Stratégies du Model Context Protocol sur AWS

AWS Conseils prescriptifs

AWS Conseils prescriptifs: Stratégies du Model Context Protocol sur AWS

Table of Contents

Stratégies du Model Context Protocol sur AWS

Amazon Web Services (contributeurs)

Mars 2026 (historique du document)

Ce guide peut vous aider à développer et à mettre en œuvre des stratégies MCP (Model Context Protocol) au sein de votre organisation afin de soutenir votre parcours vers l'IA agentique. Alors que les agents et les modèles linguistiques occupent une place de plus en plus centrale dans les opérations commerciales, la mise en place d'une stratégie MCP est essentielle au succès des solutions agentiques.

Ce guide explore trois piliers fondamentaux pour élaborer une stratégie MCP : la conception d'outils MCP, l'hébergement de serveurs MCP et la gouvernance MCP. En prenant en compte ces composants interconnectés, les entreprises peuvent créer des systèmes évolutifs, sécurisés et efficaces pour gérer le contexte des modèles dans le cadre de leurs implémentations d'IA. Ce guide fournit des informations exploitables et des conseils stratégiques aux organisations à toutes les étapes de leur parcours en matière d'IA, de l'expérimentation initiale aux déploiements de production à grande échelle. Cela les aide à développer des solutions MCP sur mesure qui répondent à leurs besoins et objectifs spécifiques.

Ces meilleures pratiques sont issues de mises en œuvre réelles d'organisations déployant le MCP à l'échelle de l'entreprise, d'une analyse des normes de spécification MCP actuelles et des leçons tirées des applications personnalisées de modèles de langage large (LLM) en production.

Les systèmes d'IA sont de plus en plus sophistiqués et robustes LLMs dans une grande variété de cas d'utilisation. LLMs excellent à comprendre le langage naturel, à générer des réponses semblables à celles des humains et à raisonner sur des informations complexes. Cependant, pour passer LLMs des interfaces conversationnelles à des systèmes capables d'accomplir des tâches complexes de manière autonome, les entreprises adoptent des architectures d'IA agentiques, des systèmes d'IA capables de percevoir leur environnement, de raisonner sur les objectifs, de prendre des décisions autonomes, d'orchestrer en plusieurs étapes et de prendre des mesures pour atteindre les objectifs au nom des utilisateurs. Cette approche agentique aide les entreprises à créer des systèmes d'IA capables de comprendre les intentions des utilisateurs grâce au langage naturel, de coordonner de manière autonome plusieurs sources de données et outils, et de proposer des expériences personnalisées à une échelle impossible avec les modèles traditionnels de demande-réponse. Pour renforcer les capacités de ces agents, les entreprises doivent fournir un accès à leurs

outils et données existants afin d'enrichir la compréhension contextuelle de l'agent et de lui permettre d'agir au nom de l'utilisateur.

MCP fournit un protocole standardisé pour l'intégration des outils d'intelligence artificielle, permettant une communication cohérente entre les agents et les ressources externes. Bien que MCP définisse lui-même la norme de communication, sa mise en œuvre efficace nécessite un examen attentif des modèles architecturaux, des modèles de sécurité, des pratiques opérationnelles et des stratégies d'optimisation des performances pour obtenir des solutions évolutives, sécurisées et maintenables.

Ce guide synthétise les leçons tirées des déploiements MCP en entreprise, en fournissant des recommandations pratiques conformes au Well-Architected Framework.AWS Il couvre les stratégies de conception d'outils MCP, d'hébergement de serveurs MCP et de gouvernance MCP, qui sont essentielles pour créer vos propres solutions MCP. Les recommandations de ce guide correspondent aux cinq piliers suivants du AWS Well-Architected Framework :

- Sécurité : isolation par jeton, informations d'identification limitées, autorisation séparée read/write
- Excellence opérationnelle — Sélection d'outils, mesures de précision, ensembles de données de référence pour les tests de régression
- Fiabilité : limitation du débit par utilisateur et par outil, délestage
- Efficacité des performances : outils adaptés au flux de travail, filtrage des outils, recherche sémantique pour réduire l'utilisation des fenêtres contextuelles
- Optimisation des coûts : serveurs MCP réutilisables au sein des équipes, réduction des coûts de jetons par demande grâce au filtrage des outils

Public visé

Ce guide est destiné aux architectes, aux développeurs et aux leaders technologiques qui mettent en œuvre des solutions d'IA agentic dans leurs organisations. Pour comprendre les concepts présentés dans ce guide, vous devez comprendre leur LLMs fonctionnement et avoir des connaissances de base sur le MCP, les outils et l'ingénierie rapide.

Objectifs

Pour créer des systèmes d'IA Agentic prêts pour la production, il faut résoudre ensemble les questions de gouvernance, d'optimisation et de sécurité afin de soutenir les politiques de votre organisation. Ce qui suit explique comment ce guide répond à ces objectifs :

- Gouvernance — Sans gouvernance centralisée, vous ne pouvez pas répondre aux questions d'audit concernant vos charges de travail liées à l'IA, notamment savoir quels agents ont accédé à quelles données, avec quelles autorisations et quand. Vous ne pouvez pas non plus appliquer le versionnement. La section de ce guide consacrée à la stratégie d'hébergement MCP explique comment les utilisateurs peuvent utiliser des serveurs MCP locaux obsolètes présentant des vulnérabilités connues en raison de l'absence de mesures d'application systématiques.

  Pour les industries réglementées, la gouvernance est essentielle. Les auditeurs souhaitent voir l'application des politiques et le suivi de l'utilisation des outils par tous les agents à partir d'un seul volet. C'est ce que permet la gouvernance du MCP.

  En suivant les recommandations de ce guide, vous pouvez améliorer la précision des tâches de 28 à 32 % dans les benchmarks évalués par des pairs. Pour plus d'informations, consultez MARCO : Multi-Agent Real-Time Chat Orchestration (site web de l'ACL Anthology). La gouvernance ne se limite pas à la conformité ; elle améliore également les performances de votre système d'IA agentic.
- Optimisation — Vos équipes peuvent créer les mêmes intégrations plusieurs fois. Par exemple, lorsque cinq équipes différentes écrivent leur propre script de requête de base de données pour que leur application d'IA communique avec leurs bases de données, cela représente cinq fois le coût de développement et cinq ensembles de listes de bogues à gérer. MCP vous permet de le créer une seule fois et de le partager avec l'ensemble de la communauté des ingénieurs. Les économies s'accumulent à mesure que le nombre de vos agents augmente.

  Il existe également un problème de coût par demande que la plupart des équipes ne remarquent pas au début. Chaque définition d'outil consomme des jetons de fenêtre contextuelle. Avec 20 outils, vous dépensez 5 000 à 10 000 jetons par invocation uniquement pour les descriptions, ainsi que pour les demandes des utilisateurs. Cela augmente la latence et les coûts d'inférence LLM et dégrade la précision, car le modèle peine à choisir le bon outil dans la liste des outils disponibles.

  Les agents qui utilisent des wrappers d'outils structurés sont environ trois fois plus précis dans les tâches de base de données que les agents qui y accèdent APIs directement (pour plus d'informations, voir Middleware for LLMs : Les outils sont essentiels pour les agents linguistiques dans les environnements complexes). La façon dont vous concevez et présentez les outils dans un modèle d'IA est importante. Ce guide recommande de donner aux outils des schémas clairs, de les adapter aux flux de travail réels plutôt qu'aux points de terminaison bruts et de limiter les informations dans la fenêtre contextuelle. La section de ce guide consacrée à la stratégie de conception des outils MCP analyse en profondeur ces aspects.

- Sécurité et conformité — Imaginez un système d'IA agentique qui hallucine une étape de nettoyage et tente de supprimer une base de données de production. Si l'agent a hérité des informations d'identification complètes de l'administrateur de l'utilisateur, la suppression peut être effectuée. Grâce à l'isolation par jeton et à des informations d'identification limitées qui n'accordent qu'un accès en lecture et en création, il échoue en toute sécurité.

  Les flux de travail régulés renforcent encore ce point. Le guide fournit des exemples (pipelines de soins de santé qui nécessitent la validation HIPAA et l'anonymisation des informations personnellement identifiables avant de traiter les données des patients). L'intégration d'une telle logique dans les outils MCP signifie que la conformité se fait de manière déterministe à chaque fois.

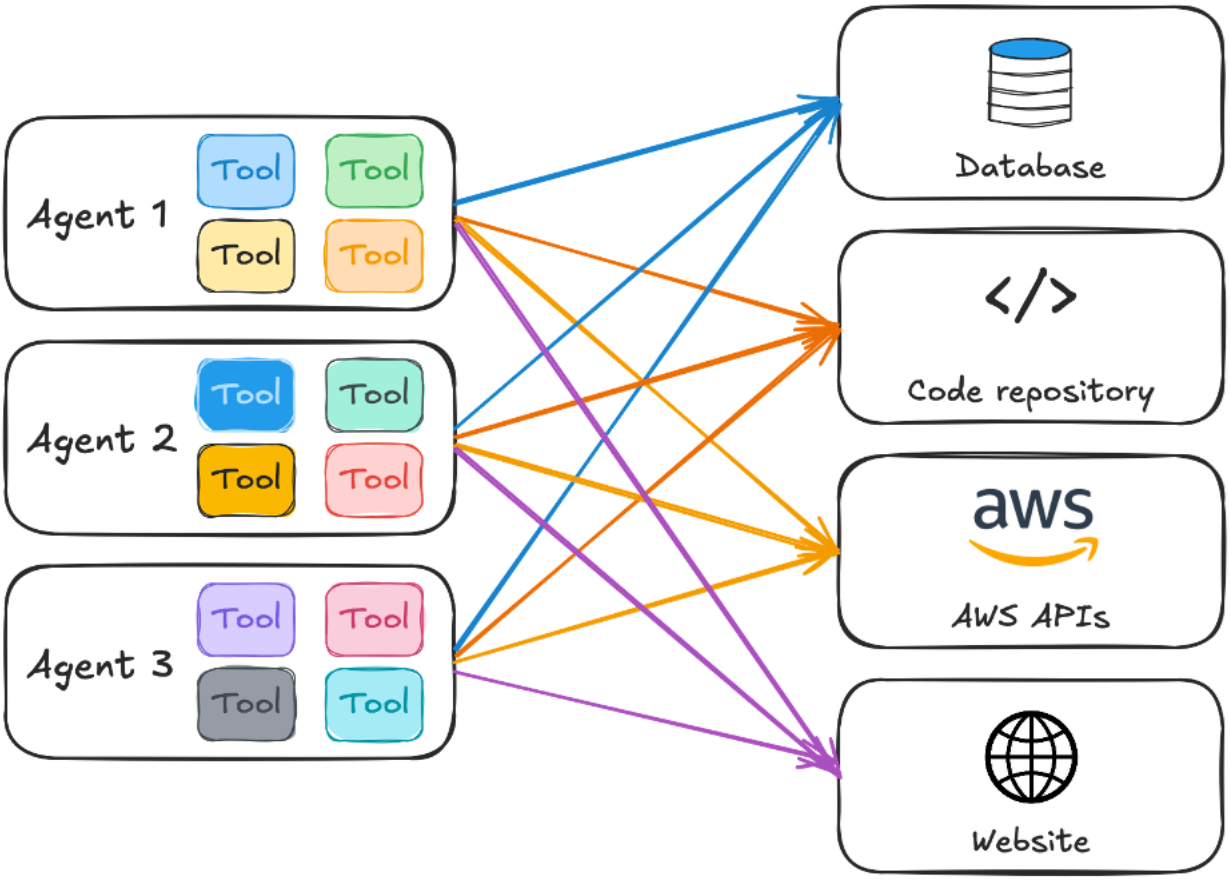
Qu'est-ce que le MCP ?

LLMs fonctionnent en prédisant une réponse à une question en fonction de leurs données d'entraînement. Cela signifie que le LLM ne peut fournir des réponses que sur les données et les événements qu'il a déjà vus. Des méthodes telles que la génération augmentée de récupération (RAG) et les bases de connaissances vous permettent d'inclure des données contextuelles. Cependant, si vous demandiez à un LLM quelles seront les prévisions météorologiques de demain ou combien de clients figurent dans votre base de données, il hallucinerait probablement ou ne serait pas en mesure de fournir une réponse, car ces questions ne relèvent pas des connaissances préétablies du LLM. Pour être en mesure de répondre à ce type de questions, un agent doit avoir accès à des fonctionnalités externes, à des données et APIs en dehors du contexte natif du LLM.

Comprendre les outils

Nous pouvons donner au LLM l'accès à des systèmes et à un contexte supplémentaires grâce à des outils.Les outils sont des fonctions confiées au LLM pour atteindre un objectif clair. Un outil peut appeler une API, interroger une base de données, effectuer des opérations de calcul, exploiter un sandbox de code, effectuer une recherche sur le Web et même invoquer un autre système d'IA ou agent-as-a-tool. Chaque outil doit inclure une description indiquant au LLM ce que fait l'outil, quand l'utiliser et quels paramètres il accepte. Cela permet au LLM de prendre des décisions nuancées quant à l'outil ou à la combinaison d'outils à invoquer en fonction des entrées de l'utilisateur. Le LLM est informé des outils mis à la disposition de l'agent, ce qui lui permet de générer des réponses demandant à l'agent d'invoquer l'outil. Par exemple, lorsque vous demandez au LLM combien de clients se trouvent dans votre base de données, le LLM renvoie une réponse à l'agent demandant d'exécuter l'`query_database`outil avec des paramètres d'entrée spécifiques. Le LLM détermine l'outil à invoquer et les entrées pour l'appel d'outil. L'agent exécute ensuite l'outil, qui convertit l'entrée en langage naturel en un appel de fonction syntaxiquement correct et exécute la requête. L'agent invoque l'outil ou les outils en fonction des instructions du LLM, et ces résultats sont renvoyés au LLM. Cela tire parti de la capacité du LLM à raisonner plutôt qu'à saisir du texte et à sélectionner les outils appropriés pour le travail.

L'image suivante montre comment chaque agent gère son propre ensemble d'outils pour chaque cible.



L'extension de l'accès aux outils peut présenter des défis pour les solutions d'IA agentic :

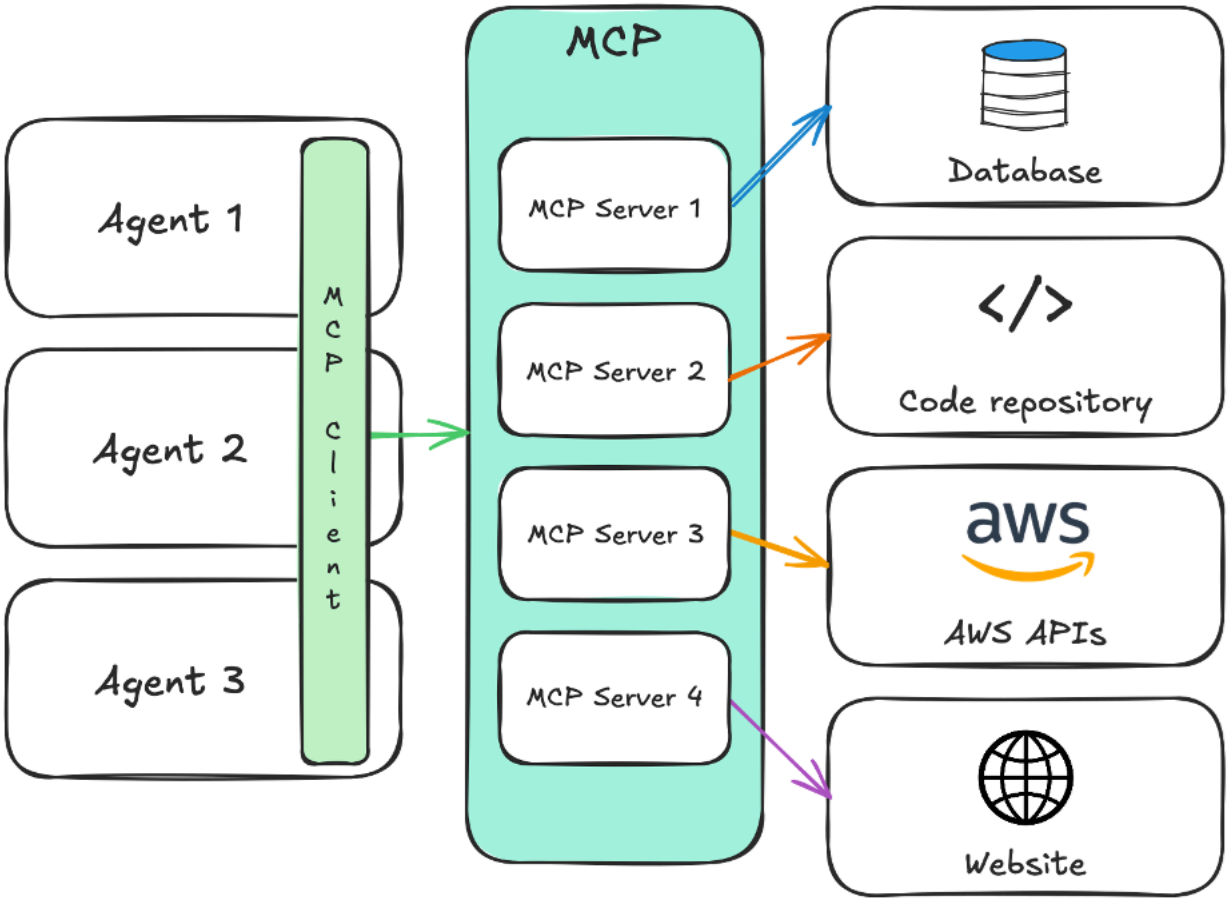
- Si chaque développeur crée son propre outil pour les mêmes fonctionnalités externes, les efforts sont dupliqués et les méthodes d'interaction avec ces fonctionnalités externes ne sont pas standardisées. Cela produit des implémentations incohérentes entre vos agents. Bien que vous puissiez résoudre ce problème en développant des outils standard dans des bibliothèques et en les distribuant, cela manque de gouvernance centralisée. Cela complique l'application des politiques de sécurité, le suivi de l'utilisation des outils, la gestion des versions entre les équipes ou le respect des normes organisationnelles. En outre, lorsque vous intégrez des outils directement à l'agent, vous devez redéployer votre agent chaque fois qu'un nouvel outil est créé ou qu'un outil existant est mis à jour.
- Fournir des outils à un LLM consomme sa fenêtre contextuelle. La fenêtre contextuelle représente le nombre de jetons (unités de texte LLMs traitées, représentant généralement des mots, des parties de mots ou des signes de ponctuation) qu'un modèle peut prendre en compte à tout moment. LLMs ont une limite de fenêtre contextuelle. Les outils et leur documentation utilisent cette fenêtre contextuelle limitée ainsi que les instructions du système et celles des utilisateurs. Lorsque la fenêtre contextuelle se remplit, les performances LLMs peuvent être dégradées en

raison de multiples facteurs : difficulté à identifier les informations pertinentes, complexité accrue du traitement et capacité de raisonnement réduite. Le défi est d'autant plus grand lorsque les définitions d'outils, les instructions du système et l'historique des conversations se disputent un espace limité dans les fenêtres contextuelles, car ils sont fournis à chaque appel de LLM.

Ainsi, le nombre d'outils et la manière dont ils sont documentés ont un impact direct sur les performances du LLM, telles que le temps de réponse et la précision.

Le MCP établit une norme universelle pour connecter les agents à des capacités externes. Il est communément appelé « USB-C pour les applications d'IA ». Au lieu d'enregistrer les outils directement auprès des agents, les serveurs MCP servent d'intermédiaire pour héberger les outils découverts et invoqués via JSON-RPC 2.0. Au lieu d'ajouter des dizaines ou des centaines d'outils différents à votre agent et de les maintenir au fil du temps, MCP vous permet d'enregistrer des serveurs MCP qui encapsulent les outils auxquels votre agent peut accéder. Cette approche normalise la manière dont les outils sont empaquetés, présentés et invoqués. Cela peut aider à relever les défis d'échelle et de gouvernance liés à l'utilisation des outils au sein de vos agents. Il dissocie également le développement et les opérations des agents des outils qu'il utilise pour les capacités externes.

La figure suivante montre les agents utilisant le protocole MCP pour accéder à des ressources externes.



Cependant, la norme MCP ne résout pas tous les problèmes de mise à l'échelle et de gouvernance. La mise en œuvre de serveurs MCP doit être associée à des stratégies efficaces de conception d'outils, d'hébergement et de gouvernance d'entreprise. Ce guide fournit les meilleures pratiques pour chaque stratégie afin de vous aider à créer et à utiliser MCP dans le cadre de vos solutions d'IA agentic.

Quand utiliser le MCP

MCP fournit une infrastructure stratégique pour étendre vos initiatives d'intelligence artificielle agentique. En centralisant la gestion et la gouvernance des outils, les serveurs MCP réduisent le coût cumulé de création et de maintenance d'intégrations personnalisées entre plusieurs agents. Cela permet d'augmenter les rendements au fur et à mesure que votre écosystème d'agents se développe.

Le MCP fait probablement partie de votre stratégie lorsque :

- Vous avez besoin d'une gouvernance centralisée pour la manière dont les agents accèdent aux systèmes et services de l'entreprise, tels que les bases de données APIs, les outils internes et les intégrations tierces.
- Les développeurs passent trop de temps à écrire des intégrations personnalisées qui ne sont pas cohérentes entre les implémentations.
- Vous disposez d'outils dupliqués susceptibles de répondre à des fonctionnalités communes.
- Vous souhaitez proposer vos outils ou données propriétaires à des consommateurs externes ou à des systèmes agentiques tiers via des interfaces MCP normalisées et gouvernées, afin de débloquer de nouvelles sources de revenus tout en préservant la sécurité et le contrôle.

Après avoir décidé que les serveurs MCP feront partie de votre stratégie, déterminez si les implémentations de serveurs MCP open source existantes répondent à vos besoins, si elles doivent être améliorées ou si vous devez créer des serveurs personnalisés. De nombreuses implémentations de serveurs MCP prédéfinies sont disponibles dans des référentiels publics et couvrent des fonctionnalités courantes telles que l'accès au système de fichiers, la navigation sur le Web, les bacs à sable de code, l'accès aux bases de données et les intégrations d'API.

Dans de nombreux cas, les serveurs MCP préexistants sont suffisants. Par exemple, AWS fournit un serveur MCP distant géré qui fournit aux assistants et aux agents IA un accès sécurisé et authentifié Services AWS via des interactions en langage naturel. Serveur AWS MCP Vous pouvez l'utiliser Serveur AWS MCP pour effectuer des AWS tâches complexes en plusieurs étapes en combinant un accès en temps réel à la AWS documentation, des appels d'API syntaxiquement corrects et des flux de travail prédéfinis appelés Agent SOPs qui suivent les meilleures pratiques. AWS AWS les teste en permanence Serveur AWS MCP pour s'assurer que les agents clients peuvent les utiliser avec succès.

Vous devez tester ces serveurs MCP existants avec vos agents afin de déterminer s'ils répondent à vos besoins d'utilisation. Si un agent ne parvient pas à terminer les flux de travail, génère des réponses incorrectes ou sous-optimales, ne parvient pas à gérer des processus complexes en plusieurs étapes, ou ne respecte pas les meilleures pratiques ou considérations de sécurité spécifiques à un domaine, vous devez envisager des améliorations dans plusieurs domaines.

Lorsque les serveurs MCP existants ne répondent pas entièrement à vos besoins et qu'ils ont du mal à utiliser correctement les outils existants ou à produire des réponses précises, envisagez ces approches d'amélioration avant de créer des serveurs personnalisés :

- Enrichissez le contexte de l'agent : si votre agent peine à utiliser correctement ou efficacement les outils d'un serveur MCP existant, pensez à compléter ces définitions d'outils par de la documentation ou des exemples supplémentaires. Cela permet de fournir un contexte supplémentaire au LLM.
- Ajoutez des outils complémentaires : étendez les serveurs MCP existants avec des outils qui accèdent à des données organisationnelles ou à un contexte supplémentaires dont les agents ont besoin pour mener à bien les flux de travail.
- Améliorez les sous-jacents APIs : simplifiez votre service APIs pour le rendre plus convivial en matière de LLM en réduisant la complexité des paramètres, en diffusant des messages d'erreur plus clairs et en proposant des valeurs par défaut judicieuses que les agents peuvent utiliser.

Alors que l'utilisation des implémentations de serveurs MCP existantes accélère le développement de fonctionnalités communes, la création de serveurs MCP personnalisés est une nécessité lorsque votre cas d'utilisation nécessite des fonctionnalités spécialisées. Les serveurs MCP personnalisés vous aident à encapsuler l'expertise du domaine, à appliquer les normes organisationnelles, à améliorer la fiabilité des agents pour les flux de travail complexes et à garantir la conformité aux exigences de sécurité. Envisagez de créer un serveur MCP personnalisé dans les situations suivantes :

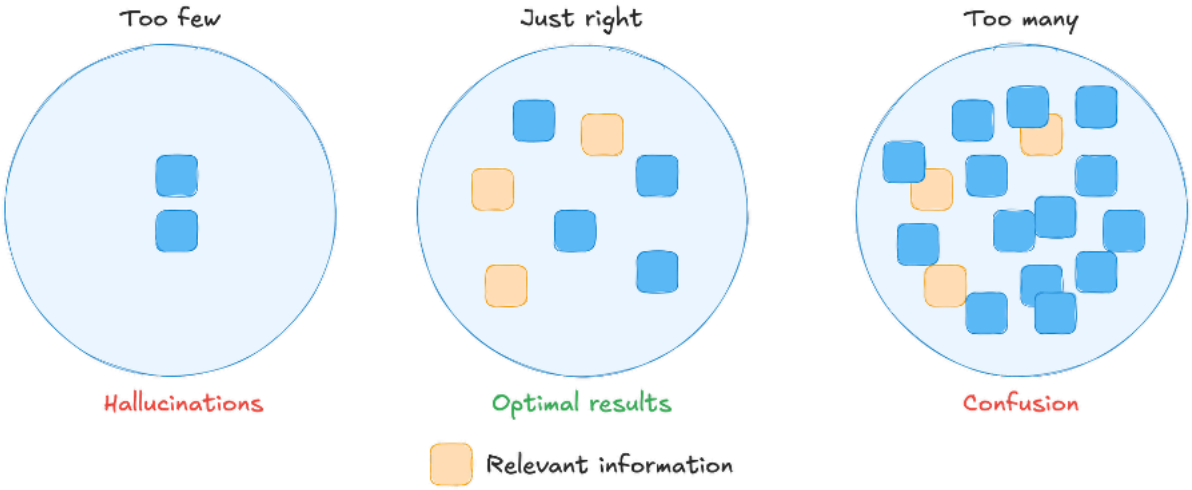
- Flux de travail spécifiques au domaine — Les flux de travail en plusieurs étapes nécessitant une expertise du domaine doivent être encapsulés dans des outils MCP personnalisés lorsque les connaissances nécessaires ne sont pas capturées dans la documentation de l'API. Par exemple, au lieu de laisser les agents orchestrer des pipelines de données de santé complexes qui doivent valider la conformité à la loi HIPAA (Health Insurance Portability and Accountability Act), anonymiser les informations personnelles et passer au format HL7 FHIR, fournissez un `process_patient_data` outil qui intègre directement l'expertise du domaine. Cela élimine la dépendance à l'égard du LLM pour orchestrer et exécuter correctement les étapes du flux de travail, ce qui améliore la cohérence et la conformité.
- Abstractions de la voie dorée : les agents peuvent avoir du mal à mettre en œuvre des approches optimales car ils ne disposent pas d'un contexte organisationnel et optent par défaut sur des modèles de base plutôt que sur les meilleures pratiques organisationnelles. Dans ces scénarios, vous pouvez appliquer des normes prescriptives en matière de coûts, de performances ou de sécurité en encapsulant ces voies privilégiées dans des outils MCP personnalisés. Par exemple, au lieu de laisser les agents déployer une infrastructure avec des paramètres par défaut qui peuvent être peu sécurisés ou inefficaces, fournissez un `deploy_secure_infrastructure` outil qui intègre directement les normes de votre entreprise.

- Orchestration multiservice complexe : au lieu de demander à l'agent d'orchestrer des flux de travail complexes en essayant de déduire la séquence et le jeu de services appropriés à utiliser à chaque étape, vous pouvez créer cette logique de manière déterministe dans un outil MCP. Vous souhaiterez peut-être également fournir une expertise sur les modèles d'intégration de services optimaux dont l'agent n'est peut-être pas au courant. Cela peut également améliorer la précision et l'efficacité de vos agents.
- Bonnes pratiques spécifiques aux services : cela est courant pour les outils axés sur la sécurité qui aident les agents à mettre en œuvre des politiques de chiffrement, des contrôles d'accès et des modèles de conformité spécifiques au service auquel ils accèdent via l'outil d'agent. En outre, si certaines bonnes pratiques opérationnelles spécifiques à un service ne sont pas évidentes, l'utilisation d'un serveur MCP peut vous aider à vous assurer qu'elles sont mises en œuvre et qu'elles ne sont pas laissées à l'appréciation d'un agent.

Stratégie de conception d'outils MCP

La tâche principale du client et du serveur MCP est de découvrir et de présenter des outils au LLM afin qu'il puisse les utiliser pour améliorer ses réponses. Cela fait de la conception d'outils MCP l'une des stratégies les plus importantes pour créer des solutions MCP efficaces. Du point de vue du modèle, les outils sont une fonction qu'ils peuvent invoquer selon les besoins pour fournir des réponses plus précises et complètes. L'interface fonctionnelle résume l'implémentation sous-jacente d'un outil, qui peut aller d'un wrapper autour d'un seul appel d'API à une logique de flux de travail complexe.

Cependant, vous devez trouver un équilibre avec la quantité d'outils fournis au LLM. S'il y a trop peu d'outils, le LLM risque de ne pas être en mesure de collecter le contexte et les informations appropriés. Il fera donc les meilleures estimations avec les informations disponibles dans le modèle. S'il y a trop d'outils, le LLM peut être confus quant à la sélection et à la séquence d'outils appropriés, ce qui peut entraîner des hallucinations. Votre objectif est d'obtenir le bon nombre d'outils. L'image suivante montre les difficultés liées à un trop petit nombre ou à un trop grand nombre d'outils.



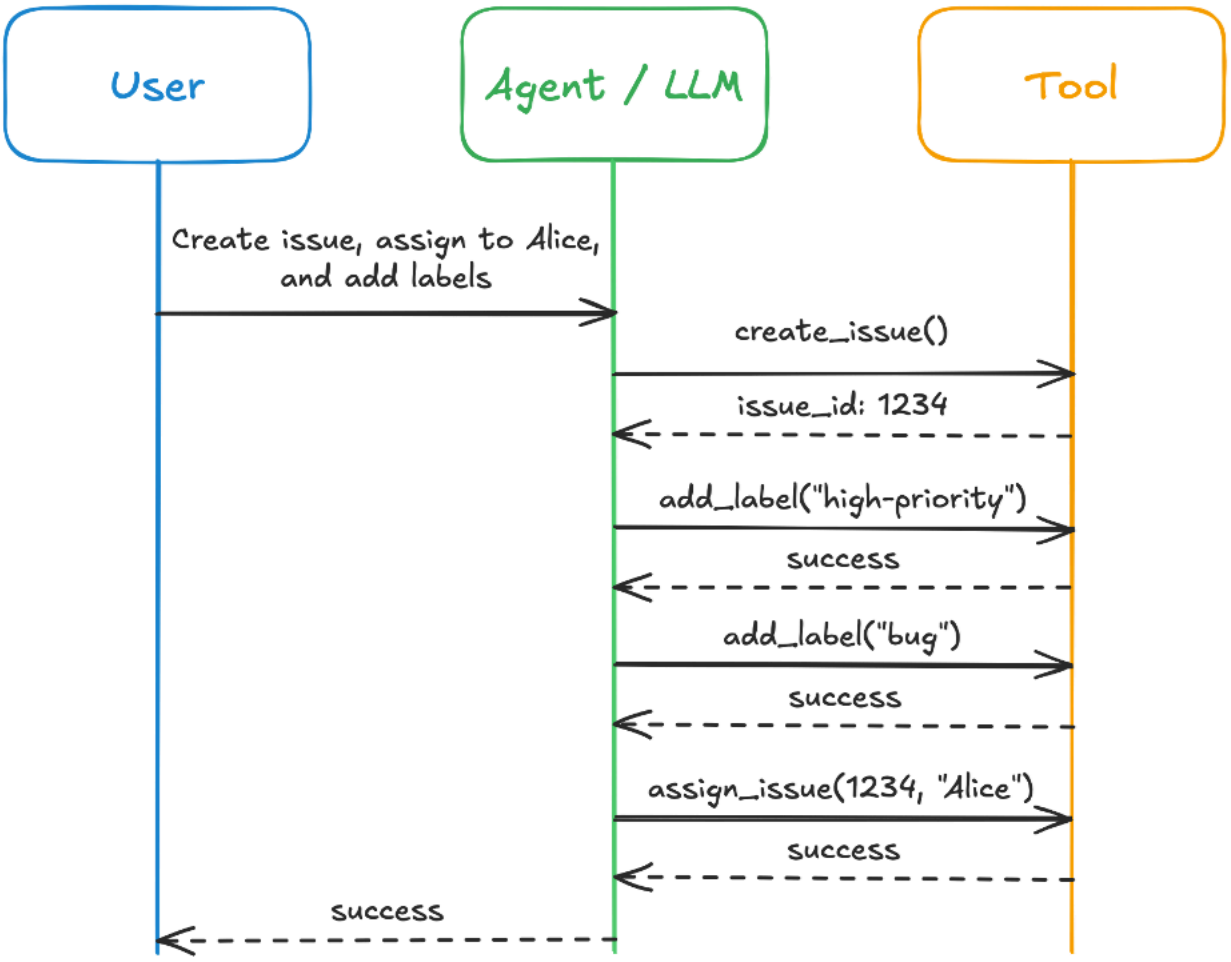
La solution nécessite de comprendre le nombre d'outils à fournir et la manière de définir le périmètre de chaque outil. La granularité de vos outils, qu'ils correspondent à des appels d'API individuels ou à des flux de travail complets, a un impact direct sur le nombre total d'outils dont les agents ont besoin et sur l'efficacité avec laquelle ils peuvent les utiliser. Cette section fournit les meilleures pratiques pour délimiter les outils MCP, créer des définitions d'outils, découvrir des outils et les organiser.

Portée de l'outil

Il existe deux approches pour développer des outils : granulaire et grossier.

Granulaire

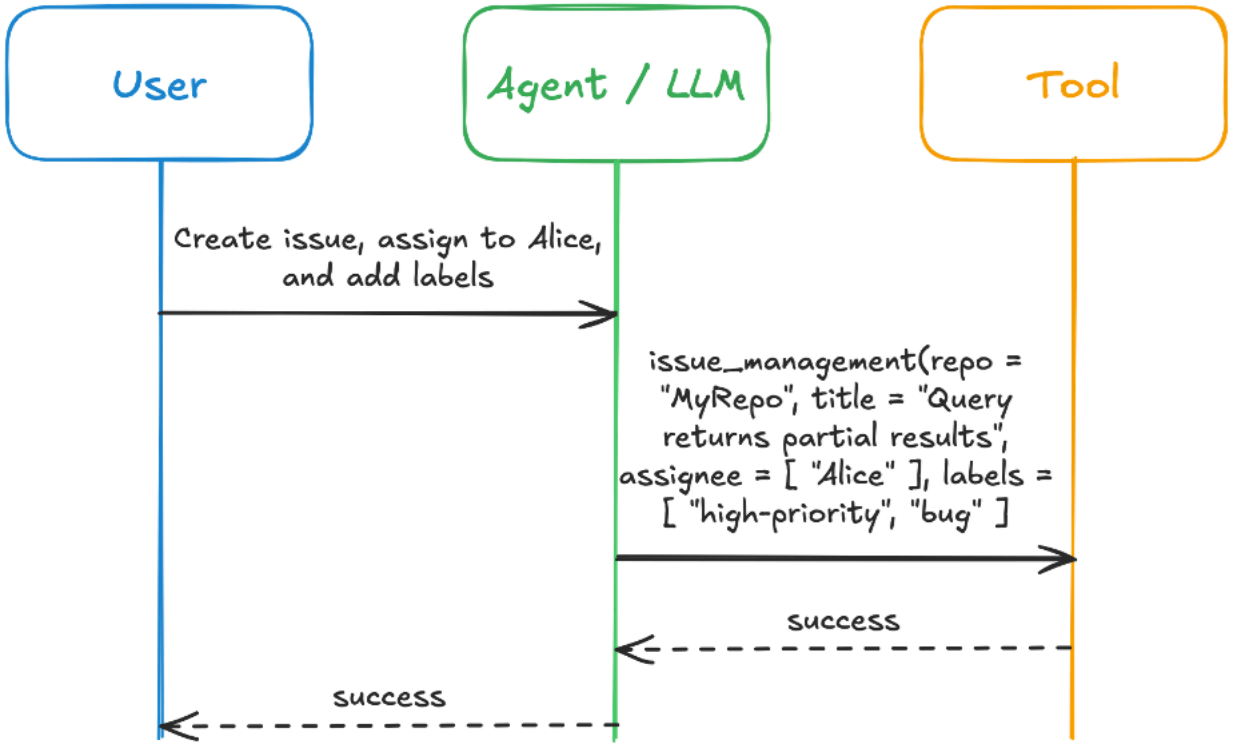
Dans une approche granulaire, vous créeriez un outil par API, action ou requête. Par exemple, vous pouvez créer des`create_issue`,`get_issue`, `add_label`ou `assign_issue`, et `close_issue` des outils pour votre dépôt Git. Cela permettrait au LLM de faire des appels granulaires à chaque API et de les orchestrer selon les besoins. Tenez compte de l'invite suivante : « Créez un problème pour le service produit intitulé « La requête ne renvoie que des résultats partiels », qualifiez-le de bogue et de prioritaire, et attribuez-le à Alice. » L'image suivante montre comment une tool-per-API approche répondrait à cette invite.



Dans cette approche, l'invite du système et chaque définition d'outil enregistrée sont fournies au LLM à chaque appel. Cela consomme du contexte supplémentaire et entraîne une pénalité de latence, car chaque appel d'outil représente un appel individuel au LLM. Cela augmente également la complexité de la gestion des erreurs dans le flux de travail.

À gros grains

Une approche grossière, ou axée sur le flux de travail, consisterait à utiliser des outils orientés vers le flux de travail. L'outil met l'accent sur end-to-end l'intention de l'utilisateur plutôt que sur la structure de l'API. Au lieu de a tool-per-API, vous avez un outil qui en appelle plusieurs de manière déterministe. APIs À l'aide de l'exemple de dépôt Git précédent, vous pouvez créer un `create_and_setup_issue` outil appelé une fois par l'agent. L'implémentation de l'outil crée le problème, ajoute des étiquettes et l'attribue à un utilisateur, en fonction des paramètres fournis à l'outil. L'image suivante montre comment une approche grossière traiterait la même invite.



Cette approche montre comment toute la complexité reste cachée dans la couche LLM. Lorsque la logique d'orchestration est intégrée à l'implémentation de l'outil, toutes les étapes séquentielles, la journalisation, la logique de nouvelle tentative, les disjoncteurs et la limitation de débit sont effectuées de manière déterministe dans l'outil. L'approche axée sur le flux de travail permet au LLM d'appeler plus facilement le bon outil avec les bons paramètres. Il est important de noter que certaines API peuvent déjà fournir une intention de flux de travail, comme l'API Amazon EC2`RunInstances`. Dans ces cas, a tool-per-API peut fournir la conception axée sur le flux de travail que vous souhaitez.

Cependant, les outils peuvent également devenir trop grossiers. Si votre seul outil de flux de travail tente de faire trop de choses et comporte de nombreux paramètres possibles, le LLM peut avoir du mal à raisonner sur la manière d'utiliser correctement l'outil. Cela peut également créer des

difficultés en termes de sélection des paramètres et de gestion des erreurs. Ainsi, le développement d'outils doit trouver un équilibre qui correspond aux intentions de l'utilisateur et qui évite d'utiliser trop ou trop de fonctionnalités dans un seul outil. Nous vous recommandons de concevoir des outils basés sur des flux de travail utilisateur complets, en regroupant les opérations qui se produisent généralement ensemble (comme trois appels d'API ou plus). Nous vous recommandons également de décomposer les outils qui dépassent huit paramètres ou plus ou qui gèrent plusieurs intentions distinctes de l'utilisateur. Testez avec de vraies instructions pour vérifier que les agents peuvent utiliser correctement chaque outil.

Si vous avez des flux de travail complexes et dynamiques qui ne peuvent pas être facilement encapsulés en tant qu'outil déterministe, vous pouvez envisager d'utiliser le modèle. agent-as-tool Au lieu que votre agent principal essaie d'orchestrer des tâches complexes dans un flux de travail, un agent spécialisé peut agir comme un outil. Ces types d'outils peuvent mettre en œuvre des processus décisionnels et des branchements avancés, et ils peuvent gérer les erreurs et réessayer une logique qui ne peut pas être facilement gérée dans un code déterministe. Ce protocole est similaire mais distinct du protocole Agent2Agent (A2A). Le protocole A2A est complémentaire, assurant l'interopérabilité et la collaboration entre les agents dans n'importe quel cadre agentique.

Nous vous recommandons de commencer par l'analyse de votre flux de travail en cartographiant vos flux de travail utilisateur les plus courants afin d'identifier les fonctionnalités de base dont chaque agent a besoin. Cela permet d'établir votre ensemble d'outils minimum viable. Sur la base de notre expérience dans le développement de serveurs MCP à grande échelle, nous recommandons les pratiques suivantes. En cas de conflit entre ces pratiques, priorisez l'intention de l'utilisateur et le flux de travail.

Bonnes pratiques en matière de cadrage des outils MCP

- Pensez aux témoignages d'utilisateurs et regroupez les opérations courantes : les outils doivent correspondre directement à l'ensemble des interactions avec les utilisateurs plutôt que de nécessiter l'orchestration de plusieurs opérations. Si les flux de travail nécessitent généralement au moins trois appels distincts, combinez-les dans un seul outil. Cela réduit la charge cognitive du LLM, minimise le nombre d'appels d'outils, réduit la consommation de contexte et la latence nécessaires à l'exécution des tâches, et améliore la précision et la latence.
- Limiter les paramètres à huit ou moins : si un outil dépasse huit paramètres, décomposez-le en plusieurs outils. LLMs difficulté à sélectionner les paramètres à mesure que la complexité augmente.

> **Note**
>
> Si les opérations de regroupement nécessitent plus de huit paramètres, privilégiez le regroupement plutôt que le nombre de paramètres, car la simplification du flux de travail est plus précieuse que des limites de paramètres strictes.

- Opérations de lecture et d'écriture distinctes : fournissez différents outils pour lire les données et les modifier. Cette séparation indique clairement quand les agents exécutent des opérations potentiellement destructrices, permet d'appliquer différentes politiques d'autorisation et réduit le risque de modifications involontaires lors de la collecte d'informations.
- Fournissez des valeurs par défaut raisonnables — Outils de conception afin que le LLM ne doive spécifier que les paramètres spécifiques à la demande individuelle. Les valeurs par défaut réduisent la complexité des paramètres et améliorent la précision de la sélection des outils en minimisant les informations sur lesquelles le LLM doit raisonner.
- Préférez l'exécution déterministe — Rendez l'exécution de l'outil et la sortie déterministes lorsque cela est possible. Les outils déterministes sont plus fiables et plus faciles à tester. Pour les flux de travail complexes qui nécessitent une orchestration intelligente, une logique de branchement ou une gestion avancée des erreurs qui ne peuvent pas être facilement gérés dans un code déterministe, envisagez d'utiliser des agents spécialisés comme outils. Toutefois, utilisez ce modèle de manière sélective car il ajoute de la complexité.

Définitions des outils

Lorsqu'un LLM reçoit une demande qu'il ne peut pas traiter directement, il passe en revue les outils disponibles pour l'aider à compléter la demande. Le LLM sélectionne les outils en fonction de sa compréhension sémantique des noms et des descriptions des outils fournis et des instructions fournies dans l'invite. Il créera ensuite une entrée basée sur le schéma d'entrée défini et s'attend à une sortie basée sur le schéma de sortie. Par conséquent, la création de définitions d'outils descriptives et de schémas d'entrée et de sortie validés est essentielle pour aider le LLM à sélectionner efficacement les outils. Il existe généralement deux approches pour créer cette documentation : l'approche de spécification de l'outil et l'approche docstring.

Approche de spécification des outils

L'approche recommandée consiste à suivre directement les spécifications de l'outil MCP lors de la définition de l'outil. L'exemple suivant est illustré à l'aide du décorateur d'outils 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:
  …
```

L'utilisation de champs standard, tels que `name`,`description`,`inputSchema`, et `outputSchema` garantit que chaque outil dispose d'une documentation cohérente que le LLM et les humains peuvent comprendre. Chaque outil doit définir ces champs au minimum et éventuellement fournir un titre et des annotations, qui sont des indications facultatives sur le comportement de l'outil. Dans la mesure du possible, utilisez des enums pour les valeurs des paramètres afin de permettre au LLM de sélectionner facilement les bonnes options. Les énumérations fonctionnent mieux pour les ensembles

finis, tels que les valeurs de statut ou de priorité, mais ne conviennent pas au texte de forme libre, aux valeurs dynamiques, aux nombres arbitraires ou aux identificateurs de ressources. Dans ces cas, fournissez plutôt des descriptions et des exemples clairs. Incluez également une valeur par défaut lorsque cela est possible afin que le LLM n'ait pas à deviner quelle est la bonne option. N'oubliez pas que les définitions d'outils sont incluses dans l'invite LLM à chaque appel, ce qui consomme de l'espace dans la fenêtre contextuelle aux côtés des instructions système et de l'historique des conversations.

Approche Docstring

Une autre approche, si vous écrivez vos outils en Python, consiste à utiliser des docstrings pour fournir la description, l'utilisation et le résultat de l'outil. Voici un exemple de cette approche :

```
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
    """
```

Les Docstrings n'appliquent pas de schéma ou de format standardisé. L'utilisation de cette approche peut donner des résultats incohérents en fonction de la manière dont les développeurs d'outils choisissent de documenter chaque outil. La définition et l'application d'une norme à l'échelle de l'organisation sont essentielles si vous suivez cette approche.

Bonnes pratiques pour les définitions d'outils MCP

- Suivez les spécifications de l'outil MCP : `name` indiquez`description`,`inputSchema`, et des `outputSchema` champs pour chaque outil. Pour les implémentations de Python, utilisez les modèles Pydantic pour fournir une documentation en ligne via des descriptions de champs, une validation automatique des types et des valeurs contraintes via des énumérations. Cela permet aux schémas de s'auto-documenter et améliore la compréhension LLM des options de paramètres valides.
- Rédigez les descriptions sous forme d'instructions — Les descriptions des outils sont des instructions qui guident la prise de décision en matière de LLM. Incluez les éléments essentiels

de l'objectif de l'outil (ce que fait l'outil), le moment où il doit être utilisé (modèles d'intention de l'utilisateur ou scénarios), le contexte de la sortie (à quoi sert la sortie), les paramètres et les conditions d'erreur.
- Fournissez des exemples concrets — L'inclusion d'exemples de flux de travail avec des valeurs réelles est le moyen le plus efficace de vous guider LLMs sur l'utilisation correcte des outils.
- Documentez les dépendances de manière explicite : incluez les prérequis, les séquences numérotées, les changements d'état et les actions de suivi.

Découverte d'outils

Il existe trois approches pour découvrir et enregistrer des outils dans votre agent auprès des serveurs MCP : définition statique, découverte dynamique et fonction de recherche.

Définition statique

Tout d'abord, vous pouvez définir statiquement les outils disponibles directement dans le code de l'agent. Dans cette approche, vous définissez un outil distant (un objet de référence côté client dans un framework tel que Strands Agent SDK) pour chaque outil fourni par le serveur MCP auquel un client MCP accède. L'exemple suivant utilise le transport HTTP streamable :

```
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])
```

L'enregistrement individuel des outils vous permet d'être très sélectif quant aux outils que vous mettez à la disposition du LLM, ce qui minimise le nombre de fenêtres contextuelles utilisées. L'inconvénient est que cela nécessite de connaître le nom des outils disponibles et peut être fragile si les outils disponibles changent sur le serveur MCP.

Découverte dynamique

L'approche suivante consiste à utiliser la découverte dynamique et à enregistrer tous les outils disponibles auprès de l'agent. Cette approche utilise le contexte de manière linéaire à mesure que de nouveaux outils sont ajoutés au serveur MCP. Voici un exemple de cette approche :

```
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)
```

Imaginons un scénario dans lequel une définition d'outil typique consomme environ 250 à 500 jetons (y compris le nom, la description et le schéma). L'enregistrement de 20 outils consommerait entre 5 000 et 10 000 jetons de votre fenêtre contextuelle. Lorsque vous disposez d'un petit nombre de serveurs MCP et que vous contrôlez le nombre d'outils, cette option est la plus simple à mettre en œuvre. Toutefois, si l'on s'attend à ce que la liste des outils augmente, cela peut créer des problèmes de gestion du contexte silencieux chez vos agents. Une autre variante de cette approche consiste à utiliser un paramètre de filtre d'outil lors de l'appel`list_tools`, tel que celui fourni par le SDK Strands Agents, afin de réduire le nombre d'outils enregistrés auprès de l'agent.

Fonction de recherche

La troisième option consiste à utiliser une fonction de recherche pour trouver les outils pertinents pendant l'exécution. Vous listez tous les outils disponibles sur votre serveur MCP, puis vous effectuez une recherche sémantique sur ces outils en fonction de l'invite de l'utilisateur. Ensuite, les outils obtenus sont enregistrés auprès de votre agent. Amazon Bedrock AgentCore Gateway fournit une fonctionnalité de recherche sémantique native qui peut faciliter la mise en œuvre de ce type de solution.

Bonnes pratiques pour la découverte d'outils MCP

- Préservation de la fenêtre contextuelle — Choisissez une approche de découverte et d'enregistrement d'outils qui préserve autant que possible votre fenêtre contextuelle.

- Utilisez des fonctionnalités de filtrage d'outils ou de recherche sémantique — Fournissez dynamiquement au LLM un ensemble d'outils parmi lesquels choisir, ce qui améliore sa précision et son efficacité lorsqu'il s'agit de choisir le bon outil. Le filtrage des outils peut fonctionner sur les noms des outils (correspondance exacte ou modèles), les descriptions des outils (correspondance sémantique) ou les balises de domaine ou de catégorie. La recherche sémantique est particulièrement efficace pour faire correspondre l'intention de l'utilisateur aux descriptions des outils. Les deux approches réduisent l'utilisation des fenêtres contextuelles.

Organisation des outils

Découvrir les bons outils et s'assurer que le LLM peut les utiliser efficacement est l'un des éléments les plus critiques d'un développement d'outils efficace. Lorsque vous commencez à développer des serveurs MCP, vous avez besoin d'une stratégie qui détermine :

- Combien d'outils sont inclus dans un serveur MCP
- Quels outils ne doivent pas être placés dans le même serveur MCP
- Comment nommer les outils pour les rendre consultables et éviter les collisions de noms (différents outils portant le même nom)
- Comment documenter les outils et le serveur MCP pour les rendre faciles à utiliser par le LLM

L'organisation de l'espace de noms est un modèle de conception qui empêche les collisions entre les noms d'outils, regroupe les fonctionnalités associées et facilite l'identification efficace des outils par LLMs. Le modèle établit une catégorisation structurée analogue aux systèmes de stockage organisés plutôt qu'à une accumulation non structurée. Nous recommandons le domain-noun-verbmodèle pour la dénomination des outils. Par exemple, `github_issue_creategithub_issue_list`,`github_issue_update`,`github_pullrequest_creat` ou`github_pullrequest_merge`. L'avantage de ce modèle est évident lorsque l'on examine le comportement du tri alphabétique. Lorsque les outils sont répertoriés par ordre alphabétique, toutes les opérations liées aux problèmes sont regroupées (`create`,,`update`)`list`, suivies des opérations de pull request (`create`,,`list`). `merge` Le nom (type de ressource) fonctionne comme une limite organisationnelle. Cette structure facilite à la fois l'analyse des outils LLM et la navigation dans la documentation humaine, car les fonctionnalités associées se regroupent naturellement.

Le serveur MCP doit être limité au niveau du domaine mais peut être subdivisé en fonction de la séparation des tâches pour les fonctionnalités qu'il fournit. Par exemple, vous pouvez avoir des serveurs MCP distincts pour les opérations d'écriture et les opérations de lecture dans une base de

données. Pour appliquer cette séparation, il est recommandé de mettre en place des garde-fous au niveau de l'agent qui limitent les serveurs MCP accessibles en fonction des intentions et des autorisations de l'utilisateur. Cela peut être réalisé grâce à une combinaison des éléments suivants :

- Chargement conditionnel du serveur : chargez le serveur MCP en lecture seule uniquement lorsque l'agent détecte des opérations de lecture dans l'entrée utilisateur.
- Filtrage basé sur les autorisations : utilisez l'autorisation de l'utilisateur pour n'accorder l'accès qu'aux serveurs MCP appropriés.

Enfin, vous souhaiterez créer une limite supérieure pour le nombre d'outils fournis par un serveur MCP. Ne faites aucune supposition quant à la manière dont les agents utiliseront votre serveur MCP. Ils peuvent naïvement énumérer tous les outils disponibles et les fournir tous au LLM. Si vous avez plus de 50 outils sur un seul serveur, vous devriez envisager de le diviser en plusieurs serveurs.

Meilleures pratiques pour l'organisation des outils MPC

- Utilisez la norme de domain-noun-verb dénomination pour les outils : mettez en œuvre des stratégies pour éviter les collisions de noms à la fois dans les serveurs MCP et dans les agents.
- Définissez une limite supérieure : limitez le nombre d'outils sur un seul serveur MCP.
- Diviser les serveurs MCP : utilisez la séparation des tâches pour diviser les serveurs MCP en groupes logiques.

Stratégie d'hébergement MCP

L'extraction des outils disponibles dans des serveurs MCP dissocie le développement de vos agents des outils disponibles. Cela pose les défis liés à l'endroit où vous hébergez votre serveur MCP et à la manière dont les outils sont organisés au sein de ces serveurs.

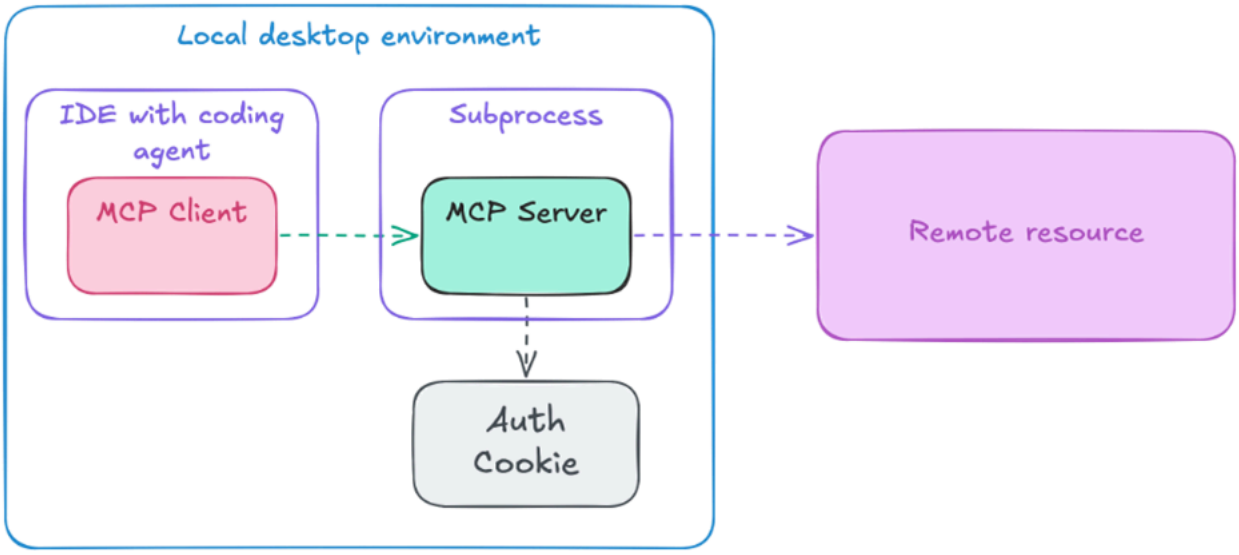
Approches d'hébergement

Il existe trois options pour héberger vos serveurs MCP : les exécuter localement sur la machine d'un utilisateur final, les héberger à distance ou les héberger via une passerelle MCP. Chaque option comporte des avantages et des inconvénients.

Hébergement local

L'hébergement local exécute le serveur MCP en tant que sous-processus sur votre machine locale avec l'agent qui communique avec le serveur en utilisant JSON-RPC sur des flux d'entrée et de sortie standard. Cette approche ne nécessite pas d'authentification entre le client et le serveur. Les outils peuvent interagir avec des applications et des fichiers locaux, utiliser des informations d'identification stockées localement et hériter de l'accès réseau de la machine locale de l'utilisateur. Il s'agit du modèle d'hébergement le plus simple et il présente plusieurs avantages.

De nombreux clients commencent à utiliser MCP en utilisant des serveurs locaux. Ils permettent aux ingénieurs d'itérer et de résoudre rapidement divers problèmes depuis leur environnement local. Prenons l'exemple d'un serveur MCP qui se connecte à un dépôt Git que l'assistant de codage d'un ingénieur utilise. Il est tout à fait logique de conserver le serveur MCP local, car il peut utiliser les informations d'identification uniques de l'ingénieur pour accéder au référentiel, sans ajouter d'appel réseau supplémentaire à un serveur MCP distant. L'image suivante montre un serveur MCP hébergé localement utilisé avec un agent de codage dans un IDE.



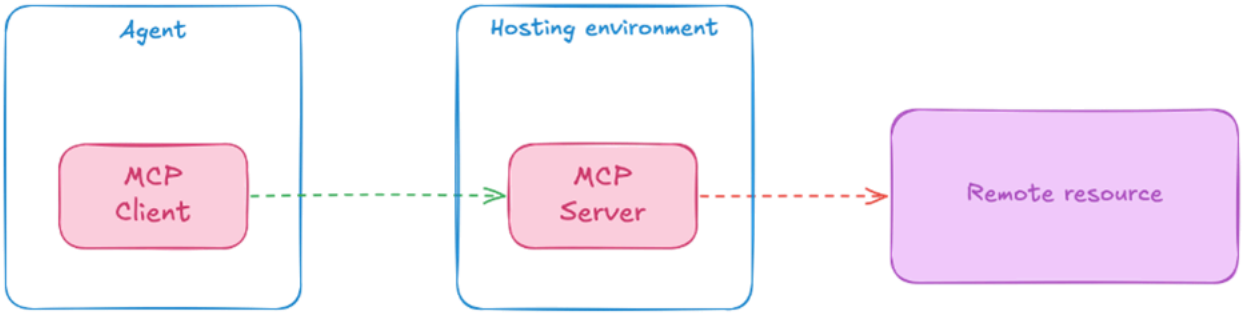
Pour ces types de déploiements, vous devez tenir compte de la manière dont les serveurs MCP sont développés et distribués. La plupart des clients développent un registre MCP dans lequel les serveurs peuvent être enregistrés et téléchargés par les utilisateurs finaux. Cela ressemble beaucoup à un registre de conteneurs dans lequel un utilisateur peut rechercher des fonctionnalités spécifiques et trouver les serveurs MCP adaptés à ses besoins.

Il existe des registres MCP publics, tels que le registre MCP officiel, et des registres hébergés par le secteur privé. Organisations alignent généralement leur stratégie de registre MCP sur les politiques existantes concernant la distribution de logiciels open source, les registres de conteneurs et la gestion interne des packages. Vous devez prendre en compte des facteurs tels que le scan de sécurité, les flux de travail d'approbation et les exigences de conformité.

Cependant, l'hébergement local présente des défis opérationnels que les organisations devraient prendre en compte. Tout d'abord, les utilisateurs finaux doivent découvrir, télécharger et configurer les serveurs MCP de manière indépendante. Cela peut compliquer la prise en main de chaque serveur MCP utilisé localement. Ensuite, vous ne pouvez pas contrôler le cycle de vie du serveur MCP, ce qui signifie que les utilisateurs peuvent continuer à exécuter localement des versions obsolètes présentant des failles de sécurité ou des fonctionnalités manquantes. Cela peut compliquer le respect des exigences de conformité. Certains outils IDEs et les outils CLI, tels que Kiro, permettent aux organisations de gérer et de contrôler les outils MCP disponibles, garantissant ainsi la cohérence et la sécurité entre les équipes.

Hébergement à distance

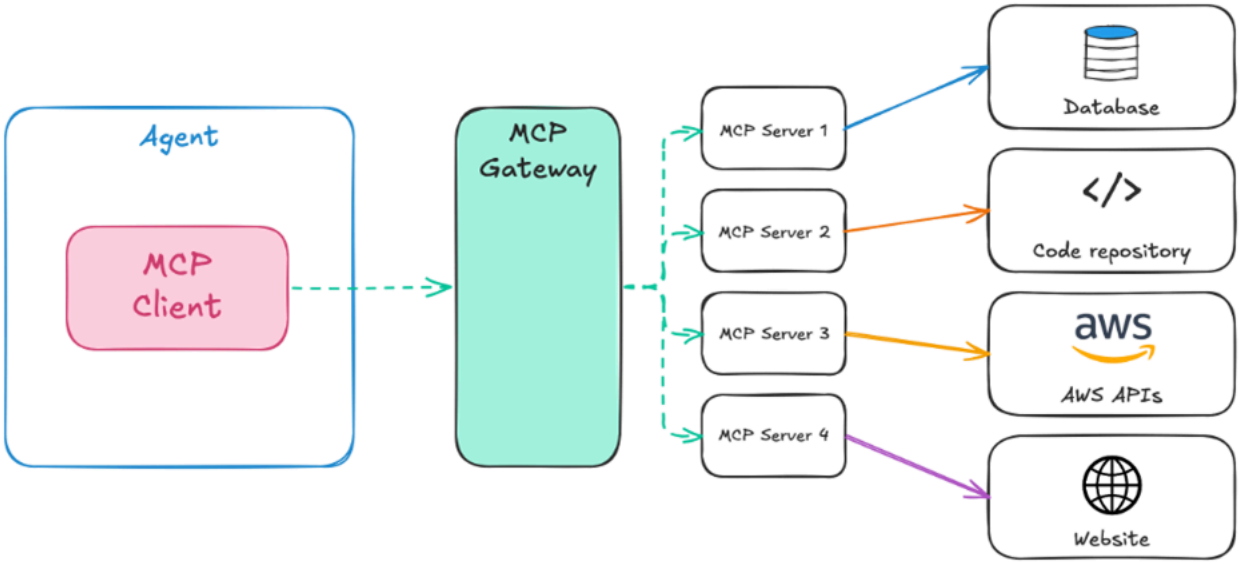
La deuxième option consiste à héberger des serveurs MCP distants accessibles via HTTP ou HTTPS. Cela permet d'accéder à n'importe quel client connecté au réseau. L'hébergement à distance vous permet de contrôler de manière centralisée l'accès aux ressources et aux fonctionnalités du MCP, de mettre en œuvre l'authentification et l'autorisation, et de contrôler le versionnement et les mises à jour de la logique du serveur MCP. L'hébergement à distance nécessite toujours l'utilisation d'un registre MCP afin que les utilisateurs finaux puissent découvrir les serveurs MCP qu'ils souhaitent utiliser avec leur agent. L'image suivante montre l'approche d'hébergement à distance.



Du point de vue du développement des agents, l'expérience est similaire, que le serveur MCP soit local ou distant. Le changement le plus important concerne la mise en œuvre de l'authentification et de l'autorisation, y compris l'accès de l'agent au serveur MCP et l'accès du serveur aux ressources externes. Les implémentations de serveurs MCP distants doivent être soigneusement planifiées afin de prendre en compte l'accès mutualisé et la gestion des privilèges. Le chapitre sur la stratégie de gouvernance du MCP contient plus d'informations sur les considérations relatives à l'authentification et à l'autorisation.

Passerelle MCP

La dernière option consiste à utiliser une passerelle MCP. Les passerelles MCP agissent comme un proxy centralisé entre les clients et les serveurs MCP, et elles orchestrent l'accès aux serveurs MCP enregistrés. Sans passerelle, chaque agent doit enregistrer chaque serveur MCP distant qu'il souhaite utiliser. Une passerelle permet à l'agent de se connecter à un point de terminaison unique qui gère l'authentification, l'autorisation, le routage et la traduction du protocole. De nouveaux serveurs et outils MCP peuvent être ajoutés dynamiquement et mis immédiatement à la disposition de l'agent. L'image suivante montre l'approche de la passerelle MCP.



Certaines solutions de passerelle, telles que Docker MCP Gateway, gèrent également le cycle de vie des serveurs MCP, en lançant des serveurs à la demande selon les besoins. Les passerelles MCP, telles qu'Amazon Bedrock AgentCore Gateway, peuvent également aider à gérer la découverte d'outils en fournissant des fonctionnalités de recherche sémantique natives. Cela fournit aux agents un point de terminaison unique pour se connecter à un client MCP et permet d'optimiser l'utilisation de leurs fenêtres contextuelles. Il en résulte des agents simples capables de choisir et d'utiliser efficacement les outils MCP. Cependant, elle présente des défis liés à l'identité similaires à ceux de l'approche du serveur MCP distant.

Bonnes pratiques pour l'hébergement de serveurs MCP

- L'éventail des options d'hébergement n'est pas universel. Une grande partie de l'utilisation des serveurs MCP est aujourd'hui locale.
- Lorsque vous commencez à utiliser des serveurs MCP distants, votre principale préoccupation est l'authentification et l'autorisation cohérentes auprès du serveur MCP et la manière dont le serveur MCP effectue l'authentification et l'autorisation des ressources en aval.
- Les passerelles MCP simplifient la connectivité, l'authentification et l'autorisation pour l'hébergement de plusieurs serveurs MCP distants. Ils fournissent également des fonctionnalités permettant d'améliorer la gestion des fenêtres contextuelles en recherchant les outils applicables.

Stratégie de gouvernance de MCP

L'autre fonctionnalité essentielle que MCP offre aux entreprises est la prise en charge de la gouvernance centralisée. Votre stratégie de gouvernance MCP doit prendre en compte l'authentification et l'autorisation des serveurs MCP ainsi que des ressources auxquelles ils accèdent. Il devrait également aborder la limitation du débit pour protéger les ressources en aval, les mesures opérationnelles pour surveiller l'utilisation et les performances des outils, et la gestion des déploiements et de la distribution des serveurs MCP.

Authentification et autorisation

L'un des aspects les plus importants de votre stratégie d'authentification et d'autorisation consiste à gérer l'accès aux ressources en aval depuis les serveurs MCP. Lorsqu'un utilisateur appelle un agent, une authentification et une autorisation sont effectuées pour garantir que l'utilisateur est autorisé à appeler l'agent. L'agent orchestre ensuite l'appel d'outils spécifiques sur les serveurs MCP. Vous devez décider comment autoriser l'accès pour chaque outil.

L'une des options est machine-to-machine l'autorisation, où le consentement ou l'interaction de l'utilisateur ne sont pas requis. Par exemple, un appel d'agent basé sur le temps utilise un serveur MCP pour collecter les journaux d'une application et les analyser. Dans ce scénario, l'agent est préautorisé à accéder aux données spécifiées. La deuxième option est l'accès délégué par l'utilisateur, dans le cadre duquel un utilisateur donne son accord pour accéder à des données et à des ressources spécifiques à l'utilisateur.

Le tableau suivant présente les modèles d'authentification et d'autorisation.

Facteur	Accès délégué par l'utilisateur	Machine-to-machine
Propriété des données	Autorisation spécifique à l'utilisateur d'accéder aux données	Données à l'échelle du système ou de l'organisation
Interaction avec l'utilisateur	L'utilisateur est présent et peut consentir	Aucune interaction avec l'utilisateur
Chronologie de l'opération	Interactif ou en temps réel	En arrière-plan, planifié ou par lots

Étendue de l'autorisation	Les autorisations varient en fonction de l'utilisateur	Autorisations cohérentes au niveau de l'agent

L'accès délégué par les utilisateurs nécessite une mise en œuvre minutieuse et doit être développé avec votre équipe de sécurité. Les agents doivent être en mesure d'évaluer quels outils un LLM a sélectionnés et s'ils nécessitent une autorisation supplémentaire. Les outils MCP doivent inclure des descriptions indiquant leurs exigences en matière d'authentification et d'autorisation et indiquant où récupérer les jetons d'accès. Les applications clientes doivent prendre en charge les demandes d'authentification intermédiaires, et le client MCP doit fournir les informations d'identification récupérées à l'agent pour chaque appel d'outil.

Vous devez vous assurer que les outils MCP disposent toujours de leurs propres jetons pour accéder aux fonctionnalités externes et que l'accès est enregistré et audité. Les informations d'identification de l'utilisateur ne doivent pas être propagées par le biais de votre système agentic. Par exemple, vos serveurs MCP ne doivent pas utiliser le même jeton pour accéder aux données que celui utilisé pour appeler l'agent. Les appels en aval doivent utiliser des jetons explicitement définis et générés à des fins spécifiques. Cela permet de fournir des garde-fous supplémentaires pour empêcher l'accès involontaire aux données pour le compte d'actions. Cela peut également aider à empêcher les hallucinations de produire des résultats imprévus. Imaginez qu'un utilisateur disposant de droits d'administrateur complets demande à un agent de cloner une base de données de production pour une utilisation en pré-production. Pour ce faire, l'utilisateur n'a besoin que `READ` d'`CREATE`autorisations. Supposons que le LLM hallucine et pense qu'il doit nettoyer l'ancienne base de données dans le cadre de cette demande. S'il réutilise les informations d'identification de l'utilisateur, il est probable qu'il réussisse, car les informations d'identification d'origine de l'utilisateur disposent d'`DELETE`autorisations. Au lieu de cela, si le serveur MCP utilise un jeton délimité intentionnellement pour la demande avec des `CREATE` autorisations justes `READ` et limitées, la tentative de suppression de la base de données de production échouera.

Vous pouvez utiliser Amazon Bedrock AgentCore Identity pour implémenter ces modèles. Assurez-vous de choisir intentionnellement si les autorisations permettant de répertorier et d'invoquer des outils hébergés par un serveur MCP impliquent l'autorisation d'accéder aux fonctionnalités externes exposées par le serveur MCP. Ce flux d'identité du serveur MCP vers la ressource et de retour vers l'utilisateur dépend du type de service d'authentification et d'autorisation utilisé. Vous devez décider de la manière dont cela est géré à grande échelle pour vos serveurs MCP.

Lorsque vous concevez vos modèles d'authentification et d'autorisation, mettez en œuvre des mécanismes d'isolation des jetons qui récupèrent différents jetons d'accès pour chaque outil consulté.

Ne réutilisez pas les jetons entre les outils et les serveurs. AgentCore L'identité fournit cette capacité d'isolation des jetons. Il gère automatiquement à la fois les jetons de charge de travail (pour machine-to-machine l'authentification) et les jetons utilisateur (pour l'accès délégué par l'utilisateur) afin de garantir une séparation appropriée et d'empêcher l'augmentation des autorisations. Cela est particulièrement important lors de l'intégration de serveurs MCP distants ou de passerelles MCP.

Bonnes pratiques pour l'authentification et l'autorisation MCP

- Séparation des jetons — Ne transmettez pas les jetons porteurs des appelants aux services en aval. Validez que le champ aud (audience) correspond au serveur qui reçoit le jeton. La déclaration d'audience précise à quel service le jeton est destiné, empêchant ainsi toute réutilisation non autorisée du jeton sur différents serveurs MCP.
- Sélectionnez une approche d'accès : choisissez entre machine-to-machine un accès délégué par l'utilisateur pour chaque outil fourni par vos serveurs MCP. Envisagez de regrouper les outils sur le même serveur MCP qui utilisent le même modèle d'authentification.

Contrôle de la charge

Comme pour tout système distribué, vous devez réfléchir à la manière de contrôler la charge de votre parc de serveurs MCP. Tout d'abord, vous devez déterminer s'il convient d'implémenter la limitation de débit dans vos serveurs MCP et où implémenter les limites. Si vous choisissez de ne pas implémenter de limitation de débit, vous répercutez toute limitation de débit effectuée par les ressources en aval. De nombreux systèmes choisissent de limiter le débit en fonction des attributs de la demande, tels que l'identifiant d'utilisateur ou de compte. Vérifiez que les demandes envoyées aux services en aval comportent les mêmes attributs afin que plusieurs utilisateurs ne soient pas affectés par la charge générée par un autre utilisateur.

Si vous choisissez d'implémenter la limitation de débit, l'approche recommandée consiste à implémenter la limitation de débit principale au niveau du serveur MCP, les services principaux fournissant une protection secondaire et les agents adaptant leur comportement en fonction des commentaires sur les limites de débit. Déterminez si les limites de débit sont par serveur MCP ou par outil. Les limites de débit par serveur MCP aident à protéger votre parc de serveurs MCP et vos services dans un environnement mutualisé. Cependant, cela peut être très grossier. Les limites de débit par outil sont conçues pour éviter de surcharger les ressources en aval qui pourraient ne pas se limiter suffisamment au débit. Si un outil en appelle plusieurs APIs, vous devez définir la limite de débit de manière à ce qu'elle corresponde au débit le plus bas autorisé par ceux-ci APIs.

Les informations sur les limites de débit transmises dans les en-têtes HTTP peuvent également constituer une mesure utile pour les utilisateurs et les systèmes automatisés afin de les aider à gérer leur propre taux de demandes et leur stratégie de nouvelles tentatives. Par exemple, vous pouvez renvoyer ces en-têtes à l'agent depuis votre serveur MCP, comme illustré dans l'exemple suivant :

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

En outre, envisagez le délestage pour protéger l'ensemble du service lorsqu'aucun client ne dépasse une limite de débit mais que la charge a un impact sur les performances du système.

Meilleures pratiques pour contrôler la charge

- Choisissez une approche de limitation du débit : prévoyez de limiter le débit des utilisateurs individuels en fonction de leur utilisation des ressources en aval ou de leur utilisation de votre serveur et de vos outils MCP.
- Envisagez le délestage : protégez votre parc de serveurs MCP contre les surcharges générales qui ne sont pas causées par un seul ou une poignée de clients.

Métriques opérationnelles

Les indicateurs clés à saisir pour les implémentations de MCP doivent se concentrer sur l'expérience client qu'ils offrent. Ces indicateurs incluent généralement l'utilisation des jetons, la précision de la sélection des outils, le nombre d'outils enregistrés auprès de l'agent et la latence des outils. Par exemple, la surveillance des jetons de sortie renvoyés par chaque outil vous permet de définir des alarmes lorsque les outils dépassent un seuil d'utilisation de la fenêtre contextuelle. Lorsqu'un outil dépasse ce seuil, vous souhaiterez peut-être revoir son comportement. Cela est également lié à la stratégie de conception des outils MCP. Les indicateurs de précision de la sélection des outils indiquent dans quelle mesure les agents choisissent les outils appropriés pour des tâches données, tandis que la vitesse d'exécution et les taux de réussite mettent en évidence les problèmes de performance et de fiabilité.

Par exemple, pour évaluer les mesures de précision relatives à la sélection et à l'utilisation des outils, les AWS équipes ont créé des ensembles de données exceptionnels pour les tests de régression. Les ensembles de données ont été générés de manière synthétique en utilisant les journaux d'invocation historiques LLMs des API lors des requêtes des utilisateurs. À l'aide des indicateurs

prédéfinis de sélection et d'utilisation des outils (tels que la précision de sélection des outils, la précision des paramètres de l'outil et la précision des appels de fonctions multitours), les AWS équipes ont pu évaluer objectivement la capacité de l'agent IA à identifier correctement les outils appropriés, à renseigner ses paramètres avec des valeurs précises et à maintenir des séquences d'invocation d'outils cohérentes tout au long des virages de conversation.

La mesure du nombre d'outils enregistrés auprès d'un agent peut vous aider à identifier les problèmes potentiels liés à la gestion des fenêtres contextuelles ainsi que les modifications des outils disponibles présentés par les serveurs MCP. Vous devez régulièrement consulter les indicateurs opérationnels qui indiquent l'expérience utilisateur avec votre serveur et vos outils MCP.

Collaborateurs

Conception

- Alex Torres, architecte de solutions senior, AWS
- Saikat Gomes, responsable senior des solutions clients, AWS
- Mike Haken, architecte principal des solutions senior, AWS
- Sreeja Das, ingénieure principale, AWS

Révision

- Ted Swinyar, responsable de l'architecture de solutions, AWS
- Raju Patil, data scientist senior, AWS

Rédaction technique

- Lilly AbouHarb, rédactrice technique senior, AWS

Historique du document

Le tableau suivant décrit les modifications importantes apportées à ce guide. Pour être averti des mises à jour à venir, abonnez-vous à un fil RSS.

Modification	Description	Date
Publication initiale	—	16 mars 2026

AWS Glossaire des directives prescriptives

Les termes suivants sont couramment utilisés dans les stratégies, les guides et les modèles fournis par les directives AWS prescriptives. Pour suggérer des entrées, veuillez utiliser le lien Faire un commentaire à la fin du glossaire.

Nombres

7 R

Sept politiques de migration courantes pour transférer des applications vers le cloud. Ces politiques s'appuient sur les 5 R identifiés par Gartner en 2011 et sont les suivantes :

- Refactor/re-architect — Déplacez une application et modifiez son architecture en tirant pleinement parti des fonctionnalités natives du cloud pour améliorer l'agilité, les performances et l'évolutivité. Cela implique généralement le transfert du système d'exploitation et de la base de données. Exemple : migrez votre base de données Oracle sur site vers l' PostgreSQL-Compatible édition Amazon Aurora.
- Replateformer (déplacer et remodeler) : transférez une application vers le cloud et introduisez un certain niveau d'optimisation pour tirer parti des fonctionnalités du cloud. Exemple : migrez votre base de données Oracle sur site vers Amazon Relational Database Service (Amazon RDS) pour Oracle dans le. AWS Cloud
- Racheter (rachat) : optez pour un autre produit, généralement en passant d'une licence traditionnelle à un modèle SaaS. Exemple : migrez votre système de gestion de la relation client (CRM) vers Salesforce.com.
- Réhéberger (lift and shift) : transférez une application vers le cloud sans apporter de modifications pour tirer parti des fonctionnalités du cloud. Exemple : migrez votre base de données Oracle sur site vers Oracle sur une instance EC2 dans le. AWS Cloud
- Relocaliser (lift and shift au niveau de l'hyperviseur) : transférez l'infrastructure vers le cloud sans acheter de nouveau matériel, réécrire des applications ou modifier vos opérations existantes. Vous migrez des serveurs d'une plateforme sur site vers un service cloud pour la même plateforme. Exemple : migrer une Microsoft Hyper-V application vers AWS.
- Retenir : conservez les applications dans votre environnement source. Il peut s'agir d'applications nécessitant une refactorisation majeure, que vous souhaitez retarder, et d'applications existantes que vous souhaitez retenir, car rien ne justifie leur migration sur le plan commercial.

- Retirer : mettez hors service ou supprimez les applications dont vous n'avez plus besoin dans votre environnement source.

A

A2 (1) Agent-to-Agent

Protocole dynamique pour la collaboration agent-agent prenant en charge la délégation de tâches et le transfert d'état.

ABAC

Voir contrôle d'accès basé sur les attributs.

services abstraits

Consultez la section Services gérés.

ACIDE

Voir atomicité, consistance, isolation, durabilité.

migration active-active

Méthode de migration de base de données dans laquelle la synchronisation des bases de données source et cible est maintenue (à l'aide d'un outil de réplication bidirectionnelle ou d'opérations d'écriture double), tandis que les deux bases de données gèrent les transactions provenant de la connexion d'applications pendant la migration. Cette méthode prend en charge la migration par petits lots contrôlés au lieu d'exiger un basculement ponctuel. Elle est plus flexible mais demande plus de travail qu'une migration active-passive.

migration active-passive

Méthode de migration de base de données dans laquelle les bases de données source et cible sont synchronisées, mais seule la base de données source gère les transactions liées à la connexion des applications pendant que les données sont répliquées vers la base de données cible. La base de données cible n'accepte aucune transaction pendant la migration.

Agent

Un système d'IA capable de raisonner, de planifier et de prendre des mesures de manière autonome à l'aide d'outils pour atteindre des objectifs.

Agent Ops

Pratiques opérationnelles pour la création, le test, le déploiement et l'exécution d'agents d'IA en production à grande échelle.

fonction d'agrégation

Fonction SQL qui agit sur un groupe de lignes et calcule une valeur de retour unique pour le groupe. Des exemples de fonctions d'agrégation incluent `SUM` et`MAX`.

AI

Voir intelligence artificielle.

AIOps

Voir les opérations d'intelligence artificielle.

anonymisation

Processus de suppression définitive d'informations personnelles dans un ensemble de données. L'anonymisation peut contribuer à protéger la vie privée. Les données anonymisées ne sont plus considérées comme des données personnelles.

anti-motif

Solution fréquemment utilisée pour un problème récurrent lorsque la solution est contre-productive, inefficace ou moins efficace qu'une solution alternative.

contrôle des applications

Une approche de sécurité qui permet d'utiliser uniquement des applications approuvées afin de protéger un système contre les logiciels malveillants.

portefeuille d'applications

Ensemble d'informations détaillées sur chaque application utilisée par une organisation, y compris le coût de génération et de maintenance de l'application, ainsi que sa valeur métier. Ces informations sont essentielles pour le processus de découverte et d'analyse du portefeuille et permettent d'identifier et de prioriser les applications à migrer, à moderniser et à optimiser.

intelligence artificielle (IA)

Domaine de l'informatique consacré à l'utilisation des technologies de calcul pour exécuter des fonctions cognitives généralement associées aux humains, telles que l'apprentissage, la résolution de problèmes et la reconnaissance de modèles. Pour plus d'informations, veuillez consulter Qu'est-ce que l'intelligence artificielle ?

opérations d'intelligence artificielle (AIOps)

Processus consistant à utiliser des techniques de machine learning pour résoudre les problèmes opérationnels, réduire les incidents opérationnels et les interventions humaines, mais aussi améliorer la qualité du service. Pour plus d'informations sur la façon dont les AIOps sont utilisées dans la stratégie de migration AWS , veuillez consulter le guide d'intégration des opérations.

chiffrement asymétrique

Algorithme de chiffrement qui utilise une paire de clés, une clé publique pour le chiffrement et une clé privée pour le déchiffrement. Vous pouvez partager la clé publique, car elle n'est pas utilisée pour le déchiffrement, mais l'accès à la clé privée doit être très restreint.

atomicité, cohérence, isolement, durabilité (ACID)

Ensemble de propriétés logicielles garantissant la validité des données et la fiabilité opérationnelle d'une base de données, même en cas d'erreur, de panne de courant ou d'autres problèmes.

contrôle d'accès par attributs (ABAC)

Pratique qui consiste à créer des autorisations détaillées en fonction des attributs de l'utilisateur, tels que le service, le poste et le nom de l'équipe. Pour plus d'informations, consultez ABAC pour AWS dans la documentation Gestion des identités et des accès AWS (IAM).

source de données faisant autorité

Emplacement où vous stockez la version principale des données, considérée comme la source d'information la plus fiable. Vous pouvez copier les données de la source de données officielle vers d'autres emplacements à des fins de traitement ou de modification des données, par exemple en les anonymisant, en les expurgant ou en les pseudonymisant.

Zone de disponibilité

Un emplacement distinct au sein d'un Région AWS réseau isolé des défaillances dans d'autres zones de disponibilité et fournissant une connectivité réseau peu coûteuse et à faible latence aux autres zones de disponibilité de la même région.

AWS Cadre d'adoption du cloud (AWS CAF)

Un cadre de directives et de meilleures pratiques visant AWS à aider les entreprises à élaborer un plan efficace pour réussir leur migration vers le cloud. AWS La CAF organise ses conseils en six domaines prioritaires appelés perspectives : les affaires, les personnes, la gouvernance, les plateformes, la sécurité et les opérations. Les perspectives d'entreprise, de personnes et de gouvernance mettent l'accent sur les compétences et les processus métier, tandis que les

perspectives relatives à la plateforme, à la sécurité et aux opérations se concentrent sur les compétences et les processus techniques. Par exemple, la perspective liée aux personnes cible les parties prenantes qui s'occupent des ressources humaines (RH), des fonctions de dotation en personnel et de la gestion des personnes. Dans cette perspective, la AWS CAF fournit des conseils pour le développement du personnel, la formation et les communications afin de préparer l'organisation à une adoption réussie du cloud. Pour plus d'informations, veuillez consulter le site Web AWS CAF et le livre blanc AWS CAF.

AWS Cadre de qualification de la charge de travail (AWS WQF)

Outil qui évalue les charges de travail liées à la migration des bases de données, recommande des stratégies de migration et fournit des estimations de travail. AWS Le WQF est inclus avec AWS Schema Conversion Tool (AWS SCT). Il analyse les schémas de base de données et les objets de code, le code d'application, les dépendances et les caractéristiques de performance, et fournit des rapports d'évaluation.

B

mauvais bot

Un bot destiné à perturber ou à nuire à des individus ou à des organisations.

BCP

Consultez la section Planification de la continuité des activités.

graphique de comportement

Vue unifiée et interactive des comportements des ressources et des interactions au fil du temps. Vous pouvez utiliser un graphique de comportement avec Amazon Detective pour examiner les tentatives de connexion infructueuses, les appels d'API suspects et les actions similaires. Pour plus d'informations, veuillez consulter Data in a behavior graph dans la documentation Detective.

système de poids fort

Système qui stocke d'abord l'octet le plus significatif. Voir aussi endianité.

classification binaire

Processus qui prédit un résultat binaire (l'une des deux classes possibles). Par exemple, votre modèle de machine learning peut avoir besoin de prévoir des problèmes tels que « Cet e-mail est-il du spam ou non ? » ou « Ce produit est-il un livre ou une voiture ? ».

filtre de Bloom

Structure de données probabiliste et efficace en termes de mémoire qui est utilisée pour tester si un élément fait partie d'un ensemble.

blue/green déploiement

Stratégie de déploiement dans laquelle vous créez deux environnements distincts mais identiques. Vous exécutez la version actuelle de l'application dans un environnement (bleu) et la nouvelle version de l'application dans l'autre environnement (vert). Cette stratégie vous permet de revenir rapidement en arrière avec un impact minimal.

bot

Application logicielle qui exécute des tâches automatisées sur Internet et simule l'activité ou l'interaction humaine. Certains robots sont utiles ou bénéfiques, comme les robots d'exploration Web qui indexent des informations sur Internet. D'autres robots, appelés « bots malveillants », sont destinés à perturber ou à nuire à des individus ou à des organisations.

botnet

Réseaux de robots infectés par des logiciels malveillants et contrôlés par une seule entité, connue sous le nom d'herder ou d'opérateur de bots. Les botnets sont le mécanisme le plus connu pour faire évoluer les bots et leur impact.

branche

Zone contenue d'un référentiel de code. La première branche créée dans un référentiel est la branche principale. Vous pouvez créer une branche à partir d'une branche existante, puis développer des fonctionnalités ou corriger des bogues dans la nouvelle branche. Une branche que vous créez pour générer une fonctionnalité est communément appelée branche de fonctionnalités. Lorsque la fonctionnalité est prête à être publiée, vous fusionnez à nouveau la branche de fonctionnalités dans la branche principale. Pour plus d'informations, consultez À propos des branches (GitHub documentation).

accès par brise-vitre

Dans des circonstances exceptionnelles et par le biais d'un processus approuvé, c'est un moyen rapide pour un utilisateur d'accéder à un accès auquel Compte AWS il n'est généralement pas autorisé. Pour plus d'informations, consultez l'indicateur Mettre en œuvre des procédures permettant de briser le verre dans le AWS Well-Architected guide.

stratégie existante (brownfield)

L'infrastructure existante de votre environnement. Lorsque vous adoptez une stratégie existante pour une architecture système, vous concevez l'architecture en fonction des contraintes des systèmes et de l'infrastructure actuels. Si vous étendez l'infrastructure existante, vous pouvez combiner des politiques brownfield (existantes) et greenfield (inédites).

cache de tampon

Zone de mémoire dans laquelle sont stockées les données les plus fréquemment consultées.

capacité métier

Ce que fait une entreprise pour générer de la valeur (par exemple, les ventes, le service client ou le marketing). Les architectures de microservices et les décisions de développement peuvent être dictées par les capacités métier. Pour plus d'informations, veuillez consulter la section Organisation en fonction des capacités métier du livre blanc Exécution de microservices conteneurisés sur AWS.

planification de la continuité des activités (BCP)

Plan qui tient compte de l'impact potentiel d'un événement perturbateur, tel qu'une migration à grande échelle, sur les opérations, et qui permet à une entreprise de reprendre ses activités rapidement.

C

CAF

Voir le cadre d'adoption du AWS cloud.

déploiement de Canary

Diffusion lente et progressive d'une version pour les utilisateurs finaux. Lorsque vous êtes sûr, vous déployez la nouvelle version et remplacez la version actuelle dans son intégralité.

CCoE

Voir le Centre d'excellence du cloud.

CDC

Consultez la section Capture des données de modification.

capture des données de modification (CDC)

Processus de suivi des modifications apportées à une source de données, telle qu'une table de base de données, et d'enregistrement des métadonnées relatives à ces modifications. Vous pouvez utiliser la CDC à diverses fins, telles que l'audit ou la réplication des modifications dans un système cible afin de maintenir la synchronisation.

ingénierie du chaos

Introduire intentionnellement des défaillances ou des événements perturbateurs pour tester la résilience d'un système. Vous pouvez utiliser AWS Fault Injection Service (AWS FIS) pour effectuer des expériences qui stressent vos AWS charges de travail et évaluer leur réponse.

CI/CD

Découvrez l'intégration continue et la livraison continue.

classification

Processus de catégorisation qui permet de générer des prédictions. Les modèles de ML pour les problèmes de classification prédisent une valeur discrète. Les valeurs discrètes se distinguent toujours les unes des autres. Par exemple, un modèle peut avoir besoin d'évaluer la présence ou non d'une voiture sur une image.

Développeur citoyen

Un utilisateur professionnel qui crée des applications d'intelligence artificielle à l'aide de plateformes sans code/low code sans compétences techniques spécialisées.

chiffrement côté client

Chiffrement des données localement, avant que la cible ne les Service AWS reçoive.

Centre d'excellence cloud (CCoE)

Une équipe multidisciplinaire qui dirige les efforts d'adoption du cloud au sein d'une organisation, notamment en développant les bonnes pratiques en matière de cloud, en mobilisant des ressources, en établissant des délais de migration et en guidant l'organisation dans le cadre de transformations à grande échelle. Pour plus d'informations, consultez les articles du CCoE sur le blog de stratégie AWS Cloud d'entreprise.

cloud computing

Technologie cloud généralement utilisée pour le stockage de données à distance et la gestion des appareils IoT. Le cloud computing est généralement associé à la technologie informatique de pointe.

modèle d'exploitation du cloud

Dans une organisation informatique, modèle d'exploitation utilisé pour créer, faire évoluer et optimiser un ou plusieurs environnements cloud. Pour plus d'informations, consultez la section Création de votre modèle d'exploitation cloud.

étapes d'adoption du cloud

Les quatre phases que les entreprises traversent généralement lorsqu'elles migrent vers AWS Cloud :

- Projet : exécution de quelques projets liés au cloud à des fins de preuve de concept et d'apprentissage
- Base : réaliser des investissements fondamentaux pour mettre à l'échelle l'adoption du cloud (par exemple, en créant une zone de destination, en définissant un CCoE ou en établissant un modèle opérationnel)
- Migration : migration d'applications individuelles
- Re-invention — Optimisation des produits et services et innovation dans le cloud

Ces étapes ont été définies par Stephen Orban dans le billet de blog The Journey Toward Cloud-First & the Stages of Adoption publié sur le blog AWS Cloud Enterprise Strategy. Pour plus d'informations sur leur lien avec la stratégie de AWS migration, consultez le guide de préparation à la migration.

CMDB

Consultez la base de données de gestion des configurations.

référentiel de code

Emplacement où le code source et d'autres ressources, comme la documentation, les exemples et les scripts, sont stockés et mis à jour par le biais de processus de contrôle de version. Les référentiels cloud courants incluent GitHub ouBitbucket Cloud. Chaque version du code est appelée branche. Dans une structure de microservice, chaque référentiel est consacré à une seule fonctionnalité. Un CI/CD pipeline unique peut utiliser plusieurs référentiels.

cache passif

Cache tampon vide, mal rempli ou contenant des données obsolètes ou non pertinentes. Cela affecte les performances, car l'instance de base de données doit lire à partir de la mémoire principale ou du disque, ce qui est plus lent que la lecture à partir du cache tampon.

données gelées

Données rarement consultées et généralement historiques. Lorsque vous interrogez ce type de données, les requêtes lentes sont généralement acceptables. Le transfert de ces données vers des niveaux ou classes de stockage moins performants et moins coûteux peut réduire les coûts.

vision par ordinateur (CV)

Domaine de l'IA qui utilise l'apprentissage automatique pour analyser et extraire des informations à partir de formats visuels tels que des images numériques et des vidéos. Par exemple, Amazon SageMaker AI fournit des algorithmes de traitement d'image pour les CV.

dérive de configuration

Pour une charge de travail, une modification de configuration par rapport à l'état attendu. Cela peut entraîner une non-conformité de la charge de travail, et cela est généralement progressif et involontaire.

base de données de gestion des configurations (CMDB)

Référentiel qui stocke et gère les informations relatives à une base de données et à son environnement informatique, y compris les composants matériels et logiciels ainsi que leurs configurations. Vous utilisez généralement les données d'une CMDB lors de la phase de découverte et d'analyse du portefeuille de la migration.

pack de conformité

Ensemble de AWS Config règles et d'actions correctives que vous pouvez assembler pour personnaliser vos contrôles de conformité et de sécurité. Vous pouvez déployer un pack de conformité en tant qu'entité unique dans une région Compte AWS et, ou au sein d'une organisation, à l'aide d'un modèle YAML. Pour plus d'informations, consultez la section Packs de conformité dans la AWS Config documentation.

intégration continue et livraison continue (CI/CD)

Processus d'automatisation des étapes de source, de construction, de test, de préparation et de production du processus de publication du logiciel. CI/CD est communément décrit comme un pipeline. CI/CD peut vous aider à automatiser les processus, à améliorer la productivité, à améliorer la qualité du code et à accélérer les livraisons. Pour plus d'informations, veuillez consulter Avantages de la livraison continue. CD peut également signifier déploiement continu. Pour plus d'informations, veuillez consulter Livraison continue et déploiement continu.

CV

Voir vision par ordinateur.

D

données au repos

Données stationnaires dans votre réseau, telles que les données stockées.

classification des données

Processus permettant d'identifier et de catégoriser les données de votre réseau en fonction de leur sévérité et de leur sensibilité. Il s'agit d'un élément essentiel de toute stratégie de gestion des risques de cybersécurité, car il vous aide à déterminer les contrôles de protection et de conservation appropriés pour les données. La classification des données est une composante du pilier de sécurité du AWS Well-Architected cadre. Pour plus d'informations, veuillez consulter Classification des données.

dérive des données

Une variation significative entre les données de production et les données utilisées pour entraîner un modèle ML, ou une modification significative des données d'entrée au fil du temps. La dérive des données peut réduire la qualité, la précision et l'équité globales des prédictions des modèles ML.

données en transit

Données qui circulent activement sur votre réseau, par exemple entre les ressources du réseau.

maillage de données

Un cadre architectural qui fournit une propriété des données distribuée et décentralisée avec une gestion et une gouvernance centralisées.

minimisation des données

Le principe de collecte et de traitement des seules données strictement nécessaires. La pratique de la minimisation des données AWS Cloud peut réduire les risques liés à la confidentialité, les coûts et l'empreinte carbone de vos analyses.

périmètre de données

Ensemble de garde-fous préventifs dans votre AWS environnement qui permettent de garantir que seules les identités fiables accèdent aux ressources fiables des réseaux attendus. Pour plus d'informations, voir Création d'un périmètre de données sur AWS.

prétraitement des données

Pour transformer les données brutes en un format facile à analyser par votre modèle de ML. Le prétraitement des données peut impliquer la suppression de certaines colonnes ou lignes et le traitement des valeurs manquantes, incohérentes ou en double.

provenance des données

Le processus de suivi de l'origine et de l'historique des données tout au long de leur cycle de vie, par exemple la manière dont les données ont été générées, transmises et stockées.

sujet des données

Personne dont les données sont collectées et traitées.

entrepôt des données

Un système de gestion des données qui prend en charge les informations commerciales, telles que les analyses. Les entrepôts de données contiennent généralement de grandes quantités de données historiques et sont généralement utilisés pour les requêtes et les analyses.

langage de définition de base de données (DDL)

Instructions ou commandes permettant de créer ou de modifier la structure des tables et des objets dans une base de données.

langage de manipulation de base de données (DML)

Instructions ou commandes permettant de modifier (insérer, mettre à jour et supprimer) des informations dans une base de données.

DDL

Voir langage de définition de base de données.

ensemble profond

Sert à combiner plusieurs modèles de deep learning à des fins de prédiction. Vous pouvez utiliser des ensembles profonds pour obtenir une prévision plus précise ou pour estimer l'incertitude des prédictions.

deep learning

Un sous-champ de ML qui utilise plusieurs couches de réseaux neuronaux artificiels pour identifier le mappage entre les données d'entrée et les variables cibles d'intérêt.

défense en profondeur

Approche de la sécurité de l'information dans laquelle une série de mécanismes et de contrôles de sécurité sont judicieusement répartis sur l'ensemble d'un réseau informatique afin de protéger la confidentialité, l'intégrité et la disponibilité du réseau et des données qu'il contient. Lorsque vous adoptez cette stratégie AWS, vous ajoutez plusieurs contrôles à différentes couches de la AWS Organizations structure afin de sécuriser les ressources. Par exemple, une approche de défense approfondie peut combiner l'authentification multifactorielle, la segmentation du réseau et le chiffrement.

administrateur délégué

Dans AWS Organizations, un service compatible peut enregistrer un compte AWS membre pour administrer les comptes de l'organisation et gérer les autorisations pour ce service. Ce compte est appelé administrateur délégué pour ce service. Pour plus d'informations et une liste des services compatibles, veuillez consulter la rubrique Services qui fonctionnent avec AWS Organizations dans la documentation AWS Organizations .

déploiement

Processus de mise à disposition d'une application, de nouvelles fonctionnalités ou de corrections de code dans l'environnement cible. Le déploiement implique la mise en œuvre de modifications dans une base de code, puis la génération et l'exécution de cette base de code dans les environnements de l'application.

environnement de développement

Voir environnement.

contrôle de détection

Contrôle de sécurité conçu pour détecter, journaliser et alerter après la survenue d'un événement. Ces contrôles constituent une deuxième ligne de défense et vous alertent en cas d'événements de sécurité qui ont contourné les contrôles préventifs en place. Pour plus d'informations, veuillez consulter la rubrique Contrôles de détection dans Implementing security controls on AWS.

cartographie de la chaîne de valeur du développement (DVSM)

Processus utilisé pour identifier et hiérarchiser les contraintes qui nuisent à la rapidité et à la qualité du cycle de vie du développement logiciel. DVSM étend le processus de cartographie de la chaîne de valeur initialement conçu pour les pratiques de production allégée. Il met l'accent sur les étapes et les équipes nécessaires pour créer et transférer de la valeur tout au long du processus de développement logiciel.

jumeau numérique

Représentation virtuelle d'un système réel, tel qu'un bâtiment, une usine, un équipement industriel ou une ligne de production. Les jumeaux numériques prennent en charge la maintenance prédictive, la surveillance à distance et l'optimisation de la production.

tableau des dimensions

Dans un schéma en étoile, table plus petite contenant les attributs de données relatifs aux données quantitatives d'une table de faits. Les attributs des tables de dimensions sont généralement des champs de texte ou des nombres discrets qui se comportent comme du texte. Ces attributs sont couramment utilisés pour la contrainte des requêtes, le filtrage et l'étiquetage des ensembles de résultats.

catastrophe

Un événement qui empêche une charge de travail ou un système d'atteindre ses objectifs commerciaux sur son site de déploiement principal. Ces événements peuvent être des catastrophes naturelles, des défaillances techniques ou le résultat d'actions humaines, telles qu'une mauvaise configuration involontaire ou une attaque de logiciel malveillant.

reprise après sinistre (DR)

La stratégie et le processus que vous utilisez pour minimiser les temps d'arrêt et les pertes de données causés par un sinistre. Pour plus d'informations, consultez la section Reprise après sinistre des charges de travail sur AWS : Restauration dans le cloud dans le AWS Well-Architected Framework.

DML

Voir langage de manipulation de base de données.

conception axée sur le domaine

Approche visant à développer un système logiciel complexe en connectant ses composants à des domaines évolutifs, ou objectifs métier essentiels, que sert chaque composant. Ce concept a été introduit par Eric Evans dans son livre, Domain-Driven Design : Tackling Complexity in the Heart of Software (Boston : Addison-Wesley Professional, 2003). Pour plus d'informations sur la manière dont vous pouvez utiliser la conception axée sur le domaine avec le modèle Strangler Fig, consultez la section Modernisation incrémentielle des anciens services Web ASP.NET Microsoft (ASMX) à l'aide de conteneurs et d'Amazon API Gateway.

DR

Consultez la section Reprise après sinistre.

détection de dérive

Suivi des écarts par rapport à une configuration de référence. Par exemple, vous pouvez l'utiliser AWS CloudFormation pour détecter la dérive des ressources du système ou AWS Control Tower pour détecter les modifications de votre zone d'atterrissage susceptibles d'affecter le respect des exigences de gouvernance.

DVSM

Voir la cartographie de la chaîne de valeur du développement.

E

EDA

Voir analyse exploratoire des données.

EDI

Voir échange de données informatisé.

informatique de périphérie

Technologie qui augmente la puissance de calcul des appareils intelligents en périphérie d'un réseau IoT. Comparé au cloud computing, l'informatique de pointe peut réduire la latence des communications et améliorer le temps de réponse.

échange de données informatisé (EDI)

L'échange automatique de documents commerciaux entre les organisations. Pour plus d'informations, voir Qu'est-ce que l'échange de données informatisé ?

chiffrement

Processus informatique qui transforme des données en texte clair, lisibles par l'homme, en texte chiffré.

clé de chiffrement

Chaîne cryptographique de bits aléatoires générée par un algorithme cryptographique. La longueur des clés peut varier, et chaque clé est conçue pour être imprévisible et unique.

endianisme

Ordre dans lequel les octets sont stockés dans la mémoire de l'ordinateur. Big-endian les systèmes stockent d'abord l'octet le plus significatif. Little-endian les systèmes stockent d'abord l'octet le moins significatif.

point de terminaison

Voir point de terminaison de service.

service de point de terminaison

Service que vous pouvez héberger sur un cloud privé virtuel (VPC) pour le partager avec d'autres utilisateurs. Vous pouvez créer un service de point de terminaison avec AWS PrivateLink et accorder des autorisations à d'autres principaux Comptes AWS ou à Gestion des identités et des accès AWS (IAM) principaux. Ces comptes ou principaux peuvent se connecter à votre service de point de terminaison de manière privée en créant des points de terminaison d'un VPC d'interface. Pour plus d'informations, veuillez consulter Création d'un service de point de terminaison dans la documentation Amazon Virtual Private Cloud (Amazon VPC).

planification des ressources d'entreprise (ERP)

Système qui automatise et gère les principaux processus métier (tels que la comptabilité, le MES et la gestion de projet) pour une entreprise.

chiffrement d'enveloppe

Processus de chiffrement d'une clé de chiffrement à l'aide d'une autre clé de chiffrement. Pour plus d'informations, consultez la section Chiffrement des enveloppes dans la documentation AWS Key Management Service (AWS KMS).

environment

Instance d'une application en cours d'exécution. Les types d'environnement les plus courants dans le cloud computing sont les suivants :

- Environnement de développement : instance d'une application en cours d'exécution à laquelle seule l'équipe principale chargée de la maintenance de l'application peut accéder. Les environnements de développement sont utilisés pour tester les modifications avant de les promouvoir dans les environnements supérieurs. Ce type d'environnement est parfois appelé environnement de test.
- Environnements inférieurs : tous les environnements de développement d'une application, tels que ceux utilisés pour les générations et les tests initiaux.

- Environnement de production : instance d'une application en cours d'exécution à laquelle les utilisateurs finaux peuvent accéder. Dans un CI/CD pipeline, l'environnement de production est le dernier environnement de déploiement.
- Environnements supérieurs : tous les environnements accessibles aux utilisateurs autres que l'équipe de développement principale. Ils peuvent inclure un environnement de production, des environnements de préproduction et des environnements pour les tests d'acceptation par les utilisateurs.

époppée

Dans les méthodologies agiles, catégories fonctionnelles qui aident à organiser et à prioriser votre travail. Les épopées fournissent une description détaillée des exigences et des tâches d'implémentation. Par exemple, les points forts de la AWS CAF en matière de sécurité incluent la gestion des identités et des accès, les contrôles de détection, la sécurité des infrastructures, la protection des données et la réponse aux incidents. Pour plus d'informations sur les épopées dans la stratégie de migration AWS , veuillez consulter le guide d'implémentation du programme.

ERP

Voir Planification des ressources d'entreprise.

analyse exploratoire des données (EDA)

Processus d'analyse d'un jeu de données pour comprendre ses principales caractéristiques. Vous collectez ou agrégez des données, puis vous effectuez des enquêtes initiales pour trouver des modèles, détecter des anomalies et vérifier les hypothèses. L'EDA est réalisée en calculant des statistiques récapitulatives et en créant des visualisations de données.

F

tableau des faits

La table centrale dans un schéma en étoile. Il stocke des données quantitatives sur les opérations commerciales. Généralement, une table de faits contient deux types de colonnes : celles qui contiennent des mesures et celles qui contiennent une clé étrangère pour une table de dimensions.

échouer rapidement

Une philosophie qui utilise des tests fréquents et progressifs pour réduire le cycle de vie du développement. C'est un élément essentiel d'une approche agile.

limite d'isolation des défauts

Dans le AWS Cloud, une limite telle qu'une zone de disponibilité Région AWS, un plan de contrôle ou un plan de données qui limite l'effet d'une panne et contribue à améliorer la résilience des charges de travail. Pour plus d'informations, consultez la section Limites d'isolation des AWS pannes.

branche de fonctionnalités

Voir la succursale.

fonctionnalités

Les données d'entrée que vous utilisez pour faire une prédiction. Par exemple, dans un contexte de fabrication, les fonctionnalités peuvent être des images capturées périodiquement à partir de la ligne de fabrication.

importance des fonctionnalités

Le niveau d'importance d'une fonctionnalité pour les prédictions d'un modèle. Il s'exprime généralement sous la forme d'un score numérique qui peut être calculé à l'aide de différentes techniques, telles que la méthode Shapley Additive Explanations (SHAP) et les gradients intégrés. Pour plus d'informations, voir Interprétabilité du modèle d'apprentissage automatique avec AWS.

transformation de fonctionnalité

Optimiser les données pour le processus de ML, notamment en enrichissant les données avec des sources supplémentaires, en mettant à l'échelle les valeurs ou en extrayant plusieurs ensembles d'informations à partir d'un seul champ de données. Cela permet au modèle de ML de tirer parti des données. Par exemple, si vous décomposez la date « 2021-05-27 00:15:37 » en « 2021 », « mai », « jeudi » et « 15 », vous pouvez aider l'algorithme d'apprentissage à apprendre des modèles nuancés associés à différents composants de données.

invitation en quelques coups

Fournir à un LLM un petit nombre d'exemples illustrant la tâche et le résultat souhaité avant de lui demander d'effectuer une tâche similaire. Cette technique est une application de l'apprentissage contextuel, dans le cadre de laquelle les modèles apprennent à partir d'exemples (prises de vue) intégrés dans des instructions. Few-shot l'envoi d'instructions peut être efficace pour les tâches qui nécessitent un formatage, un raisonnement ou une connaissance du domaine spécifiques. Voir également l'invite Zero-Shot.

FGAC

Découvrez le contrôle d'accès détaillé.

contrôle d'accès détaillé (FGAC)

Utilisation de plusieurs conditions pour autoriser ou refuser une demande d'accès.

migration instantanée (flash-cut)

Méthode de migration de base de données qui utilise la réplication continue des données par le biais de la capture des données de modification afin de migrer les données dans les plus brefs délais, au lieu d'utiliser une approche progressive. L'objectif est de réduire au maximum les temps d'arrêt.

FM

Voir le modèle de fondation.

modèle de fondation (FM)

Un vaste réseau neuronal d'apprentissage profond qui s'entraîne sur des ensembles de données massifs de données généralisées et non étiquetées. Les FM sont capables d'effectuer une grande variété de tâches générales, telles que la compréhension du langage, la génération de texte et d'images et la conversation en langage naturel. Pour plus d'informations, voir Que sont les modèles de base ?

Passerelle FM

Un intermédiaire centralisé qui contrôle et normalise l'accès aux modèles de base. Également connue sous le nom de passerelle LLM.

G

IA générative

Sous-ensemble de modèles d'IA qui ont été entraînés sur de grandes quantités de données et qui peuvent utiliser une simple invite textuelle pour créer de nouveaux contenus et artefacts, tels que des images, des vidéos, du texte et du son. Pour plus d'informations, consultez Qu'est-ce que l'IA générative.

blocage géographique

Voir les restrictions géographiques.

restrictions géographiques (blocage géographique)

Sur Amazon CloudFront, option permettant d'empêcher les utilisateurs de certains pays d'accéder aux distributions de contenu. Vous pouvez utiliser une liste d'autorisation ou une liste de blocage

pour spécifier les pays approuvés et interdits. Pour plus d'informations, consultez la section Restreindre la distribution géographique de votre contenu dans la CloudFront documentation.

Flux de travail Gitflow

Approche dans laquelle les environnements inférieurs et supérieurs utilisent différentes branches dans un référentiel de code source. Le flux de travail Gitflow est considéré comme existant, et le flux de travail basé sur les troncs est l'approche moderne préférée.

image dorée

Un instantané d'un système ou d'un logiciel utilisé comme modèle pour déployer de nouvelles instances de ce système ou logiciel. Par exemple, dans le secteur de la fabrication, une image dorée peut être utilisée pour fournir des logiciels sur plusieurs appareils et contribue à améliorer la vitesse, l'évolutivité et la productivité des opérations de fabrication des appareils.

stratégie inédite

L'absence d'infrastructures existantes dans un nouvel environnement. Lorsque vous adoptez une stratégie inédite pour une architecture système, vous pouvez sélectionner toutes les nouvelles technologies sans restriction de compatibilité avec l'infrastructure existante, également appelée brownfield. Si vous étendez l'infrastructure existante, vous pouvez combiner des politiques brownfield (existantes) et greenfield (inédites).

barrière de protection

Règle de haut niveau qui permet de régir les ressources, les politiques et la conformité au sein des unités d'organisation (UO). Les barrières de protection préventives appliquent des politiques pour garantir l'alignement sur les normes de conformité. Elles sont mises en œuvre à l'aide de politiques de contrôle des services et de limites des autorisations IAM. Les barrières de protection de détection détectent les violations des politiques et les problèmes de conformité, et génèrent des alertes pour y remédier. Ils sont implémentés à l'aide d'Amazon AWS Config AWS Security Hub CSPM GuardDuty AWS Trusted Advisor, d'Amazon Inspector et de AWS Lambda contrôles personnalisés.

rambardes (AI)

Des mécanismes de sécurité qui filtrent, valident et limitent les entrées et sorties des agents afin de garantir un comportement responsable et sûr de l'IA.

H

HA

Découvrez la haute disponibilité.

migration de base de données hétérogène

Migration de votre base de données source vers une base de données cible qui utilise un moteur de base de données différent (par exemple, Oracle vers Amazon Aurora). La migration hétérogène fait généralement partie d'un effort de réarchitecture, et la conversion du schéma peut s'avérer une tâche complexe. AWS propose AWS SCT qui facilite les conversions de schémas.

haute disponibilité (HA)

Capacité d'une charge de travail à fonctionner en continu, sans intervention, en cas de difficultés ou de catastrophes. Les systèmes HA sont conçus pour basculer automatiquement, fournir constamment des performances de haute qualité et gérer différentes charges et défaillances avec un impact minimal sur les performances.

modernisation des historiens

Approche utilisée pour moderniser et mettre à niveau les systèmes de technologie opérationnelle (OT) afin de mieux répondre aux besoins de l'industrie manufacturière. Un historien est un type de base de données utilisé pour collecter et stocker des données provenant de diverses sources dans une usine.

données de rétention

Partie de données historiques étiquetées qui n'est pas divulguée dans un ensemble de données utilisé pour entraîner un modèle d'apprentissage automatique. Vous pouvez utiliser les données de blocage pour évaluer les performances du modèle en comparant les prévisions du modèle aux données de blocage.

humain dans la boucle (HiTL)

Un modèle de flux de travail dans lequel l'exécution des agents s'arrête pour examen et approbation par l'homme aux points de décision critiques.

migration de base de données homogène

Migration de votre base de données source vers une base de données cible qui partage le même moteur de base de données (par exemple, Microsoft SQL Server vers Amazon RDS for SQL Server). La migration homogène s'inscrit généralement dans le cadre d'un effort de

réhébergement ou de replateforme. Vous pouvez utiliser les utilitaires de base de données natifs pour migrer le schéma.

données chaudes

Données fréquemment consultées, telles que les données en temps réel ou les données translationnelles récentes. Ces données nécessitent généralement un niveau ou une classe de stockage à hautes performances pour fournir des réponses rapides aux requêtes.

correctif

Solution d'urgence à un problème critique dans un environnement de production. En raison de son urgence, un correctif est généralement créé en dehors du flux de travail de DevOps publication habituel.

période de soins intensifs

Immédiatement après le basculement, période pendant laquelle une équipe de migration gère et surveille les applications migrées dans le cloud afin de résoudre les problèmes éventuels. En règle générale, cette période dure de 1 à 4 jours. À la fin de la période de soins intensifs, l'équipe de migration transfère généralement la responsabilité des applications à l'équipe des opérations cloud.

I

IaC

Considérez l'infrastructure comme un code.

politique basée sur l’identité

Politique attachée à un ou plusieurs principaux IAM qui définit leurs autorisations au sein de l' AWS Cloud environnement.

application inactive

Application dont l'utilisation moyenne du processeur et de la mémoire se situe entre 5 et 20 % sur une période de 90 jours. Dans un projet de migration, il est courant de retirer ces applications ou de les retenir sur site.

IIoT

Voir Internet industriel des objets.

infrastructure immuable

Modèle qui déploie une nouvelle infrastructure pour les charges de travail de production au lieu de mettre à jour, d'appliquer des correctifs ou de modifier l'infrastructure existante. Les infrastructures immuables sont intrinsèquement plus cohérentes, fiables et prévisibles que les infrastructures mutables. Pour plus d'informations, consultez les meilleures pratiques de déploiement à l'aide d'une infrastructure immuable dans le AWS Well-Architected Framework.

VPC entrant (d'entrée)

Dans une architecture AWS multi-comptes, un VPC qui accepte, inspecte et achemine les connexions réseau depuis l'extérieur d'une application. L'architecture de référence de sécuritéAWS recommande de configurer votre compte réseau avec des VPC entrants, sortants et d'inspection afin de protéger l'interface bidirectionnelle entre votre application et Internet en général.

migration incrémentielle

Stratégie de basculement dans le cadre de laquelle vous migrez votre application par petites parties au lieu d'effectuer un basculement complet unique. Par exemple, il se peut que vous ne transfériez que quelques microservices ou utilisateurs vers le nouveau système dans un premier temps. Après avoir vérifié que tout fonctionne correctement, vous pouvez transférer progressivement des microservices ou des utilisateurs supplémentaires jusqu'à ce que vous puissiez mettre hors service votre système hérité. Cette stratégie réduit les risques associés aux migrations de grande ampleur.

Industry 4.0

Terme introduit par Klaus Schwab en 2016 pour désigner la modernisation des processus de fabrication grâce aux avancées en matière de connectivité, de données en temps réel, d'automatisation, d'analyse et. AI/ML

infrastructure

Ensemble des ressources et des actifs contenus dans l'environnement d'une application.

infrastructure en tant que code (IaC)

Processus de mise en service et de gestion de l'infrastructure d'une application via un ensemble de fichiers de configuration. IaC est conçue pour vous aider à centraliser la gestion de l'infrastructure, à normaliser les ressources et à mettre à l'échelle rapidement afin que les nouveaux environnements soient reproductibles, fiables et cohérents.

internet industriel des objets (IIoT)

L'utilisation de capteurs et d'appareils connectés à Internet dans les secteurs industriels tels que la fabrication, l'énergie, l'automobile, les soins de santé, les sciences de la vie et l'agriculture. Pour plus d'informations, veuillez consulter Building an industrial Internet of Things (IIoT) digital transformation strategy.

VPC d'inspection

Dans une architecture AWS multi-comptes, un VPC centralisé qui gère les inspections du trafic réseau entre les VPC (identiques ou Régions AWS différents), Internet et les réseaux sur site. L'architecture de référence de sécuritéAWS recommande de configurer votre compte réseau avec des VPC entrants, sortants et d'inspection afin de protéger l'interface bidirectionnelle entre votre application et Internet en général.

Internet des objets (IoT)

Réseau d'objets physiques connectés dotés de capteurs ou de processeurs intégrés qui communiquent avec d'autres appareils et systèmes via Internet ou via un réseau de communication local. Pour plus d'informations, veuillez consulter la section Qu'est-ce que l'IoT ?.

interprétabilité

Caractéristique d'un modèle de machine learning qui décrit dans quelle mesure un être humain peut comprendre comment les prédictions du modèle dépendent de ses entrées. Pour plus d'informations, voir Interprétabilité du modèle d'apprentissage automatique avec AWS.

IoT

Voir Internet des objets.

Bibliothèque d'informations informatiques (ITIL)

Ensemble de bonnes pratiques pour proposer des services informatiques et les aligner sur les exigences métier. L'ITIL constitue la base de l'ITSM.

gestion des services informatiques (ITSM)

Activités associées à la conception, à la mise en œuvre, à la gestion et à la prise en charge de services informatiques d'une organisation. Pour plus d'informations sur l'intégration des opérations cloud aux outils ITSM, veuillez consulter le guide d'intégration des opérations.

ITIL

Consultez la bibliothèque d'informations informatiques.

ITSM

Voir Gestion des services informatiques.

L

contrôle d'accès basé sur des étiquettes (LBAC)

Une implémentation du contrôle d'accès obligatoire (MAC) dans laquelle une valeur d'étiquette de sécurité est explicitement attribuée aux utilisateurs et aux données elles-mêmes. L'intersection entre l'étiquette de sécurité utilisateur et l'étiquette de sécurité des données détermine les lignes et les colonnes visibles par l'utilisateur.

zone de destination

Une zone d'atterrissage est un AWS environnement multi-comptes bien conçu, évolutif et sécurisé. Il s'agit d'un point de départ à partir duquel vos entreprises peuvent rapidement lancer et déployer des charges de travail et des applications en toute confiance dans leur environnement de sécurité et d'infrastructure. Pour plus d'informations sur les zones de destination, veuillez consulter Setting up a secure and scalable multi-account AWS environment.

grand modèle de langage (LLM)

Un modèle d'intelligence artificielle basé sur le deep learning qui est préentraîné sur une grande quantité de données. Un LLM peut effectuer plusieurs tâches, telles que répondre à des questions, résumer des documents, traduire du texte dans d'autres langues et compléter des phrases. Pour plus d'informations, voir Que sont les LLM.

migration de grande envergure

Migration de 300 serveurs ou plus.

LBAC

Voir contrôle d'accès basé sur des étiquettes.

principe de moindre privilège

Bonne pratique de sécurité qui consiste à accorder les autorisations minimales nécessaires à l'exécution d'une tâche. Pour plus d'informations, veuillez consulter la rubrique Accorder les autorisations de moindre privilège dans la documentation IAM.

lift and shift

Voir 7 Rs.

système de poids faible

Système qui stocke d'abord l'octet le moins significatif. Voir aussi endianité.

LLM

Voir le grand modèle de langage.

environnements inférieurs

Voir environnement.

M

machine learning (ML)

Type d'intelligence artificielle qui utilise des algorithmes et des techniques pour la reconnaissance et l'apprentissage de modèles. Le ML analyse et apprend à partir de données enregistrées, telles que les données de l'Internet des objets (IoT), pour générer un modèle statistique basé sur des modèles. Pour plus d'informations, veuillez consulter Machine Learning.

branche principale

Voir la succursale.

malware

Logiciel conçu pour compromettre la sécurité ou la confidentialité de l'ordinateur. Les logiciels malveillants peuvent perturber les systèmes informatiques, divulguer des informations sensibles ou obtenir un accès non autorisé. Parmi les malwares, on peut citer les virus, les vers, les rançongiciels, les chevaux de Troie, les logiciels espions et les enregistreurs de frappe.

services gérés

Services AWS pour lequel AWS fonctionnent la couche d'infrastructure, le système d'exploitation et les plateformes, et vous accédez aux points de terminaison pour stocker et récupérer des données. Amazon Simple Storage Service (Amazon S3) et Amazon DynamoDB sont des exemples de services gérés. Ils sont également connus sous le nom de services abstraits.

système d'exécution de la fabrication (MES)

Un système logiciel pour le suivi, la surveillance, la documentation et le contrôle des processus de production qui convertissent les matières premières en produits finis dans l'atelier.

MAP

Voir Migration Acceleration Program.

MCP

Voir Model Context Protocol.

Protocole de contexte du modèle (MCP)

Protocole sans état pour la communication entre un agent et un outil.

serveur MCP

Service qui expose un ou plusieurs outils via le protocole Model Context.

mécanisme

Processus complet au cours duquel vous créez un outil, favorisez son adoption, puis inspectez les résultats afin de procéder aux ajustements nécessaires. Un mécanisme est un cycle qui se renforce et s'améliore au fur et à mesure de son fonctionnement. Pour plus d'informations, voir Création de mécanismes dans le AWS Well-Architected cadre.

compte membre

Tous, à l' Comptes AWS exception du compte de gestion, qui font partie d'une organisation dans AWS Organizations. Un compte ne peut être membre que d'une seule organisation à la fois.

MAILLES

Voir le système d'exécution de la fabrication.

Transport télémétrique en file d'attente de messages (MQTT)

Un protocole de communication léger de machine à machine (M2M), basé sur le publish/subscribemodèle, pour les appareils IoT aux ressources limitées.

microservice

Petit service indépendant qui communique via des API bien définies et qui est généralement détenu par de petites équipes autonomes. Par exemple, un système d'assurance peut inclure des microservices qui mappent à des capacités métier, telles que les ventes ou le marketing, ou à des sous-domaines, tels que les achats, les réclamations ou l'analytique. Les avantages des microservices incluent l'agilité, la flexibilité de la mise à l'échelle, la facilité de déploiement, la réutilisation du code et la résilience. Pour plus d'informations, consultez la section Intégration de microservices à l'aide de services AWS sans serveur.

architecture de microservices

Approche de création d'une application avec des composants indépendants qui exécutent chaque processus d'application en tant que microservice. Ces microservices communiquent via une interface bien définie à l'aide d'API légères. Chaque microservice de cette architecture peut être mis à jour, déployé et mis à l'échelle pour répondre à la demande de fonctions spécifiques d'une application. Pour plus d'informations, consultez la section Implémentation de microservices sur AWS.

Programme d'accélération des migrations (MAP)

Un AWS programme qui fournit un support de conseil, des formations et des services pour aider les entreprises à établir une base opérationnelle solide pour passer au cloud, et pour aider à compenser le coût initial des migrations. MAP inclut une méthodologie de migration pour exécuter les migrations héritées de manière méthodique, ainsi qu'un ensemble d'outils pour automatiser et accélérer les scénarios de migration courants.

migration à grande échelle

Processus consistant à transférer la majeure partie du portefeuille d'applications vers le cloud par vagues, un plus grand nombre d'applications étant déplacées plus rapidement à chaque vague. Cette phase utilise les bonnes pratiques et les enseignements tirés des phases précédentes pour implémenter une usine de migration d'équipes, d'outils et de processus en vue de rationaliser la migration des charges de travail grâce à l'automatisation et à la livraison agile. Il s'agit de la troisième phase de la stratégie de migration AWS.

usine de migration

Cross-functional des équipes qui rationalisent la migration des charges de travail grâce à des approches automatisées et agiles. Les équipes de Migration Factory comprennent généralement des responsables des opérations, des analystes commerciaux et des propriétaires, des ingénieurs de migration, des développeurs et DevOps des professionnels travaillant dans le cadre de sprints. Entre 20 et 50 % du portefeuille d'applications d'entreprise est constitué de modèles répétés qui peuvent être optimisés par une approche d'usine. Pour plus d'informations, veuillez consulter la rubrique discussion of migration factories et le guide Cloud Migration Factory dans cet ensemble de contenus.

métadonnées de migration

Informations relatives à l'application et au serveur nécessaires pour finaliser la migration. Chaque modèle de migration nécessite un ensemble de métadonnées de migration différent. Les

exemples de métadonnées de migration incluent le sous-réseau cible, le groupe de sécurité et le AWS compte.

modèle de migration

Tâche de migration reproductible qui détaille la stratégie de migration, la destination de la migration et l'application ou le service de migration utilisé. Exemple : réorganisez la migration vers Amazon EC2 AWS avec le service de migration d'applications.

Évaluation du portefeuille de migration (MPA)

Outil en ligne qui fournit des informations pour valider l'analyse de rentabilisation en faveur de la migration vers le. AWS Cloud La MPA propose une évaluation détaillée du portefeuille (dimensionnement approprié des serveurs, tarification, comparaison du coût total de possession, analyse des coûts de migration), ainsi que la planification de la migration (analyse et collecte des données d'applications, regroupement des applications, priorisation des migrations et planification des vagues). L'outil MPA (connexion requise) est disponible gratuitement pour tous les AWS consultants et consultants APN Partner.

Évaluation de la préparation à la migration (MRA)

Processus qui consiste à obtenir des informations sur l'état de préparation d'une organisation au cloud, à identifier les forces et les faiblesses et à élaborer un plan d'action pour combler les lacunes identifiées, à l'aide du AWS CAF. Pour plus d'informations, veuillez consulter le guide de préparation à la migration. La MRA est la première phase de la stratégie de migration AWS.

stratégie de migration

L'approche utilisée pour migrer une charge de travail vers le AWS Cloud. Pour plus d'informations, reportez-vous aux 7 R de ce glossaire et à Mobiliser votre organisation pour accélérer les migrations à grande échelle.

ML

Voir apprentissage automatique.

modernisation

Transformation d'une application obsolète (héritée ou monolithique) et de son infrastructure en un système agile, élastique et hautement disponible dans le cloud afin de réduire les coûts, de gagner en efficacité et de tirer parti des innovations. Pour plus d'informations, consultez la section Stratégie de modernisation des applications dans le AWS Cloud.

évaluation de la préparation à la modernisation

Évaluation qui permet de déterminer si les applications d'une organisation sont prêtes à être modernisées, d'identifier les avantages, les risques et les dépendances, et qui détermine dans quelle mesure l'organisation peut prendre en charge l'état futur de ces applications. Le résultat de l'évaluation est un plan de l'architecture cible, une feuille de route détaillant les phases de développement et les étapes du processus de modernisation, ainsi qu'un plan d'action pour combler les lacunes identifiées. Pour plus d'informations, consultez la section Évaluation de l'état de préparation à la modernisation des applications dans le AWS Cloud.

applications monolithiques (monolithes)

Applications qui s'exécutent en tant que service unique avec des processus étroitement couplés. Les applications monolithiques ont plusieurs inconvénients. Si une fonctionnalité de l'application connaît un pic de demande, l'architecture entière doit être mise à l'échelle. L'ajout ou l'amélioration des fonctionnalités d'une application monolithique devient également plus complexe lorsque la base de code s'élargit. Pour résoudre ces problèmes, vous pouvez utiliser une architecture de microservices. Pour plus d'informations, veuillez consulter Decomposing monoliths into microservices.

MPA

Voir Évaluation du portefeuille de migration.

MQTT

Voir Message Queuing Telemetry Transport.

classification multi-classes

Processus qui permet de générer des prédictions pour plusieurs classes (prédiction d'un résultat parmi plus de deux). Par exemple, un modèle de ML peut demander « Ce produit est-il un livre, une voiture ou un téléphone ? » ou « Quelle catégorie de produits intéresse le plus ce client ? ».

infrastructure mutable

Modèle qui met à jour et modifie l'infrastructure existante pour les charges de travail de production. Pour améliorer la cohérence, la fiabilité et la prévisibilité, le AWS Well-Architected Framework recommande l'utilisation d'une infrastructure immuable comme meilleure pratique.

O

OAC

Voir Contrôle d'accès à l'origine.

OAI

Voir l'identité d'accès à l'origine.

OCM

Voir gestion du changement organisationnel.

migration hors ligne

Méthode de migration dans laquelle la charge de travail source est supprimée au cours du processus de migration. Cette méthode implique un temps d'arrêt prolongé et est généralement utilisée pour de petites charges de travail non critiques.

OI

Consultez la section Intégration des opérations.

OLA

Voir l'accord au niveau opérationnel.

migration en ligne

Méthode de migration dans laquelle la charge de travail source est copiée sur le système cible sans être mise hors ligne. Les applications connectées à la charge de travail peuvent continuer à fonctionner pendant la migration. Cette méthode implique un temps d'arrêt nul ou minimal et est généralement utilisée pour les charges de travail de production critiques.

OPC-UA

Voir Open Process Communications - Architecture unifiée.

Communications par processus ouvert - Architecture unifiée (OPC-UA)

Protocole de communication machine à machine (M2M) pour l'automatisation industrielle. OPC-UA fournit une norme d'interopérabilité avec des schémas de chiffrement, d'authentification et d'autorisation des données.

accord au niveau opérationnel (OLA)

Accord qui précise ce que les groupes informatiques fonctionnels s'engagent à fournir les uns aux autres, afin de prendre en charge un contrat de niveau de service (SLA).

examen de l'état de préparation opérationnelle (ORR)

Une liste de questions et de bonnes pratiques associées qui vous aident à comprendre, à évaluer, à prévenir ou à réduire l'ampleur des incidents et des défaillances possibles. Pour plus d'informations, voir Examens de l'état de préparation opérationnelle (ORR) dans le AWS Well-Architected cadre.

technologie opérationnelle (OT)

Systèmes matériels et logiciels qui fonctionnent avec l'environnement physique pour contrôler les opérations, les équipements et les infrastructures industriels. Dans le secteur manufacturier, l'intégration des systèmes OT et des technologies de l'information (IT) est au cœur des transformations de l'industrie 4.0.

intégration des opérations (OI)

Processus de modernisation des opérations dans le cloud, qui implique la planification de la préparation, l'automatisation et l'intégration. Pour en savoir plus, veuillez consulter le guide d'intégration des opérations.

journal de suivi d'organisation

Un parcours créé par AWS CloudTrail qui enregistre tous les événements pour tous les membres Comptes AWS d'une organisation dans AWS Organizations. Ce journal de suivi est créé dans chaque Compte AWS qui fait partie de l'organisation et suit l'activité de chaque compte. Pour plus d'informations, consultez la section Création d'un suivi pour une organisation dans la CloudTrail documentation.

gestion du changement organisationnel (OCM)

Cadre pour gérer les transformations métier majeures et perturbatrices du point de vue des personnes, de la culture et du leadership. L'OCM aide les organisations à se préparer et à effectuer la transition vers de nouveaux systèmes et de nouvelles politiques en accélérant l'adoption des changements, en abordant les problèmes de transition et en favorisant des changements culturels et organisationnels. Dans la stratégie de AWS migration, ce cadre est appelé accélération du personnel, en raison de la rapidité du changement requise dans les projets d'adoption du cloud. Pour plus d'informations, veuillez consulter le guide OCM.

contrôle d'accès d'origine (OAC)

Dans CloudFront, une option améliorée pour restreindre l'accès afin de sécuriser votre contenu Amazon Simple Storage Service (Amazon S3). OAC prend en charge tous les compartiments S3 dans leur ensemble Régions AWS, le chiffrement côté serveur avec AWS KMS (SSE-KMS) et les `DELETE` requêtes dynamiques `PUT` adressées au compartiment S3.

identité d'accès d'origine (OAI)

Dans CloudFront, une option permettant de restreindre l'accès afin de sécuriser votre contenu Amazon S3. Lorsque vous utilisez OAI, il CloudFront crée un principal auprès duquel Amazon S3 peut s'authentifier. Les principaux authentifiés ne peuvent accéder au contenu d'un compartiment S3 que par le biais d'une distribution spécifique CloudFront . Voir également OAC, qui fournit un contrôle d'accès plus précis et amélioré.

ORR

Voir l'examen de l'état de préparation opérationnelle.

DE

Voir technologie opérationnelle.

VPC sortant (de sortie)

Dans une architecture AWS multi-comptes, un VPC qui gère les connexions réseau initiées depuis une application. L'architecture de référence de sécuritéAWS recommande de configurer votre compte réseau avec des VPC entrants, sortants et d'inspection afin de protéger l'interface bidirectionnelle entre votre application et Internet en général.

P

limite des autorisations

Politique de gestion IAM attachée aux principaux IAM pour définir les autorisations maximales que peut avoir l'utilisateur ou le rôle. Pour plus d'informations, veuillez consulter la rubrique Limites des autorisations dans la documentation IAM.

informations personnelles identifiables (PII)

Informations qui, lorsqu'elles sont consultées directement ou associées à d'autres données connexes, peuvent être utilisées pour déduire raisonnablement l'identité d'une personne. Les

exemples d'informations personnelles incluent les noms, les adresses et les informations de contact.

PII

Voir les informations personnelles identifiables.

manuel stratégique

Ensemble d'étapes prédéfinies qui capturent le travail associé aux migrations, comme la fourniture de fonctions d'opérations de base dans le cloud. Un manuel stratégique peut revêtir la forme de scripts, de runbooks automatisés ou d'un résumé des processus ou des étapes nécessaires au fonctionnement de votre environnement modernisé.

PLC

Voir contrôleur logique programmable.

PLM

Consultez la section Gestion du cycle de vie des produits.

policy

Objet capable de définir les autorisations (voir la politique basée sur l'identité), de spécifier les conditions d'accès (voir la politique basée sur les ressources) ou de définir les autorisations maximales pour tous les comptes d'une organisation dans AWS Organizations (voir la politique de contrôle des services).

persistance polyglotte

Choix indépendant de la technologie de stockage de données d'un microservice en fonction des modèles d'accès aux données et d'autres exigences. Si vos microservices utilisent la même technologie de stockage de données, ils peuvent rencontrer des difficultés d'implémentation ou présenter des performances médiocres. Les microservices sont plus faciles à mettre en œuvre, atteignent de meilleures performances, ainsi qu'une meilleure capacité de mise à l'échelle s'ils utilisent l'entrepôt de données le mieux adapté à leurs besoins.

évaluation du portefeuille

Processus de découverte, d'analyse et de priorisation du portefeuille d'applications afin de planifier la migration. Pour plus d'informations, veuillez consulter Evaluating migration readiness.

predicate

Une condition de requête qui renvoie `true` ou`false`, généralement située dans une `WHERE` clause.

prédicat pushdown

Technique d'optimisation des requêtes de base de données qui filtre les données de la requête avant le transfert. Cela réduit la quantité de données qui doivent être extraites et traitées à partir de la base de données relationnelle et améliore les performances des requêtes.

contrôle préventif

Contrôle de sécurité conçu pour empêcher qu'un événement ne se produise. Ces contrôles constituent une première ligne de défense pour empêcher tout accès non autorisé ou toute modification indésirable de votre réseau. Pour plus d'informations, veuillez consulter Preventative controls dans Implementing security controls on AWS.

principal

Entité capable d'effectuer AWS des actions et d'accéder à des ressources. Cette entité est généralement un utilisateur root pour un Compte AWS rôle IAM ou un utilisateur. Pour plus d'informations, veuillez consulter la rubrique Principal dans Termes et concepts relatifs aux rôles, dans la documentation IAM.

confidentialité dès la conception

Une approche d'ingénierie système qui prend en compte la confidentialité tout au long du processus de développement.

zones hébergées privées

Conteneur qui contient des informations concernant la façon dont vous souhaitez qu'Amazon Route 53 réponde aux requêtes DNS pour un domaine et ses sous-domaines dans un ou plusieurs VPC. Pour plus d'informations, veuillez consulter Working with private hosted zones dans la documentation Route 53.

contrôle proactif

Contrôle de sécurité conçu pour empêcher le déploiement de ressources non conformes. Ces contrôles analysent les ressources avant qu'elles ne soient provisionnées. Si la ressource n'est pas conforme au contrôle, elle n'est pas provisionnée. Pour plus d'informations, consultez le guide de référence sur les contrôles dans la AWS Control Tower documentation et consultez la section Contrôles proactifs dans Implémentation des contrôles de sécurité sur AWS.

gestion du cycle de vie des produits (PLM)

Gestion des données et des processus d'un produit tout au long de son cycle de vie, depuis la conception, le développement et le lancement, en passant par la croissance et la maturité, jusqu'au déclin et au retrait.

environnement de production

Voir environnement.

contrôleur logique programmable (PLC)

Dans le secteur manufacturier, un ordinateur hautement fiable et adaptable qui surveille les machines et automatise les processus de fabrication.

chaînage rapide

Utiliser le résultat d'une invite LLM comme entrée pour l'invite suivante afin de générer de meilleures réponses. Cette technique est utilisée pour décomposer une tâche complexe en sous-tâches ou pour affiner ou développer de manière itérative une réponse préliminaire. Cela permet d'améliorer la précision et la pertinence des réponses d'un modèle et permet d'obtenir des résultats plus précis et personnalisés.

pseudonymisation

Processus de remplacement des identifiants personnels dans un ensemble de données par des valeurs fictives. La pseudonymisation peut contribuer à protéger la vie privée. Les données pseudonymisées sont toujours considérées comme des données personnelles.

publish/subscribe (pub/sub)

Modèle qui permet des communications asynchrones entre les microservices afin d'améliorer l'évolutivité et la réactivité. Par exemple, dans un MES basé sur des microservices, un microservice peut publier des messages d'événements sur un canal auquel d'autres microservices peuvent s'abonner. Le système peut ajouter de nouveaux microservices sans modifier le service de publication.

Q

plan de requête

Série d'étapes, telles que des instructions, utilisées pour accéder aux données d'un système de base de données relationnelle SQL.

régression du plan de requêtes

Le cas où un optimiseur de service de base de données choisit un plan moins optimal qu'avant une modification donnée de l'environnement de base de données. Cela peut être dû à des

changements en termes de statistiques, de contraintes, de paramètres d'environnement, de liaisons de paramètres de requêtes et de mises à jour du moteur de base de données.

R

Matrice RACI

Voir responsable, responsable, consulté, informé (RACI).

RAG

Voir Retrieval Augmented Generation.

rançongiciel

Logiciel malveillant conçu pour bloquer l'accès à un système informatique ou à des données jusqu'à ce qu'un paiement soit effectué.

Matrice RASCI

Voir responsable, responsable, consulté, informé (RACI).

RCAC

Voir contrôle d'accès aux lignes et aux colonnes.

réplica en lecture

Copie d'une base de données utilisée en lecture seule. Vous pouvez acheminer les requêtes vers le réplica de lecture pour réduire la charge sur votre base de données principale.

réarchitecte

Voir 7 Rs.

objectif de point de récupération (RPO)

Durée maximale acceptable depuis le dernier point de récupération des données. Il détermine ce qui est considéré comme étant une perte de données acceptable entre le dernier point de reprise et l'interruption du service.

objectif de temps de récupération (RTO)

Le délai maximum acceptable entre l'interruption du service et le rétablissement du service.

refactoriser

Voir 7 Rs.

Région

Un ensemble de AWS ressources dans une zone géographique. Chacune Région AWS est isolée et indépendante des autres pour garantir tolérance aux pannes, stabilité et résilience. Pour plus d'informations, voir Spécifier ce que Régions AWS votre compte peut utiliser.

régression

Technique de ML qui prédit une valeur numérique. Par exemple, pour résoudre le problème « Quel sera le prix de vente de cette maison ? », un modèle de ML pourrait utiliser un modèle de régression linéaire pour prédire le prix de vente d'une maison sur la base de faits connus à son sujet (par exemple, la superficie en mètres carrés).

réhéberger

Voir 7 Rs.

version

Dans un processus de déploiement, action visant à promouvoir les modifications apportées à un environnement de production.

déplacer

Voir 7 Rs.

replateforme

Voir 7 Rs.

rachat

Voir 7 Rs.

résilience

La capacité d'une application à résister aux perturbations ou à s'en remettre. La haute disponibilité et la reprise après sinistre sont des considérations courantes lors de la planification de la résilience dans le AWS Cloud. Pour plus d'informations, consultez la section AWS Cloud Résilience.

politique basée sur les ressources

Politique attachée à une ressource, comme un compartiment Amazon S3, un point de terminaison ou une clé de chiffrement. Ce type de politique précise les principaux auxquels l'accès est autorisé, les actions prises en charge et toutes les autres conditions qui doivent être remplies.

matrice responsable, redevable, consulté et informé (RACI)

Une matrice qui définit les rôles et les responsabilités de toutes les parties impliquées dans les activités de migration et les opérations cloud. Le nom de la matrice est dérivé des types de responsabilité définis dans la matrice : responsable (R), responsable (A), consulté (C) et informé (I). Le type de support (S) est facultatif. Si vous incluez le support, la matrice est appelée matrice RASCI, et si vous l'excluez, elle est appelée matrice RACI.

contrôle réactif

Contrôle de sécurité conçu pour permettre de remédier aux événements indésirables ou aux écarts par rapport à votre référence de sécurité. Pour plus d'informations, veuillez consulter la rubrique Responsive controls dans Implementing security controls on AWS.

retain

Voir 7 Rs.

se retirer

Voir 7 Rs.

Génération augmentée de récupération (RAG)

Technologie d'IA générative dans laquelle un LLM fait référence à une source de données faisant autorité qui se trouve en dehors de ses sources de données de formation avant de générer une réponse. Par exemple, un modèle RAG peut effectuer une recherche sémantique dans la base de connaissances ou dans les données personnalisées d'une organisation. Pour plus d'informations, voir Qu'est-ce que RAG ?

rotation

Processus de mise à jour périodique d'un secret pour empêcher un attaquant d'accéder aux informations d'identification.

contrôle d'accès aux lignes et aux colonnes (RCAC)

Utilisation d'expressions SQL simples et flexibles dotées de règles d'accès définies. Le RCAC comprend des autorisations de ligne et des masques de colonnes.

RPO

Voir l'objectif du point de récupération.

RTO

Voir l'objectif en matière de temps de rétablissement.

runbook

Ensemble de procédures manuelles ou automatisées nécessaires à l'exécution d'une tâche spécifique. Elles visent généralement à rationaliser les opérations ou les procédures répétitives présentant des taux d'erreur élevés.

S

SAML 2.0

Un standard ouvert utilisé par de nombreux fournisseurs d'identité (IdPs). Cette fonctionnalité permet l'authentification unique fédérée (SSO), afin que les utilisateurs puissent se connecter AWS Management Console ou appeler les opérations de l' AWS API sans que vous ayez à créer un utilisateur dans IAM pour tous les membres de votre organisation. Pour plus d'informations sur la fédération SAML 2.0, veuillez consulter À propos de la fédération SAML 2.0 dans la documentation IAM.

SCADA

Voir Contrôle de supervision et acquisition de données.

SCP

Voir la politique de contrôle des services.

secret

Dans AWS Secrets Manager des informations confidentielles ou restreintes, telles qu'un mot de passe ou des informations d'identification utilisateur, que vous stockez sous forme cryptée. Il comprend la valeur secrète et ses métadonnées. La valeur secrète peut être binaire, une chaîne unique ou plusieurs chaînes. Pour plus d'informations, voir Que contient le secret d'un Secrets Manager ? dans la documentation de Secrets Manager.

sécurité dès la conception

Une approche d'ingénierie système qui prend en compte la sécurité tout au long du processus de développement.

contrôle de sécurité

Barrière de protection technique ou administrative qui empêche, détecte ou réduit la capacité d'un assaillant d'exploiter une vulnérabilité de sécurité. Il existe quatre principaux types de contrôles de sécurité : préventifs, détectifs, réactifs et proactifs.

renforcement de la sécurité

Processus qui consiste à réduire la surface d'attaque pour la rendre plus résistante aux attaques. Cela peut inclure des actions telles que la suppression de ressources qui ne sont plus requises, la mise en œuvre des bonnes pratiques de sécurité consistant à accorder le moindre privilège ou la désactivation de fonctionnalités inutiles dans les fichiers de configuration.

système de gestion des informations et des événements de sécurité (SIEM)

Outils et services qui associent les systèmes de gestion des informations de sécurité (SIM) et de gestion des événements de sécurité (SEM). Un système SIEM collecte, surveille et analyse les données provenant de serveurs, de réseaux, d'appareils et d'autres sources afin de détecter les menaces et les failles de sécurité, mais aussi de générer des alertes.

automatisation des réponses de sécurité

Action prédéfinie et programmée conçue pour répondre automatiquement à un événement de sécurité ou y remédier. Ces automatisations servent de contrôles de sécurité détectifs ou réactifs qui vous aident à mettre en œuvre les meilleures pratiques en matière AWS de sécurité. Parmi les actions de réponse automatique, citons la modification d'un groupe de sécurité VPC, l'application de correctifs à une instance Amazon EC2 ou la rotation des informations d'identification.

chiffrement côté serveur

Chiffrement des données à destination, par celui Service AWS qui les reçoit.

Politique de contrôle des services (SCP)

Politique qui propose un contrôle centralisé des autorisations pour tous les comptes d'une organisation dans AWS Organizations. Les SCP définissent des barrières de protection ou des limites aux actions qu'un administrateur peut déléguer à des utilisateurs ou à des rôles. Vous pouvez utiliser les SCP comme listes d'autorisation ou de refus, pour indiquer les services ou les actions autorisés ou interdits. Pour plus d'informations, consultez la section Politiques de contrôle des services dans la AWS Organizations documentation.

point de terminaison du service

URL du point d'entrée pour un Service AWS. Pour vous connecter par programmation au service cible, vous pouvez utiliser un point de terminaison. Pour plus d'informations, veuillez consulter la rubrique Service AWS endpoints dans Références générales AWS.

contrat de niveau de service (SLA)

Accord qui précise ce qu'une équipe informatique promet de fournir à ses clients, comme le temps de disponibilité et les performances des services.

indicateur de niveau de service (SLI)

Mesure d'un aspect des performances d'un service, tel que son taux d'erreur, sa disponibilité ou son débit.

objectif de niveau de service (SLO)

Mesure cible qui représente l'état d'un service, tel que mesuré par un indicateur de niveau de service.

modèle de responsabilité partagée

Un modèle décrivant la responsabilité que vous partagez en matière AWS de sécurité et de conformité dans le cloud. AWS est responsable de la sécurité du cloud, alors que vous êtes responsable de la sécurité dans le cloud. Pour de plus amples informations, veuillez consulter Modèle de responsabilité partagée.

IA de l'ombre

Applications d'IA non autorisées créées ou utilisées en dehors des canaux régis au sein d'une organisation.

SIEM

Consultez les informations de sécurité et le système de gestion des événements.

point de défaillance unique (SPOF)

Défaillance d'un seul composant critique d'une application susceptible de perturber le système.

SLA

Voir le contrat de niveau de service.

SLI

Voir l'indicateur de niveau de service.

SLO

Voir l'objectif de niveau de service.

modèle split-and-seed

Modèle permettant de mettre à l'échelle et d'accélérer les projets de modernisation. Au fur et à mesure que les nouvelles fonctionnalités et les nouvelles versions de produits sont définies, l'équipe principale se divise pour créer des équipes de produit. Cela permet de mettre à l'échelle

les capacités et les services de votre organisation, d'améliorer la productivité des développeurs et de favoriser une innovation rapide. Pour plus d'informations, consultez la section Approche progressive de la modernisation des applications dans le. AWS Cloud

SPOF

Voir point de défaillance unique.

schéma en étoile

Structure organisationnelle de base de données qui utilise une grande table de faits pour stocker les données transactionnelles ou mesurées et utilise une ou plusieurs tables dimensionnelles plus petites pour stocker les attributs des données. Cette structure est conçue pour être utilisée dans un entrepôt de données ou à des fins de business intelligence.

modèle de figuier étrangleur

Approche de modernisation des systèmes monolithiques en réécrivant et en remplaçant progressivement les fonctionnalités du système jusqu'à ce que le système hérité puisse être mis hors service. Ce modèle utilise l'analogie d'un figuier de vigne qui se développe dans un arbre existant et qui finit par supplanter son hôte. Le schéma a été présenté par Martin Fowler comme un moyen de gérer les risques lors de la réécriture de systèmes monolithiques. Pour un exemple d'application de ce modèle, consultez la section Modernisation progressive des anciens services Web Microsoft ASP.NET (ASMX) à l'aide de conteneurs et d'Amazon API Gateway.

sous-réseau

Plage d'adresses IP dans votre VPC. Un sous-réseau doit se trouver dans une seule zone de disponibilité.

contrôle de supervision et acquisition de données (SCADA)

Dans le secteur manufacturier, un système qui utilise du matériel et des logiciels pour surveiller les actifs physiques et les opérations de production.

chiffrement symétrique

Algorithme de chiffrement qui utilise la même clé pour chiffrer et déchiffrer les données.

tests synthétiques

Tester un système de manière à simuler les interactions des utilisateurs afin de détecter les problèmes potentiels ou de surveiller les performances. Vous pouvez utiliser Amazon CloudWatch Synthetics pour créer ces tests.

invite du système

Technique permettant de fournir un contexte, des instructions ou des directives à un LLM afin d'orienter son comportement. Les instructions du système aident à définir le contexte et à établir des règles pour les interactions avec les utilisateurs.

T

tags

Key-value des paires qui agissent comme des métadonnées pour organiser vos AWS ressources. Les balises peuvent vous aider à gérer, identifier, organiser, rechercher et filtrer des ressources. Pour plus d'informations, veuillez consulter la rubrique Balisage de vos AWS ressources.

variable cible

La valeur que vous essayez de prédire dans le cadre du ML supervisé. Elle est également qualifiée de variable de résultat. Par exemple, dans un environnement de fabrication, la variable cible peut être un défaut du produit.

liste de tâches

Outil utilisé pour suivre les progrès dans un runbook. Liste de tâches qui contient une vue d'ensemble du runbook et une liste des tâches générales à effectuer. Pour chaque tâche générale, elle inclut le temps estimé nécessaire, le propriétaire et l'avancement.

environnement de test

Voir environnement.

entraînement

Pour fournir des données à partir desquelles votre modèle de ML peut apprendre. Les données d'entraînement doivent contenir la bonne réponse. L'algorithme d'apprentissage identifie des modèles dans les données d'entraînement, qui mettent en correspondance les attributs des données d'entrée avec la cible (la réponse que vous souhaitez prédire). Il fournit un modèle de ML qui capture ces modèles. Vous pouvez alors utiliser le modèle de ML pour obtenir des prédictions sur de nouvelles données pour lesquelles vous ne connaissez pas la cible.

outil

Fonction ou API qu'un agent peut invoquer pour effectuer des opérations dans des systèmes externes.

passerelle de transit

Hub de transit de réseau que vous pouvez utiliser pour relier vos VPC et vos réseaux sur site. Pour plus d'informations, voir Qu'est-ce qu'une passerelle de transit dans la AWS Transit Gateway documentation.

flux de travail basé sur jonction

Approche selon laquelle les développeurs génèrent et testent des fonctionnalités localement dans une branche de fonctionnalités, puis fusionnent ces modifications dans la branche principale. La branche principale est ensuite intégrée aux environnements de développement, de préproduction et de production, de manière séquentielle.

accès sécurisé

Accorder des autorisations à un service que vous spécifiez pour effectuer des tâches au sein de votre organisation AWS Organizations et dans ses comptes en votre nom. Le service de confiance crée un rôle lié au service dans chaque compte, lorsque ce rôle est nécessaire, pour effectuer des tâches de gestion à votre place. Pour plus d'informations, consultez la section Utilisation AWS Organizations avec d'autres AWS services dans la AWS Organizations documentation.

réglage

Pour modifier certains aspects de votre processus d'entraînement afin d'améliorer la précision du modèle de ML. Par exemple, vous pouvez entraîner le modèle de ML en générant un ensemble d'étiquetage, en ajoutant des étiquettes, puis en répétant ces étapes plusieurs fois avec différents paramètres pour optimiser le modèle.

équipe de deux pizzas

Une petite DevOps équipe que vous pouvez nourrir avec deux pizzas. Une équipe de deux pizzas garantit les meilleures opportunités de collaboration possible dans le développement de logiciels.

U

incertitude

Un concept qui fait référence à des informations imprécises, incomplètes ou inconnues susceptibles de compromettre la fiabilité des modèles de ML prédictifs. Il existe deux types d'incertitude : l'incertitude épistémique est causée par des données limitées et incomplètes, alors que l'incertitude aléatoire est causée par le bruit et le caractère aléatoire inhérents aux données.

tâches indifférenciées

Également connu sous le nom de « levage de charges lourdes », ce travail est nécessaire pour créer et exploiter une application, mais qui n'apporte pas de valeur directe à l'utilisateur final ni d'avantage concurrentiel. Les exemples de tâches indifférenciées incluent l'approvisionnement, la maintenance et la planification des capacités.

environnements supérieurs

Voir environnement.

V

mise à vide

Opération de maintenance de base de données qui implique un nettoyage après des mises à jour incrémentielles afin de récupérer de l'espace de stockage et d'améliorer les performances.

contrôle de version

Processus et outils permettant de suivre les modifications, telles que les modifications apportées au code source dans un référentiel.

Appairage de VPC

Connexion entre deux VPC qui vous permet d'acheminer le trafic à l'aide d'adresses IP privées. Pour plus d'informations, veuillez consulter la rubrique Qu'est-ce que l'appairage de VPC ? dans la documentation Amazon VPC.

vulnérabilités

Défaut logiciel ou matériel qui compromet la sécurité du système.

W

cache actif

Cache tampon qui contient les données actuelles et pertinentes fréquemment consultées. L'instance de base de données peut lire à partir du cache tampon, ce qui est plus rapide que la lecture à partir de la mémoire principale ou du disque.

données chaudes

Données rarement consultées. Lorsque vous interrogez ce type de données, des requêtes modérément lentes sont généralement acceptables.

fonction de fenêtre

Fonction SQL qui effectue un calcul sur un groupe de lignes liées d'une manière ou d'une autre à l'enregistrement en cours. Les fonctions de fenêtre sont utiles pour traiter des tâches, telles que le calcul d'une moyenne mobile ou l'accès à la valeur des lignes en fonction de la position relative de la ligne en cours.

charge de travail

Ensemble de ressources et de code qui fournit une valeur métier, par exemple une application destinée au client ou un processus de backend.

flux de travail

Groupes fonctionnels d'un projet de migration chargés d'un ensemble de tâches spécifique. Chaque flux de travail est indépendant, mais prend en charge les autres flux de travail du projet. Par exemple, le flux de travail du portefeuille est chargé de prioriser les applications, de planifier les vagues et de collecter les métadonnées de migration. Le flux de travail du portefeuille fournit ces actifs au flux de travail de migration, qui migre ensuite les serveurs et les applications.

VER

Voir écrire une fois, lire plusieurs.

WQF

Voir le cadre AWS de qualification de la charge de travail.

écrire une fois, lire plusieurs (WORM)

Modèle de stockage qui écrit les données une seule fois et empêche leur suppression ou leur modification. Les utilisateurs autorisés peuvent lire les données autant de fois que nécessaire, mais ils ne peuvent pas les modifier. Cette infrastructure de stockage de données est considérée comme immuable.

Z

exploit Zero-Day

Une attaque, généralement un logiciel malveillant, qui tire parti d'une vulnérabilité de type « jour zéro ».

vulnérabilité de type « jour zéro »

Une faille ou une vulnérabilité non atténuée dans un système de production. Les acteurs malveillants peuvent utiliser ce type de vulnérabilité pour attaquer le système. Les développeurs prennent souvent conscience de la vulnérabilité à la suite de l'attaque.

invite Zero-Shot

Fournir à un LLM des instructions pour effectuer une tâche, mais aucun exemple (plans) pouvant aider à la guider. Le LLM doit utiliser ses connaissances pré-entraînées pour gérer la tâche. L'efficacité de l'invite zéro dépend de la complexité de la tâche et de la qualité de l'invite. Voir également les instructions en quelques clics.

application zombie

Application dont l'utilisation moyenne du processeur et de la mémoire est inférieure à 5 %. Dans un projet de migration, il est courant de retirer ces applications.

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.