

Flexibilité des données stratégiques

Les fournisseurs de bases de données traditionnelles et les fournisseurs de services cloud proposent des modèles compromis de la couche de données. Heureusement, Redis Enterprise offre une troisième voie : le meilleur des deux mondes.



Sommaire général

Le choix de la bonne base de données est une décision cruciale, dont les conséquences se font sentir dans toute l'organisation et de haut en bas de la chaîne technologique pendant des années, voire des décennies. Malheureusement, le choix d'une base de données peut s'avérer difficile et complexe, l'impact total n'apparaissant clairement que lorsque la base de données est utilisée à grande échelle ou au-delà du modèle de données initial.

Ironiquement, un ensemble limité d'options rend le choix de la base de données encore plus complexe. L'industrie des bases de données est essentiellement divisée en deux camps. Le premier défend un modèle de base de données relationnelle monolithique, tandis que le second pousse les clients à utiliser une variété de bases de données spécialisées pour chaque application. Imaginez le premier camp comme un chariot à hot-dogs : il ne propose qu'une seule chose, des hot-dogs. Le deuxième camp est comme une aire de restauration, vous choisissez parmi un éventail d'options qui peuvent ou non aller ensemble.

Bases de données relationnelles = hot dogs

Les vendeurs de hot dogs s'appuient sur un seul modèle de données (relationnel) et un seul type de logiciel (DBMS). Quel que soit le type réel de données stockées et recherchées, elles doivent être pliées et formées pour s'adapter au modèle relationnel. L'équipe de développement doit écrire cette couche supplémentaire de logiciel pour adapter une variété de problèmes de données en problèmes relationnels/tabulaires. D'un point de vue opérationnel, cela entraîne souvent des problèmes de performance ou d'échelle, car cela oblige la base de données à faire des choses pour lesquelles elle n'a pas été conçue. Peu importe la quantité de condiments qu'ils ajoutent, ces vendeurs ne proposent que des hot-dogs.

“ Ce stand vend des hot-dogs. Même si vous ajoutez du ketchup, de la moutarde ou même des cornichons, ça reste un hot-dog. Ne laissez pas quelqu'un vous dire le contraire. ”



“ Redis Enterprise propose une troisième voie qui offre le meilleur des deux mondes. ”



Bases de données cloud = aires de restauration

Selon le camp des aires de restauration, de nombreux modèles de données sont nécessaires, même pour une seule application, ce qui signifie que tout le monde doit utiliser plusieurs bases de données spécialisées. Ces bases de données sont reliées entre elles par un réseau, ce qui donne un ensemble fragile et lent de services interconnectés. Les développeurs doivent apprendre un éventail déconcertant de langages d'interrogation, de méthodes de connexion et de bibliothèques clientes. Et ils doivent se préoccuper des problèmes de synchronisation et des tolérances de défaillance lorsqu'ils gèrent des données sur plusieurs services. Les équipes d'exploitation doivent s'occuper de la configuration, de la mise à l'échelle, de la sécurisation et de la maintenance non pas d'une seule plate-forme de base de données, mais de plusieurs.

Les deux choix sont compromis. Ils intègrent les conflits et la fragilité ainsi que les problèmes de performance et de maintenabilité. Plus important encore, ils représentent un faux choix, un choix forcé. Il existe une autre voie, qui peut vous aider à éviter l'ensemble complexe et l'obligation de tout faire entrer dans un modèle de données unique.

Redis Enterprise = une troisième voie

Redis Enterprise présente une troisième voie : une interface opérationnelle unique au sein de laquelle vous pouvez déployer plusieurs bases de données. Les bases de données de Redis Enterprise offrent des modules qui étendent la fonctionnalité de Redis basée sur les clés à des modèles de données plus spécifiques tels que les graphes, les recherches, les documents et les séries chronologiques, ainsi que des capacités probabilistes, déclenchées par des événements et servant l'IA. Les développeurs interagissent avec la base de données par le biais de bibliothèques clientes Redis standard. Au lieu d'un stand à hot-dogs ou d'une aire de restauration, Redis Enterprise propose un menu complet d'options bien conçues et réfléchies provenant d'un seul et même restaurant.

Introduction

Les données sont au cœur de la plupart des logiciels modernes. La base de données est l'un des choix technologiques clés pour les responsables technologiques. Ce choix a été guidé par l'évolution du stockage lui-même. Le premier stockage de données était physique, littéralement des trous dans des cartes ou des bandes de papier. Au fil des ans, nous sommes passés aux supports magnétiques, puis aux dispositifs de stockage à semi-conducteurs. Aujourd'hui, la sophistication passe de la physique de la façon dont les bits sont stockés sur des supports physiques à la façon dont les données sont modélisées et transformées en informations utilisables.

Pourtant, la plupart des anciens logiciels de base de données ont été conçus à une époque où les niveaux d'abstraction disponibles aujourd'hui n'existaient tout simplement pas ; les logiciels ne pouvaient être livrés que sur des supports physiques.

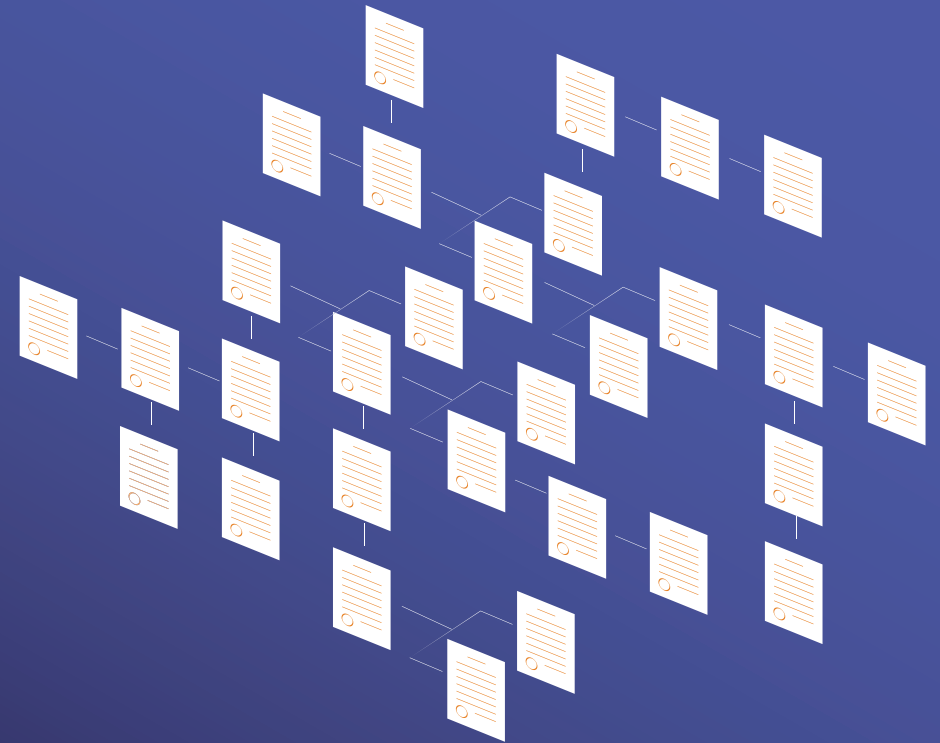
Il existe toujours une base installée massive de bases de données anciennes, même sur des systèmes non anciens. Ces bases de données reposent sur un modèle relationnel unique et monolithique. Bien que ce modèle reste utile pour modéliser certains problèmes de données, il représente un moteur de stockage (désormais) générique qui a trop souvent été utilisé et mal utilisé à la place de techniques de modélisation de données appropriées.

Les fournisseurs de services cloud, quant à eux, font totalement abstraction de la notion de machine, en fournissant des bases de données spécialement conçues à cet effet et disponibles uniquement sur leur service. Vous les connectez « simplement » à votre application. Avez-vous besoin d'une base de données avec un modèle différent ? « Il suffit » de créer une nouvelle base de données et de s'y connecter.

Il est essentiel que les bases de données dans le cloud et les bases de données traditionnelles fassent des compromis fondamentaux. Dans de nombreux cas, elles aboutissent à l'adoption cyclique de différentes approches technologiques dans des tentatives répétées de soulager la douleur d'un groupe de composants qui finissent par se faire au détriment d'autres composants.

Pour être clair : les compromis dans votre stratégie de base de données peuvent entraîner des problèmes internes (satisfaction et accomplissement du personnel, longévité du projet) et externes (performance et stabilité). Heureusement, la troisième option offre une voie médiane qui évite les pièges des bases de données patrimoniales et des fournisseurs de cloud. Une option stratégique suffisamment flexible pour s'adapter nativement aux nouveaux modèles de données et pour permettre une simplicité et une stabilité opérationnelles.

“ La plupart des anciens logiciels de base de données ont été conçus à une époque où les niveaux d'abstraction disponibles aujourd'hui n'existaient tout simplement pas. ”





Postes dans le secteur des bases de données

Compte tenu de ces deux approches concurrentes de la conception des bases de données, il n'est pas surprenant qu'une grande partie de l'industrie des bases de données se divise en deux camps : les fournisseurs de bases de données traditionnels et les fournisseurs de services cloud. Chaque camp voit le paysage de manière très différente et préconise des solutions très différentes.

Anciens fournisseurs de bases de données

Très tôt dans l'histoire de l'informatique, les gens ont commencé à réfléchir à la manière de stocker les données dans un format standardisé. Charles W. Bachman avait créé l'Integrated Data Store chez General Electric en 1963, ce qui représente la première chose que nous pourrions reconnaître aujourd'hui comme une base de données. Quelques années plus tard, en 1970, Edgar Codd a écrit « A Relational Model of Data for Large Shared Data Banks » (un modèle relationnel de données pour les grandes banques de données partagées), qui a servi de modèle pour le modèle de base de données relationnelle. Larry Ellison a repris les idées de ce document pour en faire le fondement de ce qui allait devenir la base de données Oracle, sortie en 1979.

Le modèle relationnel s'est avéré très flexible (pour l'époque) et est finalement devenu la norme de facto en matière de bases de données. Le modèle relationnel et la base de données sont devenus fonctionnellement équivalents dans la plupart des conversations. En effet, ce modèle puissant vous permet d'accomplir de nombreuses choses différentes avec une seule interface. Qu'il s'agisse d'un projet simple pour quelques utilisateurs ou d'un projet de données complexe à l'échelle de l'entreprise, le modèle relationnel était la réponse. Bien sûr, d'innombrables watts d'énergie ont été gaspillés lorsque des bases de données relationnelles massives ont été lancées et maintenues pour ne jamais effectuer d'opération plus complexe que la recherche d'éléments par une clé primaire dans un seul tableau. L'intention n'était pas de gaspiller, mais pendant des décennies, les bases de données relationnelles ont été la seule option viable, bien connue et bien comprise, si bien qu'elles ont été mises à contribution pour à peu près tout.

Des données modélisées de manière complètement différente ont été introduites dans la base de données relationnelle, créant ainsi des problèmes massifs de modélisation. Le mappage de données de graphes sur une base de données relationnelle, par exemple, crée des problèmes d'indexation et des problèmes massifs de performance. De même, les données de séries chronologiques constituent une charge de travail lourde en termes d'écriture et d'agrégation que de nombreuses bases de données relationnelles ne peuvent tout simplement pas gérer en temps réel.

“ D'innombrables watts d'énergie ont été gaspillés alors que d'énormes bases de données relationnelles étaient lancées et maintenues pour ne jamais effectuer d'opération plus complexe que la recherche d'éléments par une clé primaire dans un seul tableau. ”

Le stockage de documents dans une base de données relationnelle tente de faire passer ces types de données sans forme, sans schéma, dans un format tabulaire hautement typé et structuré. Tous ces modèles de données sont tout à fait légitimes, mais l'omniprésence des bases de données relationnelles a eu tendance à pousser la complexité et l'adaptation des modèles vers l'application. Souvent, le résultat était une base de données relationnelle contenant des données auxquelles seule une application spécifique pouvait accéder, et une base de données surchargée luttant pour suivre un modèle d'utilisation pour lequel elle n'a jamais été conçue.

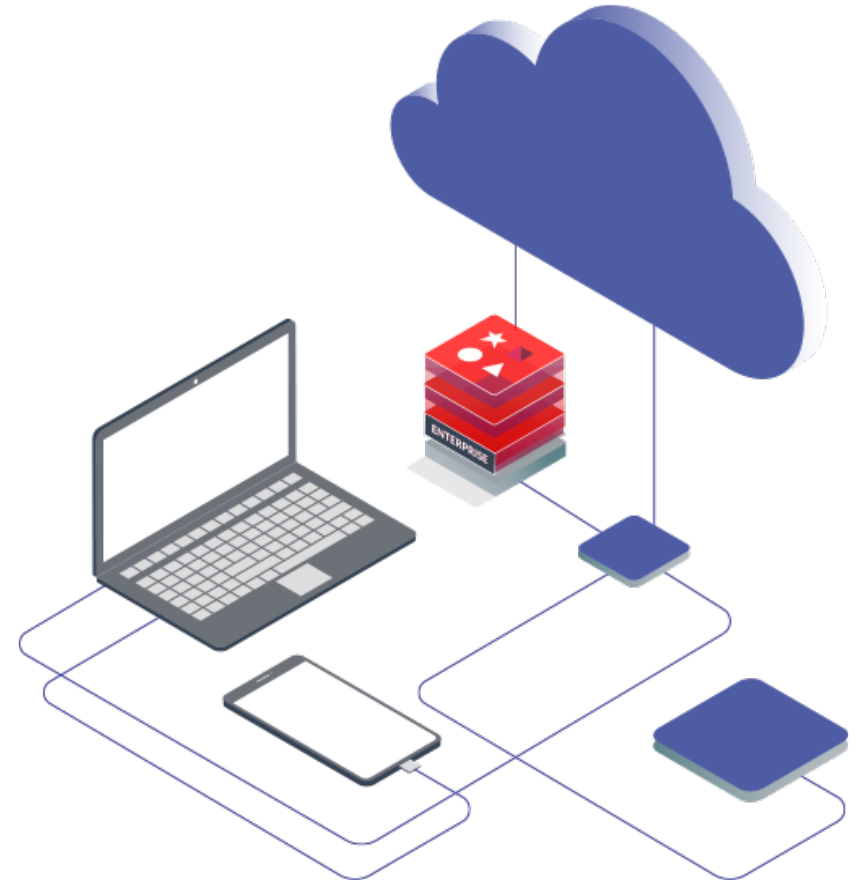
Encore une fois, cela ne veut pas dire que le modèle relationnel est totalement inapproprié. Comme tout modèle de données, il existe des problèmes pour lesquels les bases de données relationnelles constituent le choix optimal. Les applications qui comportent des données hautement relationnelles et/ou qui ne peuvent pas être dénormalisées de manière efficace conviennent toujours très bien à ce type de bases de données. Mais tout comme lorsque vous n'avez qu'un marteau, tout ressemble à un clou, si vous n'avez qu'une base de données relationnelle, tous les problèmes semblent relationnels.

Pour être honnête, le fait de se contenter d'une seule plate-forme de base de données présente certains avantages opérationnels. Les professionnels qui gèrent ces plates-formes ont rapidement et profondément appris à les maintenir stables. En tant que logiciels complexes et de grande taille, les systèmes de bases de données relationnelles peuvent être exploités au niveau organisationnel plutôt que d'être déployés spécifiquement pour un logiciel individuel, ce qui centralise la complexité et l'expertise. Bien que l'approche unique ait le potentiel de causer des problèmes de développement, elle a également créé un vecteur unique et connu à rendre opérationnel.

Sur le plan architectural, cependant, les racines de ces systèmes datent d'une époque antérieure à la naissance de nombreuses personnes qui les exploitent, avant l'internet moderne et a fortiori le cloud computing. Il est vrai qu'il y a peu, voire aucun, code source datant de la fin des années 1970 qui fonctionne encore dans cette catégorie de logiciels, mais les concepts sous-jacents sont nés à une époque fondamentalement différente.

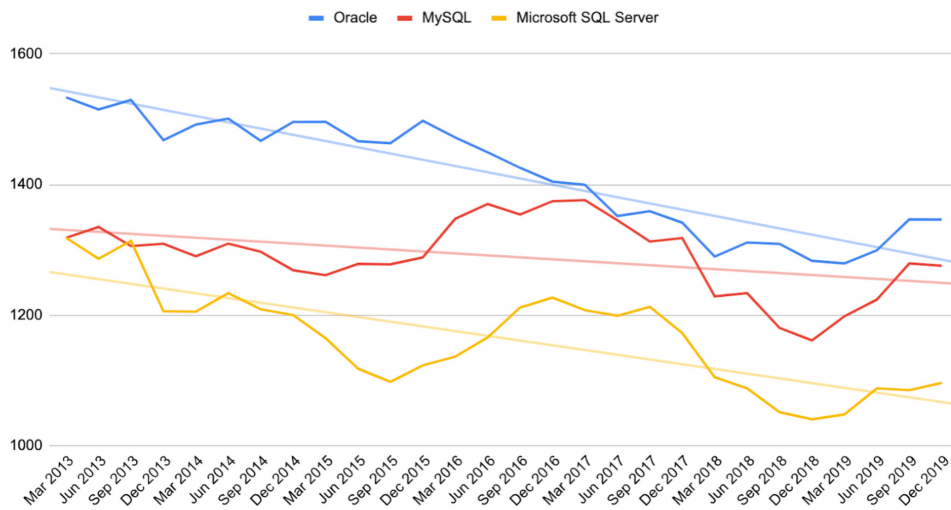
Entrez dans le cloud

Lorsque les entreprises ont commencé à transférer leurs logiciels vers le cloud, beaucoup ont essayé de le faire sans réarchitecturer leurs applications (ce que l'on appelle le « levage et déplacement »). Malheureusement, les hypothèses formulées au niveau de l'architecture pour de nombreux logiciels ne correspondaient pas à l'environnement radicalement différent du cloud, et les SGBDR ne faisaient pas exception. Des décennies d'optimisations de bas niveau et de micro-optimisations avaient produit des logiciels qui fonctionnaient avec des performances acceptables dans les environnements sur site. Mais lorsqu'ils étaient placés dans un environnement virtualisé, de nouveaux défis apparaissaient : les performances inférieures du stockage en réseau hors serveur, les ressources de calcul éphémères et le manque d'automatisation créaient un chemin rocaillieux vers le cloud. En bref, les premiers exercices de levage et de déplacement ont prouvé que les professionnels de l'exploitation doivent être des experts non seulement des logiciels de bases de données massives, mais aussi du cloud qu'ils utilisent. Le monde avait changé.



“ Les anciens fournisseurs de bases de données cherchaient à fournir une base de données à une organisation, tandis que les fournisseurs de services cloud cherchent à fournir de nombreuses bases de données à une seule application. ”

Classement des 3 meilleures bases de données relationnelles DB-engines, Trimestriel 2013-2019



L'ère moderne du cloud computing a affecté la popularité des bases de données anciennes. Si les bases de données relationnelles occupent les trois premières places du classement de DB-Engines.com, leur tendance dans le temps est clairement à la baisse. Depuis que DB-Engines a commencé à suivre les scores en 2013, il y a eu un déclin général de la popularité des principaux modèles relationnels, selon la [méthodologie de notation de DB-Engines](#).

Les ombres du cloud : les bases de données relationnelles restent populaires... pour l'instant

Fournisseurs du cloud

À l'autre bout du spectre se trouvent les fournisseurs de services cloud. Sur la base de leurs gammes de produits, les fournisseurs de services cloud semblent rejeter l'idée que le modèle relationnel est le moyen standard de stocker et d'extraire des données. Les différents modèles de données représentés dans les produits de base de données des fournisseurs de cloud suggèrent que les données de votre application ne doivent pas se plier à la base de données disponible, mais que vous devez plutôt sélectionner la ou les bases de données adaptées à votre application.

On peut considérer que les fournisseurs de cloud soutiennent NoSQL, un terme inventé dès 1998, mais dont la signification contemporaine date de 2009. À l'origine, les bases de données NoSQL s'attachaient à fournir un modèle de données unique pour prendre en charge une liste plus courte de cas d'utilisation. Elles sont conçues pour un seul modèle de données. Cependant, toute application suffisamment complexe comporte probablement un large éventail de données qui doivent être modélisées de différentes manières, de sorte qu'une seule application peut nécessiter plusieurs bases de données différentes. En effet, alors que les anciens fournisseurs de bases de données cherchaient à fournir une base de données à une organisation, les fournisseurs de cloud peuvent chercher à fournir de nombreuses bases de données à une seule application.

L'utilisation de plusieurs bases de données dans un seul projet peut aider à résoudre les problèmes des développeurs. Les bases de données relationnelles créent des résultats qui obligent les développeurs à s'appuyer sur des bibliothèques ou sur leur propre code pour rassembler et désassembler les données afin de les adapter au modèle relationnel (le **problème de l'inadéquation de l'impédance objet-relationnel**). Les entrées et les sorties des bases de données NoSQL ont tendance à ressembler à la façon dont le code de l'application modélise les données en interne. Ainsi, en tant que développeur travaillant avec des bases de données spécialisées, la proportion de temps allouée à l'écriture d'un code pertinent, permettant de résoudre les problèmes, est plus importante que dans le cas des bases de données relationnelles, où il est nécessaire de transformer les données en une forme utilisable.

Mais l'utilisation de plusieurs bases de données spécialisées dans une seule application présente ses propres inconvénients. Comme tout outil, il faut des compétences et des techniques d'utilisation. Chaque base de données exige des développeurs qu'ils apprennent non seulement la base conceptuelle de la base et du modèle, mais aussi les bibliothèques qui l'accompagnent. Par exemple, la compréhension des paramètres de connexion des bases de données est souvent une tâche non triviale qui, si elle n'est pas gérée correctement, peut entraîner des problèmes de performance ou de stabilité.

Bien qu'elles soient basées sur le cloud, les bases de données multiples présentent également des défis opérationnels. L'un de ces défis est lié à la gestion d'un grand nombre de services cloud dans votre organisation. Chaque application étant desservie par de nombreuses bases de données, vos équipes d'exploitation doivent s'occuper d'un plus grand nombre de logiciels différents. Les services cloud sont plus faciles à maintenir que les services sur site, mais la maintenance opérationnelle reste variable et non nulle. Par exemple, certaines bases de données effectuent des mises à niveau continues tandis que d'autres nécessitent une intervention. Certaines bases de données nécessitent une configuration spécifique pour votre contexte, tandis que d'autres peuvent fonctionner parfaitement par défaut.

Latence et mise à l'échelle

Outre l'impact sur la complexité globale du fait d'avoir plusieurs bases de données par application, la latence est également affectée. En outre, il peut être difficile de maîtriser la latence globale dans un système complexe. Si une seule opération d'une application doit toucher plusieurs bases de données, la latence cumulée de toutes les bases de données et de l'application affectera les performances globales de l'application. Alors qu'une plate-forme de base de données unique peut effectuer de nombreuses opérations en une seule demande, dans le cloud, chaque opération peut être effectuée sur une base de données distincte qui ajoute sa propre latence aller-retour à l'application et au retour. En effet, les opérations synchrones peuvent nécessiter de nombreux sauts de réseau aller-retour (ce qui augmente la latence), tandis que d'autres peuvent être parallélisées, de sorte que tous les sauts de réseau se produisent en même temps (dans ce cas, la latence globale serait la latence la plus élevée du groupe).

La mise à l'échelle est une autre considération. Chaque base de données supportant l'application peut évoluer différemment, mais les performances de l'application dépendront de son lien le plus faible. Même avec une infrastructure optimisée pour chaque base de données, la stratégie opérationnelle est complexe, car les performances atteignent des plateaux sur l'une ou l'autre base de données au fur et à mesure que vous évoluez.

“ Chaque base de données exige des développeurs qu'ils apprennent non seulement la base conceptuelle de la base de données et du modèle, mais aussi les bibliothèques qui l'accompagnent. ”



Attention à la couverture trop courte

Imaginez que vous essayez de faire un lit avec une couverture qui n'est pas assez longue pour couvrir le matelas. Rentrer la couverture au pied du lit laisse la tête exposée. L'aligner au niveau de la tête laisse le pied exposé. En posant la couverture au milieu, on expose à la fois la tête et le pied. Toutes les options sont inadéquates.

De même, les fournisseurs de bases de données de l'ancienne génération offrent un déploiement organisationnel et une mise à l'échelle unifiés, mais leurs modèles mal adaptés doivent être corrigés par une complexité de développement supplémentaire (voir en bas à gauche). Les fournisseurs de services cloud offrent un large éventail de types de bases de données spécialement conçues, ce qui entraîne malheureusement une complexité opérationnelle liée à la nécessité de connecter, de faire évoluer et de faire fonctionner plusieurs bases de données par application (voir en bas à droite).

En définitive, les deux approches sont inadéquates. Déplacer la complexité vers les équipes de développement ou d'exploitation ne réduit pas la complexité globale. L'effort massif d'ingénierie nécessaire pour faire passer une application d'un modèle de données unique dans une base de données ancienne à de nombreuses bases de données spécialisées dans le cloud pourrait en fait être une perte de temps s'il ne réduit pas réellement la complexité mais l'affecte simplement à une autre équipe.

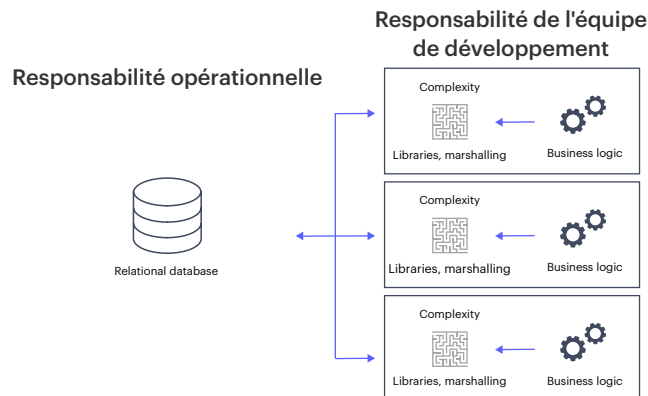
Pire encore, la migration d'une application vers le cloud peut entraîner un cycle sans fin de réusinage. Prenons l'exemple d'une application qui fonctionne bien et est stable sur une variété de modèles de données, mais qui est encastrée dans une base de données relationnelle. Même si l'application fonctionne bien, son code sous-jacent est long et



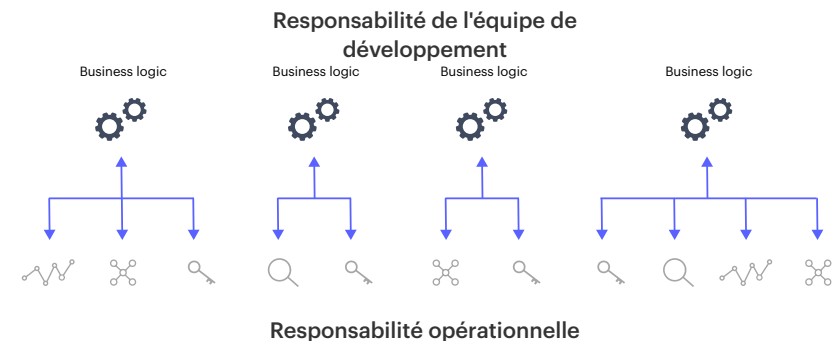
difficile à maintenir. Une grande partie de ce code ne concerne pas la logique métier, mais plutôt les problèmes d'impédance objet-relationnel. Le poids du code rend difficile l'ajout de nouvelles fonctionnalités et de corrections.

Pour résoudre ce problème, l'équipe de développement envisage, par exemple, une base de données de graphes, une base de données de séries chronologiques et une base de données de recherche qui élimineraient le besoin de grandes parties de la base de code existante. Le passage à ces bases de données spécialisées semble raisonnable. Une réingénierie a lieu et la prochaine version de l'application est écrite pour communiquer avec ces trois nouvelles bases de données.

L'équipe de développement itère rapidement sur les nouvelles fonctionnalités... mais l'environnement de production est désormais lent et instable. L'équipe d'exploitation constate des problèmes de délai de fonctionnement et a du mal à gérer le dépannage de plusieurs nouvelles bases de données. Il devient difficile de diagnostiquer le problème exact, car les solutions fournies par le cloud sont des boîtes noires qui offrent une visibilité limitée sur les bases de données problématiques. L'équipe d'exploitation commence à préconiser le passage à une autre stratégie de base de données (très probablement le retour à une base de données relationnelle) qui est plus stable et plus performante. Le cycle de la douleur et des dépenses recommence lorsque vous déplacez la couverture sur un lit qui ne convient tout simplement pas.



La complexité reste du côté de l'équipe de développement, car elle doit mapper les données relationnelles pour refléter le véritable modèle de l'application (logique métier).



La complexité de l'application réside dans la connexion et la maintenance de plusieurs bases de données spécialisées, comme le préconisent les fournisseurs de services cloud.

“ Une option stratégique consiste à trouver une voie qui incorpore le meilleur des deux mondes. Une voie qui permet à de multiples modèles de données natifs de coexister tout en évoluant de manière prévisible. ”



Flexibilité stratégique avec Redis Enterprise

Jusqu'à présent, ce document n'a abordé que deux options, aussi inadéquates soient-elles. Les approches SGBDR pour tout et bases de données multiples dans le cloud ne fonctionnent qu'à un niveau tactique, en résolvant un problème de manière régimentée, en déplaçant les difficultés au sein de l'organisation et, en fin de compte, en préparant l'organisation à des projets de réingénierie cycliques et coûteux.

Une option plus stratégique consiste à trouver une voie qui intègre le meilleur des deux mondes. Une voie qui permet à de multiples modèles de données natifs de coexister tout en évoluant de manière prévisible. Une voie qui permet aux développeurs d'utiliser une seule bibliothèque et de se connecter à un seul point de terminaison, tout en ayant accès à plusieurs modèles. Cette voie stratégique et flexible existe et elle mène à Redis Enterprise.

Redis Enterprise est une plate-forme de base de données construite autour d'un noyau de Redis. Redis, sans aucune extension, fournit un modèle d'accès aux données de type clé-valeur avec l'ajout de structures de données comme les cartes de hachage, les listes, les ensembles triés, les flux, et plus encore. Redis peut être étendu avec des modules qui introduisent des fonctionnalités entièrement nouvelles dans la base de données. Ces modules mettent en œuvre des modèles de données natifs tels que les graphes, la recherche plein texte, le stockage de documents et les séries chronologiques, ainsi que des fonctionnalités supplémentaires autour de l'intelligence artificielle et des scripts orientés sur les événements.


Les avantages de l'approche Redis Enterprise se résument à trois facteurs clés :

1. Une interface opérationnelle unique, avec plusieurs modèles de données
2. Une interface simple et unifiée
3. Née et élevée dans le cloud

Une interface opérationnelle, avec plusieurs modèles de données


Redis Enterprise offre le meilleur des deux mondes : elle propose une interface opérationnelle unique où de nombreux modèles de données peuvent coexister sur un même plan de données. Redis Enterprise a été développé à partir du besoin des utilisateurs à l'échelle de l'entreprise de rendre opérationnelles leurs instances Redis. L'écosystème de Redis a développé un système de modules robustes et enchâssables qui s'accroche profondément au cœur de Redis, permettant à Redis d'aller au-delà du modèle clé/valeur intégré et d'accueillir d'autres modèles de données. Cette approche multi-modèle lui permet d'hériter des caractéristiques opérationnelles de Redis tout en ajoutant de nouvelles capacités puissantes. À mesure que vous évoluez, il vous suffit d'ajouter des infrastructures à Redis Enterprise.

Activer un module dans Redis Enterprise est aussi simple que de cliquer sur un bouton dans une interface utilisateur, mais cela change radicalement la façon dont les données sont stockées et interrogées. Redis Labs a produit un certain nombre de modules qui permettent un certain nombre de modèles de données différents, y compris :

 **RedisSearch**
Recherche en texte plein et indexation secondaire

 **RedisGraph**
Moteur de graphes compatible avec Cypher

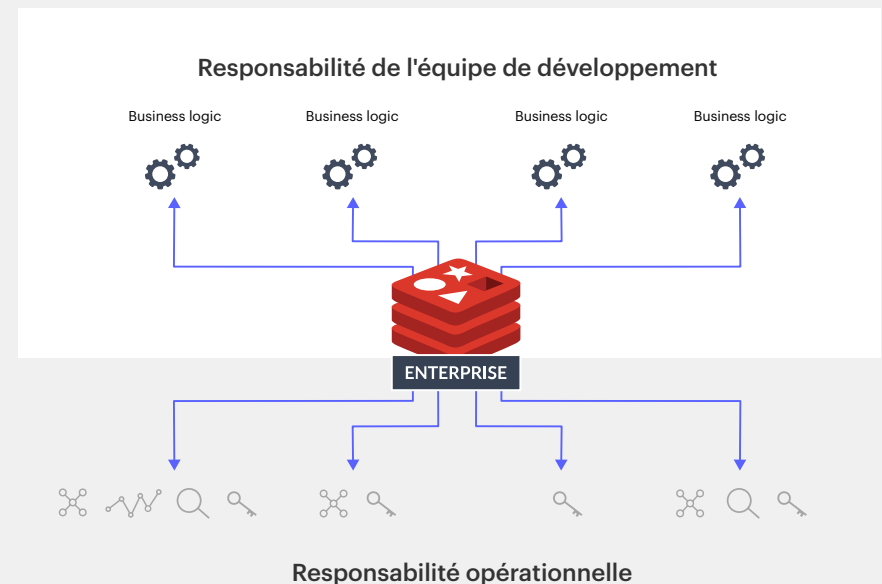
 **RedisJSON**
Stockage de documents JSON

 **RedisTimeSeries**
Enregistrez des échantillons de données dans des volumes élevés, puis procédez à l'agrégation et au déséchantillonnage à la volée

Les modules sont facultatifs, vous pouvez activer uniquement ceux dont vous avez besoin pour un projet spécifique. Redis Enterprise est multi-tenant, vous pouvez donc créer des bases de données pour des besoins individuels avec des modules spécifiques ou vous pouvez créer des bases de données avec de nombreux modules travaillant de concert, ce qui vous permet de zoomer sur les capacités spécifiques dont vous avez besoin pour votre application.

Ces modules, ainsi que la structure de données clé-valeur intégrée de Redis, permettent un large éventail d'utilisations. Il est important de comprendre que ces modules ne sont pas des capacités construites au-dessus du modèle clé-valeur, mais plutôt des modèles qui accèdent aux composants internes de Redis pour stocker des données en mémoire et les conserver sur le disque. Plusieurs de ces modèles peuvent être activés sur la même base de données, de sorte qu'il est possible, par exemple, d'utiliser la recherche en texte plein sur des nœuds de graphe ou de stocker des

“ Redis Enterprise offre le meilleur des deux mondes : elle propose une interface opérationnelle unique où de nombreux modèles de données peuvent coexister sur un même plan de données. ”



Redis Enterprise permet une séparation aisée des préoccupations de développement et d'exploitation sans compromettre la flexibilité du modèle de données.

“ Redis Enterprise est agnostique au cloud. Redis Labs n'est pas un fournisseur de cloud. Vous pouvez donc choisir les options de cloud et de déploiement, dans le cloud ou sur site, qui conviennent le mieux à votre organisation et à vos applications. ”



documents JSON à côté de données de séries chronologiques. Comme les différents modèles de données existent sur le même plan de données, même les transactions sont possibles entre les modèles.

Une interface simplifiée et unifiée

Comme une base de données Redis peut servir plusieurs modèles à partir du même plan de données, elle devient un point de connexion unique. Toutes les propriétés de connexion sont réduites en une seule et il n'y a pas de pénalités de performance ou de latence supplémentaire pour l'utilisation de plusieurs modèles de données natifs. En outre, les développeurs ne doivent comprendre qu'une seule interface programmatique pour tous les différents modèles de données - tout se résume au protocole Redis. En effet, vous bénéficiez des avantages de l'utilisation de bases de données distinctes dans un seul pack opérationnel.

Née et élevée dans le cloud

Redis a toujours été une solution pour le cloud. Redis Enterprise étend cela avec un pack global qui est conscient de l'environnement cloud avec des fonctionnalités telles que la conscience de la zone et du rack, la surveillance et les plans de données et d'administration bifurqués.

Étant donné que Redis utilise la mémoire système pour le stockage primaire des données, il crée une dynamique opérationnelle bien adaptée au fonctionnement sur n'importe quel cloud ou sur plusieurs clouds. Redis Enterprise prend également en charge Redis on Flash, qui permet au stockage basé sur SSD sur les instances cloud d'agir comme une extension de DRAM, ce qui permet une haute performance de très grands ensembles de données sur les mêmes infrastructures à un coût abordable. En tant que base de données en mémoire, Redis est prêt pour les prochaines générations de stockage de données, à mesure que l'industrie passe des disques rotatifs et du stockage Flash par blocs à un stockage durable adressable par octet.

De manière significative, Redis Enterprise est agnostique au cloud. Redis Labs n'est pas un fournisseur de cloud. Vous pouvez donc choisir le cloud qui convient le mieux à votre organisation et à vos applications. Cela permet d'éviter le verrouillage du cloud et de réaliser des déploiements de cloud hybride et multi-cloud. En outre, Redis Labs fournit une variété de méthodes de déploiement : les choix vont d'un environnement cloud entièrement géré à des déploiements sur site sous licence.

Redis Enterprise : une solution stratégique

La résolution d'un seul problème est souvent relativement peu compliquée. Faire des choix qui permettent de résoudre plusieurs problèmes est généralement plus difficile. Si l'on ajoute à cela la nécessité de résoudre de multiples problèmes actuels tout en s'attaquant à de multiples problèmes pour l'avenir, les choses peuvent devenir extrêmement complexes. C'est le cœur de la résolution stratégique des problèmes. Les choix de bases de données sont parfois faits à la légère, d'autres fois ils sont prédéterminés avant même le début d'un projet. Dans tous les cas, il ne s'agit pas d'un choix stratégique.

Redis Enterprise offre une nouvelle voie qui permet de résoudre stratégiquement les problèmes de développement et d'exploitation, non seulement aujourd'hui, mais aussi dans un avenir prévisible. Elle soulage à la fois la douleur des modèles de données en cours de développement et les performances et la stabilité des opérations, contribuant ainsi à garantir une décision qui résistera au temps.



La liberté d'apporter votre propre modèle de données

Les applications construites avec Redis Enterprise fonctionnent mieux car elles stockent les données de la manière dont votre application les utilise. Redis Enterprise prend en charge une variété de modèles de données intégrés et riches qui peuvent être utilisés en fonction de la tâche à accomplir. En tant qu'implémentations complètes, Redis Enterprise n'essaie pas de pousser les données dans des métaphores de données inappropriées, ce qui vous aide à réduire le code et la complexité au niveau de l'application. En réduisant le code et la complexité de vos applications, les développeurs sont plus productifs et se concentrent sur les problèmes opérationnels, au lieu d'essayer de faire entrer les données dans un modèle de données inapproprié.



Flexibilité opérationnelle intégrée

Comme Redis Enterprise permet à de nombreux modèles de données de coexister au sein d'un même cluster, voire d'un même espace de données, il élimine les connexions entre différentes bases de données indépendantes. **La consolidation de plusieurs modèles sous une seule connexion réduit la fragilité et l'instabilité engendrées par les connexions multiples.** Dans Redis Enterprise, vous pouvez même exécuter des transactions entre les modèles de données, ce qui n'est pas possible avec une collection de bases de données indépendantes.

“ Redis Enterprise offre une nouvelle voie qui permet de résoudre stratégiquement les problèmes de développement et d'exploitation, non seulement aujourd'hui, mais aussi dans un avenir prévisible. ”



“ Le coût total de possession d'une base de données doit prendre en compte non seulement le prix par unité de temps, mais aussi l'impact organisationnel. Votre base de données doit être suffisamment flexible pour prendre en charge de manière native plus d'un seul modèle de données, ce qui permet d'optimiser et de pérenniser la couche de données. ”



Pas de verrouillage du cloud

Comme elle ne provient pas d'un fournisseur de cloud, **Redis Enterprise n'est pas liée à un cloud spécifique : vous pouvez transférer vos données vers un tout nouveau cloud en cas de besoin, ou même les déployer dans une configuration hybride ou multi-cloud.** Vous avez un contrôle total sur l'infrastructure exécutant Redis Enterprise, vous pouvez facilement ajouter plus d'infrastructures pour soutenir un cluster. À l'intérieur du cluster, vous pouvez créer des bases de données isolées ou combinées qui ont des modèles de données spécialement conçus.



Flexibilité des infrastructures

Le coût total de possession d'une base de données doit prendre en compte non seulement le prix par unité de temps, mais aussi l'impact organisationnel. La solution que vous choisissez doit être suffisamment flexible pour prendre en charge de manière native plus d'un seul modèle de données, en optimisant et en protégeant la couche de données contre les changements imprévus. **Redis Enterprise simplifie les défis de la couche de données en donnant aux développeurs et aux architectes la liberté et la flexibilité de créer des applications qui offrent des performances sans sacrifier l'efficacité opérationnelle.** Cet équilibre sert à la fois les intérêts de ceux qui écrivent les logiciels et de ceux qui sont chargés de les faire fonctionner, tout en minimisant les charges excessives pour une partie de l'organisation.

Les fournisseurs de bases de données traditionnels et les fournisseurs de services cloud possèdent tous deux des tribunes d'où ils clament haut et fort qu'ils détiennent la véritable voie et que l'autre a complètement tort. Les deux parties ont des critiques raisonnables à formuler à l'encontre des faiblesses de la modélisation des données et de la complexité opérationnelle. Heureusement, le choix n'est pas binaire : vous avez des options en dehors de cette fausse dichotomie. Vous pouvez en savoir plus sur les avantages de Redis Enterprise sur redislabs.com/modules.



À propos de Redis Labs

Les entreprises modernes dépendent de la puissance des données en temps réel. Avec Redis Labs, les organisations offrent des expériences instantanées de manière hautement fiable et évolutive.

Redis Labs est le berceau de Redis, la base de données en mémoire la plus populaire au monde et le fournisseur commercial de Redis Enterprise. Redis Labs offre des performances supérieures, une fiabilité incomparable et une flexibilité inégalée pour la personnalisation, l'apprentissage automatique, l'IdO, la recherche, le commerce électronique, les réseaux sociaux et des solutions de mesure dans le monde entier.

Redis Labs, régulièrement classé comme un leader dans les principaux rapports d'analystes sur NoSQL, les bases de données en mémoire, les bases de données opérationnelles et la base de données en tant que service (DBaaS), est approuvé par plus de 7 400 entreprises clientes, dont cinq sociétés du Fortune 10, trois des quatre émetteurs de cartes de crédit, trois des cinq premières sociétés de communication, trois des cinq premières sociétés de soins de santé, six des huit premières sociétés technologiques et quatre des sept premiers détaillants.

Redis Enterprise, disponible en tant que service dans les clouds publics et privés, en tant que logiciel téléchargeable, dans des conteneurs et pour les déploiements hybrides cloud/sur site, fournit les cas d'utilisation populaires de Redis tels que les transactions à grande vitesse, la gestion des tâches et des files d'attente, les magasins de sessions utilisateur, l'acquisition de données en temps réel, les notifications, la mise en cache de contenu et les données de séries chronologiques.

Suivez-nous :



redislabs.com