Modele las estrategias del Protocolo de Contexto en AWS

AWS Guía prescriptiva

AWS Guía prescriptiva: Modele las estrategias del Protocolo de Contexto en AWS

Table of Contents

Introducción 1
Destinatarios previstos 2
Objetivos 3
¿Qué es el MCP? 5
Comprensión de las herramientas 5
¿Cuándo usar el MCP 8
Estrategia de diseño de herramientas MCP 12
Alcance de la herramienta 13
Granular 13
Básicos 14
Mejores prácticas para determinar el alcance de las herramientas MCP 15
Definiciones de herramientas 16
Método de especificación de herramientas 17
Enfoque de cadena de documentos 18
Mejores prácticas para las definiciones de las herramientas MCP 19
Descubrimiento de herramientas 19
Definición estática 19
Descubrimiento dinámico 20
Función de búsqueda 21
Prácticas recomendadas para el descubrimiento de herramientas MCP 21
Organización de herramientas 21
Mejores prácticas para la organización de las herramientas de MPC 22
Estrategia de alojamiento de MCP 24
Enfoques de alojamiento 24
Alojamiento local 24
Alojamiento remoto 26
Puerta de enlace MCP 26
Mejores prácticas para alojar servidores MCP 27
Estrategia de gobierno de MCP 28
Autenticación y autorización 28
Mejores prácticas para la autenticación y autorización de MCP 30
Controlar la carga 30
Mejores prácticas para controlar la carga 31
Métricas operativas 31

Colaboradores 33
Creación 33
Revisión 33
Redacción técnica 33
Historial de documentos 34
Glosario 35
..... 35
A 36
B 39
C 41
D 45
E 49
F 51
G 53
H 54
I 56
L 58
M 60
O 64
P 67
Q 70
R 70
S 73
T 77
U 79
V 80
W 80
Z 81
..... lxxxiii

Modele las estrategias del Protocolo de Contexto en AWS

Amazon Web Services (colaboradores)

Marzo de 2026 (historial del documento)

Esta guía puede ayudarlo a desarrollar e implementar estrategias de Protocolo de Contexto Modelo (MCP) en toda su organización para respaldar su viaje hacia la IA de los agentes. A medida que los agentes y los modelos lingüísticos se vuelven cada vez más fundamentales para las operaciones comerciales, establecer una estrategia de MCP es fundamental para que las soluciones de los agentes tengan éxito.

Esta guía analiza tres pilares fundamentales para desarrollar una estrategia de MCP: el diseño de las herramientas de MCP, el alojamiento de servidores de MCP y la gobernanza de MCP. Al abordar estos componentes interconectados, las organizaciones pueden crear sistemas escalables, seguros y eficaces para gestionar el contexto de los modelos en todas sus implementaciones de IA. Esta guía proporciona información práctica y orientación estratégica para las organizaciones que se encuentran en cualquier etapa de la transición de una organización a la IA, desde la experimentación inicial hasta las implementaciones de producción a gran escala. Esto les ayuda a desarrollar soluciones de MCP personalizadas que se ajusten a sus necesidades y objetivos específicos.

Estas mejores prácticas se derivan de las implementaciones reales de organizaciones que implementan el MCP a escala empresarial, de un análisis de los estándares actuales de especificación del MCP y de las lecciones aprendidas de las aplicaciones personalizadas de modelos de lenguaje de gran tamaño (LLM) en producción.

Los sistemas de IA se utilizan cada vez más sofisticados y robustos LLMs en una amplia variedad de casos de uso. LLMs destacan en la comprensión del lenguaje natural, la generación de respuestas similares a las humanas y el razonamiento sobre información compleja. Sin embargo, para pasar LLMs de ser interfaces conversacionales a sistemas capaces de realizar tareas complejas de forma autónoma, las organizaciones están adoptando arquitecturas de inteligencia artificial basadas en los agentes, sistemas de inteligencia artificial que permiten percibir su entorno, razonar en torno a los objetivos, tomar decisiones autónomas, orquestar múltiples pasos y tomar medidas para alcanzar los objetivos en nombre de los usuarios. Este enfoque agencial ayuda a las organizaciones a crear sistemas de IA que puedan entender la intención del usuario a través del lenguaje natural, coordinar de forma autónoma múltiples fuentes de datos y herramientas y ofrecer experiencias personalizadas a una escala que no era posible con los patrones tradicionales de solicitud-respuesta. Para que estos

agentes sean más capaces, las organizaciones deben proporcionar acceso a sus herramientas y datos existentes para enriquecer la comprensión contextual del agente y permitirle actuar en nombre del usuario.

El MCP proporciona un protocolo estandarizado para la integración de herramientas de inteligencia artificial, lo que permite una comunicación coherente entre los agentes y los recursos externos. Si bien el propio MCP define el estándar de comunicación, su implementación efectiva requiere una consideración cuidadosa de los patrones arquitectónicos, los modelos de seguridad, las prácticas operativas y las estrategias de optimización del rendimiento para lograr soluciones escalables, seguras y fáciles de mantener.

Esta guía sintetiza las lecciones aprendidas de las implementaciones de MCP empresariales y proporciona recomendaciones prácticas que se alinean con el marco de Well-Architected.AWS Abarca las estrategias para el diseño de herramientas de MCP, el alojamiento de servidores de MCP y la gobernanza de los MCP, que son esenciales para crear sus propias soluciones de MCP. Las recomendaciones de esta guía se basan en los siguientes cinco pilares del AWS Well-Architected Framework:

- Seguridad: aislamiento de tokens, credenciales restringidas, autorización independiente read/write
- Excelencia operativa: selección de herramientas, métricas de precisión, conjuntos de datos de referencia para pruebas de regresión
- Fiabilidad: limitación de velocidad por usuario y herramienta, reducción de carga
- Eficiencia del rendimiento: herramientas centradas en el flujo de trabajo, filtrado de herramientas y búsqueda semántica para reducir el uso de las ventanas contextuales
- Optimización de costes: servidores MCP reutilizables en todos los equipos, lo que reduce los costes de los tokens por solicitud mediante el filtrado de herramientas

Destinatarios previstos

Esta guía está dirigida a arquitectos, desarrolladores y líderes tecnológicos que están implementando soluciones de inteligencia artificial automática en sus organizaciones. Para comprender los conceptos de esta guía, debe comprender cómo LLMs funciona y tener conocimientos básicos sobre el MCP, las herramientas y la ingeniería puntual.

Objetivos

Crear sistemas de inteligencia artificial de Agentic que estén listos para la producción significa abordar la gobernanza, la optimización y la seguridad de manera conjunta para respaldar las políticas de su organización. A continuación, se explica cómo esta guía aborda estos objetivos:

- Gobernanza: sin una gobernanza centralizada, no puede responder a las preguntas de auditoría sobre sus cargas de trabajo de IA, como qué agentes accedieron a qué datos, con qué permisos y cuándo. Tampoco puede imponer el control de versiones. En la sección de estrategia de alojamiento de MCP de esta guía se explica cómo los usuarios podrían utilizar servidores MCP locales anticuados con vulnerabilidades conocidas debido a la falta de controles sistemáticos.

  Para las industrias reguladas, la gobernanza es fundamental. Los auditores desean ver la aplicación de las políticas y el seguimiento del uso de las herramientas en todos los agentes desde un único panel. La gobernanza del MCP lo proporciona.

  Si sigue las recomendaciones de esta guía, puede mejorar la precisión de las tareas entre un 28 y un 32% según los puntos de referencia revisados por pares. Para obtener más información, consulte MARCO: Multi-Agent Real-Time Chat Orchestration (sitio web de ACL Anthology). La gobernanza no se basa solo en el cumplimiento, sino que también mejora el rendimiento del sistema de IA de su agencia.
- Optimización: es posible que sus equipos creen las mismas integraciones más de una vez. Por ejemplo, cuando cinco equipos diferentes escriben su propio script de consulta de bases de datos para que su aplicación de IA se comunique con sus bases de datos, se multiplica por cinco el coste de desarrollo y el mantenimiento de cinco conjuntos de listas de errores. MCP te permite crearlo una vez y compartirlo con toda la comunidad de ingenieros. Los ahorros se acumulan a medida que aumenta el número de agentes.

  También hay un problema de coste por solicitud que la mayoría de los equipos no notan al principio. Cada definición de herramienta consume símbolos de ventana de contexto. Con 20 herramientas, gastas entre 5.000 y 10.000 fichas por invocación únicamente en descripciones, además de en las consultas de los usuarios. Esto aumenta la latencia y los costes de inferencia de LLM y reduce la precisión, ya que el modelo se esfuerza por elegir la herramienta adecuada de la lista de herramientas disponibles.

  Los agentes que utilizan envoltorios de herramientas estructurados son aproximadamente tres veces más precisos en las tareas de la base de datos que los agentes que acceden APIs directamente (para obtener más información, consulte Middleware para LLMs: Las herramientas

son fundamentales para los agentes lingüísticos en entornos complejos). La forma de diseñar y presentar las herramientas para un modelo de IA es importante. Esta guía recomienda dar a las herramientas esquemas claros, ajustarlas a los flujos de trabajo reales en lugar de a puntos finales sin procesar, y limitar la información en la ventana contextual. La sección de estrategia de diseño de herramientas MCP de esta guía profundiza en estos aspectos.

- Seguridad y cumplimiento: imagine un sistema de IA de agencia que imagina una etapa de limpieza e intenta eliminar una base de datos de producción. Si el agente heredó las credenciales de administrador completas del usuario, es posible que la eliminación se lleve a cabo. Con el aislamiento de los tokens y las credenciales restringidas que solo permiten el acceso de lectura y creación, se produce un error seguro.

  Los flujos de trabajo regulados agudizan aún más esta situación. La guía proporciona ejemplos (procesos de atención médica que requieren la validación de la HIPAA y la anonimización de la información de identificación personal antes de procesar los datos de los pacientes). Al incorporar esta lógica en las herramientas del MCP, el cumplimiento se produce siempre de forma determinista.

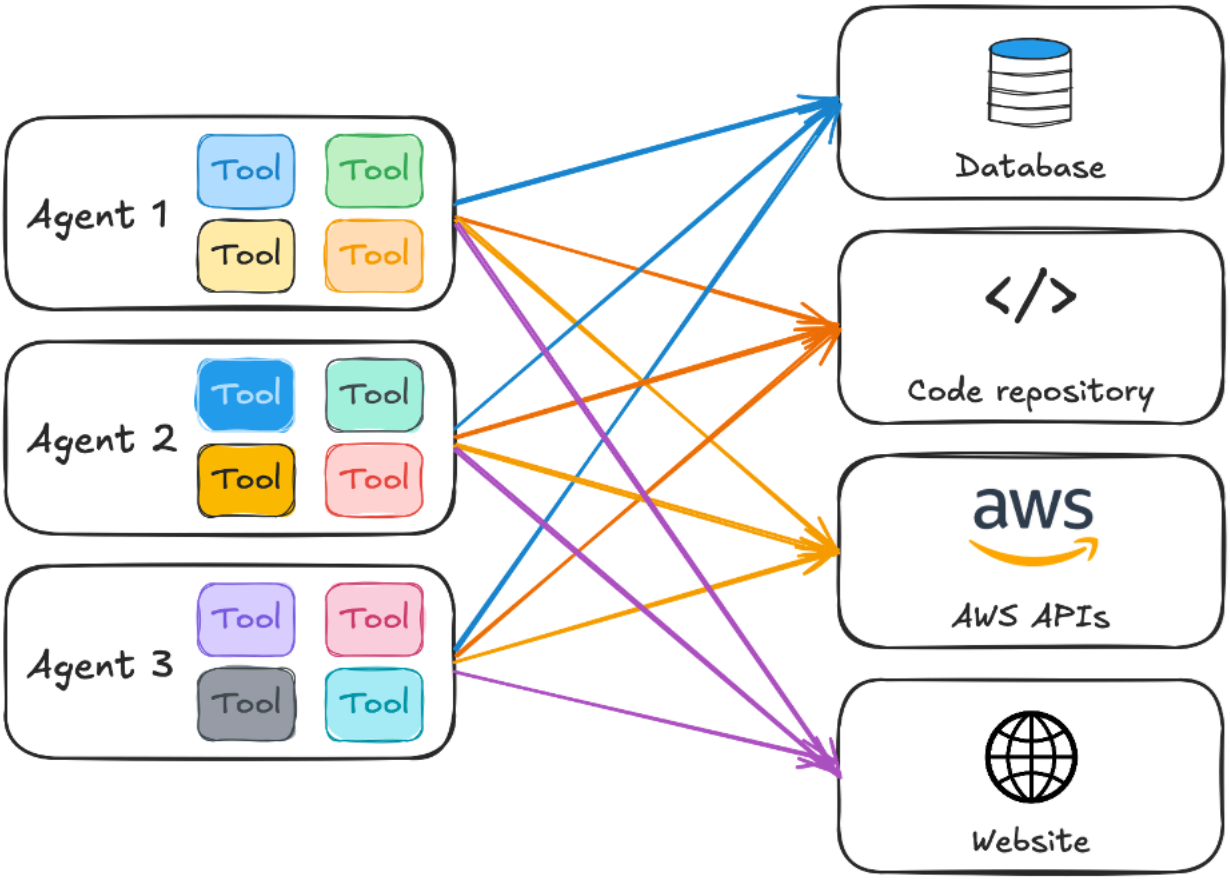
¿Qué es el MCP?

LLMs predice la respuesta a una pregunta en función de sus datos de entrenamiento. Esto significa que el LLM solo puede proporcionar respuestas sobre datos y eventos que ya haya visto. Métodos como la generación aumentada de recuperación (RAG) y las bases de conocimiento permiten incluir datos contextuales. Sin embargo, si le preguntara a un LLM cuál será la previsión meteorológica de mañana o cuántos clientes hay en su base de datos, lo más probable es que tenga alucinaciones o no sea capaz de dar una respuesta, ya que estos datos están fuera del alcance de los conocimientos previamente entrenados por el LLM. Para poder responder a este tipo de preguntas, un agente necesita tener acceso a capacidades y datos externos APIs fuera del contexto nativo del LLM.

Comprensión de las herramientas

Podemos dar al LLM acceso a sistemas y contextos adicionales a través de herramientas.Las herramientas son funciones que se le asignan al LLM para lograr un objetivo claro. Una herramienta podría llamar a una API, consultar una base de datos, realizar operaciones de calculadora, operar un entorno limitado de códigos, realizar una búsqueda en la web e incluso invocar otro sistema de IA o. agent-as-a-tool Cada herramienta debe incluir una descripción que indique al LLM qué hace la herramienta, cuándo usarla y qué parámetros acepta. Esto permite al LLM tomar decisiones matizadas sobre qué herramienta o combinación de herramientas utilizar en función de las aportaciones del usuario. Al LLM se le indica qué herramientas están disponibles para el agente, lo que le permite generar respuestas que le indiquen al agente que invoque la herramienta. Por ejemplo, cuando le pregunte al LLM cuántos clientes hay en su base de datos, el LLM enviará una respuesta al agente solicitando que ejecute la `query_database` herramienta con parámetros de entrada específicos. El LLM determina qué herramienta invocar y las entradas para la llamada a la herramienta. A continuación, el agente ejecuta la herramienta, que convierte la entrada en lenguaje natural en una llamada a una función sintácticamente correcta y ejecuta la consulta. El agente invoca la herramienta o las herramientas según las instrucciones del LLM y esos resultados se devuelven al LLM. Esto aprovecha la capacidad del LLM para razonar en lugar de introducir texto y seleccionar las herramientas adecuadas para el trabajo.

La siguiente imagen muestra cómo cada agente gestiona su propio conjunto de herramientas para cada objetivo.



Ampliar el acceso a las herramientas puede plantear desafíos para las soluciones de IA de los agentes:

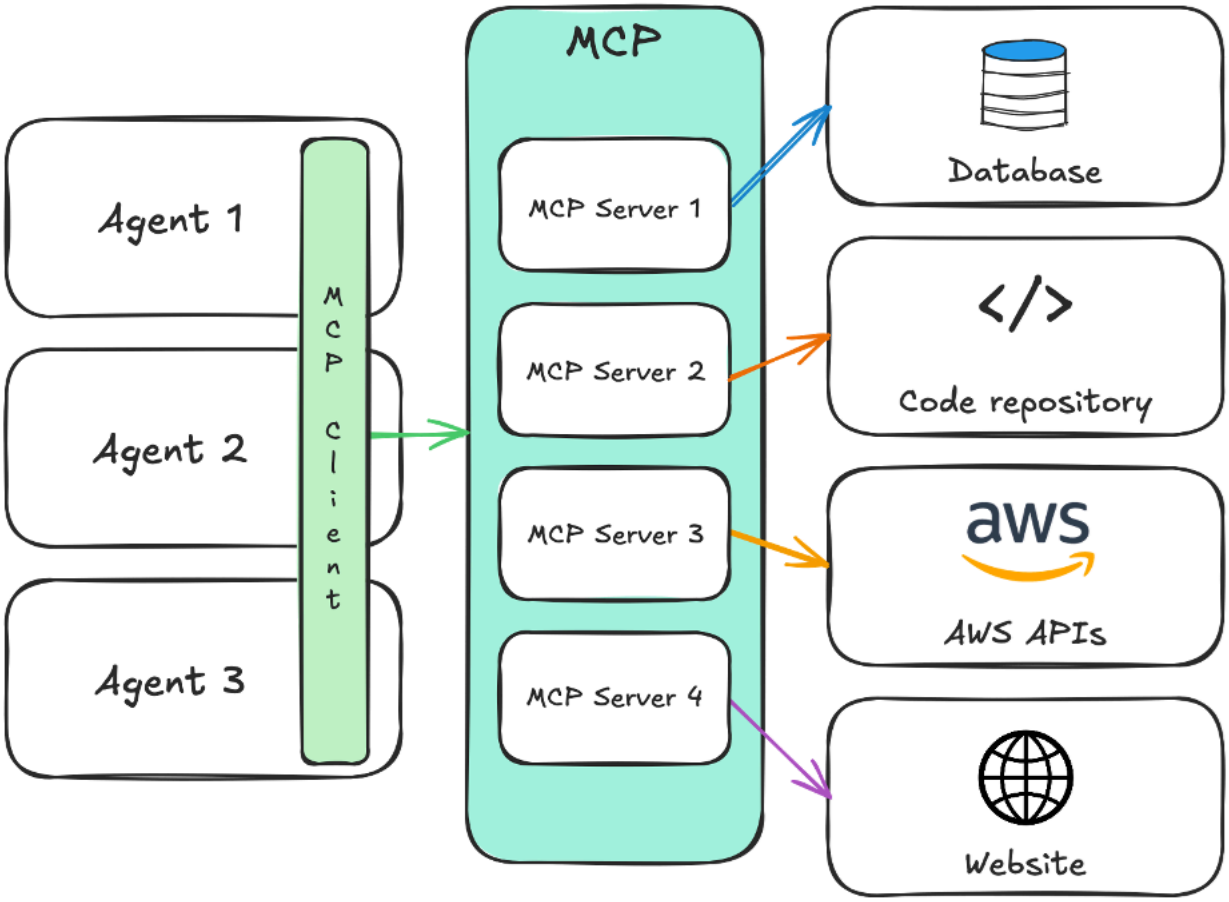
- Si cada desarrollador crea su propia herramienta para las mismas capacidades externas, hay una gran cantidad de esfuerzos duplicados y formas no estandarizadas de interactuar con estas capacidades externas. Esto produce implementaciones incoherentes entre sus agentes. Si bien podría resolver ese problema desarrollando herramientas estándar en las bibliotecas y distribuyéndolas, esto carece de una gobernanza centralizada. Esto dificulta la aplicación de las políticas de seguridad, el seguimiento del uso de las herramientas, la gestión del control de versiones en todos los equipos o la garantía del cumplimiento de las normas organizativas. Además, al integrar las herramientas directamente en el agente, debe volver a desplegarlo cada vez que se cree una nueva herramienta o se actualice una existente.
- Proporcionar herramientas a un LLM consume su ventana de contexto. La ventana de contexto es el número de símbolos (unidades de texto que se LLMs procesan, que normalmente representan palabras, partes de palabras o signos de puntuación) que un modelo puede considerar en cualquier momento dado. LLMs tienen límites de ventana de contexto. Las herramientas y su documentación ocupan esa ventana de contexto finita junto con las indicaciones del sistema y

las del usuario. A medida que se va llenando la ventana de contexto, LLMs pueden experimentar una degradación del rendimiento debido a varios factores: dificultad para identificar la información relevante, aumento de la complejidad del procesamiento y reducción de la capacidad de razonamiento. El desafío se agrava cuando las definiciones de las herramientas, las instrucciones del sistema y el historial de conversaciones compiten por un espacio limitado en la ventana contextual, ya que se proporcionan en cada invocación al LLM.

Por lo tanto, la cantidad de herramientas y la forma en que se documentan tienen un impacto directo en el rendimiento del LLM, como el tiempo de respuesta y la precisión.

El MCP establece un estándar universal para conectar los agentes con capacidades externas. Se lo conoce comúnmente como el «USB-C para aplicaciones de IA». En lugar de registrar las herramientas directamente con los agentes, los servidores MCP actúan como intermediarios para alojar las herramientas que se descubren e invocan mediante JSON-RPC 2.0. En lugar de añadir decenas o cientos de herramientas diferentes a su agente y mantenerlas a lo largo del tiempo, el MCP le permite registrar los servidores MCP que encapsulan las herramientas a las que puede acceder el agente. Este enfoque estandariza la forma en que se empaquetan, presentan e invocan las herramientas. Esto puede ayudar a abordar los desafíos de escala y gobernanza que implica el uso de las herramientas por parte de sus agentes. También separa el desarrollo y las operaciones de los agentes de las herramientas que utiliza para las capacidades externas.

La siguiente figura muestra a los agentes que utilizan el MCP para acceder a recursos externos.



Sin embargo, el estándar MCP no resuelve todos los desafíos de escalamiento y gobernanza. La implementación de los servidores MCP debe combinarse con estrategias eficaces de diseño de herramientas, alojamiento y gobierno empresarial. Esta guía proporciona las mejores prácticas para cada estrategia a fin de ayudarle a crear y utilizar el MCP como parte de las soluciones de IA de su agencia.

¿Cuándo usar el MCP

MCP proporciona una infraestructura estratégica para ampliar las iniciativas de IA de sus agencias. Al centralizar la administración y el gobierno de las herramientas, los servidores MCP reducen el costo acumulado de crear y mantener integraciones personalizadas entre varios agentes. Esto ofrece una rentabilidad cada vez mayor a medida que su ecosistema de agentes se expande.

Es probable que el MCP pase a formar parte de su estrategia cuando:

- Necesita una gobernanza centralizada sobre la forma en que los agentes acceden a los sistemas y servicios empresariales, como las bases de datos APIs, las herramientas internas y las integraciones de terceros.
- Los desarrolladores dedican demasiado tiempo a crear integraciones personalizadas que no son coherentes en todas las implementaciones.
- Tiene herramientas duplicadas que podrían ofrecer capacidades comunes.
- Desea ofrecer sus herramientas o datos patentados a consumidores externos o sistemas de agencias de terceros a través de interfaces MCP estandarizadas y gobernadas, lo que le permitirá obtener nuevas fuentes de ingresos y, al mismo tiempo, mantener la seguridad y el control.

Una vez que decida si los servidores MCP van a formar parte de su estrategia, evalúe si las implementaciones de servidores MCP de código abierto existentes satisfacen sus necesidades, si requieren mejoras o si necesita crear servidores personalizados. Muchas implementaciones de servidores MCP prediseñadas están disponibles en repositorios públicos y abarcan capacidades comunes, como el acceso a los sistemas de archivos, la navegación web, los entornos limitados de código, el acceso a bases de datos y las integraciones de API.

En muchos casos, los servidores MCP preexistentes son suficientes. Por ejemplo, AWS proporciona un servidor MCP remoto gestionado que proporciona a los asistentes y agentes de IA un acceso seguro y autenticado Servicios de AWS mediante interacciones en lenguaje natural. Servidor de AWS MCP Puede usarlo Servidor de AWS MCP para realizar AWS tareas complejas de varios pasos, combinando el acceso en tiempo real a la AWS documentación, las llamadas a la API correctas desde el punto de vista sintáctico y los flujos de trabajo prediseñados denominados agentes que siguen las mejores prácticas. SOPs AWS AWS las prueba continuamente Servidor de AWS MCP para asegurarse de que los agentes de atención al cliente puedan utilizarlas correctamente.

Debe probar estos servidores MCP existentes con sus agentes para determinar si se adaptan a sus casos de uso. Si un agente no completa los flujos de trabajo, genera respuestas incorrectas o subóptimas, no logra gestionar procesos complejos de varios pasos o pasa por alto importantes prácticas recomendadas o consideraciones de seguridad específicas de un dominio, debería considerar la posibilidad de realizar mejoras en varias dimensiones.

Cuando los servidores MCP existentes no satisfagan plenamente sus necesidades y tengan dificultades para utilizar las herramientas existentes correctamente o producir respuestas precisas, considere estos enfoques de mejora antes de crear servidores personalizados:

- Enriquezca el contexto de los agentes: si su agente tiene dificultades para utilizar de manera correcta o eficiente las herramientas de un servidor MCP existente, considere la posibilidad de complementar esas definiciones de herramientas con documentación o ejemplos adicionales. Esto ayuda a proporcionar un contexto adicional al LLM.
- Agregue herramientas complementarias: amplíe los servidores MCP existentes con herramientas que accedan a los datos organizativos adicionales o al contexto que los agentes necesitan para completar los flujos de trabajo correctamente.
- Mejore las funciones subyacentes APIs: simplifique su servicio APIs para que sea más adecuado para la gestión de problemas de aprendizaje automático reduciendo la complejidad de los parámetros, proporcionando mensajes de error más claros y ofreciendo valores predeterminados razonables que los agentes puedan utilizar.

Si bien el uso de las implementaciones de servidores MCP existentes acelera el desarrollo de capacidades comunes, la creación de servidores MCP personalizados es una necesidad cuando su caso de uso requiere una funcionalidad especializada. Los servidores MCP personalizados le ayudan a encapsular la experiencia en el campo, a aplicar los estándares organizacionales, a mejorar la confiabilidad de los agentes para flujos de trabajo complejos y a respaldar el cumplimiento de los requisitos de seguridad. Considere la posibilidad de crear un servidor MCP personalizado en las siguientes situaciones:

- Flujos de trabajo específicos del dominio: los flujos de trabajo de varios pasos que requieren experiencia en el dominio deben encapsularse en herramientas MCP personalizadas cuando la documentación de la API no incluye los conocimientos necesarios. Por ejemplo, en lugar de permitir que los agentes organicen complejos flujos de datos de atención médica que deben validar el cumplimiento de la Ley de Portabilidad y Responsabilidad de los Seguros de Salud (HIPAA), anonimizar la PII y transformarla al formato HL7 FHIR, proporcione una herramienta que incorpore directamente la experiencia en el campo. `process_patient_data` Esto elimina la dependencia del LLM para organizar y ejecutar correctamente los pasos del flujo de trabajo, lo que mejora la coherencia y el cumplimiento.
- Abstracciones de la vía dorada: los agentes pueden tener dificultades para implementar enfoques óptimos porque carecen de un contexto organizacional y utilizan patrones básicos por defecto en lugar de las mejores prácticas organizacionales. En estos escenarios, puede aplicar normas prescriptivas en materia de costes, rendimiento o seguridad encapsulando estas vías de oro en herramientas MCP personalizadas. Por ejemplo, en lugar de permitir que los agentes desplieguen una infraestructura con una configuración predeterminada que podría resultar insegura o

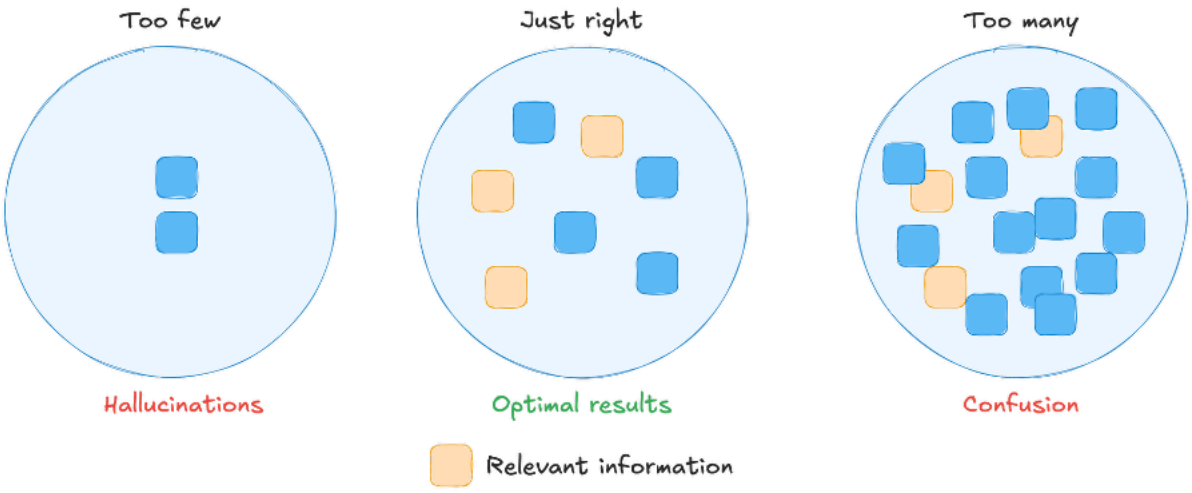
ineficiente, proporcione una `deploy_secure_infrastructure` herramienta que incorpore directamente los estándares de su organización.

- Organización compleja de varios servicios: en lugar de hacer que el agente organice flujos de trabajo complejos intentando deducir la secuencia y el conjunto de servicios correctos que debe utilizar en cada paso, puede crear esa lógica de forma determinista dentro de una herramienta de MCP. Es posible que también desee proporcionar conocimientos sobre los patrones óptimos de integración de servicios que el agente tal vez desconozca. Esto también puede mejorar la precisión y la eficiencia de sus agentes.
- Mejores prácticas específicas para cada servicio: esto es común en el caso de las herramientas centradas en la seguridad que ayudan a los agentes a implementar políticas de cifrado, controles de acceso y patrones de cumplimiento específicos del servicio al que se accede a través de la herramienta de agente. Además, si existen prácticas recomendadas operativas específicas para un servicio que no son obvias, el uso de un servidor MCP puede ayudarle a asegurarse de que se implementan y no dejan que sea un agente quien razone al respecto.

Estrategia de diseño de herramientas MCP

La tarea principal del cliente y el servidor del MCP es descubrir y presentar las herramientas al LLM para que pueda utilizarlas para mejorar sus respuestas. Esto hace que el diseño de las herramientas de MCP sea una de las estrategias más importantes para crear soluciones de MCP eficaces. Desde la perspectiva del modelo, las herramientas son una función que pueden invocar según sea necesario para proporcionar respuestas más precisas y completas. La interfaz de funciones resume la implementación subyacente de una herramienta, que puede abarcar desde una simple llamada a la API hasta una compleja lógica de flujo de trabajo.

Sin embargo, debe lograr un equilibrio con la cantidad de herramientas que se proporcionan al LLM. Si hay muy pocas herramientas, es posible que el LLM no pueda recopilar el contexto y la información correctos, por lo que hará las suposiciones más acertadas con la información disponible en el modelo. Si hay demasiadas herramientas, el LLM puede confundirse con la selección y la secuencia correctas de las herramientas, lo que puede provocar alucinaciones. Tu objetivo es conseguir el número correcto de herramientas. La siguiente imagen muestra los desafíos que supone tener muy pocas o demasiadas herramientas.



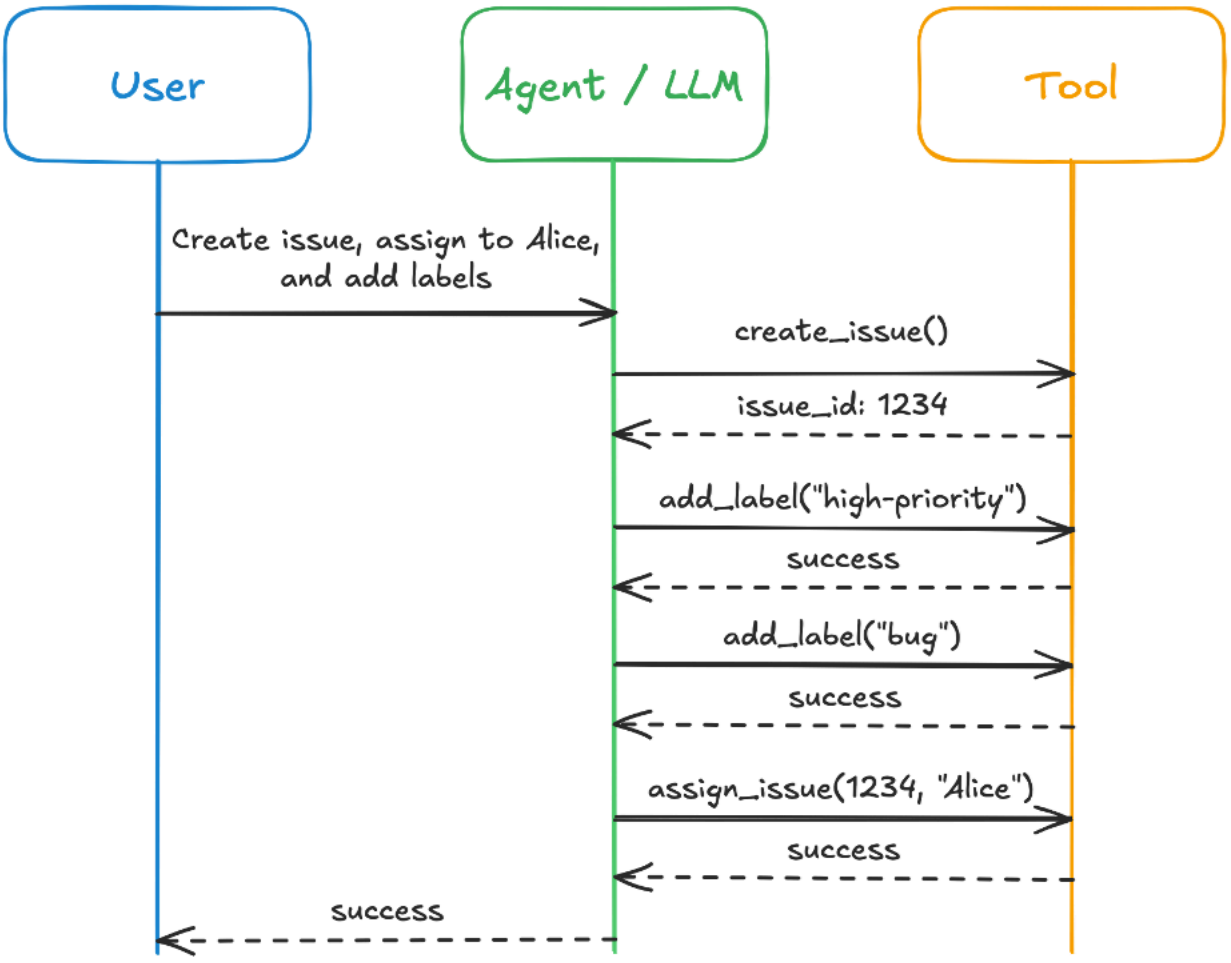
La solución requiere saber cuántas herramientas se deben proporcionar y cómo determinar el alcance de cada herramienta. La granularidad de sus herramientas, ya sea que se asignen a llamadas individuales a la API o a flujos de trabajo completos, repercute directamente en la cantidad total de herramientas que los agentes necesitan y en la eficacia con la que pueden utilizarlas. En esta sección, se proporcionan las mejores prácticas para determinar el alcance de las herramientas de MCP, crear definiciones de herramientas, descubrir herramientas y organizarlas.

Alcance de la herramienta

Existen dos enfoques para desarrollar herramientas: granulares y pormenorizadas.

Granular

En un enfoque detallado, crearía una herramienta por API, acción o consulta. Por ejemplo, puedes crear`create_issue`, `get_issue` `add_label`o `assign_issue`, y `close_issue` herramientas para tu repositorio de Git. Esto permitiría al LLM realizar llamadas granulares a cada API y organizar cada una de ellas según sea necesario. Considere la siguiente pregunta: «Cree un problema para el servicio del producto denominado «Query solo devuelve resultados parciales», etiquételo como error y de alta prioridad y asígnelo a Alice». La siguiente imagen muestra cómo respondería un tool-per-API enfoque a esta solicitud.

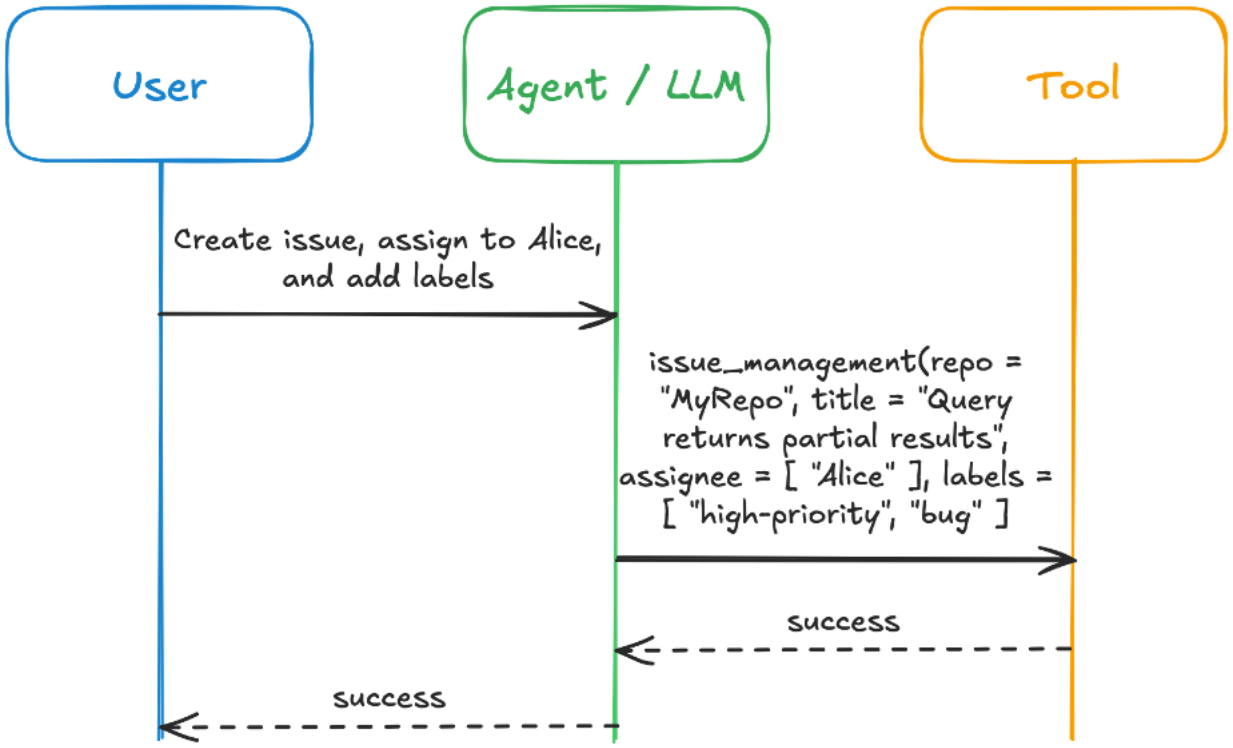


En este enfoque, el indicador del sistema y todas las definiciones de herramientas registradas se proporcionan al LLM en cada llamada. Esto consume más contexto e implica una penalización de

latencia, ya que cada llamada a la herramienta representa una llamada individual al LLM. También aumenta la complejidad de la gestión de los errores en el flujo de trabajo.

Básicos

Un enfoque detallado o basado en el flujo de trabajo serían las herramientas orientadas al flujo de trabajo. La herramienta se centra en la intención del usuario más que en la estructura de la API. end-to-end En lugar de una tool-per-API, tienes una herramienta que llama a varias de forma determinista. APIs Usando el ejemplo anterior del repositorio de Git, puedes crear una `create_and_setup_issue` herramienta a la que el agente llame una vez. La implementación de la herramienta crea el problema, añade etiquetas y lo asigna a un usuario en función de los parámetros proporcionados a la herramienta. La siguiente imagen muestra cómo un enfoque detallado procesaría el mismo mensaje.



Este enfoque muestra cómo toda la complejidad permanece oculta en la capa LLM. Cuando la lógica de orquestación está integrada en la implementación de la herramienta, todos los pasos secuenciales (el registro, la lógica de reintento, los disyuntores y la limitación de velocidad) se realizan de forma determinista en la herramienta. El enfoque basado en el flujo de trabajo facilita que el LLM invoque la herramienta correcta con los parámetros correctos. Es importante tener en cuenta que es posible que algunas API ya proporcionen la intención del flujo de trabajo, como la API

Amazon EC2`RunInstances`. En estos casos, a tool-per-API podría proporcionar el diseño orientado al flujo de trabajo que desea.

Sin embargo, las herramientas también pueden resultar demasiado gruesas. Si su única herramienta de flujo de trabajo intenta hacer demasiadas cosas y tiene muchos parámetros posibles, el LLM puede tener dificultades para razonar sobre cómo utilizar correctamente la herramienta. También puede crear problemas con la selección de parámetros y la gestión de errores. Por lo tanto, el desarrollo de herramientas debe lograr un equilibrio que se ajuste a la intención del usuario y evite que la funcionalidad de una sola herramienta sea insuficiente o excesiva. Le recomendamos que diseñe las herramientas en función de los flujos de trabajo completos de los usuarios, agrupando las operaciones que suelen producirse juntas (como tres o más llamadas a la API). También le recomendamos que descomponga las herramientas que superen ocho o más parámetros o que gestionen varias intenciones de usuario distintas. Realice pruebas con indicaciones reales para comprobar que los agentes pueden utilizar correctamente cada herramienta.

Si tiene flujos de trabajo complejos y dinámicos que no se pueden resumir fácilmente como una herramienta determinista, podría considerar la posibilidad de utilizar el patrón. agent-as-tool En lugar de que su agente principal intente organizar tareas complejas en un flujo de trabajo, un agente especializado puede actuar como una herramienta. Este tipo de herramientas permiten implementar procesos avanzados de toma de decisiones y ramificaciones, y pueden gestionar errores y reintentar una lógica que no se puede gestionar fácilmente en un código determinista. Es similar, pero distinto, al protocolo Agent2Agent (A2A). El protocolo A2A es complementario y proporciona interoperabilidad y colaboración entre agentes en cualquier marco institucional.

Le recomendamos que comience con el análisis del flujo de trabajo mapeando los flujos de trabajo de los usuarios más comunes para identificar las capacidades principales que necesita cada agente. Esto establece su conjunto de herramientas mínimo viable. Basándonos en nuestra experiencia en el desarrollo de servidores MCP a escala, recomendamos las siguientes prácticas. Cuando estas prácticas entren en conflicto, priorice la intención del usuario y el flujo de trabajo.

Mejores prácticas para determinar el alcance de las herramientas MCP

- Piense en las historias de los usuarios y agrupe las operaciones comunes: las herramientas deberían mapearse directamente para completar las interacciones de los usuarios, en lugar de requerir la organización de varias operaciones. Si los flujos de trabajo suelen requerir tres o más llamadas independientes, combínelas en una sola herramienta. Esto reduce la carga cognitiva del LLM, minimiza el número de llamadas a las herramientas, reduce el consumo de contexto y la latencia necesarios para completar las tareas, y mejora la precisión y la latencia.

- Limite los parámetros a ocho o menos: si una herramienta supera los ocho parámetros, descompóngala en varias herramientas. LLMs tienen problemas con la selección de parámetros a medida que aumenta la complejidad.

  > **Note**
  >
  > Si las operaciones de agrupación requieren más de ocho parámetros, priorice la agrupación por encima del recuento de parámetros, ya que simplificar el flujo de trabajo es más valioso que limitar estrictamente los parámetros.

- Operaciones de lectura y escritura separadas: proporcione diferentes herramientas para leer los datos y modificarlos. Esta separación hace explícito cuándo los agentes realizan operaciones potencialmente destructivas, permite diferentes políticas de autorización y reduce el riesgo de modificaciones no deseadas durante la recopilación de información.
- Proporcione valores predeterminados razonables: diseñe herramientas de modo que el LLM necesite especificar solo los parámetros que son específicos de la solicitud individual. Los valores predeterminados reducen la complejidad de los parámetros y mejoran la precisión de la selección de herramientas al minimizar la información sobre la que el LLM debe razonar.
- Prefiera la ejecución determinista: haga que la ejecución y la salida de la herramienta sean deterministas siempre que sea posible. Las herramientas deterministas son más fiables y fáciles de probar. Para flujos de trabajo complejos que requieren una orquestación inteligente, una lógica de ramificación o una gestión avanzada de errores que no se pueden gestionar fácilmente en un código determinista, considere la posibilidad de utilizar agentes especializados como herramientas. Sin embargo, utilice este patrón de forma selectiva porque añade complejidad.

Definiciones de herramientas

Cuando un LLM recibe una solicitud que no puede gestionar directamente, revisará las herramientas disponibles para ayudarle a completar la solicitud. El LLM selecciona las herramientas en función de su comprensión semántica de los nombres y descripciones de las herramientas proporcionadas y de las instrucciones que figuran en la solicitud. Luego, creará la entrada en función del esquema de entrada definido y esperará la salida en función del esquema de salida. Por lo tanto, crear definiciones de herramientas descriptivas y esquemas de entrada y salida validados es fundamental para ayudar al LLM a seleccionar las herramientas de manera efectiva. En general, existen dos enfoques para crear esta documentación: el enfoque de especificación de herramientas y el enfoque de cadena de documentos.

Método de especificación de herramientas

El enfoque recomendado consiste en seguir directamente la especificación de la herramienta MCP al definir la herramienta. El siguiente ejemplo se muestra con el decorador de herramientas Strands Agent:

```
@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:
  …
```

Utiliza campos estándar, como,`name`, `description`, `inputSchema`, y se `outputSchema` asegura de que cada herramienta tenga una documentación coherente que tanto el LLM como los humanos puedan entender. Cada herramienta debe definir estos campos como mínimo y, si lo desea, incluir un título y anotaciones, que son sugerencias opcionales sobre el comportamiento de la herramienta. Cuando sea posible, utilice enumeraciones para los valores de los parámetros para que el LLM pueda seleccionar fácilmente las opciones correctas. Las enumeraciones funcionan mejor para conjuntos finitos, como valores de estado o prioridad, pero no son adecuadas para textos de formato libre, valores dinámicos, números arbitrarios o identificadores de recursos. En esos casos, proporciona descripciones y ejemplos claros en su lugar. Incluya también un valor predeterminado cuando sea posible para que el LLM no tenga que adivinar cuál es la opción correcta. Tenga en cuenta que las definiciones de las herramientas se incluyen en el indicador LLM de cada invocación, lo que consume espacio en la ventana contextual junto con las instrucciones del sistema y el historial de conversaciones.

Enfoque de cadena de documentos

Otro enfoque, si está escribiendo sus herramientas en Python, es usar cadenas de documentación para proporcionar la descripción, el uso y el resultado de la herramienta. El siguiente es un ejemplo de este enfoque:

```
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
    """
```

Las cadenas de documentos no imponen un esquema o un formato estandarizado. El uso de este enfoque puede producir resultados incoherentes en función de la forma en que los desarrolladores de herramientas decidan documentar cada herramienta. Si se sigue este enfoque, es esencial definir y hacer cumplir un estándar que abarque a toda la organización.

Mejores prácticas para las definiciones de las herramientas MCP

- Siga las especificaciones de la herramienta MCP: proporcione `name` `descriptioninputSchema`, y `outputSchema` campos para cada herramienta. Para las implementaciones de Python, utilice los modelos de Pydantic para proporcionar documentación en línea mediante descripciones de campos, validación automática de tipos y valores restringidos mediante enumeraciones. Esto hace que los esquemas se documenten automáticamente y mejora la comprensión del LLM sobre las opciones de parámetros válidas.
- Escriba las descripciones siguiendo las instrucciones: las descripciones de las herramientas son instrucciones que guían la toma de decisiones de LLM. Incluya los componentes esenciales del propósito de la herramienta (qué hace la herramienta), cuándo usarla (patrones o escenarios de intención del usuario), el contexto del resultado (para qué se usa el resultado), los parámetros y las condiciones de error.
- Proporcione ejemplos concretos: incluir ejemplos de flujos de trabajo con valores reales es la forma más eficaz de orientar LLMs sobre el uso correcto de la herramienta.
- Documente las dependencias de forma explícita: incluya los requisitos previos, las secuencias numeradas, los cambios de estado y las acciones de seguimiento.

Descubrimiento de herramientas

Existen tres enfoques para detectar y registrar las herramientas de su agente con los servidores MCP: definición estática, descubrimiento dinámico y función de búsqueda.

Definición estática

En primer lugar, puede definir estáticamente las herramientas disponibles directamente en el código del agente. En este enfoque, se define una herramienta remota (un objeto de referencia del lado del cliente en un marco como Strands Agent SDK) para cada herramienta proporcionada por el servidor MCP a la que accede un cliente MCP. En el siguiente ejemplo, se utiliza un transporte HTTP que se puede reproducir en streaming:

```
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])
```

El registro individual de las herramientas le ayuda a ser muy selectivo con las herramientas que pone a disposición del LLM, lo que minimiza la cantidad de ventana de contexto utilizada. La desventaja es que requiere conocer los nombres de las herramientas disponibles y puede resultar frágil si las herramientas disponibles cambian en el servidor MCP.

Descubrimiento dinámico

El siguiente enfoque consiste en utilizar el descubrimiento dinámico y registrar todas las herramientas disponibles con el agente. Este enfoque consume el contexto de forma lineal a medida que se agregan más herramientas al servidor MCP. El siguiente es un ejemplo de este enfoque:

```
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)
```

Pensemos en un escenario en el que una definición de herramienta típica consume aproximadamente entre 250 y 500 fichas (incluidos el nombre, la descripción y el esquema). El registro de 20 herramientas consumiría entre 5000 y 10 000 fichas de la ventana de contexto. Si tiene un número reducido de servidores MCP y controla el número de herramientas, esta opción es la más sencilla de implementar. Sin embargo, si se espera que la lista de herramientas aumente, esto puede provocar problemas silenciosos de administración del contexto en sus agentes. Una variante alternativa de este enfoque consiste en utilizar un parámetro de filtro de herramientas al

llamar`list_tools`, como el que proporciona el SDK de Strands Agents, para reducir la cantidad de herramientas que están registradas en el agente.

Función de búsqueda

La tercera opción es utilizar una función de búsqueda para encontrar las herramientas relevantes durante el tiempo de ejecución. Enumera todas las herramientas disponibles en su servidor MCP y, a continuación, realiza una búsqueda semántica sobre esas herramientas en función de la solicitud del usuario. A continuación, las herramientas resultantes se registran en su agente. Amazon Bedrock AgentCore Gateway proporciona una capacidad de búsqueda semántica nativa que puede facilitar la implementación de este tipo de soluciones.

Prácticas recomendadas para el descubrimiento de herramientas MCP

- Preservación de la ventana de contexto: elija un enfoque de descubrimiento y registro de herramientas que conserve la mayor parte posible de la ventana de contexto.
- Utilice las funciones de filtrado de herramientas o búsqueda semántica: proporcione al LLM de forma dinámica un conjunto reducido de herramientas entre las que elegir, lo que mejora su precisión y eficacia a la hora de elegir la herramienta adecuada. El filtrado de herramientas puede funcionar con nombres de herramientas (patrones o coincidencias exactas), descripciones de herramientas (coincidencia semántica) o etiquetas de dominio o categoría. La búsqueda semántica es especialmente eficaz para comparar la intención de los usuarios con las descripciones de las herramientas. Ambos enfoques reducen el uso de la ventana de contexto.

Organización de herramientas

Descubrir las herramientas adecuadas y garantizar que el LLM pueda utilizarlas de forma eficaz es una de las partes más importantes del desarrollo de herramientas eficaces. Al empezar a desarrollar servidores MCP, necesitará una estrategia que determine:

- ¿Cuántas herramientas se incluyen en un servidor MCP?
- ¿Qué herramientas no deberían colocarse en el mismo servidor MCP
- Cómo nombrar las herramientas para que se puedan buscar y evitar colisiones de nombres (diferentes herramientas con el mismo nombre)
- ¿Cómo documentar las herramientas y el servidor MCP para que el LLM pueda utilizarlos fácilmente

La organización del espacio de nombres es un patrón de diseño que evita colisiones entre los nombres de las herramientas, agrupa las funcionalidades relacionadas y facilita la identificación eficiente de las herramientas mediante ellas. LLMs El patrón establece una categorización estructurada que es análoga a los sistemas de almacenamiento organizados y no a la acumulación no estructurada. Recomendamos el domain-noun-verbpatrón para la denominación de las herramientas. Por ejemplo, `github_issue_creategithub_issue_list`,`github_issue_update`,`github_pullrequest_creat` o`github_pullrequest_merge`. La ventaja de este patrón es evidente cuando se examina el comportamiento de la clasificación alfabética. Cuando las herramientas aparecen en orden alfabético, todas las operaciones relacionadas con el problema se agrupan (`create`,,`update`)`list`, seguidas de las operaciones de solicitud de extracción (,,). `create` `list` `merge` El sustantivo (tipo de recurso) funciona como un límite organizativo. Esta estructura facilita tanto el escaneo de la herramienta LLM como la navegación por la documentación humana, ya que las funciones relacionadas se agrupan de forma natural.

El servidor MCP debe estar limitado a nivel de dominio, pero puede subdividirse en función de la separación de funciones según las capacidades que proporciona. Por ejemplo, es posible que tenga servidores MCP independientes para las operaciones de escritura y lectura en una base de datos. Para hacer cumplir esta separación, se recomienda implementar barreras a nivel de agente que restrinjan los servidores MCP a los que se puede acceder en función de la intención y los permisos del usuario. Esto se puede lograr mediante una combinación de lo siguiente:

- Carga condicional del servidor: cargue el servidor MCP de solo lectura solo cuando el agente detecte operaciones de lectura en la entrada del usuario.
- Filtrado basado en permisos: utilice la autorización del usuario para conceder acceso únicamente a los servidores MCP adecuados.

Por último, querrá establecer un límite máximo en el número de herramientas que proporciona un servidor MCP. No haga suposiciones sobre cómo utilizarán los agentes su servidor MCP. Pueden enumerar ingenuamente todas las herramientas disponibles y proporcionárselas todas al LLM. Si tiene más de 50 herramientas en un solo servidor, debería considerar dividirlas en varios servidores.

Mejores prácticas para la organización de las herramientas de MPC

- Utilice el estándar de domain-noun-verb nomenclatura para las herramientas: implemente estrategias para evitar colisiones de nombres tanto en los servidores MCP como en los agentes.
- Establezca un límite superior: restrinja la cantidad de herramientas en un solo servidor MCP.

- Divida los servidores MCP: utilice la separación de funciones para dividir los servidores MCP en grupos lógicos.

Estrategia de alojamiento de MCP

Al resumir las herramientas disponibles en servidores MCP, el desarrollo de agentes se desvincula del desarrollo de los agentes de las herramientas disponibles. Esto presenta los desafíos relacionados con el lugar donde se aloja el servidor MCP y la forma en que se organizan las herramientas dentro de esos servidores.

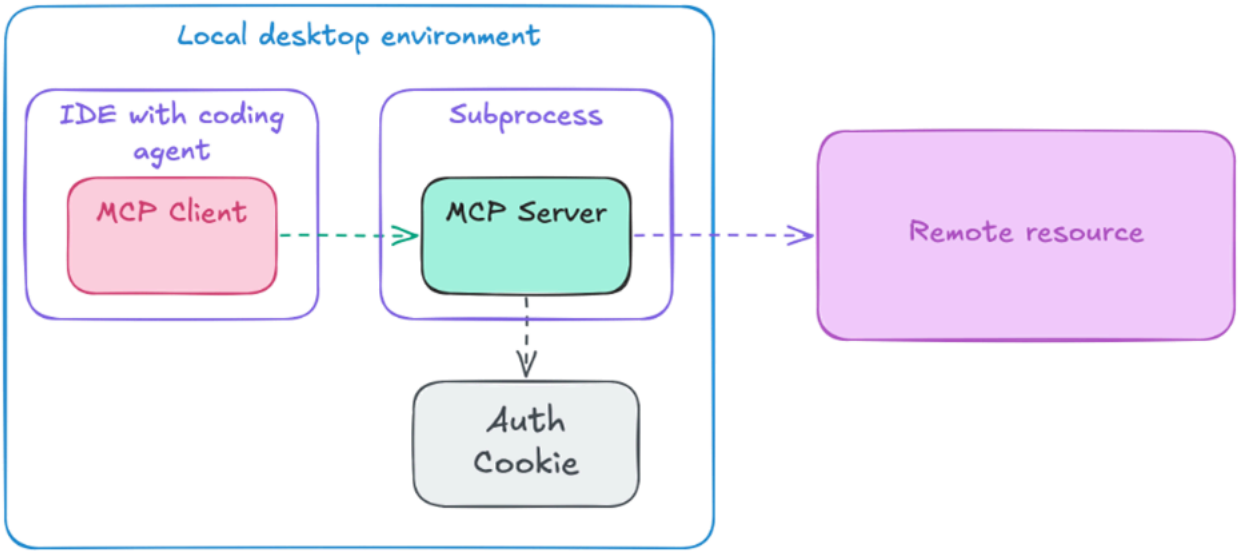
Enfoques de alojamiento

Existen tres opciones para alojar sus servidores MCP: ejecutarlos localmente en la máquina de un usuario final, alojarlos de forma remota o alojarlos a través de una puerta de enlace MCP. Cada opción tiene ventajas y desventajas.

Alojamiento local

El alojamiento local ejecuta el servidor MCP como un subproceso en su máquina local junto con el agente que se comunica con el servidor mediante JSON-RPC a través de flujos de entrada y salida estándar. Este enfoque no requiere autenticación entre el cliente y el servidor. Las herramientas pueden interactuar con aplicaciones y archivos locales, utilizar credenciales almacenadas localmente y heredar el acceso a la red de la máquina local del usuario. Este es el patrón de alojamiento más simple y tiene varias ventajas.

Muchos clientes comienzan a utilizar MCP utilizando servidores locales. Permiten a los ingenieros iterar y resolver rápidamente una variedad de problemas desde su entorno local. Pensemos en un servidor MCP que se conecta a un repositorio de Git que utiliza el asistente de programación de un ingeniero. Mantener el servidor MCP local tiene mucho sentido porque permite utilizar las credenciales exclusivas del ingeniero para acceder al repositorio y no añade una llamada de red adicional a un servidor MCP remoto. La siguiente imagen muestra un servidor MCP alojado localmente que se utiliza con un agente de codificación en un IDE.



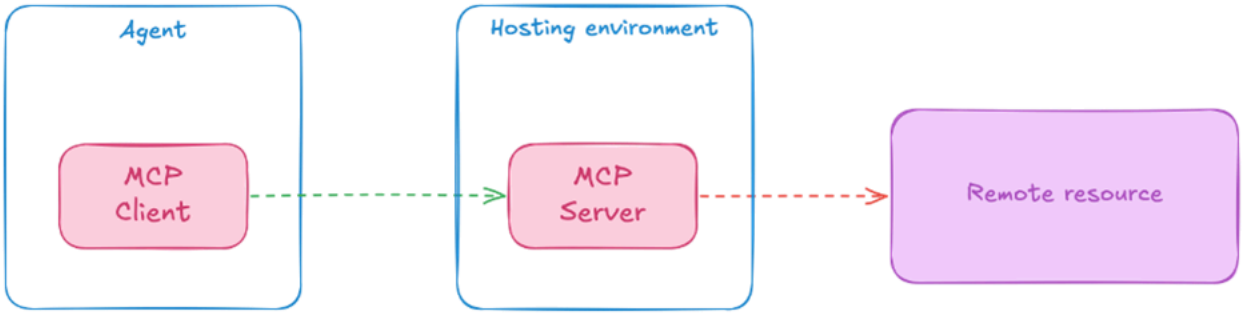
Para estos tipos de implementaciones, debe tener en cuenta cómo se desarrollan y distribuyen los servidores MCP. La mayoría de los clientes desarrollan un registro MCP en el que los usuarios finales pueden registrar y descargar los servidores. Es muy similar a un registro de contenedores, en el que un usuario puede buscar capacidades específicas y encontrar los servidores MCP que mejor se adapten a sus necesidades.

Hay registros MCP públicos, como el Registro MCP oficial, y registros alojados de forma privada. Las organizaciones suelen alinear su estrategia de registro de MCP con las políticas existentes en torno a la distribución de software de código abierto, los registros de contenedores y la administración interna de paquetes. Debe tener en cuenta factores como el análisis de seguridad, los flujos de trabajo de aprobación y los requisitos de conformidad.

Sin embargo, el alojamiento local presenta desafíos operativos que las organizaciones deberían tener en cuenta. En primer lugar, los usuarios finales deben descubrir, descargar y configurar los servidores MCP de forma independiente. Esto puede añadir complejidad a la hora de empezar con cada servidor MCP individual que utilizan de forma local. En segundo lugar, no se puede controlar el ciclo de vida del servidor MCP, lo que significa que los usuarios pueden seguir ejecutando versiones anticuadas de forma local con vulnerabilidades de seguridad o con funciones ausentes. Esto puede complicar el cumplimiento de los requisitos de conformidad. Algunas IDEs herramientas CLI, como Kiro, permiten a las organizaciones administrar y controlar qué herramientas de MCP están disponibles, lo que garantiza la coherencia y la seguridad en todos los equipos.

Alojamiento remoto

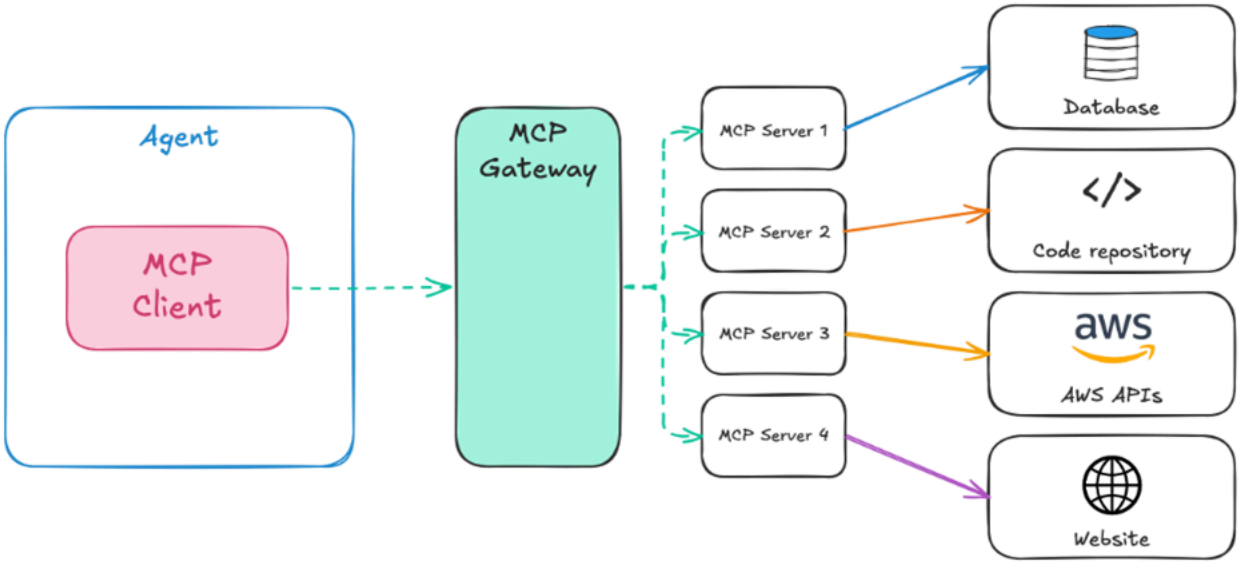
La segunda opción es alojar servidores MCP remotos a los que se accede a través de HTTP o HTTPS. Esto proporciona acceso a cualquier cliente conectado a la red. El uso del alojamiento remoto le permite controlar de forma centralizada el acceso a los recursos y capacidades del MCP, implementar la autenticación y la autorización y controlar el control de versiones y las actualizaciones de la lógica del servidor MCP. El alojamiento remoto aún requiere el uso de un registro MCP para que los usuarios finales puedan descubrir los servidores MCP que desean usar con su agente. La siguiente imagen muestra el enfoque de alojamiento remoto.



Desde la perspectiva del desarrollo de agentes, la experiencia es similar tanto si el servidor MCP es local como remoto. El cambio más significativo es la implementación de la autenticación y la autorización, que incluyen tanto el acceso del agente al servidor MCP como el acceso del servidor a los recursos externos. Las implementaciones de servidores MCP remotos deben planificarse cuidadosamente para tener en cuenta el acceso multiusuario y la administración de privilegios. El capítulo sobre la estrategia de gobierno del MCP contiene más información sobre las consideraciones de autenticación y autorización.

Puerta de enlace MCP

La última opción es usar una puerta de enlace MCP. Las pasarelas MCP actúan como un proxy centralizado entre los clientes y los servidores MCP y organizan el acceso a los servidores MCP registrados. Sin una puerta de enlace, cada agente debe registrar todos los servidores MCP remotos que desee utilizar. Una puerta de enlace permite al agente conectarse a un único punto final que administra la autenticación, la autorización, el enrutamiento y la traducción de protocolos. Los nuevos servidores y herramientas de MCP se pueden añadir de forma dinámica y ponerlos a disposición del agente de forma inmediata. La siguiente imagen muestra el enfoque de la puerta de enlace MCP.



Algunas soluciones de puerta de enlace, como la puerta de enlace MCP de Docker, también gestionan el ciclo de vida de los servidores MCP y lanzan los servidores a pedido según sea necesario. Las pasarelas MCP, como Amazon Bedrock AgentCore Gateway, también pueden ayudar a gestionar el descubrimiento de herramientas al proporcionar capacidades de búsqueda semántica nativas. Esto proporciona a los agentes un único punto final para conectarse con un cliente MCP y ayuda a optimizar su uso de la ventana de contexto. El resultado son agentes sencillos que pueden elegir y utilizar las herramientas de MCP de forma eficaz. Sin embargo, presenta desafíos relacionados con la identidad similares a los del enfoque de servidor MCP remoto.

Mejores prácticas para alojar servidores MCP

- La gama de opciones de alojamiento no es única para todos. Gran parte del uso de los servidores MCP en la actualidad es local.
- Al empezar a utilizar servidores MCP remotos, lo principal que debe tener en cuenta es la autenticación y la autorización coherentes del servidor MCP y la forma en que el servidor MCP lleva a cabo la autenticación y la autorización de los recursos intermedios.
- Las puertas de enlace MCP simplifican la conectividad, la autenticación y la autorización para alojar varios servidores MCP remotos. También proporcionan capacidades para mejorar la administración de las ventanas contextuales mediante la búsqueda de las herramientas aplicables.

Estrategia de gobierno de MCP

La otra capacidad fundamental que ofrece el MCP a las organizaciones es el apoyo a la gobernanza centralizada. Su estrategia de gobierno del MCP debe abordar la autenticación y la autorización tanto de los servidores del MCP como de los recursos a los que acceden. También debe abordar la limitación de la velocidad para proteger los recursos intermedios, las métricas operativas para monitorear el uso y el rendimiento de las herramientas y la administración de las implementaciones y la distribución de los servidores MCP.

Autenticación y autorización

Una de las partes más importantes de su estrategia de autenticación y autorización es administrar el acceso descendente a los recursos desde los servidores MCP. Cuando un usuario llama a un agente, se realizan la autenticación y la autorización para garantizar que el usuario tenga permisos para llamar al agente. Luego, el agente organiza las llamadas a herramientas específicas en los servidores MCP. Debe decidir cómo autorizar el acceso por herramienta.

Una opción es la machine-to-machine autorización, en la que no se requiere el consentimiento o la interacción del usuario. Por ejemplo, la invocación de un agente basada en el tiempo utiliza un servidor MCP para recopilar los registros de una aplicación y analizarlos. En este escenario, el agente está preautorizado para acceder a los datos especificados. La segunda opción es el acceso delegado por el usuario, en el que el usuario da su consentimiento para acceder a datos y recursos específicos del usuario.

La siguiente tabla muestra los patrones de autenticación y autorización.

Factor	Acceso delegado por el usuario	Machine-to-machine
Propiedad de los datos	Autorización de datos específica del usuario	Datos de todo el sistema o la organización
Interacción con el usuario	El usuario está presente y puede dar su consentimiento	Sin interacción con el usuario
Tiempo de operación	Interactivo o en tiempo real	En segundo plano, programad o o por lotes

Ámbito de los permisos	Los permisos varían según el usuario	Permisos uniformes a nivel de agente

El acceso delegado por el usuario requiere una implementación cuidadosa y debe desarrollarse con su equipo de seguridad. Los agentes deben poder evaluar qué herramientas ha seleccionado un LLM y si requieren una autorización adicional. Las herramientas de MCP deben incluir descripciones para indicar sus requisitos de autenticación y autorización y dónde recuperar los tokens de acceso. Las aplicaciones cliente deben admitir las solicitudes de autenticación intermedia y el cliente MCP debe devolver las credenciales recuperadas al agente cada vez que utilice la herramienta.

Debe asegurarse de que las herramientas de MCP siempre tengan sus propios tokens para acceder a las capacidades externas y de que el acceso esté registrado y auditado. Las credenciales de usuario no deben propagarse a través del sistema de su agencia. Por ejemplo, sus servidores MCP no deberían usar el mismo token para acceder a los datos que se utilizó para invocar al agente. Las llamadas posteriores deben usar tokens con un alcance explícito y generados con un propósito específico. Esto ayuda a proporcionar barreras adicionales para evitar el acceso no deseado a los datos en nombre de las acciones. También puede ayudar a evitar que las alucinaciones produzcan resultados no deseados. Imagine que un usuario con todos los permisos de administrador pide a un agente que clone una base de datos de producción para utilizarla en la fase de preproducción. Para ello, el usuario solo necesita `CREATE` permisos `READ` y permisos. Supongamos que el LLM alucina y cree que necesita limpiar la antigua base de datos como parte de esta solicitud. Si reutiliza las credenciales del usuario, es probable que lo consiga porque las credenciales originales del usuario tienen permisos. `DELETE` Por el contrario, si el servidor MCP utiliza un token cuyo alcance se ha reducido intencionadamente para la solicitud, solo con `CREATE` permisos `READ` y permisos, no se podrá eliminar la base de datos de producción.

Puede utilizar Amazon Bedrock AgentCore Identity para ayudar a implementar estos patrones. Asegúrese de elegir intencionalmente si los permisos para enumerar e invocar las herramientas alojadas en un servidor MCP implican el uso de las capacidades externas que expone el servidor MCP. Este flujo de identidad desde el servidor MCP al recurso y de vuelta al usuario depende del tipo de servicio de autenticación y autorización que se utilice. Debe decidir cómo se gestiona esto a escala para sus servidores MCP.

Al diseñar sus patrones de autenticación y autorización, implemente mecanismos de aislamiento de tokens que recuperen diferentes tokens de acceso para cada herramienta a la que se acceda. No reutilices los tokens entre herramientas y servidores. AgentCore La identidad proporciona esta capacidad de aislamiento de tokens. Administra automáticamente tanto los tokens de carga de

trabajo (para la machine-to-machine autenticación) como los de usuario (para el acceso delegado por el usuario) para garantizar una separación adecuada y evitar la escalada de permisos. Esto es especialmente importante cuando se incorporan servidores MCP remotos o puertas de enlace MCP.

Mejores prácticas para la autenticación y autorización de MCP

- Separación de fichas: no transfiera las fichas portadoras de las personas que llaman a los servicios intermedios. Comprueba que el campo aud (audiencia) coincida con el servidor que recibe el token. La afirmación de audiencia especifica a qué servicio está destinado el token, lo que impide la reutilización no autorizada del token en diferentes servidores MCP.
- Seleccione un enfoque de acceso: elija entre machine-to-machine un acceso delegado por el usuario para cada herramienta que proporcionen sus servidores MCP. Considere la posibilidad de agrupar las herramientas en el mismo servidor MCP que utilicen el mismo patrón de autenticación.

Controlar la carga

Al igual que con cualquier sistema distribuido, debe considerar cómo controlar la carga en su flota de servidores MCP. En primer lugar, considere si debe implementar la limitación de velocidad en sus servidores MCP y dónde implementarla. Si decide no implementar la limitación de velocidad, transferirá cualquier limitación de velocidad aplicada por los recursos descendentes. Muchos sistemas optan por limitar la tasa en función de los atributos de la solicitud, como el identificador de usuario o de cuenta. Compruebe que las solicitudes enviadas a los servicios descendentes tengan los mismos atributos para que varios usuarios no se vean afectados por la carga generada por otro usuario.

Si opta por implementar la limitación de velocidad, el enfoque recomendado es implementar la limitación de velocidad principal a nivel del servidor MCP, con servicios de backend que proporcionen protección secundaria y que los agentes adapten su comportamiento en función de los comentarios sobre el límite de velocidad. Considere si los límites de velocidad son por servidor MCP o por herramienta. Los límites de velocidad por servidor MCP ayudan a proteger su flota de servidores MCP y sus servicios en un entorno multiusuario. Sin embargo, eso puede ser muy grosero. Los límites de velocidad por herramienta están diseñados para evitar que los recursos intermedios se abrumen y que tal vez no se limiten lo suficiente por sí mismos. Si una herramienta realiza varias llamadas APIs, debe establecer el límite de frecuencia para que se ajuste a la frecuencia más baja permitida por esas herramientas. APIs

La transmisión de la información sobre el límite de frecuencia en los encabezados HTTP también puede ser una métrica útil para los usuarios y los sistemas automatizados, ya que les ayuda a gestionar su propia tasa de solicitudes y su estrategia de reintentos. Por ejemplo, puede enviar estos encabezados al agente desde su servidor MCP, como se muestra en el siguiente ejemplo:

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

Además, considere la posibilidad de reducir la carga para proteger el servicio en general cuando ningún cliente supere un límite de velocidad pero la carga afecte al rendimiento del sistema.

Mejores prácticas para controlar la carga

- Elija un enfoque de limitación de velocidad: planifique limitar la tarifa de los usuarios individuales en función del uso que hagan de los recursos intermedios o del servidor y las herramientas de MCP.
- Considere la posibilidad de reducir la carga: proteja su flota de servidores MCP de una sobrecarga general que no esté provocada por un solo cliente o por un puñado de clientes.

Métricas operativas

Las métricas clave que se deben recopilar para las implementaciones de MCP deben centrarse en la experiencia de cliente que ofrecen. Estas métricas suelen incluir el uso de los tokens, la precisión de la selección de herramientas, la cantidad de herramientas registradas en el agente y la latencia de las herramientas. Por ejemplo, la supervisión de los tokens de salida devueltos por cada herramienta permite configurar alarmas cuando las herramientas superan un umbral de uso de la ventana de contexto. Cuando una herramienta supera ese umbral, es posible que desee revisar el comportamiento de la herramienta. Esto también se relaciona con la estrategia de diseño de la herramienta MCP. Las métricas de precisión de la selección de herramientas indican qué tan bien los agentes eligen las herramientas adecuadas para determinadas tareas, mientras que la velocidad de ejecución y las tasas de éxito destacan los cuellos de botella en el rendimiento y los problemas de fiabilidad.

Por ejemplo, para evaluar las métricas de precisión de la selección y el uso de las herramientas, los AWS equipos crearon conjuntos de datos básicos para las pruebas de regresión. Los conjuntos de datos se generaron sintéticamente a partir de registros históricos de invocación de la API tras las

consultas LLMs de los usuarios. Con las métricas predefinidas de selección y uso de herramientas (como la precisión de la selección de herramientas, la precisión de los parámetros de la herramienta y la precisión de las llamadas a funciones de varios turnos), AWS los equipos pudieron evaluar objetivamente la capacidad del agente de IA para identificar correctamente las herramientas adecuadas, rellenar sus parámetros con valores precisos y mantener secuencias de invocación de herramientas coherentes en los turnos de conversación.

Medir las métricas sobre la cantidad de herramientas registradas en un agente puede ayudarlo a identificar los posibles desafíos de administración de ventanas de contexto, así como los cambios en las herramientas disponibles que presentan los servidores MCP. Debe revisar periódicamente las métricas operativas que indican la experiencia del usuario con el servidor y las herramientas de MCP.

Colaboradores

Creación

- Alex Torres, arquitecto sénior de soluciones, AWS
- Saikat Gomes, gerente sénior de soluciones para clientes, AWS
- Mike Haken, arquitecto principal sénior de soluciones, AWS
- Sreeja Das, ingeniera principal, AWS

Revisión

- Ted Swinyar, director de arquitectura de soluciones, AWS
- Raju Patil, científico de datos sénior, AWS

Redacción técnica

- Lilly AbouHarb, escritora técnica sénior, AWS

Historial de documentos

En la siguiente tabla, se describen cambios significativos de esta guía. Si quiere recibir notificaciones de futuras actualizaciones, puede suscribirse a las notificaciones RSS.

Cambio	Descripción	Fecha
Publicación inicial	—	16 de marzo de 2026

AWS Glosario de orientación prescriptiva

Los siguientes son términos de uso común en las estrategias, guías y patrones proporcionados por la Guía AWS prescriptiva. Para sugerir entradas, utilice el enlace Enviar comentarios al final del glosario.

Números

Las 7 R

Siete estrategias de migración comunes para trasladar aplicaciones a la nube. Estas estrategias se basan en las 5 R que Gartner identificó en 2011 y consisten en lo siguiente:

- Refactor/re-architect — Mueva una aplicación y modifique su arquitectura aprovechando al máximo las funciones nativas de la nube para mejorar la agilidad, el rendimiento y la escalabilidad. Por lo general, esto implica trasladar el sistema operativo y la base de datos. Ejemplo: migre su base de datos Oracle local a la PostgreSQL-Compatible edición Amazon Aurora.
- Redefinir la plataforma (transportar y redefinir): traslade una aplicación a la nube e introduzca algún nivel de optimización para aprovechar las capacidades de la nube. Ejemplo: Migrar la base de datos Oracle en las instalaciones a Amazon Relational Database Service (Amazon RDS) para Oracle en la nube de Nube de AWS.
- Recomprar (readquirir): cambie a un producto diferente, lo cual se suele llevar a cabo al pasar de una licencia tradicional a un modelo SaaS. Ejemplo: migre su sistema de gestión de relaciones con los clientes (CRM) a Salesforce.com.
- Volver a alojar (migrar mediante lift-and-shift): traslade una aplicación a la nube sin hacer cambios para aprovechar las funcionalidades de la nube. Ejemplo: Migrar la base de datos de Oracle en las instalaciones a Oracle en una instancia de EC2 en la Nube de AWS.
- Reubicar: (migrar el hipervisor mediante lift and shift): traslade la infraestructura a la nube sin comprar equipo nuevo, reescribir aplicaciones o modificar las operaciones actuales. Los servidores se migran de una plataforma en las instalaciones a un servicio en la nube para la misma plataforma. Ejemplo: migrar una Microsoft Hyper-V aplicación a AWS.
- Retener (revisitar): conserve las aplicaciones en el entorno de origen. Estas pueden incluir las aplicaciones que requieren una refactorización importante, que desee posponer para más adelante, y las aplicaciones heredadas que desee retener, ya que no hay ninguna justificación empresarial para migrarlas.

- Retirar: retire o elimine las aplicaciones que ya no sean necesarias en un entorno de origen.

A

A2A () Agent-to-Agent

Un protocolo completo para la colaboración entre agentes que facilita la delegación de tareas y la transferencia de estados.

ABAC

Consulte control de acceso basado en atributos.

servicios abstractos

Consulte servicios administrados.

ACID

Consulte atomicidad, consistencia, aislamiento, durabilidad.

migración activa-activa

Método de migración de bases de datos en el que las bases de datos de origen y destino se mantienen sincronizadas (mediante una herramienta de replicación bidireccional o mediante operaciones de escritura doble) y ambas bases de datos gestionan las transacciones de las aplicaciones conectadas durante la migración. Este método permite la migración en lotes pequeños y controlados, en lugar de requerir una transición única. Es más flexible, pero requiere más trabajo que una migración activa-pasiva.

migración activa-pasiva

Método de migración de bases de datos en el que las bases de datos de origen y destino se mantienen sincronizadas, pero solo la de origen gestiona las transacciones de las aplicaciones conectadas, mientras los datos se replican en la de destino. La base de datos de destino no acepta ninguna transacción durante la migración.

Agente

Un sistema de IA que puede razonar, planificar y tomar medidas de forma autónoma utilizando herramientas para alcanzar los objetivos.

Agent Ops

Prácticas operativas para crear, probar, implementar y ejecutar agentes de IA en producción a escala.

función de agregación

Función SQL que actúa en un grupo de filas y calcula un único valor de devolución para el grupo. Entre los ejemplos de funciones de agregación se incluyen `SUM` y `MAX`.

IA

Consulte inteligencia artificial.

AIOps

Consulte operaciones de inteligencia artificial

anonimización

El proceso de eliminar permanentemente la información personal de un conjunto de datos. La anonimización puede ayudar a proteger la privacidad personal. Los datos anonimizados ya no se consideran datos personales.

antipatrones

Una solución que se utiliza con frecuencia para un problema recurrente en el que la solución es contraproducente, ineficaz o menos eficaz que una alternativa.

control de aplicaciones

Enfoque de seguridad que permite usar de manera exclusiva aplicaciones aprobadas para ayudar a proteger un sistema contra el malware.

cartera de aplicaciones

Recopilación de información detallada sobre cada aplicación que utiliza una organización, incluido el costo de creación y mantenimiento de la aplicación y su valor empresarial. Esta información es clave para el proceso de detección y análisis de la cartera y ayuda a identificar y priorizar las aplicaciones que se van a migrar, modernizar y optimizar.

inteligencia artificial (IA)

El campo de la informática que se dedica al uso de tecnologías informáticas para realizar funciones cognitivas que suelen estar asociadas a los seres humanos, como el aprendizaje, la resolución de problemas y el reconocimiento de patrones. Para más información, consulte ¿Qué es la inteligencia artificial?

operaciones de inteligencia artificial (AIOps)

El proceso de utilizar técnicas de machine learning para resolver problemas operativos, reducir los incidentes operativos y la intervención humana, y mejorar la calidad del servicio. Para obtener más información sobre cómo se utiliza AIOps en la estrategia de migración de AWS , consulte la Guía de integración de operaciones.

cifrado asimétrico

Algoritmo de cifrado que utiliza un par de claves, una clave pública para el cifrado y una clave privada para el descifrado. Puede compartir la clave pública porque no se utiliza para el descifrado, pero el acceso a la clave privada debe estar sumamente restringido.

atomicidad, consistencia, aislamiento, durabilidad (ACID)

Conjunto de propiedades de software que garantizan la validez de los datos y la fiabilidad operativa de una base de datos, incluso en caso de errores, cortes de energía u otros problemas.

control de acceso basado en atributos (ABAC)

La práctica de crear permisos detallados basados en los atributos del usuario, como el departamento, el puesto de trabajo y el nombre del equipo. Para obtener más información, consulte ABAC AWS en la documentación AWS Identity and Access Management (IAM).

origen de datos fidedigno

Ubicación en la que se almacena la versión principal de los datos, que se considera la fuente de información más fiable. Puede copiar los datos del origen de datos autorizado a otras ubicaciones con el fin de procesarlos o modificarlos, por ejemplo, anonimizarlos, redactarlos o seudonimizarlos.

Zona de disponibilidad

Una ubicación distinta dentro de una Región de AWS que está aislada de los fallos en otras zonas de disponibilidad y que proporciona una conectividad de red económica y de baja latencia a otras zonas de disponibilidad de la misma región.

AWS Marco de adopción de la nube (AWS CAF)

Un marco de directrices y mejores prácticas AWS para ayudar a las organizaciones a desarrollar un plan eficiente y eficaz para migrar con éxito a la nube. AWS CAF organiza la orientación en seis áreas de enfoque denominadas perspectivas: negocios, personas, gobierno, plataforma, seguridad y operaciones. Las perspectivas empresariales, humanas y de gobernanza se centran en las habilidades y los procesos empresariales; las perspectivas de plataforma, seguridad y

operaciones se centran en las habilidades y los procesos técnicos. Por ejemplo, la perspectiva humana se dirige a las partes interesadas que se ocupan de los Recursos Humanos (RR. HH.), las funciones del personal y la administración de las personas. Desde esta perspectiva, AWS CAF proporciona orientación para el desarrollo, la formación y la comunicación de las personas a fin de preparar a la organización para una adopción exitosa de la nube. Para obtener más información, consulte la Página web de AWS CAF y el Documento técnico de AWS CAF.

AWS Marco de calificación de la carga de trabajo (AWS WQF)

Herramienta que evalúa las cargas de trabajo de migración de bases de datos, recomienda estrategias de migración y proporciona estimaciones de trabajo. AWS WQF se incluye con AWS Schema Conversion Tool ().AWS SCT Analiza los esquemas de bases de datos y los objetos de código, el código de las aplicaciones, las dependencias y las características de rendimiento y proporciona informes de evaluación.

B

bot malicioso

Bot destinado a causar interrupciones o daños a personas u organizaciones.

BCP

Consulte planificación de la continuidad del negocio.

gráfico de comportamiento

Una vista unificada e interactiva del comportamiento de los recursos y de las interacciones a lo largo del tiempo. Puede utilizar un gráfico de comportamiento con Amazon Detective para examinar los intentos de inicio de sesión fallidos, las llamadas sospechosas a la API y acciones similares. Para obtener más información, consulte Datos en un gráfico de comportamiento en la documentación de Detective.

sistema big-endian

Un sistema que almacena primero el byte más significativo. Consulte también endianidad.

clasificación binaria

Un proceso que predice un resultado binario (una de las dos clases posibles). Por ejemplo, es posible que su modelo de ML necesite predecir problemas como “¿Este correo electrónico es spam o no es spam?” o “¿Este producto es un libro o un automóvil?”.

filtro de floración

Estructura de datos probabilística y eficiente en términos de memoria que se utiliza para comprobar si un elemento es miembro de un conjunto.

blue/green despliegue

Estrategia de implementación en la que se crean dos entornos separados, pero idénticos. La versión actual de la aplicación se ejecuta en un entorno (azul) y la nueva versión de la aplicación se ejecuta en el otro entorno (verde). Esta estrategia lo ayuda a hacer reversiones rápidas con un impacto mínimo.

bot

Aplicación de software que ejecuta tareas automatizadas a través de Internet y simula la actividad o interacción humana. Algunos bots son útiles o beneficiosos, como los rastreadores web que indexan la información de Internet. Otros bots, conocidos como bots maliciosos, tienen como objetivo causar interrupciones o daños a personas u organizaciones.

botnet

Redes de bots infectadas por malware y que están bajo el control de una sola parte, conocida como pastor de bots u operador de bots. Las botnets son el mecanismo más conocido para escalar los bots y su impacto.

branch

Área contenida de un repositorio de código. La primera rama que se crea en un repositorio es la rama principal. Puede crear una rama nueva a partir de una rama existente y, a continuación, desarrollar características o corregir errores en la rama nueva. Una rama que se genera para crear una característica se denomina comúnmente rama de característica. Cuando la característica se encuentra lista para su lanzamiento, se vuelve a combinar la rama de característica con la rama principal. Para obtener más información, consulte Acerca de las sucursales (GitHub documentación).

acceso de emergencia

En circunstancias excepcionales y mediante un proceso aprobado, es una forma rápida de que un usuario pueda acceder a un Cuenta de AWS sitio al que normalmente no tiene permisos de acceso. Para obtener más información, consulte el indicador de implementación de procedimientos rompe-cristales en la AWS Well-Architected guía.

estrategia de implementación sobre infraestructura existente

La infraestructura existente en su entorno. Al adoptar una estrategia de implementación sobre infraestructura existente para una arquitectura de sistemas, se diseña la arquitectura en función de las limitaciones de los sistemas y la infraestructura actuales. Si está ampliando la infraestructura existente, puede combinar las estrategias de implementación sobre infraestructuras existentes y de implementación desde cero.

caché de búfer

El área de memoria donde se almacenan los datos a los que se accede con más frecuencia.

capacidad empresarial

Lo que hace una empresa para generar valor (por ejemplo, ventas, servicio al cliente o marketing). Las arquitecturas de microservicios y las decisiones de desarrollo pueden estar impulsadas por las capacidades empresariales. Para obtener más información, consulte la sección Organizado en torno a las capacidades empresariales del documento técnico Ejecutar microservicios en contenedores en AWS.

planificación de la continuidad del negocio (BCP)

Plan que aborda el posible impacto de un evento disruptivo, como una migración a gran escala en las operaciones y permite a la empresa reanudar las operaciones rápidamente.

C

CAF

Consulte AWS Cloud Adoption Framework.

implementación canario

Lanzamiento lento e incremental de una versión para los usuarios finales. Cuando tenga mayor confianza en la nueva versión, la implementa y reemplaza la versión actual en su totalidad.

CCoE

Consulte Centro de excelencia en la nube.

CDC

Consulte captura de datos de cambios.

captura de datos de cambio (CDC)

Proceso de seguimiento de los cambios en un origen de datos, como una tabla de base de datos, y registro de los metadatos relacionados con el cambio. Puede utilizar los CDC para diversos fines, como auditar o replicar los cambios en un sistema de destino para mantener la sincronización.

ingeniería del caos

Introducción intencionada de fallos o eventos disruptivos para poner a prueba la resiliencia de un sistema. Puedes usar AWS Fault Injection Service (AWS FIS) para realizar experimentos que estresen tus AWS cargas de trabajo y evalúen su respuesta.

CI/CD

Consulte integración continua y entrega continua.

clasificación

Un proceso de categorización que permite generar predicciones. Los modelos de ML para problemas de clasificación predicen un valor discreto. Los valores discretos siempre son distintos entre sí. Por ejemplo, es posible que un modelo necesite evaluar si hay o no un automóvil en una imagen.

Desarrollador ciudadano

Un usuario empresarial que crea aplicaciones de IA utilizando plataformas sin code/low código sin conocimientos técnicos especializados.

cifrado del cliente

Cifrado de datos localmente, antes de que el objetivo los Servicio de AWS reciba.

Centro de excelencia en la nube (CCoE)

Equipo multidisciplinario que impulsa los esfuerzos de adopción de la nube en toda la organización, incluido el desarrollo de las prácticas recomendadas en la nube, la movilización de recursos, el establecimiento de plazos de migración y la dirección de la organización durante las transformaciones a gran escala. Para obtener más información, consulte las publicaciones de CCoE en el blog de estrategia Nube de AWS empresarial.

computación en la nube

La tecnología en la nube que se utiliza normalmente para la administración de dispositivos de IoT y el almacenamiento de datos de forma remota. La computación en la nube suele estar relacionada con la tecnología de computación de periferia.

modelo operativo en la nube

En una organización de TI, el modelo operativo que se utiliza para crear, madurar y optimizar uno o más entornos de nube. Para obtener más información, consulte Creación de su modelo operativo de nube.

etapas de adopción de la nube

Las siguientes son las cuatro fases por las que suelen pasar las empresas cuando migran a la Nube de AWS:

- Proyecto: ejecución de algunos proyectos relacionados con la nube con fines de prueba de concepto y aprendizaje
- Fundamento: realización de inversiones fundamentales para escalar la adopción de la nube (p. ej., crear una zona de aterrizaje, definir un CCoE, establecer un modelo de operaciones)
- Migración: migración de aplicaciones individuales
- Re-invention — Optimizar los productos y servicios e innovar en la nube

Stephen Orban definió estas etapas en la entrada del blog The Journey Toward Cloud-First & the Stages of Adoption del blog Nube de AWS Enterprise Strategy. Para obtener información sobre su relación con la estrategia de AWS migración, consulte la guía de preparación para la migración.

CMDB

Consulte base de datos de administración de configuración.

repositorio de código

Una ubicación donde el código fuente y otros activos, como documentación, muestras y scripts, se almacenan y actualizan mediante procesos de control de versiones. Algunos repositorios en la nube comunes son GitHub o Bitbucket Cloud. Cada versión del código se denomina rama. En una estructura de microservicios, cada repositorio se encuentra dedicado a una única funcionalidad. Una sola CI/CD canalización puede utilizar varios repositorios.

caché en frío

Una caché de búfer que está vacía no está bien poblada o contiene datos obsoletos o irrelevantes. Esto afecta al rendimiento, ya que la instancia de la base de datos debe leer desde la memoria principal o el disco, lo que es más lento que leer desde la memoria caché del búfer.

datos fríos

Datos a los que se accede con poca frecuencia y que suelen ser históricos. Al consultar este tipo de datos, normalmente se aceptan consultas lentas. Trasladar estos datos a niveles o clases de almacenamiento de menor rendimiento y menos costosos puede reducir los costos.

visión artificial (CV)

Campo de la IA que utiliza el machine learning para analizar y extraer información de formatos visuales, como imágenes y videos digitales. Por ejemplo, Amazon SageMaker AI proporciona algoritmos de procesamiento de imágenes para CV.

deriva de configuración

En el caso de una carga de trabajo, un cambio en la configuración con respecto al estado esperado. Podría provocar que la carga de trabajo deje de cumplir las normas y, por lo general, es gradual e involuntaria.

base de datos de administración de configuración (CMDB)

Repositorio que almacena y administra información sobre una base de datos y su entorno de TI, incluidos los componentes de hardware y software y sus configuraciones. Por lo general, los datos de una CMDB se utilizan en la etapa de detección y análisis de la cartera de productos durante la migración.

paquete de conformidad

Un conjunto de AWS Config reglas y medidas correctivas que puede reunir para personalizar sus controles de conformidad y seguridad. Puede implementar un paquete de conformidad como una entidad única en una región Cuenta de AWS y, o en una organización, mediante una plantilla YAML. Para obtener más información, consulta los paquetes de conformidad en la documentación. AWS Config

integración y entrega continuas () CI/CD

El proceso de automatización de las etapas de origen, creación, prueba, puesta en escena y producción del proceso de publicación del software. CI/CD se describe comúnmente como una canalización. CI/CD puede ayudarlo a automatizar los procesos, mejorar la productividad, mejorar la calidad del código y entregar más rápido. Para obtener más información, consulte Beneficios de la entrega continua. CD también puede significar implementación continua. Para obtener más información, consulte Entrega continua frente a implementación continua.

CV

Consulte visión artificial.

D

datos en reposo

Datos que están estacionarios en la red, como los datos que se encuentran almacenados.

clasificación de datos

Un proceso para identificar y clasificar los datos de su red en función de su importancia y sensibilidad. Es un componente fundamental de cualquier estrategia de administración de riesgos de ciberseguridad porque lo ayuda a determinar los controles de protección y retención adecuados para los datos. La clasificación de los datos es un componente del pilar de seguridad del AWS Well-Architected Framework. Para obtener más información, consulte Clasificación de datos.

deriva de datos

Una variación significativa entre los datos de producción y los datos que se utilizaron para entrenar un modelo de machine learning, o un cambio significativo en los datos de entrada a lo largo del tiempo. La deriva de datos puede reducir la calidad, la precisión y la imparcialidad generales de las predicciones de los modelos de machine learning.

datos en tránsito

Datos que se mueven de forma activa por la red, por ejemplo, entre los recursos de la red.

malla de datos

Marco de arquitectura que proporciona una propiedad de datos distribuida y descentralizada con una administración y una gobernanza centralizadas.

minimización de datos

El principio de recopilar y procesar solo los datos estrictamente necesarios. Practicar la minimización de los datos Nube de AWS puede reducir los riesgos de privacidad, los costos y la huella de carbono de la analítica.

perímetro de datos

Un conjunto de barreras preventivas en su AWS entorno que ayudan a garantizar que solo las identidades confiables accedan a los recursos confiables desde las redes esperadas. Para obtener más información, consulte Crear un perímetro de datos sobre. AWS

preprocesamiento de datos

Transformar los datos sin procesar en un formato que su modelo de ML pueda analizar fácilmente. El preprocesamiento de datos puede implicar eliminar determinadas columnas o filas y corregir los valores faltantes, incoherentes o duplicados.

procedencia de los datos

El proceso de rastrear el origen y el historial de los datos a lo largo de su ciclo de vida, por ejemplo, la forma en que se generaron, transmitieron y almacenaron los datos.

titular de los datos

Persona cuyos datos se recopilan y procesan.

almacenamiento de datos

Sistema de administración de datos que respalda la inteligencia empresarial, como los análisis. Los almacenes de datos suelen contener grandes cantidades de datos históricos y, por lo general, se utilizan para las consultas y los análisis.

lenguaje de definición de datos (DDL)

Instrucciones o comandos para crear o modificar la estructura de tablas y objetos de una base de datos.

lenguaje de manipulación de datos (DML)

Instrucciones o comandos para modificar (insertar, actualizar y eliminar) la información de una base de datos.

DDL

Consulte lenguaje de definición de bases de datos.

conjunto profundo

Combinar varios modelos de aprendizaje profundo para la predicción. Puede utilizar conjuntos profundos para obtener una predicción más precisa o para estimar la incertidumbre de las predicciones.

aprendizaje profundo

Un subcampo del ML que utiliza múltiples capas de redes neuronales artificiales para identificar el mapeo entre los datos de entrada y las variables objetivo de interés.

defensa en profundidad

Un enfoque de seguridad de la información en el que se distribuyen cuidadosamente una serie de mecanismos y controles de seguridad en una red informática para proteger la confidencialidad, la integridad y la disponibilidad de la red y de los datos que contiene. Al adoptar esta estrategia AWS, se añaden varios controles en diferentes capas de la AWS Organizations estructura para ayudar a proteger los recursos. Por ejemplo, un enfoque de defensa en profundidad podría combinar la autenticación multifactor, la segmentación de la red y el cifrado.

administrador delegado

En AWS Organizations, un servicio compatible puede registrar una cuenta de AWS miembro para administrar las cuentas de la organización y gestionar los permisos de ese servicio. Esta cuenta se denomina administrador delegado para ese servicio. Para obtener más información y una lista de servicios compatibles, consulte Servicios que funcionan con AWS Organizations en la documentación de AWS Organizations .

Implementación

El proceso de hacer que una aplicación, características nuevas o correcciones de código se encuentren disponibles en el entorno de destino. La implementación abarca implementar cambios en una base de código y, a continuación, crear y ejecutar esa base en los entornos de la aplicación.

entorno de desarrollo

Consulte entorno.

control de detección

Un control de seguridad que se ha diseñado para detectar, registrar y alertar después de que se produzca un evento. Estos controles son una segunda línea de defensa, ya que lo advierten sobre los eventos de seguridad que han eludido los controles preventivos establecidos. Para obtener más información, consulte Controles de detección en Implementación de controles de seguridad en AWS.

asignación de flujos de valor para el desarrollo (DVSM)

Proceso que se utiliza para identificar y priorizar las restricciones que afectan negativamente a la velocidad y la calidad en el ciclo de vida del desarrollo de software. DVSM amplía el proceso de asignación del flujo de valor diseñado originalmente para las prácticas de fabricación ajustada. Se centra en los pasos y los equipos necesarios para crear y transferir valor a través del proceso de desarrollo de software.

gemelo digital

Representación virtual de un sistema del mundo real, como un edificio, una fábrica, un equipo industrial o una línea de producción. Los gemelos digitales son compatibles con el mantenimiento predictivo, la supervisión remota y la optimización de la producción.

tabla de dimensiones

En un esquema en estrella, tabla más pequeña que contiene los atributos de datos sobre los datos cuantitativos en una tabla de hechos. Los atributos de la tabla de dimensiones suelen ser campos de texto o números discretos que se comportan como texto. Estos atributos se suelen utilizar para restringir consultas, filtrarlas y etiquetar los conjuntos de resultados.

desastre

Un evento que impide que una carga de trabajo o un sistema cumplan sus objetivos empresariales en su ubicación principal de implementación. Estos eventos pueden ser desastres naturales, fallos técnicos o el resultado de acciones humanas, como una configuración incorrecta involuntaria o un ataque de malware.

recuperación de desastres (DR)

Estrategia y proceso que utiliza para minimizar el tiempo de inactividad y la pérdida de datos a causa de un desastre. Para obtener más información, consulte Recuperación de cargas de trabajo ante desastres en AWS: Recuperación en la nube en el AWS Well-Architected marco.

DML

Consulte lenguaje de manipulación de bases de datos.

diseño basado en el dominio

Un enfoque para desarrollar un sistema de software complejo mediante la conexión de sus componentes a dominios en evolución, o a los objetivos empresariales principales, a los que sirve cada componente. Eric Evans introdujo este concepto en su libro Domain-Driven Design: Tackling Complexity in the Heart of Software (Boston: Addison-Wesley Professional, 2003). Para obtener información sobre cómo utilizar el diseño basado en dominios con el patrón de higos estranguladores, consulte Modernización gradual de los servicios web antiguos de ASP.NET Microsoft (ASMX) mediante contenedores y Amazon API Gateway.

DR

Consulte recuperación ante desastres.

Detección de desviaciones

Seguimiento de las desviaciones con respecto a una configuración con línea de base. Por ejemplo, puedes usarlo AWS CloudFormation para detectar desviaciones en los recursos del sistema o puedes usarlo AWS Control Tower para detectar cambios en tu landing zone que puedan afectar al cumplimiento de los requisitos de gobierno.

DVSM

Consulte asignación de flujos de valor para el desarrollo.

E

EDA

Consulte análisis de datos de tipo exploratorio.

EDI

Consulte intercambio electrónico de datos.

computación en la periferia

La tecnología que aumenta la potencia de cálculo de los dispositivos inteligentes en la periferia de una red de IoT. En comparación con la computación en la nube, la computación de periferia puede reducir la latencia de la comunicación y mejorar el tiempo de respuesta.

intercambio electrónico de datos (EDI)

Intercambio automatizado de documentos comerciales entre organizaciones. Para más información, consulte ¿Qué es el intercambio electrónico de datos?

cifrado

Proceso de computación que transforma datos de texto plano, que son legibles por humanos, en texto cifrado.

clave de cifrado

Cadena criptográfica de bits aleatorios que se genera mediante un algoritmo de cifrado. Las claves pueden variar en longitud y cada una se ha diseñado para ser impredecible y única.

endianidad

El orden en el que se almacenan los bytes en la memoria del ordenador. Big-endian los sistemas almacenan primero el byte más significativo. Little-endian los sistemas almacenan primero el byte menos significativo.

punto de conexión

Consulte punto de conexión de servicio.

servicio de punto de conexión

Servicio que puede alojar en una nube privada virtual (VPC) para compartir con otros usuarios. Puede crear un servicio de punto final con AWS PrivateLink entidades principales Cuentas de AWS o AWS Identity and Access Management (de IAM) y conceder permisos a ellas. Estas cuentas o entidades principales pueden conectarse a su servicio de punto de conexión de forma privada mediante la creación de puntos de conexión de VPC de interfaz. Para obtener más información, consulte Creación de un servicio de punto de conexión en la documentación de Amazon Virtual Private Cloud (Amazon VPC).

planificación de recursos empresariales (ERP)

Sistema que automatiza y administra los procesos empresariales clave (como la contabilidad, MES y la administración de proyectos) de una empresa.

cifrado de sobre

El proceso de cifrar una clave de cifrado con otra clave de cifrado. Para obtener más información, consulte el cifrado de sobres en la documentación de AWS Key Management Service (AWS KMS).

entorno

Una instancia de una aplicación en ejecución. Los siguientes son los tipos de entornos más comunes en la computación en la nube:

- entorno de desarrollo: instancia de una aplicación en ejecución que solo se encuentra disponible para el equipo principal responsable del mantenimiento de la aplicación. Los entornos de desarrollo se utilizan para probar los cambios antes de promocionarlos a los entornos superiores. Este tipo de entorno a veces se denomina entorno de prueba.
- entornos inferiores: todos los entornos de desarrollo de una aplicación, como los que se utilizan para las compilaciones y pruebas iniciales.

- entorno de producción: instancia de una aplicación en ejecución a la que pueden acceder los usuarios finales. En un CI/CD proceso, el entorno de producción es el último entorno de implementación.
- entornos superiores: todos los entornos a los que pueden acceder usuarios que no sean del equipo de desarrollo principal. Esto puede incluir un entorno de producción, entornos de preproducción y entornos para las pruebas de aceptación por parte de los usuarios.

epopeya

En las metodologías ágiles, son categorías funcionales que ayudan a organizar y priorizar el trabajo. Las epopeyas brindan una descripción detallada de los requisitos y las tareas de implementación. Por ejemplo, las epopeyas AWS de seguridad de CAF incluyen la gestión de identidades y accesos, los controles de detección, la seguridad de la infraestructura, la protección de datos y la respuesta a incidentes. Para obtener más información sobre las epopeyas en la estrategia de migración de AWS , consulte la Guía de implementación del programa.

ERP

Consulte planificación de recursos empresariales.

análisis de datos de tipo exploratorio (EDA)

El proceso de analizar un conjunto de datos para comprender sus características principales. Se recopilan o agregan datos y, a continuación, se realizan las investigaciones iniciales para encontrar patrones, detectar anomalías y comprobar las suposiciones. El EDA se realiza mediante el cálculo de estadísticas resumidas y la creación de visualizaciones de datos.

F

tabla de hechos

Tabla central de un esquema en estrella. Almacena datos cuantitativos sobre operaciones empresariales. Por lo general, una tabla de hechos contiene dos tipos de columnas: las que contienen medidas y las que contienen una clave externa para una tabla de dimensiones.

Fail Fast

Filosofía que utiliza pruebas frecuentes e incrementales para reducir el ciclo de vida del desarrollo. Es una parte fundamental de los enfoques ágiles.

límite de aislamiento de errores

En el Nube de AWS, un límite, como una zona de disponibilidad Región de AWS, un plano de control o un plano de datos, que limita el efecto de una falla y ayuda a mejorar la resiliencia de las cargas de trabajo. Para más información, consulte AWS Fault Isolation Boundaries.

rama de característica

Consulte rama.

características

Los datos de entrada que se utilizan para hacer una predicción. Por ejemplo, en un contexto de fabricación, las características pueden ser imágenes que se capturan periódicamente desde la línea de fabricación.

importancia de las características

La importancia que tiene una característica para las predicciones de un modelo. Por lo general, esto se expresa como una puntuación numérica que se puede calcular mediante diversas técnicas, como las explicaciones aditivas de Shapley (SHAP) y los gradientes integrados. Para obtener más información, consulte Interpretabilidad del modelo de aprendizaje automático con. AWS

transformación de funciones

Optimizar los datos para el proceso de ML, lo que incluye enriquecer los datos con fuentes adicionales, escalar los valores o extraer varios conjuntos de información de un solo campo de datos. Esto permite que el modelo de ML se beneficie de los datos. Por ejemplo, si divide la fecha del “27 de mayo de 2021 00:15:37” en “jueves”, “mayo”, “2021” y “15”, puede ayudar al algoritmo de aprendizaje a aprender patrones matizados asociados a los diferentes componentes de los datos.

peticiones con pocos pasos

Proporcionar a un LLM una pequeña cantidad de ejemplos que demuestren la tarea y el resultado deseado antes de pedirle que lleve a cabo una tarea similar. Esta técnica es una aplicación del aprendizaje contextual, en el que los modelos aprenden a partir de ejemplos (tomas) integrados en las instrucciones. Few-shot Las indicaciones pueden ser eficaces para tareas que requieren un formato, un razonamiento o un conocimiento del dominio específicos. Consulte también peticiones desde cero.

FGAC

Consulte control de acceso detallado.

control de acceso preciso (FGAC)

El uso de varias condiciones que tienen por objetivo permitir o denegar una solicitud de acceso.

migración relámpago

Método de migración de bases de datos que utiliza la replicación continua de datos mediante la captura de datos de cambio para migrar los datos en el menor tiempo posible, en lugar de utilizar un enfoque gradual. El objetivo es reducir al mínimo el tiempo de inactividad.

FM

Consulte modelo fundacional.

Modelo fundacional (FM)

Gran red neuronal de aprendizaje profundo que se entrenó con conjuntos de datos masivos de datos generalizados y no etiquetados. Los FM pueden hacer una amplia variedad de tareas generales, como comprender el lenguaje, generar texto e imágenes y conversar en lenguaje natural. Para más información, consulte ¿Qué son los modelos fundacionales?

Puerta de enlace FM

Un intermediario centralizado que controla y normaliza el acceso a los modelos básicos. También se conoce como puerta de enlace LLM.

G

IA generativa

Subconjunto de modelos de IA que se entrenaron con grandes cantidades de datos y que pueden utilizar una simple petición de texto para crear contenido y artefactos nuevos, como imágenes, videos, texto y audio. Para más información, consulte ¿Qué es la IA generativa?

bloqueo geográfico

Consulte restricciones geográficas.

restricciones geográficas (bloqueo geográfico)

En Amazon CloudFront, una opción para impedir que los usuarios de países específicos accedan a las distribuciones de contenido. Puede utilizar una lista de permitidos o bloqueados para especificar los países aprobados y prohibidos. Para obtener más información, consulta Restringir la distribución geográfica del contenido en la CloudFront documentación.

Flujo de trabajo de Gitflow

: Un enfoque en el que los entornos inferiores y superiores utilizan diferentes ramas en un repositorio de código fuente. El flujo de trabajo de Gitflow se considera heredado, mientras que el flujo de trabajo basado en enlaces troncales es el enfoque moderno preferido.

imagen dorada

: Instantánea de un sistema o software que se usa como plantilla para implementar nuevas instancias de ese sistema o software. Por ejemplo, en la fabricación, una imagen dorada se puede utilizar para aprovisionar software en varios dispositivos y ayuda a mejorar la velocidad, la escalabilidad y la productividad de las operaciones de fabricación de dispositivos.

estrategia de implementación desde cero

: La ausencia de infraestructura existente en un entorno nuevo. Al adoptar una estrategia de implementación desde cero para una arquitectura de sistemas, puede seleccionar todas las tecnologías nuevas sin que estas deban ser compatibles con una infraestructura existente, lo que también se conoce como implementación sobre infraestructura existente. Si está ampliando la infraestructura existente, puede combinar las estrategias de implementación sobre infraestructuras existentes y de implementación desde cero.

barrera de protección

: Una regla de alto nivel que ayuda a regular los recursos, las políticas y la conformidad en todas las unidades organizativas (OU). Las barreras de protección preventivas aplican políticas para garantizar la alineación con los estándares de conformidad. Se implementan mediante políticas de control de servicios y límites de permisos de IAM. Las barreras de protección de detección detectan las vulneraciones de las políticas y los problemas de conformidad, y generan alertas para su corrección. Se implementan mediante Amazon AWS Config AWS Security Hub CSPM GuardDuty AWS Trusted Advisor, Amazon Inspector y AWS Lambda cheques personalizados.

barandas (AI)

: Mecanismos de seguridad que filtran, validan y restringen las entradas y salidas de los agentes para ayudar a garantizar un comportamiento responsable y seguro de la IA.

H

HA

: Consulte alta disponibilidad.

migración heterogénea de bases de datos

Migración de la base de datos de origen a una base de datos de destino que utilice un motor de base de datos diferente (por ejemplo, de Oracle a Amazon Aurora). La migración heterogénea suele ser parte de un esfuerzo de rediseño de la arquitectura y convertir el esquema puede ser una tarea compleja. AWS ofrece AWS SCT, lo cual ayuda con las conversiones de esquemas.

alta disponibilidad (HA)

La capacidad de una carga de trabajo para funcionar de forma continua, sin intervención, en caso de desafíos o desastres. Los sistemas de alta disponibilidad están diseñados para realizar una conmutación por error automática, ofrecer un rendimiento de alta calidad de forma constante y gestionar diferentes cargas y fallos con un impacto mínimo en el rendimiento.

modernización histórica

Un enfoque utilizado para modernizar y actualizar los sistemas de tecnología operativa (TO) a fin de satisfacer mejor las necesidades de la industria manufacturera. Un histórico es un tipo de base de datos que se utiliza para recopilar y almacenar datos de diversas fuentes en una fábrica.

datos de reserva

Parte de los datos históricos etiquetados que se ocultan de un conjunto de datos que se utiliza para entrenar un modelo de machine learning. Puede utilizar los datos de reserva para evaluar el rendimiento del modelo mediante la comparación de las predicciones del modelo con los datos de reserva.

human-in-the-loop (HiTL)

Un patrón de flujo de trabajo en el que la ejecución de los agentes se detiene para su revisión y aprobación por parte de una persona en los puntos de decisión críticos.

migración homogénea de bases de datos

Migración de la base de datos de origen a una base de datos de destino que comparte el mismo motor de base de datos (por ejemplo, Microsoft SQL Server a Amazon RDS para SQL Server). La migración homogénea suele formar parte de un esfuerzo para volver a alojar o redefinir la plataforma. Puede utilizar las utilidades de bases de datos nativas para migrar el esquema.

datos recientes

Datos a los que se accede con frecuencia, como datos en tiempo real o datos traslacionales recientes. Por lo general, estos datos requieren un nivel o una clase de almacenamiento de alto rendimiento para proporcionar respuestas rápidas a las consultas.

hotfix

Una solución urgente para un problema crítico en un entorno de producción. Debido a su urgencia, una revisión suele realizarse fuera del flujo de trabajo habitual de las DevOps versiones.

periodo de hiperatención

Periodo, inmediatamente después de la transición, durante el cual un equipo de migración administra y monitorea las aplicaciones migradas en la nube para solucionar cualquier problema. Por lo general, este periodo dura de 1 a 4 días. Al final del periodo de hiperatención, el equipo de migración suele transferir la responsabilidad de las aplicaciones al equipo de operaciones en la nube.

I

IaC

Consulte infraestructura como código.

políticas basadas en identidades

Política asociada a uno o más directores de IAM que define sus permisos en el entorno. Nube de AWS

aplicación inactiva

Aplicación que utiliza un promedio de CPU y memoria de entre 5 y 20 por ciento durante un periodo de 90 días. En un proyecto de migración, es habitual retirar estas aplicaciones o mantenerlas en las instalaciones.

IIoT

Consulte Internet de las cosas industrial.

infraestructura inmutable

Modelo que implementa una nueva infraestructura para las cargas de trabajo de producción en lugar de actualizar o modificar la infraestructura existente o aplicarle revisiones. Las infraestructuras inmutables son de manera intrínseca más coherentes, fiables y predecibles que las infraestructuras mutables. Para obtener más información, consulte las mejores prácticas del Framework para implementar con una infraestructura inmutable. AWS Well-Architected

VPC entrante (de entrada)

En una arquitectura de AWS cuentas múltiples, una VPC que acepta, inspecciona y enruta las conexiones de red desde fuera de una aplicación. La Arquitectura de referencia de seguridad de AWS recomienda configurar su cuenta de red con VPC entrantes, salientes y de inspección para proteger la interfaz bidireccional entre su aplicación e Internet en general.

migración gradual

Estrategia de transición en la que se migra la aplicación en partes pequeñas en lugar de realizar una transición única y completa. Por ejemplo, puede trasladar inicialmente solo unos pocos microservicios o usuarios al nuevo sistema. Tras comprobar que todo funciona correctamente, puede trasladar microservicios o usuarios adicionales de forma gradual hasta que pueda retirar su sistema heredado. Esta estrategia reduce los riesgos asociados a las grandes migraciones.

Industria 4.0

Un término que Klaus Schwab introdujo en 2016 para referirse a la modernización de los procesos de fabricación mediante avances en la conectividad, los datos en tiempo real, la automatización, el análisis y. AI/ML

infraestructura

Todos los recursos y activos que se encuentran en el entorno de una aplicación.

infraestructura como código (IaC)

Proceso de aprovisionamiento y administración de la infraestructura de una aplicación mediante un conjunto de archivos de configuración. La IaC se ha diseñado para ayudarlo a centralizar la administración de la infraestructura, estandarizar los recursos y escalar con rapidez a fin de que los entornos nuevos sean repetibles, fiables y consistentes.

Internet de las cosas industrial (IIoT)

El uso de sensores y dispositivos conectados a Internet en los sectores industriales, como el productivo, el eléctrico, el automotriz, el sanitario, el de las ciencias de la vida y el de la agricultura. Para obtener más información, consulte Creación de una estrategia de transformación digital del Internet de las cosas industrial (IIoT).

VPC de inspección

En una arquitectura de AWS cuentas múltiples, una VPC centralizada que gestiona las inspecciones del tráfico de red entre las VPC (iguales o Regiones de AWS diferentes), Internet y las redes locales. La Arquitectura de referencia de seguridad de AWS recomienda configurar su

cuenta de red con VPC entrantes, salientes y de inspección para proteger la interfaz bidireccional entre su aplicación e Internet en general.

Internet de las cosas (IoT)

Red de objetos físicos conectados con sensores o procesadores integrados que se comunican con otros dispositivos y sistemas a través de Internet o de una red de comunicación local. Para obtener más información, consulte ¿Qué es IoT?.

interpretabilidad

Característica de un modelo de machine learning que describe el grado en que un ser humano puede entender cómo las predicciones del modelo dependen de sus entradas. Para obtener más información, consulte Interpretabilidad del modelo de aprendizaje automático con. AWS

IoT

Consulte Internet de las cosas.

biblioteca de información de TI (ITIL)

Conjunto de prácticas recomendadas para ofrecer servicios de TI y alinearlos con los requisitos empresariales. La ITIL proporciona la base para la ITSM.

administración de servicios de TI (ITSM)

Actividades asociadas con el diseño, la implementación, la administración y el soporte de los servicios de TI para una organización. Para obtener información sobre la integración de las operaciones en la nube con las herramientas de ITSM, consulte la Guía de integración de operaciones.

ITIL

Consulte biblioteca de información de TI.

ITSM

Consulte administración de servicios de TI.

L

control de acceso basado en etiquetas (LBAC)

Una implementación del control de acceso obligatorio (MAC) en la que a los usuarios y a los propios datos se les asigna explícitamente un valor de etiqueta de seguridad. La intersección

entre la etiqueta de seguridad del usuario y la etiqueta de seguridad de los datos determina qué filas y columnas puede ver el usuario.

zona de aterrizaje

Una landing zone es un AWS entorno multicuenta bien diseñado, escalable y seguro. Este es un punto de partida desde el cual las empresas pueden lanzar e implementar rápidamente cargas de trabajo y aplicaciones con confianza en su entorno de seguridad e infraestructura. Para obtener más información sobre las zonas de aterrizaje, consulte Configuración de un entorno de AWS seguro y escalable con varias cuentas.

modelo de lenguaje de gran tamaño (LLM)

Modelo de IA de aprendizaje profundo que se entrenó previamente con una gran cantidad de datos. Un LLM puede llevar a cabo varias tareas, como responder preguntas, resumir documentos, traducir textos a otros idiomas y completar oraciones. Para más información, consulte ¿Qué es un LLM (modelo de lenguaje de gran tamaño)?

migración grande

Migración de 300 servidores o más.

LBAC

Consulte control de acceso basado en etiquetas.

privilegio mínimo

La práctica recomendada de seguridad que consiste en conceder los permisos mínimos necesarios para realizar una tarea. Para obtener más información, consulte Aplicar permisos de privilegio mínimo en la documentación de IAM.

migrar mediante lift-and-shift

Consulte Las 7 R.

sistema little-endian

Un sistema que almacena primero el byte menos significativo. Consulte también endianidad.

LLM

Consulte modelo de lenguaje de gran tamaño.

entornos inferiores

Consulte entorno.

M

machine learning (ML)

Un tipo de inteligencia artificial que utiliza algoritmos y técnicas para el reconocimiento y el aprendizaje de patrones. El ML analiza y aprende de los datos registrados, como los datos del Internet de las cosas (IoT), para generar un modelo estadístico basado en patrones. Para más información, consulte Machine learning.

rama principal

Consulte rama.

malware

Software diseñado para comprometer la seguridad o la privacidad de la computadora. El malware podría interrumpir los sistemas informáticos, filtrar información confidencial u obtener acceso no autorizado. Algunos ejemplos de malware son los virus, los gusanos, el ransomware, los troyanos, el spyware y los registradores de pulsaciones de teclas.

Servicios administrados

Servicios de AWS en el que AWS opera la capa de infraestructura, el sistema operativo y las plataformas, y se accede a los puntos finales para almacenar y recuperar datos. Amazon Simple Storage Service (Amazon S3) y Amazon DynamoDB son ejemplos de servicios administrados. También se conocen como servicios abstractos.

sistema de ejecución de fabricación (MES)

Sistema de software para seguir, supervisar, documentar y controlar los procesos de producción que convierten las materias primas en productos acabados en la zona de producción.

MAP

Consulte Programa de aceleración de la migración.

MCP

Consulte Model Context Protocol.

Protocolo de contexto para modelos (MCP)

Un protocolo sin estado para la comunicación entre el agente y la herramienta.

Servidor MCP

Un servicio que expone una o más herramientas a través del protocolo Model Context.

mecanismo

Proceso completo mediante el que se crea una herramienta, se impulsa su adopción y, a continuación, se inspeccionan los resultados para hacer ajustes. Un mecanismo es un ciclo que se refuerza y mejora por sí mismo a medida que funciona. Para obtener más información, consulte Creación de mecanismos en el AWS Well-Architected marco.

cuenta de miembro

Todas las Cuentas de AWS demás cuentas, excepto la de administración, que forman parte de una organización AWS Organizations. Una cuenta no puede pertenecer a más de una organización a la vez.

MES

Consulte sistema de ejecución de fabricación.

Message Queuing Telemetry Transport (MQTT)

Un protocolo de comunicación ligero de máquina a máquina (M2M), basado en el publish/subscribepatrón, para dispositivos de IoT con recursos limitados.

microservicio

Un servicio pequeño e independiente que se comunica a través de API bien definidas y que, por lo general, es propiedad de equipos pequeños e independientes. Por ejemplo, un sistema de seguros puede incluir microservicios que se adapten a las capacidades empresariales, como las de ventas o marketing, o a subdominios, como las de compras, reclamaciones o análisis. Los beneficios de los microservicios incluyen la agilidad, la escalabilidad flexible, la facilidad de implementación, el código reutilizable y la resiliencia. Para obtener más información, consulte Integrar microservicios mediante servicios sin servidor. AWS

arquitectura de microservicios

Un enfoque para crear una aplicación con componentes independientes que ejecutan cada proceso de la aplicación como un microservicio. Estos microservicios se comunican a través de una interfaz bien definida mediante API ligeras. Cada microservicio de esta arquitectura se puede actualizar, implementar y escalar para satisfacer la demanda de funciones específicas de una aplicación. Para obtener más información, consulte Implementación de microservicios en. AWS

Programa de aceleración de la migración (MAP)

Un AWS programa que proporciona soporte de consultoría, formación y servicios para ayudar a las organizaciones a crear una base operativa sólida para migrar a la nube y para ayudar a

compensar el costo inicial de las migraciones. El MAP incluye una metodología de migración para ejecutar las migraciones antiguas de forma metódica y un conjunto de herramientas para automatizar y acelerar los escenarios de migración más comunes.

migración a escala

Proceso de transferencia de la mayoría de la cartera de aplicaciones a la nube en oleadas, con más aplicaciones desplazadas a un ritmo más rápido en cada oleada. En esta fase, se utilizan las prácticas recomendadas y las lecciones aprendidas en las fases anteriores para implementar una fábrica de migración de equipos, herramientas y procesos con el fin de agilizar la migración de las cargas de trabajo mediante la automatización y la entrega ágil. Esta es la tercera fase de la estrategia de migración de AWS.

fábrica de migración

Cross-functional equipos que agilizan la migración de las cargas de trabajo mediante enfoques ágiles y automatizados. Los equipos de las fábricas de migración suelen estar compuestos por analistas y propietarios de operaciones, ingenieros de migración, desarrolladores y DevOps profesionales que trabajan a pasos agigantados. Entre el 20 y el 50 por ciento de la cartera de aplicaciones empresariales se compone de patrones repetidos que pueden optimizarse mediante un enfoque de fábrica. Para obtener más información, consulte la discusión sobre las fábricas de migración y la Guía de fábricas de migración a la nube en este contenido.

metadatos de migración

Información sobre la aplicación y el servidor que se necesita para completar la migración. Cada patrón de migración requiere un conjunto diferente de metadatos de migración. Algunos ejemplos de metadatos de migración son la subred de destino, el grupo de seguridad y AWS la cuenta.

patrón de migración

Tarea de migración repetible que detalla la estrategia de migración, el destino de la migración y la aplicación o el servicio de migración utilizados. Ejemplo: rehospede la migración a Amazon EC2 AWS con Application Migration Service.

Migration Portfolio Assessment (MPA)

Herramienta en línea que proporciona información a fin de validar los argumentos comerciales necesarios para migrar a la Nube de AWS. La MPA ofrece una evaluación detallada de la cartera (adecuación del tamaño de los servidores, precios, comparaciones del costo total de propiedad, análisis de los costos de migración), así como una planificación de la migración (análisis y recopilación de datos de aplicaciones, agrupación de aplicaciones, priorización de la migración y

planificación de oleadas). La herramienta MPA (requiere iniciar sesión) está disponible de forma gratuita para todos los AWS consultores y consultores de los socios de APN.

Evaluación de la preparación para la migración (MRA)

Proceso que consiste en obtener información sobre el estado de preparación de una organización para la nube, identificar sus puntos fuertes y débiles y elaborar un plan de acción para cerrar las brechas identificadas mediante el AWS CAF. Para obtener más información, consulte la Guía de preparación para la migración. La MRA es la primera fase de la estrategia de migración de AWS.

estrategia de migración

Enfoque utilizado para migrar una carga de trabajo a la Nube de AWS. Para más información, consulte la entrada Las 7 R de este glosario y también Mobilize your organization to accelerate large-scale migrations.

ML

Consulte machine learning.

modernización

Transformar una aplicación obsoleta (antigua o monolítica) y su infraestructura en un sistema ágil, elástico y de alta disponibilidad en la nube para reducir los gastos, aumentar la eficiencia y aprovechar las innovaciones. Para más información, consulte Strategy for modernizing applications in the Nube de AWS.

evaluación de la preparación para la modernización

Evaluación que ayuda a determinar la preparación para la modernización de las aplicaciones de una organización; identifica los beneficios, los riesgos y las dependencias; y determina qué tan bien la organización puede soportar el estado futuro de esas aplicaciones. El resultado de la evaluación es un esquema de la arquitectura objetivo, una hoja de ruta que detalla las fases de desarrollo y los hitos del proceso de modernización y un plan de acción para abordar las brechas identificadas. Para más información, consulte Evaluating modernization readiness for applications in the Nube de AWS.

aplicaciones monolíticas (monolitos)

Aplicaciones que se ejecutan como un único servicio con procesos estrechamente acoplados. Las aplicaciones monolíticas presentan varios inconvenientes. Si una característica de la aplicación experimenta un aumento en la demanda, se debe escalar toda la arquitectura. Agregar o mejorar las características de una aplicación monolítica también se vuelve más complejo a medida que crece la base de código. Para solucionar problemas con la aplicación, puede utilizar

una arquitectura de microservicios. Para obtener más información, consulte Descomposición de monolitos en microservicios.

MPA

Consulte Migration Portfolio Assessment.

MQTT

Consulte Message Queuing Telemetry Transport.

clasificación multiclase

Un proceso que ayuda a generar predicciones para varias clases (predice uno de más de dos resultados). Por ejemplo, un modelo de ML podría preguntar "¿Este producto es un libro, un automóvil o un teléfono?" o "¿Qué categoría de productos es más interesante para este cliente?".

infraestructura mutable

Modelo que actualiza y modifica la infraestructura actual para las cargas de trabajo de producción. Para mejorar la coherencia, la confiabilidad y la previsibilidad, el AWS Well-Architected Marco recomienda el uso de una infraestructura inmutable como práctica recomendada.

O

OAC

Consulte control de acceso de origen.

OAI

Consulte identidad de acceso de origen.

OCM

Consulte administración del cambio organizacional.

migración fuera de línea

Método de migración en el que la carga de trabajo de origen se elimina durante el proceso de migración. Este método implica un tiempo de inactividad prolongado y, por lo general, se utiliza para cargas de trabajo pequeñas y no críticas.

OI

Consulte integración de operaciones.

OLA

Consulte acuerdo de nivel operativo.

migración en línea

Método de migración en el que la carga de trabajo de origen se copia al sistema de destino sin que se desconecte. Las aplicaciones que están conectadas a la carga de trabajo pueden seguir funcionando durante la migración. Este método implica un tiempo de inactividad nulo o mínimo y, por lo general, se utiliza para cargas de trabajo de producción críticas.

OPC-UA

Consulte Open Process Communications: arquitectura unificada.

Comunicaciones de proceso abierto: arquitectura unificada () OPC-UA

Un protocolo de comunicación de máquina a máquina (M2M) para la automatización industrial. OPC-UA proporciona un estándar de interoperabilidad con esquemas de cifrado, autenticación y autorización de datos.

acuerdo de nivel operativo (OLA)

Acuerdo que aclara lo que los grupos de TI operativos se comprometen a ofrecerse entre sí, para respaldar un acuerdo de nivel de servicio (SLA).

revisión de la preparación operativa (ORR)

Lista de comprobación de preguntas y prácticas recomendadas asociadas que son útiles para comprender, evaluar, prevenir o reducir el alcance de los incidentes y posibles errores. Para obtener más información, consulte las revisiones de preparación operativa (ORR) en el AWS Well-Architected marco.

tecnología operativa (TO)

Sistemas de hardware y software que funcionan con el entorno físico para controlar las operaciones, los equipos y la infraestructura industriales. En el sector de la fabricación, la integración de los sistemas de TO y tecnología de la información (TI) es un enfoque clave para las transformaciones de la industria 4.0.

integración de operaciones (OI)

Proceso de modernización de las operaciones en la nube, que implica la planificación de la preparación, la automatización y la integración. Para obtener más información, consulte la Guía de integración de las operaciones.

registro de seguimiento organizativo

Un registro creado por y AWS CloudTrail que registra todos los eventos Cuentas de AWS de una organización AWS Organizations. Este registro de seguimiento se crea en cada Cuenta de AWS que forma parte de la organización y realiza un seguimiento de la actividad en cada cuenta. Para obtener más información, consulte Crear un registro para una organización en la CloudTrail documentación.

administración del cambio organizacional (OCM)

Marco para administrar las transformaciones empresariales importantes y disruptivas desde la perspectiva de las personas, la cultura y el liderazgo. La OCM ayuda a las empresas a prepararse para nuevos sistemas y estrategias y a realizar la transición a ellos, al acelerar la adopción de cambios, abordar los problemas de transición e impulsar cambios culturales y organizacionales. En la estrategia de AWS migración, este marco se denomina aceleración de personal, debido a la velocidad de cambio que requieren los proyectos de adopción de la nube. Para obtener más información, consulte la Guía de OCM.

control de acceso de origen (OAC)

En CloudFront, una opción mejorada para restringir el acceso y proteger el contenido del Amazon Simple Storage Service (Amazon S3). El OAC admite todos los buckets de S3 Regiones de AWS, el cifrado del lado del servidor con AWS KMS (SSE-KMS) y `DELETE` las solicitudes `PUT` y dinámicas al bucket de S3.

identidad de acceso de origen (OAI)

En CloudFront, una opción para restringir el acceso y proteger el contenido de Amazon S3. Cuando utiliza OAI, CloudFront crea un principal con el que Amazon S3 puede autenticarse. Los directores autenticados solo pueden acceder al contenido de un bucket de S3 a través de una distribución específica. CloudFront Consulte también el OAC, que proporciona un control de acceso más detallado y mejorado.

ORR

Consulte revisión de la preparación operativa.

OT

Consulte tecnología operativa.

VPC saliente (de salida)

En una arquitectura de AWS cuentas múltiples, una VPC que gestiona las conexiones de red que se inician desde una aplicación. La Arquitectura de referencia de seguridad de AWS recomienda

configurar su cuenta de red con VPC entrantes, salientes y de inspección para proteger la interfaz bidireccional entre su aplicación e Internet en general.

P

límite de permisos

Una política de administración de IAM que se adjunta a las entidades principales de IAM para establecer los permisos máximos que puede tener el usuario o el rol. Para obtener más información, consulte Límites de permisos en la documentación de IAM.

información de identificación personal (PII)

Información que, vista directamente o combinada con otros datos relacionados, puede utilizarse para deducir de manera razonable la identidad de una persona. Algunos ejemplos de información de identificación personal son los nombres, las direcciones y la información de contacto.

PII

Consulte información de identificación personal.

manual de estrategias

Conjunto de pasos predefinidos que capturan el trabajo asociado a las migraciones, como la entrega de las funciones de operaciones principales en la nube. Un manual puede adoptar la forma de scripts, manuales de procedimientos automatizados o resúmenes de los procesos o pasos necesarios para operar un entorno modernizado.

PLC

Consulte controlador lógico programable.

PLM

Consulte administración del ciclo de vida del producto.

policy

Objeto que puede definir permisos (consulte política basada en identidad), especificar las condiciones de acceso (consulte política basada en recursos) o definir los permisos máximos para todas las cuentas de una organización de AWS Organizations (consulte política de control de servicio).

persistencia políglota

Elegir de forma independiente la tecnología de almacenamiento de datos de un microservicio en función de los patrones de acceso a los datos y otros requisitos. Si sus microservicios tienen la misma tecnología de almacenamiento de datos, pueden enfrentarse a desafíos de implementación o experimentar un rendimiento deficiente. Los microservicios se implementan más fácilmente y logran un mejor rendimiento y escalabilidad si utilizan el almacén de datos que mejor se adapte a sus necesidades.

evaluación de cartera

Proceso de detección, análisis y priorización de la cartera de aplicaciones para planificar la migración. Para obtener más información, consulte la Evaluación de la preparación para la migración.

predicate

Condición de consulta que devuelve `true` o `false`. En general, se encuentra en una cláusula `WHERE`.

inserción de predicados

Técnica de optimización de consultas en bases de datos que filtra los datos de la consulta antes de transferirlos. Esta técnica reduce la cantidad de datos de la base de datos relacional que se tienen que recuperar y procesar. Además, mejora el rendimiento de las consultas.

control preventivo

Un control de seguridad diseñado para evitar que ocurra un evento. Estos controles son la primera línea de defensa para evitar el acceso no autorizado o los cambios no deseados en la red. Para obtener más información, consulte Controles preventivos en Implementación de controles de seguridad en AWS.

entidad principal

Una entidad AWS que puede realizar acciones y acceder a los recursos. Esta entidad suele ser un usuario raíz para un Cuenta de AWS rol de IAM o un usuario. Para obtener más información, consulte Entidad principal en Términos y conceptos de roles en la documentación de IAM.

Privacidad desde el diseño

Enfoque de ingeniería de sistemas que tiene en cuenta la privacidad durante todo el proceso de desarrollo.

zonas alojadas privadas

Contenedor que aloja información acerca de cómo desea que responda Amazon Route 53 a las consultas de DNS de un dominio y sus subdominios en una o varias VPC. Para obtener más información, consulte Uso de zonas alojadas privadas en la documentación de Route 53.

control proactivo

Control de seguridad que se diseñó para evitar la implementación de recursos que no cumplan con la normativa. Estos controles analizan los recursos antes de aprovisionarlos. Si el recurso no cumple con los requisitos del control, no se aprovisiona. Para obtener más información, consulte la guía de referencia de controles en la AWS Control Tower documentación y consulte Controles proactivos en Implementación de controles de seguridad en AWS.

administración del ciclo de vida del producto (PLM)

Administración de los datos y los procesos de un producto a lo largo de todo su ciclo de vida, desde el diseño, el desarrollo y el lanzamiento, pasando por el crecimiento y la madurez, hasta la reducción de su uso y su retirada.

entorno de producción

Consulte entorno.

controlador lógico programable (PLC)

En el sector de la fabricación, computadora adaptable y altamente fiable que supervisa las máquinas y automatiza los procesos de fabricación.

encadenamiento de peticiones

Uso de la salida de una petición de LLM como entrada para la siguiente petición a fin de generar mejores respuestas. Esta técnica se utiliza para dividir una tarea compleja en tareas secundarias o para refinar o ampliar de forma iterativa una respuesta preliminar. Ayuda a mejorar la precisión y la relevancia de las respuestas de un modelo y permite obtener resultados más detallados y personalizados.

seudonimización

El proceso de reemplazar los identificadores personales de un conjunto de datos por valores de marcadores de posición. La seudonimización puede ayudar a proteger la privacidad personal. Los datos seudonimizados siguen considerándose datos personales.

publish/subscribe (pub/sub)

Patrón que permite establecer comunicaciones asíncronas entre microservicios para mejorar la escalabilidad y la capacidad de respuesta. Por ejemplo, en un MES basado en microservicios, un microservicio puede publicar mensajes de eventos en un canal al que se pueden suscribir otros microservicios. El sistema puede agregar nuevos microservicios sin cambiar el servicio de publicación.

Q

plan de consulta

Serie de pasos, como instrucciones, que se utilizan para acceder a los datos de un sistema de base de datos relacional SQL.

regresión del plan de consulta

El optimizador de servicios de la base de datos elige un plan menos óptimo que antes de un cambio determinado en el entorno de la base de datos. Los cambios en estadísticas, restricciones, configuración del entorno, enlaces de parámetros de consultas y actualizaciones del motor de base de datos PostgreSQL pueden provocar una regresión del plan.

R

Matriz RACI

Consulte responsable, fiable, consultada e informada (RACI).

RAG

Consulte generación aumentada por recuperación.

ransomware

Software malicioso que se ha diseñado para bloquear el acceso a un sistema informático o a los datos hasta que se efectúe un pago.

Matriz RASCI

Consulte responsable, fiable, consultada e informada (RACI).

RCAC

Consulte control de acceso por filas y columnas.

réplica de lectura

Una copia de una base de datos que se utiliza con fines de solo lectura. Puede enrutar las consultas a la réplica de lectura para reducir la carga en la base de datos principal.

rediseñar

Consulte Las 7 R.

objetivo de punto de recuperación (RPO)

La cantidad de tiempo máximo aceptable desde el último punto de recuperación de datos. Esto determina qué se considera una pérdida de datos aceptable entre el último punto de recuperación y la interrupción del servicio.

objetivo de tiempo de recuperación (RTO)

La demora máxima aceptable entre la interrupción del servicio y el restablecimiento del servicio.

refactorizar

Consulte Las 7 R.

Region

Conjunto de AWS recursos en un área geográfica. Cada uno Región de AWS está aislado e independiente de los demás para proporcionar tolerancia a las fallas, estabilidad y resiliencia. Para más información, consulte Specify which Regiones de AWS your account can use.

regresión

Una técnica de ML que predice un valor numérico. Por ejemplo, para resolver el problema de "¿A qué precio se venderá esta casa?", un modelo de ML podría utilizar un modelo de regresión lineal para predecir el precio de venta de una vivienda en función de datos conocidos sobre ella (por ejemplo, los metros cuadrados).

volver a alojar

Consulte Las 7 R.

versión

En un proceso de implementación, el acto de promover cambios en un entorno de producción.

reubicar

Consulte Las 7 R.

redefinir la plataforma

Consulte Las 7 R.

recomprar

Consulte Las 7 R.

resiliencia

Capacidad de una aplicación para resistir interrupciones o recuperarse de ellas. Al planificar la resiliencia en la Nube de AWS, la alta disponibilidad y la recuperación ante desastres son consideraciones comunes. Para más información, consulte Resiliencia en la Nube de AWS.

política basada en recursos

Una política asociada a un recurso, como un bucket de Amazon S3, un punto de conexión o una clave de cifrado. Este tipo de política especifica a qué entidades principales se les permite el acceso, las acciones compatibles y cualquier otra condición que deba cumplirse.

matriz responsable, confiable, consultada e informada (RACI)

Una matriz que define las funciones y responsabilidades de todas las partes involucradas en las actividades de migración y las operaciones de la nube. El nombre de la matriz se deriva de los tipos de responsabilidad definidos en la matriz: responsable (R), contable (A), consultado (C) e informado (I). El tipo de soporte (S) es opcional. Si incluye el soporte, la matriz se denomina matriz RASCI y, si la excluye, se denomina matriz RACI.

control receptivo

Un control de seguridad que se ha diseñado para corregir los eventos adversos o las desviaciones con respecto a su base de seguridad. Para obtener más información, consulte Controles receptivos en Implementación de controles de seguridad en AWS.

retain

Consulte Las 7 R.

retirar

Consulte Las 7 R.

Generación aumentada de recuperación (RAG)

Tecnología de IA generativa mediante la que un LLM hace referencia a un origen de datos autorizado que se encuentra fuera de sus orígenes de datos de entrenamiento antes de generar una respuesta. Por ejemplo, un modelo de RAG podría hacer una búsqueda semántica en la base de conocimientos o en los datos personalizados de una organización. Para más información, consulte ¿Qué es RAG (generación aumentada por recuperación)?

rotación

Proceso mediante el que periódicamente se actualiza un secreto para que resulte más difícil que un atacante pueda acceder a las credenciales.

control de acceso por filas y columnas (RCAC)

El uso de expresiones SQL básicas y flexibles que tienen reglas de acceso definidas. El RCAC consta de permisos de fila y máscaras de columnas.

RPO

Consulte objetivo de punto de recuperación.

RTO

Consulte objetivo de tiempo de recuperación.

manual de procedimientos

Conjunto de procedimientos manuales o automatizados necesarios para realizar una tarea específica. Por lo general, se diseñan para agilizar las operaciones o los procedimientos repetitivos con altas tasas de error.

S

SAML 2.0

Un estándar abierto que utilizan muchos proveedores de identidad (IdPs). Esta función permite el inicio de sesión único (SSO) federado, de modo que los usuarios pueden iniciar sesión Consola de administración de AWS o llamar a las operaciones de la AWS API sin tener que crear un usuario en IAM para todos los miembros de la organización. Para obtener más información sobre la federación basada en SAML 2.0, consulte Acerca de la federación basada en SAML 2.0 en la documentación de IAM.

SCADA

Consulte control de supervisión y adquisición de datos.

SCP

Consulte política de control de servicio.

secreta

En AWS Secrets Manager, información confidencial o restringida, como una contraseña o credenciales de usuario, que se almacena de forma cifrada. Se compone del valor del secreto y de sus metadatos. El valor del secreto puede ser binario, una sola cadena o varias cadenas. Para más información, consulte What's in a Secrets Manager secret? en la documentación de Secrets Manager.

seguridad desde el diseño

Enfoque de ingeniería de sistemas que tiene en cuenta la seguridad durante todo el proceso de desarrollo.

control de seguridad

Barrera de protección técnica o administrativa que impide, detecta o reduce la capacidad de un agente de amenazas para aprovechar una vulnerabilidad de seguridad. Existen cuatro tipos de controles de seguridad principales: preventivos, de detección, de respuesta y proactivos.

refuerzo de la seguridad

Proceso de reducir la superficie expuesta a ataques para hacerla más resistente a los ataques. Esto puede incluir acciones, como la eliminación de los recursos que ya no se necesitan, la implementación de prácticas recomendadas de seguridad consistente en conceder privilegios mínimos o la desactivación de características innecesarias en los archivos de configuración.

sistema de información sobre seguridad y administración de eventos (SIEM)

Herramientas y servicios que combinan sistemas de administración de información sobre seguridad (SIM) y de administración de eventos de seguridad (SEM). Un sistema de SIEM recopila, monitorea y analiza los datos de servidores, redes, dispositivos y otras fuentes para detectar amenazas y brechas de seguridad y generar alertas.

automatización de la respuesta de seguridad

Acción predefinida y programada que está diseñada para responder automáticamente a un evento de seguridad o corregirlo. Estas automatizaciones sirven como controles de seguridad

preventivos o adaptables que le ayudan a implementar las mejores prácticas AWS de seguridad. La modificación de un grupo de seguridad de VPC, la aplicación de revisiones a una instancia de Amazon EC2 o la rotación de credenciales son algunos ejemplos de acciones de respuesta automatizadas.

cifrado del servidor

Cifrado de los datos en su destino, por parte de Servicio de AWS quien los recibe.

política de control de servicio (SCP)

Una política que proporciona un control centralizado de los permisos de todas las cuentas de una organización en AWS Organizations. Las SCP definen barreras de protección o establecen límites a las acciones que un administrador puede delegar en los usuarios o roles. Puede utilizar las SCP como listas de permitidos o rechazados, para especificar qué servicios o acciones se encuentra permitidos o prohibidos. Para obtener más información, consulte las políticas de control del servicio en la AWS Organizations documentación.

punto de enlace de servicio

La URL del punto de entrada de un Servicio de AWS. Para conectarse mediante programación a un servicio de destino, puede utilizar un punto de conexión. Para obtener más información, consulte Puntos de conexión de Servicio de AWS en Referencia general de AWS.

acuerdo de nivel de servicio (SLA)

Acuerdo que aclara lo que un equipo de TI se compromete a ofrecer a los clientes, como el tiempo de actividad y el rendimiento del servicio.

indicador de nivel de servicio (SLI)

Medición de un aspecto del rendimiento de un servicio, como la tasa de errores, la disponibilidad o el rendimiento.

objetivo de nivel de servicio (SLO)

Métrica objetivo que representa el estado de un servicio medido mediante un indicador de nivel de servicio.

modelo de responsabilidad compartida

Un modelo que describe la responsabilidad con AWS la que compartes la seguridad y el cumplimiento de la nube. AWS es responsable de la seguridad de la nube, mientras que usted es responsable de la seguridad en la nube. Para obtener más información, consulte el Modelo de responsabilidad compartida.

Shadow AI

Aplicaciones de IA no autorizadas creadas o utilizadas fuera de los canales regulados dentro de una organización.

SIEM

Consulte sistema de administración de eventos e información de seguridad.

único punto de error (SPOF)

Error en un único componente crítico de una aplicación que puede interrumpir el sistema.

SLA

Consulte acuerdo de nivel de servicio.

SLI

Consulte indicador de nivel de servicio.

SLO

Consulte objetivo de nivel de servicio.

modelo de dividir y sembrar

Un patrón para escalar y acelerar los proyectos de modernización. A medida que se definen las nuevas funciones y los lanzamientos de los productos, el equipo principal se divide para crear nuevos equipos de productos. Esto ayuda a ampliar las capacidades y los servicios de su organización, mejora la productividad de los desarrolladores y apoya la innovación rápida. Para más información, consulte Phased approach to modernizing applications in the Nube de AWS.

SPOF

Consulte único punto de error.

esquema en estrella

Estructura organizativa de una base de datos que utiliza una tabla de hechos de gran tamaño para almacenar datos transaccionales o medidos y una o varias tablas dimensionales más pequeñas para almacenar los atributos de los datos. Esta estructura está diseñada para utilizarse en un almacén de datos o con fines de inteligencia empresarial.

patrón de higo estrangulador

Un enfoque para modernizar los sistemas monolíticos mediante la reescritura y el reemplazo gradual de las funciones del sistema hasta que se pueda desmantelar el sistema heredado.

Este patrón utiliza la analogía de una higuera que crece hasta convertirse en un árbol estable y, finalmente, se apodera y reemplaza a su host. El patrón fue presentado por Martin Fowler como una forma de gestionar el riesgo al reescribir sistemas monolíticos. Para ver un ejemplo de cómo aplicar este patrón, consulte Modernización gradual de los servicios web antiguos de Microsoft ASP.NET (ASMX) mediante contenedores y Amazon API Gateway.

subred

Un intervalo de direcciones IP en la VPC. Una subred debe residir en una sola zona de disponibilidad.

control de supervisión y adquisición de datos (SCADA)

En el sector de la fabricación, sistema que utiliza hardware y software para supervisar los activos físicos y las operaciones de producción.

cifrado simétrico

Un algoritmo de cifrado que utiliza la misma clave para cifrar y descifrar los datos.

pruebas sintéticas

Prueba de un sistema de manera que simule las interacciones de los usuarios para detectar posibles problemas o supervisar el rendimiento. Puede usar Amazon CloudWatch Synthetics para crear estas pruebas.

petición del sistema

Técnica para proporcionar contexto, instrucciones o pautas a un LLM para dirigir su comportamiento. Las peticiones del sistema ayudan a establecer el contexto y las reglas para las interacciones con los usuarios.

T

etiquetas

Key-value pares que actúan como metadatos para organizar sus AWS recursos. Las etiquetas pueden ayudar a administrar, identificar, organizar, buscar y filtrar recursos de . Para obtener más información, consulte Etiquetado de los recursos de AWS.

variable de destino

El valor que intenta predecir en el ML supervisado. Esto también se conoce como variable de resultado. Por ejemplo, en un entorno de fabricación, la variable objetivo podría ser un defecto del producto.

lista de tareas

Herramienta que se utiliza para hacer un seguimiento del progreso mediante un manual de procedimientos. La lista de tareas contiene una descripción general del manual de procedimientos y una lista de las tareas generales que deben completarse. Para cada tarea general, se incluye la cantidad estimada de tiempo necesario, el propietario y el progreso.

entorno de prueba

Consulte entorno.

entrenamiento

Proporcionar datos de los que pueda aprender su modelo de ML. Los datos de entrenamiento deben contener la respuesta correcta. El algoritmo de aprendizaje encuentra patrones en los datos de entrenamiento que asignan los atributos de los datos de entrada al destino (la respuesta que desea predecir). Genera un modelo de ML que captura estos patrones. Luego, el modelo de ML se puede utilizar para obtener predicciones sobre datos nuevos para los que no se conoce el destino.

herramienta

Una función o API que un agente puede invocar para realizar operaciones en sistemas externos.

puerta de enlace de tránsito

Centro de tránsito de red que puede utilizar para interconectar las VPC y las redes en las instalaciones. Para obtener más información, consulte Qué es una pasarela de tránsito en la AWS Transit Gateway documentación.

flujo de trabajo basado en enlaces troncales

Un enfoque en el que los desarrolladores crean y prueban características de forma local en una rama de característica y, a continuación, combinan esos cambios en la rama principal. Luego, la rama principal se adapta a los entornos de desarrollo, preproducción y producción, de forma secuencial.

acceso de confianza

Otorgar permisos a un servicio que especifique para realizar tareas en su organización AWS Organizations y en sus cuentas en su nombre. El servicio de confianza crea un rol vinculado al servicio en cada cuenta, cuando ese rol es necesario, para realizar las tareas de administración por usted. Para obtener más información, consulte AWS Organizations Utilización con otros AWS servicios en la AWS Organizations documentación.

ajuste

Cambiar aspectos de su proceso de formación a fin de mejorar la precisión del modelo de ML. Por ejemplo, puede entrenar el modelo de ML al generar un conjunto de etiquetas, incorporar etiquetas y, luego, repetir estos pasos varias veces con diferentes ajustes para optimizar el modelo.

equipo de dos pizzas

Un DevOps equipo pequeño al que puedes alimentar con dos pizzas. Un equipo formado por dos integrantes garantiza la mejor oportunidad posible de colaboración en el desarrollo de software.

U

incertidumbre

Un concepto que hace referencia a información imprecisa, incompleta o desconocida que puede socavar la fiabilidad de los modelos predictivos de ML. Hay dos tipos de incertidumbre: la incertidumbre epistémica se debe a datos limitados e incompletos, mientras que la incertidumbre aleatoria se debe al ruido y la aleatoriedad inherentes a los datos.

tareas indiferenciadas

También conocido como tareas arduas, es el trabajo que es necesario para crear y operar una aplicación, pero que no proporciona un valor directo al usuario final ni proporciona una ventaja competitiva. Algunos ejemplos de tareas indiferenciadas son la adquisición, el mantenimiento y la planificación de la capacidad.

entornos superiores

Consulte entorno.

V

succión

Una operación de mantenimiento de bases de datos que implica limpiar después de las actualizaciones incrementales para recuperar espacio de almacenamiento y mejorar el rendimiento.

control de versión

Procesos y herramientas que realizan un seguimiento de los cambios, como los cambios en el código fuente de un repositorio.

Emparejamiento de VPC

Conexión entre dos VPC que permite enrutar el tráfico mediante direcciones IP privadas. Para obtener más información, consulte ¿Qué es una interconexión de VPC? en la documentación de Amazon VPC.

vulnerabilidad

Defecto de software o hardware que pone en peligro la seguridad del sistema.

W

caché caliente

Un búfer caché que contiene datos actuales y relevantes a los que se accede con frecuencia. La instancia de base de datos puede leer desde la caché del búfer, lo que es más rápido que leer desde la memoria principal o el disco.

datos templados

Datos a los que el acceso es infrecuente. Al consultar este tipo de datos, normalmente se aceptan consultas moderadamente lentas.

función de ventana

Función SQL que hace un cálculo en un grupo de filas que se relacionan de alguna manera con el registro actual. Las funciones de ventana son útiles para las tareas de procesamiento, como calcular una media móvil o acceder al valor de las filas en función de la posición relativa de la fila actual.

carga de trabajo

Conjunto de recursos y código que ofrece valor comercial, como una aplicación orientada al cliente o un proceso de backend.

flujo de trabajo

Grupos funcionales de un proyecto de migración que son responsables de un conjunto específico de tareas. Cada flujo de trabajo es independiente, pero respalda a los demás flujos de trabajo del proyecto. Por ejemplo, el flujo de trabajo de la cartera es responsable de priorizar las aplicaciones, planificar las oleadas y recopilar los metadatos de migración. El flujo de trabajo de la cartera entrega estos recursos al flujo de trabajo de migración, que luego migra los servidores y las aplicaciones.

WORM

Consulte escritura única y lectura múltiple.

WQF

Consulte AWS Workload Qualification Framework.

escritura única y lectura múltiple (WORM)

Modelo de almacenamiento que escribe los datos una sola vez y evita que se eliminen o modifiquen. Los usuarios autorizados pueden leer los datos tantas veces como sea necesario, pero no los pueden cambiar. Esta infraestructura de almacenamiento de datos se considera inmutable.

Z

ataque de día cero

Ataque, normalmente de malware, que se aprovecha de una vulnerabilidad de día cero.

vulnerabilidad de día cero

Un defecto o una vulnerabilidad sin mitigación en un sistema de producción. Los agentes de amenazas pueden usar este tipo de vulnerabilidad para atacar el sistema. Los desarrolladores suelen darse cuenta de la vulnerabilidad a raíz del ataque.

peticiones desde cero

Proporcionar a un LLM instrucciones para llevar a cabo una tarea, pero sin ejemplos (pasos) que puedan ayudar a guiarlo. El LLM debe usar los conocimientos del entrenamiento previo para

llevar a cabo la tarea. La eficacia de la petición desde cero depende de la complejidad de la tarea y de la calidad de la petición. Consulte también peticiones con pocos pasos.

aplicación zombi

Aplicación que utiliza un promedio de CPU y memoria menor al 5 por ciento. En un proyecto de migración, es habitual retirar estas aplicaciones.

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la version original de inglés, prevalecerá la version en inglés.