Choix d'une base de données AWS vectorielle pour les cas d'utilisation de RAG

AWS Conseils prescriptifs

AWS Conseils prescriptifs: Choix d'une base de données AWS vectorielle pour les cas d'utilisation de RAG

Table of Contents

Choix d'une base de données AWS vectorielle pour les cas d'utilisation de RAG

Mayuri Shinde, Ivan Cui et Anand Bukkapatnam Tirumala, Amazon Web Services

Mars 2026 (historique du document)

Les bases de données vectorielles sont de plus en plus importantes pour les organisations qui mettent en œuvre des applications d'IA générative. Ces bases de données stockent et gèrent des vecteurs, qui sont des représentations numériques de données permettant de traiter du texte, des images et d'autres contenus de manière à saisir leur signification et leurs relations.

À mesure que les entreprises explorent les options des bases de données vectorielles AWS, elles doivent comprendre les fonctionnalités, les compromis et les meilleures pratiques des différentes solutions. Ce guide vous aide à comparer les magasins de vecteurs couramment utilisés AWS et à prendre des décisions éclairées quant aux options les mieux adaptées à vos besoins ou à votre cas d'utilisation spécifiques. Que vous mettiez en œuvre la génération augmentée de récupération (RAG), que vous développiez des systèmes de recommandation ou que vous développiez d'autres applications d'IA, ce guide fournit un cadre qui vous aidera à évaluer et à choisir une solution de base de données vectorielle.

Public visé

Ce guide est destiné aux personnes occupant les rôles suivants :

- Scientifiques des données et ingénieurs en apprentissage automatique (ML) qui utilisent des bases de données vectorielles pour stocker et récupérer des données de grande dimension pour les modèles de machine learning.
- Ingénieurs de données qui conçoivent et mettent en œuvre des pipelines de données comprenant des bases de données vectorielles pour le stockage et le traitement de données de grande dimension.
- MLOps ingénieurs qui utilisent des bases de données vectorielles dans le cadre du pipeline ML pour stocker et diffuser les sorties de modèles ou les représentations intermédiaires.
- Ingénieurs logiciels qui intègrent des bases de données vectorielles dans des applications nécessitant des systèmes de recherche ou de recommandation de similarité.

- DevOps ingénieurs chargés du déploiement et de la maintenance des bases de données vectorielles dans les environnements de production, afin de garantir l'évolutivité et la fiabilité.
- Les chercheurs en IA qui utilisent des bases de données vectorielles pour stocker et analyser de grands ensembles de données d'intégrations ou de vecteurs de caractéristiques.
- Chefs de produits d'IA qui ont besoin de comprendre les capacités et les limites des bases de données vectorielles pour prendre des décisions éclairées concernant les fonctionnalités et l'architecture des produits.

Vue d'ensemble des vecteurs

Les vecteurs sont des représentations numériques qui aident les machines à comprendre et à traiter les données. Dans le domaine de l'IA générative, ils répondent à deux objectifs principaux :

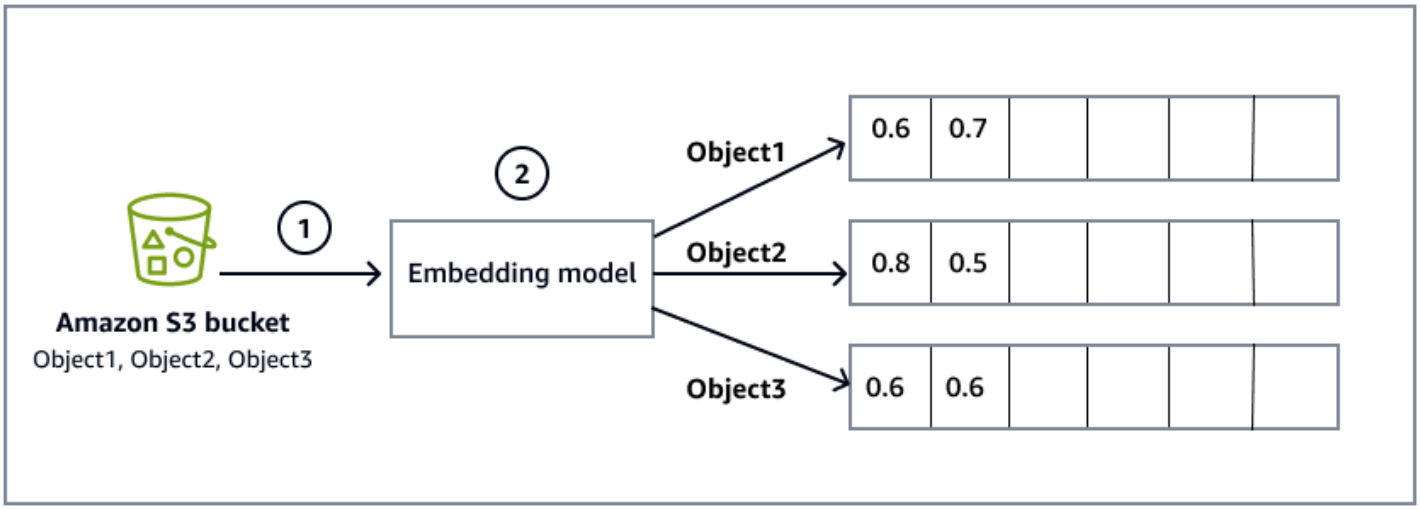
- Représentation des espaces latents qui capturent la structure des données sous forme compressée
- Création d'intégrations pour des données, telles que des mots, des phrases et des images

Les modèles d'intégration tels que Word2Vec GloVeet Amazon Titan Text Embeddings convertissent les données en vecteurs grâce à un processus appelé intégration. Ces modèles d'intégration peuvent effectuer les opérations suivantes :

- Apprenez à partir du contexte pour représenter les mots sous forme de vecteurs
- Placer des mots similaires plus près les uns des autres dans l'espace vectoriel
- Permettre aux machines de traiter les données dans un espace continu

Le schéma suivant fournit une vue d'ensemble détaillée du processus d'intégration :

1. Un bucket Amazon Simple Storage Service (Amazon S3) contient des fichiers qui sont les sources de données à partir desquelles le système lit et traite les informations. Le compartiment Amazon S3 est spécifié lors de la configuration de la base de connaissances Amazon Bedrock, qui inclut également la synchronisation des données avec la base de connaissances.
2. Le modèle d'intégration convertit les données brutes des fichiers objets du compartiment Amazon S3 en intégrations vectorielles. Par exemple, `Object1` est converti en un vecteur `[0.6, 0.7, ...]` qui représente son contenu dans un espace multidimensionnel.



Les intégrations de mots sont cruciales pour le traitement du langage naturel (NLP) car elles permettent :

- Capturez les relations sémantiques entre les mots
- Permettre la génération de texte contextuellement pertinent
- Alimentez les grands modèles linguistiques (LLM) pour produire des réponses semblables à celles des humains

Vue d'ensemble des bases de données vectorielles

Une base de données vectorielle est un système spécialisé qui stocke et interroge efficacement des vecteurs de grande dimension. Ces bases de données sont fondamentales pour les applications RAG (Retrieval Augmented Generation).

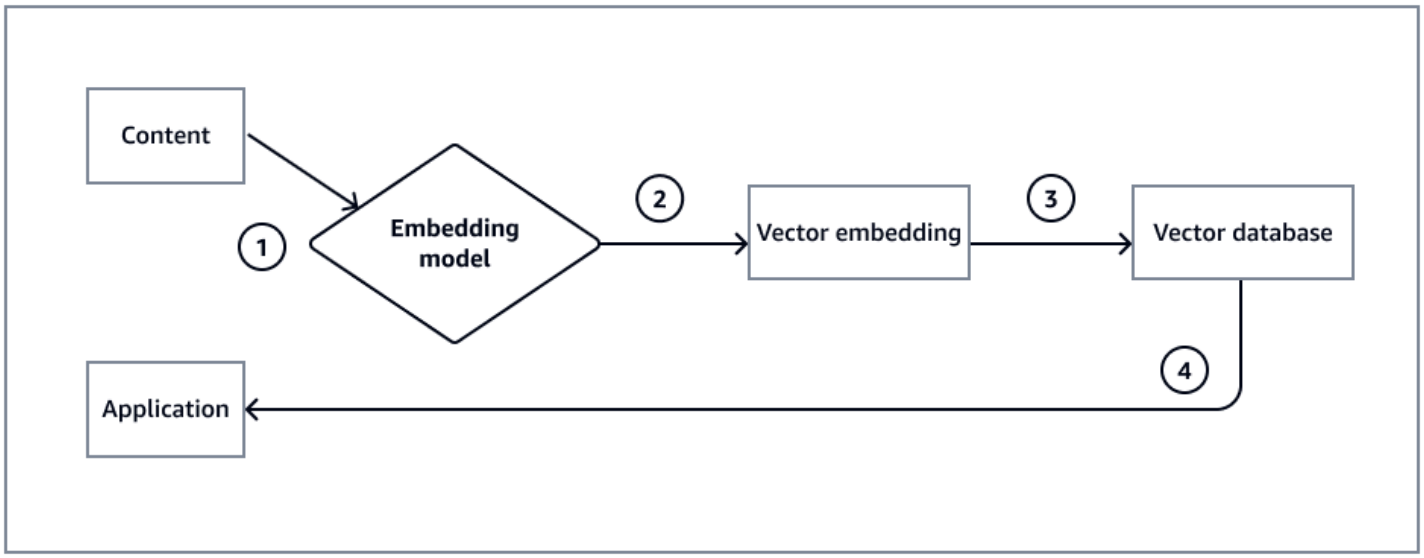
Les bases de données vectorielles gèrent la conversion et le stockage des données de la manière suivante :

- Les objets (tels que les fichiers audio, les images et les fichiers texte) sont convertis en vecteurs à l'aide de modèles d'intégration.
- Les vecteurs sont stockés dans des formats de données spécialisés.
- Les bases de données vectorielles permettent des recherches rapides de similarité.

Les bases de données vectorielles offrent plusieurs avantages essentiels par rapport aux bases de données traditionnelles, ce qui les rend particulièrement adaptées aux défis actuels en matière de données. Ils sont spécifiquement optimisés pour les opérations vectorielles et gèrent efficacement les données de grande dimension. Ils se spécialisent également dans les recherches de similarité auxquelles les bases de données traditionnelles ont du mal à faire face. Au-delà de ces fonctionnalités de base, les bases de données vectorielles sont conçues pour répondre aux demandes évolutives des applications de machine learning et d'IA générative. Ils excellent dans le stockage vectoriel à grande échelle et utilisent l'informatique distribuée pour équilibrer les charges de travail sur plusieurs nœuds. Cela garantit l'évolutivité et les performances à mesure que les volumes de données augmentent.

Le schéma suivant montre une implémentation de RAG :

1. Le contenu, tel que les documents, les PDF ou les fichiers texte, est introduit dans le modèle d'intégration sous forme de données brutes à traiter.
2. Le modèle d'intégration transforme les données brutes en vecteurs numériques, qui représentent la signification sémantique du contenu.
3. Les intégrations vectorielles générées sont stockées dans une base de données vectorielle optimisée pour le stockage et la récupération de vecteurs de grande dimension.
4. Les applications peuvent désormais interroger la base de données vectorielle en réponse à des cas d'utilisation tels que la recherche sémantique et la recommandation de contenu.



Le choix d'une base de données vectorielle inappropriée pour une solution RAG peut entraîner des difficultés et des limites importantes, notamment les suivantes :

- Mauvaises performances des requêtes
- Les goulets d'étranglement liés à l'évolutivité
- Les défis de l'ingestion de données
- Absence de fonctionnalités avancées, telles que le filtrage et le classement
- Difficultés d'intégration avec d'autres systèmes
- Problèmes de persistance et de durabilité
- Problèmes de simultanéité et de cohérence dans les environnements comportant plusieurs utilisateurs
- Coûts de licence plus élevés ou dépendance vis-à-vis d'un fournisseur
- Soutien et ressources communautaires limités
- Risques potentiels en matière de sécurité et de conformité

Options de base de données vectorielles

AWS propose une gamme variée de solutions de bases de données vectorielles pour répondre à différents cas d'utilisation et exigences dans les applications d'IA générative. Ces options peuvent être classées globalement en services de base de données individuels et en offres de services gérés, chacune présentant des caractéristiques et des avantages distincts. Comprendre ces options est essentiel pour les entreprises qui cherchent à mettre en œuvre efficacement des fonctionnalités de recherche vectorielle tout en maintenant des performances, une évolutivité et une rentabilité optimales.

Pour plus d'informations sur les solutions de base de données vectorielles, consultez les sections suivantes :

- Options de base de données vectorielles individuelles
- Option de service géré
- Choisir la bonne base de données vectorielle

Options de base de données vectorielles individuelles

Les options de base de données vectorielles individuelles disponibles AWS incluent Amazon Kendra, Amazon OpenSearch Service, AmazonRDS for PostgreSQL avec pgvector, Amazon MemoryDB, Amazon DocumentDB, Amazon DocumentDB, AmazonNeptune Analytics et Amazon S3 Vector. (Extension open source, pgvector ajoute la possibilité de stocker et de rechercher des intégrations vectorielles générées par ML.) Ces solutions proposent différentes approches de la recherche vectorielle, permettant aux entreprises de choisir en fonction de leur infrastructure existante, de leurs exigences techniques et de leurs cas d'utilisation spécifiques.

Amazon Kendra

Amazon Kendra est un service de recherche intelligent destiné aux entreprises qui utilise le traitement du langage naturel et des algorithmes d'apprentissage automatique avancés pour renvoyer des réponses spécifiques aux questions de recherche à partir de vos données. Amazon Kendra simplifie la mise en œuvre de la fonctionnalité de recherche, ce qui en fait une solution backend efficace pour les applications d'IA générative.

Les autres fonctionnalités clés d'Amazon Kendra sont les suivantes :

- Connexions natives à plus de 40 sources de données
- Fonctionnalités intégrées de préparation des données
- Configuration rapide ne nécessitant pas d'expertise technique approfondie

Les avantages d'Amazon Kendra sont les suivants :

- Traitement automatisé des données (découpage, ingestion, extraction)
- De puissantes options de personnalisation :
  - Recherche par facettes
  - Analyses de recherche
  - Optimisation de la pertinence des recherches
- Accès programmatique simple via le AWS SDK pour Python (Boto3)

Pour plus d'informations, consultez la section Avantages d'Amazon Kendra dans la documentation Amazon Kendra.

Amazon OpenSearch Service

Amazon OpenSearch Service est un service géré qui vous aide à déployer, exploiter et dimensionner des clusters OpenSearch de services dans le AWS Cloud.

Les principales fonctionnalités du OpenSearch Service sont les suivantes :

- Moteur de recherche et d'analyse open source
- Architecture distribuée
- Traitement des données en temps réel

Certains avantages de l'utilisation du OpenSearch Service sont les suivants :

- Scalabilité horizontale
- RESTful Support de l'API
- Gère les données structurées et non structurées
- Analyse des données en temps réel
- Adapté à différentes tailles de déploiement

Pour plus d'informations, consultez la section Fonctionnalités d'Amazon OpenSearch Service dans la documentation du OpenSearch service.

Amazon RDS pour PostgreSQL avec pgvector

Amazon RDS for PostgreSQL avec pgvector AWS associe le service de base de données relationnelle gérée à l'extension de traitement vectoriel de PostgreSQL. Cette combinaison permet aux entreprises de stocker et d'interroger des vecteurs de grande dimension tout en gérant Amazon RDS. La solution est particulièrement adaptée aux applications d'intelligence artificielle génératives qui nécessitent des opérations vectorielles en temps réel sans la surcharge liée à la gestion de l'infrastructure de base de données.

Les principaux avantages d'Amazon RDS pour PostgreSQL avec pgvector sont les suivants :

- Haute disponibilité
- Basculement automatique
- Rentable (pay-per-use)
- Surveillance intégrée
- Intégration de données vectorielles en temps réel

Pour plus d'informations, consultez les avantages d'Amazon RDS dans la documentation Amazon RDS.

Amazon DocumentDB

Amazon DocumentDB (compatible avec MongoDB) est une base de données de documents qui offre des fonctionnalités de recherche vectorielle natives dans les versions 5.0 et ultérieures. Il combine la flexibilité du stockage de documents basé sur JSON avec la recherche vectorielle, prenant en charge à la fois les méthodes d'indexation hiérarchiques navigables Small World (HNSW) et Inverted File Flat (). IVFFlat

Les principales fonctionnalités d'Amazon DocumentDB sont les suivantes :

- Stockez et indexez des vecteurs jusqu'à 2 000 dimensions (jusqu'à 16 000 dimensions sans indexation)
- Temps de réponse en millisecondes pour les recherches de similarité vectorielle

- Support pour les mesures de distance entre produits euclidiens, cosinus et points
- Intégration parfaite avec les applications compatibles MongoDB existantes

Utilisez Amazon DocumentDB dans les situations suivantes :

- Pour les applications qui utilisent déjà MongoDB APIs et qui ont besoin de fonctionnalités de recherche vectorielle
- Pour les cas d'utilisation nécessitant des structures de données documentaires flexibles associées à une recherche sémantique
- Pour les scénarios nécessitant à la fois des requêtes documentaires traditionnelles et des recherches de similarité vectorielle
- Pour les applications proposant des recommandations de produits, des personnalisations, des assistants de chat et des services de détection des fraudes

Pour plus d'informations, consultez la section Recherche vectorielle pour Amazon DocumentDB dans la documentation Amazon DocumentDB.

Amazon MemoryDB

Amazon MemoryDB est une base de données en mémoire compatible Redis qui fournit les performances de recherche vectorielle les plus rapides parmi les bases de données vectorielles les plus populaires sur. AWS Il fournit des latences de requête inférieures à la milliseconde avec une durabilité dans les zones de multidisponibilité.

Les principales fonctionnalités de MemoryDB sont les suivantes :

- Stockez les données des applications et des millions de vecteurs dans une seule base de données
- Temps de réponse aux requêtes et aux mises à jour à un chiffre en millisecondes
- Les taux de rappel les plus élevés pour les performances les plus rapides sur AWS
- Support pour un maximum de 32 768 dimensions par vecteur
- Fonctionnalités de recherche sémantique et de mise en cache en temps réel

Utilisez MemoryDB dans les situations suivantes :

- Pour les applications en temps réel qui nécessitent une latence très faible (inférieure à 10 ms)

- Pour les charges de travail à haut débit avec des millions de demandes par jour
- Pour les cas d'utilisation tels que les moteurs de recommandation en temps réel, la mise en cache sémantique et la détection d'anomalies
- Pour les applications nécessitant à la fois un stockage de données en mémoire et des fonctionnalités de recherche vectorielle

Pour plus d'informations, consultez la section Recherche vectorielle dans la documentation de MemoryDB.

Amazon Neptune Analytics

Amazon Neptune Analytics est un moteur d'analyse graphique qui offre des fonctionnalités de recherche vectorielle natives, ce qui le rend idéal pour les cas d'utilisation de Graph Retrieval Augmented Generation (GraphRag). Il combine la recherche de similarité vectorielle avec des traversées de graphes et des algorithmes.

Les principales fonctionnalités de Neptune Analytics sont les suivantes :

- Analysez des dizaines de milliards de relations en quelques secondes
- Combinez la recherche vectorielle avec des algorithmes graphiques (recherche de trajectoire, détection de communauté, centralité)
- Support pour les applications GraphRag avec des connaissances topologiques
- Jusqu'à 80 fois plus rapide que les solutions d'analyse graphique existantes
- Intégration à Amazon Bedrock pour une gestion complète de GraphRag

Utilisez Neptune Analytics dans les situations suivantes :

- Pour les applications GraphRag qui nécessitent des graphes de connaissances avec intégrations vectorielles
- Pour les cas d'utilisation qui nécessitent de parcourir des relations complexes parallèlement à la similarité vectorielle
- Pour les applications qui nécessitent des réponses explicables de l'IA avec un contexte relationnel
- Pour des scénarios tels que les vues à 360 degrés des clients, les réseaux de détection des fraudes et la découverte de connaissances

Pour plus d'informations, consultez la documentation Amazon Neptune Analytics.

Amazon S3 Vectors

Amazon S3 Vectors est le premier magasin d'objets cloud doté AWS de capacités natives de stockage vectoriel et de requêtes. Il fournit un stockage vectoriel spécialement conçu et optimisé en termes de coûts pour les applications d'IA nécessitant une grande échelle.

Les principales fonctionnalités d'Amazon S3 Vectors sont les suivantes :

- Stockage pour jusqu'à 2 milliards de vecteurs par index avec prise en charge de jusqu'à 10 000 index par compartiment de vecteurs
- Latence de requête inférieure à 100 ms optimisée pour le stockage à long terme et les modèles d'accès peu fréquents
- Jusqu'à 90 % de réduction des coûts pour les opérations vectorielles par rapport aux bases de données vectorielles spécialisées
- Architecture sans serveur avec mise à l'échelle automatique et durabilité de 99,999999999 % (11 s)

Utilisez les vecteurs Amazon S3 dans les situations suivantes :

- Pour les applications qui nécessitent le stockage de milliards de vecteurs à un coût minimal
- Pour les charges de travail qui tolèrent une latence de requête inférieure à une seconde (100 ms ou plus) au lieu de 10 ms
- Pour la conservation des vecteurs à long terme et les cas d'utilisation de l'archivage
- Pour les applications RAG avec des modèles de récupération peu fréquents
- Pour les entreprises qui privilégient l'économie du stockage par rapport à une latence extrêmement faible

Amazon S3 Vectors s'intègre nativement aux bases de connaissances Amazon Bedrock et fonctionne bien dans les architectures à plusieurs niveaux avec Amazon Service. OpenSearch Vous pouvez utiliser les vecteurs Amazon S3 pour le stockage à froid et utiliser le OpenSearch service pour les requêtes chaudes.

Pour plus d'informations, consultez la section Utilisation des vecteurs S3 et des compartiments vectoriels dans la documentation Amazon S3.

Option de service géré

Amazon Bedrock Knowledge Bases représente l'approche AWS entièrement gérée de la mise en œuvre de bases de données vectorielles. La flexibilité des options de stockage du service, combinée à ses fonctionnalités de gestion automatisée, le rend particulièrement utile pour les entreprises qui cherchent à mettre en œuvre le RAG sans gérer une infrastructure complexe.

Avec les bases de connaissances Amazon Bedrock, vous pouvez créer, gérer et consulter des bases de connaissances qui améliorent vos modèles de base à l'aide de RAG. Ce service simplifie le processus complexe de mise en œuvre de RAG en gérant l'intégralité du pipeline d'ingestion, de vectorisation et de récupération des données.

Les principaux avantages des bases de connaissances Amazon Bedrock sont les suivants :

- Traitement des données simplifié
  - Ingestion et segmentation automatiques des données
  - Extraction de texte intégrée à partir de plusieurs formats de fichiers
  - Génération d'intégrations vectorielles gérées
  - Extraction et indexation automatiques des métadonnées
- Implémentation rationalisée du RAG
  - Stratégies de récupération préconfigurées
  - Optimisation automatique des fenêtres contextuelles
  - Réglage de la pertinence intégré
  - Fonctionnalités de recherche sémantique prêtes à l'emploi
- Sécurité et gouvernance
  - Contrôles intégrés Gestion des identités et des accès AWS (IAM)
  - Chiffrement des données au repos et en transit
  - Prise en charge de VPC
  - Journalisation des audits avec AWS CloudTrail

Les bases de connaissances Amazon Bedrock prennent en charge plusieurs options de boutiques vectorielles, notamment :

- Amazon Aurora PostgreSQL avec pgvector

- Amazon Neptune Analytics
- Amazon EMR sans serveur
- Amazon S3 Vectors
- Pinecone
- Redis Enterprise Cloud

Ce service géré gère l'ingestion, la vectorisation et la récupération automatisées des données. Cela simplifie les implémentations de RAG.

Pour obtenir des informations détaillées sur chaque boutique vectorielle prise en charge, consultez la documentation des bases de connaissances Amazon Bedrock.

Choisir la bonne base de données vectorielle

Sélectionnez votre base de données vectorielle en fonction de ces facteurs de décision clés :

- Si vous avez besoin d'une base de données de documents compatible avec MongoDB avec recherche vectorielle, choisissez Amazon DocumentDB. C'est idéal lorsque votre application utilise MongoDB APIs et que vous souhaitez ajouter des fonctionnalités de recherche sémantique sans gérer une infrastructure vectorielle distincte.
- Si vous avez besoin d'une latence très faible pour les applications en temps réel, choisissez Amazon MemoryDB. Cela permet d'obtenir les performances de recherche vectorielle les plus rapides AWS avec des temps de réponse inférieurs à la milliseconde. Il est idéal pour les moteurs de recommandation en temps réel et les applications à haut débit.
- Si vous avez besoin de représentations de connaissances basées sur des graphes avec recherche vectorielle, choisissez Amazon Neptune Analytics. C'est la solution idéale pour les applications GraphRag qui doivent traverser des relations complexes et effectuer des requêtes basées sur des graphes parallèlement à des recherches vectorielles, fournissant ainsi des réponses explicables par l'IA.
- Si vous devez associer des requêtes relationnelles à une recherche vectorielle, choisissez Amazon Aurora PostgreSQL avec pgvector. Cette option est idéale lorsque votre application nécessite à la fois des opérations SQL traditionnelles et des recherches de similarité vectorielle au sein de la même base de données.

- Si vous avez besoin de requêtes à haut débit avec une latence inférieure à 10 ms, choisissez Amazon Service. OpenSearch Il excelle dans la gestion des requêtes à haute fréquence et des applications en temps réel et inclut de récentes améliorations de l'accélération du GPU.
- Si vous devez stocker des milliards de vecteurs de manière rentable, optez pour Amazon S3 Vectors. Cette option permet de réaliser jusqu'à 90 % d'économies et est idéale pour les applications dont les modèles de récupération sont peu fréquents (de quelques minutes à quelques heures entre les requêtes) et qui peuvent tolérer une latence inférieure à 100 ms.
- Si vous avez besoin d'une recherche en texte intégral associée à une recherche vectorielle, choisissez Amazon OpenSearch Service. Cette option combine de puissantes fonctionnalités de recherche en texte intégral et de recherche vectorielle sur une seule plateforme.

Comparaison de bases de données vectorielles

AWS propose plusieurs approches pour implémenter des fonctionnalités de recherche vectorielle, allant des bases de données vectorielles individuelles aux bases de connaissances Amazon Bedrock, qui est un service entièrement géré. Lors de l'évaluation de ces options, les entreprises doivent prendre en compte divers aspects, notamment l'architecture, l'évolutivité, les capacités d'intégration, les caractéristiques de performance et les fonctionnalités de sécurité.

Bases de données vectorielles individuelles

Le tableau suivant fournit un aperçu des principales fonctionnalités de plusieurs solutions de base de données vectorielles AWS individuelles, en mettant l'accent sur leurs architectures, leurs capacités de mise à l'échelle, leurs intégrations de sources de données et leurs caractéristiques de performance.

Fonctionnalité	Amazon Kendra	Amazon OpenSearch Service	Amazon RDS pour SQLwith Postgre/pgvector	Amazon DocumentDB	Amazon MemoryDB	Amazon Neptune Analytics	Amazon S3 Vectors
Cas d'utilisation principal	Recherche d'entreprise et RAG	Recherche et analyse distribuées	Base de données relationnelle avec support vectoriel	Base de données de documents avec recherche vectorielle	Recherche vectorielle en mémoire en temps réel	Analyse graphique avec recherche vectorielle	Stockage vectoriel optimisé en termes de coûts
Architecture	Entièrement géré	Cluster distribué	Base de données relationnelle	Orienté vers les documents	Base de données en mémoire	Moteur d'analyse graphique	Stockage d'objets sans serveur

Modèle de données	Basé sur des documents	Documents JSON	Tables relationnelles	Documents JSON	Valeur-clé avec JSON	Graphe de propriétés	Stockage d'objets
Dimensions du vecteur	Géré automatiquement	Jusqu'à 16 000	Configurable	Jusqu'à 2 000 (indexé) ; 16 000 (non indexé)	Jusqu'à 32 768	Configurable	Jusqu'à 4 096
Méthodes d'indexation	Automatique	NSW, FIV	NSW, IVFFlat	NSW, IVFFlat	HNSW	Graphe et vecteur natifs	Automatique
Métriques de distance	Automatique	Cosine, Euclidean, produit à points	Cosine, produit euclidien, produit intérieur	Cosine, Euclidean, produit à points	Cosine, produit euclidien, produit intérieur	Cosinus euclidien	Cosinus euclidien
Latence des requêtes	Sous-seconde	Moins de 10 ms (accéléré par le GPU)	10 à 100 ms	Milliseconde	Moins d'une milliseconde	Sous-seconde	Moins de 100 ms
Modèle d'échelle	Automatique	Horizontal (ajouter des nœuds)	Répliques verticales et répliques en lecture	Horizontal (ajouter des instances)	Vertical et répliques	Automatique	Automatique (sans serveur)

Vecteurs maximaux	Gérées	Des milliards (en fonction du cluster)	Millions (en fonction de l'instance)	Des millions par collection	Des millions par base de données	Milliards	2 milliards par indice ; 10 000 indices par compartiment
Débit	Élevée	Très élevé (milliers de QPS)	Medium	Élevée	Très élevé (millions de demandes par jour)	Élevée	Moyen (optimisé pour les requêtes peu fréquentes)
Durabilité des données	99,999999999 % (11 9 s)	Configurable avec des répliques	99,99 % (Multi-AZ)	99,99 % (Multi-AZ)	99,99 % (Multi-AZ)	99,99 %	99,999999999 % (11 9 s)
Modèle de cohérence	Éventuel	Éventuel (configurable)	Fort (ACIDE)	Éventuel	Solide	Solide	Solide
Fonctionnalités supplémentaires	40 connecteurs de données ou plus, NLP	Recherche en texte intégral, analyses, tableaux de bord	Requêtes SQL, transactions ACID	Compatibilité avec l'API MongoDB	Compatibilité avec l'API Redis, mise en cache	Algorithmes graphiques, traversées	Intégration avec Amazon S3, politiques de cycle de vie

Modèle de tarification	Paiement par requête et par stockage	Heures d'ouverture et stockage des instances	Heures d'ouverture et stockage des instances	Heures d'ouverture et stockage des instances	Heures d'ouverture et stockage des instances	Unités de capacité et de stockage	Stockage, requêtes et transfert de données
Optimisation des coûts	Basé sur l'utilisation	Instances réservées, auto-scaling	Instances réservées, Aurora Serverless	Instances réservées	Instances réservées	Scalabilité automatique	Jusqu'à 90 % d'économies par rapport à une solution spécialisée DBs
Idéal pour	Recherche d'entreprise avec configuration minimale	Requêtes à haut débit et à faible latence	Charges de travail SQL et vectoriel les hybrides	Applications compatibles avec MongoDB nécessitant des vecteurs	Applications en temps réel à très faible latence	GraphRag et graphes de connaissances	Stockage rentable et à long terme
Modèle de requête idéal	Recherches fréquentes dans les entreprises	Requêtes en temps réel à haute fréquence	Requêtes SQL et vectoriel les mixtes	Requêtes de documents avec recherche sémantique	Des millions de demandes par jour	Parcours de graphes avec recherche vectorielle	Demandes peu fréquentes (de quelques minutes à quelques heures)

Complexité de configuration	Faible (entièrement géré)	Moyen (configuration en cluster)	Moyen (configuration de l'extension)	Moyen (configuration en cluster)	Moyen (configuration en cluster)	Faible (entièrement géré)	Faible (sans serveur)
Expertise d'équipe requise	Minimale	OpenSearch ou Elasticsearch	PostgreSQL, SQL	MongoDB	Redis	Bases de données graphiques	Amazon S3, concepts vectoriels de base

Service géré — Bases de connaissances Amazon Bedrock

Les bases de connaissances Amazon Bedrock fournissent une solution entièrement gérée avec plusieurs options de stockage vectoriel. Le tableau suivant compare ces options de stockage.

Fonctionnalité	Aurora Postgre pgvector SQLwith	Neptune Analytics	OpenSearch Service sans serveur	Vecteurs Amazon S3	Pomme de pin	RedisEnterprise Cloud
Cas d'utilisation principal	Base de données relationnelle avec vecteur RAG	Recherche vectorielle basée sur des graphes pour GraphRag	Gestion des connaissances RAG	RAG vectoriel optimisé en termes de coûts	Recherche vectorielle performante	Recherche vectorielle en mémoire
Architecture	Relationnel entièrement géré	Analyses graphiques entièrement gérées	Entièrement géré sans serveur	Stockage d'objets sans serveur	Cloud hybride entièrement géré	Entièrement géré en mémoire

Modèle de données	Tables relationnelles	Graphe de propriétés	Documents JSON	Stockage d'objets	Vecteurs spécialement conçus	Valeur-clé avec vecteurs
Stockage vectoriel	Grâce à l'extension pgvector	Vecteurs graphiques natifs	À travers le OpenSearch moteur	Stockage vectoriel natif d'Amazon S3	Base de données vectorielle native	Vecteurs en mémoire
Intégration avec Amazon Bedrock	Natif	Natif	Natif	Natif	Natif	Natif
Ingestion automatique	Oui (via Amazon Bedrock)	Oui (via Amazon Bedrock)	Oui (via Amazon Bedrock)	Oui (via Amazon Bedrock)	Oui (via Amazon Bedrock)	Oui (via Amazon Bedrock)
Vectorisation automatique	Oui (via Amazon Bedrock)	Oui (via Amazon Bedrock)	Oui (via Amazon Bedrock)	Oui (via Amazon Bedrock)	Oui (via Amazon Bedrock)	Oui (via Amazon Bedrock)
Mise à l’échelle	Mise à l'échelle automatique (Aurora Serverless)	Mise à l'échelle automatique des graphiques	Sans serveur automatique	Automatique (milliards de vecteurs)	Capsules à dimensionnement automatique	Clusters à dimensionnement automatique
Les performances des requêtes	Haut pour le mode relationnel ou vectoriel	Haut pour les vecteurs de graphes	Élevée	Moyen (latence de 100 ms ou plus)	Très élevée	Très élevée

Vecteurs maximaux	Millions (en fonction de l'instance)	Milliards	Milliards	2 milliards par indice	Milliards	Millions (en fonction de la mémoire)
Fonctionnalités supplémentaires	Requêtes SQL, transactions ACID	Algorithmes graphiques, traversées	Recherche en texte intégral, analyse	Cycle de vie d'Amazon S3, hiérarchisation	Filtrage des métadonnées, espaces de noms	Structures de données Redis, mise en cache
Optimisation des coûts	Modéré (Aurora Serverless)	Modéré (unités de capacité)	Élevé (sans serveur, pay-per-use)	Très élevé (jusqu'à 90 % d'économies)	Modéré (tarification basée sur les dosettes)	Faible (mémoire haut de gamme)
Idéal pour	Charges de SQL/vecteur travail hybrides	Graphes de connaissances connectés	Texte intégral avec recherche vectorielle	Vecteurs à long terme et à accès peu fréquent	Recherche vectorielle en temps réel à grande échelle	Besoins de latence très faible
Modèle de requête idéal	Requêtes SQL et vectorielles mixtes	Parcours de graphes à l'aide de vecteurs	Recherches fréquentes avec outils d'analyse	Récupération peu fréquente (de quelques minutes à quelques heures)	Requêtes en temps réel à haute fréquence	Des millions de demandes par seconde

Configuration avec Amazon Bedrock	Simple (géré par Amazon Bedrock)	Simple (géré par Amazon Bedrock)	Simple (géré par Amazon Bedrock)	Simple (géré par Amazon Bedrock)	Simple (géré par Amazon Bedrock)	Simple (géré par Amazon Bedrock)
Résidence des données	Régions AWS	Régions AWS	Régions AWS	Régions AWS	Multi-cloud (AWS et autres)	Multi-cloud (AWS et autres)
Modèle de tarification	Heures d'ouverture et stockage des instances	Unités de capacité et de stockage	Calcul et stockage (sans serveur)	Stockage, requêtes et transfert	Heures d'ouverture et stockage du pod	Heures d'ouverture et stockage des nœuds

Choisir entre des options individuelles et des options gérées

Considération	Choisissez une base de données vectorielle individuelle	Choisissez les bases de connaissances Amazon Bedrock (gérées)
Implémentation du RAG	Vous voulez un contrôle total sur le pipeline RAG	Vous voulez un RAG entièrement géré avec une configuration minimale
Personnalisation	Vous avez besoin d'une logique de récupération et d'un prétraitement personnalisés	Les modèles RAG standard répondent à vos besoins
Infrastructures existantes	La base de données est déjà déployée	Vous reprenez un nouveau départ ou souhaitez une gestion simplifiée
Expertise de l'équipe	Votre équipe possède une expertise en administration de bases de données	Vous préférez vous concentrer sur la logique des applications plutôt que sur l'infrastructure

Complexité d'intégration	Vous avez besoin d'une intégration approfondie avec les systèmes existants	Vous souhaitez une intégration rapide avec les modèles Amazon Bedrock
Frais généraux d'exploitation	Vous pouvez gérer les opérations de base de données	Vous souhaitez AWS gérer les opérations
Structure des coûts	Vous préférez la tarification directe des bases de données	Vous préférez une tarification Amazon Bedrock unifiée
Délai de mise sur le marché	Vous avez le temps de procéder à une mise en œuvre personnalisée	Vous avez besoin d'un déploiement rapide

Comparaisons de coûts et considérations

Comprendre la structure des coûts des différentes options de base de données vectorielles est essentiel pour prendre des décisions de mise en œuvre éclairées. Le tableau suivant décrit certaines considérations financières clés pour diverses solutions de base de données vectorielles, y compris les bases de données individuelles et les services gérés. Chaque option comporte des facteurs de tarification distincts qui peuvent avoir un impact sur le coût total de possession (TCO), qu'il s'agisse des pay-as-you-go modèles, de l'infrastructure ou des coûts opérationnels.

Base de données vectorielle	Modèle de coûts	Considérations relatives aux coûts
Amazon Kendra	Payez au fur et à mesure, en fonction des demandes	Les coûts peuvent varier en fonction du nombre de requêtes et de la quantité de données indexées. Des frais supplémentaires peuvent s'appliquer pour le stockage et le transfert de données. Pour plus d'informations, consultez les tarifs d'Amazon Kendra.
Amazon OpenSearch Service	Payez au fur et à mesure, en fonction des heures d'utilisation de l'instance et du stockage	Les coûts incluent les heures d'instance, le stockage (volumes Amazon EBS), le transfert de données et le UltraWarm stockage optionnel . Les instances réservées peuvent permettre de réaliser jusqu'à 30 % d'économies. Les instances accélérées par GPU offrent un meilleur rapport prix/performances pour les charges de travail vectoriel les. Pour plus d'informations,

		consultez les tarifs d'Amazon OpenSearch Service.
Open source OpenSearch	Open source, sans frais directs (vous n'avez pas à payer pour télécharger ou utiliser le logiciel, et il n'y a aucun coût de licence)	Les coûts incluent l'infrast ructure (tels que les serveurs et le stockage) et les coûts opérationnels (tels que la maintenance et la surveilla nce). Organisations doivent prévoir un budget pour le personnel nécessaire à la gestion et à la maintenance de l'infrastructure.
Amazon RDS pour PostgreSQ L avec pgvector	Payez au fur et à mesure, en fonction de votre consommat ion	Les coûts incluent les types d'instances de base de données, le stockage, le transfert de données et les sauvegardes. Des frais supplémentaires peuvent s'appliquer pour le transfert de données, les types d'instanc es et le stockage au-delà du niveau AWS gratuit. Pour plus d'informations, consultez Tarification d'Amazon RDS.

Amazon DocumentDB	Payez au fur et à mesure, en fonction des heures d'utilisation de l'instance et du stockage	Les coûts incluent les heures d'instance, le stockage (Go par mois), les I/O demandes, le stockage de sauvegarde et le transfert de données. Les clusters élastiques permettent une mise à l'échelle dynamique. Instances réservées disponibles pour l'optimisation des coûts. Pour plus d'informations, consultez la tarification d'Amazon DocumentDB.
Amazon MemoryDB	Payez au fur et à mesure, en fonction des heures d'ouverture des nœuds et du stockage des données	Les coûts incluent les heures de nœud (par type de nœud), le stockage des données (Go par heure), le stockage des instantanés et le transfert de données. Les nœuds réservés peuvent permettre de réaliser jusqu'à 55 % d'économies. Optimisé pour les charges de travail à haut débit et à faible latence. Pour plus d'informations, consultez la tarification d'Amazon MemoryDB.

Amazon Neptune Analytics	Payez au fur et à mesure, en fonction des unités de capacité	Les coûts incluent les unités de capacité Neptune (NCUs), le stockage (Go par mois) et le transfert de données. Mise à l'échelle automatiq ue en fonction de la charge de travail, sans engagemen t initial. Un minimum de 128 NCUs est requis. Pour plus d'informations, consultez la tarification d'Amazon Neptune.
Amazon S3 Vectors	Payez au fur et à mesure, en fonction de l'espace de stockage et des demandes	Les coûts incluent le stockage (Go par mois), les requêtes PUT et GET, la gestion des index vectoriels et le transfert de données. Permet de réaliser jusqu'à 90 % d'économies par rapport aux bases de données vectoriel les spécialisées. Les politique s de hiérarchisation intellige nte et de cycle de vie Amazon S3 sont disponibles pour une optimisation supplémentaire. Pour plus d’informations, consultez Tarification Amazon S3.

Bases de connaissances Amazon Bedrock	Payez au fur et à mesure, en fonction de votre consommation	Les coûts peuvent varier en fonction de l'utilisation de la base de connaissances et de services supplémentaires tels qu'Amazon OpenSearch Serverless. Des frais supplémentaires peuvent s'appliquer pour le stockage des données, le transfert de données et les fonctionnalités supplémentaires. Pour plus d'informations sur les tarifs, consultez la section Tarification d'Amazon OpenSearch Service.

Cas d'utilisation de bases de données vectori

Les exemples suivants montrent comment différentes options de base de données vectorielles peuvent être utilisées efficacement pour améliorer la gestion des connaissances, améliorer l'efficacité opérationnelle et obtenir de meilleurs résultats commerciaux. Ces cas d'utilisation illustrent les applications pratiques des solutions de base de données vectorielles évoquées plus haut dans ce guide et fournissent un aperçu de leurs performances et avantages réels.

Gestion des connaissances avec Amazon Kendra

Problème avec le client — L'un des plus grands entrepreneurs généraux du Japon était confronté à une baisse de son personnel expérimenté. L'entreprise avait besoin d'un moyen de transférer efficacement les connaissances et les compétences du personnel expérimenté à la jeune génération. Ils avaient besoin d'une solution pour saisir et diffuser les connaissances complexes en ingénierie de la construction et les expériences passées.

AWS solution — Pour résoudre ce problème, le client s'est tourné vers Amazon Kendra, une solution d'intelligence artificielle capable de gérer rapidement et précisément sa base de connaissances interne et d'autoriser les requêtes en langage naturel. Grâce à Amazon Kendra, les employés peuvent désormais trouver les informations dont ils ont besoin beaucoup plus rapidement, ce qui améliore la productivité et facilite le transfert de connaissances entre le personnel expérimenté et le personnel plus jeune.

Impact — En mettant en œuvre un chatbot génératif basé sur l'IA alimenté par Amazon Kendra, l'entreprise a créé une plateforme de connaissances unifiée. Le chatbot permet aux employés d'accéder rapidement aux connaissances techniques et aux expériences passées en génie de la construction. Cette solution a considérablement amélioré l'efficacité du transfert de connaissances et des processus de prise de décision au sein de l'organisation, contribuant ainsi à garantir que la précieuse expertise est préservée et facilement accessible à tous les employés. Le coût de cette solution peut varier en fonction de votre utilisation et de votre configuration. Pour une estimation détaillée des coûts, consultez le Calculateur de tarification AWS. Pour l'estimation des coûts des bases de données vectorielles, consultez la section Comparaison des coûts et considérations de ce guide ou consultez les tarifs d'Amazon Kendra.

Pour plus d'informations sur les autres cas d'utilisation, consultez les clients d'Amazon Kendra.

Analyses en temps réel avec OpenSearch Serverless

Problème client — Un fournisseur de services financiers de premier plan a dû relever le défi de gérer un énorme écosystème de données. Elle a traité 300 millions d'autorisations et 90 milliards de transactions par an, accumulant environ 1,1 pétaoctet (Po) de données. Le système existant, qui dessert 300 000 utilisateurs ayant besoin d'accéder à plus de 6 000 rapports, avait besoin d'être modernisé pour assurer une cohérence globale et permettre une prise de décision en temps réel.

AWS solution — L'architecture de la solution a utilisé des modèles de base disponibles via Amazon Bedrock (notamment Anthropic, Sonnet 3, Sonnet 3.5 et Haiku) pour le traitement du langage naturel. Le client a choisi OpenSearch Serverless comme base de données vectorielle pour son évolutivité supérieure et sa capacité à gérer efficacement l'énorme volume de données. Cette architecture a permis le traitement fluide des requêtes complexes et la génération de rapports dynamiques.

Impact — La mise en œuvre a permis d'augmenter la productivité de 50 % en éliminant le besoin de générer manuellement plus de 100 tableaux de bord de business intelligence. Les utilisateurs peuvent désormais générer des rapports par le biais de requêtes en langage naturel avec des temps de réponse compris entre 20 et 40 secondes. Le coût de cette solution peut varier en fonction de votre utilisation et de votre configuration. Pour une estimation détaillée des coûts, consultez le Calculateur de tarification AWS. Pour l'estimation des coûts des bases de données vectorielles, consultez la section Comparaison des coûts et considérations de ce guide ou consultez les tarifs d'Amazon OpenSearch Service. Pour plus d'informations sur les autres cas d'utilisation par les clients, consultez Amazon OpenSearch Serverless.

Prochaines étapes et ressources

Après avoir examiné ce guide, considérez les actions suivantes pour passer de la compréhension à la mise en œuvre :

1. Évaluez vos besoins actuels :
   - Évaluez votre infrastructure de base de données et votre expertise existantes.
   - Documentez vos exigences spécifiques en matière de recherche vectorielle.
   - Définissez vos objectifs de performance, de mise à l'échelle et de coûts.
2. Choisissez l'une des options suivantes pour tester les options de base de données vectorielles :
   - Option 1 : Configurez une preuve de concept à l'aide de votre solution de base de données vectorielle préférée.
   - Option 2 : testez des exemples de jeux de données dans les bases de connaissances Amazon Bedrock. Essayez l'expérience de création rapide pour une base de connaissances Amazon Bedrock. Par exemple, consultez la section Création rapide d'une base de connaissances Aurora PostgreSQL pour Amazon Bedrock dans la documentation Aurora.
3. Passez en revue les ressources supplémentaires.
4. Obtenez l'aide d'un expert :
   - Contactez votre Compte AWS équipe ou vos architectes de AWS solutions pour obtenir des conseils de mise en œuvre.
   - Collaborez avec AWS des partenaires spécialisés dans les bases de données vectorielles.
5. Planifiez votre déploiement en production :
   - Créez une stratégie de migration en cas de migration à partir de bases de données existantes.
   - Développez un plan de mise à l'échelle pour la solution que vous avez choisie.
   - Concevez vos procédures de surveillance et de maintenance.

Ressources

Les ressources suivantes peuvent vous aider à choisir une base de données vectorielle.

AWS articles de blog

- Accélérez le développement de vos applications d'IA générative avec Amazon Bedrock Knowledge Bases, Quick Create et Amazon Aurora Serverless
- Explication des fonctionnalités de la base de données vectorielle d'Amazon OpenSearch Service
- Explorez en profondeur les magasins de données vectorielles à l'aide des bases de connaissances Amazon Bedrock
- Tirez parti de pgvector et Amazon Aurora PostgreSQL pour le traitement du langage naturel, les chatbots et l'analyse des sentiments

AWS documentation de service

- Choix d'un service AWS de base de données
- Comment fonctionnent les bases de connaissances Amazon Bedrock
- Documentation de Neptune Analytics
- Présentation d'Amazon Web Services : bases de données
- Utilisation d'Aurora PostgreSQL comme base de connaissances pour Amazon Bedrock
- Utilisation d'Amazon Aurora PostgreSQL
- Amazon DocumentDB
- Amazon MemoryDB
- Amazon S3 Vectors

Autres AWS ressources

- Bases de connaissances Amazon Bedrock
- Bases de données vectorielles et intégrations
- Bases de données vectorielles pour les applications d'IA générative
- Que sont les intégrations dans le Machine Learning ?

Autres ressources

- À propos de PostgreSQL

- documentation pgvector
- Pinecone comme base de connaissances pour Amazon Bedrock
- Redis Enterprise Cloud sur AWS

Historique du document

Le tableau suivant décrit les modifications importantes apportées à ce guide, intitulé Choisir une base de données AWS vectorielle pour les cas d'utilisation de RAG. Pour être averti des mises à jour à venir, abonnez-vous à un fil RSS.

Modification	Description	Date
Ajouté Services AWS	Nous avons ajouté des informations sur l'utilisation d'Amazon DocumentDB, d'Amazon MemoryDB, d'Amazon S3 Vectors et d'Amazon Neptune Analytics.	30 mars 2026
Publication initiale	—	6 mars 2025

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.