksichtigt. Die Plattform richtet sich an Produktion.

7. Sicherheit & Governance 10/10

Anthropic legt besonderen Wert auf Sicherheit, Alignment (z. B. Sonnet 4.5) und Governance, was sich auch in der Agent SDK-Ausrichtung zeigt.

Das Claude Agent SDK von Anthropic bietet eine leistungsfähige und moderne Basis zur Erstellung von Agentenlösungen. Die Plattform erlaubt hohe Anpassbarkeit und ist bereits für Enterprise-Einsatz gestaltet, inklusive Sicherheits- und Governance-Funktionen.

Einige Bereiche, wie etwa die explizite RAG-/Vektordatenbank-Integration oder die Größe der Entwickler-Community, sind noch nicht auf dem Niveau der etabliertesten Konkurrenz. Auch die Lernkurve ist etwas höher – anspruchsvollere Szenarien erfordern Entwicklungsressourcen und Expertise.



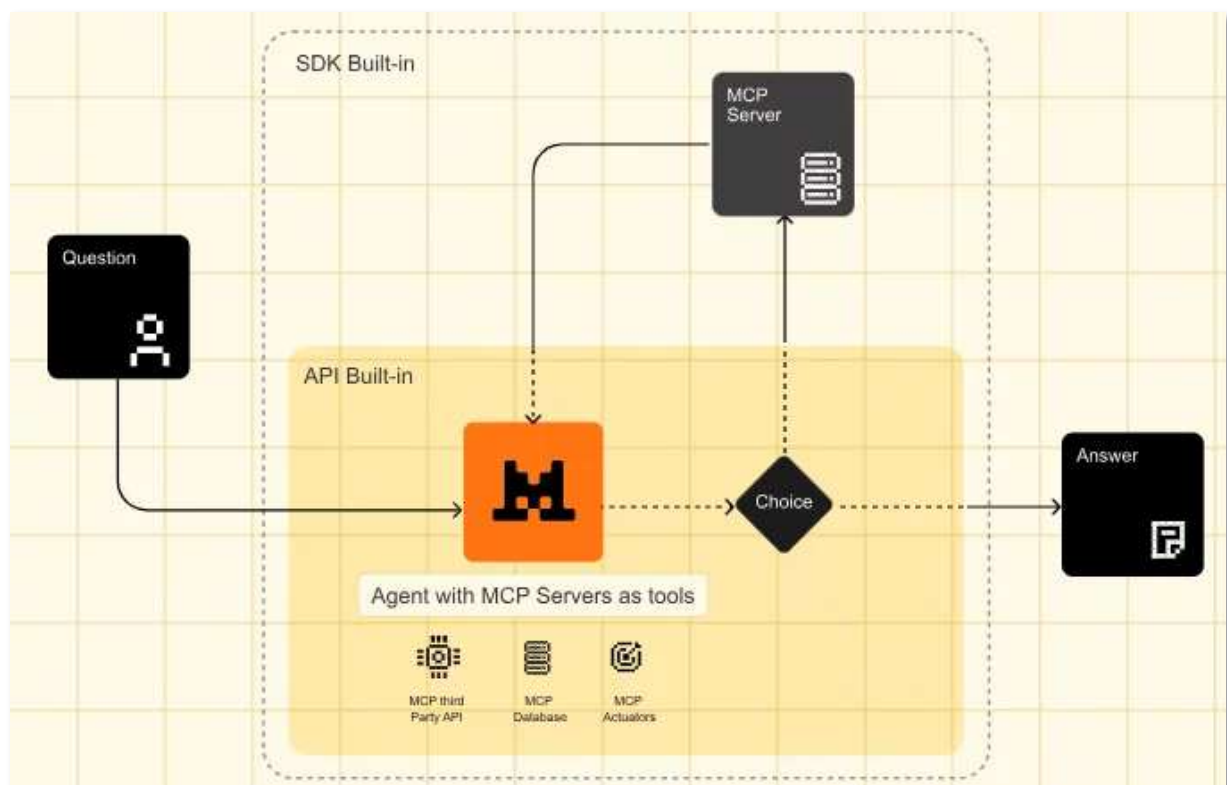
MISTRAL AI_



Mistral bietet mit **Le Chat** eine Chat-Plattform, die direkt Agenten einbindet:

- Innerhalb von Le Chat können User über die Agents-Tab personalisierte Agenten mit Namen, Beschreibung, Tonfall, Guardrails, Tools (z. B. Websuche, Code-Ausführung, Canvas, Gmail/Calendar) erstellen, testen und freigeben — inklusive Live-Vorschau und niedriger Latenz.
- **Agents API** (Mai 2025): Entwicklertools für komplexe Agenten mit Connectors (Code Exec, Image Generation, Web Search, Dokument-Library, MCP-Protokoll), persistentem Gedächtnis, Konversationen, Streaming und Multi-Agent-Orchestrierung.
- Agenten können rollenbasiert orchestriert, Kontext über Konversationen hinweg bewahrt und in Pipelines eingebunden werden (z. B. Entwickler-Agent, Finanz-Analyst, Reise- oder Ernährungsagent).

Le Chat Enterprise integriert einen Agent Builder und erlaubt die Verwendung selbst erstellter Agenten via "@Agent-Name" im Chat



1. Funktionsumfang 5,3/10

- **Anzahl unterstützter LLMs:** Die Mistral Agents API unterstützt derzeit die Modelle mistral-medium-latest und mistral-large-latest, mit Plänen, weitere Modelle zu integrieren. 2/10
- **Multi Agent Architekturen und Hierarchien:** Die API ermöglicht die Orchestrierung von Multi-Agent-Workflows, einschließlich der Übergabe von Aufgaben zwischen Agenten. 8/10
- **Workflow-Funktionalität:** Die Lösungen lassen sich in bestehende Systeme integrieren, um Arbeitsabläufe zu automatisieren und zu optimieren. 4/10
- **Umfassende API und Schnittstellenkapazitäten:** Die API bietet umfangreiche Funktionen zur Integration von Agenten in verschiedene Anwendungen und Systeme. 4/10
- **MCP Unterstützung:** Die Unterstützung des Model Context Protocol (MCP) als Client. 6/10
- **RAG und Vektordatenbank Integration:** Die Lösungen unterstützen Retrieval-Augmented Generation (RAG) und die Integration von Vektordatenbanken, um Agenten mit externem Wissen zu versorgen. 5/10
- **Session übergreifendes Gedächtnis:** Die Implementierung eines sessionübergreifenden Gedächtnisses ermöglicht es Agenten, sich an frühere Interaktionen zu erinnern und diese Informationen für zukünftige Aufgaben zu nutzen. 6/10
- **Persistente Workflow Steuerung:** Die Lösungen bieten Mechanismen zur Überwachung und Steuerung von Agenten-Workflows, einschließlich der Möglichkeit, den Status von Aufgaben zu verfolgen und bei Bedarf einzugreifen. 7/10

2. Offenheit / Open Source 9/10

Die Mistral Agents API ist ein Open-Source-Framework, das Entwicklern die Freiheit gibt, Agenten nach ihren Bedürfnissen anzupassen und zu erweitern.

3. Anpassbarkeit 6/10

Die API ermöglicht es Entwicklern, Agenten mit benutzerdefinierten Funktionen und Tools auszustatten, einschließlich der Integration von Drittanbieter-Bibliotheken und eigenen APIs.

4. Verbreitung & Community 5/10

Die Lösungen haben eine wachsende Nutzerbasis und eine aktive Entwicklergemeinschaft, die zur Weiterentwicklung beiträgt.

5. Erlernbarkeit 6/10

Die Bereitstellung von umfangreichen Schulungsressourcen, Dokumentationen und Support erleichtert den Einstieg in die Nutzung der Lösungen.

6. Enterprise-Tauglichkeit 9/10

Die Lösungen sind für den Unternehmenseinsatz konzipiert und bieten Skalierbarkeit, Sicherheit und Compliance, die für große Organisationen erforderlich sind.

7. Sicherheit & Governance 9/10

Die Implementierung von Sicherheits- und Governance-Aspekten gewährleistet den verantwortungsvollen Einsatz der Lösungen in verschiedenen Umgebungen.

Die **Agentic-AI-Lösungen innerhalb** der Mistral Agents API bieten eine leistungsstarke und flexible Plattform für die Entwicklung und Verwaltung von KI-gesteuerten Agenten. Die API ermöglicht die Erstellung komplexer Multi-Agent-Systeme, die in der Lage sind, spezialisierte Aufgaben zu übernehmen und miteinander zu kommunizieren. Durch die Unterstützung verschiedener Large Language Models (LLMs) und die Integration in Mistral-Dienste wie die Embeddings API bietet die Agents API eine skalierbare und sichere Plattform für den Unternehmenseinsatz. Insgesamt stellt die Mistral Agents API eine vielversprechende Lösung für die Entwicklung moderner, agentenbasierter KI-Anwendungen dar.



Amazon Bedrock

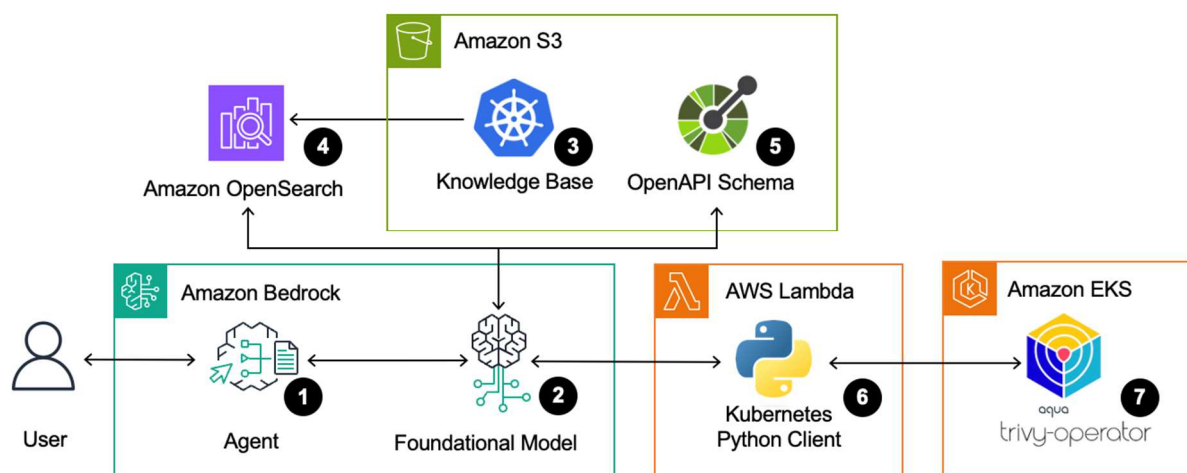
Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

Amazon Bedrock Agents ist der Hauptservice für die Erstellung intelligenter Agenten, die komplexe Aufgaben durch die Orchestrierung von API-Aufrufen ausführen können. Diese Agenten können mit verschiedenen Foundation Models (Claude, Llama, Titan, Jurassic) betrieben werden und nutzen eine Kombination aus Reasoning, Tool-Nutzung und Kontextverwaltung. Die Agenten können automatisch API-Dokumentationen interpretieren, Funktionsaufrufe planen und mehrstufige Workflows ausführen.

Sie umfasst modulare Komponenten wie Runtime, Memory (Kontextmanagement), Identity (sichere Authentifizierung), Gateway (Tool-Integration), Code Interpreter (ausführbaren Code in Sandbox-Umgebung) und Browser Tool (cloudbasierte Web-Interaktion) sowie Observability für Monitoring und Telemetrie. Agentorientierte Workflows lassen sich hiermit sicher, skalierbar und geräteübergreifend bereitstellen.

Parallel dazu bietet AWS auch das Open-Source-SDK Strands Agents (Python-basiert) zum schnellen Entwickeln von Agenten an. **Strands Agents SDK** ist ein Open-Source-Software Development Kit (SDK), das eine modellgetriebene und einfache Möglichkeit zum Entwickeln autonomer KI-Agenten (AI Agents) bietet. Es wurde für Entwickler optimiert, die leistungsfähige Agenten mit minimalem Code-Aufwand erstellen möchten – vom Prototyp bis zur Produktion

Neben den eigenen Angeboten, bietet AWS eine breite Auswahl an Agenten und Tools über den **AWS Marketplace** an.



1. Funktionsumfang 7,3/10

- **Anzahl unterstützter LLMs:** Bedrock bietet sofortigen Zugriff auf zahlreiche vortrainierte Foundation-Modelle verschiedener Anbieter (Anthropic Claude, Amazon Titan, Cohere, Mistral etc.) über eine einheitliche API. 8/10
- **Multi-Agent-Architekturen und Hierarchien:** Bedrock unterstützt visuelle Workflows (Bedrock Flows) und AgentCore-Komponenten für Agenten-Orchestrierung, Memory, Gateway usw. 7/10
- **Workflow-Funktionalität:** Bedrock bietet einen Low-Code/API-First Ansatz mit visuellem Builder (Flows) 7/10
- **Anzahl API und Schnittstellen:** Bedrock integriert sicher mit AWS-Services, Guards, Knowledge Bases, AgentCore und bietet API-/Visual-Flows (Amazon Web Services, Inc.). 9/10
- **MCP Unterstützung:** MCP-Server Einbindung möglich. 7/10
- **RAG und Vektordatenbank-Integration:** Bedrock unterstützt explizit Knowledge Bases für Retrieval-Augmented Generation (RAG). 5/10
- **Session-übergreifendes Gedächtnis:** Bedrock bietet AgentCore Memory (Short- und Long-Term Memory) für Agenten 8/10
- **Persistente Workflow-Steuerung:** Bedrock liefert robuste Orchestrierung: AgentCore Runtime, Observability, Checkpoints, Scale (Amazon Web Services, Inc.). 9/10

2. Offenheit / Open Source 4/10

Bedrock ist ein proprietärer AWS-Dienst. Strands Agent ist Open Source.

3. Anpassbarkeit 8/10

Bedrock erlaubt Modellanpassung mittels Fine-Tuning, Prompt-Engineering, Knowledge Bases. Visual Flows ergänze.

4. Verbreitung & Community 7/10

AWS wird von über 10.000 Organisationen genutzt als Basis für die Bedrock Dienste.

5. Erlernbarkeit 5/10

Die Bereitstellung von umfangreichen Schulungsressourcen, Dokumentationen und Support erleichtert den Einstieg in die Nutzung der Lösungen.

6. Enterprise-Tauglichkeit 10/10

Höchste Sicherheit (ISO, HIPAA etc.), Guardrails, Compliance, AgentCore-Scale.

7. Sicherheit & Governance 10/10

Guardrails, Datenschutz (keine Datennutzung zum Training), Compliance (ISO, HIPAA, FedRAMP etc.), Monitoring

Amazon Bedrock ist eine leistungsfähige, vollständig verwaltete Generative-AI-Plattform, ideal für schnelle Produktiv-Einsätze mit vortrainierten Foundation-Modellen. Dank Flows, AgentCore, Knowledge Bases und Guardrails ermöglicht es visuelle Low-Code-Orchestrierung, RAG-Anwendungen und robuste Sicherheitsmechanismen. Der Einstieg gelingt rasch, ohne Infrastrukturmanagement – perfekt für Teams mit begrenzter ML-Expertise und Projekte, die schnell skalieren sollen.

„**Strands Agents SDK**“ ist ein Open-Source-Software Development Kit (SDK), das eine modellgetriebene und einfache Möglichkeit zum Entwickeln autonomer KI-Agenten (AI Agents) bietet.



Azure AI Agent Service

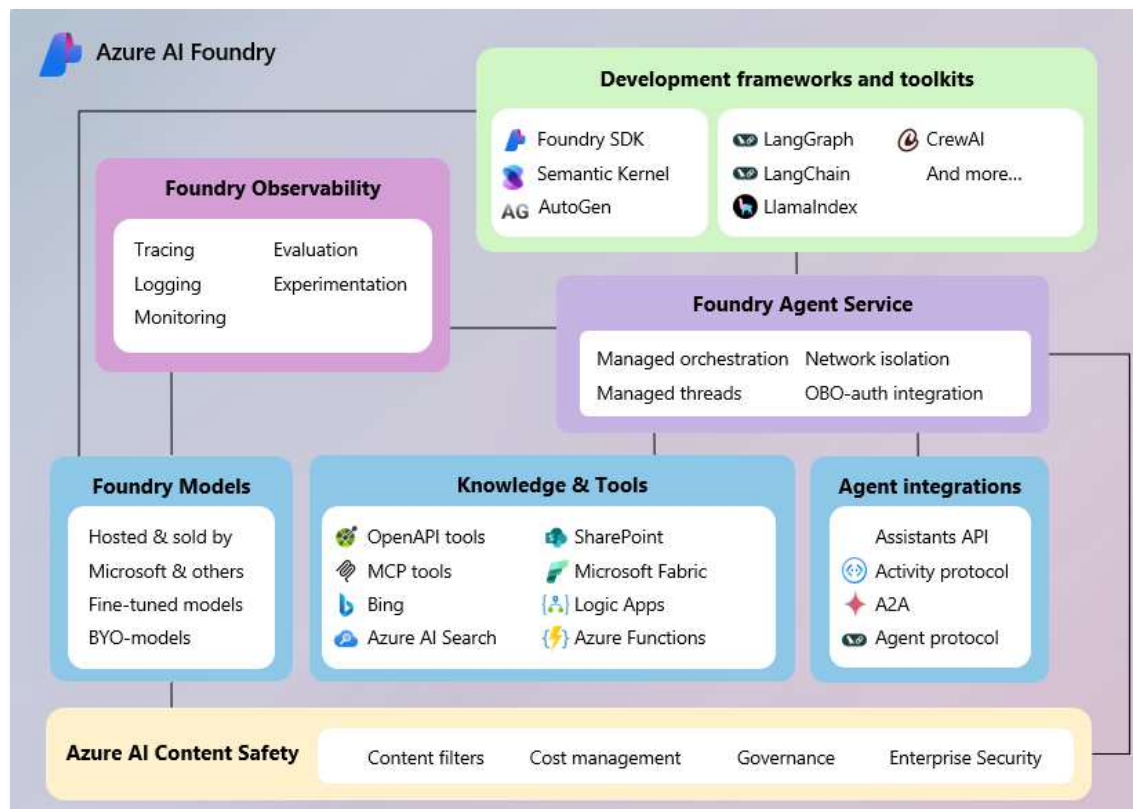
Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

Azure nutzt das neue **Azure AI Foundry Agent Service** als Produktionsplattform für intelligente Agenten. Dieses System verbindet Modelle, Tools, Governance und Frameworks in einem sicheren, skalierbaren Runtime-System.

Agenten sind in Azure Foundry modulare Einheiten mit spezifischen Rollen, ausgestattet mit LLM, Instruktionen und Tools. Sie unterstützen Multi-Agent-Orchestrierung, Tools wie Bing Search, Logic Apps oder Azure Functions, volle Observability (Tracing, Thread-Struktur, Application Insights) sowie Trust-Funktionen wie RBAC, Netzwerkisolation, Verschlüsselung und Content-Safety.

Zusätzlich ermöglicht die GA-Version Features wie „Connected Agents“ für Multi-Agent-Systeme, „Deep Research tool“ (mehrstufige Recherche) und MCP-Integration sowie Auslöser über Logic Apps und Debugging via VS Code Erweiterung.

Daneben bietet Microsoft mit dem Copilot Studio Low-Code-Umgebung, enge Integration mit Microsoft 365.



1. Funktionsumfang 7/10

- **Anzahl unterstützter LLMs:** Microsoft bietet Zugang zu verschiedenen LLMs – darunter Azure OpenAI (GPT-4, GPT-3.5), xAI's Grok, Meta Llama, Cohere und mehr. 8/10
- **Multi-Agent-Architekturen und Hierarchien:** Azure AI Foundry und Agent Service erlauben Multi-Agent-Orchestrierung (Agent Factory Patterns) – einschließlich Tool-Nutzung, Planung, Reflexion, ReAct-Pattern, Multi-Agent-Kollaboration. 7/10
- **Workflow-Funktionalität:** Agent Service bietet No-Code/Low-Code-Erstellung, Integration in Logic Apps, Azure Functions, APIs; Copilot Studio unterstützt visuelle Gestaltung von Agenten und Workflows 7/10
- **Anzahl API und Schnittstellen:** Integration mit SharePoint, Bing, Azure AI Search, Fabric, Logic Apps, Functions, OpenAPI, Action Tools wie Code Interpreter etc. 7/10
- **MCP Unterstützung:** Der Dienst unterstützt MCP: Agenten können über MCP Tooling kommunizieren und externe MCP-Server anbinden. 7/10
- **RAG und Vektordatenbank-Integration:** Agenten können Wissen über Azure AI Search, Bing, SharePoint und weitere Schritt-für-Schritt-Retrieval-Tools einbinden (RAG-Pipelines) 5/10
- **Session-übergreifendes Gedächtnis:** Azure AI Agent Service managt conversation state und agent-internes Memory über Sitzungen 8/10
- **Persistente Workflow-Steuerung:** Unterstützt Observability, OpenTelemetry-Metriken, Logging, Tracing, State Management, Wiederaufnahme, Tool Calls, Replay & Monitoring 9/10

2. Offenheit / Open Source 4/10

Die Azure-Lösungen sind proprietäre Azure-Services.

3. Anpassbarkeit 7/10

Azure AI Foundry & Agent Service ermöglichen weitgehende Anpassung: eigene Modelle, Fine-Tuning, Tool-Kits, visuelle & Code-basierte Gestaltung, Copilot Tuning,

4. Verbreitung & Community 7/10

Azure Agent-Lösungen sind bereits bei vielen Organisationen im Einsatz; Microsoft sieht Agentic-AI als zentrales Zukunftselement. Copilot & Foundry verankert im Microsoft-Ökosystem.

5. Erlernbarkeit 5/10

Copilot Studio ermöglicht niederschweligen Einstieg via Low-Code / Natural Language. Die Foundry/Agent Service ist eher pro-entwicklerorientiert und hat moderate Lernkurve.

6. Enterprise-Tauglichkeit 10/10

Azure bietet Enterprise-Grade Features: Sicherheit, Compliance, Entra ID, RBAC, Netzwerkisolierung, MVP-Tools, Governance, Observability, SLA-geeignete Infrastruktur.

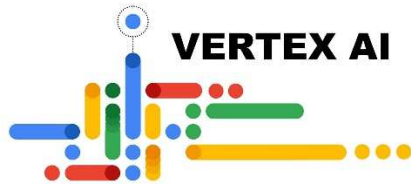
7. Sicherheit & Governance 10/10

Robuste Sicherheits- und Governance-Features: Microsoft Entra Agent ID, Purview-Integration, Compliance Controls, DLP, RBAC, sichere Authentifizierung und Isolation.

Die **Agentic-AI-Lösungen auf Microsoft Azure**, namentlich Azure AI Agent Service zusammen mit Azure AI Foundry und Microsoft Copilot Studio, bilden ein umfassendes, enterprise-taugliches System zur Entwicklung, Orchestrierung und Verwaltung autonomer KI-Agenten.

Die Plattform stützt sich auf einen umfangreichen Modellkatalog mit über 1.900 LLMs und unterstützt Multi-Agent-Designs, RAG-Workflows, MCP-Kommunikation.

Azure bietet eine fortschrittliche, sichere und skalierbare Umgebung für Agentic-AI. Es setzt jedoch die Nutzung verschiedener Komponenten voraus und die Integration scheint teilweise mehr auf einer marketing- als auf einer tatsächlichen technischen Grundlage zu basieren.

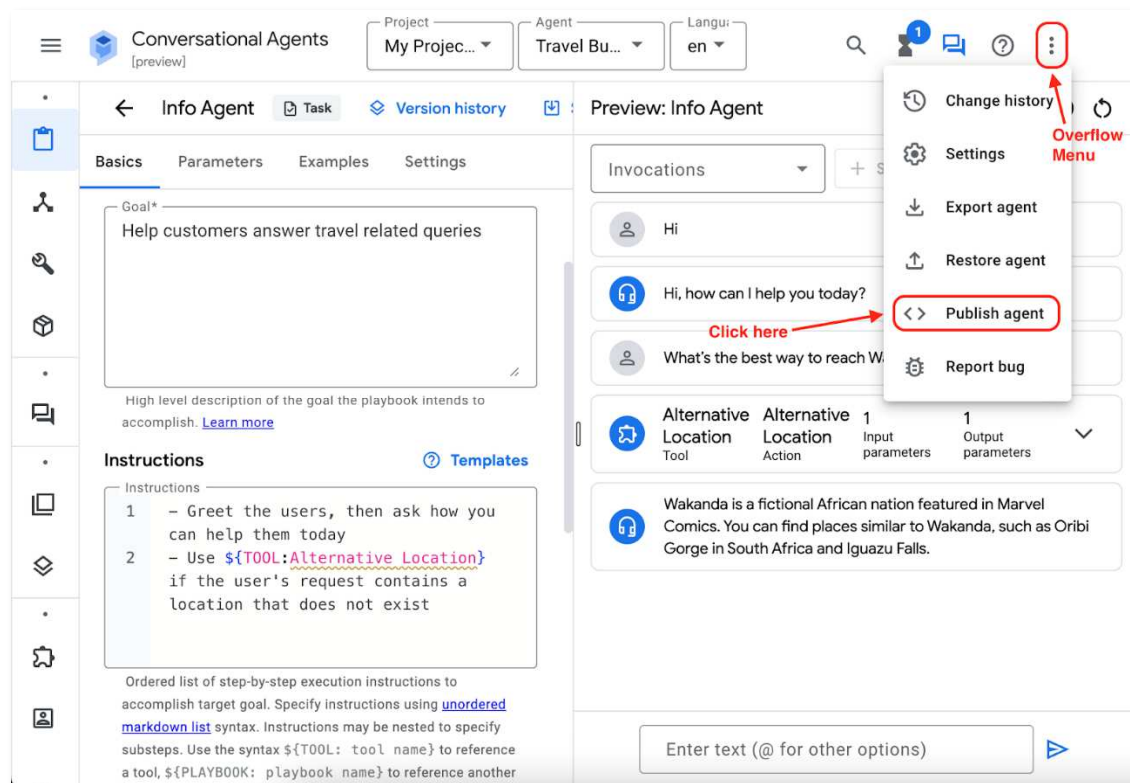


Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

Der **Vertex AI Agent Builder** ist Googles zentrale Plattform zum Erstellen intelligenter, generativer KI-Agenten. Er kombiniert klassische Chatbot-Technologien wie Intent- und Entity-Erkennung sowie zustandsbasierte Dialogsteuerung mit leistungsfähigen Large Language Models (LLMs) aus Googles Gemini-/PaLM-Familie.

Dadurch lassen sich sowohl strukturierte, regelbasierte Dialoge als auch offene, natürliche Konversationen in einem System umsetzen. Die Plattform ist Teil von Vertex AI und bietet eine No-Code/Low-Code-Umgebung, die den gesamten Lebenszyklus eines KI-Agenten – von der Entwicklung über Tests bis hin zu Deployment und Skalierung – abdeckt.

Sie unterstützt verschiedene Kanäle wie Web, Mobile, Voice oder Callcenter, erlaubt die Integration von Unternehmensdaten und ist cloud-native, sicher und skalierbar. Typische Einsatzbereiche sind Kundensupport, virtuelle Callcenter-Agenten oder interne Assistenten für HR, IT oder Wissensmanagement. Kurz gesagt: Der Vertex AI Agent Builder vereinfacht die Entwicklung leistungsfähiger, flexibler und generativer Conversational Agents für Unternehmen.



1. Funktionsumfang 6,9/10

- **Anzahl unterstützter LLMs:** Unterstützt Google Gemini-Modelle sowie Integrationen mit Frameworks wie LangChain und CrewAI. ADK unterstützt weitere Modelle. 4/10
- **Multi Agent Architekturen und Hierarchien:** Ermöglicht die Erstellung von Multi-Agenten-Systemen mit Subagenten und Agent2Agent-Kommunikation. 9/10
- **Workflow-Funktionalität:** Bietet Integration mit Google Cloud-Diensten wie Apigee, Vertex AI Search und AlloyDB. Kein klassischen Workflow tool. 7/10
- **Anzahl API und Schnittstellen:** Verfügt über REST APIs, Pub/Sub-Integration und Unterstützung für das Model Context Protocol (MCP) 5/10
- **MCP Unterstützung:** Unterstützt MCP für die Kommunikation zwischen Agenten und externen Systemen. 7/10
- **RAG und Vektordatenbank Integration:** Integriert Retrieval-Augmented Generation (RAG) mit Tools wie Vertex AI Search und AlloyDB. 6/10
- **Session übergreifendes Gedächtnis:** Bietet Sitzungsmanagement und Memory Bank für persistente Konversationen. 8/10
- **Persistente Workflow Steuerung:** Unterstützt Observability, Logging und Tracing für Agenten-Workflows. 9/10

2. Offenheit / Open Source 5/10

Die Cloud Lösungen sind proprietär, jedoch mit Unterstützung für Open-Source-Tools wie LangChain und CrewAI. Das ADK ist ein Open-Source-Framework, das Entwicklern die Freiheit gibt, Agenten nach ihren Bedürfnissen anzupassen und zu erweitern.

3. Anpassbarkeit 7/10

Ermöglicht die Erstellung benutzerdefinierter Agenten mit Unterstützung für eigene Modelle und Frameworks

4. Verbreitung & Community 5/10

Wird von Google Cloud unterstützt, mit wachsender Nutzerbasis und Community

5. Erlernbarkeit 6/10

Bietet Low-Code-Optionen und umfassende Dokumentation für Entwickler. Das ADK bietet eine strukturierte Entwicklungsumgebung mit einer interaktiven Benutzeroberfläche, die den Lernprozess erleichtert. Dennoch erfordert die Komplexität der Multi-Agent-Architekturen Einarbeitungszeit.

6. Enterprise-Tauglichkeit 10/10

Integriert in die Google Cloud-Infrastruktur mit Sicherheits- und Compliance-Funktionen

7. Sicherheit & Governance 10/10

Bietet Sicherheitsfunktionen wie IAM, Verschlüsselung und Compliance-Tools.

Die **Agentic-AI-Lösungen auf der Google Cloud Platform**, insbesondere der **Vertex AI Agent Builder** und **Agent Assist**, bieten eine leistungsstarke und skalierbare Infrastruktur für die Entwicklung und Bereitstellung von KI-Agenten. Mit Unterstützung für Google Gemini-Modelle, Integration von Frameworks wie LangChain und CrewAI sowie Tools wie AlloyDB und Vertex AI Search ermöglichen sie die Erstellung komplexer Multi-Agenten-Systeme.

Die Plattform bietet sowohl Entwicklern als auch Citizen-Developern Low-Code-Optionen und umfassende APIs für die Anpassung und Integration in bestehende Systeme. Obwohl die Lösungen proprietär sind, bieten sie durch die Unterstützung von Open-Source-Tools Flexibilität bei der Entwicklung. Mit Sicherheitsfunktionen wie IAM, Verschlüsselung und Compliance-Tools ist die Plattform für den Unternehmenseinsatz geeignet.

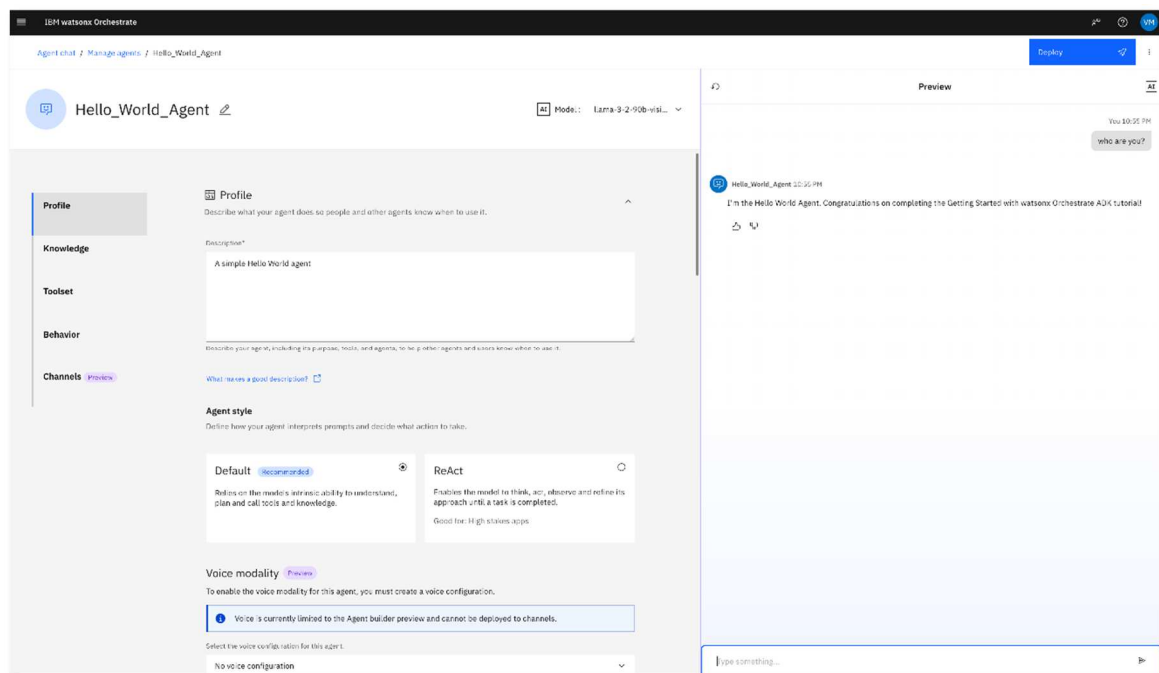


Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

IBM bietet mit **watsonx Orchestrate** eine umfassende Plattform zur Erstellung und Verwaltung intelligenter Agenten. Die Highlights:

- **Agent Builder & Flow Builder (No-Code):** Ermöglicht es, AI-Agenten und Workflows ohne Programmierung zu erstellen, zu konfigurieren und zu testen – ideal für Business-Nutzer.
- **Pre-built Agents:** Vordefinierte Agenten für Sales und Procurement, ausgestattet mit Domänen-Wissen, Integrationen und Tools, sind nun allgemein verfügbar.
- **Agent-Katalog & Management:** Steuerung, Monitoring und Governance aller Agenten über ein zentrales Interface.
- **AI Gateway:** Flexible Auswahl von LLMs – entweder IBMs eigene Granite-Modelle oder kundeneigene Modelle – als Basis für Agenten.

Diese Lösung eignet sich ideal für Business-Teams, die ohne großen Entwicklungsaufwand produktionsreife Agenten erstellen und verwalten wollen.



1. Funktionsumfang 6,5/10

- **Anzahl unterstützter LLMs:** IBM Watsonx unterstützt eine Vielzahl von Large Language Models (LLMs), darunter eigene Modelle wie Granite sowie Open-Source-Modelle wie Meta LLaMA 2 und Mistral. 5/10
- **Multi Agent Architekturen und Hierarchien:** Die Plattform ermöglicht die Erstellung und Verwaltung von virtuellen Agenten. Watsonx Assistant unterstützt die Integration von Backend-Systemen und Robotic Process Automation (RPA), um Aufgaben effizient zu erledigen. 5/10
- **Workflow-Funktionalität:** Die Plattform ermöglicht die Automatisierung von Geschäftsabläufen und die nahtlose Einbindung in bestehende IT-Infrastrukturen. 4/10
- **Anzahl API und Schnittstellen:** IBM stellt umfangreiche APIs zur Verfügung, darunter REST- und Python-Schnittstellen, um mit Watsonx-Diensten zu interagieren. 5/10
- **MCP Unterstützung:** Mit Watson-X können externe MCP-Server integriert und eigene MCP-Server erstellt werden. 8/10
- **RAG und Vektordatenbank Integration:** Watsonx unterstützt Retrieval-Augmented Generation (RAG) und die Integration von Vektordatenbanken, um Agenten mit externem Wissen zu versorgen. Dies ermöglicht eine kontextualisierte und präzise Beantwortung von Anfragen. 8/10
- **Session übergreifendes Gedächtnis:** Die Plattform ermöglicht die Speicherung von Interaktionsdaten über verschiedene Sitzungen hinweg. 8/10
- **Persistente Workflow Steuerung:** IBM Watsonx bietet Funktionen zur Überwachung und Steuerung von Workflows, einschließlich der Möglichkeit, den Status von Aufgaben zu verfolgen und bei Bedarf einzugreifen. 9/10

2. Offenheit / Open Source 1/10

IBM Watsonx ist eine proprietäre Plattform.

3. Anpassbarkeit 7/10

Die Plattform bietet umfangreiche Anpassungsoptionen.

4. Verbreitung & Community 4/10

IBM Watsonx wird von vielen Unternehmen weltweit genutzt. Die Plattform hat eine wachsende Nutzerbasis und eine aktive Entwicklergemeinschaft.

5. Erlernbarkeit 5/10

Die benutzerfreundliche Oberfläche und die verfügbaren Tutorials unterstützen Entwickler bei der schnellen Implementierung.

6. Enterprise-Tauglichkeit 10/10

IBM Watsonx ist speziell für den Unternehmenseinsatz konzipiert und bietet Skalierbarkeit, Sicherheit und Compliance, die für große Organisationen erforderlich sind. Die Plattform unterstützt hybride Cloud-Umgebungen und ermöglicht eine nahtlose Integration in bestehende IT-Infrastrukturen.

7. Sicherheit & Governance 10/10

Watsonx integriert Governance-Funktionen, um eine verantwortungsvolle Nutzung von KI sicherzustellen. Die Plattform unterstützt die Überwachung von Modellen, die Einhaltung von Vorschriften und die Minimierung von KI-Bias, um ethische Standards zu wahren.

Die **IBM Watsonx-Lösungen** bieten eine umfassende Plattform für die Entwicklung und Bereitstellung von KI-Anwendungen im Unternehmensumfeld. Mit Unterstützung für eine Vielzahl von LLMs, darunter sowohl proprietäre als auch Open-Source-Modelle, ermöglicht Watsonx die Erstellung maßgeschneiderter KI-Lösungen für verschiedene Geschäftsanforderungen. Watsonx ist jedoch ein „geschlossenes System“, und die Produktstrategie von IBM wirkt insgesamt etwas unübersichtlich und wenig konsistent – ein Muster, das sich bereits bei früheren ML-Plattformen gezeigt hat.

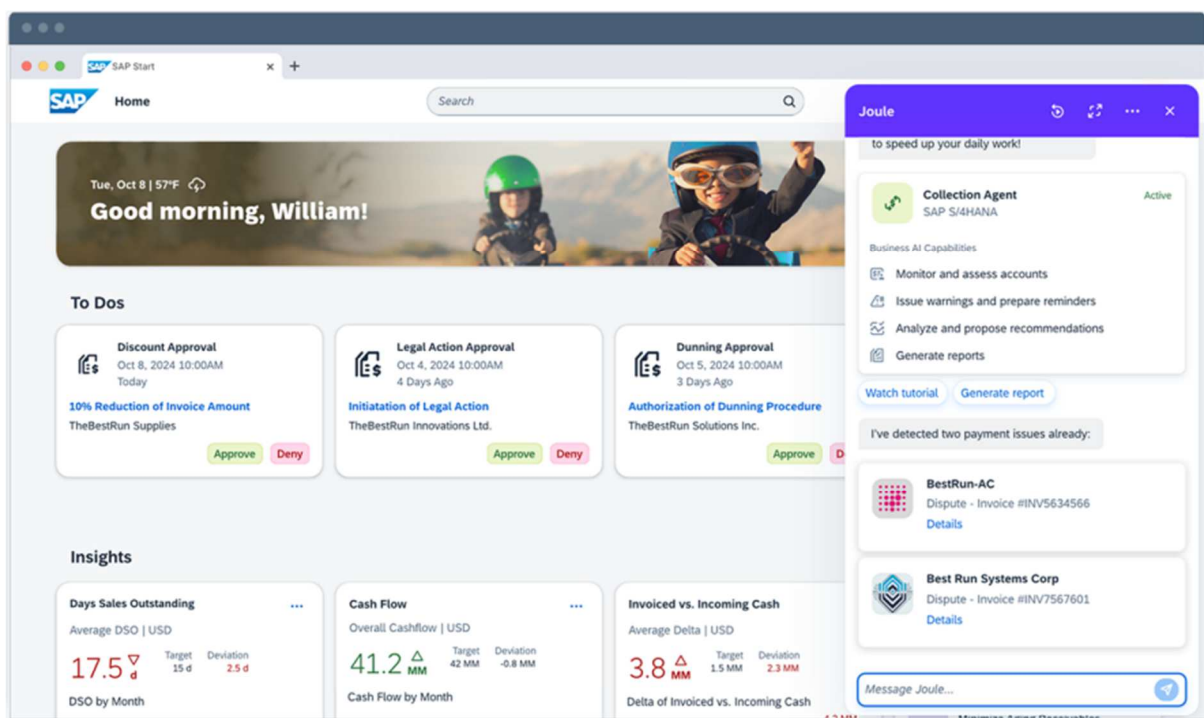


Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

SAP verfolgt eine Agentenstrategie innerhalb seiner **Business Technology Platform (BTP)** unter dem Schlagwort **Joule Agents**:

- **Content-Based Agents (low-code)**: Erzeugt über den **Joule Studio Agent Builder**. Schnell, konfigurierbar, REST-/OData-Integration, sicher und skalierbar – ideal für Geschäftsprozesse.
- **Code-Based Agents**: Für komplexere Anwendungen lässt sich Code, Frameworks (z. B. LangGraph, AutoGen) und maßgeschneiderte Integrationen einsetzen, um granulare Kontrolle über Logik und Workflows zu gewährleisten.
- **SAP AI Agent Architektur**: Integriert tief in Geschäftsabläufe über Business Data Cloud, BTP und generative AI Hub – Agenten planen, agieren, reflektieren autonom im Unternehmenskontext.

SAPs Ansatz kombiniert einfache Konfiguration mit der Möglichkeit zu hochgradig angepassten Agenten – besonders relevant in unternehmensweiten Automatisierungsprojekten.



1. Funktionsumfang 6,4/10

- **Anzahl unterstützter LLMs:** SAP nutzt verschiedene Large Language Models (LLMs) und integriert diese in ihre Business-AI-Lösungen, um kontextualisierte Entscheidungen und Aktionen zu ermöglichen. 5/10
- **Multi Agent Architekturen und Hierarchien:** Joule Agents ermöglichen die Bildung von Agententeams, die komplexe Geschäftsprozesse autonom ausführen, indem sie miteinander kommunizieren und zusammenarbeiten, aber nur begrenzte Agenten-Hierarchien. 5/10
- **Workflow-Funktionalität:** Die Agentic-AI-Lösungen von SAP sind nahtlos in die SAP Business Technology Platform (BTP) integriert und unterstützen klassische Geschäftsprozesse wie Finanzbuchhaltung, Personalwesen und Lieferkettenmanagement. 5/10
- **Anzahl API und Schnittstellen:** SAP bietet APIs und SDKs für die Integration von AI-Agenten in bestehende Systeme und Anwendungen, einschließlich Unterstützung für ABAP, Python und JavaScript. 5/10
- **MCP Unterstützung:** Die Plattform unterstützt die Nutzung MCP-Servern. 7/10
- **RAG und Vektordatenbank Integration:** Die Agentic-AI-Lösungen von SAP unterstützen Retrieval-Augmented Generation (RAG) und nutzen Vektordatenbanken für die Informationsbeschaffung und -verarbeitung. 7/10
- **Session übergreifendes Gedächtnis:** SAP-Agenten können über Sitzungen hinweg Kontext speichern und nutzen, um konsistente und personalisierte Interaktionen zu ermöglichen. 9/10
- **Persistente Workflow Steuerung:** Die Agentic-AI-Lösungen von SAP bieten Funktionen zur Überwachung, Protokollierung und Nachverfolgung von Agentenaktivitäten, um Transparenz und Kontrolle zu gewährleisten. 8/10

2. Offenheit / Open Source 2/10

SAP verfolgt einen offenen Ansatz und bietet Entwicklern Tools und Ressourcen, um AI-Agenten anzupassen und zu erweitern. Dennoch sind die Kernlösungen proprietär.

3. Anpassbarkeit 5/10

SAP ermöglicht es Unternehmen, AI-Agenten an ihre spezifischen Geschäftsanforderungen anzupassen, indem sie benutzerdefinierte Regeln, Workflows und Integrationen erstellen. Aber keine Integration eigenen Codes (z.B. in Python).

4. Verbreitung & Community 6/10

SAP hat eine riesige Kundenbasis und eine aktive Entwicklergemeinschaft, die die Agentic-AI-Lösungen nutzt und weiterentwickelt

5. Erlernbarkeit 5/10

SAP bietet umfangreiche Schulungsressourcen, Dokumentationen und Support, um Entwicklern und Unternehmen den Einstieg in die Agentic-AI-Lösungen zu erleichtern. Insgesamt komplexes Produkt.

6. Enterprise-Tauglichkeit 10/10

SAPs Agentic-AI-Lösungen sind für den Unternehmenseinsatz konzipiert und bieten Skalierbarkeit, Sicherheit und Compliance, die für große Organisationen erforderlich sind.

7. Sicherheit & Governance 10/10

SAP legt großen Wert auf Sicherheits- und Governance-Aspekte, einschließlich Datenschutz, Compliance und ethische KI-Praktiken, um den verantwortungsvollen Einsatz von AI-Agenten zu gewährleisten.

Die **Agentic-AI-Lösungen von SAP**, insbesondere die **Joule Agents**, der **SAP Generative AI Hub** und **SAP AI Core**, bieten eine leistungsstarke und skalierbare Plattform für die Entwicklung und Bereitstellung von KI-Agenten. Die Integration in die SAP Business Technology Platform (BTP) gewährleistet eine nahtlose Verbindung mit bestehenden Systemen und Datenquellen. Für AI-Agenten Lösungen im SAP-Kontext sicher eine Top-Wahl, als übergreifende Anwendung für die Erstellung von AI-Agenten gibt es geeignetere Lösungen.

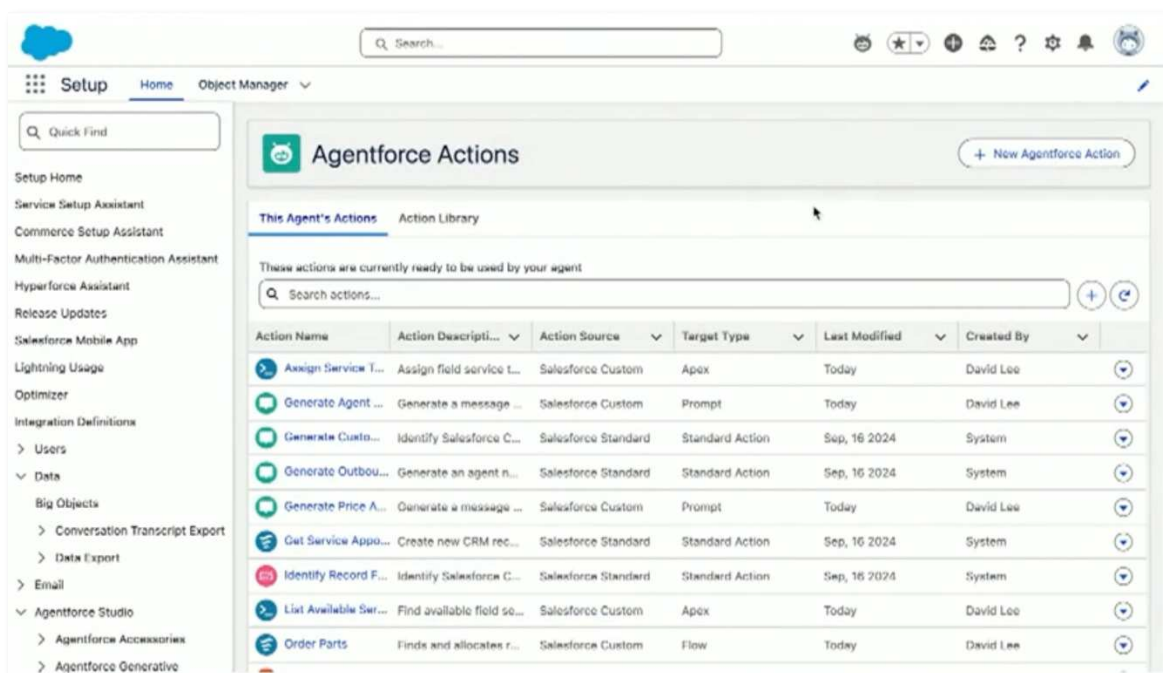


Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

Salesforce bündelt Agenten-Funktionalitäten unter dem Banner **Agentforce**, dem Nachfolger von Einstein:

- **Agentforce Framework:** Ermöglicht den Aufbau autonomer, zielorientierter KI-Agenten, die mit CRM-Daten arbeiten, handeln und komplexe Aufgaben selbstständig erledigen können.
- **Out-of-the-box Agenten:** Vorgefertigte Rollen wie Service Agent, Sales Coach, SDR, Buyer, Campaign Optimizer etc., lassen sich schnell anpassen und
- **Agent Builder & Model Builder:** Low-Code-Tools zur Anpassung oder Kreation neuer Agenten, inkl. Anbindung eigener LLMs, Flows, Prompts oder API-Integrationen.
- **Einstein Service Agent:** Automatisiert Kundenservice-Prozesse über Multichannel-Support, nutzt Vertrauensebenen, Maskierung und Guardrails für sichere Antworten, und arbeitet eng mit menschlichen Agenten zusammen.
- **Reasoning Engine & Trust Layer:** Agentforce 2.0 bietet tieferes Verständnis und schichtweise Sicherheit (Datenschutz, Verhaltensregeln), plus kontrollierte menschliche Übergabe.

Salesforce Agentforce zielt darauf ab, Agenten in CRM-Prozesse nahtlos und sicher zu integrieren – mit viel Out-of-box-Power, aber auch Anpassungsmöglichkeiten.



1. Funktionsumfang 6,3/10

- **Anzahl unterstützter LLMs:** Agentforce nutzt Large Language Models (LLMs) wie OpenAI's ChatGPT und Salesforce-eigene Modelle, um kontextualisierte und personalisierte Antworten zu generieren. 2/10
- **Multi Agent Architekturen und Hierarchien:** Agentforce ermöglicht die Erstellung und Verwaltung mehrerer autonomer Agenten, die in verschiedenen Geschäftsbereichen wie Vertrieb, Marketing und Kundenservice agieren, aber keine echten multihierarchischen AI-Agenten. 6/10
- **Workflow-Funktionalität:** Die Plattform integriert sich nahtlos in bestehende Salesforce-Workflows und -Prozesse, einschließlich der Nutzung von Salesforce Flows und Apex-Code. 5/10
- **Anzahl API und Schnittstellen:** Agentforce bietet APIs und SDKs für die Integration in externe Systeme und Anwendungen, einschließlich Unterstützung für REST und GraphQL. 6/10
- **MCP Unterstützung:** Die Plattform unterstützt die Nutzung MCP-Servern. 7/10
- **RAG und Vektordatenbank Integration:** Agentforce ermöglicht die Integration von Retrieval-Augmented Generation (RAG) und Vektordatenbanken, um relevante Informationen aus großen Datenmengen abzurufen und zu nutzen. 8/10
- **Session übergreifendes Gedächtnis:** Die Plattform unterstützt die Speicherung und Nutzung von Konversationskontext über mehrere Interaktionen hinweg, um konsistente und personalisierte Erlebnisse zu bieten 8/10
- **Persistente Workflow Steuerung:** Agentforce bietet Funktionen zur Überwachung, Protokollierung und Nachverfolgung von Agentenaktivitäten, um Transparenz und Kontrolle zu gewährleisten. (Persistent Systems) 8/10

2. Offenheit / Open Source 3/10

Salesforce verfolgt einen offenen Ansatz, indem es Entwicklern Tools und Ressourcen zur Anpassung und Erweiterung von Agentforce bietet. Dennoch sind die Kernlösungen proprietär.

3. Anpassbarkeit 6/10

Agentforce ermöglicht es Unternehmen, KI-Agenten an ihre spezifischen Geschäftsanforderungen anzupassen, indem sie benutzerdefinierte Regeln, Workflows und Integrationen erstellen. Aber keine Integration eigenen Codes (z.B. in Python).

4. Verbreitung & Community 6/10

Salesforce hat eine breite Kundenbasis und eine aktive Entwicklergemeinschaft, die Agentforce nutzt und weiterentwickelt.

5. Erlernbarkeit 6/10

Salesforce bietet umfangreiche Schulungsressourcen, Dokumentationen und Support, um Entwicklern und Unternehmen den Einstieg in Agentforce zu erleichtern.

6. Enterprise-Tauglichkeit 10/10

Agentforce ist für den Unternehmenseinsatz konzipiert und bietet Skalierbarkeit, Sicherheit und Compliance, die für große Organisationen erforderlich sind.

7. Sicherheit & Governance 10/10

Salesforce legt großen Wert auf Sicherheits- und Governance-Aspekte, einschließlich Datenschutz, Compliance und ethische KI-Praktiken, um den verantwortungsvollen Einsatz von Agentforce zu gewährleisten.

Die **Agentic-AI-Lösungen von Salesforce**, insbesondere die **Agentforce**-Plattform, bieten eine leistungsstarke und skalierbare Infrastruktur für die Entwicklung und Bereitstellung von KI-Agenten in Unternehmensumgebungen. Die Plattform integriert sich nahtlos in bestehende Salesforce-Workflows und -Prozesse und bietet umfangreiche APIs und SDKs für die Integration in externe Systeme. Für AI-Agenten Lösungen im Salesforce-Kontext sicher eine Top-Wahl, als übergreifende Anwendung für die Erstellung von AI-Agenten gibt es geeignetere Lösungen.



Funktionsumfang	
Offenheit / Open Source	
Anpassbarkeit	
Verbreitung & Community	
Leichte Erlernbarkeit	
Enterprise-Tauglichkeit	
Sicherheit & Governance	

Databricks fokussiert sich auf Agenten für datengetriebene Workflows mit hoher Governance:

- **AI Agent Bricks:** Produktions-optimierte Agenten, die auf Unternehmensdaten trainiert werden. Tools für Agentenbau unterstützen Qualität, Kostenoptimierung, synthetische Daten und automatische Evaluierung.
- **Agent Framework / Tools:** Erstellung von Agentenerweiterungen über Unity Catalog-Funktionen oder Code-basierte Tools (Python, SDKs), integriert in Spark-Ökosystem. Unterstützt strukturierte und unstrukturierte Datenzugriffe sowie REST-APIs
- **Model Context Protocol (MCP):** Standardisiert Tool-Zugriffe für Agenten (Managed oder eigener MCP-Server).

Databricks bietet dank seiner Data-Lakehouse-Architektur eine starke Grundlage für Agenten, die datenintensive, automatisierte und beobachtbare Prozesse ausführen.

Name	State	Served entities	Tags	Task	Created by	Last modified
dspy_rag_chat	Ready	main.chenmoney.dsp... (v 10)			Chen Chen	5 days ago
agents_mosaic_catalog-ll...	Ready	mosaic_catalog.llac_schema			Benjamin Brubaker	6 days ago
agents_smurching-default...	Not ready (Update canceled)	smurching.default.cuj_doc_b			William Smurching	9 days ago
agents_ivesh-cookbook...	Ready	avesh.cookbook.park_ranger			Avesh Singh	11 days ago
Llama_v3_2_3b_instruct	Not ready (Stopped)	system.ai.llama_v3_2_... (v 4)		Chat	Alvin Huang	12 days ago
agents_main-luis_moros...	Ready	main.luis_moros.simple_dspy			luis.moros@databricks.com	12 days ago
rfl_bot	Ready	demo_catalog_np.rfl... (v 15)			Andreas Jansen@Databricks	13 days ago
meta_llama_3_70b_instru...	Ready	system.ai.meta_llama... (v 2)		Chat	Benjamin Brubaker	14 days ago
category_demand_samaya	Ready	field_demos.samaya... (v 1)			Samaya-mathew@databricks	14 days ago
test-budget	Ready	models.default.addmo... (v 1)	foo: bar		Alexander Parris	14 days ago
genie-snap-aliquery	Ready	system.ai.meta_llama... (v 4)		Chat	chung.m@databricks.com	15 days ago

1. Funktionsumfang 6/10

- **Anzahl unterstützter LLMs:** Das Mosaic AI Agent Framework ermöglicht die Entwicklung von Agenten mit verschiedenen Large Language Models (LLMs), einschließlich OpenAI's GPT-Modelle und Anthropic's Claude. 4/10
- **Multi Agent Architekturen und Hierarchien:** Das Framework unterstützt die Erstellung komplexer Agentensysteme mit mehreren Agenten, die miteinander kommunizieren und kooperieren können, aber begrenzte Hierarchien. 5/10
- **Workflow-Funktionalität:** Mosaic AI lässt sich nahtlos in bestehende Databricks-Workflows integrieren, einschließlich Delta Tables und MLflow. 5/10
- **Anzahl API und Schnittstellen:** Das Framework bietet APIs und SDKs für die Integration von Agenten in externe Systeme und Anwendungen. 4/10
- **MCP Unterstützung:** Mosaic AI unterstützt die Nutzung von Kontextinformationen, um personalisierte und relevante Antworten zu generieren. 7/10
- **RAG und Vektordatenbank Integration:** Das Framework ermöglicht die Entwicklung von Retrieval-Augmented Generation (RAG)-Anwendungen und die Integration von Vektordatenbanken für die Informationsbeschaffung. 7/10
- **Session übergreifendes Gedächtnis:** Mosaic AI unterstützt die Speicherung und Nutzung von Konversationskontext über mehrere Interaktionen hinweg. 8/10
- **Persistente Workflow Steuerung:** Das Framework bietet Funktionen zur Überwachung, Protokollierung und Nachverfolgung von Agentenaktivitäten. 8/10

2. Offenheit / Open Source 8/10

Mosaic AI ist ein Open-Source-Framework, das Entwicklern die Freiheit gibt, Agenten nach ihren Bedürfnissen anzupassen und zu erweitern.

3. Anpassbarkeit 7/10

Das Framework ermöglicht es Unternehmen, Agenten an ihre spezifischen Geschäftsanforderungen anzupassen, indem sie benutzerdefinierte Regeln, Workflows und Integrationen erstellen.

4. Verbreitung & Community 5/10

Databricks hat eine breite Kundenbasis und eine aktive Entwicklergemeinschaft, die Mosaic AI nutzt und weiterentwickelt.

5. Erlernbarkeit 5/10

Databricks bietet umfangreiche Schulungsressourcen, Dokumentationen und Support, um Entwicklern und Unternehmen den Einstieg in Mosaic AI zu erleichtern.

6. Enterprise-Tauglichkeit 10/10

Mosaic AI ist für den Unternehmenseinsatz konzipiert und bietet Skalierbarkeit, Sicherheit und Compliance, die für große Organisationen erforderlich sind.

7. Sicherheit & Governance 10/10

Das Framework bietet Funktionen zur Sicherstellung der Datensicherheit, einschließlich Zugriffskontrollen, Datenprotokollierung und Compliance-Überwachung.

Das **Mosaic AI Agent Framework** von Databricks bietet eine leistungsstarke und skalierbare Plattform für die Entwicklung und Bereitstellung von Agentic-AI-Lösungen in Unternehmensumgebungen. Mit Unterstützung für verschiedene Large Language Models (LLMs) und der Integration in bestehende Databricks-Workflows ermöglicht es Entwicklern, komplexe Agentensysteme zu erstellen, die in der Lage sind, autonome Entscheidungen zu treffen und Aufgaben effizient auszuführen. Mit einem starken Fokus auf Sicherheit, Governance und Enterprise-Tauglichkeit stellt Databricks sicher, dass die Mosaic AI-Lösungen verantwortungsvoll und im Einklang mit regulatorischen Anforderungen eingesetzt werden können.

Wer hat die Studie erstellt?

Der **AI-Agenten-Kompass** wurde erstellt von der **oetti-ds GmbH**, einer unabhängigen Beratungsboutique mit Fokus auf Data Science, Machine Learning und Künstliche Intelligenz. Das Unternehmen ist herstellernerutral und produktunabhängig und verfügt über breite Umsetzungserfahrung in den Bereichen KI, Generative AI, AI-Agenten, MCP-Server, lokale LLM-Installationen, RAG-Integrationen, Fraud Detection, Marketing-Automatisierung, Kampagnenmanagement, Churn Prevention, Root Cause Analysis, Sprach- und Bilderkennung, Prozessautomatisierung, Datenstrategie, AI-Implementierung, Analytics-Plattformen sowie Softwareauswahl.

Zu den Kunden zählen namhafte Unternehmen wie Mercedes-Benz, Deutsche Bank, EnBW, REWE digital, GfK, arvato, TÜV Rheinland und Schwäbisch Hall.

Autor der Studie



Michael Oettinger ist Gründer und Geschäftsführer der oetti-ds GmbH und ausgewiesener Experte für Künstliche Intelligenz mit über 20 Jahren Projekterfahrung in den Bereichen Künstliche Intelligenz, Machine-Learning und Datenanalyse.

Er hat zahlreiche erfolgreiche Projekte für führende Unternehmen umgesetzt und verfügt über fundierte Expertise in relevanten Technologien und Tools (u. a. SQL, Python, Generative AI, AI-Agenten, RAG, MCP-Server und Cloud-Plattformen).

Er ist Autor des Fachbuchs *Data Science & AI* (3. Auflage, 2023) und unterrichtet als Dozent für Künstliche Intelligenz an der Hochschule der Medien in Stuttgart.



Bitte folgen sie mir auf LinkedIn: [linkedin.com/in/michael-oettinger/](https://www.linkedin.com/in/michael-oettinger/)

Jetzt einen Termin für ein Beratungsgespräch vereinbaren!



oetti-ds GmbH
Römerschanzenstr. 1
D - 82110 Germering

michael.oettinger@oetti-ds.de



<https://calendly.com/michael-oettinger-oetti-ds/kennenlernen>