



Intégration de microservices à l'aide de services AWS sans serveur

AWS Directives prescriptives



AWS Directives prescriptives: Intégration de microservices à l'aide de services AWS sans serveur

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Les marques et la présentation commerciale d'Amazon ne peuvent être utilisées en relation avec un produit ou un service qui n'est pas d'Amazon, d'une manière susceptible de créer une confusion parmi les clients, ou d'une manière qui dénigre ou discrédite Amazon. Toutes les autres marques commerciales qui ne sont pas la propriété d'Amazon appartiennent à leurs propriétaires respectifs, qui peuvent ou non être affiliés ou connectés à Amazon, ou sponsorisés par Amazon.

Table of Contents

Introduction	1
Public visé	2
Objectifs	2
Sécurité	2
Schémas de communication	3
Communication synchrone	3
Communication asynchrone	5
Tirez et oubliez	6
Vérification des réclamations	6
Rappel	7
Communication bidirectionnelle	8
Options de coordination	11
Orchestration	11
Exemple : Step Functions	11
Exemple : Amazon MWAA	14
Principales différences entre Step Functions et Amazon MWAA	16
Chorégraphie	16
Choix de votre approche de coordination	17
Gérer APIs	18
Amazon API Gateway	18
Authentification et autorisation	18
Clés d'API et limites de débit	19
Public et privé APIs	19
Quand utiliser API Gateway	20
Messagerie	21
Amazon SQS	21
Interrogation	22
Conseils	22
Amazon SNS	23
Conseils	24
Amazon EventBridge	24
Conseils	25
AWS AppSync Événements et API Gateway	26
Conseils	26

FAQ	28
Comment puis-je combiner différents modèles d'intégration ?	28
Quel est le principal avantage de l'utilisation d'une architecture de microservices ?	28
Comment puis-je implémenter la gestion des erreurs ?	28
Quels sont les avantages du modèle de vérification des sinistres ?	28
Quels sont les avantages du modèle de rappel ?	29
Puis-je implémenter une communication bidirectionnelle ?	29
Comment optimiser l'utilisation des fonctions Lambda ?	29
Quelles sont les principales différences entre l'utilisation d'Amazon SNS et ? EventBridge	29
Ressources	30
Service AWS documentation	30
Lectures supplémentaires	30
Historique du document	31
Glossaire	32
#	32
A	33
B	36
C	38
D	42
E	46
F	49
G	51
H	52
I	54
L	56
M	57
O	62
P	65
Q	68
R	68
S	71
T	76
U	77
V	78
W	78
Z	79

..... lxxxi

Intégration de microservices à l'aide de services AWS sans serveur

Tabby Ward, Abhishek Agawane et Matt Kahn, Amazon Web Services

Septembre 2025 ([historique du document](#))

Un aspect important de la modernisation des logiciels de votre entreprise consiste à sélectionner les modèles architecturaux appropriés qui permettent l'agilité et la réactivité face à l'évolution des besoins de l'entreprise. Dans certaines applications, une architecture monolithique est un choix courant. Cependant, pour de nombreuses entreprises, la [transformation de monolithes en microservices](#) peut constituer une stratégie de modernisation efficace lorsque le cas d'utilisation correspond aux avantages des microservices.

Les microservices et les monolithes ne s'excluent pas mutuellement : de nombreuses entreprises prospères utilisent les deux modèles ensemble, les monolithes modulaires desservant certains domaines et les microservices en gérant d'autres.

Lorsque les microservices font partie de votre architecture, plusieurs services peuvent être appelés pour récupérer les données d'une transaction commerciale. La mise en œuvre de ces intégrations nécessite une conception minutieuse pour relever les défis potentiels tels que la cohérence des données, la latence et la complexité opérationnelle. Lorsque les microservices sont correctement intégrés, ils peuvent offrir des avantages tels qu'une mise à l'échelle indépendante, une vitesse de développement améliorée et des optimisations potentielles des coûts.

Le guide fait partie d'une série de contenus qui couvre l'approche de modernisation des applications recommandée par AWS. La série inclut également :

- [Stratégie de modernisation des applications dans AWS Cloud](#)
- [Approche progressive de la modernisation des applications dans AWS Cloud](#)
- [Évaluation de l'état de préparation à la modernisation des applications dans AWS Cloud](#)
- [Décomposer les monolithes en microservices](#)

Public visé

Ce guide s'adresse aux propriétaires d'applications, aux chefs d'entreprise, aux architectes, aux responsables techniques et aux chefs de projet qui ont déterminé que les microservices étaient adaptés à leur cas d'utilisation spécifique. Le guide présente plusieurs modèles de communication synchrone et asynchrone entre les microservices en utilisant le mode sans serveur Services AWS tel qu'Amazon API AWS Lambda Gateway pour l'autonomie et l'évolutivité.

Objectifs

En utilisant ce guide pour intégrer vos nouveaux microservices, vous pouvez transformer efficacement l'architecture de votre entreprise en une architecture de microservices. Cela permet de s'adapter rapidement aux besoins fluctuants de l'entreprise grâce à une évolutivité élevée, à une résilience améliorée, à une livraison continue et à une isolation des défaillances. Une architecture de microservices accélère également l'innovation, car chaque microservice peut être déployé et testé individuellement.

Une architecture de microservices peut également contribuer à réduire les délais de commercialisation de vos produits ou services, car chaque microservice possède une base de code indépendante qui facilite et accélère l'ajout de nouvelles fonctionnalités et leur itération.

Sécurité

Vous devez sécuriser correctement vos microservices pour protéger l'intégrité de vos services et de vos données, mais vous assurer que la sécurité n'a pas d'impact négatif sur les performances de votre application.

Dans un environnement de microservices, vous devez prendre en compte la manière dont chaque service authentifiera et autorisera les demandes qu'il reçoit d'un client externe ou d'un autre microservice. Tenez également compte de la manière dont chaque service pourra accéder aux autres en toute sécurité Services AWS.

L'accès à Services AWS doit être accordé par le biais de rôles à portée étroite [Gestion des identités et des accès AWS \(IAM\)](#). Le fait d'assumer un rôle IAM fournit au microservice des informations d'identification IAM à court terme sous la forme d'une clé d'accès, d'un secret d'accès et d'un jeton de session. Ils sont utilisés par les différents kits de développement logiciel (SDKs) pour signer les demandes à l'aide Services AWS de [AWS Signature Version 4 \(SigV4\)](#).

Schémas de communication

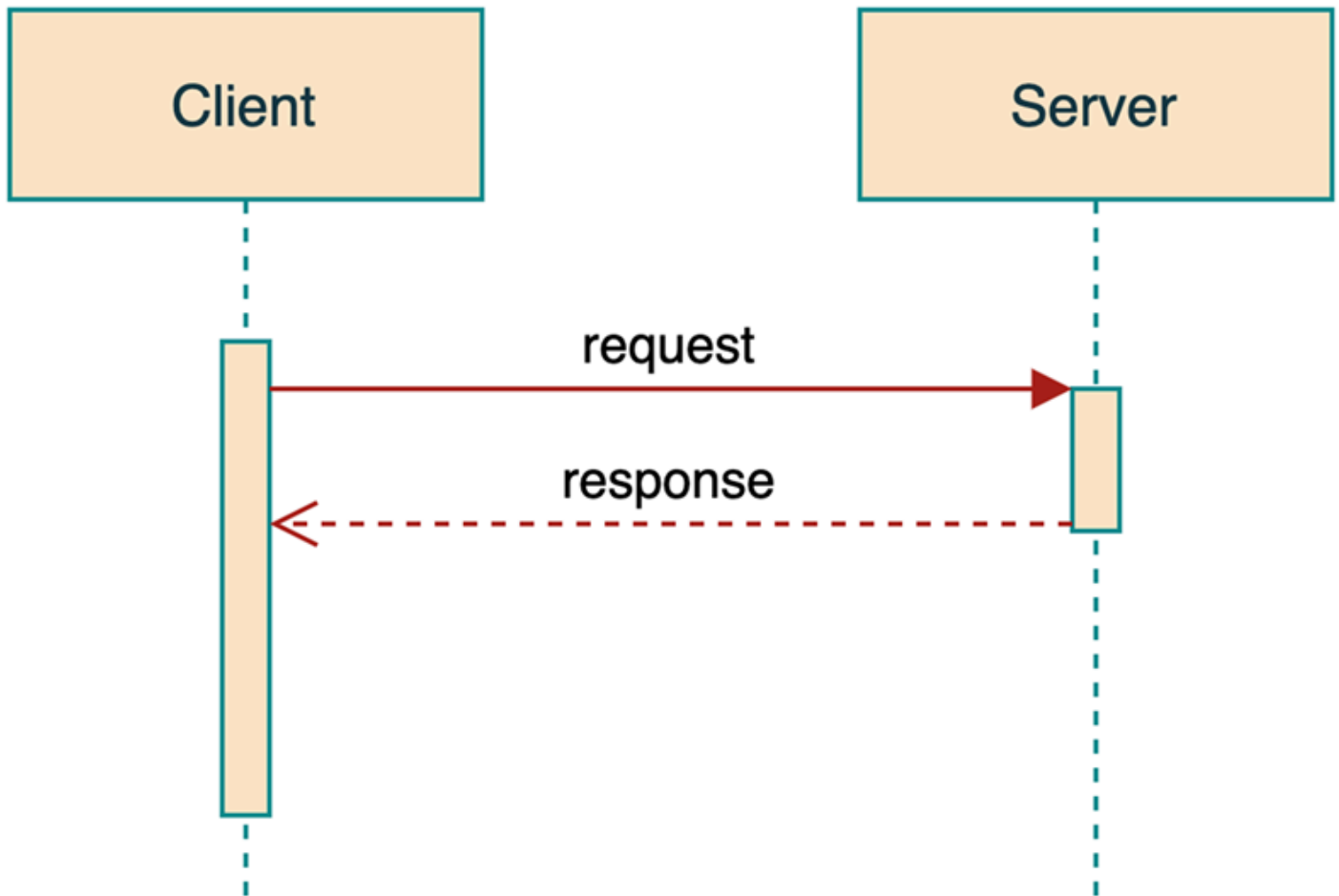
Dans une architecture de microservices, la communication s'effectue selon deux modèles principaux : synchrone et asynchrone. Dans une communication synchrone, l'appelant attend une réponse avant de poursuivre, comme dans le cas d'un appel d'API REST HTTP en temps réel. La communication asynchrone suit un schéma basé sur les messages dans lequel l'appelant poursuit le traitement sans attendre de réponse, par exemple lors de l'utilisation de files de messages. Les sections suivantes examinent en détail la mise en œuvre, les avantages et les cas d'utilisation de chaque modèle.

Rubriques

- [Communication synchrone](#)
- [Communication asynchrone](#)

Communication synchrone

Dans le cadre d'une communication synchrone, un client lance une demande auprès d'un service, comme l'illustre le schéma suivant. Les exemples incluent une demande d'extraction d'informations, telle qu'une HTTP GET demande, ou une demande de mutation de données, telle qu'une HTTP PUT demande. Dans les deux cas, le client attend que le serveur réponde avant de poursuivre. Les appels synchrones sont familiers à la plupart des développeurs, sont faciles à mettre en œuvre et à résoudre et, dans de nombreux cas, constituent la norme de communication largement acceptée.



Les avantages de la communication synchrone incluent :

- Contrôle de flux prévisible : exécution déterministe et cycles de demande-réponse clairs, plus faciles à comprendre par rapport à une communication asynchrone.
- Forte cohérence — Confirmation immédiate des modifications des données et des mises à jour de l'état.
- Gestion simple des erreurs — Propagation directe des erreurs et des exceptions.
- Débogage simplifié — Suivi et surveillance simples des demandes.
- Support des protocoles — Protocoles bien établis tels que HTTP et REST, qui simplifient la mise en œuvre.

La communication synchrone présente certains inconvénients :

- Couplage étroit — Dépendance directe des services par rapport à la disponibilité des uns et des autres.
- Impact sur le réseau : augmentation de la charge du réseau en raison de connexions ouvertes constantes.
- Utilisation des ressources : augmentation de l'utilisation de la mémoire due au maintien des états de connexion.
- Défaillances en cascade : capacité à ce que les problèmes rencontrés dans un service se propagent rapidement dans le système.

Communication asynchrone

À l'inverse, dans une communication asynchrone, le client envoie une demande à un service mais ne reçoit pas de réponse immédiate. Dans ce cas, le client ne reçoit généralement qu'un accusé de réception indiquant que la demande a été acceptée.

Les avantages de la communication asynchrone incluent :

- Support d'architecture piloté par les événements : parfaitement adapté aux modèles de recherche d'événements et de ségrégation des responsabilités des requêtes de commande (CQRS).
- Meilleure gestion des ressources — Capacité des services à traiter les demandes en fonction de leurs capacités.
- Isolation améliorée des défaillances : découplage des services, afin d'éviter les défaillances en cascade.
- Gestion des pics de charge — Meilleure gestion des pics de trafic grâce à la mise en file d'attente des messages.

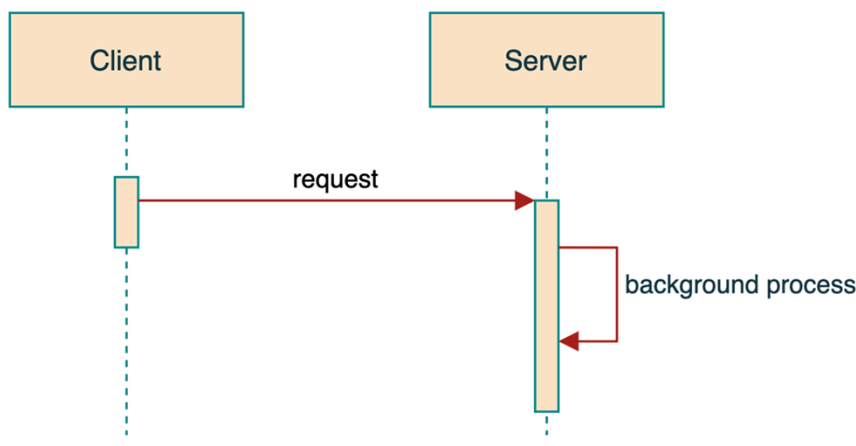
Les inconvénients incluent la complexité. Par exemple :

- Si le client a besoin du résultat de l'opération asynchrone, la mise en œuvre d'un mécanisme pour récupérer ou recevoir ce résultat nécessite plus d'efforts.
- Il peut s'avérer plus difficile de résoudre les problèmes liés aux opérations asynchrones, car le dépannage nécessite l'examen des journaux de plusieurs systèmes.
- Il peut être plus difficile de tester des opérations asynchrones, car les tests nécessitent une coordination entre plusieurs systèmes et services.

Les approches de communication asynchrone incluent le fire and forget, la vérification des réclamations, le rappel et la communication bidirectionnelle.

Tirez et oubliez

Dans le schéma « fire and forget », un client envoie une demande au serveur et reçoit de manière synchrone un accusé de réception indiquant que le serveur a reçu le message et qu'il va le traiter. Cependant, le traitement proprement dit n'a pas encore eu lieu et le client ne sait pas quand ni comment il sera effectué. Le schéma suivant illustre ce schéma.



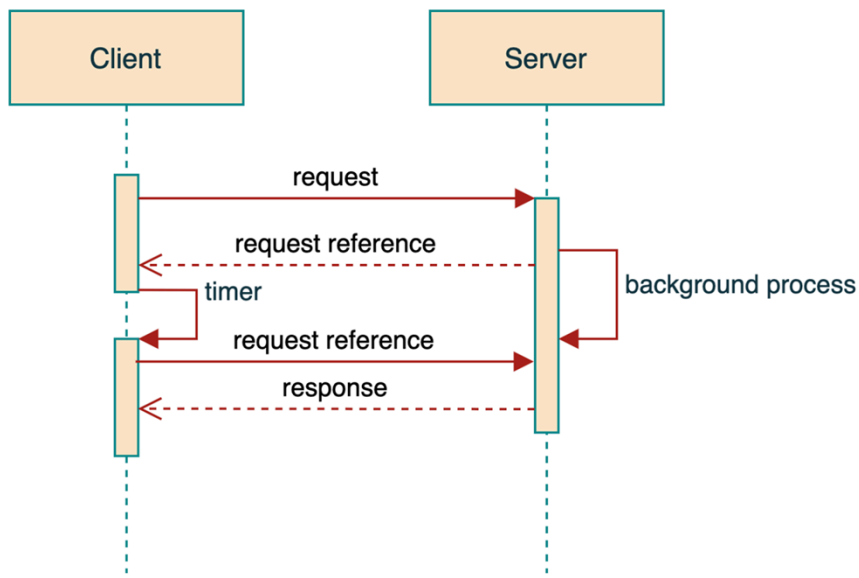
Dans ce cas, le service ne doit pas envoyer d'accusé de réception tant que l'objet n'est pas conservé de manière durable. Cette persistance peut être implémentée sous la forme d'une opération d'écriture dans une base de données ou en plaçant un élément dans une file d'attente.

Considérations supplémentaires :

- Implémentez l'idempotencie pour gérer les messages dupliqués. En d'autres termes, chaque message ne doit être traité qu'une seule fois.
- Envisagez les [files d'attente contenant des lettres mortes](#) en cas d'échec du traitement.
- Surveillez les taux de réussite du traitement des messages.

Vérification des réclamations

Si un client a besoin du résultat d'un appel de service, vous pouvez créer le service pour émettre un chèque de réclamation lorsqu'il reçoit une demande. Le schéma suivant illustre ce schéma. La vérification des réclamations est implémentée sous la forme d'un identifiant que le service renvoie dans son accusé de réception. Le client peut utiliser cet identifiant ultérieurement pour vérifier l'état de la demande et pour récupérer le résultat lorsque la demande est terminée.



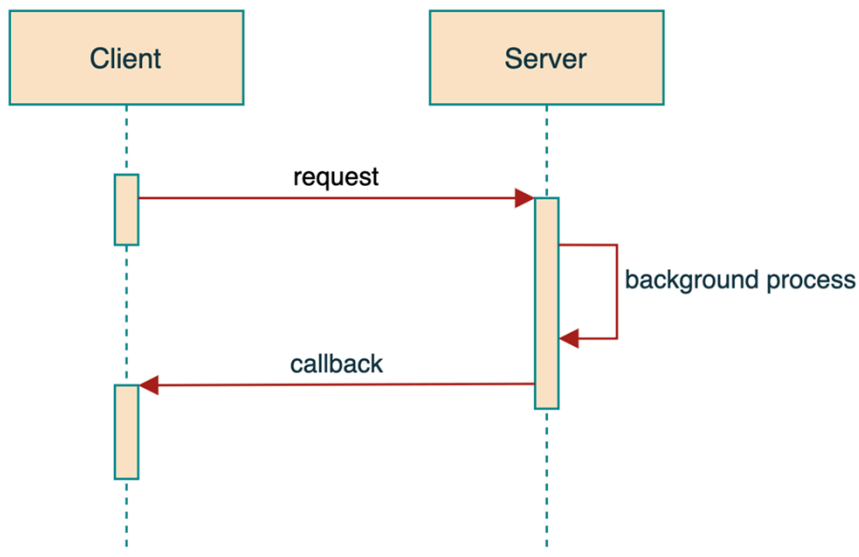
Les clients doivent mettre en place un mécanisme de sondage pour obtenir les résultats. Cela peut être automatisé (par exemple, un contrôle peut être effectué toutes les n minutes) ou mis en œuvre manuellement, lorsque le contrôle est effectué en réponse à un autre événement ou à une autre action de l'utilisateur. Les services qui mettent en œuvre le modèle de vérification des réclamations doivent indiquer clairement la durée de validité d'une vérification des réclamations.

Bonnes pratiques :

- Implémentez un ralentissement exponentiel pour les sondages.
- Définissez une durée de vie appropriée (TTL) pour les vérifications de sinistres.
- Fournissez des points de terminaison de statut pour le suivi des progrès.

Rappel

Dans le modèle de rappel, un client envoie une demande à un service et fournit un emplacement à contacter par le service une fois le traitement terminé. Le client n'attend pas de résultat et le traitement se poursuit. Le service est chargé de contacter le site une fois le traitement terminé et de fournir le résultat. Les types courants d'emplacements pour les réponses sont le REST APIs ou les files d'attente. Le schéma suivant illustre le modèle de rappel.

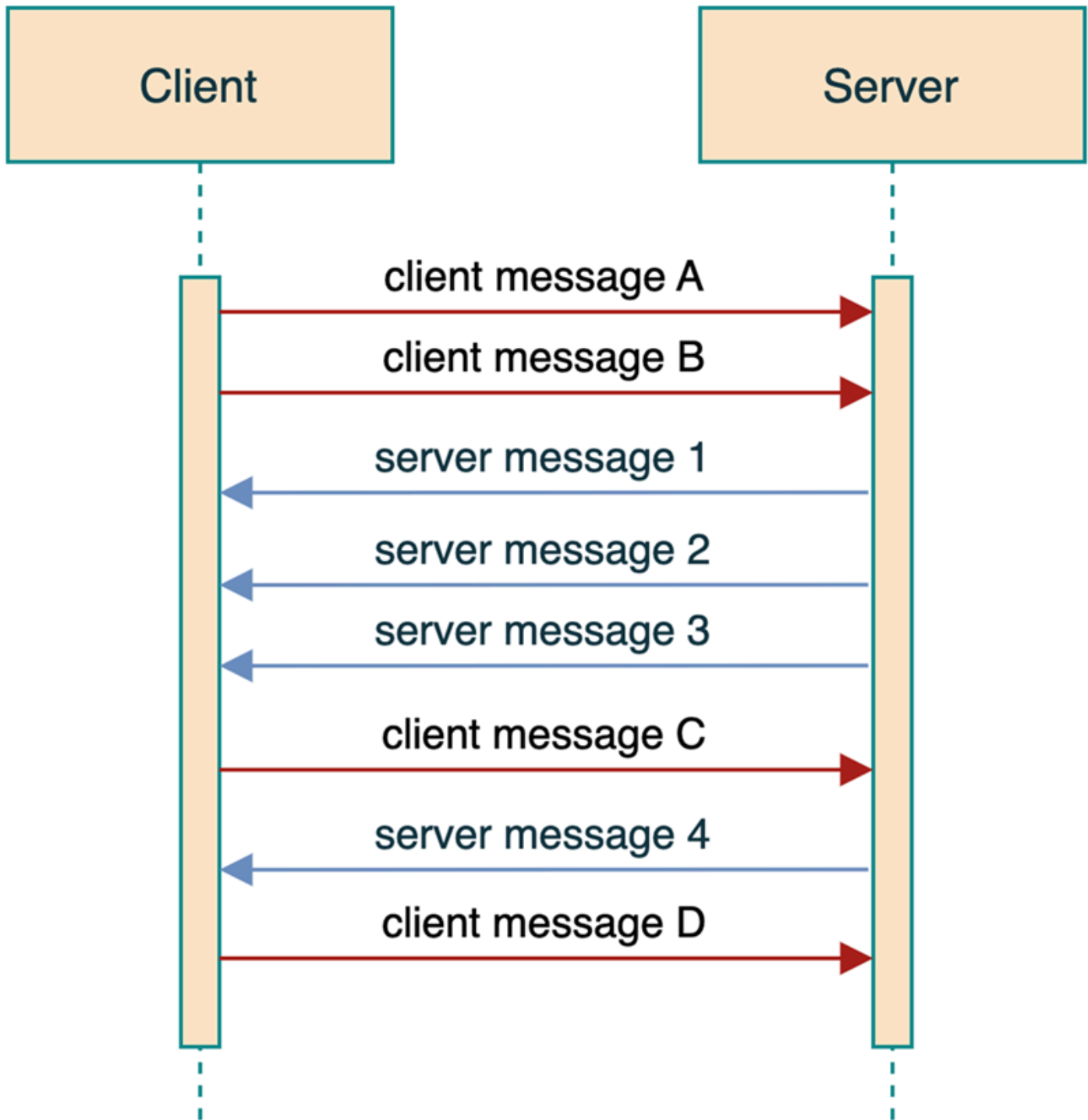


Mise en œuvre :

- Implémentez des mécanismes de nouvelle tentative en cas d'échec des rappels.
- Sécurisez l'emplacement du rappel comme vous le feriez pour les autres services.
- Gérez les délais de rappel.

Communication bidirectionnelle

Pour implémenter la communication bidirectionnelle, vous devez créer une connexion dynamique entre un client et un service, qui permet à la fois au client et au service d'envoyer et de traiter des messages. Le diagramme suivant en est l'illustration. Bien que la communication soit asynchrone, le service doit être capable de prendre en charge une connexion ouverte pour chaque client.



Considérations relatives à l'implémentation :

- Ordre des messages
 - Numéros de séquence

- Stratégies de partition
- Ordre des messages
- Gestion des états
 - Modèles d'approvisionnement pour les événements
 - Réconciliation entre États
 - Modèles de cohérence
- Gestion des erreurs
 - [files d'attente de lettres mortes](#)
 - Politiques de nouvelles tentatives
 - [Disjoncteurs](#)
 - Stratégies de repli
- Surveillance et observabilité
 - Corrélation IDs
 - Suivi des messages
 - Métriques de performances
 - Indicateurs de santé du système

Options de coordination

Les communications synchrones et asynchrones fonctionnent bien pour un client qui appelle un seul service ou un petit nombre de services. Cependant, dans un environnement réel, cette communication peut rapidement devenir compliquée et difficile à adapter. La réalisation d'une unité de travail peut nécessiter plusieurs microservices, qui peuvent être interdépendants. Souvent, ces interactions sont modélisées sous forme de flux de travail. Il existe deux approches pour concevoir ces flux de travail : l'orchestration et la chorégraphie.

Rubriques

- [Orchestration](#)
- [Chorégraphie](#)
- [Choix de votre approche de coordination](#)

Orchestration

Dans cette approche, un seul orchestrateur est chargé d'appeler chaque microservice, de déterminer s'il convient d'émettre des appels en séquence ou en parallèle, de manipuler les réponses de chaque service en cours de route et de compiler le résultat final. Un orchestrateur peut mélanger des invocations synchrones et asynchrones.

[AWS Step Functions](#) et [Amazon Managed Workflows for Apache Airflow \(Amazon MWAA\)](#) sont d'excellents choix pour les orchestrateurs de flux de travail.

L'orchestration est un bon choix lorsque votre processus comporte des branches logiques et que vous avez besoin d'un emplacement unique pour encapsuler cette logique. C'est également utile lorsque vous souhaitez implémenter le modèle de vérification asynchrone des réclamations. Par exemple, les flux de travail standard de Step Functions peuvent suspendre un flux de travail et attendre un rappel d'un autre service. L'utilisation d'un orchestrateur améliore également la surveillance et l'observabilité d'un processus.

Exemple : Step Functions

Vous pouvez utiliser Step Functions pour coordonner plusieurs fonctions Lambda et d'autres Services AWS fonctions pour créer des flux de travail complexes destinés à l'intégration de microservices.

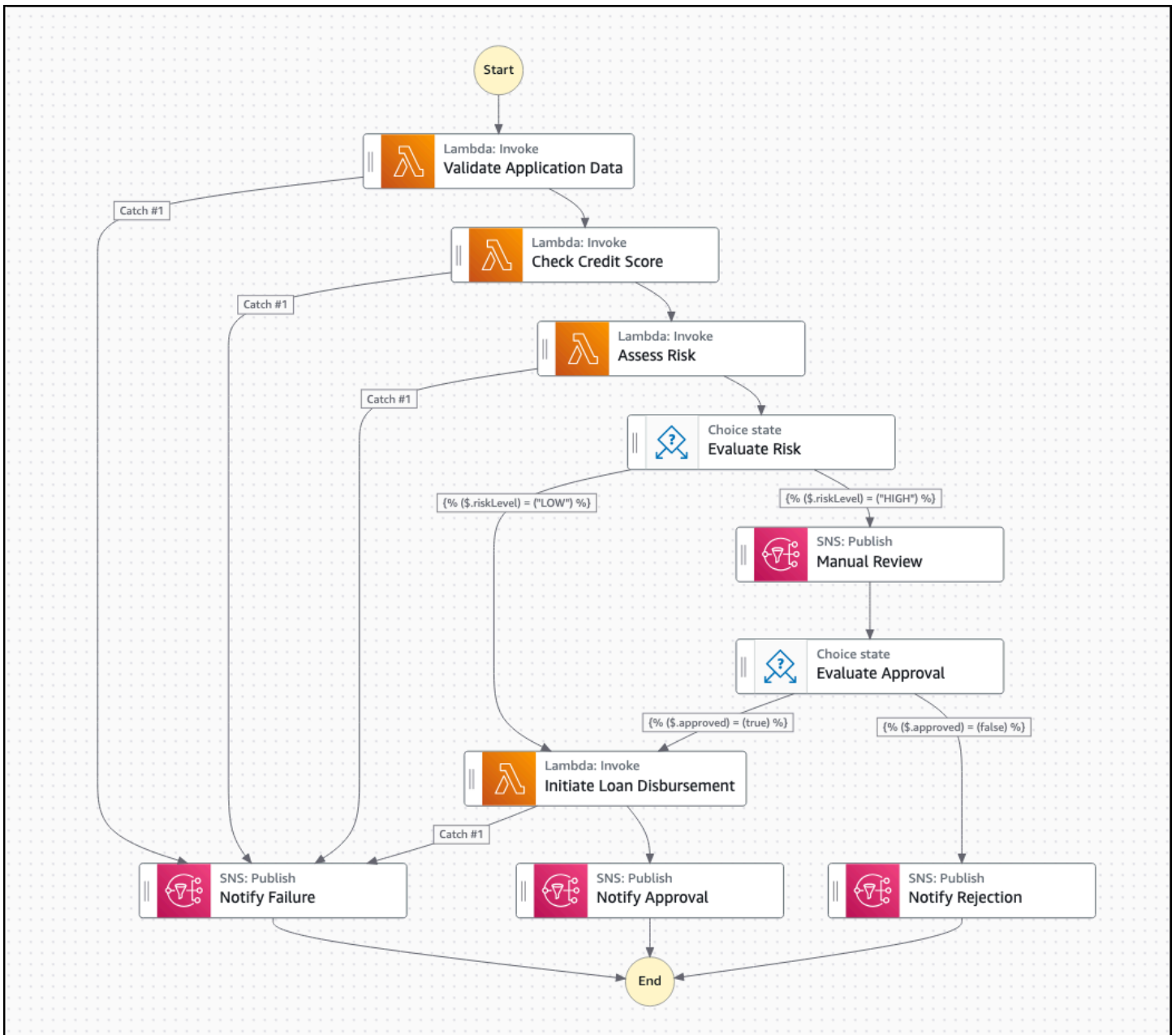
Cette option est particulièrement utile pour les processus de longue durée en plusieurs étapes impliquant plusieurs microservices.

Vous devriez envisager d'utiliser Step Functions si :

- L'intégration de vos microservices implique des processus complexes en plusieurs étapes.
- Vous devez maintenir l'état des opérations de longue durée.
- Vous souhaitez implémenter la gestion des erreurs et une logique de nouvelle tentative au niveau du flux de travail.
- Vous devez coordonner les opérations synchrones et asynchrones.

Step Functions propose un éditeur visuel pour concevoir des flux de travail complexes, ce qui simplifie le processus de création et de gestion des machines d'état. Il fournit des mécanismes intégrés de gestion des erreurs, notamment une logique de nouvelle tentative et une gestion de l'état d'erreur, qui améliorent la fiabilité et la robustesse de vos applications. Les flux de travail standard prennent en charge des processus de longue durée pouvant aller jusqu'à un an, ce qui convient aux flux de travail qui s'étendent sur de longues périodes. Cette option sépare la logique d'orchestration du code de l'application, ce qui réduit considérablement la complexité du code. Cela signifie que les développeurs peuvent se concentrer sur la logique métier de base tandis que Step Functions gère le contrôle du flux et la coordination des composants distribués.

Prenons l'exemple d'un processus d'approbation de prêt dans le cadre d'une demande de services financiers, illustré dans le schéma suivant. Le processus commence lorsqu'une demande de prêt est soumise.



Dans la machine à états illustrée dans le schéma précédent, Step Functions orchestre les étapes suivantes :

- Valider les données de l'application (fonction Lambda)
- Vérifier le pointage de crédit (fonction Lambda qui appelle une API externe)
- Évaluer le risque (fonction Lambda)
- En cas de risque élevé, passer à une révision manuelle (tâche d'approbation humaine)
- En cas d'approbation, lancer le décaissement du prêt (fonction Lambda)

- Envoyer une notification au candidat (Amazon SNS)

Vous pouvez utiliser cette approche pour gérer de manière fiable un processus complexe et potentiellement long, grâce à la gestion des erreurs intégrée et à la possibilité d'inclure des étapes automatisées et manuelles.

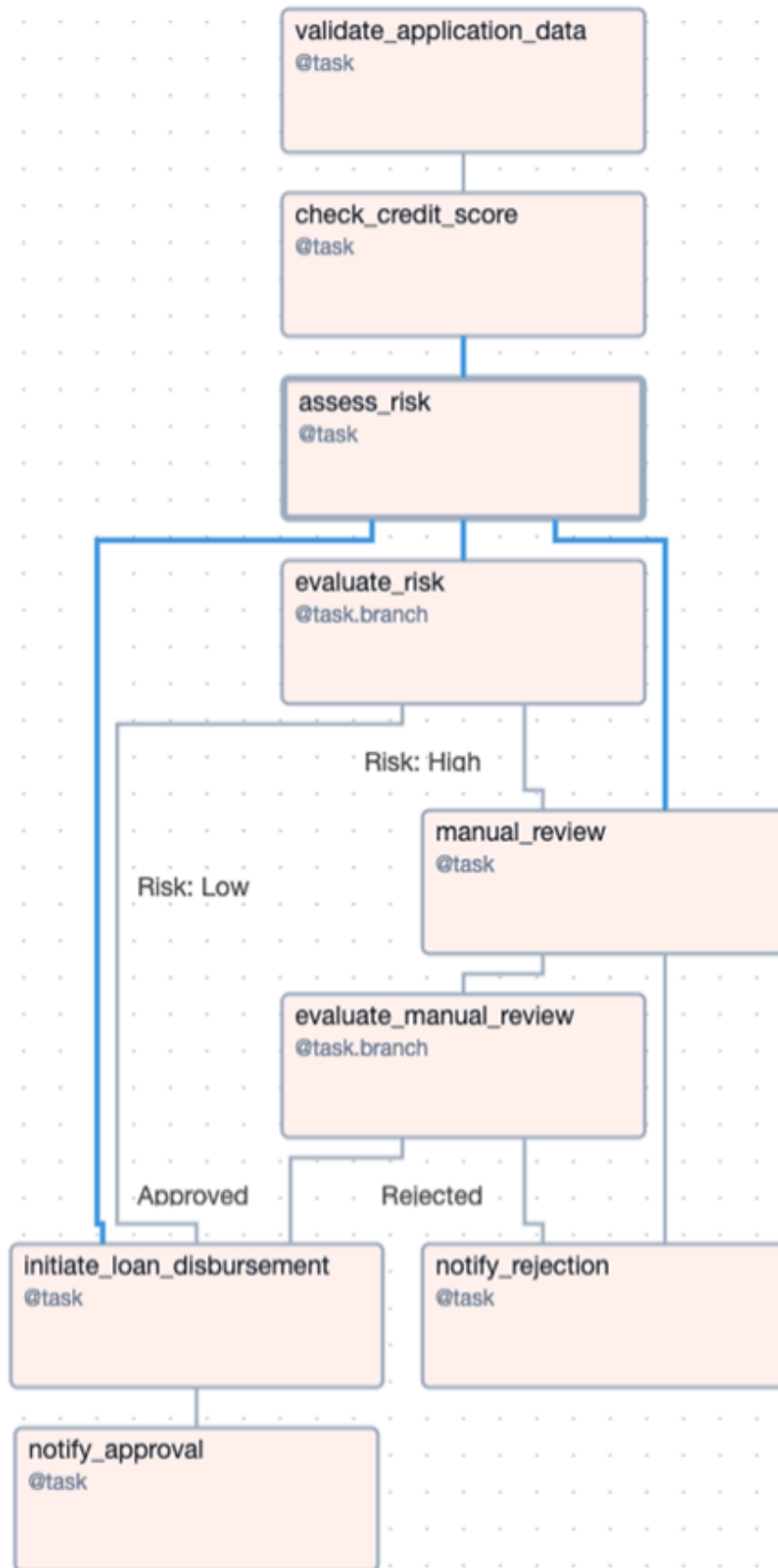
Considérations :

- Concevez votre machine à états avec soin pour gérer tous les scénarios possibles.
- Effectuez les étapes en parallèle dans la mesure du possible.
- Utilisez les mécanismes intégrés de gestion des erreurs et de nouvelle tentative de Step Functions pour les défaillances permanentes et temporaires.
- Envisagez d'utiliser des [flux de travail standard ou express](#) en fonction de votre cas d'utilisation. Les flux de travail express peuvent être préférables pour les flux de travail de courte durée ou de volume élevé.
- [Surveillez les indicateurs d'exécution](#) pour optimiser votre flux de travail.
- Utilisez des flux de travail imbriqués pour encapsuler et réutiliser les fonctionnalités sur plusieurs machines à états.
- Pour les flux de travail complexes, pensez à utiliser [Amazon Bedrock Agents](#) comme alternative à Step Functions.

Pour plus d'informations, consultez la [documentation Step Functions](#).

Exemple : Amazon MWAA

Si votre organisation utilise déjà Apache Airflow, Amazon MWAA est un choix naturel en tant qu'orchestrateur de flux de travail. Dans Apache Airflow, vous créez vos flux de travail sous forme de graphes acycliques dirigés (DAGs) à l'aide de Python. La représentation DAG de la machine à états illustrée dans la section Step Functions peut ressembler à ceci :



Pour plus d'informations sur l'utilisation DAGs, consultez la [documentation Amazon MWAA](#).

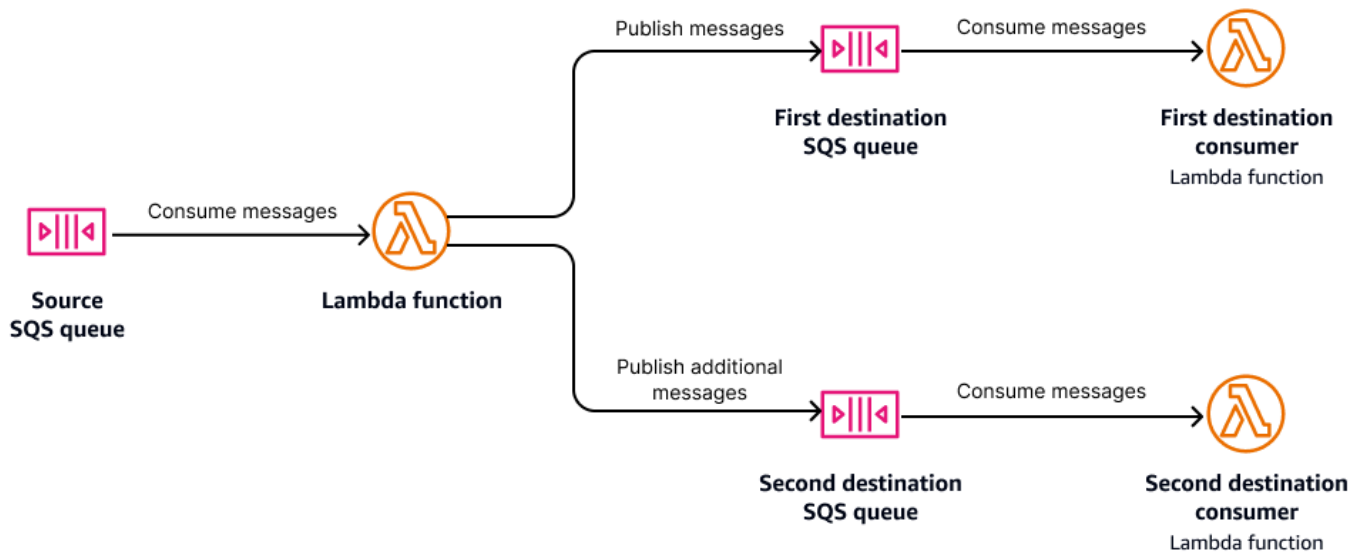
Principales différences entre Step Functions et Amazon MWAA

- Step Functions est un service sans serveur entièrement géré. Il n'y a donc aucune infrastructure à préprovisionner et il n'est pas nécessaire de planifier une fenêtre de maintenance. Amazon MWAA doit être déployé à l'avance, et vous choisissez la taille et le nombre de nœuds de votre cluster.
- Dans Step Functions, vous pouvez créer des machines d'état de différentes manières, notamment dans Workflow Studio, directement au format JSON ou en utilisant le AWS Cloud Development Kit (AWS CDK). Apache Airflow est écrit DAGs en Python.
- Avec Step Functions, vous n'avez aucun frais à payer lorsqu'aucun flux de travail n'est en cours d'exécution. Avec Amazon MWAA, vous devrez payer des frais même s'il n'en fonctionne pas DAGs .

Chorégraphie

Dans un système chorégraphié, les composants individuels reçoivent une tâche, exécutent un certain travail et émettent éventuellement une tâche pour un travail ultérieur à exécuter. Il n'existe aucun mécanisme d'orchestration central. La chorégraphie facilite la mise à l'échelle des services de manière indépendante, car chaque service fonctionne de manière relativement isolée. Il exécute un travail lorsqu'il reçoit du travail, quel que soit le débit dont le service est capable. La chorégraphie est souvent au cœur d'une [architecture événementielle](#) (EDA).

Dans le schéma suivant, il n'y a aucune coordination entre les fonctions Lambda. Chaque fonction traite uniquement les messages de la file d'attente abonée. Chaque fonction est responsable de sa propre gestion des erreurs et peut contrôler la simultanéité, par exemple, si une dépendance en aval a une limite de demandes par seconde (RPS).



Un EDA offre un certain nombre d'avantages, tels que le couplage souple des services et l'extensibilité. Une discussion complète des principes de l'EDA dépasse le cadre de ce guide. Pour en savoir plus, consultez :

- [AWS Well-Architected Framework — Lentille d'application sans serveur](#)
- [Présentation de l'architecture axée sur les événements](#) (Serverless Land)
- [Transition vers une architecture axée sur les événements](#) (Guide du développeur sans serveur)

Choix de votre approche de coordination

La chorégraphie et l'orchestration ont toutes deux leur utilité lors de l'intégration de microservices. Choisissez une chorégraphie dans les limites d'un microservice unique, où vous avez un contrôle total sur les dépendances. Choisissez l'orchestration lorsque vous travaillez au-delà des limites des microservices. Par exemple, plusieurs microservices participant à une transaction distribuée bénéficieront de l'orchestration pour tenir compte de l'annulation en cas de défaillance. Les microservices qui gèrent des événements susceptibles d'intéresser d'autres microservices bénéficieront de la chorégraphie et d'une architecture axée sur les événements.

Le modèle de saga est un modèle courant pour mettre en œuvre le rollback lorsque plusieurs systèmes sont impliqués dans une seule transaction.

Gérer APIs

Une gestion appropriée des API rend vos microservices accessibles aux consommateurs internes et externes. AWS propose une variété de services que vous pouvez utiliser ensemble pour exposer en toute sécurité votre microservice APIs. Ces services vous permettent de renforcer votre sécurité APIs et de mettre en œuvre la surveillance et l'observabilité à partir d'un emplacement central. Vous pouvez également utiliser [Amazon CloudFront](#) pour améliorer les performances si vos utilisateurs APIs sont géographiquement éloignés de l'endroit Région AWS où les services sont hébergés.

Amazon API Gateway

[Amazon API Gateway](#) est un service entièrement géré qui permet aux développeurs de créer, publier, gérer, surveiller et sécuriser REST, WebSocket APIs à n'importe quelle échelle. Vous pouvez utiliser API Gateway pour implémenter de nombreux modèles décrits dans la section [Modèles de communication](#) de ce guide.

Il existe deux principaux types de REST APIs : REST et HTTP. Les deux types sont compatibles RESTful APIs mais offrent des fonctionnalités différentes. Pour déterminer celui qui répond le mieux à vos besoins, consultez [Choisir entre REST APIs et HTTP APIs](#) dans la documentation d'API Gateway. Cette section du guide se concentre sur API Gateway REST APIs.

En utilisant API Gateway comme point d'entrée, vous disposez d' APIs un point unique pour mettre en œuvre les préoccupations communes, telles que la validation et la sécurité des demandes. API Gateway REST APIs propose la [validation des demandes](#), qui vous permet de définir le format de vos demandes à l'aide du [schéma JSON](#). API Gateway valide les demandes entrantes par rapport au schéma que vous avez défini et rejette les demandes mal formées.

Authentification et autorisation

API Gateway REST APIs prend en charge les mécanismes d'authentification (AuthN) et d'autorisation (AuthZ) suivants :

- IAM — Si vous utilisez IAM, les demandes adressées à votre API doivent être signées à l'aide de [AWS Signature Version 4 \(SigV4\)](#).
- Amazon Cognito — API Gateway validera le jeton porteur présenté comme ayant été émis par un groupe d'utilisateurs Amazon Cognito. Vous pouvez également configurer votre groupe

d'utilisateurs Amazon Cognito pour l'intégrer à un fournisseur d'identité (IdP) tiers, si vous en utilisez déjà un. Vous pouvez également utiliser un groupe d'utilisateurs Amazon Cognito pour l'authentification machine-to-machine (M2M).

- AWS Lambda authorizer — API Gateway invoquera une fonction Lambda que vous spécifiez pour effectuer les vérifications que vous souhaitez, afin de déterminer si une demande doit être autorisée.

Pour plus d'informations, consultez la section [Contrôler et gérer l'accès à REST APIs](#) dans la documentation d'API Gateway.

Clés d'API et limites de débit

Vous pouvez contrôler qui est autorisé à vous appeler APIs et à quel rythme en utilisant des clés API et des plans d'utilisation. Les clés d'API ne doivent pas être utilisées pour l'authentification mais peuvent être utilisées conjointement avec les schémas mentionnés précédemment. Les utilisateurs n'ont pas toujours besoin de fournir leur propre clé d'API. Par exemple, les autorisateurs Lambda peuvent renvoyer une clé d'API à un utilisateur. Le plan d'utilisation vous permet de définir le débit, la limite de rafale et le quota mensuel. Pour plus d'informations, consultez la section [Plans d'utilisation et clés d'API pour REST APIs](#) dans la documentation d'API Gateway.

Public et privé APIs

Les API Gateway APIs REST accessibles via Internet prennent en charge deux types de points de terminaison :

- Optimisé pour les périphériques, ce qui signifie que les demandes des appelants sont acheminées vers un CloudFront point de présence (POP) proche. Cela peut se traduire par une amélioration des performances pour des clients géographiquement disparates.
- Régional, ce qui signifie que les demandes sont acheminées vers une ressource au sein d'une ressource spécifique Région AWS. C'est un bon choix lorsque tous vos clients se trouvent à proximité de la région où votre API est déployée.

API Gateway REST prend APIs également en charge les points de terminaison d'API privés, accessibles depuis un cloud privé virtuel (VPC) à l'aide d'un point de terminaison VPC d'interface. Vous pouvez également partager en toute sécurité un REST privé APIs en créant des points de terminaison VPC d'interface dans d'autres VPCs et même dans d'autres. Comptes AWS Pour plus

d'informations, consultez la section [Types de points de terminaison d'API pour REST APIs](#) dans la documentation d'API Gateway.

Quand utiliser API Gateway

API Gateway est un bon choix pour les services RESTful Web et les WebSocket connexions en temps réel. Lorsque vous l'utilisez WebSocket APIs dans API Gateway, vous pouvez ajouter un comportement pour les événements de connexion et de déconnexion, tels que le stockage de la connexion IDs dans un magasin de données externe associé aux attributs du client. Vous pouvez également acheminer les demandes vers des comportements personnalisés à l'aide d'attributs de message.

REST et WebSocket APIs peuvent s'intégrer directement à de nombreuses applications Services AWS sans nécessiter de ressources de calcul distinctes telles que les fonctions Lambda. Cela peut améliorer les performances et réduire les coûts.

REST APIs prend en charge le routage basé sur les chemins et les en-têtes, et vous pouvez les utiliser séparément ou ensemble. Un modèle courant consiste à fournir une API REST comme porte d'entrée pour un certain nombre de personnes APIs, à mettre en œuvre des préoccupations communes, comme indiqué précédemment, puis à se comporter comme un proxy inverse et à acheminer les demandes autorisées vers le point de terminaison d'API approprié.

Messagerie

Comme indiqué dans la section [Modèles de communication](#), vous pouvez utiliser la messagerie pour communiquer de manière synchrone ou asynchrone entre les services. Il existe de nombreux services AWS sans serveur parmi lesquels choisir, et votre choix doit être basé sur vos besoins d'intégration. Par exemple, si vous avez besoin d'une livraison ordonnée de messages, vous devez choisir un service tel qu'Amazon Simple Queue Service (Amazon SQS) ou Amazon Simple Notification Service (Amazon SNS). Les deux services prennent en charge la livraison selon le principe du premier entré, premier sorti (FIFO), contrairement à Amazon EventBridge, qui ne le fait pas.

Les sections suivantes traitent de ces services plus en détail.

Rubriques

- [Amazon SQS](#)
- [Amazon SNS](#)
- [Amazon EventBridge](#)
- [AWS AppSync Événements et API Gateway](#)

Amazon SQS

[Amazon SQS](#) prend en charge les files d'attente standard, qui ne garantissent pas les commandes, et les files d'attente FIFO, qui garantissent les commandes au sein d'un groupe de messages donné.

Les files d'attente sont une méthode courante pour chorégraphier des microservices et fournissent un stockage durable des messages pendant 14 jours au maximum. Les files d'attente sont remplies par les producteurs et drainées par les consommateurs. Lorsque vous l'utilisez AWS Lambda en tant que consommateur, vous pouvez configurer une file d'attente SQS comme source d'événements. Dans ce cas, le mappage des sources d'événements du service Lambda (ESM) interroge la file d'attente pour vous et envoie des messages à votre fonction Lambda lorsqu'ils sont disponibles. Les microservices qui s'exécutent sur d'autres types de services informatiques, tels qu'Amazon Elastic Container Service (Amazon ECS) ou Amazon Elastic Compute Cloud (EC2Amazon), doivent implémenter leur propre mécanisme d'interrogation pour récupérer les nouveaux messages de la file d'attente lorsqu'ils sont disponibles.

Lambda ESM pour Amazon SQS prend également en charge le filtrage des messages, ce qui vous permet de traiter uniquement un sous-ensemble de messages dans une file d'attente en fonction du contenu du corps du message.

Interrogation

Amazon SQS prend en charge le sondage court et le sondage long des messages. Un court sondage interroge un sous-ensemble de serveurs pour trouver les messages disponibles et les renvoie immédiatement. Cependant, il se peut qu'il ne renvoie pas tous les messages disponibles. Cela est utile lorsque votre application doit consommer des messages le plus rapidement possible ou ne peut tolérer une attente prolongée.

Un long sondage attend qu'un laps de temps configurable se soit écoulé ou qu'un nombre configurable de messages ait été reçu avant de renvoyer les messages. Cela peut réduire le nombre de sondages vides, c'est-à-dire le nombre de sondages pour lesquels aucun message n'est renvoyé, en particulier pour les files d'attente qui ne reçoivent pas beaucoup de messages. La réduction du nombre de sondages vides peut réduire vos coûts Amazon SQS, car ce service est facturé pour chaque demande, et chaque opération d'interrogation est une demande.

Conseils

Les files d'attente sont un bon choix lorsque :

- Vous souhaitez découpler les composants et vous n'avez pas besoin d'une communication synchrone entre eux.
- Vous communiquez entre des composants qui ont des accords de niveau de service (SLAs) ou des objectifs de niveau de service () différents en termes de disponibilité. SLOs
- Vous n'avez généralement qu'un seul consommateur pour un ensemble de messages.

Envisagez une autre option si :

- Vous avez besoin d'une communication synchrone.
- Vous avez besoin d'une logique de routage complexe pour envoyer des messages au bon consommateur.

Amazon SNS

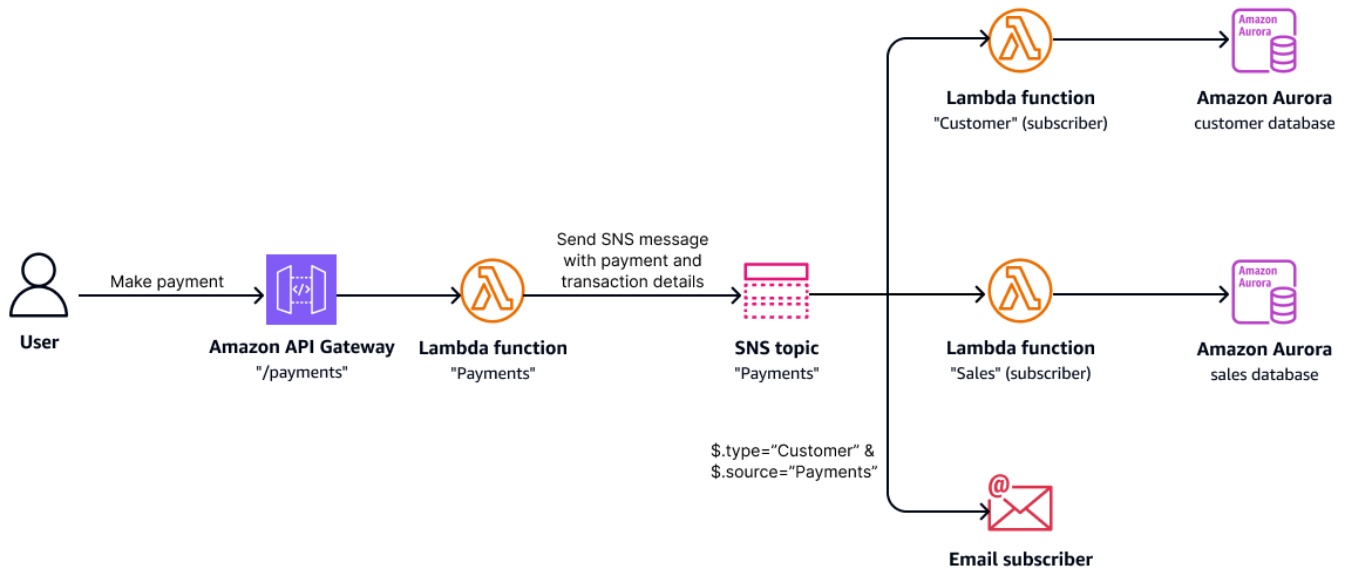
[Amazon SNS](#) vous permet de créer des rubriques standard et FIFO. Les rubriques sont utilisées pour implémenter une publish/subscribe architecture (pub/sub). Amazon SNS prend en charge différents types d'abonnement, notamment les e-mails, les SMS (en supposant que vous ayez configuré une identité d'origine, telle qu'un numéro gratuit ou un code long à 10 chiffres), les points de terminaison HTTP (S) et les files d'attente SQS. Les abonnements des utilisateurs finaux, tels que les e-mails et les SMS, à une rubrique SNS doivent être confirmés par l'abonné. Amazon SNS permet aux services de se déployer à grande échelle, ce qui signifie qu'un seul message peut être envoyé à un nombre potentiellement important d'abonnés. Une rubrique standard SNS a une limite par défaut de 12,5 millions d'abonnements.

Dans un environnement de microservices, une rubrique SNS est utile pour dissocier le routage des messages et la logique de distribution d'un éditeur. Cela peut être mis en œuvre à l'aide de filtres thématiques. Conceptuellement, les filtres de sujets sont assez similaires aux EventBridge règles d'Amazon, mais ils sont configurés pour chaque abonné au lieu d'être disponibles depuis un emplacement centralisé. Supposons, par exemple, que vous ayez :

- Un service de commande, qui traite les commandes.
- Un service de distribution qui gère le traitement des commandes.
- Un service de fidélité, qui attribue aux membres des points de fidélité pour leurs commandes.

Lorsqu'une commande est prête à être exécutée, elle publie un message dans un sujet. Le service Fulfillment s'abonne au sujet mais n'applique pas de filtre, car il souhaite connaître toutes les commandes. Imaginez que vous avez un service de fidélité, qui est chargé d'attribuer des points aux membres lorsqu'ils passent une commande. Cependant, toutes les commandes ne sont pas passées par les membres. Le service de fidélisation s'abonnerait au sujet mais mettrait en œuvre un filtre d'abonnement pour vérifier un attribut indiquant si la commande était destinée à un membre ou à un invité.

Prenons le cas où un système reçoit une demande d'un utilisateur final pour effectuer un paiement, comme illustré dans le schéma suivant. Dans ce cas, plusieurs systèmes en aval doivent savoir que la demande a été faite afin que diverses actions puissent être entreprises. Lorsque vous utilisez Amazon SNS, les paiements sont publiés sur une rubrique SNS, et les fonctions Lambda s'abonnent à cette rubrique pour mettre à jour les bases de données des clients et des ventes. En outre, un abonnement par e-mail (qui doit être confirmé par le client) envoie un e-mail de confirmation au client à l'aide d'un filtre d'abonnement.



Conseils

Certaines des fonctionnalités décrites dans cette section pour Amazon SNS recourent les fonctionnalités proposées par un bus d'événements, telles que. EventBridge Envisagez d'utiliser Amazon SNS dans les cas suivants :

- Vous aurez un grand nombre d'abonnés à un sujet.
- Vous souhaitez utiliser des types d'abonnement (tels que les e-mails ou les SMS), qui ne sont pas pris en charge de manière native par EventBridge.
- Les abonnés doivent être en mesure de déterminer leur filtre d'abonnement.
- Vous avez besoin d'une livraison commandée aux abonnés (par groupe de messages).

Si vous avez de nombreux sujets et que des abonnements et des filtres sont utilisés pour acheminer les messages entre les microservices, EventBridge c'est probablement le meilleur choix.

Amazon EventBridge

[Amazon EventBridge](#) est un service de bus d'événements sans serveur qui sert souvent de base à une architecture pilotée par les événements (EDA). Vous pouvez également l'utiliser pour acheminer et distribuer des messages de manière asynchrone entre les microservices. En utilisant EventBridge, les producteurs publient des événements dans un bus. Vous configurez des règles qui correspondent aux événements en fonction de leur contenu et vous choisissez une ou plusieurs cibles auxquelles

les événements correspondant à cette règle seront envoyés. EventBridge prend en charge un [large éventail de règles cibles](#). L'utilisation d'un bus d'événements vous permet de dissocier les producteurs des consommateurs et de consolider votre logique de routage et de livraison.

Dans EventBridge, vous pouvez également créer des règles planifiées, afin de pouvoir effectuer des actions à des moments précis. Vous pouvez définir des événements à l'aide d'expressions basées sur le cron et sur le taux.

[EventBridgePipes](#) vous permet de diriger des messages d'une [source](#) vers une [cible](#) sans utiliser un service informatique tel que AWS Lambda. Supposons, par exemple, que vous ayez une file d'attente SQS qui reçoit des messages, ce qui devrait déclencher une machine à AWS Step Functions états. Au lieu de créer une fonction Lambda dotée d'un mappage de source d'événements pour consommer les messages de la file d'attente et d'écrire du code à l'aide d'un AWS SDK pour appeler la machine à états, vous pouvez utiliser EventBridge Pipes pour le faire à votre place sans écrire de code personnalisé.

EventBridge est couramment utilisé avec d'autres services de messagerie tels qu'Amazon SQS et Amazon SNS. Par exemple, la transmission d'événements vers une file d'attente SQS donne à un service de réception la flexibilité nécessaire pour consommer des messages lorsqu'il le peut, à un rythme indépendant de celui auquel les événements sont produits. De même, vous pouvez proposer des événements qui devraient être diffusés à un grand nombre d'abonnés à un sujet SNS.

Conseils

À utiliser EventBridge lorsque :

- Vous n'avez pas besoin d'une communication synchrone entre les services.
- Vous souhaitez dissocier la logique de routage des messages de vos microservices. Les microservices produisent uniquement des événements et les publient dans le bus des événements, et les services intéressés créent des règles correspondant à ces événements et les proposent.
- Vous devez transmettre des messages d'un service pris en charge à un autre.

Envisagez d'autres services dans les situations où :

- Vous avez besoin d'un ordre strict des événements. Dans ces cas, considérez les files d'attente Amazon SQS FIFO ou les rubriques Amazon SNS FIFO. Vous pouvez également envisager des services de diffusion d'événements tels qu'Amazon Kinesis Data Streams ou Amazon Managed Streaming for Apache Kafka (Amazon MSK).

AWS AppSync Événements et API Gateway

AWS AppSync Events et Amazon API Gateway fournissent tous deux une WebSocket expérience gérée pour vos microservices.

[AWS AppSync Events](#) offre une expérience simplifiée pour la messagerie en temps réel en utilisant un WebSocket. AWS AppSync Events prend en charge les messages monodiffusion et multidiffusion, ainsi que le regroupement flexible de canaux dans des espaces de noms, avec prise en charge des caractères génériques. Les microservices peuvent communiquer entre eux de différentes manières à l'aide des AWS AppSync événements. Par exemple, un service qui reçoit des données en temps réel peut transformer et publier les données sur le canal approprié, où les abonnés les recevront en temps réel.

[API Gateway](#) prend également en charge WebSocket APIs. Vous pouvez définir des intégrations avec Amazon DynamoDB Services AWS, par exemple, AWS Lambda et configurer des expressions de sélection d'itinéraires mappées à ces intégrations. API Gateway possède des itinéraires spéciaux que vous pouvez utiliser pour autoriser et gérer vos WebSocket connexions. En fonction de vos besoins, vous pouvez stocker les informations de WebSocket connexion dans un magasin de données tel que DynamoDB. Grâce à ces informations, les messages peuvent être publiés sur des WebSocket connexions spécifiques via une API REST, en fonction d'un identifiant de connexion spécifique.

Conseils

Utilisez AWS AppSync les événements lorsque :

- Vous avez plusieurs canaux de messagerie regroupés dans des espaces de noms et vous souhaitez publier des groupes de canaux et vous y abonner en utilisant des caractères génériques.
- Vos communications se font principalement entre différents systèmes au lieu d'être entre eux Services AWS.

Utilisez API Gateway WebSocket APIs lorsque :

- Vous souhaitez permettre aux clients d'avoir des connexions persistantes en temps réel aux Service AWS intégrations.
- Vous souhaitez gérer vous-même WebSocket les connexions. Par exemple, vous pouvez autoriser d'autres systèmes à envoyer des messages à un client en particulier après avoir consulté son identifiant de connexion.

- Vous souhaitez utiliser les fonctionnalités d'API Gateway, telles que les déploiements par étapes ou les intégrations de proxy, ou vous souhaitez configurer vos propres sous-protocoles.

FAQ

Comment puis-je combiner différents modèles d'intégration ?

Dans la plupart des cas, vous souhaitez combiner des modèles d'intégration. Par exemple, vous pouvez l'utiliser AWS Step Functions pour orchestrer un processus qui appelle un service distant en utilisant le modèle de vérification des réclamations. Vous pouvez également avoir un processus orchestré qui place les messages dans des files d'attente, qui, à leur tour, déclenchent des services chorégraphiés.

Quel est le principal avantage de l'utilisation d'une architecture de microservices ?

Les principaux avantages incluent la mise à l'échelle indépendante des services, une meilleure isolation des pannes, une vitesse de développement accrue grâce au travail d'équipe parallèle et la capacité de livraison et de déploiement continu (CI/CD).

Comment puis-je implémenter la gestion des erreurs dans ces modèles ?

Vous pouvez implémenter la gestion des erreurs en utilisant les mécanismes intégrés dans Services AWS. Par exemple, les AWS Lambda fonctions peuvent être configurées avec une logique de nouvelle tentative, et Amazon SQS prend en charge les files d'attente contenant des lettres mortes pour gérer les défaillances persistantes. De plus, Step Functions fournit des mécanismes de gestion des erreurs et de nouvelle tentative au niveau du flux de travail.

Quels sont les avantages de l'utilisation du modèle de vérification des sinistres dans les communications asynchrones ?

Le modèle de vérification des réclamations permet aux clients de recevoir un identifiant lors de la soumission de la demande. Cet identifiant peut être utilisé ultérieurement pour vérifier l'état et récupérer le résultat. Ce modèle profite aux clients en fournissant un mécanisme permettant de sonder les résultats sans attendre de manière synchrone. Pour plus d'informations, consultez la section [Vérification des réclamations](#) plus haut dans ce guide.

Comment le modèle de rappel améliore-t-il la communication asynchrone dans les microservices ?

Le modèle de rappel améliore la communication asynchrone en permettant au client de fournir un emplacement à contacter par le service une fois le traitement terminé. Cela permet au client de ne plus attendre une réponse et de poursuivre ses autres tâches. Pour plus d'informations, consultez la section [Callback](#) plus haut dans ce guide.

Puis-je implémenter une communication bidirectionnelle dans les microservices en utilisant les modèles décrits ?

Vous pouvez implémenter une communication bidirectionnelle en créant une connexion dynamique entre un client et un service, afin qu'ils puissent à la fois envoyer et traiter des messages de manière asynchrone. Cela nécessite que le service prenne en charge une connexion ouverte pour chaque client. Pour plus d'informations, consultez la section [Communication bidirectionnelle](#) plus haut dans ce guide.

Comment optimiser l'utilisation des fonctions Lambda dans les modèles de communication asynchrones ?

Vous pouvez optimiser les fonctions Lambda en vous assurant qu'elles sont suffisamment puissantes pour gérer d'éventuelles duplications de messages, en utilisant les fonctionnalités d'Amazon SQS telles que les groupes de messages pour les commandes et en implémentant de longs interrogations afin de réduire les coûts. En outre, vous pouvez surveiller les mesures d'exécution pour identifier les opportunités d'optimisation.

Quelles sont les principales différences entre l'utilisation d'Amazon SNS et EventBridge le modèle ? pub/sub

Amazon SNS envoie un message unique à tous les abonnés, qui peut inclure des données inutiles pour certains abonnés. Amazon EventBridge permet un contrôle plus précis en vous permettant d'avoir plusieurs règles correspondant à un seul événement, chaque règle déclenchant un service ou une action en aval différent. Pour plus d'informations, consultez [Amazon SNS et les EventBridge](#) sections précédentes de ce guide.

Ressources

Service AWS documentation

- [Amazon API Gateway](#)
- [Événements AWS AppSync](#)
- [Amazon EventBridge](#)
- [Amazon MWSAA](#)
- [Amazon SNS](#)
- [Amazon SQS](#)
- [AWS Step Functions](#)

Lectures supplémentaires

- [Stratégie de modernisation des applications dans AWS Cloud](#)
- [Approche progressive de la modernisation des applications dans AWS Cloud](#)
- [Évaluation de l'état de préparation à la modernisation des applications dans AWS Cloud](#)
- [Décomposer les monolithes en microservices](#)
- [Mise en œuvre de modèles d'intégration d'entreprise avec AWS les services de messagerie : point-to-point canaux](#)
- [Messagerie Pub/Sub : notifications d'événements asynchrones](#)

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
Mises à jour majeures	Guide développé, réorganisé et mis à jour pour refléter les Service AWS mises à jour.	10 septembre 2025
Publication initiale	—	11 janvier 2021

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.
- **Replatformer (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'une 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 ou Bitbucket 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).

environnement

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 transactionnelles 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

I

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'exception des 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/subscribe modèle, 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 sécurité AWS](#) 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.