Modellieren Sie Strategien des Context Protocol auf AWS

AWS Präskriptive Leitlinien

AWS Präskriptive Leitlinien: Modellieren Sie Strategien des Context Protocol auf AWS

Table of Contents

Einführung 1
Zielgruppe 2
Ziele 3
Was ist MCP? 5
Tools verstehen 5
Wann sollte MCP verwendet werden 8
Strategie für die Entwicklung von MCP-Tools 12
Umfang der Tools 13
Granular 13
Grobkörnig 14
Bewährte Methoden für die Festlegung des Umfangs von MCP-Tools 15
Werkzeugdefinitionen 16
Ansatz zur Werkzeugspezifikation 17
Docstring-Ansatz 18
Bewährte Methoden für Definitionen von MCP-Tools 19
Erkennung von Tools 19
Statische Definition 19
Dynamische Erkennung 20
Suchfunktion 21
Bewährte Methoden für die Entdeckung von MCP-Tools 21
Organisation der Tools 21
Bewährte Methoden für die Organisation von MPC-Tools 22
MCP-Hosting-Strategie 24
Hosting-Ansätze 24
Lokales Hosting 24
Remote-Hosting 26
MCP-Gateway 26
Bewährte Methoden für das Hosten von MCP-Servern 27
MCP-Verwaltungsstrategie 28
Authentifizierung und Autorisierung 28
Bewährte Methoden für die MCP-Authentifizierung und -Autorisierung 30
Steuerung der Last 30
Bewährte Methoden zur Steuerung der Auslastung 31
Operationelle Metriken 31

Mitwirkende 33
Verfassen 33
Überprüfend 33
Technisches Schreiben 33
Dokumentverlauf 34
Glossar 35
..... 35
A 36
B 39
C 41
D 45
E 49
F 51
G 53
H 55
I 56
L 59
M 60
O 64
P 67
Q 70
R 71
S 74
T 78
U 80
V 80
W 81
Z 82
..... lxxxiii

Modellieren Sie Context-Protokoll-Strategien auf AWS

Amazon Web Services (Mitwirkende)

März 2026 (Verlauf der Dokumente)

Dieser Leitfaden kann Ihnen bei der Entwicklung und Implementierung von MCP-Strategien (Model Context Protocol) in Ihrem gesamten Unternehmen helfen, um Ihre KI-Aktivitäten zu unterstützen. Da Agenten und Sprachmodelle für den Geschäftsbetrieb immer zentraler werden, ist die Etablierung einer MCP-Strategie für erfolgreiche Agentenlösungen von entscheidender Bedeutung.

In diesem Leitfaden werden drei grundlegende Säulen für den Aufbau einer MCP-Strategie untersucht: MCP-Tooldesign, MCP-Serverhosting und MCP-Governance. Durch die Berücksichtigung dieser miteinander verbundenen Komponenten können Unternehmen skalierbare, sichere und effektive Systeme für die Verwaltung des Modellkontextes in ihren KI-Implementierungen einrichten. Diese Leitlinien bieten umsetzbare Erkenntnisse und strategische Leitlinien für Unternehmen in jeder Phase ihrer KI-Entwicklung, von ersten Experimenten bis hin zu umfassenden Produktionsbereitstellungen. Dies hilft ihnen, maßgeschneiderte MCP-Lösungen zu entwickeln, die auf ihre spezifischen Bedürfnisse und Ziele zugeschnitten sind.

Diese Best Practices basieren auf realen Implementierungen von Unternehmen, die MCP auf Unternehmensebene einsetzen, einer Analyse der aktuellen MCP-Spezifikationsstandards und Erfahrungen aus kundenspezifischen Large Language Model (LLM) -Anwendungen in der Produktion.

KI-Systeme werden LLMs in einer Vielzahl von Anwendungsfällen immer ausgefeilter und robuster. LLMs zeichnen sich dadurch aus, dass sie natürliche Sprache verstehen, menschenähnliche Reaktionen erzeugen und komplexe Informationen nachvollziehen können. Um jedoch LLMs von Konversationsschnittstellen zu Systemen überzugehen, die komplexe Aufgaben autonom ausführen können, setzen Unternehmen auf agentische KI-Architekturen, KI-Systeme, die ihre Umgebung wahrnehmen, über Ziele nachdenken, autonome Entscheidungen treffen, mehrere Schritte koordinieren und Maßnahmen ergreifen können, um Ziele im Namen der Benutzer zu erreichen. Dieser agentische Ansatz hilft Unternehmen dabei, KI-Systeme zu entwickeln, die Benutzerabsichten mithilfe natürlicher Sprache verstehen, autonom über mehrere Datenquellen und Tools hinweg koordinieren und personalisierte Erlebnisse in einem Umfang bieten können, der mit herkömmlichen Anfrage-Antwort-Mustern nicht möglich war. Um diese Agenten leistungsfähiger zu machen, müssen Unternehmen Zugriff auf ihre vorhandenen Tools und Daten gewähren, um das

kontextuelle Verständnis der Agenten zu erweitern und es ihnen zu ermöglichen, im Namen eines Benutzers zu handeln.

MCP bietet ein standardisiertes Protokoll für die Integration von KI-Tools, das eine konsistente Kommunikation zwischen Agenten und externen Ressourcen ermöglicht. MCP selbst definiert den Kommunikationsstandard, aber seine effektive Implementierung erfordert eine sorgfältige Berücksichtigung von Architekturmustern, Sicherheitsmodellen, betrieblichen Praktiken und Strategien zur Leistungsoptimierung, um skalierbare, sichere und wartungsfreundliche Lösungen zu erreichen.

Dieser Leitfaden fasst die Erfahrungen aus MCP-Implementierungen in Unternehmen zusammen und bietet umsetzbare Empfehlungen, die auf das Well-Architected Framework abgestimmt sind.AWS Es behandelt Strategien für das Design von MCP-Tools, das MCP-Serverhosting und die MCP-Governance, die für die Entwicklung Ihrer eigenen MCP-Lösungen unerlässlich sind. Die Empfehlungen in diesem Leitfaden beziehen sich auf die folgenden fünf Säulen des AWS Well-Architected Framework:

- Sicherheit — Token-Isolierung, eingeschränkte Zugangsdaten, separate Autorisierung read/write
- Operational Excellence — Genauigkeitsmetriken für die Werkzeugauswahl, wertvolle Datensätze für Regressionstests
- Zuverlässigkeit — Geschwindigkeitsbegrenzung pro Benutzer und pro Tool, Lastabbau
- Leistungseffizienz — Workflow-spezifische Tools, Toolfilterung, semantische Suche zur Reduzierung der Nutzung von Kontextfenstern
- Kostenoptimierung — Teamübergreifend wiederverwendbare MCP-Server, geringere Token-Kosten pro Anfrage durch Toolfilterung

Zielgruppe

Dieser Leitfaden richtet sich an Architekten, Entwickler und Technologieführer, die agentische KI-Lösungen in ihren Organisationen implementieren. Um die Konzepte in diesem Leitfaden zu verstehen, sollten Sie die LLMs Funktionsweise verstehen und über grundlegende Kenntnisse in Bezug auf MCP, Tools und Prompt Engineering verfügen.

Ziele

Um agentische KI-Systeme zu entwickeln, die einsatzbereit sind, müssen Sie gemeinsam Lösungen für Governance, Optimierung und Sicherheit finden, um die Richtlinien Ihres Unternehmens zu unterstützen. Im Folgenden wird erklärt, wie dieser Leitfaden diese Ziele erreicht:

- Steuerung — Ohne zentralisierte Steuerung können Sie keine Prüfungsfragen zu Ihren KI-Workloads beantworten. Dazu gehört auch, welche Agenten wann mit welchen Berechtigungen auf welche Daten zugegriffen haben. Sie können auch keine Versionierung erzwingen. Im Abschnitt zur MCP-Hosting-Strategie dieses Handbuchs wird erklärt, wie Benutzer aufgrund fehlender systematischer Maßnahmen veraltete lokale MCP-Server mit bekannten Sicherheitslücken betreiben könnten.

  Für regulierte Branchen ist Unternehmensführung von entscheidender Bedeutung. Prüfer möchten die Durchsetzung von Richtlinien und die Nachverfolgung der Toolnutzung für alle Agenten von einem einzigen Fenster aus verfolgen können. MCP Governance bietet das.

  Wenn Sie die Empfehlungen in diesem Leitfaden befolgen, können Sie die Aufgabengenauigkeit bei von Experten begutachteten Benchmarks um 28-32% verbessern. Weitere Informationen finden Sie unter MARCO: Multi-Agent-Echtzeit-Chat-Orchestrierung (ACL Anthology-Website). Bei Governance geht es nicht nur um Compliance, sondern auch um die Verbesserung der Leistung Ihres KI-Systems für Agenturen.
- Optimierung — Ihre Teams können dieselben Integrationen mehr als einmal erstellen. Wenn beispielsweise fünf verschiedene Teams ihr eigenes Datenbankabfrageskript für ihre KI-Anwendung schreiben, um mit ihren Datenbanken zu kommunizieren, sind das Fünffache der Entwicklungskosten und fünf Sätze von Buglisten, die verwaltet werden müssen. Mit MCP können Sie es einmal erstellen und dann mit der gesamten Engineering-Community teilen. Die Einsparungen summieren sich, wenn die Anzahl Ihrer Agenten wächst.

  Es gibt auch ein Problem mit den Kosten pro Anfrage, das die meisten Teams zunächst nicht bemerken. Jede Tooldefinition verwendet Kontextfenster-Token. Bei 20 Tools geben Sie zusätzlich zu den Benutzeranfragen 5.000 bis 10.000 Token pro Aufruf allein für Beschreibungen aus. Dies erhöht die Latenz und die Kosten für LLM-Inferenzen und verschlechtert die Genauigkeit, da das Modell Schwierigkeiten hat, das richtige Tool aus der Liste der verfügbaren Tools auszuwählen.

  Agenten, die strukturierte Tool-Wrapper verwenden, sind bei Datenbankaufgaben etwa dreimal genauer als Agenten, die APIs direkt darauf zugreifen (weitere Informationen finden Sie unter Middleware für LLMs: Tools sind entscheidend für Sprachagenten in komplexen Umgebungen).

Es ist wichtig, wie Sie Tools für ein KI-Modell entwerfen und präsentieren. In diesem Leitfaden wird empfohlen, den Tools klare Schemata zu geben, sie auf tatsächliche Workflows statt auf unformatierte Endpunkte zu beschränken und die Informationen im Kontextfenster einzuschränken. Der Abschnitt zur Strategie zur Entwicklung von MCP-Tools in diesem Leitfaden befasst sich eingehend mit diesen Aspekten.

- Sicherheit und Compliance — Stellen Sie sich ein agentisches KI-System vor, das einen Säuberungsschritt halluziniert und versucht, eine Produktionsdatenbank zu löschen. Wenn der Agent die vollständigen Administratoranmeldeinformationen des Benutzers geerbt hat, wird die Löschung möglicherweise durchgeführt. Mit Token-Isolierung und abgespeckten Anmeldeinformationen, die nur Lese- und Erstellungszugriff gewähren, schlägt der Vorgang problemlos fehl.

  Geregelte Arbeitsabläufe verschärfen dies noch weiter. Der Leitfaden enthält Beispiele (Pipelines im Gesundheitswesen, für die eine HIPAA-Validierung und Anonymisierung personenbezogener Daten erforderlich sind, bevor Patientendaten verarbeitet werden). Die Einbettung einer solchen Logik in MCP-Tools bedeutet, dass die Einhaltung der Vorschriften jedes Mal deterministisch erfolgt.

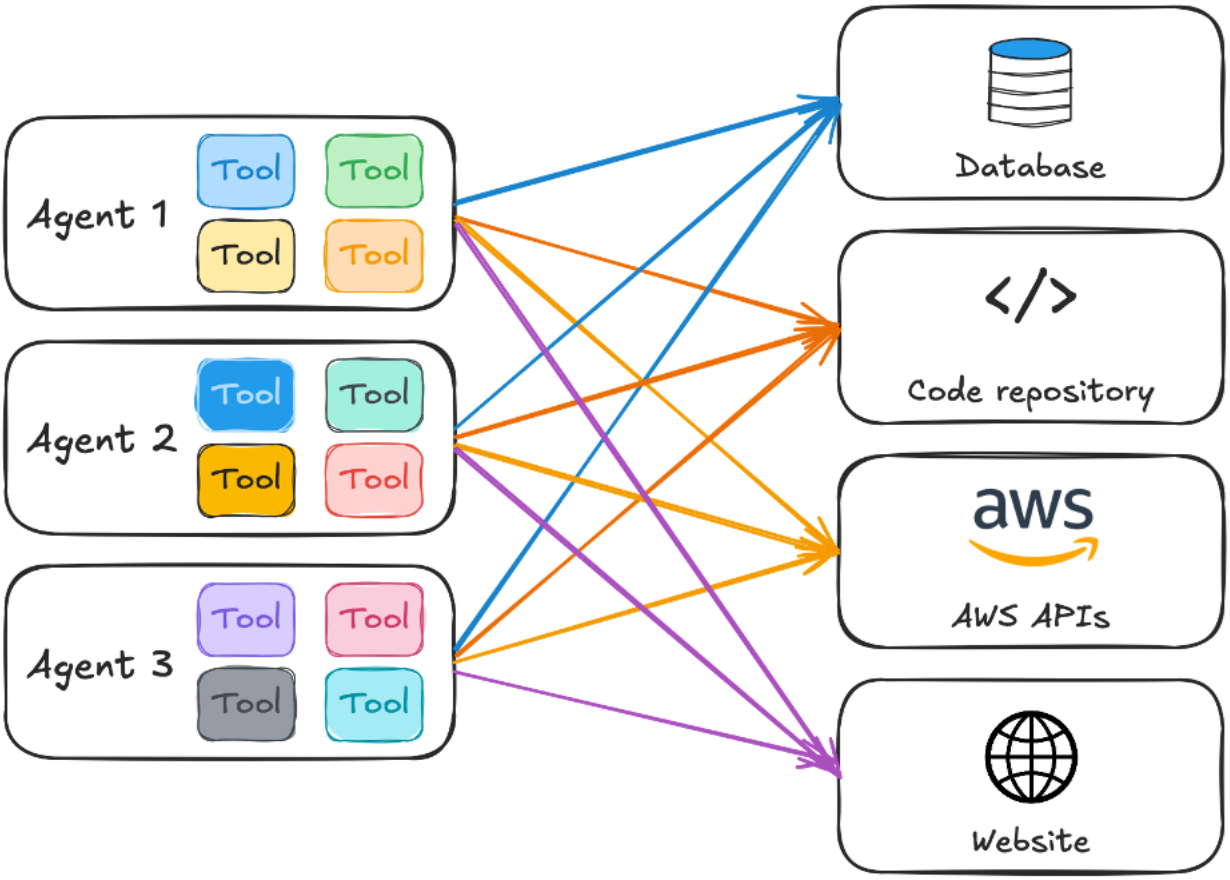
Was ist MCP?

LLMs arbeiten, indem sie anhand ihrer Trainingsdaten eine Antwort auf eine Aufforderung vorhersagen. Das bedeutet, dass das LLM nur Antworten auf Daten und Ereignisse geben kann, die es bereits gesehen hat. Methoden wie Retrieval Augmented Generation (RAG) und Wissensdatenbanken ermöglichen es Ihnen, Kontextdaten einzubeziehen. Wenn Sie jedoch einen LLM fragen würden, wie die Wettervorhersage für morgen aussehen wird oder wie viele Kunden sich in Ihrer Datenbank befinden, würde er wahrscheinlich halluzinieren oder nicht in der Lage sein, eine Antwort zu geben, da diese nicht dem vorab geschulten Wissen des LLM entsprechen. Um diese Art von Fragen beantworten zu können, benötigt ein Agent Zugriff auf externe Funktionen und Daten, die sich APIs außerhalb des ursprünglichen Kontextes des LLM befinden.

Tools verstehen

Wir können dem LLM mithilfe von Tools Zugriff auf zusätzliche Systeme und Kontexte gewähren.Tools sind Funktionen, die dem LLM zur Erreichung eines klaren Ziels übertragen werden. Ein Tool könnte eine API aufrufen, eine Datenbank abfragen, Rechenoperationen ausführen, eine Code-Sandbox bedienen, eine Websuche durchführen und sogar ein anderes KI-System aufrufen oder. agent-as-a-tool Jedes Tool sollte eine Beschreibung enthalten, aus der dem LLM hervorgeht, was das Tool tut, wann es verwendet werden soll und welche Parameter es akzeptiert. Auf diese Weise kann das LLM auf der Grundlage der Benutzereingaben nuancierte Entscheidungen darüber treffen, welches Tool oder welche Kombination von Tools aufgerufen werden soll. Der LLM wird darüber informiert, welche Tools dem Agenten zur Verfügung stehen, sodass er Antworten generieren kann, die den Agenten anweisen, das Tool aufzurufen. Wenn Sie den LLM beispielsweise fragen, wie viele Kunden sich in Ihrer Datenbank befinden, sendet der LLM eine Antwort an den Agenten zurück und fordert ihn auf, das `query_database` Tool mit bestimmten Eingabeparametern auszuführen. Das LLM bestimmt, welches Tool aufgerufen werden soll und welche Eingaben für den Werkzeugabruf erforderlich sind. Der Agent führt dann das Tool aus, das die Eingabe in natürlicher Sprache in einen syntaktisch korrekten Funktionsaufruf umwandelt und die Abfrage ausführt.
Der Agent ruft das Tool oder die Tools auf der Grundlage der Anweisung des LLM auf, und diese Ergebnisse werden an das LLM zurückgegeben. Dabei wird die Fähigkeit des LLM zur Argumentation gegenüber textbasierten Eingaben ausgenutzt und die geeigneten Tools für den Job ausgewählt.

Die folgende Abbildung zeigt, wie jeder Agent sein eigenes Toolset für jedes Ziel verwaltet.



Die Skalierung des Zugriffs auf Tools kann KI-Lösungen für Agenturen vor Herausforderungen stellen:

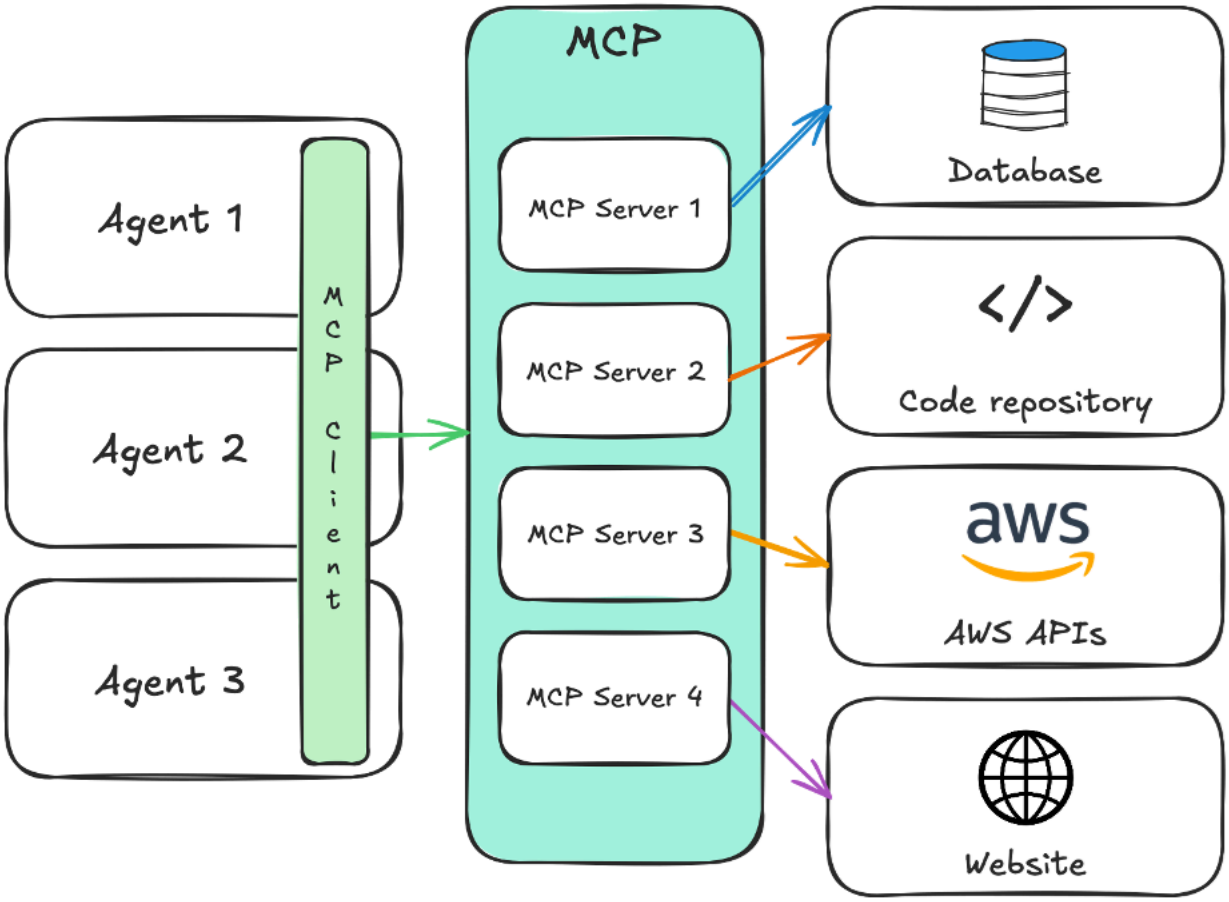
- Wenn jeder Entwickler sein eigenes Tool für dieselben externen Funktionen entwickelt, bedeutet das eine Menge doppelten Aufwand und unstandardisierte Interaktionsmöglichkeiten mit diesen externen Funktionen. Dies führt zu inkonsistenten Implementierungen bei Ihren Agenten. Sie könnten dieses Problem zwar lösen, indem Sie Standardtools in Bibliotheken entwickeln und verteilen, aber es fehlt eine zentrale Steuerung. Dies macht es schwierig, Sicherheitsrichtlinien durchzusetzen, die Nutzung von Tools nachzuverfolgen, die Versionierung teamübergreifend zu verwalten oder die Einhaltung von Unternehmensstandards sicherzustellen. Wenn Sie Tools direkt in den Agenten einbetten, müssen Sie Ihren Agenten außerdem jedes Mal neu bereitstellen, wenn ein neues Tool erstellt oder ein vorhandenes aktualisiert wird.
- Wenn Sie einem LLM Tools zur Verfügung stellen, wird dessen Kontextfenster beansprucht. Das Kontextfenster ist die Anzahl der Tokens (Texteinheiten, die LLMs verarbeitet werden — typischerweise stehen sie für Wörter, Wortteile oder Satzzeichen), die ein Modell gleichzeitig berücksichtigen kann. LLMs haben Beschränkungen für das Kontextfenster. Tools und ihre Dokumentation verwenden dieses endliche Kontextfenster zusammen mit System-

und Benutzeraufforderungen. Wenn sich das Kontextfenster füllt, LLMs kann es aufgrund mehrerer Faktoren zu Leistungseinbußen kommen: Schwierigkeiten beim Identifizieren relevanter Informationen, erhöhte Verarbeitungskomplexität und verringerte Denkfähigkeit. Die Herausforderung wird noch verschärft, wenn Tooldefinitionen, Systemaufforderungen und der Konversationsverlauf um den begrenzten Platz im Kontextfenster konkurrieren, da sie bei jedem LLM-Aufruf bereitgestellt werden.

Daher wirken sich die Anzahl der Tools und deren Dokumentation direkt auf die Leistung des LLM aus, z. B. auf Reaktionszeit und Genauigkeit.

MCP etabliert einen universellen Standard für die Verbindung von Agenten mit externen Funktionen. Es wird allgemein als „USB-C für KI-Anwendungen“ bezeichnet. Anstatt Tools direkt bei Agenten zu registrieren, fungieren MCP-Server als Vermittler für das Hosten von Tools, die über JSON-RPC 2.0 erkannt und aufgerufen werden. Anstatt Ihrem Agenten Dutzende oder Hunderte verschiedener Tools hinzuzufügen und diese im Laufe der Zeit zu verwalten, können Sie mit MCP MCP-Server registrieren, die die Tools kapseln, auf die Ihr Agent zugreifen kann. Dieser Ansatz standardisiert die Art und Weise, wie Tools verpackt, präsentiert und aufgerufen werden. Dies kann dazu beitragen, die Herausforderungen in Bezug auf Umfang und Steuerung zu bewältigen, die mit der Nutzung von Tools in Ihren Agenten verbunden sind. Außerdem werden die Entwicklung und der Betrieb der Agenten von den Tools entkoppelt, die für externe Funktionen verwendet werden.

Die folgende Abbildung zeigt Agenten, die MCP für den Zugriff auf externe Ressourcen verwenden.



Der MCP-Standard löst jedoch nicht alle Skalierungs- und Governance-Herausforderungen. Die Implementierung von MCP-Servern muss mit effektiven Strategien für Tooldesign, Hosting und Unternehmensführung kombiniert werden. Dieser Leitfaden enthält bewährte Methoden für jede Strategie, um Sie bei der Entwicklung und Verwendung von MCP als Teil Ihrer agentischen KI-Lösungen zu unterstützen.

Wann sollte MCP verwendet werden

MCP bietet eine strategische Infrastruktur für die Skalierung Ihrer agentischen KI-Initiativen. Durch die Zentralisierung von Toolmanagement und -steuerung reduzieren MCP-Server die Gesamtkosten für die Erstellung und Wartung benutzerdefinierter Integrationen für mehrere Agenten. Dies führt zu steigenden Renditen, wenn Ihr Agenten-Ökosystem wächst.

MCP wird wahrscheinlich Teil Ihrer Strategie, wenn:

- Sie benötigen eine zentrale Steuerung dafür, wie Agenten auf Unternehmenssysteme und -dienste wie Datenbanken APIs, interne Tools und Integrationen von Drittanbietern zugreifen.

- Entwickler verbringen zu viel Zeit damit, benutzerdefinierte Integrationen zu schreiben, die nicht in allen Implementierungen konsistent sind.
- Sie verfügen über doppelte Tools, die allgemeine Funktionen erfüllen könnten.
- Sie möchten Ihre firmeneigenen Tools oder Daten externen Verbrauchern oder Agentensystemen von Drittanbietern über standardisierte, geregelte MCP-Schnittstellen anbieten und so neue Einnahmequellen erschließen und gleichzeitig Sicherheit und Kontrolle gewährleisten.

Nachdem Sie entschieden haben, dass MCP-Server Teil Ihrer Strategie sein werden, sollten Sie prüfen, ob die vorhandenen Open-Source-MCP-Serverimplementierungen Ihren Anforderungen entsprechen, ob sie erweitert werden müssen oder ob Sie benutzerdefinierte Server erstellen müssen. Viele vorgefertigte MCP-Serverimplementierungen sind in öffentlichen Repositorys verfügbar und decken allgemeine Funktionen wie Dateisystemzugriff, Web-Browsing, Code-Sandboxes, Datenbankzugriff und API-Integrationen ab.

In vielen Fällen sind bereits vorhandene MCP-Server ausreichend. AWS Stellt beispielsweise einen verwalteten Remote-MCP-Server bereit AWS MCP -Server, der KI-Assistenten und -Agenten sicheren, authentifizierten Zugriff auf Interaktionen in natürlicher Sprache ermöglicht. AWS-Services Sie können den verwenden, AWS MCP -Server um komplexe, mehrstufige AWS Aufgaben auszuführen, indem Sie Echtzeitzugriff auf AWS Dokumentation, syntaktisch korrekte API-Aufrufe und vorgefertigte Workflows namens Agent kombinieren, die bewährten Methoden folgen. SOPs AWS AWS testet sie kontinuierlich AWS MCP -Server , um sicherzustellen, dass Kundenagenten sie erfolgreich einsetzen können.

Sie sollten diese vorhandenen MCP-Server zusammen mit Ihren Agenten testen, um festzustellen, ob sie Ihren Anwendungsfällen entsprechen. Wenn ein Agent Workflows nicht abschließt, falsche oder suboptimale Antworten generiert, komplexe mehrstufige Prozesse nicht bewältigt oder wichtige domänenspezifische Best Practices oder Sicherheitsüberlegungen übersieht, sollten Sie Verbesserungen in verschiedenen Dimensionen in Betracht ziehen.

Wenn vorhandene MCP-Server Ihre Anforderungen nicht vollständig erfüllen und sie Schwierigkeiten haben, die vorhandenen Tools korrekt zu verwenden oder präzise Antworten zu liefern, sollten Sie diese Verbesserungsansätze in Betracht ziehen, bevor Sie benutzerdefinierte Server erstellen:

- Den Agentenkontext erweitern — Wenn Ihr Agent Schwierigkeiten hat, die Tools auf einem vorhandenen MCP-Server korrekt oder effizient zu verwenden, sollten Sie erwägen, diese Tooldefinitionen durch zusätzliche Dokumentation oder Beispiele zu ergänzen. Dies trägt dazu bei, dem LLM zusätzlichen Kontext zu bieten.

- Zusätzliche Tools hinzufügen — Erweitern Sie bestehende MCP-Server um Tools, die auf zusätzliche Unternehmensdaten oder den Kontext zugreifen, den Agenten benötigen, um Workflows erfolgreich abzuschließen.
- Verbessern Sie die zugrunde liegenden Daten APIs — Vereinfachen Sie Ihren Service, APIs um ihn LLM-freundlicher zu gestalten, indem Sie die Komplexität der Parameter reduzieren, klarere Fehlermeldungen bereitstellen und sinnvolle Standardwerte bereitstellen, die Agenten verwenden können.

Die Verwendung vorhandener MCP-Serverimplementierungen beschleunigt zwar die Entwicklung allgemeiner Funktionen, aber der Aufbau kundenspezifischer MCP-Server ist unverzichtbar, wenn Ihr Anwendungsfall spezielle Funktionen erfordert. Benutzerdefinierte MCP-Server helfen Ihnen dabei, Fachwissen zu bündeln, Unternehmensstandards durchzusetzen, die Zuverlässigkeit der Agenten für komplexe Workflows zu verbessern und die Einhaltung von Sicherheitsanforderungen zu unterstützen. Erwägen Sie die Erstellung eines benutzerdefinierten MCP-Servers in den folgenden Situationen:

- Domänenspezifische Workflows — Mehrstufige Workflows, die Fachwissen erfordern, sollten in benutzerdefinierten MCP-Tools zusammengefasst werden, wenn das erforderliche Wissen nicht in der API-Dokumentation erfasst ist. Anstatt beispielsweise Agenten die Orchestrierung komplexer Daten-Pipelines im Gesundheitswesen zu überlassen, die die Einhaltung des Health Insurance Portability and Accountability Act (HIPAA) validieren, personenbezogene Daten anonymisieren und in das HL7 FHIR-Format umwandeln müssen, sollten Sie ein `process_patient_data` Tool bereitstellen, das die Fachkenntnisse direkt einbettet. Dadurch entfällt die Abhängigkeit vom LLM für die korrekte Orchestrierung und Ausführung der Workflow-Schritte, wodurch Konsistenz und Compliance verbessert werden.
- Abstraktionen auf dem Goldenen Pfad — Agenten haben möglicherweise Schwierigkeiten, optimale Ansätze zu implementieren, weil ihnen der organisatorische Kontext fehlt und sie sich eher an grundlegende Muster als an den bewährten Methoden der Organisation orientieren. In diesen Szenarien können Sie verbindliche Standards in Bezug auf Kosten, Leistung oder Sicherheit durchsetzen, indem Sie diese goldenen Pfade in benutzerdefinierten MCP-Tools zusammenfassen. Anstatt Agenten beispielsweise die Infrastruktur mit Standardeinstellungen bereitstellen zu lassen, die möglicherweise unsicher oder ineffizient sind, sollten Sie ein `deploy_secure_infrastructure` Tool bereitstellen, das die Standards Ihres Unternehmens direkt integriert.
- Komplexe Orchestrierung mehrerer Dienste — Anstatt den Agenten dazu zu bringen, komplexe Workflows zu orchestrieren, indem er versucht, die richtige Reihenfolge und den richtigen Satz

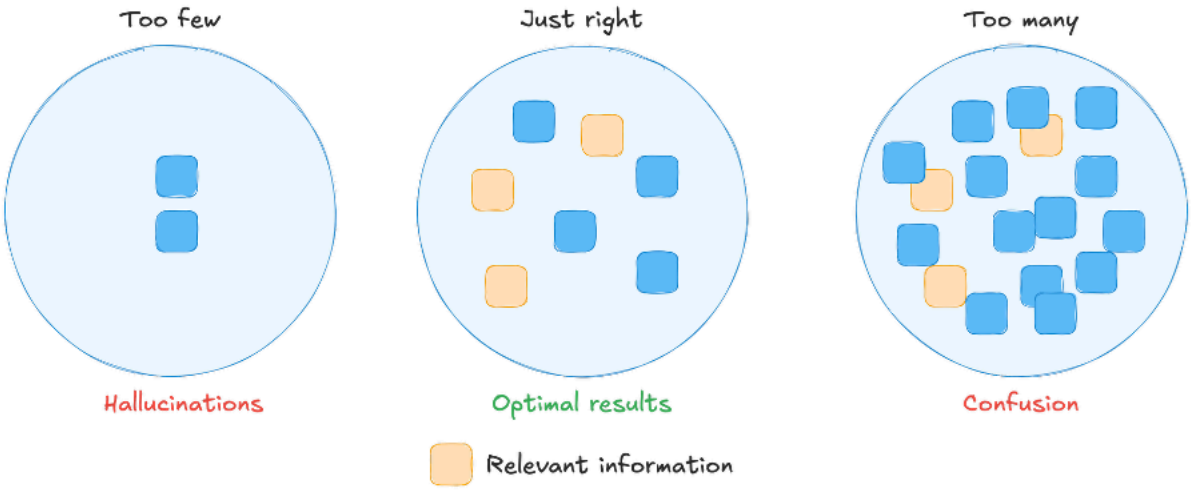
von Diensten abzuleiten, die in jedem Schritt verwendet werden sollen, können Sie diese Logik deterministisch in einem MCP-Tool aufbauen. Möglicherweise möchten Sie auch Fachwissen über optimale Serviceintegrationsmuster bereitstellen, von denen der Agent möglicherweise nichts weiß. Dies kann auch die Genauigkeit und Effizienz Ihrer Agenten verbessern.

- Servicespezifische Best Practices — Dies ist bei sicherheitsorientierten Tools üblich, die Agenten bei der Implementierung von Verschlüsselungsrichtlinien, Zugriffskontrollen und Compliance-Mustern unterstützen, die für den Service spezifisch sind, auf den über das Agententool zugegriffen wird. Wenn es dienstspezifische bewährte Methoden für den Betrieb gibt, die nicht offensichtlich sind, können Sie mithilfe eines MCP-Servers sicherstellen, dass diese implementiert werden und nicht dem Agenten überlassen werden, sich darüber Gedanken zu machen.

Strategie für die Entwicklung von MCP-Tools

Die Hauptaufgabe des MCP-Clients und -Servers besteht darin, Tools zu finden und dem LLM vorzustellen, damit es sie zur Verbesserung seiner Antworten verwenden kann. Dies macht das MCP-Tooldesign zu einer der wichtigsten Strategien für die Entwicklung effektiver MCP-Lösungen. Aus Sicht des Modells handelt es sich bei Tools um Funktionen, die sie bei Bedarf aufrufen können, um genauere und vollständigere Antworten zu liefern. Die Funktionsschnittstelle abstrahiert die zugrunde liegende Implementierung eines Tools, die von einem Wrapper für einen einzelnen API-Aufruf bis hin zu komplexer Workflow-Logik reichen kann.

Sie müssen jedoch ein Gleichgewicht mit der Menge der Tools finden, die dem LLM zur Verfügung gestellt werden. Wenn zu wenige Tools zur Verfügung stehen, ist das LLM möglicherweise nicht in der Lage, den richtigen Kontext und die richtigen Informationen zu sammeln, sodass es anhand der im Modell verfügbaren Informationen die beste Schätzung anstellt. Wenn zu viele Tools zur Verfügung stehen, kann das LLM verwirrt sein, was die richtige Auswahl und Reihenfolge der Werkzeuge angeht, was zu Halluzinationen führen kann. Ihr Ziel ist es, die Anzahl der Tools genau richtig einzustellen. Die folgende Abbildung zeigt die Herausforderungen, die mit zu wenigen und zu vielen Tools verbunden sind.



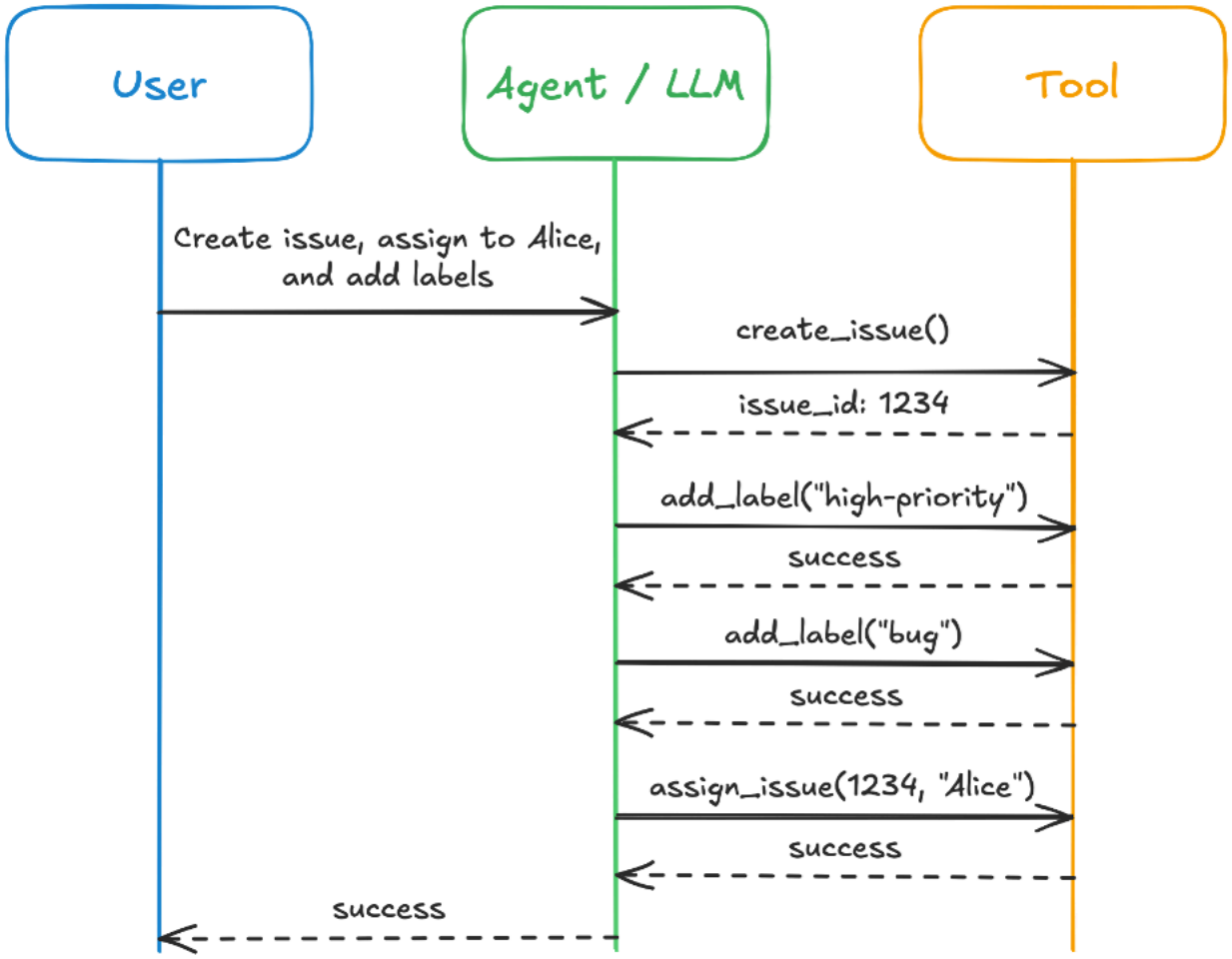
Die Lösung setzt voraus, dass Sie wissen, wie viele Tools bereitgestellt werden müssen und wie die einzelnen Tools auszulegen sind. Die Granularität Ihrer Tools, unabhängig davon, ob sie einzelnen API-Aufrufen oder kompletten Workflows zugeordnet sind, wirkt sich direkt auf die Gesamtzahl der Tools aus, die Agenten benötigen, und darauf, wie effektiv sie diese verwenden können. In diesem Abschnitt finden Sie bewährte Methoden für die Festlegung des Anwendungsbereichs von MCP-Tools, die Erstellung von Tooldefinitionen, die Suche nach Tools und deren Organisation.

Umfang der Tools

Es gibt zwei Ansätze für die Entwicklung von Tools: detaillierte und grobkörnige.

Granular

Bei einem detaillierten Ansatz würden Sie ein Tool pro API, Aktion oder Abfrage erstellen. Sie könnten beispielsweise,`create_issue`, `get_issue` `add_label`assign_issue, und `close_issue` Tools für Ihr Git-Repository erstellen. Dies würde es dem LLM ermöglichen, detaillierte Aufrufe an jede API zu tätigen und jede API nach Bedarf zu orchestrieren. Stellen Sie sich die folgende Aufforderung vor: „Erstellen Sie ein Problem für den Produktservice mit dem Namen 'Abfrage gibt nur Teilergebnisse', kennzeichnen Sie es als Fehler mit hoher Priorität und weisen Sie es Alice zu.“ Die folgende Abbildung zeigt, wie ein tool-per-API Ansatz auf diese Aufforderung reagieren würde.

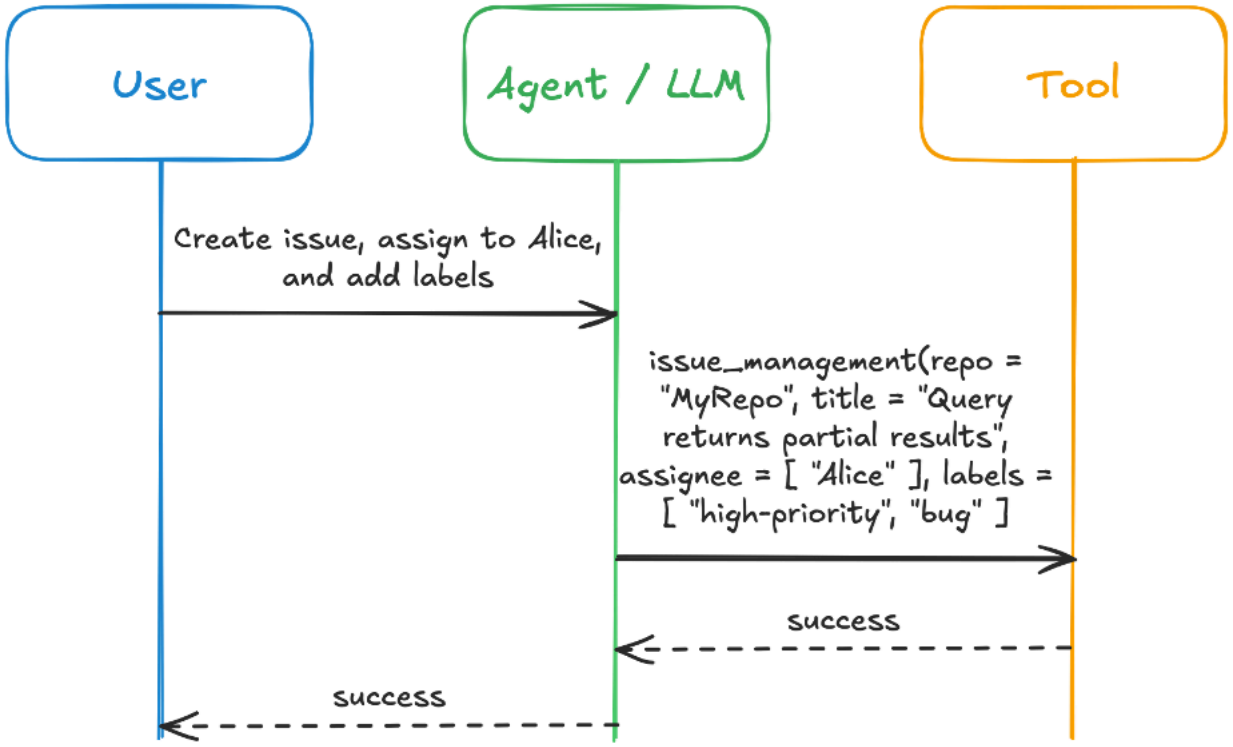


Bei diesem Ansatz werden dem LLM bei jedem Aufruf die Systemaufforderung und jede registrierte Tooldefinition zur Verfügung gestellt. Dadurch wird zusätzlicher Kontext verbraucht und es kommt zu

einer Latenzeinbuße, da jeder Tool-Aufruf einen individuellen Aufruf an das LLM darstellt. Dies erhöht auch die Komplexität der Fehlerbehandlung innerhalb des Workflows.

Grobkörnig

Ein grobkörniger oder workflow-orientierter Ansatz wären Tools, die auf Workflows ausgerichtet sind. Das Tool konzentriert sich auf die Benutzerabsicht und nicht auf die API-Struktur. end-to-end Anstelle von einem haben Sie ein Tool tool-per-API, das deterministisch viele aufruft. APIs Mit dem vorherigen Git-Repository-Beispiel könnten Sie ein `create_and_setup_issue` Tool erstellen, das einmal vom Agenten aufgerufen wird. Die Tool-Implementierung erstellt das Problem, fügt Labels hinzu und weist es einem Benutzer zu, basierend auf den Parametern, die dem Tool zur Verfügung gestellt wurden. Die folgende Abbildung zeigt, wie ein grober Ansatz dieselbe Aufforderung verarbeiten würde.



Dieser Ansatz zeigt, dass die gesamte Komplexität vor der LLM-Ebene verborgen bleibt. Wenn die Orchestrierungslogik in die Toolimplementierung eingebettet ist, werden alle sequentiellen Schritte, die Protokollierung, die Wiederholungslogik, die Schutzschalter und die Ratenbegrenzung deterministisch im Tool ausgeführt. Der workflow-gesteuerte Ansatz erleichtert es dem LLM, das richtige Tool mit den richtigen Parametern aufzurufen. Es ist wichtig zu beachten, dass einige APIs möglicherweise bereits Workflow-Intent bereitstellen, z. B. die Amazon EC2 `RunInstances` EC2-API. In diesen Fällen bietet a tool-per-API möglicherweise das von Ihnen gewünschte workflow-orientierte Design.

Werkzeuge können jedoch auch zu grobkörnig werden. Wenn Ihr einzelnes Workflow-Tool versucht, zu viele Dinge zu tun und viele mögliche Parameter hat, kann es für den LLM schwierig sein, zu überlegen, wie das Tool richtig verwendet werden soll. Es kann auch zu Herausforderungen bei der Parameterauswahl und Fehlerbehandlung führen. Daher muss bei der Entwicklung von Tools ein Gleichgewicht gefunden werden, das den Benutzerabsichten entspricht und zu wenig oder zu viel Funktionalität in einem einzigen Tool vermeidet. Wir empfehlen Ihnen, Tools so zu entwickeln, dass sie komplette Benutzerworkflows berücksichtigen und Vorgänge, die häufig zusammen auftreten (z. B. drei oder mehr API-Aufrufe), bündeln. Wir empfehlen außerdem, Tools zu zerlegen, die acht oder mehr Parameter überschreiten oder mehrere unterschiedliche Benutzerabsichten berücksichtigen. Testen Sie mit echten Eingabeaufforderungen, um sicherzustellen, dass die Agenten jedes Tool korrekt verwenden können.

Wenn Sie über komplexe und dynamische Workflows verfügen, die sich nicht einfach als deterministisches Tool zusammenfassen lassen, könnten Sie die Verwendung des Musters in Betracht ziehen. agent-as-tool Anstatt dass Ihr primärer Agent versucht, komplexe Aufgaben in einem Workflow zu orchestrieren, kann ein spezialisierter Agent als Tool fungieren. Mit diesen Tools können erweiterte Entscheidungsprozesse und Verzweigungen implementiert werden. Außerdem können sie Fehler und Wiederholungslogik behandeln, die in deterministischem Code nicht einfach zu handhaben ist. Dies ähnelt dem Agent2Agent (A2A) -Protokoll, unterscheidet sich jedoch von diesem. Das A2A-Protokoll ist komplementär und bietet Interoperabilität und Zusammenarbeit zwischen Agenten in jedem agentischen Framework.

Wir empfehlen Ihnen, mit Ihrer Workflow-Analyse zu beginnen, indem Sie Ihre gängigsten Benutzerworkflows abbilden, um die Kernfunktionen zu ermitteln, die jeder Agent benötigt. Auf diese Weise wird Ihr minimales praktikables Toolset festgelegt. Aufgrund unserer Erfahrung mit der Entwicklung von MCP-Servern in großem Maßstab empfehlen wir die folgenden Vorgehensweisen. Wenn diese Praktiken miteinander in Konflikt geraten, sollten Sie Benutzerabsicht und Arbeitsablauf priorisieren.

Bewährte Methoden für die Festlegung des Umfangs von MCP-Tools

- Denken Sie an Anwenderberichte und bündeln Sie gängige Abläufe — Tools sollten sich direkt den gesamten Benutzerinteraktionen zuordnen lassen, anstatt die Orchestrierung mehrerer Operationen zu erfordern. Wenn Workflows häufig drei oder mehr separate Aufrufe erfordern, kombinieren Sie sie zu einem einzigen Tool. Dies reduziert die kognitive Belastung des LLM, minimiert die Anzahl der Tool-Aufrufe, reduziert den Kontextverbrauch und die Latenz, die für die Ausführung von Aufgaben erforderlich sind, und verbessert die Genauigkeit und Latenz.

- Beschränken Sie die Parameter auf acht oder weniger — Wenn ein Werkzeug mehr als acht Parameter hat, teilen Sie es in mehrere Tools auf. LLMs Probleme mit der Parameterauswahl haben, wenn die Komplexität zunimmt.

> **Note**
>
> Wenn für Bündelungsvorgänge mehr als acht Parameter erforderlich sind, sollten Sie der Bündelung Vorrang vor der Anzahl der Parameter einräumen, da die Vereinfachung des Workflows wertvoller ist als strenge Parametergrenzwerte.

- Getrennte Lese- und Schreibvorgänge — Stellen Sie verschiedene Tools zum Lesen und Ändern von Daten bereit. Diese Trennung macht deutlich, wann Agenten potenziell zerstörerische Operationen ausführen, ermöglicht unterschiedliche Autorisierungsrichtlinien und reduziert das Risiko unbeabsichtigter Änderungen bei der Datenerfassung.
- Stellen Sie sinnvolle Standardeinstellungen bereit — Entwerfen Sie die Tools so, dass das LLM nur die Parameter angeben muss, die für die jeweilige Anfrage spezifisch sind. Standardeinstellungen reduzieren die Komplexität der Parameter und verbessern die Genauigkeit der Werkzeugauswahl, indem sie die Informationen minimieren, über die das LLM nachdenken muss.
- Bevorzugen Sie die deterministische Ausführung — Machen Sie die Ausführung und Ausgabe des Werkzeugs nach Möglichkeit deterministisch. Deterministische Tools sind zuverlässiger und einfacher zu testen. Für komplexe Workflows, die intelligente Orchestrierung, Verzweigungslogik oder erweiterte Fehlerbehandlung erfordern und die sich in deterministischem Code nicht einfach verwalten lassen, sollten Sie die Verwendung spezialisierter Agenten als Tools in Betracht ziehen. Verwenden Sie dieses Muster jedoch selektiv, da es die Komplexität erhöht.

Werkzeugdefinitionen

Wenn ein LLM eine Anfrage erhält, die es nicht direkt bearbeiten kann, überprüft es die verfügbaren Tools, um die Anfrage zu bearbeiten. Das LLM wählt Tools auf der Grundlage seines semantischen Verständnisses der Namen und Beschreibungen der bereitgestellten Tools sowie aller in der Aufforderung enthaltenen Anweisungen aus. Es erstellt dann Eingaben auf der Grundlage des definierten Eingabeschemas und erwartet eine Ausgabe auf der Grundlage des Ausgabeschemas. Daher ist die Erstellung beschreibender Werkzeugdefinitionen und validierter Eingabe- und Ausgabeschemas von entscheidender Bedeutung, um dem LLM bei der effektiven Auswahl von Tools zu helfen. Im Allgemeinen gibt es zwei Ansätze für die Erstellung dieser Dokumentation: den Ansatz der Toolspezifikation und den Docstring-Ansatz.

Ansatz zur Werkzeugspezifikation

Der empfohlene Ansatz besteht darin, sich bei der Definition des Tools direkt an die MCP-Werkzeugspezifikation zu halten. Das folgende Beispiel wird mit dem Strands Agent Tool Decorator gezeigt:

```
@tool(
  name = "search_website",
  description = "This tool searches the provided website for semantic matches to the
 query provided",
  inputSchema = {
    "json": {
      "type": "object",
      "properties": {
        "url": {
          "type": "string",
          "description": "The url of the website to load and search."
        },
        "query": {
          "type": "string",
          "description": "The content you want to try and match in the website."
      }
    },
    "required": ["url", "query"]
  },
  outputSchema = {
    "json": {
      "type": "object",
      "properties": {
      "results": {
          "type": "array",
          "items": {
            "type": "string"
          }
        }
      }
    }
  }
)
def search_website:
  …
```

Durch die Verwendung von Standardfeldern wie, `name`, `description`, und wird `outputSchema` sichergestellt`inputSchema`, dass jedes Tool über eine konsistente Dokumentation verfügt, die sowohl der LLM als auch die Menschen verstehen können. Jedes Tool sollte diese Felder mindestens definieren und optional einen Titel und Anmerkungen bereitstellen, bei denen es sich um optionale Hinweise zum Verhalten des Tools handelt. Verwenden Sie nach Möglichkeit Enums für Parameterwerte, um es dem LLM zu erleichtern, die richtigen Optionen auszuwählen. Aufzählungen eignen sich am besten für endliche Mengen wie Status- oder Prioritätswerte, eignen sich jedoch nicht für Freiformtext, dynamische Werte, willkürliche Zahlen oder Ressourcenbezeichner. Geben Sie in diesen Fällen stattdessen klare Beschreibungen und Beispiele an. Geben Sie nach Möglichkeit auch einen Standardwert an, damit der LLM nicht erraten muss, was die richtige Option ist. Denken Sie daran, dass Tooldefinitionen bei jedem Aufruf in der LLM-Eingabeaufforderung enthalten sind und neben den Systemanweisungen und dem Konversationsverlauf auch Platz im Kontextfenster beanspruchen.

Docstring-Ansatz

Ein anderer Ansatz, wenn Sie Ihre Werkzeuge in Python schreiben, besteht darin, Docstrings zu verwenden, um die Beschreibung, Verwendung und Ausgabe des Werkzeugs bereitzustellen. Im Folgenden finden Sie ein Beispiel für diesen Ansatz:

```
def search_website(url: str, query: str) -> list:

    """
    This tool loads the specified website and then attempts to find content that
 matches the provided query through semantic search. It provides back a list of strings
 that are the sentences that match the query.
    Args:
      url: the website url to load
      query: the content you want to semantically match in the website
    """
```

Docstrings erzwingen weder ein Schema noch ein standardisiertes Format. Die Verwendung dieses Ansatzes kann zu inkonsistenten Ergebnissen führen, je nachdem, wie Toolentwickler die einzelnen Tools dokumentieren. Wenn Sie diesen Ansatz verfolgen, ist es wichtig, einen unternehmensweiten Standard zu definieren und durchzusetzen.

Bewährte Methoden für Definitionen von MCP-Tools

- Halten Sie sich an die MCP-Werkzeugspezifikation — geben Sie `outputSchema` Felder `name`,`description`,`inputSchema`, und für jedes Werkzeug an. Verwenden Sie für Python-Implementierungen Pydantic-Modelle, um Inline-Dokumentation durch Feldbeschreibungen, automatische Typvalidierung und eingeschränkte Werte durch Enums bereitzustellen. Dadurch dokumentieren sich Schemas selbst und das LLM-Verständnis gültiger Parameteroptionen verbessert sich.
- Schreiben Sie Beschreibungen als Eingabeaufforderungen — Toolbeschreibungen sind Anweisungen, die die LLM-Entscheidungsfindung erleichtern. Geben Sie die wesentlichen Bestandteile des Tools an (was das Tool tut), wann es verwendet werden soll (Muster oder Szenarien der Benutzerabsichten), den Kontext der Ausgabe (wofür die Ausgabe verwendet wird), Parameter und Fehlerbedingungen.
- Geben Sie konkrete Beispiele an — Die Einbeziehung von Workflow-Beispielen mit tatsächlichen Werten ist die effektivste Methode, um LLMs Hinweise zur korrekten Verwendung des Tools zu geben.
- Dokumentieren Sie Abhängigkeiten explizit — Geben Sie Voraussetzungen, nummerierte Sequenzen, Statusänderungen und Folgemaßnahmen an.

Erkennung von Tools

Es gibt drei Ansätze für die Erkennung und Registrierung von Tools in Ihrem Agenten auf MCP-Servern: statische Definition, dynamische Erkennung und Suchfunktion.

Statische Definition

Zunächst können Sie die verfügbaren Tools direkt im Agentencode statisch definieren. Bei diesem Ansatz definieren Sie für jedes vom MCP-Server bereitgestellte Tool, auf das ein MCP-Client zugreift, ein Remote-Tool (ein clientseitiges Referenzobjekt in einem Framework wie dem Strands Agent SDK). Das folgende Beispiel verwendet streambaren HTTP-Transport:

```
from mcp.client.streamable_http import streamablehttp_client
from strands import Agent
from strands.tools.mcp import MCPClient

streamable_http_mcp_client = MCPClient(
    lambda: streamablehttp_client("https://mcp1:8000/mcp")
```

```
)

reverse_text = RemoteTool(
    name="reverseText",
    client=streamable_http_mcp_client
)

agent = Agent(tools=[reverse_text])
```

Durch die Registrierung einzelner Tools können Sie bei der Auswahl der Tools, die Sie dem LLM zur Verfügung stellen, sehr selektiv vorgehen, wodurch der Umfang des verwendeten Kontextfensters minimiert wird. Der Nachteil besteht darin, dass die Namen der verfügbaren Tools bekannt sein müssen. Außerdem kann es instabil sein, wenn sich die verfügbaren Tools auf dem MCP-Server ändern.

Dynamische Erkennung

Der nächste Ansatz besteht darin, die dynamische Erkennung zu verwenden und alle verfügbaren Tools beim Agenten zu registrieren. Bei diesem Ansatz wird der Kontext linear verbraucht, wenn dem MCP-Server mehr Tools hinzugefügt werden. Im Folgenden finden Sie ein Beispiel für diesen Ansatz:

```
from mcp.client.streamable_http import streamablehttp_client
from strands import Agent
from strands.tools.mcp import MCPClient

streamable_http_mcp_client = MCPClient(
    lambda: streamablehttp_client("https://mcp1:8000/mcp")
)

with streamable_http_mcp_client:
    tools = streamable_http_mcp_client.list_tools_sync()
    agent = Agent(tools=tools)
```

Stellen Sie sich ein Szenario vor, in dem eine typische Werkzeugdefinition etwa 250 bis 500 Token (einschließlich Name, Beschreibung und Schema) verbraucht. Die Registrierung von 20 Tools würde 5.000 bis 10.000 Token Ihres Kontextfensters verbrauchen. Wenn Sie über eine kleine Anzahl von MCP-Servern verfügen und die Kontrolle über die Anzahl der Tools haben, ist diese Option am einfachsten zu implementieren. Wenn jedoch erwartet wird, dass die Liste der Tools wächst, kann dies zu unbemerkten Problemen mit der Kontextverwaltung bei Ihren Agenten führen. Eine alternative Variante dieses Ansatzes besteht darin, beim Aufrufen einen Toolfilterparameter

zu verwenden`list_tools`, wie ihn das Strands Agents SDK bietet, um die Anzahl der Tools zu reduzieren, die beim Agenten registriert sind.

Suchfunktion

Die dritte Möglichkeit besteht darin, eine Suchfunktion zu verwenden, um während der Laufzeit relevante Tools zu finden. Sie listen alle verfügbaren Tools auf Ihrem MCP-Server auf und führen dann basierend auf der Benutzeraufforderung eine semantische Suche über diese Tools durch. Anschließend werden die resultierenden Tools bei Ihrem Agenten registriert. Amazon Bedrock AgentCore Gateway bietet eine native semantische Suchfunktion, die die Implementierung dieser Art von Lösung erleichtern kann.

Bewährte Methoden für die Entdeckung von MCP-Tools

- Beibehaltung des Kontextfensters — Wählen Sie einen Ansatz zur Erkennung und Registrierung von Tools, bei dem Ihr Kontextfenster so weit wie möglich erhalten bleibt.
- Nutzen Sie Funktionen zum Filtern von Tools oder zur semantischen Suche — Stellen Sie dem LLM dynamisch eine Reihe von Tools zur Auswahl zur Verfügung, wodurch die Genauigkeit und Effektivität bei der Auswahl des richtigen Tools verbessert wird. Die Toolfilterung kann anhand von Werkzeugnamen (exakte Übereinstimmung oder Muster), Werkzeugbeschreibungen (semantischer Abgleich) oder Domänen- oder Kategorie-Tags erfolgen. Die semantische Suche ist besonders effektiv, wenn es darum geht, die Absicht der Benutzer mit den Werkzeugbeschreibungen abzugleichen. Beide Ansätze reduzieren die Nutzung von Kontextfenstern.

Organisation der Tools

Die richtigen Tools zu finden und sicherzustellen, dass das LLM sie effektiv einsetzen kann, ist einer der wichtigsten Aspekte einer effektiven Toolentwicklung. Wenn Sie mit der Entwicklung von MCP-Servern beginnen, benötigen Sie eine Strategie, die Folgendes bestimmt:

- Wie viele Tools gehören zu einem MCP-Server
- Welche Tools sollten nicht auf demselben MCP-Server installiert werden
- Wie benennt man Tools, um sie durchsuchbar zu machen und Namenskollisionen zu vermeiden (verschiedene Tools mit demselben Namen)
- Wie dokumentiert man die Tools und den MCP-Server, damit sie vom LLM einfach verwendet werden können

Die Namespace-Organisation ist ein Entwurfsmuster, das Kollisionen von Werkzeugnamen verhindert, zusammengehörige Funktionen gruppiert und eine effiziente Werkzeugidentifikation ermöglicht. LLMs Das Muster legt eine strukturierte Kategorisierung fest, die eher einer unstrukturierten Akkumulation entspricht, als einer unstrukturierten Akkumulation. Wir empfehlen das domain-noun-verbMuster für die Benennung von Tools. Zum Beispiel, `github_issue_creategithub_issue_list`,`github_issue_update`,`github_pullrequest_creat` oder`github_pullrequest_merge`. Der Vorteil dieses Musters wird deutlich, wenn man das alphabetische Sortierverhalten untersucht. Wenn Tools alphabetisch aufgelistet sind, gruppieren sich alle problembezogenen Operationen (`create`,,`update`)`list`, gefolgt von Pull-Request-Vorgängen (`create`,,). `list` `merge` Das Substantiv (Ressourcentyp) dient als organisatorische Grenze. Diese Struktur erleichtert sowohl das Scannen von LLM-Tools als auch die Navigation in der menschlichen Dokumentation, da verwandte Funktionen auf natürliche Weise gruppiert werden.

Der MCP-Server sollte auf Domänenebene begrenzt sein, kann aber aufgrund der Aufgabentrennung für die von ihm bereitgestellten Funktionen unterteilt werden. Beispielsweise könnten Sie separate MCP-Server für Schreib- und Lesevorgänge in einer Datenbank haben. Um diese Trennung durchzusetzen, empfiehlt es sich, Leitplanken auf Agentenebene zu implementieren, die je nach Benutzerabsicht und -berechtigungen einschränken, auf welche MCP-Server zugegriffen werden kann. Dies kann durch eine Kombination der folgenden Maßnahmen erreicht werden:

- Bedingtes Laden des Servers — Der MCP-Server mit Schreibschutz wird nur geladen, wenn der Agent Lesevorgänge in der Benutzereingabe erkennt.
- Rechtebasierte Filterung — Verwenden Sie die Benutzerautorisierung, um nur den entsprechenden MCP-Servern Zugriff zu gewähren.

Schließlich sollten Sie eine Obergrenze für die Anzahl der von einem MCP-Server bereitgestellten Tools festlegen. Machen Sie keine Annahmen darüber, wie Agenten Ihren MCP-Server verwenden werden. Sie listen möglicherweise naiv alle verfügbaren Tools auf und stellen sie alle dem LLM zur Verfügung. Wenn Sie mehr als 50 Tools auf einem einzelnen Server haben, sollten Sie erwägen, ihn auf mehrere Server aufzuteilen.

Bewährte Methoden für die Organisation von MPC-Tools

- Verwenden Sie den domain-noun-verb Benennungsstandard für Tools — Implementieren Sie Strategien, um Namenskollisionen sowohl auf MCP-Servern als auch auf Agenten zu verhindern.
- Obergrenze festlegen — Beschränken Sie die Anzahl der Tools auf einem einzelnen MCP-Server.

- MCP-Server aufteilen — Verwenden Sie die Aufgabentrennung, um MCP-Server in logische Gruppen zu unterteilen.

MCP-Hosting-Strategie

Durch die Zusammenfassung der verfügbaren Tools auf MCP-Servern wird Ihre Agentenentwicklung von den verfügbaren Tools entkoppelt. Dies führt zu den Herausforderungen, die sich daraus ergeben, wo Sie Ihren MCP-Server hosten und wie die Tools auf diesen Servern organisiert sind.

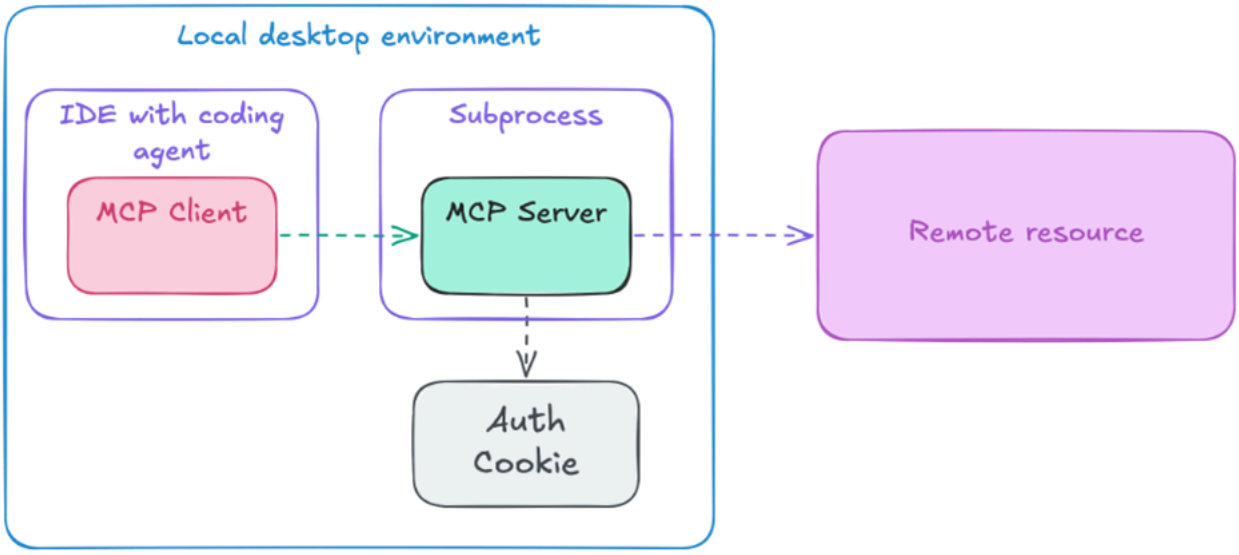
Hosting-Ansätze

Es gibt drei Möglichkeiten, Ihre MCP-Server zu hosten: Sie können sie lokal auf einem Endbenutzercomputer ausführen, sie remote hosten oder sie über ein MCP-Gateway hosten. Jede Option hat Vor- und Nachteile.

Lokales Hosting

Beim lokalen Hosting wird der MCP-Server als Unterprozess auf Ihrem lokalen Computer zusammen mit dem Agenten ausgeführt, der mithilfe von JSON-RPC über standardmäßige Eingabe- und Ausgabestreams mit dem Server kommuniziert. Dieser Ansatz erfordert keine Authentifizierung zwischen dem Client und dem Server. Tools können mit lokalen Anwendungen und Dateien interagieren, lokal gespeicherte Anmeldeinformationen verwenden und übernehmen den Netzwerkzugriff des lokalen Computers des Benutzers. Dies ist das einfachste Hosting-Muster und bietet mehrere Vorteile.

Viele Kunden beginnen mit MCP und verwenden lokale Server. Sie ermöglichen es Ingenieuren, eine Vielzahl von Problemen in ihrer lokalen Umgebung schnell zu wiederholen und zu lösen. Stellen Sie sich einen MCP-Server vor, der eine Verbindung zu einem Git-Repository herstellt, das der Programmierassistent eines Ingenieurs verwendet. Es ist sehr sinnvoll, den MCP-Server lokal zu halten, da er die eindeutigen Anmeldeinformationen des Technikers für den Zugriff auf das Repository verwenden kann und kein zusätzlicher Netzwerkaufruf zu einem entfernten MCP-Server hinzugefügt wird. Die folgende Abbildung zeigt einen lokal gehosteten MCP-Server, der mit einem Codierungsagenten in einer IDE verwendet wird.



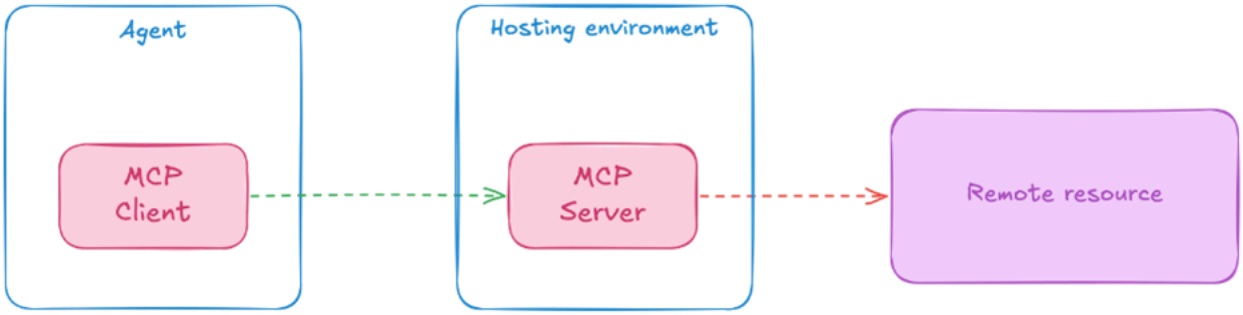
Bei diesen Bereitstellungstypen müssen Sie berücksichtigen, wie die MCP-Server entwickelt und verteilt werden. Die meisten Kunden entwickeln eine MCP-Registrierung, in der Server von Endbenutzern registriert und heruntergeladen werden können. Es ist einer Container-Registry sehr ähnlich, in der ein Benutzer nach bestimmten Funktionen suchen und die MCP-Server finden kann, die seinen Anforderungen entsprechen.

Es gibt öffentliche MCP-Register, wie z. B. Official MCP Registry, und es gibt privat gehostete Registries. Organizations richten ihre MCP-Registrierungsstrategie in der Regel an bestehenden Richtlinien rund um die Verteilung von Open-Source-Software, Container-Registrierungen und internes Paketmanagement aus. Sie sollten Faktoren wie Sicherheitsscans, Genehmigungsworkflows und Compliance-Anforderungen berücksichtigen.

Lokales Hosting bringt jedoch betriebliche Herausforderungen mit sich, die Unternehmen berücksichtigen sollten. Zunächst müssen Endbenutzer MCP-Server unabhängig voneinander ermitteln, herunterladen und konfigurieren. Dies kann die Komplexität der ersten Schritte mit jedem einzelnen MCP-Server erhöhen, den sie lokal verwenden. Zweitens können Sie den Lebenszyklus des MCP-Servers nicht kontrollieren, was bedeutet, dass Benutzer möglicherweise weiterhin veraltete Versionen mit Sicherheitslücken oder fehlenden Funktionen lokal ausführen. Dies kann die Einhaltung von Compliance-Anforderungen erschweren. Einige IDEs und CLI-Tools, wie Kiro, ermöglichen es Unternehmen, zu verwalten und zu kontrollieren, welche MCP-Tools verfügbar sind, wodurch Konsistenz und Sicherheit zwischen den Teams gewährleistet werden.

Remote-Hosting

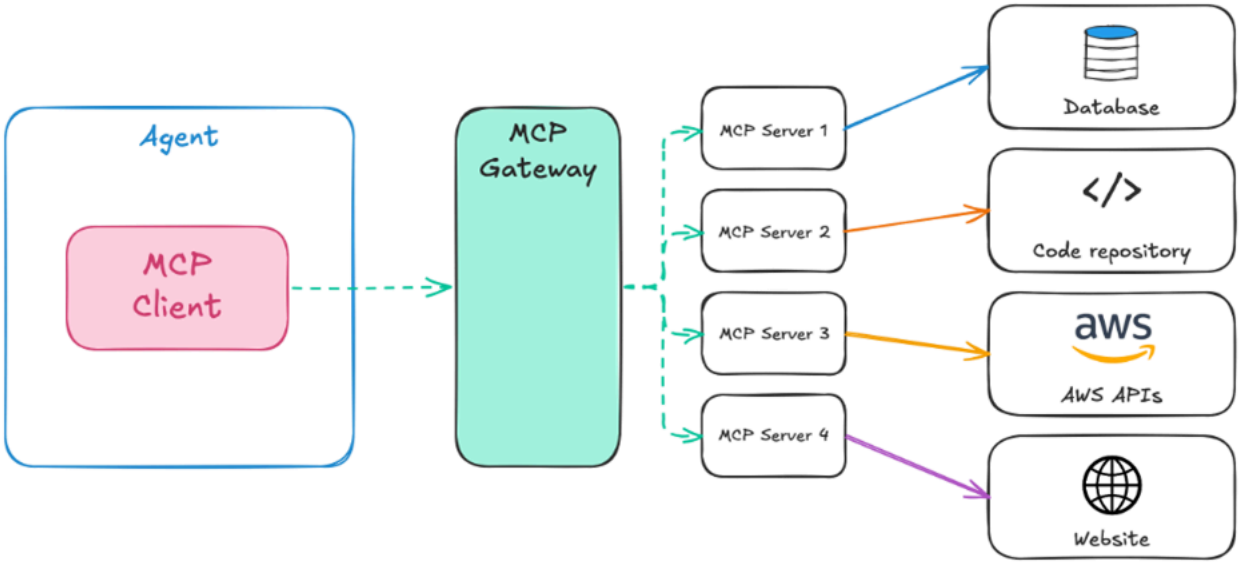
Die zweite Option besteht darin, Remote-MCP-Server zu hosten, auf die über HTTP oder HTTPS zugegriffen wird. Dies ermöglicht den Zugriff auf jeden mit dem Netzwerk verbundenen Client. Mithilfe von Remote-Hosting können Sie den Zugriff auf MCP-Ressourcen und -Funktionen zentral steuern, Authentifizierung und Autorisierung implementieren und die Versionierung und Aktualisierung der MCP-Serverlogik steuern. Für das Remote-Hosting ist weiterhin die Verwendung einer MCP-Registrierung erforderlich, damit Endbenutzer die MCP-Server ermitteln können, die sie mit ihrem Agenten verwenden möchten. Die folgende Abbildung zeigt den Remote-Hosting-Ansatz.



Aus Sicht der Agentenentwicklung ist die Erfahrung ähnlich, unabhängig davon, ob der MCP-Server lokal oder remote ist. Die wichtigste Änderung ist die Implementierung von Authentifizierung und Autorisierung, was sowohl den Zugriff des Agenten auf den MCP-Server als auch den Zugriff des Servers auf externe Ressourcen einschließt. Die Implementierung von Remote-MCP-Servern muss sorgfältig geplant werden, um den Mehrmandantenzugriff und die Rechteverwaltung zu berücksichtigen. Das Kapitel zur MCP-Governance-Strategie enthält weitere Informationen zu Überlegungen zur Authentifizierung und Autorisierung.

MCP-Gateway

Die letzte Option ist die Verwendung eines MCP-Gateways. MCP-Gateways fungieren als zentraler Proxy zwischen MCP-Clients und -Servern und orchestrieren den Zugriff auf die registrierten MCP-Server. Ohne Gateway muss jeder Agent jeden Remote-MCP-Server registrieren, den er möglicherweise verwenden möchte. Ein Gateway ermöglicht es dem Agenten, eine Verbindung zu einem einzelnen Endpunkt herzustellen, der die Authentifizierung, Autorisierung, das Routing und die Protokollübersetzung verwaltet. Neue MCP-Server und -Tools können dynamisch hinzugefügt und dem Agenten sofort zur Verfügung gestellt werden. Die folgende Abbildung zeigt den MCP-Gateway-Ansatz.



Einige Gateway-Lösungen, wie Docker MCP Gateway, verwalten auch den Lebenszyklus der MCP-Server und starten Server bei Bedarf auf Abruf. MCP-Gateways wie Amazon Bedrock AgentCore Gateway können auch bei der Verwaltung der Tool-Erkennung helfen, indem sie native semantische Suchfunktionen bereitstellen. Dies bietet Agenten einen einzigen Endpunkt, über den sie sich mit einem MCP-Client verbinden können, und hilft, die Nutzung ihres Kontextfensters zu optimieren. Das Ergebnis sind einfache Agenten, die MCP-Tools auswählen und effektiv nutzen können. Es birgt jedoch ähnliche identitätsbezogene Herausforderungen wie der Ansatz mit Remote-MCP-Servern.

Bewährte Methoden für das Hosten von MCP-Servern

- Das Spektrum der Hosting-Optionen ist nicht einheitlich. Ein Großteil der Nutzung von MCP-Servern erfolgt heute lokal.
- Wenn Sie anfangen, MCP-Remote-Server zu verwenden, sollten Sie vor allem auf eine konsistente Authentifizierung und Autorisierung für den MCP-Server achten und darauf, wie der MCP-Server die Authentifizierung und Autorisierung für nachgelagerte Ressourcen durchführt.
- MCP-Gateways vereinfachen die Konnektivität sowie die Authentifizierung und Autorisierung für das Hosten mehrerer Remote-MCP-Server. Sie bieten auch Funktionen zur Verbesserung der Verwaltung von Kontextfenstern, indem nach geeigneten Tools gesucht wird.

MCP-Verwaltungsstrategie

Die andere wichtige Funktion, die MCP Unternehmen bietet, ist die Unterstützung zentralisierter Governance. Ihre MCP-Governance-Strategie sollte sich mit der Authentifizierung und Autorisierung sowohl der MCP-Server als auch der Ressourcen befassen, auf die sie zugreifen. Sie sollte sich auch mit der Ratenbegrenzung befassen, um nachgelagerte Ressourcen zu schützen, betriebliche Kennzahlen zur Überwachung der Nutzung und Leistung der Tools sowie die Verwaltung der Bereitstellung und Verteilung von MCP-Servern.

Authentifizierung und Autorisierung

Einer der wichtigsten Bestandteile Ihrer Authentifizierungs- und Autorisierungsstrategie ist die Verwaltung des Downstream-Ressourcenzugriffs von MCP-Servern aus. Wenn ein Benutzer einen Agenten aufruft, werden Authentifizierung und Autorisierung durchgeführt, um sicherzustellen, dass der Benutzer berechtigt ist, den Agenten anzurufen. Anschließend orchestriert der Agent den Aufruf bestimmter Tools auf MCP-Servern. Sie müssen für jedes Tool entscheiden, wie der Zugriff autorisiert werden soll.

Eine Option ist die machine-to-machine Autorisierung, bei der keine Zustimmung oder Interaktion des Benutzers erforderlich ist. Beispielsweise verwendet ein zeitbasierter Agentenaufruf einen MCP-Server, um Protokolle von einer Anwendung zu sammeln und zu analysieren. In diesem Szenario ist der Agent vorab autorisiert, auf die angegebenen Daten zuzugreifen. Die zweite Option ist der vom Benutzer delegierte Zugriff, bei dem ein Benutzer seine Zustimmung zum Zugriff auf benutzerspezifische Daten und Ressourcen erteilt.

Die folgende Tabelle zeigt Authentifizierungs- und Autorisierungsmuster.

Faktor	Vom Benutzer delegierter Zugriff	Machine-to-machine
Eigentum an Daten	Benutzerspezifische Autorisierung von Daten	System- oder organisationsweite Daten
Interaktion mit dem Benutzer	Der Benutzer ist anwesend und kann zustimmen	Keine Benutzerinteraktion

Zeitpunkt des Vorgangs	Interaktiv oder in Echtzeit	Hintergrund, geplant oder stapelweise
Umfang der Genehmigung	Die Berechtigungen variieren je nach Benutzer	Konsistente Berechtigungen auf Agentenebene

Der vom Benutzer delegierte Zugriff erfordert eine sorgfältige Implementierung und sollte zusammen mit Ihrem Sicherheitsteam entwickelt werden. Agenten müssen in der Lage sein, zu beurteilen, welche Tools ein LLM ausgewählt hat und ob für sie eine zusätzliche Autorisierung erforderlich ist. MCP-Tools müssen Beschreibungen enthalten, aus denen hervorgeht, welche Authentifizierungs- und Autorisierungsanforderungen sie haben und wo Zugriffstoken abgerufen werden können. Client-Anwendungen müssen Zwischenauthentifizierungsanforderungen unterstützen, und der MCP-Client muss die abgerufenen Anmeldeinformationen bei jedem Tool-Aufruf an den Agenten zurücksenden.

Sie sollten sicherstellen, dass MCP-Tools immer über eigene Token für den Zugriff auf externe Funktionen verfügen und dass der Zugriff protokolliert und geprüft wird. Benutzeranmeldedaten sollten nicht über Ihr Agentensystem weitergegeben werden. Beispielsweise sollten Ihre MCP-Server nicht dasselbe Token für den Zugriff auf Daten verwenden, mit dem der Agent aufgerufen wurde. Downstream-Aufrufe sollten speziell generierte Token mit explizitem Gültigkeitsbereich verwenden. Dies trägt dazu bei, zusätzliche Schutzmaßnahmen bereitzustellen, um einen unbeabsichtigten Datenzugriff im Rahmen von Aktionen zu verhindern. Auf diese Weise kann auch verhindert werden, dass Halluzinationen unbeabsichtigte Folgen haben. Stellen Sie sich vor, ein Benutzer mit vollen Administratorrechten bittet einen Agenten, eine Produktionsdatenbank zur Verwendung in der Vorproduktion zu klonen. Dazu benötigt der Benutzer lediglich `CREATE` Berechtigungen`READ`. Nehmen wir an, das LLM halluziniert und glaubt, dass es im Rahmen dieser Anfrage die alte Datenbank bereinigen muss. Wenn die Anmeldeinformationen des Benutzers wiederverwendet werden, wäre dies wahrscheinlich erfolgreich, da die ursprünglichen Anmeldeinformationen des Benutzers über Berechtigungen verfügen. `DELETE` Wenn der MCP-Server stattdessen ein Token verwendet, das bewusst auf den Umfang der Daten für die Anfrage beschränkt ist `READ` und nur die `CREATE` Rechte und Rechte hat, würde der Versuch, die Produktionsdatenbank zu löschen, fehlschlagen.

Sie können Amazon Bedrock AgentCore Identity verwenden, um diese Muster zu implementieren. Stellen Sie sicher, dass Sie bewusst entscheiden, ob die Berechtigungen zum Auflisten und Aufrufen von Tools, die von einem MCP-Server gehostet werden, auch Berechtigungen für die externen Funktionen beinhalten, die der MCP-Server bereitstellt. Dieser Identitätsfluss vom MCP-Server zur Ressource und zurück zum Benutzer hängt von der Art des verwendeten Authentifizierungs- und

Autorisierungsdienstes ab. Sie müssen entscheiden, wie dies in großem Umfang für Ihre MCP-Server gehandhabt wird.

Implementieren Sie bei der Gestaltung Ihrer Authentifizierungs- und Autorisierungsmuster Mechanismen zur Token-Isolierung, die für jedes Tool, auf das zugegriffen wird, unterschiedliche Zugriffstoken abrufen. Verwenden Sie Token nicht erneut zwischen Tools und Servern. AgentCore Identity bietet diese Fähigkeit zur Token-Isolierung. Es verwaltet automatisch sowohl Workload-Token (zur machine-to-machine Authentifizierung) als auch Benutzertoken (für vom Benutzer delegierten Zugriff), um eine korrekte Trennung sicherzustellen und eine Eskalation von Berechtigungen zu verhindern. Dies ist besonders wichtig, wenn Remote-MCP-Server oder MCP-Gateways integriert werden.

Bewährte Methoden für die MCP-Authentifizierung und -Autorisierung

- Token-Trennung — Geben Sie keine Inhaber-Token von Anrufern an nachgelagerte Dienste weiter. Stellen Sie sicher, dass das Feld aud (Audience) mit dem Server übereinstimmt, der das Token empfängt. Der Zielgruppenanspruch gibt an, für welchen Dienst das Token bestimmt ist, und verhindert so die unbefugte Wiederverwendung von Token auf verschiedenen MCP-Servern.
- Wählen Sie einen Zugriffsansatz — Wählen Sie für jedes Tool, das Ihre MCP-Server bereitstellen, zwischen machine-to-machine und vom Benutzer delegiertem Zugriff. Erwägen Sie, Tools auf demselben MCP-Server zu gruppieren, die dasselbe Authentifizierungsmuster verwenden.

Steuerung der Last

Wie bei jedem verteilten System müssen Sie überlegen, wie Sie die Last in Ihrer MCP-Serverflotte kontrollieren können. Zunächst überlegen Sie, ob Sie eine Ratenbegrenzung auf Ihren MCP-Servern implementieren sollten und wo die Grenzwerte implementiert werden sollen. Wenn Sie sich dafür entscheiden, keine Ratenbegrenzung zu implementieren, geben Sie jegliche Ratenbegrenzung weiter, die von nachgelagerten Ressourcen vorgenommen wird. Viele Systeme entscheiden sich für eine Ratenbegrenzung auf der Grundlage von Anforderungsattributen, wie z. B. einer Benutzer- oder Konto-ID. Stellen Sie sicher, dass die Anfragen, die an nachgelagerte Dienste gesendet werden, dieselben Attribute aufweisen, sodass mehrere Benutzer nicht durch die Belastung durch einen anderen Benutzer beeinträchtigt werden.

Wenn Sie sich für die Implementierung einer Ratenbegrenzung entscheiden, empfiehlt es sich, die primäre Ratenbegrenzung auf MCP-Serverebene zu implementieren, wobei Back-End-Dienste sekundären Schutz bieten und Agenten ihr Verhalten auf der Grundlage von Ratenbegrenzungen

anpassen. Überlegen Sie, ob die Ratenbegrenzungen pro MCP-Server oder pro Tool gelten. Serverratenbeschränkungen pro MCP tragen zum Schutz Ihrer MCP-Serverflotte und -Services in einer Umgebung mit mehreren Mandanten bei. Das kann jedoch sehr grobkörnig sein. Die Ratenbegrenzungen pro Werkzeug sollen verhindern, dass nachgelagerte Ressourcen überlastet werden, die sich selbst möglicherweise nicht ausreichend einschränken. Wenn ein Tool mehrere Funktionen aufruft APIs, sollten Sie das Ratenlimit so einstellen, dass es sich an der niedrigsten Rate orientiert, die für diese APIs Tools zulässig ist.

Die Weitergabe von Informationen zur Ratenbegrenzung in HTTP-Headern kann auch eine nützliche Metrik für Benutzer und automatisierte Systeme sein, um ihre eigene Anforderungsrate und Wiederholungsstrategie zu verwalten. Sie können diese Header beispielsweise von Ihrem MCP-Server an den Agenten zurücksenden, wie im folgenden Beispiel gezeigt:

```
X-RateLimit-Limit: 100
X-RateLimit-Remaining: 45
X-RateLimit-Reset: 1640995200
```

Darüber hinaus sollten Sie einen Lastabbau in Betracht ziehen, um den gesamten Service zu schützen, wenn kein einziger Kunde ein Ratenlimit überschreitet, die Last jedoch die Systemleistung beeinträchtigt.

Bewährte Methoden zur Steuerung der Auslastung

- Wählen Sie einen Ansatz zur Ratenbegrenzung — Planen Sie eine Ratenbegrenzung für einzelne Benutzer ein, entweder auf der Grundlage ihrer Nutzung nachgelagerter Ressourcen oder aufgrund ihrer Nutzung Ihres MCP-Servers und Ihrer Tools.
- Ziehen Sie Lastabbau in Betracht — Schützen Sie Ihre MCP-Serverflotte vor allgemeiner Überlastung, die nicht von einem einzelnen oder einer Handvoll Kunden verursacht wird.

Operationelle Metriken

Die wichtigsten Kennzahlen, die für MCP-Implementierungen erfasst werden müssen, sollten sich auf das Kundenerlebnis konzentrieren, das sie bieten. Zu diesen Kennzahlen gehören in der Regel die Token-Nutzung, die Genauigkeit der Toolauswahl, die Anzahl der beim Agenten registrierten Tools und die Latenz der Tools. Durch die Überwachung der von den einzelnen Tools zurückgegebenen Ausgabetokens können Sie beispielsweise Alarme einrichten, wenn Tools einen Schwellenwert für die Nutzung von Kontextfenstern überschreiten. Wenn ein Tool diesen

Schwellenwert überschreitet, sollten Sie das Verhalten des Tools überprüfen. Dies steht auch im Zusammenhang mit der Designstrategie des MCP-Tools. Kennzahlen zur Genauigkeit der Toolauswahl geben Aufschluss darüber, wie gut die Agenten die geeigneten Tools für bestimmte Aufgaben auswählen, während Ausführungsgeschwindigkeit und Erfolgsquoten Leistungsengpässe und Zuverlässigkeitsprobleme aufzeigen.

Um beispielsweise die Genauigkeitsmetriken für die Toolauswahl und den Tool-Einsatz zu bewerten, erstellten AWS Teams Gold-Datasets für Regressionstests. Die Datensätze wurden synthetisch generiert, indem historische API-Aufrufprotokolle bei Benutzeranfragen verwendet LLMs wurden. Anhand der vordefinierten Metriken zur Werkzeugauswahl und zum Werkzeugeinsatz (wie Genauigkeit der Werkzeugauswahl, Genauigkeit der Werkzeugparameter und Genauigkeit von Funktionsaufrufen mit mehreren Turns) konnten die AWS Teams objektiv beurteilen, ob der KI-Agent in der Lage ist, die geeigneten Tools korrekt zu identifizieren, ihre Parameter mit genauen Werten zu füllen und über Gesprächsrunden hinweg kohärente Werkzeugaufrufsequenzen aufrechtzuerhalten.

Durch die Messung von Kennzahlen zur Anzahl der bei einem Agenten registrierten Tools können Sie potenzielle Herausforderungen bei der Verwaltung von Kontextfenstern sowie Änderungen an den verfügbaren Tools von MCP-Servern identifizieren. Sie sollten regelmäßig Betriebskennzahlen überprüfen, die Aufschluss über die Benutzererfahrung mit Ihrem MCP-Server und den Tools geben.

Mitwirkende

Verfassen

- Alex Torres, leitender Lösungsarchitekt, AWS
- Saikat Gomes, Senior Manager für Kundenlösungen, AWS
- Mike Haken, leitender Lösungsarchitekt, AWS
- Sreeja Das, Chefingenieur, AWS

Überprüfend

- Ted Swinyar, Leiter des Lösungsarchitekten, AWS
- Raju Patil, leitender Datenwissenschaftler, AWS

Technisches Schreiben

- Lilly AbouHarb, leitende technische Redakteurin, AWS

Dokumentverlauf

In der folgenden Tabelle werden wichtige Änderungen in diesem Leitfaden beschrieben. Um Benachrichtigungen über zukünftige Aktualisierungen zu erhalten, können Sie einen RSS-Feed abonnieren.

Änderung	Beschreibung	Datum
Erste Veröffentlichung	—	16. März 2026

AWS Glossar zu präskriptiven Leitlinien

Die folgenden Begriffe werden häufig in Strategien, Leitfäden und Mustern verwendet, die von AWS Prescriptive Guidance bereitgestellt werden. Um Einträge vorzuschlagen, verwenden Sie bitte den Link Feedback geben am Ende des Glossars.

Zahlen

7 Rs

Sieben gängige Migrationsstrategien für die Verlagerung von Anwendungen in die Cloud. Diese Strategien bauen auf den 5 Rs auf, die Gartner 2011 identifiziert hat, und bestehen aus folgenden Elementen:

- Refactor/re-architect — Verschieben Sie eine Anwendung und ändern Sie ihre Architektur, indem Sie alle Vorteile der Cloud-nativen Funktionen nutzen, um Agilität, Leistung und Skalierbarkeit zu verbessern. Dies beinhaltet in der Regel die Portierung des Betriebssystems und der Datenbank. Beispiel: Migrieren Sie Ihre lokale Oracle-Datenbank auf die Amazon Aurora PostgreSQL-Compatible Edition.
- Plattformwechsel (Lift and Reshape) – Verschieben Sie eine Anwendung in die Cloud und führen Sie ein gewisses Maß an Optimierung ein, um die Cloud-Funktionen zu nutzen. Beispiel: Migrieren Sie Ihre lokale Oracle-Datenbank zu Amazon Relational Database Service (Amazon RDS) für Oracle in der. AWS Cloud
- Neukauf (Drop and Shop) – Wechseln Sie zu einem anderen Produkt, indem Sie typischerweise von einer herkömmlichen Lizenz zu einem SaaS-Modell wechseln. Beispiel: Migrieren Sie Ihr Kundenbeziehungsmanagement (CRM) -System zu. Salesforce.com
- Hostwechsel (Lift and Shift) – Verschieben Sie eine Anwendung in die Cloud, ohne Änderungen vorzunehmen, um die Cloud-Funktionen zu nutzen. Beispiel: Migrieren Sie Ihre lokale Oracle-Datenbank zu Oracle auf einer EC2-Instanz in der. AWS Cloud
- Verschieben (Lift and Shift auf Hypervisor-Ebene) – Verlagern Sie die Infrastruktur in die Cloud, ohne neue Hardware kaufen, Anwendungen umschreiben oder Ihre bestehenden Abläufe ändern zu müssen. Sie migrieren Server von einer lokalen Plattform zu einem Cloud-Dienst für dieselbe Plattform. Beispiel: Migrieren Sie eine Microsoft Hyper-V Anwendung zu AWS.
- Beibehaltung (Wiederaufgreifen) – Bewahren Sie Anwendungen in Ihrer Quellumgebung auf. Dazu können Anwendungen gehören, die einen umfangreichen Faktorwechsel erfordern und

die Sie auf einen späteren Zeitpunkt verschieben möchten, sowie ältere Anwendungen, die Sie beibehalten möchten, da es keine geschäftliche Rechtfertigung für ihre Migration gibt.

- Außerbetriebnahme – Dekommissionierung oder Entfernung von Anwendungen, die in Ihrer Quellumgebung nicht mehr benötigt werden.

A

A2A () Agent-to-Agent

Ein Stateful-Protokoll für die Zusammenarbeit zwischen Agenten, das die Delegierung von Aufgaben und die Zustandsübertragung unterstützt.

ABAC

Siehe attributbasierte Zugriffskontrolle.

abstrahierte Dienste

Siehe Managed Services.

ACID

Siehe Atomarität, Konsistenz, Isolierung und Haltbarkeit.

Aktiv-Aktiv-Migration

Eine Datenbankmigrationsmethode, bei der die Quell- und Zieldatenbanken synchron gehalten werden (mithilfe eines bidirektionalen Replikationstools oder dualer Schreibvorgänge) und beide Datenbanken Transaktionen von miteinander verbundenen Anwendungen während der Migration verarbeiten. Diese Methode unterstützt die Migration in kleinen, kontrollierten Batches, anstatt einen einmaligen Cutover zu erfordern. Es ist flexibler, erfordert aber mehr Arbeit als eine aktiv-passive Migration.

Aktiv-Passiv-Migration

Eine Datenbankmigrationsmethode, bei der die Quell- und Zieldatenbanken synchron gehalten werden, aber nur die Quelldatenbank verarbeitet Transaktionen von verbindenden Anwendungen, während Daten in die Zieldatenbank repliziert werden. Die Zieldatenbank akzeptiert während der Migration keine Transaktionen.

Agent

Ein KI-System, das mithilfe von Tools selbständig Überlegungen anstellen, planen und Maßnahmen ergreifen kann, um Ziele zu erreichen.

Agent Ops

Operative Verfahren zum Erstellen, Testen, Bereitstellen und Ausführen von KI-Agenten in der Produktion im großen Maßstab.

Aggregatfunktion

Eine SQL-Funktion, die mit einer Gruppe von Zeilen arbeitet und einen einzelnen Rückgabewert für die Gruppe berechnet. Beispiele für Aggregatfunktionen sind `SUM` und`MAX`.

AI

Siehe künstliche Intelligenz.

AIOps

Siehe Operationen mit künstlicher Intelligenz.

Anonymisierung

Der Prozess des dauerhaften Löschens personenbezogener Daten in einem Datensatz. Anonymisierung kann zum Schutz der Privatsphäre beitragen. Anonymisierte Daten gelten nicht mehr als personenbezogene Daten.

Anti-Muster

Eine häufig verwendete Lösung für ein wiederkehrendes Problem, bei dem die Lösung kontraproduktiv, ineffektiv oder weniger wirksam als eine Alternative ist.

Anwendungssteuerung

Ein Sicherheitsansatz, bei dem nur zugelassene Anwendungen verwendet werden können, um ein System vor Schadsoftware zu schützen.

Anwendungsportfolio

Eine Sammlung detaillierter Informationen zu jeder Anwendung, die von einer Organisation verwendet wird, einschließlich der Kosten für die Erstellung und Wartung der Anwendung und ihres Geschäftswerts. Diese Informationen sind entscheidend für den Prozess der Portfoliofindung und -analyse und hilft bei der Identifizierung und Priorisierung der Anwendungen, die migriert, modernisiert und optimiert werden sollen.

künstliche Intelligenz (KI)

Das Gebiet der Datenverarbeitungswissenschaft, das sich der Nutzung von Computertechnologien zur Ausführung kognitiver Funktionen widmet, die typischerweise mit

Menschen in Verbindung gebracht werden, wie Lernen, Problemlösen und Erkennen von Mustern. Weitere Informationen finden Sie unter Was ist künstliche Intelligenz?

Operationen mit künstlicher Intelligenz (AIOps)

Der Prozess des Einsatzes von Techniken des Machine Learning zur Lösung betrieblicher Probleme, zur Reduzierung betrieblicher Zwischenfälle und menschlicher Eingriffe sowie zur Steigerung der Servicequalität. Weitere Informationen zur Verwendung von AIOps in der AWS -Migrationsstrategie finden Sie im Leitfaden zur Betriebsintegration.

Asymmetrische Verschlüsselung

Ein Verschlüsselungsalgorithmus, der ein Schlüsselpaar, einen öffentlichen Schlüssel für die Verschlüsselung und einen privaten Schlüssel für die Entschlüsselung verwendet. Sie können den öffentlichen Schlüssel teilen, da er nicht für die Entschlüsselung verwendet wird. Der Zugriff auf den privaten Schlüssel sollte jedoch stark eingeschränkt sein.

Atomizität, Konsistenz, Isolierung, Haltbarkeit (ACID)

Eine Reihe von Softwareeigenschaften, die die Datenvalidität und betriebliche Zuverlässigkeit einer Datenbank auch bei Fehlern, Stromausfällen oder anderen Problemen gewährleisten.

Attributbasierte Zugriffskontrolle (ABAC)

Die Praxis, detaillierte Berechtigungen auf der Grundlage von Benutzerattributen wie Abteilung, Aufgabenrolle und Teamname zu erstellen. Weitere Informationen finden Sie unter ABAC AWS in der AWS Identity and Access Management (IAM-) Dokumentation.

autoritative Datenquelle

Ein Ort, an dem Sie die primäre Version der Daten speichern, die als die zuverlässigste Informationsquelle angesehen wird. Sie können Daten aus der maßgeblichen Datenquelle an andere Speicherorte kopieren, um die Daten zu verarbeiten oder zu ändern, z. B. zu anonymisieren, zu redigieren oder zu pseudonymisieren.

Availability Zone

Ein bestimmter Standort innerhalb einer AWS-Region , der vor Ausfällen in anderen Availability Zones geschützt ist und kostengünstige Netzwerkkonnektivität mit niedriger Latenz zu anderen Availability Zones in derselben Region bietet.

AWS Framework für die Einführung der Cloud (AWS CAF)

Ein Framework mit Richtlinien und bewährten Verfahren, das Unternehmen bei der Entwicklung eines effizienten und effektiven Plans für den erfolgreichen Umstieg auf die Cloud unterstützt.

AWS AWS CAF unterteilt die Leitlinien in sechs Schwerpunktbereiche, die als Perspektiven bezeichnet werden: Unternehmen, Mitarbeiter, Unternehmensführung, Plattform, Sicherheit und Betrieb. Die Perspektiven Geschäft, Mitarbeiter und Unternehmensführung konzentrieren sich auf Geschäftskompetenzen und -prozesse, während sich die Perspektiven Plattform, Sicherheit und Betriebsabläufe auf technische Fähigkeiten und Prozesse konzentrieren. Die Personalperspektive zielt beispielsweise auf Stakeholder ab, die sich mit Personalwesen (HR), Personalfunktionen und Personalmanagement befassen. Aus dieser Perspektive bietet AWS CAF Leitlinien für Personalentwicklung, Schulung und Kommunikation, um das Unternehmen auf eine erfolgreiche Cloud-Einführung vorzubereiten. Weitere Informationen finden Sie auf der AWS -CAF-Webseite und dem AWS -CAF-Whitepaper.

AWS Workload-Qualifizierungsrahmen (AWS WQF)

Ein Tool, das Workloads bei der Datenbankmigration bewertet, Migrationsstrategien empfiehlt und Arbeitsschätzungen bereitstellt. AWS WQF ist in () enthalten. AWS Schema Conversion Tool AWS SCT Es analysiert Datenbankschemas und Codeobjekte, Anwendungscode, Abhängigkeiten und Leistungsmerkmale und stellt Bewertungsberichte bereit.

B

schlechter Bot

Ein Bot, der Einzelpersonen oder Organisationen stören oder ihnen Schaden zufügen soll.

BCP

Siehe Planung der Geschäftskontinuität.

Verhaltensdiagramm

Eine einheitliche, interaktive Ansicht des Ressourcenverhaltens und der Interaktionen im Laufe der Zeit. Sie können ein Verhaltensdiagramm mit Amazon Detective verwenden, um fehlgeschlagene Anmeldeversuche, verdächtige API-Aufrufe und ähnliche Vorgänge zu untersuchen. Weitere Informationen finden Sie unter Daten in einem Verhaltensdiagramm in der Detective-Dokumentation.

Big-Endian-System

Ein System, welches das höchstwertige Byte zuerst speichert. Siehe auch Endianness.

Binäre Klassifikation

Ein Prozess, der ein binäres Ergebnis vorhersagt (eine von zwei möglichen Klassen). Beispielsweise könnte Ihr ML-Modell möglicherweise Probleme wie „Handelt es sich bei dieser E-Mail um Spam oder nicht?“ vorhersagen müssen oder „Ist dieses Produkt ein Buch oder ein Auto?“

Bloom-Filter

Eine probabilistische, speichereffiziente Datenstruktur, mit der getestet wird, ob ein Element Teil einer Menge ist.

blue/green Einsatz

Eine Bereitstellungsstrategie, bei der Sie zwei separate, aber identische Umgebungen erstellen. Sie führen die aktuelle Anwendungsversion in einer Umgebung (blau) und die neue Anwendungsversion in der anderen Umgebung (grün) aus. Mit dieser Strategie können Sie schnell und mit minimalen Auswirkungen ein Rollback durchführen.

Bot

Eine Softwareanwendung, die automatisierte Aufgaben über das Internet ausführt und menschliche Aktivitäten oder Interaktionen simuliert. Manche Bots sind nützlich oder nützlich, wie z. B. Webcrawler, die Informationen im Internet indexieren. Einige andere Bots, sogenannte bösartige Bots, sollen Einzelpersonen oder Organisationen stören oder ihnen Schaden zufügen.

Botnetz

Netzwerke von Bots, die mit Malware infiziert sind und unter der Kontrolle einer einzigen Partei stehen, die als Bot-Herder oder Bot-Operator bezeichnet wird. Botnetze sind der bekannteste Mechanismus zur Skalierung von Bots und ihrer Wirkung.

branch

Ein containerisierter Bereich eines Code-Repositorys. Der erste Zweig, der in einem Repository erstellt wurde, ist der Hauptzweig. Sie können einen neuen Zweig aus einem vorhandenen Zweig erstellen und dann Feature entwickeln oder Fehler in dem neuen Zweig beheben. Ein Zweig, den Sie erstellen, um ein Feature zu erstellen, wird allgemein als Feature-Zweig bezeichnet. Wenn das Feature zur Veröffentlichung bereit ist, führen Sie den Feature-Zweig wieder mit dem Hauptzweig zusammen. Weitere Informationen finden Sie unter Über Branches (GitHub Dokumentation).

Zugang durch Glasbruch

Unter außergewöhnlichen Umständen und im Rahmen eines genehmigten Verfahrens ist dies eine schnelle Methode für einen Benutzer, auf einen Bereich zuzugreifen AWS-Konto , für den er in der Regel keine Zugriffsrechte besitzt. Weitere Informationen finden Sie in den Leitlinien unter dem Indikator „Glasbruchverfahren implementieren“. AWS Well-Architected

Brownfield-Strategie

Die bestehende Infrastruktur in Ihrer Umgebung. Wenn Sie eine Brownfield-Strategie für eine Systemarchitektur anwenden, richten Sie sich bei der Gestaltung der Architektur nach den Einschränkungen der aktuellen Systeme und Infrastruktur. Wenn Sie die bestehende Infrastruktur erweitern, könnten Sie Brownfield- und Greenfield-Strategien mischen.

Puffer-Cache

Der Speicherbereich, in dem die am häufigsten abgerufenen Daten gespeichert werden.

Geschäftsfähigkeit

Was ein Unternehmen tut, um Wert zu generieren (z. B. Vertrieb, Kundenservice oder Marketing). Microservices-Architekturen und Entwicklungsentscheidungen können von den Geschäftskapazitäten beeinflusst werden. Weitere Informationen finden Sie im Abschnitt Organisiert nach Geschäftskapazitäten des Whitepapers Ausführen von containerisierten Microservices in AWS.

Planung der Geschäftskontinuität (BCP)

Ein Plan, der die potenziellen Auswirkungen eines störenden Ereignisses, wie z. B. einer groß angelegten Migration, auf den Betrieb berücksichtigt und es einem Unternehmen ermöglicht, den Betrieb schnell wieder aufzunehmen.

C

CAF

Weitere Informationen finden Sie unter AWS Framework für die Cloud-Einführung.

Bereitstellung auf Kanaren

Die langsame und schrittweise Veröffentlichung einer Version für Endbenutzer. Wenn Sie sich sicher sind, stellen Sie die neue Version bereit und ersetzen die aktuelle Version vollständig.

CCoE

Weitere Informationen finden Sie im Cloud Center of Excellence.

CDC

Siehe Erfassung von Änderungsdaten.

Erfassung von Datenänderungen (CDC)

Der Prozess der Nachverfolgung von Änderungen an einer Datenquelle, z. B. einer Datenbanktabelle, und der Aufzeichnung von Metadaten zu der Änderung. Sie können CDC für verschiedene Zwecke verwenden, z. B. für die Prüfung oder Replikation von Änderungen in einem Zielsystem, um die Synchronisation aufrechtzuerhalten.

Chaos-Technik

Absichtliches Einführen von Ausfällen oder Störungsereignissen, um die Widerstandsfähigkeit eines Systems zu testen. Sie können AWS Fault Injection Service (AWS FIS) verwenden, um Experimente durchzuführen, die Ihre AWS Workloads stress, und deren Reaktion zu bewerten.

CI/CD

Siehe Continuous Integration und Continuous Delivery.

Klassifizierung

Ein Kategorisierungsprozess, der bei der Erstellung von Vorhersagen hilft. ML-Modelle für Klassifikationsprobleme sagen einen diskreten Wert voraus. Diskrete Werte unterscheiden sich immer voneinander. Beispielsweise muss ein Modell möglicherweise auswerten, ob auf einem Bild ein Auto zu sehen ist oder nicht.

Citizen Developer

Ein Geschäftsanwender, der KI-Anwendungen mithilfe von Plattformen ohne Programmierkenntnisse erstellt. code/low

clientseitige Verschlüsselung

Lokale Verschlüsselung von Daten, bevor das Ziel sie AWS-Service empfängt.

Cloud-Kompetenzzentrum (CCoE)

Ein multidisziplinäres Team, das die Cloud-Einführung in der gesamten Organisation vorantreibt, einschließlich der Entwicklung bewährter Cloud-Methoden, der Mobilisierung von Ressourcen, der Festlegung von Migrationszeitplänen und der Begleitung der Organisation durch groß angelegte

Transformationen. Weitere Informationen finden Sie in den CCoE-Beiträgen im AWS Cloud Enterprise Strategy Blog.

Cloud Computing

Die Cloud-Technologie, die typischerweise für die Ferndatenspeicherung und das IoT-Gerätemanagement verwendet wird. Cloud Computing ist häufig mit Edge-Computing-Technologie verbunden.

Cloud-Betriebsmodell

In einer IT-Organisation das Betriebsmodell, das zum Aufbau, zur Weiterentwicklung und Optimierung einer oder mehrerer Cloud-Umgebungen verwendet wird. Weitere Informationen finden Sie unter Aufbau Ihres Cloud-Betriebsmodells.

Phasen der Einführung der Cloud

Die vier Phasen, die Unternehmen bei der Migration in der Regel durchlaufen AWS Cloud:

- Projekt – Durchführung einiger Cloud-bezogener Projekte zu Machbarkeitsnachweisen und zu Lernzwecken
- Fundament – Grundlegende Investitionen tätigen, um Ihre Cloud-Einführung zu skalieren (z. B. Einrichtung einer Landing Zone, Definition eines CCoE, Einrichtung eines Betriebsmodells)
- Migration – Migrieren einzelner Anwendungen
- Re-invention — Optimierung von Produkten und Dienstleistungen sowie Innovation in der Cloud

Diese Phasen wurden von Stephen Orban im Blogbeitrag The Journey Toward Cloud-First & the Stages of Adoption im AWS Cloud Enterprise Strategy-Blog definiert. Informationen darüber, wie sie mit der AWS Migrationsstrategie zusammenhängen, finden Sie im Leitfaden zur Vorbereitung der Migration.

CMDB

Siehe Datenbank für das Konfigurationsmanagement.

Code-Repository

Ein Ort, an dem Quellcode und andere Komponenten wie Dokumentation, Beispiele und Skripts gespeichert und im Rahmen von Versionskontrollprozessen aktualisiert werden. Zu den gängigen Cloud-Repositorys gehören GitHub oderBitbucket Cloud. Jede Version des Codes wird Zweig genannt. In einer Microservice-Struktur ist jedes Repository einer einzelnen Funktionalität gewidmet. Eine einzelne CI/CD Pipeline kann mehrere Repositorys verwenden.

Kalter Cache

Ein Puffer-Cache, der leer oder nicht gut gefüllt ist oder veraltete oder irrelevante Daten enthält. Dies beeinträchtigt die Leistung, da die Datenbank-Instance aus dem Hauptspeicher oder der Festplatte lesen muss, was langsamer ist als das Lesen aus dem Puffercache.

Kalte Daten

Daten, auf die selten zugegriffen wird und die in der Regel historisch sind. Bei der Abfrage dieser Art von Daten sind langsame Abfragen in der Regel akzeptabel. Durch die Verlagerung dieser Daten auf leistungsschwächere und kostengünstigere Speicherstufen oder -klassen können Kosten gesenkt werden.

Computer Vision (CV)

Ein Bereich der KI, der maschinelles Lernen nutzt, um Informationen aus visuellen Formaten wie digitalen Bildern und Videos zu analysieren und zu extrahieren. Amazon SageMaker AI bietet beispielsweise Bildverarbeitungsalgorithmen für CV.

Drift in der Konfiguration

Bei einer Arbeitslast eine Änderung der Konfiguration gegenüber dem erwarteten Zustand. Dies kann dazu führen, dass der Workload nicht mehr richtlinienkonform wird, und zwar in der Regel schrittweise und unbeabsichtigt.

Verwaltung der Datenbankkonfiguration (CMDB)

Ein Repository, das Informationen über eine Datenbank und ihre IT-Umgebung speichert und verwaltet, inklusive Hardware- und Softwarekomponenten und deren Konfigurationen. In der Regel verwenden Sie Daten aus einer CMDB in der Phase der Portfolioerkennung und -analyse der Migration.

Konformitätspaket

Eine Sammlung von AWS Config Regeln und Abhilfemaßnahmen, die Sie zusammenstellen können, um Ihre Konformitäts- und Sicherheitsprüfungen individuell anzupassen. Mithilfe einer YAML-Vorlage können Sie ein Conformance Pack als einzelne Entität in einer AWS-Konto AND-Region oder unternehmensweit bereitstellen. Weitere Informationen finden Sie in der Dokumentation unter Conformance Packs. AWS Config

kontinuierliche Integration und kontinuierliche Bereitstellung () CI/CD

Der Prozess der Automatisierung der Quell-, Build-, Test-, Staging- und Produktionsphasen des Softwareveröffentlichungsprozesses. CI/CD wird allgemein als Pipeline beschrieben. CI/CD

kann Ihnen helfen, Prozesse zu automatisieren, die Produktivität zu steigern, die Codequalität zu verbessern und schneller zu liefern. Weitere Informationen finden Sie unter Vorteile der kontinuierlichen Auslieferung. CD kann auch für kontinuierliche Bereitstellung stehen. Weitere Informationen finden Sie unterKontinuierliche Auslieferung im Vergleich zu kontinuierlicher Bereitstellung.

CV

Siehe Computer Vision.

D

Daten im Ruhezustand

Daten, die in Ihrem Netzwerk stationär sind, z. B. Daten, die sich im Speicher befinden.

Datenklassifizierung

Ein Prozess zur Identifizierung und Kategorisierung der Daten in Ihrem Netzwerk auf der Grundlage ihrer Kritikalität und Sensitivität. Sie ist eine wichtige Komponente jeder Strategie für das Management von Cybersecurity-Risiken, da sie Ihnen hilft, die geeigneten Schutz- und Aufbewahrungskontrollen für die Daten zu bestimmen. Die Datenklassifizierung ist ein Bestandteil der Sicherheitssäule des AWS Well-Architected Frameworks. Weitere Informationen finden Sie unter Datenklassifizierung.

Datendrift

Eine signifikante Variation zwischen den Produktionsdaten und den Daten, die zum Trainieren eines ML-Modells verwendet wurden, oder eine signifikante Änderung der Eingabedaten im Laufe der Zeit. Datendrift kann die Gesamtqualität, Genauigkeit und Fairness von ML-Modellvorhersagen beeinträchtigen.

Daten während der Übertragung

Daten, die sich aktiv durch Ihr Netzwerk bewegen, z. B. zwischen Netzwerkressourcen.

Datennetz

Ein architektonisches Framework, das verteilte, dezentrale Dateneigentum mit zentraler Verwaltung und Steuerung ermöglicht.

Datenminimierung

Das Prinzip, nur die Daten zu sammeln und zu verarbeiten, die unbedingt erforderlich sind. Durch Datenminimierung im AWS Cloud können Datenschutzrisiken, Kosten und der CO2-Fußabdruck Ihrer Analysen reduziert werden.

Datenperimeter

Eine Reihe präventiver Schutzmaßnahmen in Ihrer AWS Umgebung, die sicherstellen, dass nur vertrauenswürdige Identitäten auf vertrauenswürdige Ressourcen von erwarteten Netzwerken zugreifen. Weitere Informationen finden Sie unter Aufbau eines Datenperimeters auf. AWS

Vorverarbeitung der Daten

Rohdaten in ein Format umzuwandeln, das von Ihrem ML-Modell problemlos verarbeitet werden kann. Die Vorverarbeitung von Daten kann bedeuten, dass bestimmte Spalten oder Zeilen entfernt und fehlende, inkonsistente oder doppelte Werte behoben werden.

Herkunft der Daten

Der Prozess der Nachverfolgung des Ursprungs und der Geschichte von Daten während ihres gesamten Lebenszyklus, z. B. wie die Daten generiert, übertragen und gespeichert wurden.

betroffene Person

Eine Person, deren Daten gesammelt und verarbeitet werden.

Data Warehouse

Ein Datenverwaltungssystem, das Business Intelligence wie Analysen unterstützt. Data Warehouses enthalten in der Regel große Mengen historischer Daten und werden in der Regel für Abfragen und Analysen verwendet.

Datenbankdefinitionssprache (DDL)

Anweisungen oder Befehle zum Erstellen oder Ändern der Struktur von Tabellen und Objekten in einer Datenbank.

Datenbankmanipulationssprache (DML)

Anweisungen oder Befehle zum Ändern (Einfügen, Aktualisieren und Löschen) von Informationen in einer Datenbank.

DDL

Siehe Datenbankdefinitionssprache.

Deep-Ensemble

Mehrere Deep-Learning-Modelle zur Vorhersage kombinieren. Sie können Deep-Ensembles verwenden, um eine genauere Vorhersage zu erhalten oder um die Unsicherheit von Vorhersagen abzuschätzen.

Deep Learning

Ein ML-Teilbereich, der mehrere Schichten künstlicher neuronaler Netzwerke verwendet, um die Zuordnung zwischen Eingabedaten und Zielvariablen von Interesse zu ermitteln.

Tiefgreifende Verteidigung

Ein Ansatz zur Informationssicherheit, bei dem eine Reihe von Sicherheitsmechanismen und -kontrollen sorgfältig in einem Computernetzwerk verteilt werden, um die Vertraulichkeit, Integrität und Verfügbarkeit des Netzwerks und der darin enthaltenen Daten zu schützen. Wenn Sie diese Strategie anwenden AWS, fügen Sie mehrere Steuerelemente auf verschiedenen Ebenen der AWS Organizations Struktur hinzu, um die Ressourcen zu schützen. Ein umfassender Verteidigungsansatz könnte beispielsweise Multi-Faktor-Authentifizierung, Netzwerksegmentierung und Verschlüsselung kombinieren.

delegierter Administrator

In kann ein kompatibler Dienst ein AWS Mitgliedskonto registrieren AWS Organizations, um die Konten der Organisation zu verwalten und die Berechtigungen für diesen Dienst zu verwalten. Dieses Konto wird als delegierter Administrator für diesen Service bezeichnet. Weitere Informationen und eine Liste kompatibler Services finden Sie unter Services, die mit AWS Organizations funktionieren in der AWS Organizations -Dokumentation.

Einsatz

Der Prozess, bei dem eine Anwendung, neue Feature oder Codekorrekturen in der Zielumgebung verfügbar gemacht werden. Die Bereitstellung umfasst das Implementieren von Änderungen an einer Codebasis und das anschließende Erstellen und Ausführen dieser Codebasis in den Anwendungsumgebungen.

Entwicklungsumgebung

Siehe Umgebung.

Detektivische Kontrolle

Eine Sicherheitskontrolle, die darauf ausgelegt ist, ein Ereignis zu erkennen, zu protokollieren und zu warnen, nachdem ein Ereignis eingetreten ist. Diese Kontrollen stellen eine zweite Verteidigungslinie dar und warnen Sie vor Sicherheitsereignissen, bei denen die vorhandenen

präventiven Kontrollen umgangen wurden. Weitere Informationen finden Sie unter Detektivische Kontrolle in Implementierung von Sicherheitskontrollen in AWS.

Abbildung des Wertstroms in der Entwicklung (DVSM)

Ein Prozess zur Identifizierung und Priorisierung von Einschränkungen, die sich negativ auf Geschwindigkeit und Qualität im Lebenszyklus der Softwareentwicklung auswirken. DVSM erweitert den Prozess der Wertstromanalyse, der ursprünglich für Lean-Manufacturing-Praktiken konzipiert wurde. Es konzentriert sich auf die Schritte und Teams, die erforderlich sind, um durch den Softwareentwicklungsprozess Mehrwert zu schaffen und zu steigern.

digitaler Zwilling

Eine virtuelle Darstellung eines realen Systems, z. B. eines Gebäudes, einer Fabrik, einer Industrieanlage oder einer Produktionslinie. Digitale Zwillinge unterstützen vorausschauende Wartung, Fernüberwachung und Produktionsoptimierung.

Maßtabelle

In einem Sternschema eine kleinere Tabelle, die Datenattribute zu quantitativen Daten in einer Faktentabelle enthält. Bei Attributen von Dimensionstabellen handelt es sich in der Regel um Textfelder oder diskrete Zahlen, die sich wie Text verhalten. Diese Attribute werden häufig zum Einschränken von Abfragen, zum Filtern und zur Kennzeichnung von Ergebnismengen verwendet.

Katastrophe

Ein Ereignis, das verhindert, dass ein Workload oder ein System seine Geschäftsziele an seinem primären Einsatzort erfüllt. Diese Ereignisse können Naturkatastrophen, technische Ausfälle oder das Ergebnis menschlichen Handelns sein, z. B. unbeabsichtigte Fehlkonfigurationen oder ein Malware-Angriff.

Disaster Recovery (DR)

Die Strategie und der Prozess, die Sie zur Minimierung von Ausfallzeiten und Datenverlusten aufgrund einer Katastrophe anwenden. Weitere Informationen finden Sie unter Disaster Recovery von Workloads unter AWS: Wiederherstellung in der Cloud im AWS Well-Architected Framework.

DML

Siehe Sprache zur Datenbankmanipulation.

Domainorientiertes Design

Ein Ansatz zur Entwicklung eines komplexen Softwaresystems, bei dem seine Komponenten mit sich entwickelnden Domains oder Kerngeschäftszielen verknüpft werden, denen jede

Komponente dient. Dieses Konzept wurde von Eric Evans in seinem Buch Domain-Driven Design: Tackling Complexity in the Heart of Software (Boston: Addison-Wesley Professional, 2003) vorgestellt. Informationen darüber, wie Sie domänengesteuertes Design mit dem Strangler-Fig-Muster verwenden können, finden Sie unter Schrittweise Modernisierung älterer Microsoft ASP.NET (ASMX) -Webservices mithilfe von Containern und Amazon API Gateway.

DR

Siehe Disaster Recovery.

Erkennung von Driften

Verfolgung von Abweichungen von einer Basiskonfiguration Sie können es beispielsweise verwenden, AWS CloudFormation um Abweichungen bei den Systemressourcen zu erkennen, oder Sie können AWS Control Tower damit Änderungen in Ihrer landing zone erkennen, die sich auf die Einhaltung von Governance-Anforderungen auswirken könnten.

DVSM

Siehe Abbildung der Wertströme in der Entwicklung.

E

EDA

Siehe explorative Datenanalyse.

EDI

Siehe elektronischer Datenaustausch.

Edge-Computing

Die Technologie, die die Rechenleistung für intelligente Geräte an den Rändern eines IoT-Netzwerks erhöht. Im Vergleich zu Cloud Computing kann Edge Computing die Kommunikationslatenz reduzieren und die Reaktionszeit verbessern.

elektronischer Datenaustausch (EDI)

Der automatisierte Austausch von Geschäftsdokumenten zwischen Organisationen. Weitere Informationen finden Sie unter Was ist elektronischer Datenaustausch.

Verschlüsselung

Ein Rechenprozess, der Klartextdaten, die für Menschen lesbar sind, in Chiffretext umwandelt.

Verschlüsselungsschlüssel

Eine kryptografische Zeichenfolge aus zufälligen Bits, die von einem Verschlüsselungsalgorithmus generiert wird. Schlüssel können unterschiedlich lang sein, und jeder Schlüssel ist so konzipiert, dass er unvorhersehbar und einzigartig ist.

Endianismus

Die Reihenfolge, in der Bytes im Computerspeicher gespeichert werden. Big-endian Systeme speichern das höchstwertige Byte zuerst. Little-endian Systeme speichern das niedrigstwertige Byte zuerst.

Endpunkt

Siehe Service-Endpunkt.

Endpunkt-Services

Ein Service, den Sie in einer Virtual Private Cloud (VPC) hosten können, um ihn mit anderen Benutzern zu teilen. Sie können einen Endpunktdienst mit anderen AWS-Konten oder AWS Identity and Access Management (IAM AWS PrivateLink -) Prinzipalen erstellen und diesen Berechtigungen gewähren. Diese Konten oder Prinzipale können sich privat mit Ihrem Endpunktservice verbinden, indem sie Schnittstellen-VPC-Endpunkte erstellen. Weitere Informationen finden Sie unter Einen Endpunkt-Service erstellen in der Amazon Virtual Private Cloud (Amazon VPC)-Dokumentation.

Unternehmensressourcenplanung (ERP)

Ein System, das wichtige Geschäftsprozesse (wie Buchhaltung, MES und Projektmanagement) für ein Unternehmen automatisiert und verwaltet.

Envelope-Verschlüsselung

Der Prozess der Verschlüsselung eines Verschlüsselungsschlüssels mit einem anderen Verschlüsselungsschlüssel. Weitere Informationen finden Sie unter Envelope-Verschlüsselung in der AWS Key Management Service (AWS KMS) -Dokumentation.

Umgebung

Eine Instance einer laufenden Anwendung. Die folgenden Arten von Umgebungen sind beim Cloud-Computing üblich:

- Entwicklungsumgebung – Eine Instance einer laufenden Anwendung, die nur dem Kernteam zur Verfügung steht, das für die Wartung der Anwendung verantwortlich ist. Entwicklungsumgebungen werden verwendet, um Änderungen zu testen, bevor sie in höhere

Umgebungen übertragen werden. Diese Art von Umgebung wird manchmal als Testumgebung bezeichnet.

- Niedrigere Umgebungen – Alle Entwicklungsumgebungen für eine Anwendung, z. B. solche, die für erste Builds und Tests verwendet wurden.
- Produktionsumgebung – Eine Instance einer laufenden Anwendung, auf die Endbenutzer zugreifen können. In einer CI/CD Pipeline ist die Produktionsumgebung die letzte Bereitstellungsumgebung.
- Höhere Umgebungen – Alle Umgebungen, auf die auch andere Benutzer als das Kernentwicklungsteam zugreifen können. Dies kann eine Produktionsumgebung, Vorproduktionsumgebungen und Umgebungen für Benutzerakzeptanztests umfassen.

Epics

In der agilen Methodik sind dies funktionale Kategorien, die Ihnen helfen, Ihre Arbeit zu organisieren und zu priorisieren. Epics bieten eine allgemeine Beschreibung der Anforderungen und Implementierungsaufgaben. Zu den Sicherheitsepen AWS von CAF gehören beispielsweise Identitäts- und Zugriffsmanagement, Detektivkontrollen, Infrastruktursicherheit, Datenschutz und Reaktion auf Vorfälle. Weitere Informationen zu Epics in der AWS -Migrationsstrategie finden Sie im Leitfaden zur Programm-Implementierung.

ERP

Siehe Enterprise Resource Planning.

Explorative Datenanalyse (EDA)

Der Prozess der Analyse eines Datensatzes, um seine Hauptmerkmale zu verstehen. Sie sammeln oder aggregieren Daten und führen dann erste Untersuchungen durch, um Muster zu finden, Anomalien zu erkennen und Annahmen zu überprüfen. EDA wird durchgeführt, indem zusammenfassende Statistiken berechnet und Datenvisualisierungen erstellt werden.

F

Faktentabelle

Die zentrale Tabelle in einem Sternschema. Sie speichert quantitative Daten über den Geschäftsbetrieb. In der Regel enthält eine Faktentabelle zwei Arten von Spalten: Spalten, die Kennzahlen enthalten, und Spalten, die einen Fremdschlüssel für eine Dimensionstabelle enthalten.

schnell scheitern

Eine Philosophie, die häufige und inkrementelle Tests verwendet, um den Entwicklungslebenszyklus zu verkürzen. Dies ist ein wichtiger Bestandteil eines agilen Ansatzes.

Grenze zur Fehlerisolierung

Dabei handelt es sich um eine Grenze AWS Cloud, z. B. eine Availability Zone AWS-Region, eine Steuerungsebene oder eine Datenebene, die die Auswirkungen eines Fehlers begrenzt und die Widerstandsfähigkeit von Workloads verbessert. Weitere Informationen finden Sie unter Grenzen zur AWS Fehlerisolierung.

Feature-Zweig

Siehe Zweig.

Features

Die Eingabedaten, die Sie verwenden, um eine Vorhersage zu treffen. In einem Fertigungskontext könnten Feature beispielsweise Bilder sein, die regelmäßig von der Fertigungslinie aus aufgenommen werden.

Bedeutung der Feature

Wie wichtig ein Feature für die Vorhersagen eines Modells ist. Dies wird in der Regel als numerischer Wert ausgedrückt, der mit verschiedenen Techniken wie Shapley Additive Explanations (SHAP) und integrierten Gradienten berechnet werden kann. Weitere Informationen finden Sie unter Interpretierbarkeit von Modellen für maschinelles Lernen mit AWS.

Featuretransformation

Daten für den ML-Prozess optimieren, einschließlich der Anreicherung von Daten mit zusätzlichen Quellen, der Skalierung von Werten oder der Extraktion mehrerer Informationssätze aus einem einzigen Datenfeld. Das ermöglicht dem ML-Modell, von den Daten profitieren. Wenn Sie beispielsweise das Datum „27.05.2021 00:15:37“ in „2021“, „Mai“, „Donnerstag“ und „15“ aufschlüsseln, können Sie dem Lernalgorithmus helfen, nuancierte Muster zu erlernen, die mit verschiedenen Datenkomponenten verknüpft sind.

Eingabeaufforderung mit wenigen Klicks

Bereitstellung einer kleinen Anzahl von Beispielen, die die Aufgabe und das gewünschte Ergebnis veranschaulichen, bevor das LLM aufgefordert wird, eine ähnliche Aufgabe auszuführen. Bei dieser Technik handelt es sich um eine Anwendung des kontextbezogenen Lernens,

bei der Modelle anhand von Beispielen (Aufnahmen) lernen, die in Eingabeaufforderungen eingebettet sind. Few-shot Eingabeaufforderungen können bei Aufgaben, die spezifische Formatierungs-, Argumentations- oder Fachkenntnisse erfordern, effektiv sein. Siehe auch Zero-Shot-Eingabeaufforderung.

FGAC

Siehe detaillierte Zugriffskontrolle.

Feinkörnige Zugriffskontrolle (FGAC)

Die Verwendung mehrerer Bedingungen, um eine Zugriffsanfrage zuzulassen oder abzulehnen.

Flash-Cut-Migration

Eine Datenbankmigrationsmethode, bei der eine kontinuierliche Datenreplikation durch Erfassung von Änderungsdaten verwendet wird, um Daten in kürzester Zeit zu migrieren, anstatt einen schrittweisen Ansatz zu verwenden. Ziel ist es, Ausfallzeiten auf ein Minimum zu beschränken.

FM

Siehe Fundamentmodell.

Fundamentmodell (FM)

Ein großes neuronales Deep-Learning-Netzwerk, das mit riesigen Datensätzen generalisierter und unbeschrifteter Daten trainiert wurde. FMs sind in der Lage, eine Vielzahl allgemeiner Aufgaben zu erfüllen, z. B. Sprache zu verstehen, Text und Bilder zu generieren und Konversationen in natürlicher Sprache zu führen. Weitere Informationen finden Sie unter Was sind Foundation-Modelle.

FM-Gateway

Ein zentraler Vermittler, der den Zugriff auf Basismodelle kontrolliert und normalisiert. Wird auch als LLM-Gateway bezeichnet.

G

Generative KI

Eine Untergruppe von KI-Modellen, die mit großen Datenmengen trainiert wurden und mithilfe einer einfachen Textaufforderung neue Inhalte und Artefakte wie Bilder, Videos, Text und Audio erstellen können. Weitere Informationen finden Sie unter Was ist Generative KI.

Geoblocking

Siehe geografische Einschränkungen.

Geografische Einschränkungen (Geoblocking)

Bei Amazon eine Option CloudFront, um zu verhindern, dass Benutzer in bestimmten Ländern auf Inhaltsverteilungen zugreifen. Sie können eine Zulassungsliste oder eine Sperrliste verwenden, um zugelassene und gesperrte Länder anzugeben. Weitere Informationen finden Sie in der Dokumentation unter Beschränkung der geografischen Verteilung Ihrer Inhalte. CloudFront

Gitflow-Workflow

Ein Ansatz, bei dem niedrigere und höhere Umgebungen unterschiedliche Zweige in einem Quellcode-Repository verwenden. Der Gitflow-Workflow gilt als veraltet, und der Trunk-basierte Workflow ist der moderne, bevorzugte Ansatz.

goldenes Bild

Ein Snapshot eines Systems oder einer Software, der als Vorlage für die Bereitstellung neuer Instanzen dieses Systems oder dieser Software verwendet wird. In der Fertigung kann ein Golden Image beispielsweise zur Bereitstellung von Software auf mehreren Geräten verwendet werden und trägt so zur Verbesserung der Geschwindigkeit, Skalierbarkeit und Produktivität bei der Geräteherstellung bei.

Greenfield-Strategie

Das Fehlen vorhandener Infrastruktur in einer neuen Umgebung. Bei der Einführung einer Neuausrichtung einer Systemarchitektur können Sie alle neuen Technologien ohne Einschränkung der Kompatibilität mit der vorhandenen Infrastruktur auswählen, auch bekannt als Brownfield. Wenn Sie die bestehende Infrastruktur erweitern, könnten Sie Brownfield- und Greenfield-Strategien mischen.

Integritätsschutz

Eine allgemeine Regel, die dabei hilft, Ressourcen, Richtlinien und die Einhaltung von Vorschriften in allen Organisationseinheiten (OUs) zu regeln. Präventiver Integritätsschutz setzt Richtlinien durch, um die Einhaltung von Standards zu gewährleisten. Sie werden mithilfe von Service-Kontrollrichtlinien und IAM-Berechtigungsgrenzen implementiert. Detektivischer Integritätsschutz erkennt Richtlinienverstöße und Compliance-Probleme und generiert Warnmeldungen zur Abhilfe. Sie werden mithilfe von AWS Config, AWS Security Hub CSPM, Amazon GuardDuty AWS Trusted Advisor, Amazon Inspector und benutzerdefinierten AWS Lambda Prüfungen implementiert.

Leitplanken (KI)

Sicherheitsmechanismen, die Eingaben und Ausgaben von Agenten filtern, validieren und einschränken, um ein verantwortungsbewusstes und sicheres Verhalten der KI zu gewährleisten.

H

HEKTAR

Siehe Hochverfügbarkeit.

Heterogene Datenbankmigration

Migrieren Sie Ihre Quelldatenbank in eine Zieldatenbank, die eine andere Datenbank-Engine verwendet (z. B. Oracle zu Amazon Aurora). Eine heterogene Migration ist in der Regel Teil einer Neuarchitektur, und die Konvertierung des Schemas kann eine komplexe Aufgabe sein. AWS bietet AWS SCT, welches bei Schemakonvertierungen hilft.

hohe Verfügbarkeit (HA)

Die Fähigkeit eines Workloads, im Falle von Herausforderungen oder Katastrophen kontinuierlich und ohne Eingreifen zu arbeiten. HA-Systeme sind so konzipiert, dass sie automatisch ein Failover durchführen, gleichbleibend hohe Leistung bieten und unterschiedliche Lasten und Ausfälle mit minimalen Leistungseinbußen bewältigen.

historische Modernisierung

Ein Ansatz zur Modernisierung und Aufrüstung von Betriebstechnologiesystemen (OT), um den Bedürfnissen der Fertigungsindustrie besser gerecht zu werden. Ein Historian ist eine Art von Datenbank, die verwendet wird, um Daten aus verschiedenen Quellen in einer Fabrik zu sammeln und zu speichern.

Holdout-Daten

Ein Teil historischer, beschrifteter Daten, der aus einem Datensatz zurückgehalten wird, der zum Trainieren eines Modells für maschinelles Lernen verwendet wird. Sie können Holdout-Daten verwenden, um die Modellleistung zu bewerten, indem Sie die Modellvorhersagen mit den Holdout-Daten vergleichen.

Der Mensch im Kreis (HiTL)

Ein Workflow-Muster, bei dem die Ausführung von Agenten an kritischen Entscheidungspunkten unterbrochen wird, um von einem Mitarbeiter geprüft und genehmigt zu werden.

Homogene Datenbankmigration

Migrieren Sie Ihre Quelldatenbank zu einer Zieldatenbank, die dieselbe Datenbank-Engine verwendet (z. B. Microsoft SQL Server zu Amazon RDS für SQL Server). Eine homogene Migration ist in der Regel Teil eines Hostwechsels oder eines Plattformwechsels. Sie können native Datenbankserviceprogramme verwenden, um das Schema zu migrieren.

heiße Daten

Daten, auf die häufig zugegriffen wird, z. B. Echtzeitdaten oder aktuelle Translationsdaten. Für diese Daten ist in der Regel eine leistungsstarke Speicherebene oder -klasse erforderlich, um schnelle Abfrageantworten zu ermöglichen.

Hotfix

Eine dringende Lösung für ein kritisches Problem in einer Produktionsumgebung. Aufgrund seiner Dringlichkeit wird ein Hotfix normalerweise außerhalb des typischen DevOps Release-Workflows erstellt.

Hypercare-Phase

Unmittelbar nach dem Cutover, der Zeitraum, in dem ein Migrationsteam die migrierten Anwendungen in der Cloud verwaltet und überwacht, um etwaige Probleme zu beheben. In der Regel dauert dieser Zeitraum 1–4 Tage. Am Ende der Hypercare-Phase überträgt das Migrationsteam in der Regel die Verantwortung für die Anwendungen an das Cloud-Betriebsteam.

I

IaC

Sehen Sie sich Infrastruktur als Code an.

Identitätsbasierte Richtlinie

Eine Richtlinie, die einem oder mehreren IAM-Prinzipalen zugeordnet ist und deren Berechtigungen innerhalb der AWS Cloud Umgebung definiert.

Leerlaufanwendung

Eine Anwendung mit einer durchschnittlichen CPU- und Arbeitsspeicherauslastung zwischen 5 und 20 Prozent über einen Zeitraum von 90 Tagen. In einem Migrationsprojekt ist es üblich, diese Anwendungen außer Betrieb zu nehmen oder sie On-Premises beizubehalten.

IIoT

Siehe Industrielles Internet der Dinge.

unveränderliche Infrastruktur

Ein Modell, das eine neue Infrastruktur für Produktionsworkloads bereitstellt, anstatt die bestehende Infrastruktur zu aktualisieren, zu patchen oder zu modifizieren. Unveränderliche Infrastrukturen sind von Natur aus konsistenter, zuverlässiger und vorhersehbarer als veränderliche Infrastrukturen. Weitere Informationen finden Sie in der Best Practice Deploy using immutable infrastructure im Framework. AWS Well-Architected

Eingehende (ingress) VPC

In einer Architektur AWS mit mehreren Konten ist dies eine VPC, die Netzwerkverbindungen von außerhalb einer Anwendung akzeptiert, überprüft und weiterleitet. Die AWS -Referenzarchitektur für die Sicherheit empfiehlt, Ihr Netzwerkkonto mit eingehenden und ausgehenden VPCs und Inspektions-VPCs einzurichten, um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet zu schützen.

Inkrementelle Migration

Eine Cutover-Strategie, bei der Sie Ihre Anwendung in kleinen Teilen migrieren, anstatt eine einziges vollständiges Cutover durchzuführen. Beispielsweise könnten Sie zunächst nur einige Microservices oder Benutzer auf das neue System umstellen. Nachdem Sie sich vergewissert haben, dass alles ordnungsgemäß funktioniert, können Sie weitere Microservices oder Benutzer schrittweise verschieben, bis Sie Ihr Legacy-System außer Betrieb nehmen können. Diese Strategie reduziert die mit großen Migrationen verbundenen Risiken.

Industrie 4.0

Ein Begriff, der 2016 von Klaus Schwab eingeführt wurde und sich auf die Modernisierung von Fertigungsprozessen durch Fortschritte in den Bereichen Konnektivität, Echtzeitdaten, Automatisierung, Analytik und bezieht. AI/ML

Infrastruktur

Alle Ressourcen und Komponenten, die in der Umgebung einer Anwendung enthalten sind.

Infrastructure as Code (IaC)

Der Prozess der Bereitstellung und Verwaltung der Infrastruktur einer Anwendung mithilfe einer Reihe von Konfigurationsdateien. IaC soll Ihnen helfen, das Infrastrukturmanagement zu zentralisieren, Ressourcen zu standardisieren und schnell zu skalieren, sodass neue Umgebungen wiederholbar, zuverlässig und konsistent sind.

Industrielles Internet der Dinge (IIoT)

Einsatz von mit dem Internet verbundenen Sensoren und Geräten in Industriesektoren wie Fertigung, Energie, Automobilindustrie, Gesundheitswesen, Biowissenschaften und Landwirtschaft. Mehr Informationen finden Sie unter Aufbau einer digitalen Transformationsstrategie für das industrielle Internet der Dinge (IIoT).

Inspektions-VPC

In einer Architektur AWS mit mehreren Konten eine zentralisierte VPC, die Inspektionen des Netzwerkverkehrs zwischen VPCs (in derselben oder unterschiedlichen AWS-Regionen), dem Internet und lokalen Netzwerken verwaltet. Die AWS -Referenzarchitektur für die Sicherheit empfiehlt, Ihr Netzwerkkonto mit eingehenden und ausgehenden VPCs und Inspektions-VPCs einzurichten, um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet zu schützen.

Internet of Things (IoT)

Das Netzwerk verbundener physischer Objekte mit eingebetteten Sensoren oder Prozessoren, das über das Internet oder über ein lokales Kommunikationsnetzwerk mit anderen Geräten und Systemen kommuniziert. Weitere Informationen finden Sie unter Was ist IoT?

Interpretierbarkeit

Ein Merkmal eines Modells für Machine Learning, das beschreibt, inwieweit ein Mensch verstehen kann, wie die Vorhersagen des Modells von seinen Eingaben abhängen. Weitere Informationen finden Sie unter Interpretierbarkeit von Modellen für maschinelles Lernen mit. AWS

IoT

Siehe Internet der Dinge.

IT information library (ITIL, IT-Informationsbibliothek)

Eine Reihe von bewährten Methoden für die Bereitstellung von IT-Services und die Abstimmung dieser Services auf die Geschäftsanforderungen. ITIL bietet die Grundlage für ITSM.

T service management (ITSM, IT-Servicemanagement)

Aktivitäten im Zusammenhang mit der Gestaltung, Implementierung, Verwaltung und Unterstützung von IT-Services für eine Organisation. Informationen zur Integration von Cloud-Vorgängen mit ITSM-Tools finden Sie im Leitfaden zur Betriebsintegration.

BIS

Siehe IT-Informationsbibliothek.

ITSM

Siehe IT-Servicemanagement.

L

Labelbasierte Zugangskontrolle (LBAC)

Eine Implementierung der Mandatory Access Control (MAC), bei der den Benutzern und den Daten selbst jeweils explizit ein Sicherheitslabelwert zugewiesen wird. Die Schnittmenge zwischen der Benutzersicherheitsbeschriftung und der Datensicherheitsbeschriftung bestimmt, welche Zeilen und Spalten für den Benutzer sichtbar sind.

Landing Zone

Eine landing zone ist eine gut strukturierte AWS Umgebung mit mehreren Konten, die skalierbar und sicher ist. Dies ist ein Ausgangspunkt, von dem aus Ihre Organisationen Workloads und Anwendungen schnell und mit Vertrauen in ihre Sicherheits- und Infrastrukturumgebung starten und bereitstellen können. Weitere Informationen zu Landing Zones finden Sie unter Einrichtung einer sicheren und skalierbaren AWS -Umgebung mit mehreren Konten..

großes Sprachmodell (LLM)

Ein Deep-Learning-KI-Modell, das anhand einer riesigen Datenmenge vorab trainiert wurde. Ein LLM kann mehrere Aufgaben ausführen, z. B. Fragen beantworten, Dokumente zusammenfassen, Text in andere Sprachen übersetzen und Sätze vervollständigen. Weitere Informationen finden Sie unter Was sind LLMs.

Große Migration

Eine Migration von 300 oder mehr Servern.

LBAC

Siehe Labelbasierte Zugriffskontrolle.

Geringste Berechtigung

Die bewährte Sicherheitsmethode, bei der nur die für die Durchführung einer Aufgabe erforderlichen Mindestberechtigungen erteilt werden. Weitere Informationen finden Sie unter Geringste Berechtigungen anwenden in der IAM-Dokumentation.

Lift and Shift

Siehe 7 Rs.

Little-Endian-System

Ein System, welches das niedrigwertigste Byte zuerst speichert. Siehe auch Endianness.

LLM

Siehe großes Sprachmodell.

Niedrigere Umgebungen

Siehe Umgebung.

M

Machine Learning (ML)

Eine Art künstlicher Intelligenz, die Algorithmen und Techniken zur Mustererkennung und zum Lernen verwendet. ML analysiert aufgezeichnete Daten, wie z. B. Daten aus dem Internet der Dinge (IoT), und lernt daraus, um ein statistisches Modell auf der Grundlage von Mustern zu erstellen. Weitere Informationen finden Sie unter Machine Learning.

Hauptzweig

Siehe Filiale.

Malware

Software, die entwickelt wurde, um die Computersicherheit oder den Datenschutz zu gefährden. Malware kann Computersysteme stören, vertrauliche Informationen durchsickern lassen oder sich unbefugten Zugriff verschaffen. Beispiele für Malware sind Viren, Würmer, Ransomware, Trojaner, Spyware und Keylogger.

verwaltete Dienste

AWS-Services für die die Infrastrukturebene, das Betriebssystem und die Plattformen AWS betrieben werden, und Sie greifen auf die Endgeräte zu, um Daten zu speichern und abzurufen. Amazon Simple Storage Service (Amazon S3) und Amazon DynamoDB sind Beispiele für Managed Services. Diese werden auch als abstrakte Dienste bezeichnet.

Manufacturing Execution System (MES)

Ein Softwaresystem zur Verfolgung, Überwachung, Dokumentation und Steuerung von Produktionsprozessen, bei denen Rohstoffe in der Fertigung zu fertigen Produkten umgewandelt werden.

MAP

Siehe Migration Acceleration Program.

MCP

Siehe Model Context Protocol.

Model Context Protocol (MCP)

Ein zustandsloses Protokoll für die Kommunikation zwischen Agenten und Tool.

MCP-Server

Ein Dienst, der ein oder mehrere Tools über das Model Context Protocol verfügbar macht.

Mechanismus

Ein vollständiger Prozess, bei dem Sie ein Tool erstellen, die Akzeptanz des Tools vorantreiben und anschließend die Ergebnisse überprüfen, um Anpassungen vorzunehmen. Ein Mechanismus ist ein Zyklus, der sich im Laufe seiner Tätigkeit selbst verstärkt und verbessert. Weitere Informationen finden Sie unter Mechanismen im AWS Well-Architected Framework erstellen.

Mitgliedskonto

Alle AWS-Konten außer dem Verwaltungskonto, die Teil einer Organisation in sind AWS Organizations. Ein Konto kann jeweils nur Mitglied einer Organisation sein.

MES

Siehe Manufacturing Execution System.

Message Queuing-Telemetrietransport (MQTT)

Ein leichtes, auf dem publish/subscribeMuster basierendes M2M-Kommunikationsprotokoll (Machine-to-Machine) für IoT-Geräte mit beschränkten Ressourcen.

Microservice

Ein kleiner, unabhängiger Service, der über klar definierte APIs kommuniziert und in der Regel kleinen, eigenständigen Teams gehört. Ein Versicherungssystem kann beispielsweise Microservices beinhalten, die Geschäftsfunktionen wie Vertrieb oder Marketing oder Subdomains wie Einkauf, Schadenersatz oder Analytik zugeordnet sind. Zu den Vorteilen von Microservices gehören Agilität, flexible Skalierung, einfache Bereitstellung, wiederverwendbarer Code und Ausfallsicherheit. Weitere Informationen finden Sie unter Integration von Microservices mithilfe serverloser Dienste. AWS

Microservices-Architekturen

Ein Ansatz zur Erstellung einer Anwendung mit unabhängigen Komponenten, die jeden Anwendungsprozess als Microservice ausführen. Diese Microservices kommunizieren über eine klar definierte Schnittstelle mithilfe einfacher APIs. Jeder Microservice in dieser Architektur kann aktualisiert, bereitgestellt und skaliert werden, um den Bedarf an bestimmten Funktionen einer Anwendung zu decken. Weitere Informationen finden Sie unter Implementieren von Microservices auf. AWS

Migration Acceleration Program (MAP)

Ein AWS Programm, das Beratung, Unterstützung, Schulungen und Services bietet, um Unternehmen dabei zu unterstützen, eine solide betriebliche Grundlage für die Umstellung auf die Cloud zu schaffen und die anfänglichen Kosten von Migrationen auszugleichen. MAP umfasst eine Migrationsmethode für die methodische Durchführung von Legacy-Migrationen sowie eine Reihe von Tools zur Automatisierung und Beschleunigung gängiger Migrationsszenarien.

Migration in großem Maßstab

Der Prozess, bei dem der Großteil des Anwendungsportfolios in Wellen in die Cloud verlagert wird, wobei in jeder Welle mehr Anwendungen schneller migriert werden. In dieser Phase werden die bewährten Verfahren und Erkenntnisse aus den früheren Phasen zur Implementierung einer Migrationsfabrik von Teams, Tools und Prozessen zur Optimierung der Migration von Workloads durch Automatisierung und agile Bereitstellung verwendet. Dies ist die dritte Phase der AWS - Migrationsstrategie.

Migrationsfabrik

Cross-functional Teams, die die Migration von Workloads durch automatisierte, agile Ansätze optimieren. Zu den Teams von Migration Factory gehören in der Regel Betriebsanalysten und Eigentümer, Migrationsingenieure, Entwickler und DevOps Experten, die in Sprints arbeiten. Zwischen 20 und 50 Prozent eines Unternehmensanwendungsportfolios bestehen aus sich wiederholenden Mustern, die durch einen Fabrik-Ansatz optimiert werden können. Weitere Informationen finden Sie in Diskussion über Migrationsfabriken und den Leitfaden zur Cloud-Migration-Fabrik in diesem Inhaltssatz.

Migrationsmetadaten

Die Informationen über die Anwendung und den Server, die für den Abschluss der Migration benötigt werden. Für jedes Migrationsmuster ist ein anderer Satz von Migrationsmetadaten erforderlich. Beispiele für Migrationsmetadaten sind das Zielsubnetz, die Sicherheitsgruppe und AWS das Konto.

Migrationsmuster

Eine wiederholbare Migrationsaufgabe, in der die Migrationsstrategie, das Migrationsziel und die verwendete Migrationsanwendung oder der verwendete Migrationsservice detailliert beschrieben werden. Beispiel: Rehost-Migration zu Amazon EC2 mit AWS Application Migration Service.

Migration Portfolio Assessment (MPA)

Ein Online-Tool, das Informationen zur Validierung des Geschäftsszenarios für die Migration auf das bereitstellt. AWS Cloud MPA bietet eine detaillierte Portfoliobewertung (richtige Servergröße, Preisgestaltung, Gesamtbetriebskostenanalyse, Migrationskostenanalyse) sowie Migrationsplanung (Anwendungsdatenanalyse und Datenerfassung, Anwendungsgruppierung, Migrationspriorisierung und Wellenplanung). Das MPA-Tool (Anmeldung erforderlich) steht allen AWS Beratern und APN-Partnerberatern kostenlos zur Verfügung.

Migration Readiness Assessment (MRA)

Der Prozess, bei dem mithilfe des AWS CAF Erkenntnisse über den Cloud-Bereitschaftsstatus eines Unternehmens gewonnen, Stärken und Schwächen identifiziert und ein Aktionsplan zur Schließung festgestellter Lücken erstellt wird. Weitere Informationen finden Sie im Benutzerhandbuch für Migration Readiness. MRA ist die erste Phase der AWS - Migrationsstrategie.

Migrationsstrategie

Der Ansatz, der verwendet wurde, um einen Workload auf den AWS Cloud zu migrieren. Weitere Informationen finden Sie im Eintrag 7 Rs in diesem Glossar und unter Mobilisieren Sie Ihr Unternehmen, um umfangreiche Migrationen zu beschleunigen.

ML

Siehe maschinelles Lernen.

Modernisierung

Umwandlung einer veralteten (veralteten oder monolithischen) Anwendung und ihrer Infrastruktur in ein agiles, elastisches und hochverfügbares System in der Cloud, um Kosten zu senken, die Effizienz zu steigern und Innovationen zu nutzen. Weitere Informationen finden Sie unter Strategie zur Modernisierung von Anwendungen in der AWS Cloud.

Bewertung der Modernisierungsfähigkeit

Eine Bewertung, anhand derer festgestellt werden kann, ob die Anwendungen einer Organisation für die Modernisierung bereit sind, Vorteile, Risiken und Abhängigkeiten identifiziert und ermittelt

wird, wie gut die Organisation den zukünftigen Status dieser Anwendungen unterstützen kann. Das Ergebnis der Bewertung ist eine Vorlage der Zielarchitektur, eine Roadmap, in der die Entwicklungsphasen und Meilensteine des Modernisierungsprozesses detailliert beschrieben werden, sowie ein Aktionsplan zur Behebung festgestellter Lücken. Weitere Informationen finden Sie unter Evaluierung der Modernisierungsbereitschaft von Anwendungen in der AWS Cloud.

Monolithische Anwendungen (Monolithen)

Anwendungen, die als ein einziger Service mit eng gekoppelten Prozessen ausgeführt werden. Monolithische Anwendungen haben verschiedene Nachteile. Wenn ein Anwendungs-Feature stark nachgefragt wird, muss die gesamte Architektur skaliert werden. Das Hinzufügen oder Verbessern der Feature einer monolithischen Anwendung wird ebenfalls komplexer, wenn die Codebasis wächst. Um diese Probleme zu beheben, können Sie eine Microservices-Architektur verwenden. Weitere Informationen finden Sie unter Zerlegen von Monolithen in Microservices.

MPA

Siehe Bewertung des Migrationsportfolios.

MQTT

Siehe Message Queuing-Telemetrietransport.

Mehrklassen-Klassifizierung

Ein Prozess, der dabei hilft, Vorhersagen für mehrere Klassen zu generieren (wobei eines von mehr als zwei Ergebnissen vorhergesagt wird). Ein ML-Modell könnte beispielsweise fragen: „Ist dieses Produkt ein Buch, ein Auto oder ein Telefon?“ oder „Welche Kategorie von Produkten ist für diesen Kunden am interessantesten?“

veränderbare Infrastruktur

Ein Modell, das die bestehende Infrastruktur für Produktionsworkloads aktualisiert und modifiziert. Um die Konsistenz, Zuverlässigkeit und Vorhersagbarkeit zu verbessern, empfiehlt das AWS Well-Architected Framework die Verwendung einer unveränderlichen Infrastruktur als bewährte Methode.

O

OAC

Siehe Origin Access Control.

EICHE

Siehe Zugriffsidentität von Origin.

COM

Siehe organisatorisches Change-Management.

Offline-Migration

Eine Migrationsmethode, bei der der Quell-Workload während des Migrationsprozesses heruntergefahren wird. Diese Methode ist mit längeren Ausfallzeiten verbunden und wird in der Regel für kleine, unkritische Workloads verwendet.

OI

Siehe Betriebsintegration.

OLA

Siehe Vereinbarung auf operativer Ebene.

Online-Migration

Eine Migrationsmethode, bei der der Quell-Workload auf das Zielsystem kopiert wird, ohne offline genommen zu werden. Anwendungen, die mit dem Workload verbunden sind, können während der Migration weiterhin funktionieren. Diese Methode beinhaltet keine bis minimale Ausfallzeit und wird in der Regel für kritische Produktionsworkloads verwendet.

OPC-UA

Siehe Open Process Communications — Unified Architecture.

Offene Prozesskommunikation — Einheitliche Architektur (OPC-UA)

Ein Machine-to-Machine-Kommunikationsprotokoll (M2M) für die industrielle Automatisierung. OPC-UA bietet einen Interoperabilitätsstandard mit Datenverschlüsselungs-, Authentifizierungs- und Autorisierungsschemata.

Vereinbarung auf Betriebsebene (OLA)

Eine Vereinbarung, in der klargestellt wird, welche funktionalen IT-Gruppen sich gegenseitig versprechen zu liefern, um ein Service Level Agreement (SLA) zu unterstützen.

Überprüfung der Betriebsbereitschaft (ORR)

Eine Checkliste mit Fragen und zugehörigen bewährten Methoden, die Ihnen helfen, Vorfälle und mögliche Ausfälle zu verstehen, zu bewerten, zu verhindern oder deren Umfang zu reduzieren.

Weitere Informationen finden Sie unter Operational Readiness Readiness Reviews (ORR) im AWS Well-Architected Framework.

Betriebstechnologie (OT)

Hardware- und Softwaresysteme, die mit der physischen Umgebung zusammenarbeiten, um industrielle Abläufe, Ausrüstung und Infrastruktur zu steuern. In der Fertigung ist die Integration von OT- und Informationstechnologie (IT) -Systemen ein zentraler Schwerpunkt der Industrie 4.0-Transformationen.

Betriebsintegration (OI)

Der Prozess der Modernisierung von Abläufen in der Cloud, der Bereitschaftsplanung, Automatisierung und Integration umfasst. Weitere Informationen finden Sie im Leitfaden zur Betriebsintegration.

Organisationspfad

Ein Pfad, der von erstellt wird und in AWS CloudTrail dem alle Ereignisse für alle AWS-Konten in einer Organisation protokolliert werden. AWS Organizations Diese Spur wird in jedem AWS-Konto , der Teil der Organisation ist, erstellt und verfolgt die Aktivität in jedem Konto. Weitere Informationen finden Sie in der CloudTrail Dokumentation unter Einen Trail für eine Organisation erstellen.

Organisatorisches Veränderungsmanagement (OCM)

Ein Framework für das Management wichtiger, disruptiver Geschäftstransformationen aus Sicht der Mitarbeiter, der Kultur und der Führung. OCM hilft Organisationen dabei, sich auf neue Systeme und Strategien vorzubereiten und auf diese umzustellen, indem es die Akzeptanz von Veränderungen beschleunigt, Übergangsprobleme angeht und kulturelle und organisatorische Veränderungen vorantreibt. In der AWS Migrationsstrategie wird dieses Framework aufgrund der Geschwindigkeit des Wandels, der bei Projekten zur Cloud-Einführung erforderlich ist, als Mitarbeiterbeschleunigung bezeichnet. Weitere Informationen finden Sie im OCM-Handbuch.

Ursprungszugriffskontrolle (OAC)

In CloudFront, eine erweiterte Option zur Zugriffsbeschränkung, um Ihre Amazon Simple Storage Service (Amazon S3) -Inhalte zu sichern. OAC unterstützt alle S3-Buckets insgesamt AWS-Regionen, serverseitige Verschlüsselung mit AWS KMS (SSE-KMS) sowie dynamische PUT und `DELETE` Anfragen an den S3-Bucket.

Ursprungszugriffsidentität (OAI)

In CloudFront, eine Option zur Zugriffsbeschränkung, um Ihre Amazon S3 S3-Inhalte zu sichern. Wenn Sie OAI verwenden, CloudFront erstellt es einen Principal, mit dem sich Amazon S3 authentifizieren kann. Authentifizierte Principals können nur über eine bestimmte Distribution auf Inhalte in einem S3-Bucket zugreifen. CloudFront Siehe auch OAC, das eine detailliertere und verbesserte Zugriffskontrolle bietet.

ORR

Weitere Informationen finden Sie unter Überprüfung der Betriebsbereitschaft.

NICHT

Siehe Betriebstechnologie.

Ausgehende (egress) VPC

In einer Architektur AWS mit mehreren Konten eine VPC, die Netzwerkverbindungen verarbeitet, die von einer Anwendung aus initiiert werden. Die AWS -Referenzarchitektur für die Sicherheit empfiehlt, Ihr Netzwerkkonto mit eingehenden und ausgehenden VPCs und Inspektions-VPCs einzurichten, um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet zu schützen.

P

Berechtigungsgrenze

Eine IAM-Verwaltungsrichtlinie, die den IAM-Prinzipalen zugeordnet ist, um die maximalen Berechtigungen festzulegen, die der Benutzer oder die Rolle haben kann. Weitere Informationen finden Sie unter Berechtigungsgrenzen für IAM-Entitys in der IAM-Dokumentation.

persönlich identifizierbare Informationen (PII)

Informationen, die, wenn sie direkt betrachtet oder mit anderen verwandten Daten kombiniert werden, verwendet werden können, um vernünftige Rückschlüsse auf die Identität einer Person zu ziehen. Beispiele für personenbezogene Daten sind Namen, Adressen und Kontaktinformationen.

Personenbezogene Daten

Siehe persönlich identifizierbare Informationen.

Playbook

Eine Reihe vordefinierter Schritte, die die mit Migrationen verbundenen Aufgaben erfassen, z. B. die Bereitstellung zentraler Betriebsfunktionen in der Cloud. Ein Playbook kann die Form von Skripten, automatisierten Runbooks oder einer Zusammenfassung der Prozesse oder Schritte annehmen, die für den Betrieb Ihrer modernisierten Umgebung erforderlich sind.

PLC

Siehe programmierbare Logiksteuerung.

PLM

Siehe Produktlebenszyklusmanagement.

policy

Ein Objekt, das Berechtigungen definieren (siehe identitätsbasierte Richtlinie), Zugriffsbedingungen spezifizieren (siehe ressourcenbasierte Richtlinie) oder die maximalen Berechtigungen für alle Konten in einer Organisation definieren kann AWS Organizations (siehe Dienststeuerungsrichtlinie).

Polyglotte Beharrlichkeit

Unabhängige Auswahl der Datenspeichertechnologie eines Microservices auf der Grundlage von Datenzugriffsmustern und anderen Anforderungen. Wenn Ihre Microservices über dieselbe Datenspeichertechnologie verfügen, kann dies zu Implementierungsproblemen oder zu Leistungseinbußen führen. Microservices lassen sich leichter implementieren und erzielen eine bessere Leistung und Skalierbarkeit, wenn sie den Datenspeicher verwenden, der ihren Anforderungen am besten entspricht.

Portfoliobewertung

Ein Prozess, bei dem das Anwendungsportfolio ermittelt, analysiert und priorisiert wird, um die Migration zu planen. Weitere Informationen finden Sie in Bewerten der Migrationsbereitschaft.

predicate

Eine Abfragebedingung, die `true` oder zurückgibt`false`, was üblicherweise in einer Klausel vorkommt. `WHERE`

Prädikat Pushdown

Eine Technik zur Optimierung von Datenbankabfragen, bei der die Daten in der Abfrage vor der Übertragung gefiltert werden. Dadurch wird die Datenmenge reduziert, die aus der relationalen Datenbank abgerufen und verarbeitet werden muss, und die Abfrageleistung wird verbessert.

Präventive Kontrolle

Eine Sicherheitskontrolle, die verhindern soll, dass ein Ereignis eintritt. Diese Kontrollen stellen eine erste Verteidigungslinie dar, um unbefugten Zugriff oder unerwünschte Änderungen an Ihrem Netzwerk zu verhindern. Weitere Informationen finden Sie unter Präventive Kontrolle in Implementierung von Sicherheitskontrollen in AWS.

Prinzipal

Eine Entität AWS , die Aktionen ausführen und auf Ressourcen zugreifen kann. Bei dieser Entität handelt es sich in der Regel um einen Root-Benutzer für eine AWS-Konto, eine IAM-Rolle oder einen Benutzer. Weitere Informationen finden Sie unter Prinzipal in Rollenbegriffe und -konzepte in der IAM-Dokumentation.

Datenschutz von Natur aus

Ein systemtechnischer Ansatz, der den Datenschutz während des gesamten Entwicklungsprozesses berücksichtigt.

Privat gehostete Zonen

Ein Container, der Informationen darüber enthält, wie Amazon Route 53 auf DNS-Abfragen für eine Domain und ihre Subdomains innerhalb einer oder mehrerer VPCs reagieren soll. Weitere Informationen finden Sie unter Arbeiten mit privat gehosteten Zonen in der Route-53-Dokumentation.

proaktive Steuerung

Eine Sicherheitskontrolle, die den Einsatz nicht richtlinienkonformer Ressourcen verhindern soll. Mit diesen Steuerelementen werden Ressourcen gescannt, bevor sie bereitgestellt werden. Wenn die Ressource nicht mit der Steuerung konform ist, wird sie nicht bereitgestellt. Weitere Informationen finden Sie im Referenzhandbuch zu Kontrollen in der AWS Control Tower Dokumentation und unter Proaktive Kontrollen unter Implementierung von Sicherheitskontrollen am AWS.

Produktlebenszyklusmanagement (PLM)

Das Management von Daten und Prozessen für ein Produkt während seines gesamten Lebenszyklus, vom Design, der Entwicklung und Markteinführung über Wachstum und Reife bis hin zur Markteinführung und Markteinführung.

Produktionsumgebung

Siehe Umgebung.

Speicherprogrammierbare Steuerung (SPS)

In der Fertigung ein äußerst zuverlässiger, anpassungsfähiger Computer, der Maschinen überwacht und Fertigungsprozesse automatisiert.

schnelle Verkettung

Verwenden Sie die Ausgabe einer LLM-Eingabeaufforderung als Eingabe für die nächste Aufforderung, um bessere Antworten zu generieren. Diese Technik wird verwendet, um eine komplexe Aufgabe in Unteraufgaben zu unterteilen oder um eine vorläufige Antwort iterativ zu verfeinern oder zu erweitern. Sie trägt dazu bei, die Genauigkeit und Relevanz der Antworten eines Modells zu verbessern und ermöglicht detailliertere, personalisierte Ergebnisse.

Pseudonymisierung

Der Prozess, bei dem persönliche Identifikatoren in einem Datensatz durch Platzhalterwerte ersetzt werden. Pseudonymisierung kann zum Schutz der Privatsphäre beitragen. Pseudonymisierte Daten gelten weiterhin als personenbezogene Daten.

publish/subscribe (pub/sub)

Ein Muster, das asynchrone Kommunikation zwischen Microservices ermöglicht, um die Skalierbarkeit und Reaktionsfähigkeit zu verbessern. In einem auf Microservices basierenden MES kann ein Microservice beispielsweise Ereignismeldungen in einem Kanal veröffentlichen, den andere Microservices abonnieren können. Das System kann neue Microservices hinzufügen, ohne den Veröffentlichungsservice zu ändern.

Q

Abfrageplan

Eine Reihe von Schritten, wie Anweisungen, die für den Zugriff auf die Daten in einem relationalen SQL-Datenbanksystem verwendet werden.

Abfrageplanregression

Wenn ein Datenbankserviceoptimierer einen weniger optimalen Plan wählt als vor einer bestimmten Änderung der Datenbankumgebung. Dies kann durch Änderungen an Statistiken, Beschränkungen, Umgebungseinstellungen, Abfrageparameter-Bindungen und Aktualisierungen der Datenbank-Engine verursacht werden.

R

RACI-Matrix

Siehe verantwortlich, rechenschaftspflichtig, konsultiert, informiert (RACI).

RAG

Siehe Erweiterte Generierung beim Abrufen.

Ransomware

Eine bösartige Software, die entwickelt wurde, um den Zugriff auf ein Computersystem oder Daten zu blockieren, bis eine Zahlung erfolgt ist.

RASCI-Matrix

Siehe verantwortlich, rechenschaftspflichtig, konsultiert, informiert (RACI).

RCAC

Siehe Zugriffskontrolle für Zeilen und Spalten.

Read Replica

Eine Kopie einer Datenbank, die nur für Lesezwecke verwendet wird. Sie können Abfragen an das Lesereplikat weiterleiten, um die Belastung auf Ihrer Primärdatenbank zu reduzieren.

neu strukturieren

Siehe 7 Rs.

Recovery Point Objective (RPO)

Die maximal zulässige Zeitspanne seit dem letzten Datenwiederherstellungspunkt. Damit wird festgelegt, was als akzeptabler Datenverlust zwischen dem letzten Wiederherstellungspunkt und der Serviceunterbrechung gilt.

Wiederherstellungszeitziel (RTO)

Die maximal zulässige Verzögerung zwischen der Betriebsunterbrechung und der Wiederherstellung des Dienstes.

Refaktorierung

Siehe 7 Rs.

Region

Eine Sammlung von AWS Ressourcen in einem geografischen Gebiet. Jeder AWS-Region ist isoliert und unabhängig von den anderen, um Fehlertoleranz, Stabilität und Belastbarkeit zu gewährleisten. Weitere Informationen finden Sie unter Geben Sie an, was AWS-Regionen Ihr Konto verwenden kann.

Regression

Eine ML-Technik, die einen numerischen Wert vorhersagt. Zum Beispiel, um das Problem „Zu welchem Preis wird dieses Haus verkauft werden?“ zu lösen Ein ML-Modell könnte ein lineares Regressionsmodell verwenden, um den Verkaufspreis eines Hauses auf der Grundlage bekannter Fakten über das Haus (z. B. die Quadratmeterzahl) vorherzusagen.

rehosten

Siehe 7 Rs.

Veröffentlichung

In einem Bereitstellungsprozess der Akt der Förderung von Änderungen an einer Produktionsumgebung.

umziehen

Siehe 7 Rs.

neue Plattform

Siehe 7 Rs.

Rückkauf

Siehe 7 Rs.

Ausfallsicherheit

Die Fähigkeit einer Anwendung, Störungen zu widerstehen oder sich von ihnen zu erholen. Hochverfügbarkeit und Notfallwiederherstellung sind häufig Überlegungen bei der Planung der Ausfallsicherheit in der. AWS Cloud Weitere Informationen finden Sie unter AWS Cloud Resilienz.

Ressourcenbasierte Richtlinie

Eine mit einer Ressource verknüpfte Richtlinie, z. B. ein Amazon-S3-Bucket, ein Endpunkt oder ein Verschlüsselungsschlüssel. Diese Art von Richtlinie legt fest, welchen Prinzipalen der Zugriff gewährt wird, welche Aktionen unterstützt werden und welche anderen Bedingungen erfüllt sein müssen.

RACI-Matrix (verantwortlich, rechenschaftspflichtig, konsultiert, informiert)

Eine Matrix, die die Rollen und Verantwortlichkeiten für alle Parteien definiert, die an Migrationsaktivitäten und Cloud-Vorgängen beteiligt sind. Der Matrixname leitet sich von den in der Matrix definierten Zuständigkeitstypen ab: verantwortlich (R), rechenschaftspflichtig (A), konsultiert (C) und informiert (I). Der Unterstützungstyp (S) ist optional. Wenn Sie Unterstützung einbeziehen, wird die Matrix als RASCI-Matrix bezeichnet, und wenn Sie sie ausschließen, wird sie als RACI-Matrix bezeichnet.

Reaktive Kontrolle

Eine Sicherheitskontrolle, die darauf ausgelegt ist, die Behebung unerwünschter Ereignisse oder Abweichungen von Ihren Sicherheitsstandards voranzutreiben. Weitere Informationen finden Sie unter Reaktive Kontrolle in Implementieren von Sicherheitskontrollen in AWS.

Beibehaltung

Siehe 7 Rs.

zurückziehen

Siehe 7 Rs.

Retrieval Augmented Generation (RAG)

Eine generative KI-Technologie, bei der ein LLM auf eine maßgebliche Datenquelle verweist, die sich außerhalb seiner Trainingsdatenquellen befindet, bevor eine Antwort generiert wird. Ein RAG-Modell könnte beispielsweise eine semantische Suche in der Wissensdatenbank oder in benutzerdefinierten Daten einer Organisation durchführen. Weitere Informationen finden Sie unter Was ist RAG.

Drehung

Der Vorgang, bei dem ein Geheimnis regelmäßig aktualisiert wird, um es einem Angreifer zu erschweren, auf die Anmeldeinformationen zuzugreifen.

Zugriffskontrolle für Zeilen und Spalten (RCAC)

Die Verwendung einfacher, flexibler SQL-Ausdrücke mit definierten Zugriffsregeln. RCAC besteht aus Zeilenberechtigungen und Spaltenmasken.

RPO

Siehe Recovery Point Objective.

RTO

Siehe Ziel für die Erholungszeit.

Runbook

Eine Reihe manueller oder automatisierter Verfahren, die zur Ausführung einer bestimmten Aufgabe erforderlich sind. Diese sind in der Regel darauf ausgelegt, sich wiederholende Operationen oder Verfahren mit hohen Fehlerquoten zu rationalisieren.

S

SAML 2.0

Ein offener Standard, den viele Identitätsanbieter (IdPs) verwenden. Diese Funktion ermöglicht föderiertes Single Sign-On (SSO), sodass sich Benutzer bei den API-Vorgängen anmelden AWS-Managementkonsole oder die AWS API-Operationen aufrufen können, ohne dass Sie einen Benutzer in IAM für alle in Ihrer Organisation erstellen müssen. Weitere Informationen zum SAML-2.0.-basierten Verbund finden Sie unter Über den SAML-2.0-basierten Verbund in der IAM-Dokumentation.

SCADA

Siehe Aufsichtskontrolle und Datenerfassung.

SCP

Siehe Richtlinie zur Dienstkontrolle.

Secret

Interne AWS Secrets Manager, vertrauliche oder eingeschränkte Informationen, wie z. B. ein Passwort oder Benutzeranmeldedaten, die Sie in verschlüsselter Form speichern. Es besteht aus dem geheimen Wert und seinen Metadaten. Der geheime Wert kann binär, eine einzelne Zeichenfolge oder mehrere Zeichenketten sein. Weitere Informationen finden Sie unter Was ist in einem Secrets Manager Manager-Geheimnis? in der Secrets Manager Manager-Dokumentation.

Sicherheit durch Design

Ein systemtechnischer Ansatz, der die Sicherheit während des gesamten Entwicklungsprozesses berücksichtigt.

Sicherheitskontrolle

Ein technischer oder administrativer Integritätsschutz, der die Fähigkeit eines Bedrohungsakteurs, eine Schwachstelle auszunutzen, verhindert, erkennt oder einschränkt. Es gibt vier Haupttypen von Sicherheitskontrollen: präventiv, detektiv, reaktionsschnell und proaktiv.

Härtung der Sicherheit

Der Prozess, bei dem die Angriffsfläche reduziert wird, um sie widerstandsfähiger gegen Angriffe zu machen. Dies kann Aktionen wie das Entfernen von Ressourcen, die nicht mehr benötigt werden, die Implementierung der bewährten Sicherheitsmethode der Gewährung geringster Berechtigungen oder die Deaktivierung unnötiger Feature in Konfigurationsdateien umfassen.

System zur Verwaltung von Sicherheitsinformationen und Ereignissen (security information and event management – SIEM)

Tools und Services, die Systeme für das Sicherheitsinformationsmanagement (SIM) und das Management von Sicherheitsereignissen (SEM) kombinieren. Ein SIEM-System sammelt, überwacht und analysiert Daten von Servern, Netzwerken, Geräten und anderen Quellen, um Bedrohungen und Sicherheitsverletzungen zu erkennen und Warnmeldungen zu generieren.

Automatisierung von Sicherheitsreaktionen

Eine vordefinierte und programmierte Aktion, die darauf ausgelegt ist, automatisch auf ein Sicherheitsereignis zu reagieren oder es zu beheben. Diese Automatisierungen dienen als detektive oder reaktionsschnelle Sicherheitskontrollen, die Sie bei der Implementierung bewährter AWS Sicherheitsmethoden unterstützen. Beispiele für automatisierte Antwortaktionen sind das Ändern einer VPC-Sicherheitsgruppe, das Patchen einer Amazon EC2 EC2-Instance oder das Rotieren von Anmeldeinformationen.

Serverseitige Verschlüsselung

Verschlüsselung von Daten am Zielort durch denjenigen AWS-Service , der sie empfängt.

Service-Kontrollrichtlinie (SCP)

Eine Richtlinie, die eine zentrale Kontrolle über die Berechtigungen für alle Konten in einer Organisation in AWS Organizations ermöglicht. SCPs definieren Integritätsschutz oder legen Grenzwerte für Aktionen fest, die ein Administrator an Benutzer oder Rollen delegieren kann. Sie können SCPs als Zulassungs- oder Ablehnungslisten verwenden, um festzulegen, welche Services oder Aktionen zulässig oder verboten sind. Weitere Informationen finden Sie in der AWS Organizations Dokumentation unter Richtlinien zur Dienststeuerung.

Service-Endpunkt

Die URL des Einstiegspunkts für einen AWS-Service. Sie können den Endpunkt verwenden, um programmgesteuert eine Verbindung zum Zielservice herzustellen. Weitere Informationen finden Sie unter AWS-Service -Endpunkte in der Allgemeine AWS-Referenz.

Service Level Agreement (SLA)

Eine Vereinbarung, in der klargestellt wird, was ein IT-Team seinen Kunden zu bieten verspricht, z. B. in Bezug auf Verfügbarkeit und Leistung der Services.

Service-Level-Indikator (SLI)

Eine Messung eines Leistungsaspekts eines Dienstes, z. B. seiner Fehlerrate, Verfügbarkeit oder Durchsatz.

Service-Level-Ziel (SLO)

Eine Zielkennzahl, die den Zustand eines Dienstes darstellt, gemessen anhand eines Service-Level-Indikators.

Modell der geteilten Verantwortung

Ein Modell, das die Verantwortung beschreibt, mit der Sie gemeinsam AWS für Cloud-Sicherheit und Compliance verantwortlich sind. AWS ist für die Sicherheit der Cloud verantwortlich, während Sie für die Sicherheit in der Cloud verantwortlich sind. Weitere Informationen finden Sie unter Modell der geteilten Verantwortung.

Schatten-KI

Nicht autorisierte KI-Anwendungen, die außerhalb der kontrollierten Kanäle innerhalb eines Unternehmens erstellt oder verwendet wurden.

SIEM

Siehe Sicherheitsinformations- und Event-Management-System.

Single Point of Failure (SPOF)

Ein Fehler in einer einzelnen, kritischen Komponente einer Anwendung, der das System stören kann.

SLA

Siehe Service Level Agreement.

SLI

Siehe Service-Level-Indikator.

ALSO

Siehe Service-Level-Ziel.

Split-and-Seed-Modell

Ein Muster für die Skalierung und Beschleunigung von Modernisierungsprojekten. Sobald neue Features und Produktversionen definiert werden, teilt sich das Kernteam auf, um neue Produktteams zu bilden. Dies trägt zur Skalierung der Fähigkeiten und Services Ihrer Organisation bei, verbessert die Produktivität der Entwickler und unterstützt schnelle Innovationen. Weitere Informationen finden Sie unter Schrittweiser Ansatz zur Modernisierung von Anwendungen in der. AWS Cloud

SPOTTEN

Siehe Single Point of Failure.

Sternschema

Eine Datenbank-Organisationsstruktur, die eine große Faktentabelle zum Speichern von Transaktions- oder Messdaten und eine oder mehrere kleinere dimensionale Tabellen zum Speichern von Datenattributen verwendet. Diese Struktur ist für die Verwendung in einem Data Warehouse oder für Business Intelligence-Zwecke konzipiert.

Strangler-Fig-Muster

Ein Ansatz zur Modernisierung monolithischer Systeme, bei dem die Systemfunktionen schrittweise umgeschrieben und ersetzt werden, bis das Legacy-System außer Betrieb genommen werden kann. Dieses Muster verwendet die Analogie einer Feigenrebe, die zu einem etablierten Baum heranwächst und schließlich ihren Wirt überwindet und ersetzt. Das Muster wurde eingeführt von Martin Fowler als Möglichkeit, Risiken beim Umschreiben monolithischer Systeme zu managen. Ein Beispiel für die Anwendung dieses Musters finden Sie unter Schrittweise Modernisierung älterer Microsoft ASP.NET (ASMX) -Webservices mithilfe von Containern und Amazon API Gateway.

Subnetz

Ein Bereich von IP-Adressen in Ihrer VPC. Ein Subnetz muss sich in einer einzigen Availability Zone befinden.

Aufsichtskontrolle und Datenerfassung (SCADA)

In der Fertigung ein System, das Hardware und Software zur Überwachung von Sachanlagen und Produktionsabläufen verwendet.

Symmetrische Verschlüsselung

Ein Verschlüsselungsalgorithmus, der denselben Schlüssel zum Verschlüsseln und Entschlüsseln der Daten verwendet.

synthetisches Testen

Testen eines Systems auf eine Weise, die Benutzerinteraktionen simuliert, um potenzielle Probleme zu erkennen oder die Leistung zu überwachen. Sie können Amazon CloudWatch Synthetics verwenden, um diese Tests zu erstellen.

Systemaufforderung

Eine Technik, mit der einem LLM Kontext, Anweisungen oder Richtlinien zur Verfügung gestellt werden, um sein Verhalten zu steuern. Systemaufforderungen helfen dabei, den Kontext festzulegen und Regeln für Interaktionen mit Benutzern festzulegen.

T

tags

Key-value Paare, die als Metadaten für die Organisation Ihrer AWS Ressourcen dienen. Mit Tags können Sie Ressourcen verwalten, identifizieren, organisieren, suchen und filtern. Weitere Informationen finden Sie unter Markieren Ihrer AWS -Ressourcen.

Zielvariable

Der Wert, den Sie in überwachtem ML vorhersagen möchten. Dies wird auch als Ergebnisvariable bezeichnet. In einer Fertigungsumgebung könnte die Zielvariable beispielsweise ein Produktfehler sein.

Aufgabenliste

Ein Tool, das verwendet wird, um den Fortschritt anhand eines Runbooks zu verfolgen. Eine Aufgabenliste enthält eine Übersicht über das Runbook und eine Liste mit allgemeinen Aufgaben, die erledigt werden müssen. Für jede allgemeine Aufgabe werden der geschätzte Zeitaufwand, der Eigentümer und der Fortschritt angegeben.

Testumgebungen

Siehe Umgebung.

Training

Daten für Ihr ML-Modell bereitstellen, aus denen es lernen kann. Die Trainingsdaten müssen die richtige Antwort enthalten. Der Lernalgorithmus findet Muster in den Trainingsdaten, die die Attribute der Input-Daten dem Ziel (die Antwort, die Sie voraussagen möchten) zuordnen. Es gibt ein ML-Modell aus, das diese Muster erfasst. Sie können dann das ML-Modell verwenden, um Voraussagen für neue Daten zu erhalten, bei denen Sie das Ziel nicht kennen.

tool

Eine Funktion oder API, die ein Agent aufrufen kann, um Operationen in externen Systemen auszuführen.

Transit-Gateway

Ein Transit-Gateway ist ein Netzwerk-Transit-Hub, mit dem Sie Ihre VPCs und On-Premises-Netzwerke miteinander verbinden können. Weitere Informationen finden Sie in der AWS Transit Gateway Dokumentation unter Was ist ein Transit-Gateway.

Stammbasierter Workflow

Ein Ansatz, bei dem Entwickler Feature lokal in einem Feature-Zweig erstellen und testen und diese Änderungen dann im Hauptzweig zusammenführen. Der Hauptzweig wird dann sequentiell für die Entwicklungs-, Vorproduktions- und Produktionsumgebungen erstellt.

Vertrauenswürdiger Zugriff

Gewährung von Berechtigungen für einen Dienst, den Sie angeben, um Aufgaben in Ihrer Organisation AWS Organizations und in deren Konten in Ihrem Namen auszuführen. Der vertrauenswürdige Service erstellt in jedem Konto eine mit dem Service verknüpfte Rolle, wenn diese Rolle benötigt wird, um Verwaltungsaufgaben für Sie auszuführen. Weitere Informationen finden Sie in der AWS Organizations Dokumentation unter Verwendung AWS Organizations mit anderen AWS Diensten.

Optimieren

Aspekte Ihres Trainingsprozesses ändern, um die Genauigkeit des ML-Modells zu verbessern. Sie können das ML-Modell z. B. trainieren, indem Sie einen Beschriftungssatz generieren, Beschriftungen hinzufügen und diese Schritte dann mehrmals unter verschiedenen Einstellungen wiederholen, um das Modell zu optimieren.

Zwei-Pizzen-Team

Ein kleines DevOps Team, das Sie mit zwei Pizzen ernähren können. Eine Teamgröße von zwei Pizzen gewährleistet die bestmögliche Gelegenheit zur Zusammenarbeit bei der Softwareentwicklung.

U

Unsicherheit

Ein Konzept, das sich auf ungenaue, unvollständige oder unbekannte Informationen bezieht, die die Zuverlässigkeit von prädiktiven ML-Modellen untergraben können. Es gibt zwei Arten von Unsicherheit: Epistemische Unsicherheit wird durch begrenzte, unvollständige Daten verursacht, wohingegen aleatorische Unsicherheit durch Rauschen und Randomisierung verursacht wird, die in den Daten liegt.

undifferenzierte Aufgaben

Diese Arbeit wird auch als Schwerstarbeit bezeichnet. Dabei handelt es sich um Arbeiten, die zwar für die Erstellung und den Betrieb einer Anwendung erforderlich sind, aber dem Endbenutzer keinen direkten Mehrwert bieten oder keinen Wettbewerbsvorteil bieten. Beispiele für undifferenzierte Aufgaben sind Beschaffung, Wartung und Kapazitätsplanung.

höhere Umgebungen

Siehe Umgebung.

V

Vacuuming

Ein Vorgang zur Datenbankwartung, bei dem die Datenbank nach inkrementellen Aktualisierungen bereinigt wird, um Speicherplatz zurückzugewinnen und die Leistung zu verbessern.

Versionskontrolle

Prozesse und Tools zur Nachverfolgung von Änderungen, z. B. Änderungen am Quellcode in einem Repository.

VPC-Peering

Eine Verbindung zwischen zwei VPCs, mit der Sie den Datenverkehr mithilfe von privaten IP-Adressen weiterleiten können. Weitere Informationen finden Sie unter Was ist VPC-Peering? in der Amazon-VPC-Dokumentation.

Schwachstelle

Ein Software- oder Hardwarefehler, der die Sicherheit des Systems gefährdet.

W

Warmer Cache

Ein Puffer-Cache, der aktuelle, relevante Daten enthält, auf die häufig zugegriffen wird. Die Datenbank-Instance kann aus dem Puffer-Cache lesen, was schneller ist als das Lesen aus dem Hauptspeicher oder von der Festplatte.

warme Daten

Daten, auf die selten zugegriffen wird. Bei der Abfrage dieser Art von Daten sind mäßig langsame Abfragen in der Regel akzeptabel.

Fensterfunktion

Eine SQL-Funktion, die eine Berechnung für eine Gruppe von Zeilen durchführt, die sich in irgendeiner Weise auf den aktuellen Datensatz beziehen. Fensterfunktionen sind nützlich für die Verarbeitung von Aufgaben wie die Berechnung eines gleitenden Durchschnitts oder für den Zugriff auf den Wert von Zeilen auf der Grundlage der relativen Position der aktuellen Zeile.

Workload

Ein Workload ist eine Sammlung von Ressourcen und Code, die einen Unternehmenswert bietet, wie z. B. eine kundenorientierte Anwendung oder ein Backend-Prozess.

Workstream

Funktionsgruppen in einem Migrationsprojekt, die für eine bestimmte Reihe von Aufgaben verantwortlich sind. Jeder Workstream ist unabhängig, unterstützt aber die anderen Workstreams im Projekt. Der Portfolio-Workstream ist beispielsweise für die Priorisierung von Anwendungen, die Wellenplanung und die Erfassung von Migrationsmetadaten verantwortlich. Der Portfolio-Workstream liefert diese Komponenten an den Migrations-Workstream, der dann die Server und Anwendungen migriert.

WURM

Mal schreiben, viele lesen.

WQF

Siehe AWS Workload-Qualifizierungsrahmen.

einmal schreiben, viele lesen (WORM)

Ein Speichermodell, das Daten ein einziges Mal schreibt und verhindert, dass die Daten gelöscht oder geändert werden. Autorisierte Benutzer können die Daten so oft wie nötig lesen, aber sie können sie nicht ändern. Diese Datenspeicherinfrastruktur wird als unveränderlich angesehen.

Z

Zero-Day-Exploit

Ein Angriff, in der Regel Malware, der eine Zero-Day-Sicherheitslücke ausnutzt.

Zero-Day-Sicherheitslücke

Ein unfehlbarer Fehler oder eine Sicherheitslücke in einem Produktionssystem. Bedrohungsakteure können diese Art von Sicherheitslücke nutzen, um das System anzugreifen. Entwickler werden aufgrund des Angriffs häufig auf die Sicherheitsanfälligkeit aufmerksam.

Eingabeaufforderung ohne Vorwarnung

Bereitstellung von Anweisungen für die Ausführung einer Aufgabe an einen LLM, jedoch ohne Beispiele (Schnappschüsse), die ihm als Orientierungshilfe dienen könnten. Der LLM muss sein vortrainiertes Wissen einsetzen, um die Aufgabe zu bewältigen. Die Effektivität von Zero-Shot Prompting hängt von der Komplexität der Aufgabe und der Qualität der Aufforderung ab. Siehe auch Few-Shot-Eingabeaufforderungen.

Zombie-Anwendung

Eine Anwendung, deren durchschnittliche CPU- und Arbeitsspeichernutzung unter 5 Prozent liegt. In einem Migrationsprojekt ist es üblich, diese Anwendungen außer Betrieb zu nehmen.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.