Création d'architectures multi-locataires pour l'IA agentic sur AWS

AWS Conseils prescriptifs

AWS Conseils prescriptifs: Création d'architectures multi-locataires pour l'IA agentic sur AWS

Table of Contents

Création d'architectures multi-locataires pour l'IA agentic sur AWS

Aaron Sempf et Tod Golding, Amazon Web Services

Juillet 2025 (historique du document)

L'IA agentic représente un changement de paradigme disruptif qui oblige les entreprises à repenser la manière de créer, de livrer et d'exploiter leurs systèmes. Le modèle agentique permet aux équipes d'explorer de nouvelles façons de décomposer les systèmes en un ou plusieurs agents qui créent de nouvelles voies, possibilités et valeurs.

La plupart des discussions agentic portent sur les outils, les cadres et les modèles utilisés pour créer et implémenter des agents. Nous devons non seulement adopter de bons outils pour créer des agents, mais également de nouveaux protocoles d'intégration, des stratégies d'authentification et des mécanismes de découverte qui peuvent servir de base aux architectures agentiques.

Alors que le nombre d'outils agentic augmente, les équipes doivent également réfléchir à la manière dont leurs agents répondent aux défis de l'architecture plus traditionnelle. L'échelle, le voisinage bruyant, la résilience, les coûts et l'efficacité opérationnelle sont des sujets fondamentaux qui doivent être évalués lors de la conception, de la création et du déploiement d'agents. Quel que soit le degré d'autonomie et d'intelligence des agents, nous devons également veiller à ce qu'ils réalisent des économies d'échelle, soient efficaces et agiles conformément aux besoins de l'entreprise.

L'objectif de ce guide est d'explorer les différentes dimensions des empreintes agentiques. Cela inclut l'examen des différents modèles de déploiement et de consommation des agents et la mise en évidence des différentes stratégies de création d'agents répondant aux objectifs architecturaux. Cela implique également d'examiner la manière dont les agents peuvent être utilisés dans un environnement multi-tenant en introduisant des structures internes qui sont généralement requises dans un environnement multi-tenant.

Public visé

Ce guide s'adresse aux architectes, aux développeurs et aux leaders technologiques qui souhaitent créer des systèmes mutualisés basés sur l'IA.

Objectifs

Ce guide vous aide à accomplir les tâches suivantes :

- Comprenez les déploiements d'agents multi-locataires, en explorant à la fois les modèles cloisonnés et groupés, et l'impact du contexte des locataires sur la mise en œuvre des agents
- Explorez la gestion des agents, y compris l'intégration, l'isolation des locataires et la gestion des ressources dans les environnements à fournisseur unique ou multiple
- Évaluez les aspects des agents multi-locataires, notamment la propriété, la surveillance et les tests des données

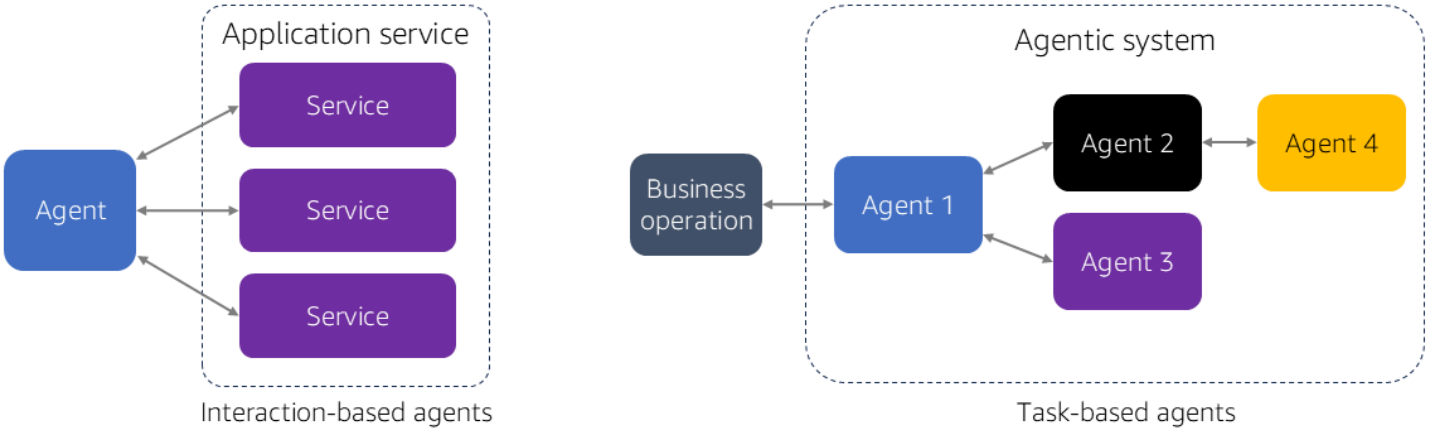
À propos de cette série de contenus

Ce guide fait partie d'une série sur l'IA agentique sur. AWS Pour plus d'informations et pour consulter les autres guides de cette série, consultez Agentic AI sur le site Web de AWS Prescriptive Guidance.

Principes fondamentaux des agents

Avant d'aborder les détails de l'architecture, nous devons décrire les différents rôles que jouent les agents, car le terme « agent » est un terme surchargé qui peut être appliqué à de nombreux cas d'utilisation. Commençons par quelques termes généraux qui peuvent aider à les classer.

Au niveau le plus externe, nous devons commencer par classer le rôle et la nature des agents. C'est un défi car il existe un large éventail de scénarios dans lesquels les agents peuvent être utilisés pour résoudre un certain nombre de problèmes. Dans le cadre de cette discussion, nous nous concentrons toutefois sur ce que signifie introduire un agent dans une application ou un système. Dans ce modèle, nous soulignons comment et où les agents peuvent le mieux enrichir l'expérience de votre système. Les options que vous choisissez influencent la manière dont vos agents sont créés, intégrés et appliqués à différents domaines et cas d'utilisation. Le schéma suivant montre deux modèles agentiques utilisés par les constructeurs.

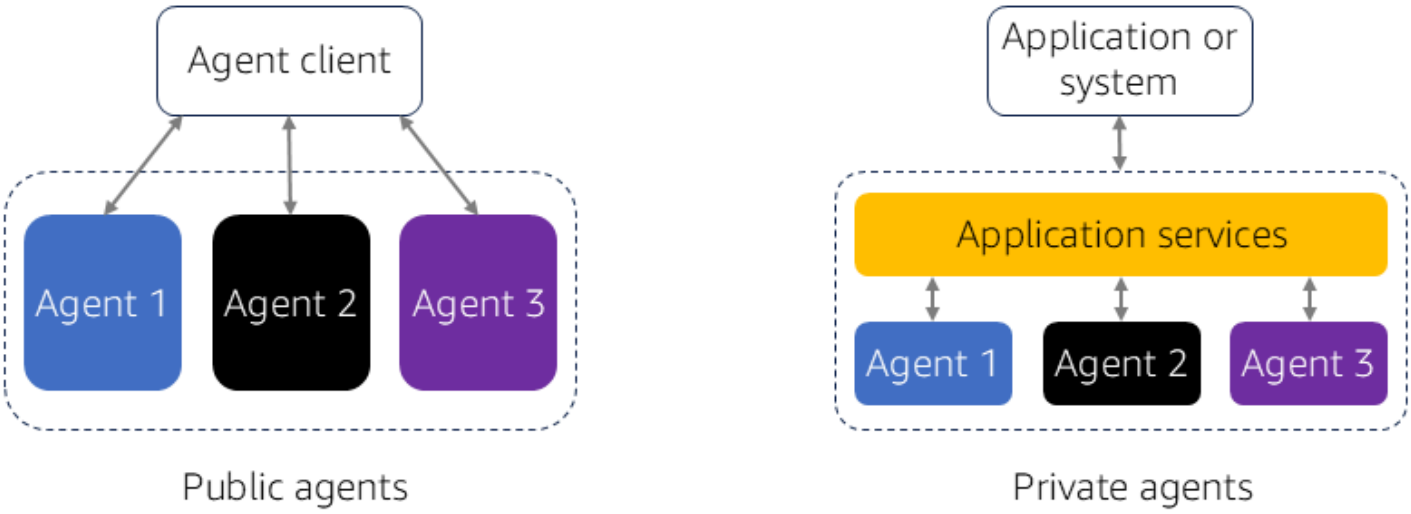


Sur le côté gauche du diagramme se trouve un agent basé sur l'interaction. Dans ce mode, un agent crée une vue sur un système existant pour orchestrer les interactions avec les services sous-jacents afin d'atteindre un objectif ou un résultat. L'essentiel est que l'agent soit ajouté à un système en tant qu'approche alternative pour piloter les fonctionnalités et les capacités du système. Imaginez, par exemple, qu'un éditeur de logiciels indépendant (ISV) dispose d'un système comptable doté d'une expérience utilisateur utilisée pour effectuer des opérations. L'agent basé sur l'interaction simplifie l'interaction avec ces fonctionnalités existantes. Il s'agit moins d'apprendre comment atteindre un objectif vague que de proposer un moyen d'orchestrer des parcours connus.

En revanche, le système basé sur les tâches sur le côté droit du diagramme représente une approche différente. Les agents de ce système utilisent leurs connaissances et leurs capacités pour apprendre à accomplir des tâches et à obtenir des résultats commerciaux. On pourrait soutenir

que les deux modèles produisent des résultats commerciaux, mais un modèle basé sur les tâches repose sur les agents eux-mêmes pour déterminer comment atteindre un résultat. Ces agents sont moins déterministes et s'appuient plutôt sur leur capacité à apprendre et à évoluer. En revanche, les agents basés sur l'interaction sont principalement conçus pour orchestrer un ensemble de capacités connues. Ces différences affectent la manière dont vous créez, définissez et intégrez les agents pour soutenir votre entreprise.

Nous avons également besoin de termes qui décrivent comment et où nous déployons les agents. L'emplacement d'un agent dans le périmètre de votre système peut influencer la manière dont celui-ci est construit, défini et sécurisé. Le schéma suivant décrit deux modèles distincts qui pourraient être appliqués aux agents.

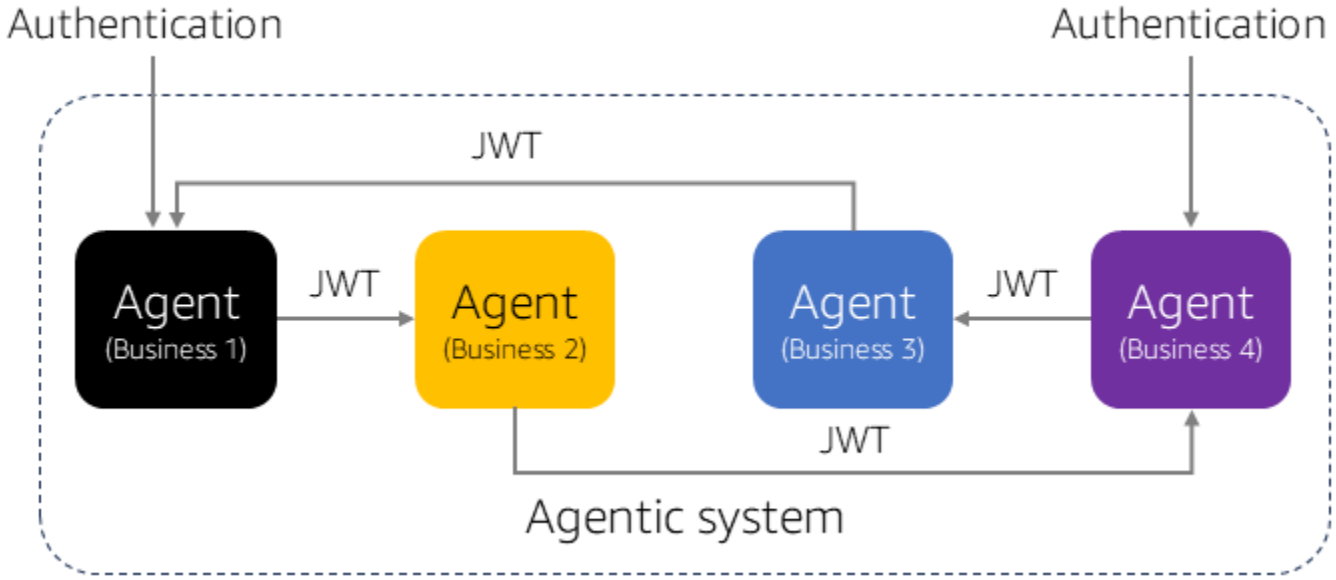


Sur le côté gauche du schéma se trouve un système de déploiement avec trois agents différents. Les agents sont exposés à des clients externes qui peuvent être d'autres agents ou applications. Dans ce modèle, les agents sont appelés agents publics.

En revanche, le schéma de droite montre les agents participant à la mise en œuvre de la solution. Dans ce cas, il existe une série de services d'application utilisés par les utilisateurs ou les systèmes. Ces utilisateurs interagissent avec l'application sans se rendre compte que les agents font partie de l'expérience. Les agents sont ensuite invoqués et orchestrés par les services du système sous-jacent. Les agents déployés de cette manière sont appelés agents privés.

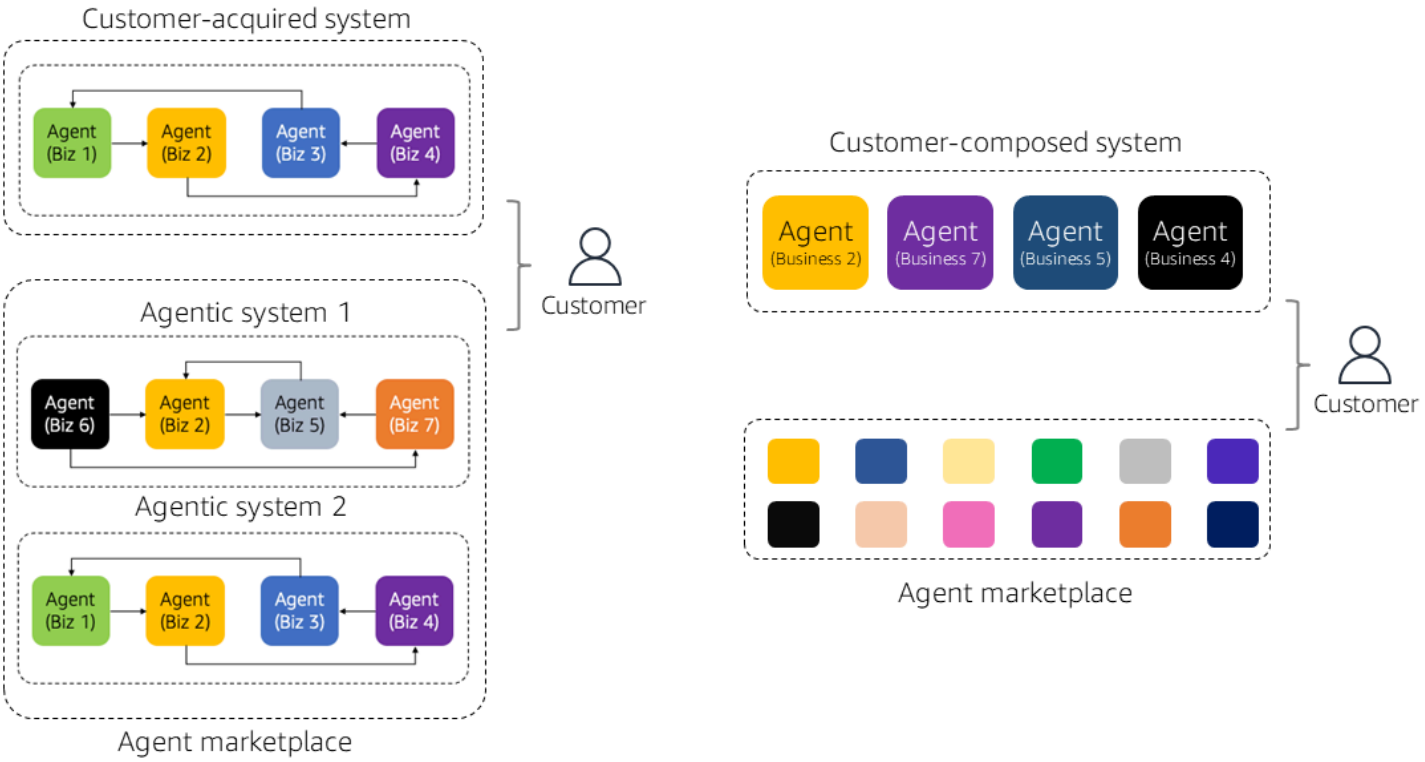
Une grande partie de la valeur d'un agent se concentre sur le modèle public dans lequel les fournisseurs peuvent publier leurs agents dans le but de les intégrer à d'autres agents tiers. Les agents feraient alors partie d'un maillage ou d'un réseau de services interconnectés qui, collectivement, sont en mesure de répondre à de nombreux cas d'utilisation. Bien que ces agents

puissent être utilisés dans de nombreux domaines, le cas d' business-to-businessutilisation est naturel. Le schéma suivant fournit une vue conceptualisée de ce à quoi peut ressembler l'assemblage d'un agent de recouvrement capable de résoudre un problème spécifique.



Le diagramme montre quatre agents commerciaux qui travaillent ensemble pour atteindre un ensemble d'objectifs. Lorsque les agents sont composés de cette façon, ils représentent un système agentique, et il existe de nombreuses variantes de tels systèmes. Il peut s'agir d'un ensemble préemballé d'agents collaborateurs généralement consommés en une seule unité. Le système peut également être assemblé dynamiquement par les clients qui souhaitent sélectionner une combinaison d'agents répondant le mieux à leurs besoins.

Les deux approches offrent des voies viables pour l'intégration des agents. Certains agents sont conçus dans l'espoir d'être intégrés dans des systèmes spécifiques leur permettant de maximiser leur valeur, leur portée et leur impact. Cette notion de système agentique soulève également des questions sur la manière dont les agents sont acquis, et il pourrait y avoir de nombreuses façons de résoudre ce problème. Le schéma suivant fournit des exemples de la manière dont ces agents et systèmes peuvent être créés par le biais d'expériences transactionnelles.

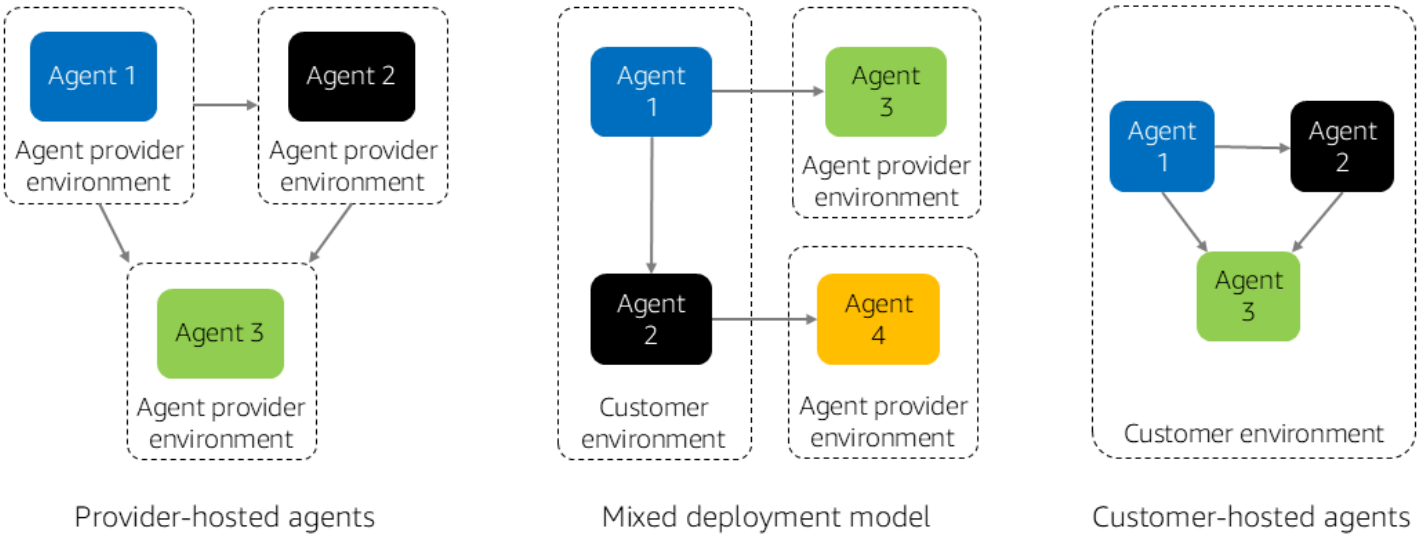


Deux exemples d'expériences de marché sont présentés. Sur le côté gauche, une place de marché est utilisée pour acquérir des systèmes préemballés. Dans ce scénario, le marché découvre et intègre des systèmes répondant à des objectifs plus larges nécessitant l'intégration et l'orchestration de plusieurs agents.

L'exemple sur le côté droit montre un marché où les agents sont découverts et composés en systèmes agentiques. Dans ce scénario, les clients peuvent créer n'importe quel système d'agents compatibles et intégrés pour répondre à leurs besoins. La capacité d'assembler les agents de cette manière dépend du modèle de compatibilité et des exigences d'intégration de chaque agent.

Considérations concernant l'hébergement des agents

Maintenant que vous avez une idée des concepts agentiques généraux, voyons ce que signifie héberger et gérer ces agents. Nous devons réfléchir à la manière et à l'endroit où les calculs s'exécutent, à leur évolutivité, à leur fonctionnement et à leur gestion. Dans le même temps, certains modèles que nous nous attendons à voir en tant qu'agents sont plus largement appliqués et adoptés. Le schéma suivant montre un exemple de permutations probables.



Trois stratégies distinctes sont représentées ici. Sur le côté gauche du diagramme, vous voyez un modèle dans lequel nos agents sont hébergés, dimensionnés et gérés dans les environnements de chaque fournisseur d'agents. Ces agents sont publiés et consommés en tant que services, fonctionnant selon ce que l'on appelle un modèle d'agent en tant que service (AaaS). Sur le côté droit se trouve un modèle dans lequel les agents d'un fournisseur sont tous hébergés dans un environnement client dédié.

Au centre du schéma se trouve un modèle de déploiement mixte qui combine ces deux stratégies, en hébergeant certains agents localement dans l'environnement du client et en interagissant avec certains agents hébergés à distance dans l'environnement d'un fournisseur.

Une quatrième option (non illustrée) pourrait être celle où les agents sont conçus sous la forme de services à faible code ou sans code, mis à l'échelle et gérés par les services d'infrastructure des agents. Nous ne les aborderons pas en détail car l'architecture et l'hébergement des agents gérés sont principalement dictés par l'organisation propriétaire des services.

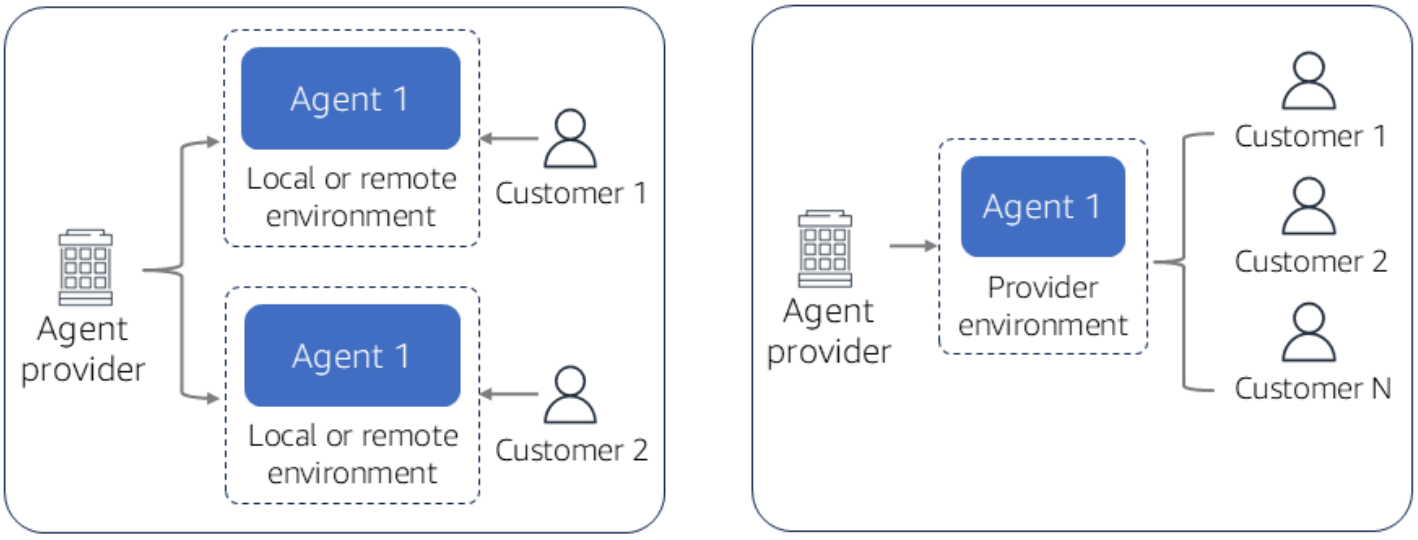
Vous pouvez imaginer l'éventail de facteurs susceptibles d'influencer l'adoption de l'un de ces modèles. Les contraintes de conformité, de réglementation et de sécurité, par exemple, peuvent pousser quelqu'un à se tourner vers des agents hébergés par le client. L'évolutivité, l'agilité et l'efficacité pourraient pousser les entreprises à adopter davantage le modèle AaaS.

Le concept clé ici est que les agents peuvent être déployés et hébergés de nombreuses manières. C'est à vous de déterminer la meilleure façon d'utiliser les agents. L'encombrement, la sécurité et le déploiement, entre autres facteurs, influent de manière significative sur la façon dont vous abordez les agents de construction et d'exploitation. Les agents privés et publics, par exemple, peuvent avoir des conceptions et des cycles de vie de publication différents.

Les agents rencontrent la multilocation

Il est facile de considérer les agents comme des éléments de base, les agents étant considérés comme une série de composants autonomes assemblés pour répondre aux besoins d'un domaine ou d'un problème commercial spécifique. Là où cela devient plus intéressant, c'est lorsque nous commençons à réfléchir à la manière dont ces agents sont conditionnés et consommés par les fournisseurs. À bien des égards, un agent devient une source de coûts et de revenus pour une entreprise. Les fournisseurs d'agents doivent tenir compte des différentes personnes qui consomment leurs services, du profil de consommation de ces personnes et des stratégies de monétisation qui permettent aux fournisseurs d'agents de créer des modèles de tarification et de hiérarchisation adaptés aux consommateurs.

Les fournisseurs d'agents peuvent prendre en charge plusieurs modèles de déploiement de leurs agents afin de répondre aux besoins des clients. Le schéma suivant présente une vue conceptuelle des deux principaux modèles de déploiement d'agents.



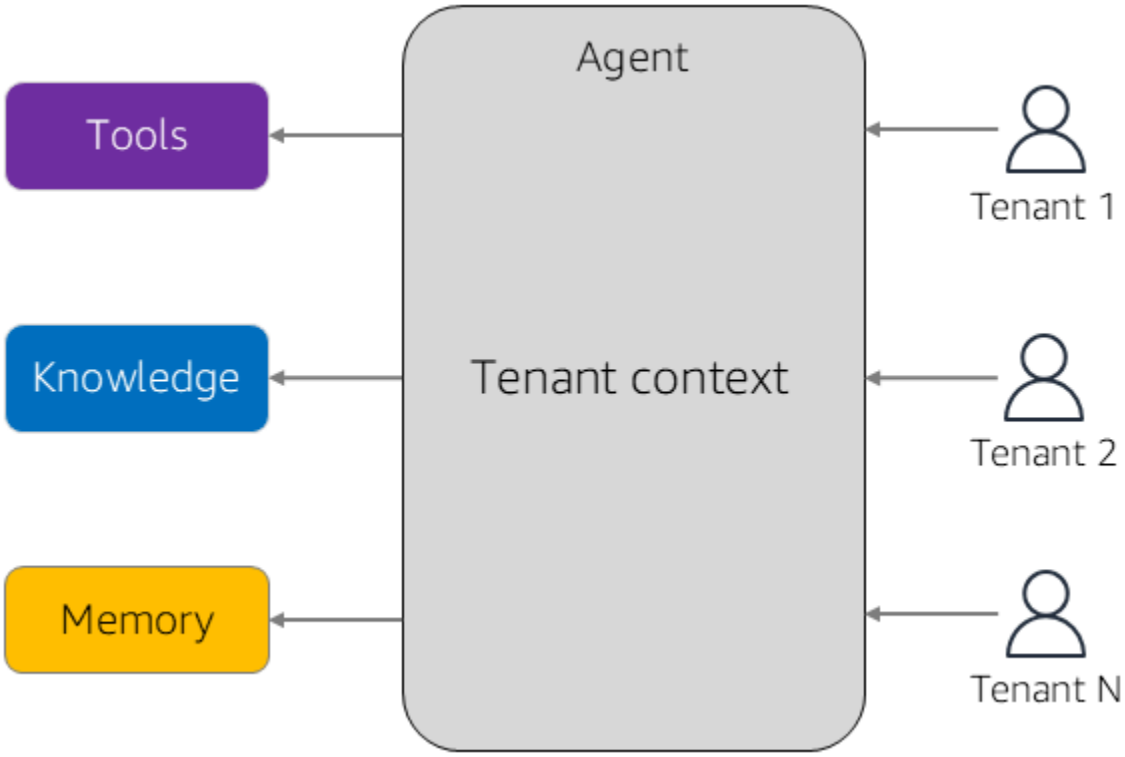
Le côté gauche du schéma montre le modèle d'agent dédié au client. Un fournisseur d'agent crée un agent en déployant une instance d'agent distincte pour chaque client intégré. Avec cette approche, les capacités de l'agent et sa capacité à acquérir des connaissances seraient limitées à l'environnement d'un client donné. Cela finit par représenter une expérience par client qui hérite de certaines des complexités et des avantages liés à la prise en charge d'environnements clients dédiés.

En revanche, le schéma sur le côté droit du diagramme comporte un seul agent déployé dans l'environnement du fournisseur. L'agent traite les demandes de plusieurs clients, évolue et apprend

en fonction de l'expérience collective de tous les clients. Chaque nouveau client ajouté représentera simplement un autre client valide de l'agent. L'agent fonctionne comme un modèle d'agent en tant que service (AaaS), en utilisant des structures partagées pour répondre aux besoins du client. Dans les deux cas, les consommateurs d'agents peuvent être des applications, des systèmes ou même d'autres agents.

Il existe deux manières d'envisager le modèle AaaS. Le modèle ci-dessus offre la même expérience à tous les clients. Cela signifie que le personnel interne de l'agent n'inclura aucun niveau de spécialisation tenant compte du contexte du client demandeur. En général, pour ce mode, l'hypothèse est que la nature du champ d'action, des objectifs et de la valeur d'un agent est centrée sur un ensemble partagé de ressources, de connaissances et de résultats qui sont appliqués universellement à tous les clients.

L'approche alternative de l'AaaS est celle où le contexte des clients influence l'expérience et la mise en œuvre de l'agent. Le schéma suivant fournit une vue conceptuelle de l'empreinte d'un agent AaaS dans ce contexte.



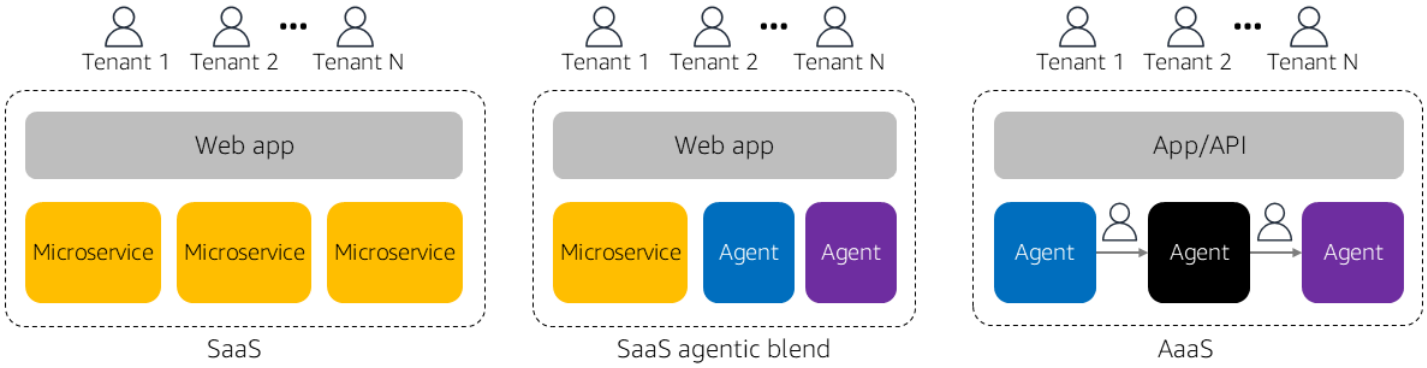
Dans cette vue AaaS, l'origine et le contexte des demandes entrantes affectent de manière significative l'empreinte de l'agent. Les ressources, les actions et les outils qui font partie de l'implémentation sous-jacente de l'agent peuvent varier en fonction de chaque demande entrante du locataire. La valeur d'un agent est liée à sa capacité à utiliser le contexte du locataire pour parvenir

à des actions et à des résultats influencés par l'état, les connaissances et d'autres facteurs du locataire. Certaines demandes peuvent produire un résultat unique pour le locataire, tandis que d'autres peuvent conduire à des résultats plus personnalisés par locataire. Cela ajoute une nouvelle dimension à la capacité d'apprentissage de l'agent, notamment en étant plus contextuel et en acquérant et en appliquant des connaissances qui améliorent les résultats ciblés.

Pour les fournisseurs, le modèle AaaS présente de nombreux avantages. Lorsque plusieurs clients utilisent un seul agent, le fournisseur a de meilleures chances de réaliser des économies d'échelle, de renforcer l'efficacité opérationnelle, de contrôler les coûts et de créer une expérience de gestion unifiée. Cela a le potentiel d'améliorer l'agilité, l'innovation et la croissance du secteur des agents.

Ces qualités se recoupent avec les mêmes principes qui sous-tendent l'adoption du modèle SaaS (logiciel en tant que service). Essentiellement, le modèle AaaS est conçu comme un service mutualisé qui hérite de nombreux attributs d'échelle, de résilience, d'isolation, d'intégration et opérationnels identiques à ceux d'un environnement SaaS. À bien des égards, l'expérience AaaS s'inspire largement des stratégies et pratiques utilisées par les fournisseurs de SaaS, mais il est raisonnable de séparer ces termes. Pour nos besoins, l'accent est principalement mis sur les implications liées aux agents de construction et d'exploitation qui ont besoin d'un soutien multilocataire.

Pour un système capable de traiter tous les utilisateurs de la même manière et ne nécessitant pas la gestion de données persistantes, sensibles ou spécifiques au client, la notion de location affecterait très peu ses agents. Pour les systèmes censés servir plusieurs clients tout en préservant l'isolation des données, la personnalisation et la connaissance du contexte, la prise en charge de plusieurs locataires peut être un élément essentiel de la conception, de la stratégie et des objectifs d'un agent. Le schéma suivant montre comment la mutualisation peut être utilisée dans des environnements agentiques.



Sur le côté gauche de ce schéma se trouve une architecture multi-locataires classique. Il inclut une application Web et une série de microservices qui mettent en œuvre une logique métier.

Plusieurs locataires utilisent l'infrastructure partagée de cet environnement, qui évolue pour répondre aux charges de travail changeantes d'une population de locataires en constante évolution. L'environnement est exploité et géré par le biais d'une seule vitre pour tous les locataires.

Imaginez comment ce modèle mental correspond à l'agent sur le côté droit de ce diagramme. Un agent exécute un modèle AaaS utilisé par un ou plusieurs locataires. Les agents peuvent provenir de plusieurs fournisseurs avec un contexte de locataire circulant entre eux, car une seule instance d'un agent doit traiter les demandes de plusieurs locataires.

L'exemple au milieu de ce diagramme est un modèle hybride dans lequel les agents font partie de l'expérience globale du SaaS. Certaines parties du système sont mises en œuvre selon un modèle plus traditionnel, tandis que d'autres font appel à des agents. Ce schéma est susceptible d'être courant pour de nombreuses offres SaaS, en particulier pour les organisations qui passent à une expérience agentique. Il est courant que ce modèle persiste, car tous les systèmes ne sont pas fournis en tant que pur AaaS. Notez également que la mutualisation s'applique toujours aux agents du modèle. Bien que les agents puissent être intégrés dans un système, ils peuvent toujours traiter les demandes de plusieurs locataires.

Il est naturel de se demander si la mutualisation est vraiment importante. Vous pourriez faire valoir qu'un agent traite les demandes, de sorte que le soutien à la location peut avoir peu d'effet. Mais au fur et à mesure que nous étudions les implications agentiques du multi-tenant, la location peut affecter directement la manière dont les agents influencent la manière dont les outils, la mémoire, les données et les autres composants de l'agent sont accessibles, déployés et configurés pour prendre en charge les locataires individuels. La location influence également la manière dont la mise à l'échelle, la limitation, la tarification, la hiérarchisation et d'autres aspects commerciaux s'appliquent à l'architecture de votre agent.

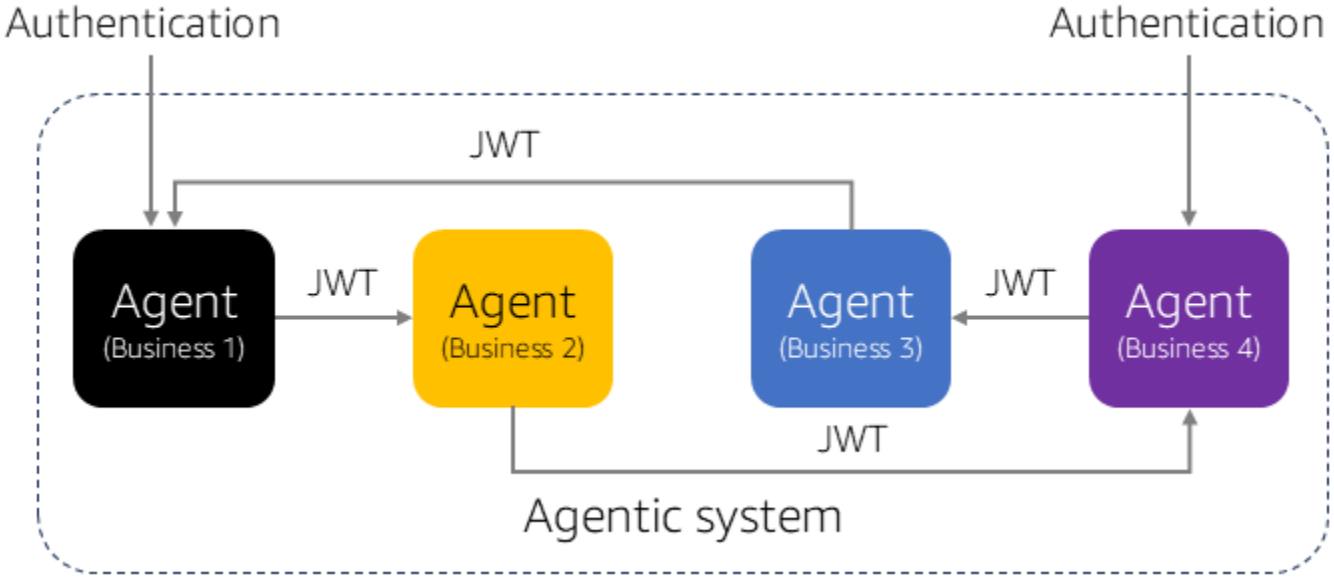
L'un des points à retenir est qu'il existe des cas d'utilisation d'agentic qui nécessitent une assistance multi-locataires. Le défi consiste à déterminer comment la mutualisation façonne la conception globale et l'architecture de votre expérience agentique. Pour certains agents, le support mutualisé représente une capacité de différenciation, permettant aux agents d'appliquer un contexte spécifique au locataire aux agents qui fournissent des résultats ciblés.

Dans les sections suivantes, vous verrez en quoi la terminologie et les modèles de conception que nous créons pour décrire les architectures SaaS multi-locataires seront utiles. Ces concepts peuvent être adoptés par le modèle AaaS en empruntant des aspects utiles, qui introduisent de nouveaux concepts spécifiques aux agents là où ils sont nécessaires.

Identité, contexte du locataire et systèmes agentiques

Ajouter le contexte du locataire à des agents individuels n'est pas particulièrement difficile. Dans de nombreux cas, les équipes peuvent s'appuyer sur des mécanismes classiques qui lient les utilisateurs et les systèmes aux locataires et transmettent des jetons tenant compte des locataires aux agents. Cela est pertinent lorsque nous examinons la manière dont le contexte et l'identité des locataires prennent en charge plusieurs agents. Dans ce modèle, les locataires doivent être liés à une identité qui couvre tous les agents collaborateurs.

En général, le domaine agentique nécessite un modèle d'identité plus transversal qui s'aligne sur les besoins actuels et émergents des systèmes agentiques. Les fournisseurs d'agents ont besoin de mécanismes d'identité qui prennent en charge les modèles uniques de sécurité, de conformité et d'autorisation fournis avec les systèmes d'agentic d'exploitation. Cela est particulièrement difficile dans les environnements où les systèmes sont composés par des clients ou d'autres agents. Chaque agent intégré doit associer son identité et son contexte de locataire aux interactions avec les agents. Le schéma suivant met en évidence les défis potentiels liés à l'identité et au contexte des locataires qui font partie des interactions agent-to-agent (a2a).



Ce schéma montre une série d'agents créés par les fournisseurs interagissant dans le cadre du système agentic que nous avons abordé. Il est désormais modernisé en fonction de l'identité et du contexte du locataire. Ce scénario est un exemple de système agentique qui prend en charge plusieurs points d'entrée. Nous partons du principe que chaque agent de ce système a besoin de son propre mécanisme d'authentification pour attribuer le système ou l'utilisateur à un locataire donné.

Lorsque ces agents interagissent, le contexte du locataire est transmis à un jeton Web JSON (JWT) qui sera utilisé pour autoriser l'accès et injecter le contexte du locataire dans l'agent.

Conceptuellement, la principale différence avec ce scénario est que les agents se déploient et opèrent de manière indépendante, ce qui signifie que chaque agent doit être en mesure de résoudre son identité et d'autoriser l'accès. L'essentiel est que son identité doit avoir une certaine capacité distribuée pour répondre aux besoins du système agentique au sens large. Il doit également y avoir un alignement sur la manière dont les agents partagent le contexte des locataires.

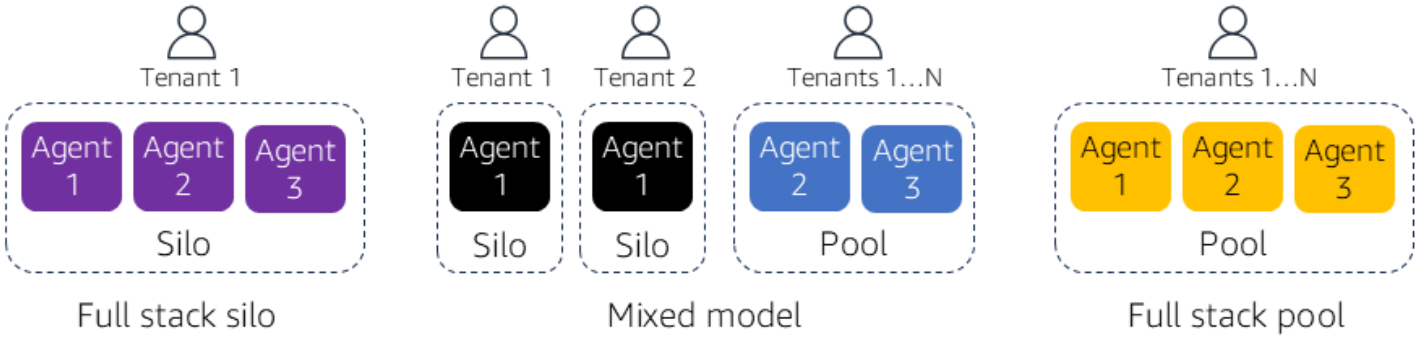
Appliquer la valeur commerciale du SaaS à l'AAaS

En général, lorsque nous examinons l'utilisation d'un système dans un as-a-service modèle, nous prenons en compte la nature de l'expérience et la manière dont son empreinte technique et opérationnelle améliore les résultats commerciaux. Lors de l'adoption du SaaS, par exemple, les entreprises utilisent les économies d'échelle, l'efficacité opérationnelle, les profils de coûts et l'agilité pour stimuler la croissance, les marges et l'innovation.

Les agents fournis sous forme d'AaaS sont susceptibles de viser des résultats commerciaux similaires. En prenant en charge plusieurs locataires, un agent peut aligner la consommation de ressources sur les activités des locataires. Cela permet de réaliser des économies d'échelle, comme c'est le cas avec les environnements SaaS traditionnels. L'AaaS permet également aux entreprises de gérer, d'exploiter et de déployer des agents de manière à permettre des mises à disposition fréquentes et à accroître l'agilité des fournisseurs d'agents. L'essentiel est que le modèle AaaS ne dépend pas de la technologie. Il crée et met en œuvre des stratégies commerciales qui favorisent la croissance, rationalisent l'adoption et simplifient les opérations.

Modèles de déploiement d'agents

Dans le cadre d'une expérience AaaS de base, un fournisseur peut déployer des agents selon différents modèles. De nombreux facteurs influencent la manière dont les agents sont déployés pour répondre aux besoins des clients, de performance, de conformité, de géographie et de sécurité. Les différentes stratégies de déploiement influent sur la façon dont un agent est conçu, mis en œuvre et utilisé. C'est ici que nous pouvons introduire des termes classiques de mutualisation pour étiqueter différentes stratégies de déploiement. Le schéma suivant montre les différentes permutations pour le déploiement d'agents dans un environnement AaaS.



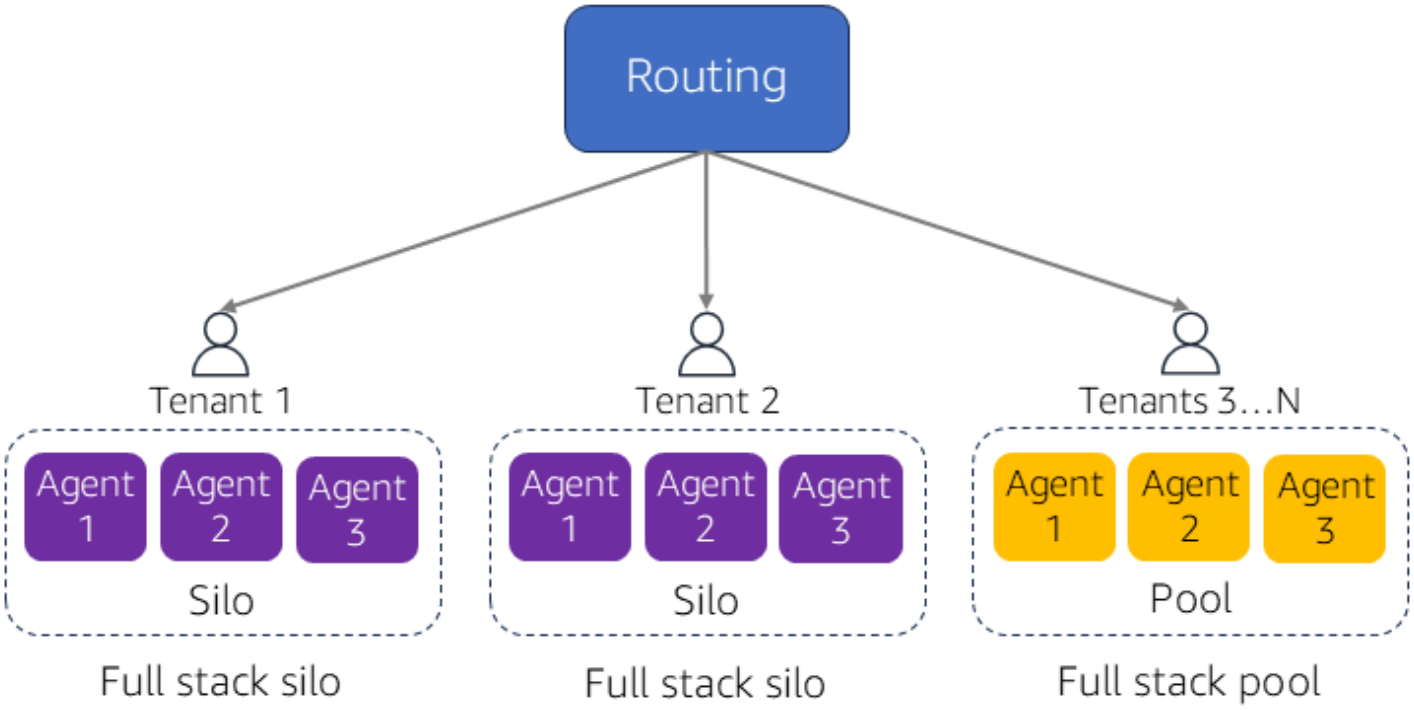
Ce schéma représente trois modes de déploiement de l'agent. Sur le côté gauche se trouve un modèle cloisonné, dans lequel chaque locataire bénéficie d'une expérience totalement isolée et d'un ensemble d'agents dédiés. Dans ce scénario, les agents ne partagent pas les environnements de calcul, de ressources ou d'exécution entre les locataires.

L'exemple du milieu illustre un modèle hybride, dans lequel les locataires utilisent une combinaison d'agents cloisonnés et groupés. Par exemple, l'agent 1 est déployé en mode cloisonné (chaque locataire reçoit une instance dédiée) tandis que les agents 2 et 3 fonctionnent selon un modèle groupé, partageant les ressources entre les locataires.

Sur le côté droit se trouve un modèle entièrement mutualisé, dans lequel tous les agents sont partagés entre les locataires, offrant un déploiement multilocataire classique. Dans ce scénario, les locataires tirent parti d'une infrastructure de calcul, de mémoire et de service commune pour exécuter les agents.

L'idée est que les agents peuvent opérer selon différents modèles de déploiement, avec des ressources informatiques et dépendantes dédiées (cloisonnées) ou partagées (mises en commun) entre les locataires. Ces stratégies de déploiement ne s'excluent pas mutuellement. Les services des agents répondent souvent à un large éventail de besoins des clients, combinant les deux modèles

pour équilibrer les performances, l'isolation, les coûts et l'évolutivité. Le schéma suivant montre un système agentic qui prend en charge plusieurs configurations de déploiement dans le même environnement opérationnel.

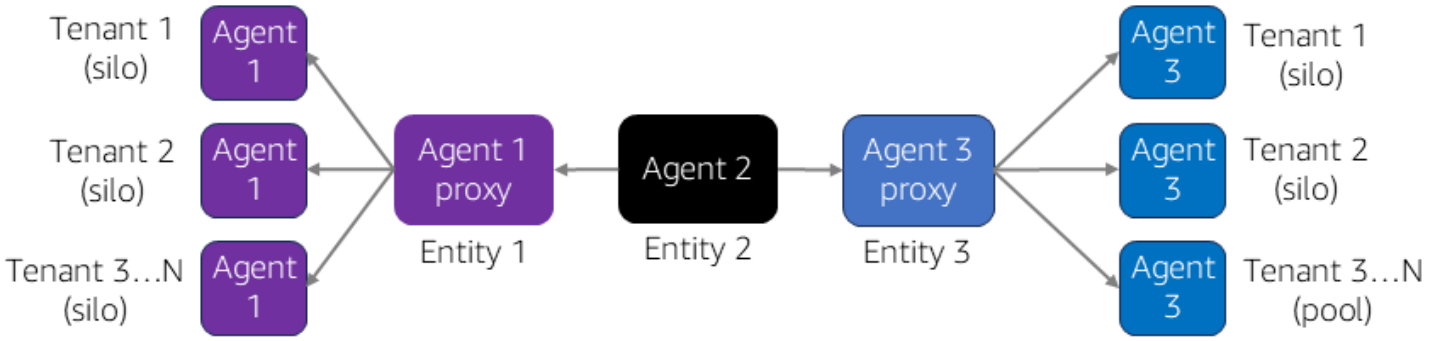


Dans ce schéma, un fournisseur d'agents dispose de trois agents déployés via un agent en tant que service (AaaS). Ils prennent en charge deux types de locataires. Sur le côté gauche, deux locataires ont des exigences de conformité et de performance auxquelles ils répondent par le biais d'un modèle de silo complet. Le locataire restant sur le côté droit fonctionne selon un modèle groupé dans lequel les locataires partagent les ressources.

Si l'objectif est l'agilité et l'efficacité opérationnelle, essayez de limiter les effets associés à la prise en charge de modèles de déploiement par locataire. Cela implique de mettre en place des mécanismes de routage et d'autres mécanismes d'expérience qui permettent aux agents d'être gérés, exploités et déployés via un seul écran.

Si vous créez un agent dans un environnement à code faible ou nul, il n'y aura aucune notion d'agents cloisonnés ou groupés. Au lieu de cela, les agents peuvent être entièrement gérés par un autre agent. Les modèles cloisonnés et groupés s'appliquent davantage aux environnements dans lesquels une organisation contrôle la construction et l'empreinte de l'agent. Dans ce cas, les équipes doivent réfléchir au modèle de déploiement à prendre en charge.

À première vue, ces modèles de déploiement n'affectent pas directement le fonctionnement d'un agent dans un système plus large. Il se peut qu'un agent n'ait aucune connaissance directe des autres agents déployés dans un modèle en silo ou groupé. Au lieu de cela, ces stratégies de déploiement peuvent être mises en œuvre dans le cadre d'une structure de routage au sein d'un environnement. Le schéma suivant montre un exemple de la manière dont les modèles cloisonnés et groupés peuvent être mis en œuvre à l'aide d'une stratégie de routage.



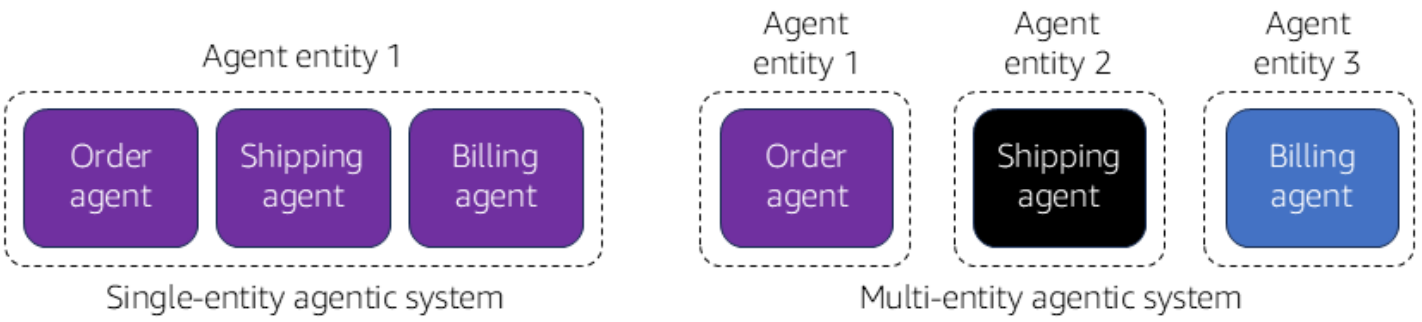
Cet exemple inclut trois agents de trois fournisseurs différents. Chaque fournisseur d'agent a la possibilité de mettre en œuvre sa propre stratégie de déploiement. Par exemple, l'agent 1 utilise un proxy pour distribuer les demandes entrantes à un ensemble d'agents locataires cloisonnés. L'agent 2 ne nécessite aucun routage et prend en charge toutes les demandes des locataires via un seul agent groupé. L'agent 3 est un modèle de déploiement hybride dans lequel certains locataires sont cloisonnés et d'autres regroupés.

Si et comment vous choisissez de prendre en charge ces modèles de déploiement, cela dépend de la nature de votre solution. Il se peut que vous n'ayez pas besoin de prendre en charge l'un ou l'autre modèle. Il se peut toutefois que vous deviez envisager de soutenir cette stratégie, par exemple en matière de conformité, de voisinage bruyant, de performance ou de hiérarchisation.

Présentation et application du contexte du locataire

Si nous créons des agents qui prennent en charge la mutualisation, nous devons commencer par réfléchir à la manière de configurer le contexte du locataire, qui sera utilisé pour appliquer des politiques, des stratégies et des mécanismes spécifiques au locataire dans le cadre de la mise en œuvre de l'agent.

Au niveau le plus élémentaire, vous pouvez introduire le contexte du locataire dans les agents grâce aux outils et mécanismes courants que nous utilisons dans les architectures multi-locataires classiques. Cela peut se faire par le biais d'une clé d' OAuthAPI ou de divers autres mécanismes de validation. De nombreux exemples mettent l'accent sur la résolution d'un système ou d'un utilisateur authentifié par une clé de jeton Web JSON (JWT) contenant le contexte du locataire. Le JWT est ensuite propagé dans le système. Cela devient plus intéressant lorsque nous examinons comment composer des systèmes agentiques. Le schéma suivant montre un exemple de deux types d'environnements agentiques.



Dans ce schéma, le modèle sur le côté gauche représente un système agentique dans lequel tous les agents sont détenus, gérés et hébergés par une seule entité. Lorsque vous avez le contrôle total de l'expérience, vous pouvez utiliser des stratégies classiques pour faire passer les locataires à chaque agent.

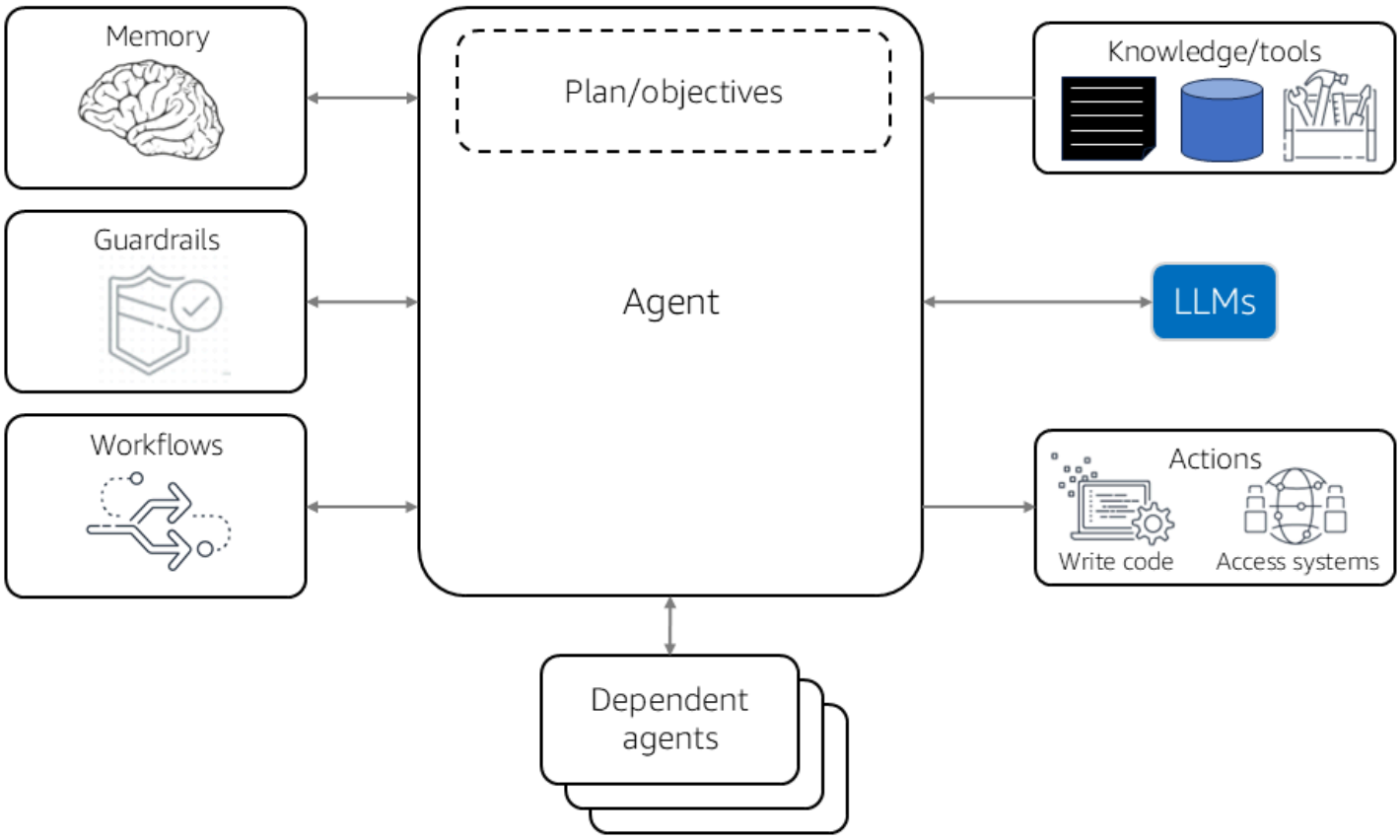
Le modèle de droite, qui peut être plus courant, représente un système d'agents couvrant plusieurs entités. Les agents sont conçus, gérés et exploités de manière indépendante, de sorte qu'ils disposent chacun de leurs propres schémas d'authentification et d'autorisation. Le défi ici est que nous avons besoin d'un moyen universel de résoudre et de partager le contexte des locataires entre ces agents. Cela repose sur un modèle plus distribué dans lequel chaque agent doit être en mesure d'authentifier les systèmes ou les utilisateurs et de les transmettre à un locataire conformément aux mécanismes appliqués.

Former des agents conscients des besoins des locataires

La mutualisation influence la manière dont nous mettons en œuvre des agents individuels. Lorsqu'un agent traite les demandes, considérez comment le contexte du locataire affecte la manière dont un agent accède aux données, prend des décisions et invoque des actions. Pour mieux comprendre comment et où la mutualisation affecte le profil de votre agent, déterminez d'abord comment les constructions peuvent faire partie d'un agent.

Le défi réside dans le fait que le champ d'application, la nature et la conception des agents sont loin d'être concrets, car les fournisseurs font leurs propres choix quant à la conception d'une expérience d'agent. En fin de compte, l'intérêt d'un agent est qu'il s'agit d'un service d'apprentissage autonome qui peut accéder à une gamme d'outils, de sources de données et de mémoire afin de déterminer la meilleure façon de résoudre une tâche.

Il est moins important de savoir exactement quelles stratégies et quels modèles un agent utilise. Dans un modèle multi-tenant, il est plus important d'identifier la manière dont les différentes parties d'un agent sont configurées, consultées et appliquées. Envisagez un environnement d'agents potentiels qui s'appuie sur une série de ressources et de mécanismes pour atteindre ses objectifs. Le schéma suivant montre un exemple d'un tel agent.

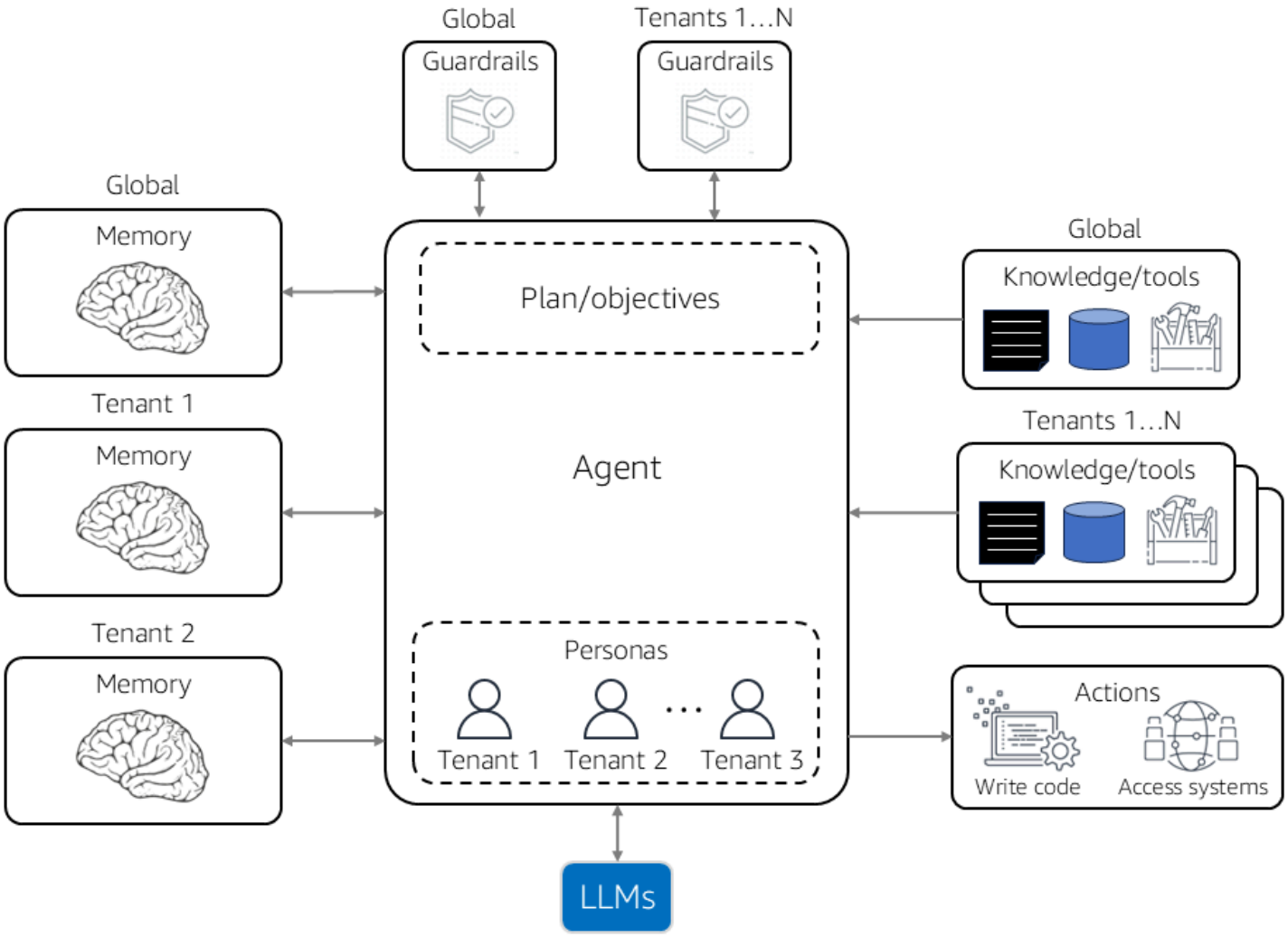


Ce diagramme représente une gamme complète de possibilités agentiques, présentant divers outils et mécanismes pouvant être combinés pour atteindre un objectif. Sur le côté gauche du diagramme, notez comment un agent dépend de la mémoire dans son contexte, des garde-fous pour définir les politiques qui guident ses activités et des flux de travail destinés à des tâches spécifiques. Certains pourraient faire valoir que les flux de travail ne devraient pas être inclus dans ce contexte, mais il existe des scénarios dans lesquels les flux de travail font partie intégrante d'une expérience agentique.

Le côté droit du diagramme montre comment des entrées telles que les connaissances et les outils peuvent fournir des informations et un contexte supplémentaires qui améliorent les capacités de l'agent. L'agent produit ensuite des actions, telles que l'écriture de code ou l'accès à des systèmes. Le bas du diagramme montre comment les agents dépendent d'un ou de plusieurs agents internes ou tiers qui peuvent être orchestrés dans le cadre d'un système plus large.

Nous pouvons maintenant réfléchir à ce que signifie introduire la mutualisation. La location nous oblige à réfléchir à la manière et à l'endroit où un agent introduit des stratégies et des mécanismes qui dictent les comportements et les actions. Cela ajoute une autre dimension à la façon dont nous envisageons les agents en termes de connaissances, d'apprentissage, d'outils et de mémoire.

Voyons maintenant comment modifier ce modèle pour prendre en charge la mutualisation. Le schéma suivant montre un exemple de modèle multi-agents.



Dans ce schéma, nous introduisons les personnalités des locataires destinées à façonner la manière dont un agent intègre le contexte des locataires. Par exemple, sur le côté gauche du schéma, la mémoire de l'agent est modifiée pour prendre en charge la mémoire spécifique au locataire. Il en va de même sur le côté droit du schéma, où l'agent soutient les connaissances et les outils spécifiques au locataire. Le même support est également appliqué aux rambardes.

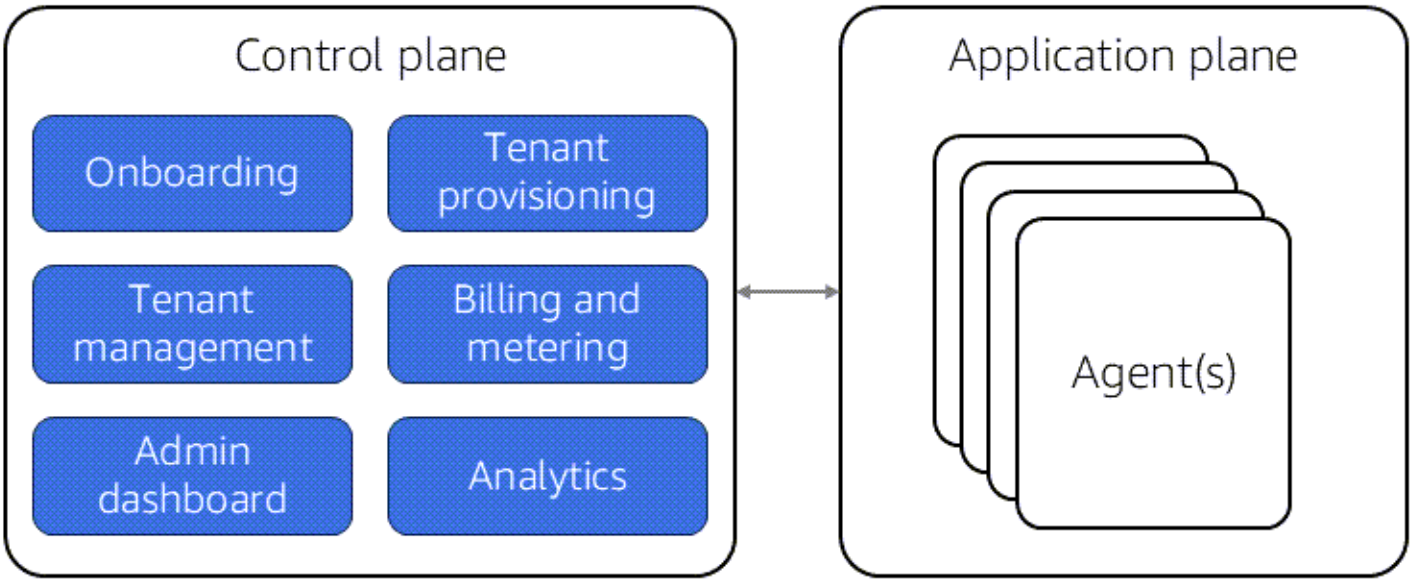
Il peut s'agir d'un exemple extrême, car tous les aspects d'un agent multi-tenant ne nécessitent pas de ressources par locataire. Le fait est que vous devriez réfléchir à la manière dont l'adaptation de votre agent à des locataires spécifiques peut améliorer son efficacité. Cette approche permet à votre agent d'accroître son impact et sa valeur, de fournir un contexte plus pertinent dans ses réponses et de développer des capacités spécialisées. L'agent sera alors en mesure d'apprendre, de s'adapter et d'exécuter des tâches parfaitement adaptées à différentes personnalités.

L'idée principale est que le contexte du locataire influe directement sur la façon dont vous créez des agents. Il peut également façonner les interactions des locataires avec des entités externes, y compris d'autres agents. La création d'un agent multi-locataires présente des défis traditionnels tels que les voisins bruyants, l'isolement des locataires, la hiérarchisation, la limitation et la gestion des coûts. La conception et l'architecture de votre agent doivent tenir compte de ces concepts fondamentaux du multi-tenant, que nous explorerons dans la section suivante.

Utilisation de plans de contrôle dans des environnements agentiques

Les meilleures pratiques multi-locataires divisent souvent les implémentations en deux parties distinctes : un plan de contrôle et un plan d'application. Le plan de contrôle fournit un écran unique permettant d'accéder aux mécanismes opérationnels, de gestion et d'orchestration qui concernent tous les locataires de l'environnement. Le plan applicatif est l'endroit où résident la logique métier, les fonctionnalités et les fonctionnalités.

Cette division des responsabilités s'applique également aux modèles agentiques. Un agent mutualisé nécessite un certain degré de centralisation de la gestion, des opérations et des informations, et il est logique de répondre en permanence à ces besoins par le biais d'un plan de contrôle. Le schéma suivant montre une vue conceptuelle de la façon dont ces plans sont divisés dans un environnement d'agent en tant que service (AaaS).

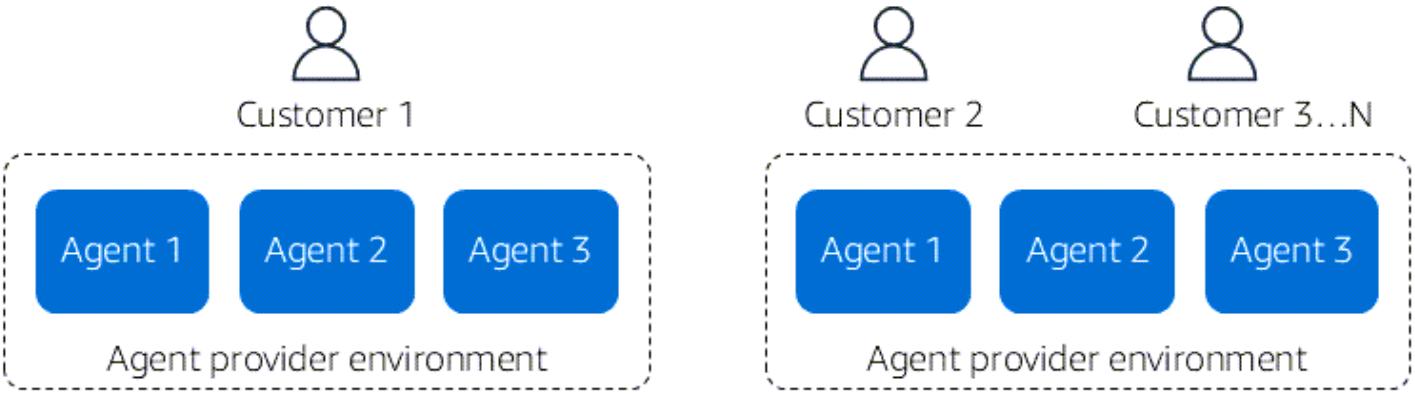


Ce schéma montre la séparation traditionnelle des plans de contrôle et d'application. Ce qui est nouveau, c'est que le plan de contrôle gère désormais les agents qui constituent un environnement AaaS. Le plan de contrôle interagit avec tous les agents car nous supposons que les agents sont créés, gérés et déployés par un seul fournisseur.

Ce modèle introduit des niveaux de complexité supplémentaires, notamment en ce qui concerne le cycle de vie des agents et la coordination avec des tiers, mais conserve la séparation fondamentale des préoccupations. Le plan de contrôle fournit toujours les mêmes fonctionnalités de base en

orchestrant la configuration des agents, en garantissant l'observabilité des locataires et des agents, en collectant les données de consommation et de mesure pour la facturation et en gérant les politiques des locataires.

Ce scénario devient plus complexe si vous considérez un système multi-agents qui intègre des agents de différents fournisseurs. Le schéma suivant montre un exemple d'un tel modèle.



Ce diagramme représente quatre agents de différents fournisseurs qui font partie d'un système multi-agentic. Les fournisseurs tiers continuent d'exploiter et de déployer chaque agent, qui est configuré pour permettre l'accès autorisé d'un ou de plusieurs fournisseurs. Les agents restent toutefois sous le contrôle du fournisseur, de sorte que chaque agent conserve son propre plan de contrôle.

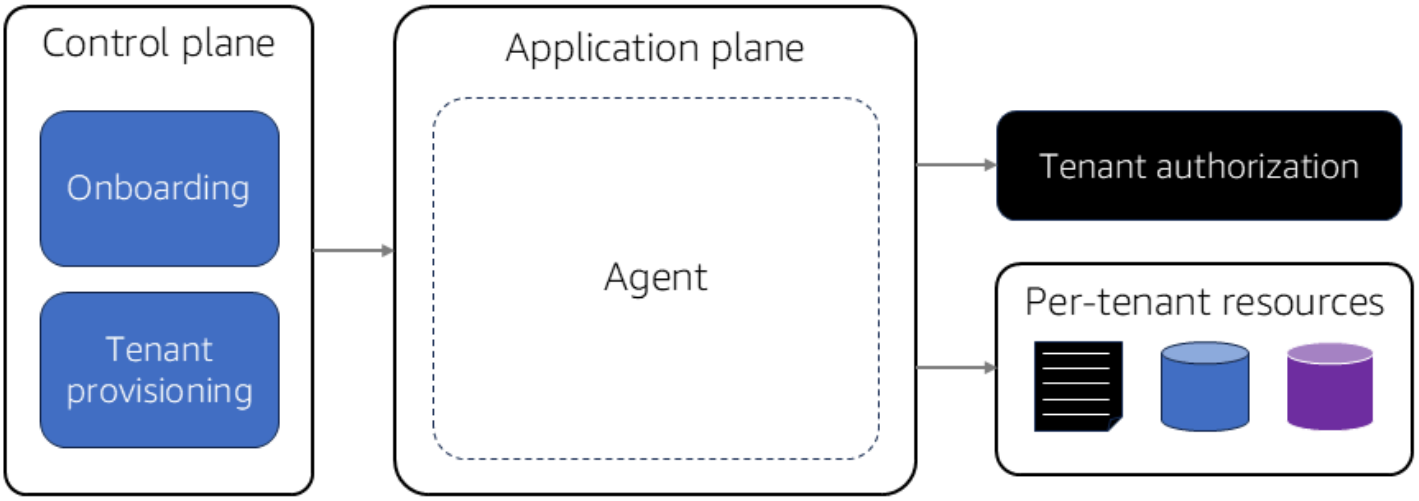
Essentiellement, ces agents multi-locataires se comportent comme des services tiers qui s'intègrent à d'autres agents. Ils doivent donc disposer de leur propre plan de contrôle pour centraliser le fonctionnement, la configuration et la gestion des capacités d'un agent.

Nous partons du principe que les agents sont des services indépendants qui fonctionnent dans le cadre d'une expérience hébergée par un fournisseur. Mais cela peut ne pas être clair dans un scénario où un agent consommateur impose plus de contraintes quant à la manière et au lieu d'héberger un agent.

Intégrer les locataires aux agents

L'intégration est généralement un élément essentiel de tout environnement AaaS. La façon dont vous créez, configurez et approvisionnez les locataires implique souvent de nombreux éléments mobiles, intégrations et outils. L'expérience d'intégration des agents peut nécessiter les mêmes services que ceux que l'on trouve dans un plan de contrôle AaaS, notamment l'identité des locataires, la hiérarchisation, le provisionnement des ressources par locataire et la configuration des politiques des locataires.

Votre approche en matière d'intégration des agents est influencée par l'encombrement et le modèle de location de votre environnement d'agence. Les agents cloisonnés et groupés ont chacun leurs propres nuances, et le choix d'utiliser un ou plusieurs agents a également une incidence sur le processus d'intégration. Le schéma suivant montre une vue conceptuelle de l'impact de l'intégration sur la configuration d'un agent.



Chaque fois que vous embarquez un agent, le plan de contrôle doit prendre les mesures nécessaires pour permettre au locataire d'accéder à l'agent. La procédure d'introduction des locataires varie en fonction du modèle d'autorisation de l'agent, mais supposons que vous allez créer une identité de locataire qui associe les demandes des agents à des locataires individuels. Ce contexte de locataire dicte l'expérience de l'agent en l'appliquant aux itinéraires, aux étendues et au contrôle d'accès.

L'intégration peut également vous obliger à configurer les ressources par locataire utilisées par un agent. C'est ici que le service de provisionnement des locataires du plan de contrôle connecte votre agent aux données et ressources spécifiques au locataire qu'il consulte.

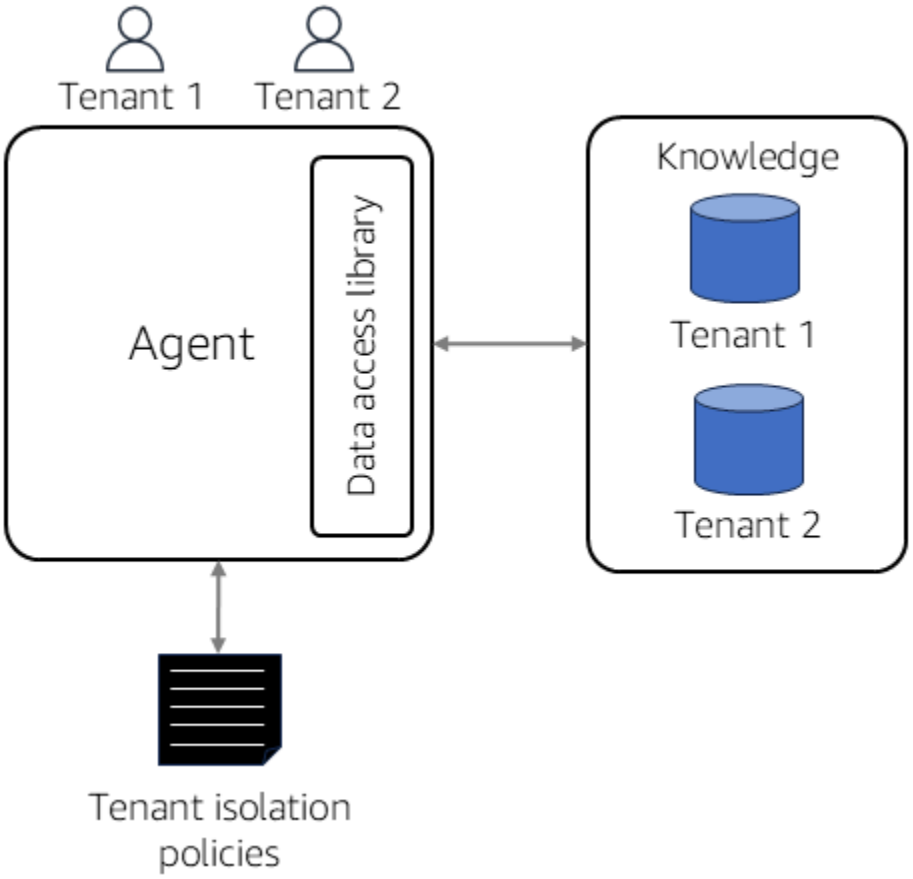
Si votre système repose sur l'intégration d'agents tiers, vous devez également répondre aux besoins de ces agents lors du processus d'intégration. Son fonctionnement dépend des mécanismes de sécurité et d'intégration permettant d'autoriser l'accès entre les agents. Idéalement, les étapes requises pour orchestrer et configurer l' agent-to-agentauthentification et l'autorisation sont traitées par le biais d'une intégration automatisée.

Renforcer l'isolement des locataires

L'isolation des locataires est un concept qui s'applique à tous les environnements multi-locataires. Cela signifie que vos politiques et stratégies garantissent qu'un locataire ne peut pas accéder aux autres ressources du locataire. Pour les agents à locataires multiples, vous devrez peut-être introduire des structures et des mécanismes qui aident à faire respecter les exigences d'isolation des locataires de l'agent.

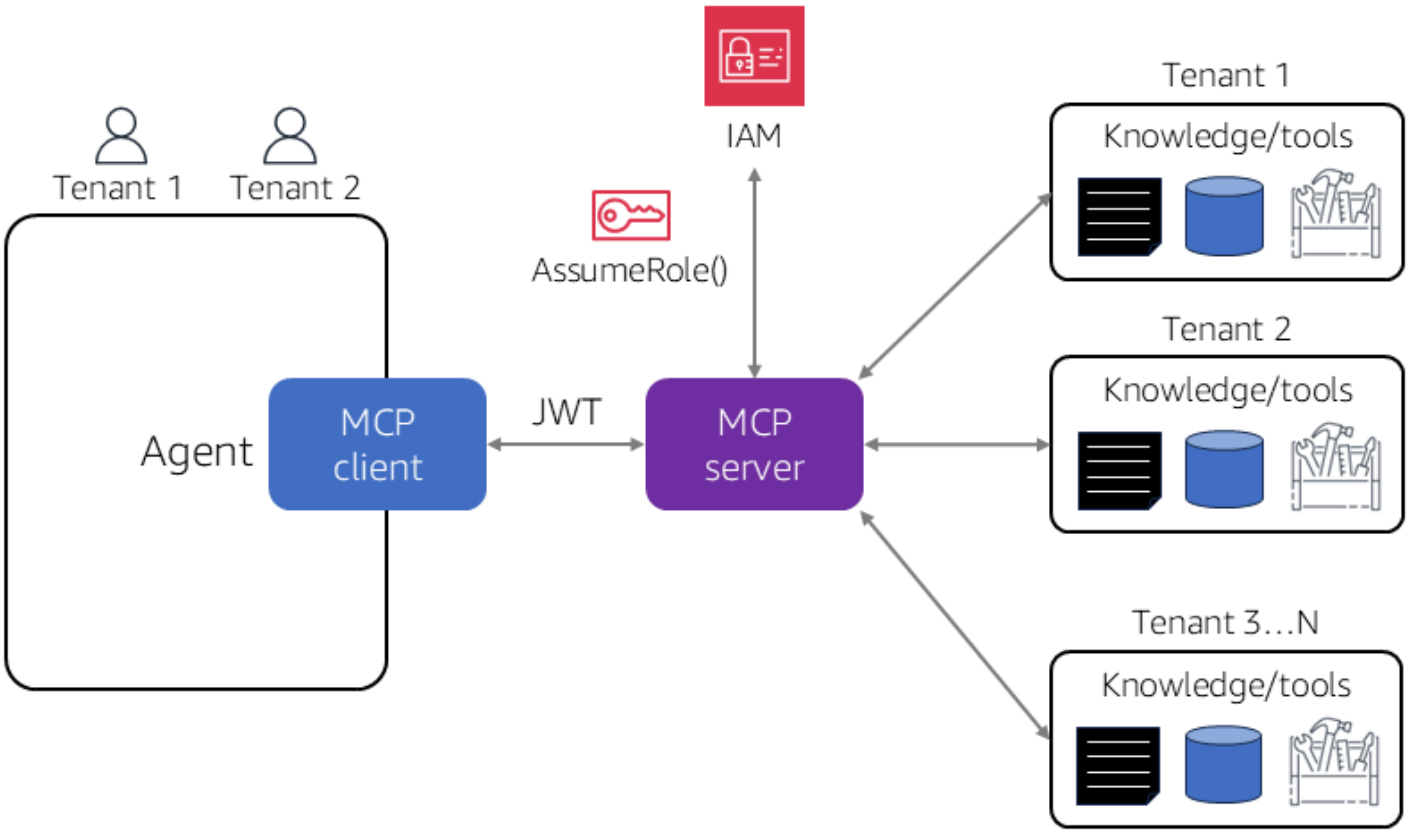
L'application de l'isolation des locataires s'apparente à d'autres stratégies utilisant des systèmes multilocataires traditionnels. En général, lorsque vous créez une architecture AaaS, identifiez toute zone de votre système dans laquelle une demande ou une action peut accéder aux ressources afin de déterminer si la demande dépasse les limites des locataires. Par exemple, les microservices peuvent dépendre de tables Amazon DynamoDB dédiées par locataire. Cela vous oblige à mettre en place des politiques garantissant que la table d'un locataire ne soit pas accessible à un autre locataire.

Dans ce cas, considérez l'isolement du locataire du point de vue de l'agent et ses interactions avec l'une de ses ressources par locataire. Le schéma suivant montre un exemple conceptuel de la manière dont les agents appliquent des politiques d'isolation des locataires pour contrôler l'accès aux ressources des locataires.



Sur le côté droit de ce schéma, l'agent dispose de connaissances par locataire qui sont stockées dans des bases de données vectorielles distinctes. Lorsque l'agent traite une demande, il examine le contexte dans lequel le locataire fait la demande. Sur cette base, l'agent applique une politique d'isolation appropriée pour garantir que les locataires ne peuvent pas accéder aux données ou aux ressources en dehors de leurs limites désignées.

Si votre agent utilise un protocole MCP (Model Context Protocol), il peut également implémenter votre modèle d'isolation des locataires. Le schéma suivant montre un exemple de la manière d'introduire le MCP et d'appliquer des politiques d'isolation.



Le MCP est un protocole standardisé qu'un agent utilise pour s'intégrer à tous les outils, données et ressources. Dans cet exemple, un client MCP et un serveur MCP interagissent avec les connaissances et les outils spécifiques au locataire présentés sur le côté droit du schéma. Le contexte du locataire passe du client au serveur, et le serveur utilise ce contexte pour obtenir des informations d'identification définies par le locataire auprès du service Gestion des identités et des accès AWS (IAM). Les informations d'identification contrôlent l'accès aux ressources de chaque locataire, garantissant ainsi qu'un locataire peut accéder aux ressources d'un autre locataire.

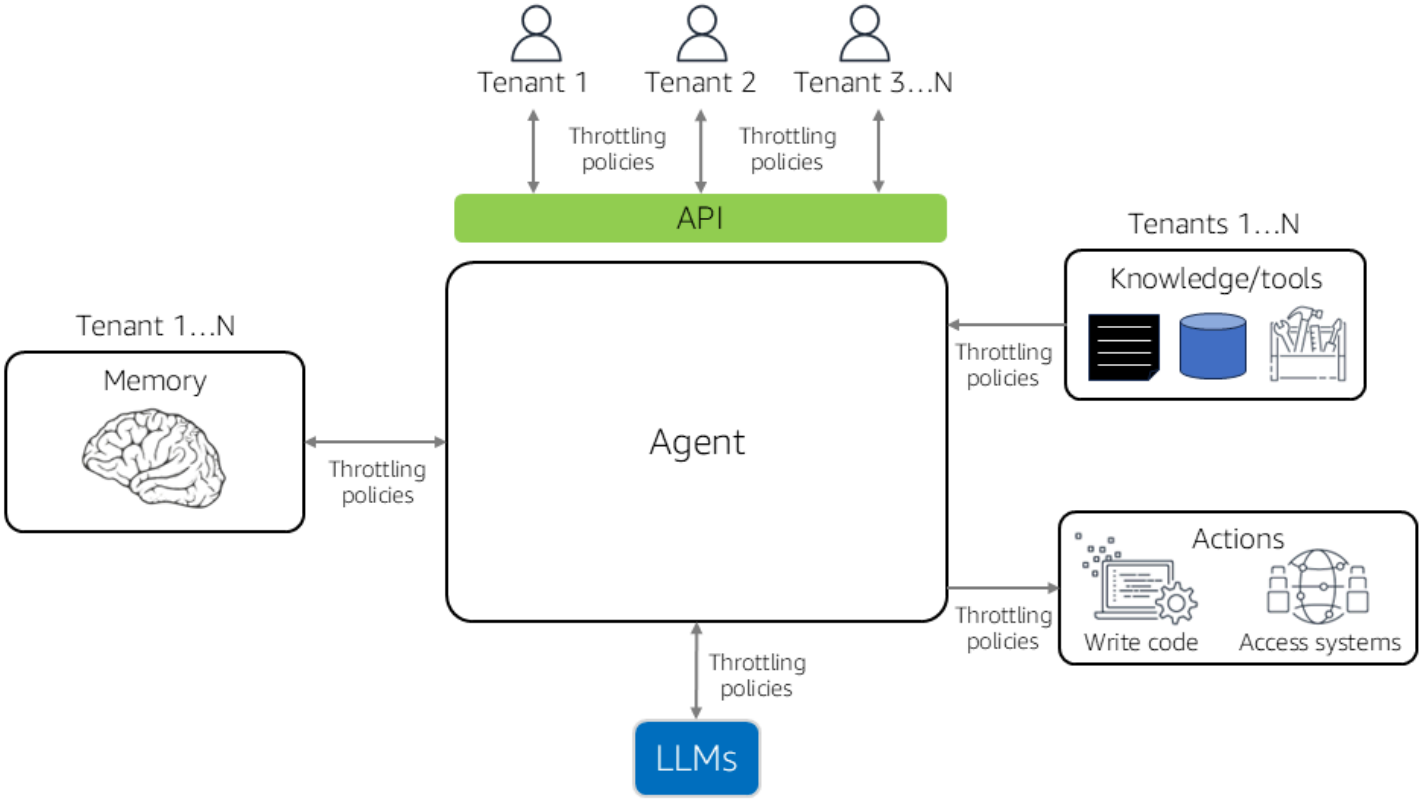
À mesure que les agents intègrent le multi-tenant, ils doivent introduire des mécanismes qui appliquent les politiques d'isolation des locataires lorsqu'ils traitent les demandes. Dans certains cas, l'IAM peut contribuer à limiter l'accès aux ressources des locataires. Dans d'autres cas, vous devrez peut-être introduire d'autres outils ou cadres pour appliquer les politiques d'isolation des locataires.

Voisin et agents bruyants

Dans un environnement AaaS à locataires multiples où plusieurs locataires partagent un agent, réfléchissez à l'endroit et à la manière d'introduire des politiques qui empêchent les voisins bruyants. Les politiques peuvent introduire une limitation à usage général qui s'applique à toutes les

consommations, ou vous pouvez avoir des politiques basées sur les locataires ou les niveaux qui appliquent une limitation en fonction d'un personnage donné. Vous pourriez imposer des restrictions de consommation plus importantes aux locataires de base qu'aux locataires de niveau supérieur.

Cette notion de régulation peut être appliquée à plusieurs points de l'architecture. Le schéma suivant montre un exemple de certains domaines dans lesquels il est possible d'introduire des politiques de voisinage bruyant.



Lors de notre examen précédent de la mise en œuvre de plusieurs agents, nous avons examiné les différentes ressources que votre agent peut utiliser, en mettant en évidence le potentiel de ressources par locataire au sein d'un agent. Chaque point de contact est un domaine susceptible d'entraîner l'introduction de politiques de limitation, qui permettent de garantir que les locataires ne dépassent pas les limites de consommation de votre système ou les politiques de hiérarchisation d'un locataire.

Les meilleurs endroits pour introduire des protections contre le bruit des voisins sont les points de l'architecture où les locataires partagent les ressources. Ces composants partagés ou regroupés, tels que le calcul, la mémoire et les grands modèles de langage APIs, sont les plus susceptibles de subir une dégradation des performances si un seul locataire consomme de manière disproportionnée.

L'un des endroits naturels pour appliquer la régulation est le point d'entrée de l'agent, parfois appelé « bord extérieur ». Ici, vous pouvez introduire des limites globales ou tenant-tier-based de débit avant que l'agent ne commence à traiter la demande. Le throttling peut également être appliqué plus profondément dans le chemin d'exécution, par exemple lorsque l'agent appelle un LLM, accède à la mémoire ou invoque des outils partagés.

Ces politiques vous aident à appliquer une utilisation équitable, à maintenir la résilience des agents sous charge et à garantir une expérience cohérente pour tous les locataires. En fonction de vos objectifs, vous pouvez vous concentrer sur la protection générale du système (résilience) ou sur la gestion détaillée de l'expérience des locataires (par exemple, avec des droits basés sur des niveaux).

Données, opérations et tests

Agents et propriété des données

Un examen de la mise en œuvre de l'agent met en évidence les scénarios dans lesquels un agent s'appuie sur les données d'un locataire donné. Dans ce cas, considérez le cycle de vie des données et, plus important encore, leur lieu de stockage. Cela est particulièrement important pour les secteurs et les cas d'utilisation où la nature des données influence la manière dont un agent y accède.

Les fournisseurs d'AaaS doivent évaluer comment résoudre les problèmes de données dans un environnement mutualisé, qui peuvent affecter l'intégration, l'isolation et les opérations d'un agent. Les nuances et les stratégies applicables varient en fonction des outils, des technologies et des données que vous utilisez. Vous pouvez aborder cette question de nombreuses manières, ce dont vous devez être conscient lorsque vous créez une offre AaaS.

Opérations d'agents à locataires multiples

Lorsque vous créez des environnements d'agents, réfléchissez à la manière de faire fonctionner et de gérer vos agents. En tant que fournisseur, vous avez besoin de métriques, de données, d'informations et de journaux qui vous permettent de surveiller l'état de santé, l'échelle et l'activité d'un agent. Cela est plus prononcé dans un environnement agentique à locataires multiples où vous souhaiterez comprendre comment les locataires individuels consomment les ressources des agents.

Cela est encore plus important dans les environnements multi-agents lorsque vous avez besoin d'informations sur les interactions entre agents. Pouvoir établir le profil et suivre les activités entre les agents peut être essentiel pour résoudre les problèmes qui affectent l'échelle, la précision et l'efficacité de votre système.

Les équipes opérationnelles peuvent également établir le profil des interactions LLM pour avoir une meilleure idée des charges que les agents supportent. LLMs Ces données sont essentielles pour affiner la mise en œuvre de l'agent. Cela peut également donner aux équipes opérationnelles une idée de la manière dont les agents et la location affectent le profil de coût global d'un système.

Formation et tests d'agents multi-locataires

L'un des défis associés aux agents de construction est qu'ils sont censés apprendre et évoluer. Cela signifie également que nous devons tester notre agent, le perfectionner et améliorer sa précision

avant de le mettre en production. Il existe de nombreux domaines dans lesquels vous pouvez inspecter et évaluer si votre agent évalue et catégorise correctement l'intention ou choisit et invoque les outils et actions appropriés. La liste des variables est longue, mais il s'agit en fin de compte de garantir que votre agent trouve des résultats qui répondent à vos objectifs.

L'examen de tous les éléments mobiles et des principes associés aux agents de test dépasse le cadre de ce document, mais notez que les stratégies de test ajoutent de la complexité aux environnements AaaS à locataires multiples. Par exemple, si un agent possède des données, de la mémoire et d'autres structures qui sont appliquées de manière contextuelle à chaque locataire, les résultats de l'agent peuvent être façonnés par les ressources de chaque locataire.

Si vous utilisez un agent pour simuler un scénario, vous devrez peut-être étendre votre simulation aux cas d'utilisation spécifiques au locataire. En conséquence, vous devez affiner les procédures de validation pour tenir compte des cas où les critères de validation diffèrent pour chaque locataire.

Considérations et discussion

Quelle est la place du SaaS ?

Les experts du secteur débattent activement de la manière dont les agents influencent le paysage du logiciel en tant que service (SaaS). S'il est vrai que les agents modifient les logiciels de nombreux systèmes, il est exagéré de suggérer aux agents de rendre les modèles de prestation obsolètes. Certains fournisseurs de SaaS seront probablement perturbés par l'adoption d'agents, tandis que d'autres pourraient entièrement repenser leur proposition de valeur en s'appuyant sur un modèle d'agent en tant que service (AaaS). D'autres peuvent trouver un équilibre en introduisant de manière sélective des agents pour répondre à des besoins spécifiques.

Ce sujet est intéressant car l'adoption des meilleurs principes du SaaS pourrait représenter la prochaine évolution du SaaS. Cela peut signifier que le SaaS est en train de disparaître ou que les principes fondamentaux du SaaS sont regroupés et mis en œuvre dans un modèle basé sur des agents. Il est probablement moins important de décider où la terminologie aboutira en fin de compte, mais il semble peu probable que le concept du SaaS disparaisse. Il est plus probable que les agents façonneront l'empreinte du SaaS.

En fin de compte, nous devons décider quelles stratégies peuvent être appliquées à l'AaaS, ce qui signifie permettre aux organisations d'adopter des architectures agentiques et des stratégies commerciales afin que les fournisseurs puissent maximiser l'efficacité, la valeur et l'impact de leurs systèmes agentiques. Les agents ne sont pas des boîtes noires. Les agents consomment des ressources, font évoluer les opérations, dépendent des données et génèrent des coûts, autant de facteurs que les fournisseurs doivent prendre en compte. Les fournisseurs d'agents doivent évaluer comment les principes du multi-tenant peuvent façonner les offres de services et optimiser les modèles opérationnels.

Explication

Le paysage agentique continue d'évoluer, les conceptions variant en fonction des domaines, des cas d'utilisation prévus et des industries cibles. Cette évolution implique notamment d'affiner davantage notre vision des stratégies, des modèles et des compromis que les architectes prennent en compte lorsqu'ils conçoivent et construisent des agents.

Une stratégie d'agent complète doit être alignée à la fois sur les objectifs commerciaux et techniques. Cela inclut la définition des marchés cibles et des personnalités, l'établissement de stratégies de

tarification et de gestion des ressources, et la détermination de la manière dont les agents s'intègrent dans des systèmes plus vastes. Ces considérations sont particulièrement importantes lors de la fourniture d'un service AaaS, où l'évolutivité, la rentabilité et l'innovation sont les principaux objectifs.

Les capacités opérationnelles sont tout aussi importantes. L'environnement doit prendre en charge la surveillance de l'activité des agents, des indicateurs de santé et des modèles d'utilisation. Cela devient plus complexe dans les systèmes multi-agents, où les opérations doivent être coordonnées entre des agents indépendants.

Dans l'ensemble, cette discussion sur les agents ne fait qu'effleurer la surface des diverses considérations architecturales qui pourraient faire partie des systèmes agentiques. Au-delà de la sélection des outils et LLMs des frameworks appropriés, le succès dépend de la création d'une architecture répondant aux exigences de l'entreprise en matière d'évolutivité, d'efficacité, de déploiement et de mutualisation.

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	—	14 juillet 2025

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.

épopée

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.