

QUELLE BASE DE DONNÉES POUR QUEL USAGE ?

Forces et faiblesses des bases relationnelles et modèles NoSQL



Ce tableau comparatif confronte les principales caractéristiques des bases SQL et NoSQL, en établissant la distinction entre les 4 grandes catégories de bases NoSQL (NoSQL clé/valeur, NoSQL orienté document, NoSQL colonne et NoSQL graphe).

Pour chaque type de SGBD, vous trouverez une synthèse comparative détaillant les choix effectués et mettant l'accent sur des cas d'usage.

ÉTUDE COMPARATIVE

INTRODUCTION :

QUEL SYSTÈME DE GESTION DE BASE DE DONNÉES PRIVILÉGIÉ EN FONCTION DE VOTRE CONTEXTE MÉTIER ? NOSQL OU SGBD DE TYPE RELATIONNEL ?

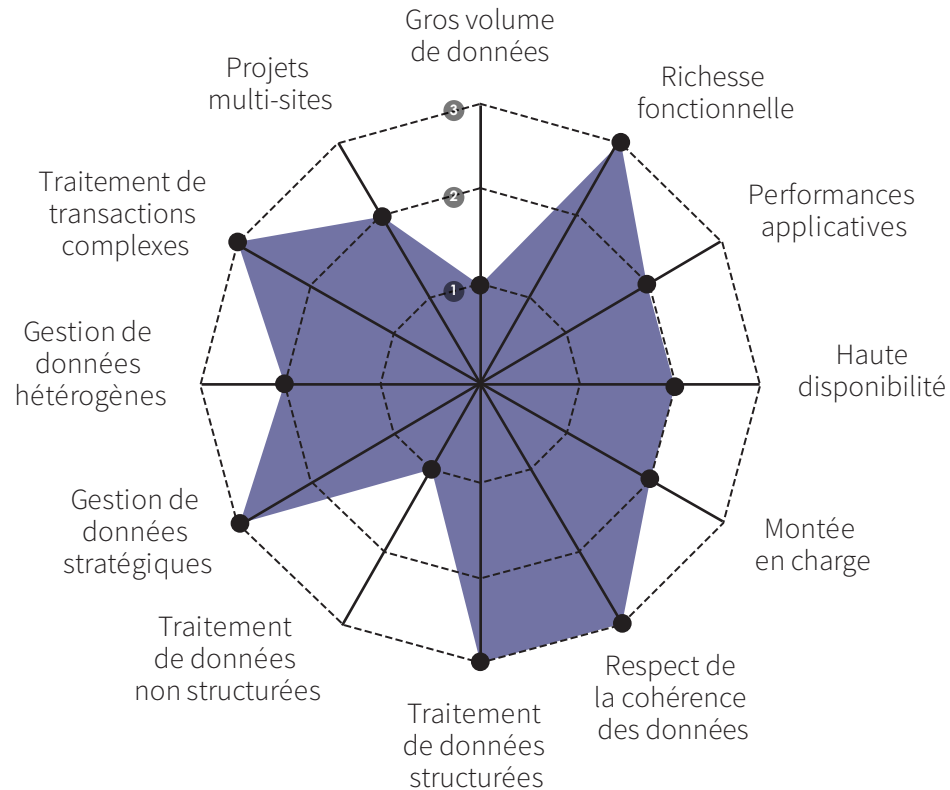
Alors que les bases de données relationnelles (Oracle DB, SQL Server...) sont relativement généralistes et correspondent à des cas d'utilisation variés, les technologies NoSQL sont au contraire très spécifiques et permettent de personnaliser la solution de gestion des données pour chaque cas d'utilisation.

Quels sont les avantages et les inconvénients de chaque SGBD en fonction des caractéristiques de votre projet (gros volumes de données à traiter, traitement de transactions complexes, besoin de monter en charge...) et du type d'application concerné (web, GED, e-commerce...) ? Obtenez une vision synthétique des différents critères comparatifs à prendre en compte pour évaluer les différentes solutions de SGBD et orienter vos choix.



LES BASES DE DONNÉES RELATIONNELLES

CARACTERISTIQUES DU PROJET



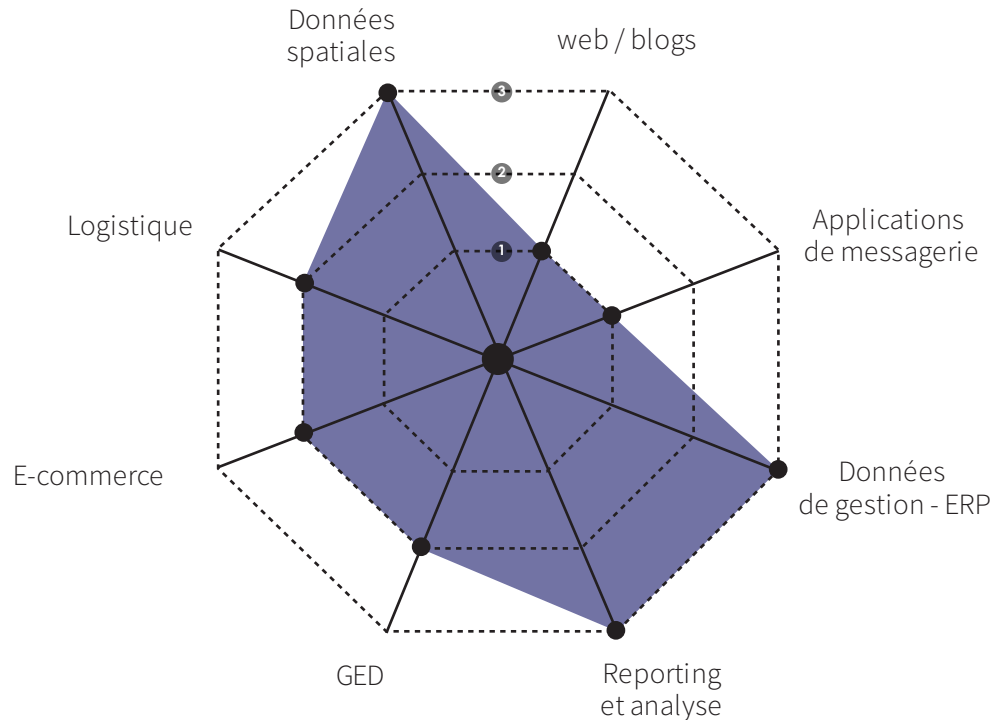
Conçus pour répondre à une multiplicité d'usages, les SGBDR (systèmes de bases de données relationnelles) sont adaptés aux projets qui nécessitent un bon équilibre entre la complexité des données à traiter et le volume. La plupart de ces bases utilisent SQL comme langage standard d'accès aux données. Contrairement aux bases NoSQL, leur organisation en table réduit leur usage au traitement des données structurées.

Les bases SQL sont donc peu adaptées aux médias sociaux, aux flux vidéo et audio, à l'internet des objets et de façon générale au traitement des données fortement hétérogènes. On peut prendre l'exemple d'une entreprise qui souhaite stocker des données météorologiques, par nature très hétérogènes (car les sources, mesures, schémas et unités utilisés seront très différents). Dans un tel contexte métier, le schéma statique fortement structuré de la base relationnelle ne conviendra pas. On préférera le modèle plus souple et évolutif offert par les bases NoSQL.



LES BASES DE DONNÉES RELATIONNELLES

TYPE D'APPLICATION



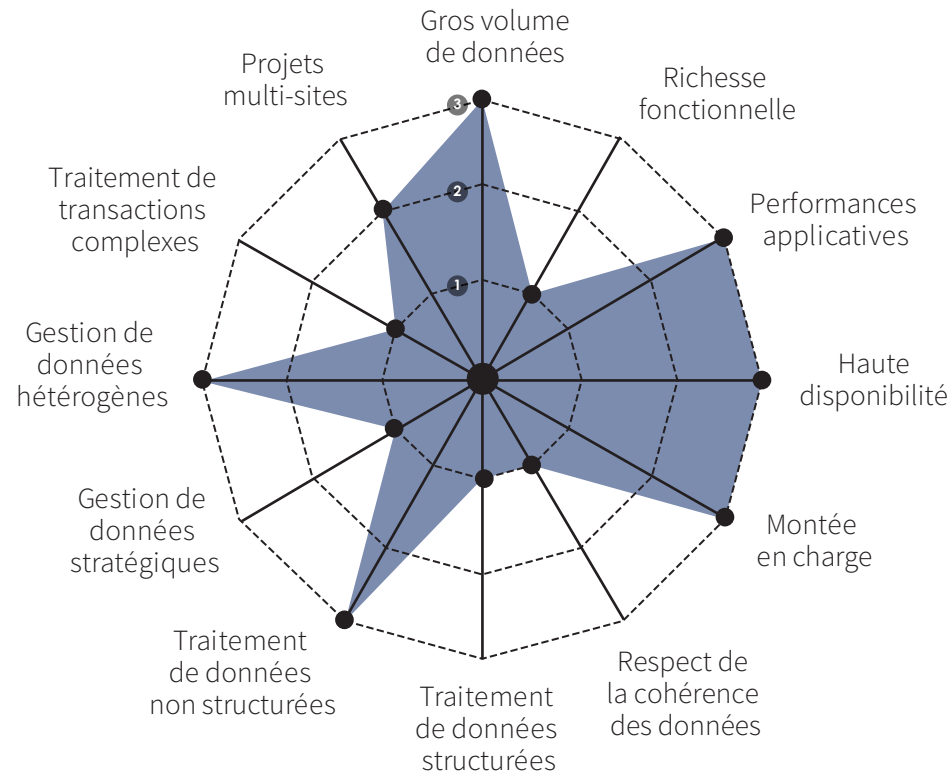
En revanche, la richesse fonctionnelle des bases relationnelles et la robustesse de leur implémentation ACID (atomicité, cohérence, isolation, durabilité) les rendent particulièrement adaptées à la gestion des données stratégiques (transaction financières, applications gérant des données financières, etc...).

Au final, les systèmes relationnels privilégient la cohérence et l'intégrité des données plutôt que leur haute disponibilité. Ce modèle convient bien aux données de gestion qui sont souvent très structurées, avec des besoins de manipulation à partir de plusieurs briques logicielles clientes et des besoins de recherches multicritères ainsi que des manipulations complexes. Les volumes de données étant généralement assez modestes dans les applications de gestion, il est rarement nécessaire de distribuer le traitement, ce qui confirme l'intérêt de recourir aux BDD relationnelles dans ce type de projets.



LES BASES DE DONNÉES NoSQL ORIENTÉES CLÉ/VALEUR

CARACTERISTIQUES DU PROJET



Les bases de données NoSQL orientées clé/valeur constituent la forme la plus simple de base NoSQL. Elles sont souvent utilisées pour des projets applicatifs qui nécessitent des performances élevées mais qui reposent sur des jeux de données relativement simples.

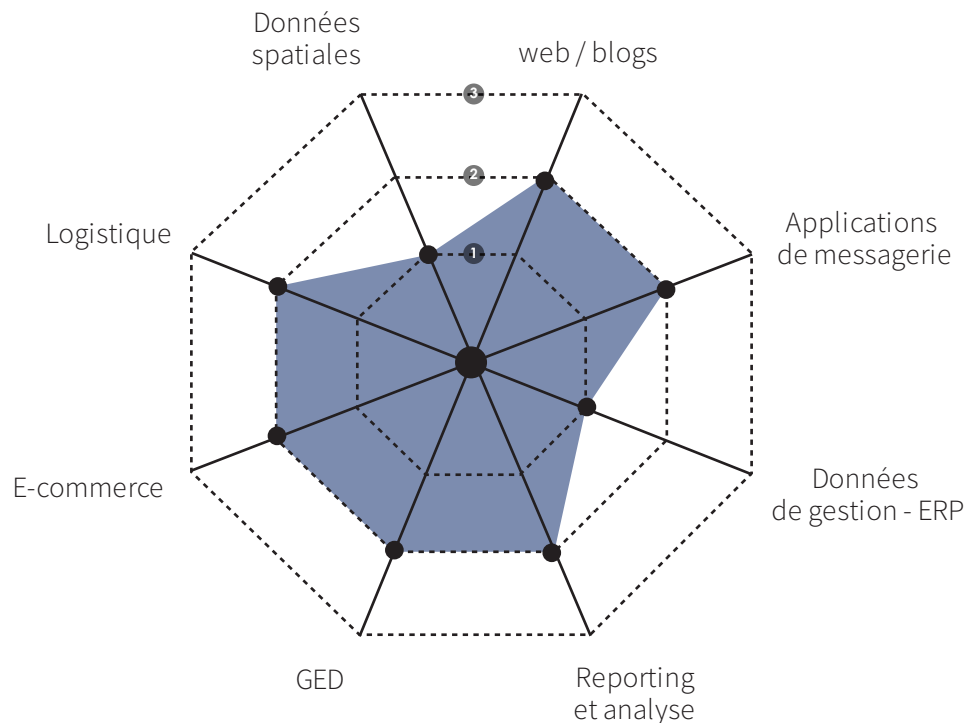
La richesse fonctionnelle des bases NoSQL orientées clé/valeur est bien inférieure à celle offerte par les bases relationnelles. A titre d'exemple, une base comme Redis permettra de gérer des transactions, enchaîner des commandes (pipeline) ou gérer des listes, mais en aucun cas d'effectuer des recherches full text ou du géospatial. La représentation des données ne facilitera donc pas le travail du gestionnaire de la base.

Par contre, une base de données NoSQL clé/valeur conviendra parfaitement si vous avez besoin d'effectuer des recherches très rapides. On peut prendre l'exemple d'une application mobile de commande sur smartphone.



LES BASES DE DONNÉES NoSQL ORIENTÉES CLÉ/VALEUR

TYPE D'APPLICATION



Pour ce type d'usage, le temps de réponse sera primordial. Les bases NoSQL clé/valeur conviennent bien à cette configuration car elles peuvent être facilement dimensionnées pour répondre aux besoins en termes de traitement et de trafic. De même, elles seront efficaces dans le cadre de la gestion de profils utilisateur ou produit.

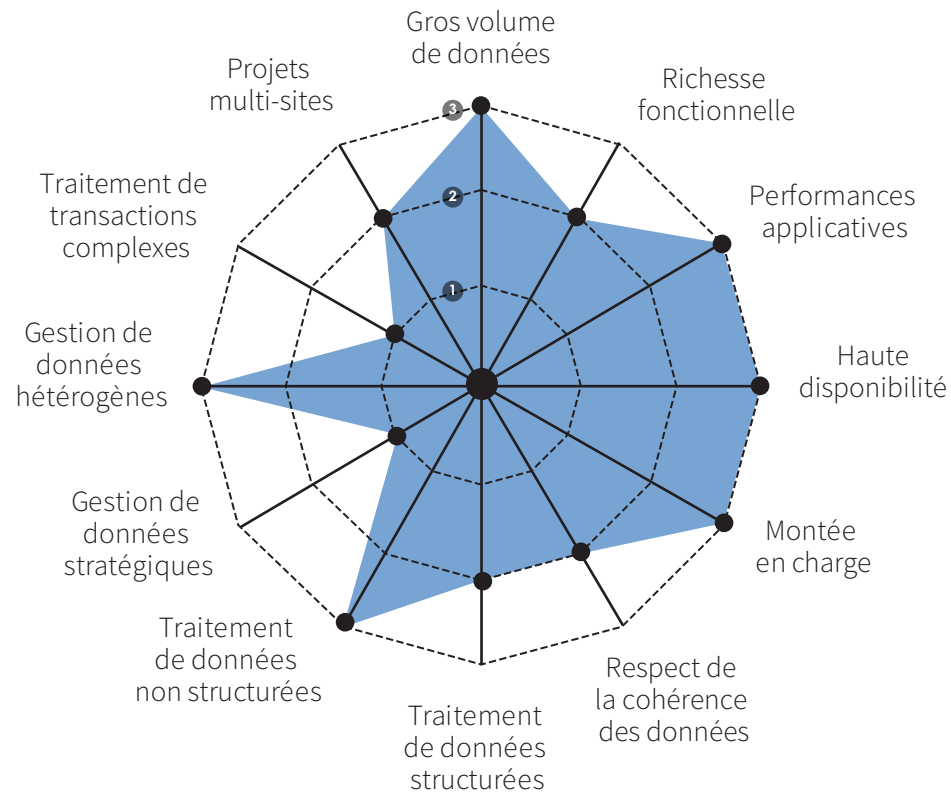
Au final, ce type de base est un choix intéressant dans l'hypothèse où :

- les données sont codées de multiples façons sans schéma rigoureux,
- vous avez besoin de performances
- vous n'avez pas à réaliser de recherches poussées



LES BASES DE DONNÉES NoSQL ORIENTÉES DOCUMENT

CARACTERISTIQUES DU PROJET



Contrairement aux bases NoSQL de données de type clé/valeur, les bases orientées document imposent une structure aux données. Au lieu de conserver les données non structurées dans ce qui constitue la « valeur », elles les décrivent dans une véritable structure de documents (très souvent au format JSON, ou XML). Ainsi, les bases NoSQL orientées document peuvent être utilisées pour des systèmes de gestion de contenus. Par contre, elles seront peu indiquées pour le traitement de transactions très complexes.

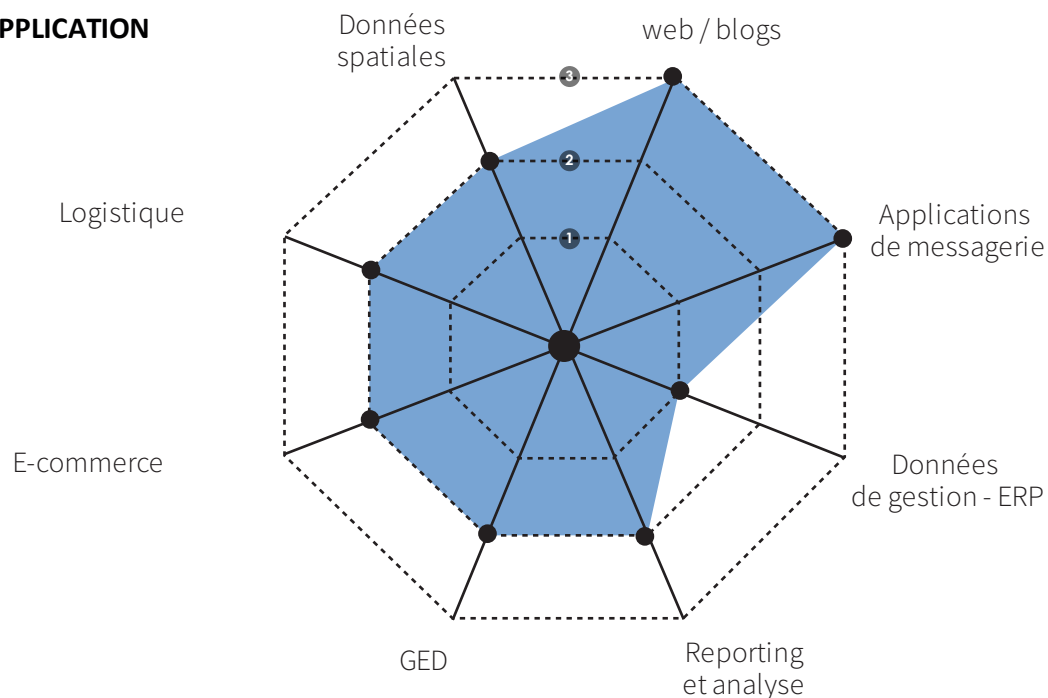
Au niveau de leur richesse fonctionnelle, ces bases sont supérieures aux modèles NoSQL clé/valeur (sans pour autant égaler les bases relationnelles). Une base comme MongoDB ou Lucene permettra de réaliser des recherches relativement poussées : full text, par synonymes, géospatiale, fuzzy, etc...

Pour revenir sur les usages, on peut souligner que les bases NoSQL orientées documents



LES BASES DE DONNÉES NoSQL ORIENTÉES DOCUMENT

TYPE D'APPLICATION



A noter : Classiquement, le traitement des données spatiales n'est pas l'apanage des bases NoSQL, ce qui explique le classement effectué dans le tableau comparatif. Cela dit, plusieurs moteurs NoSQL ont commencé à implémenter des fonctionnalités de calcul spatial, comme les calculs de distance entre les points de MongoDB.

sont particulièrement indiquées pour les blogs, ces derniers étant essentiellement constitués de textes (articles, commentaires) et de listes (mots-clés, catégories).

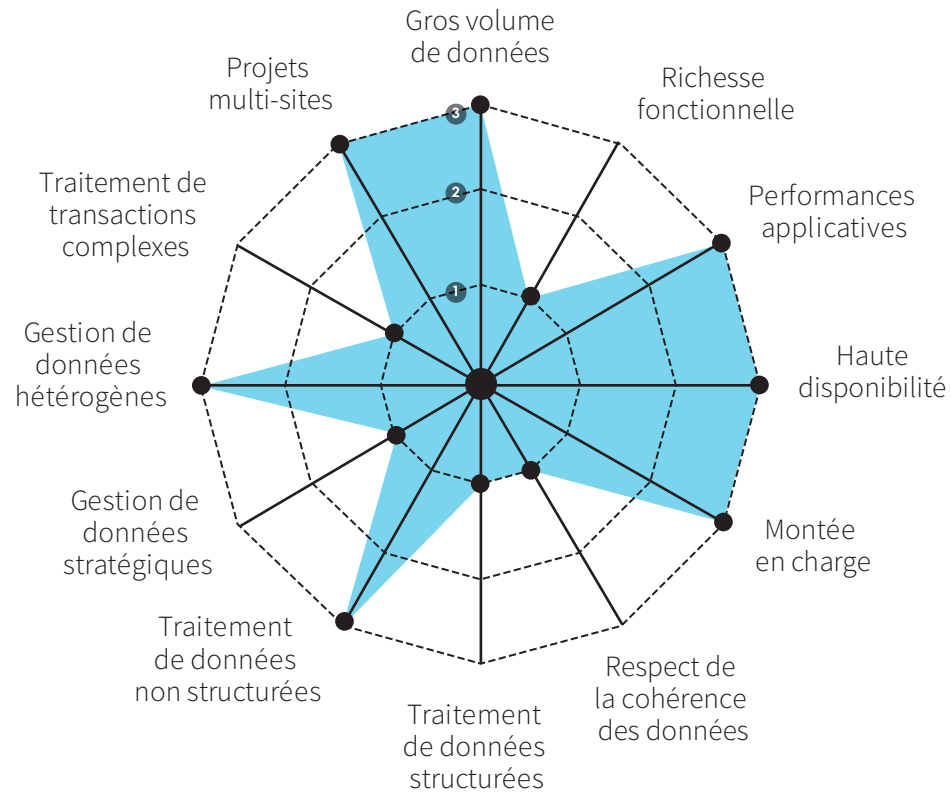
Dans son ouvrage « les bases NoSQL et le big data », Rudi Bruchez souligne l'intérêt de faire appel à ce type de base dans un tel contexte : « Un moteur NoSQL orienté documents, comme MongoDB ou CouchDB, est particulièrement indiqué pour ce genre d'application, auquel on peut ajouter les applications de messagerie, les bases de connaissance, les magazines. Il n'y a aucun avantage à utiliser un moteur relationnel dans ce type d'application.

Pour les fonctionnalités de recherche, les index des moteurs relationnels ne seraient d'aucune utilité sur du texte long. L'indexation en plein texte des documents JSON avec un outil comme Elasticsearch est donc la meilleure solution. ».



LES BASES DE DONNÉES NOSQL ORIENTÉES COLONNE

CARACTERISTIQUES DU PROJET



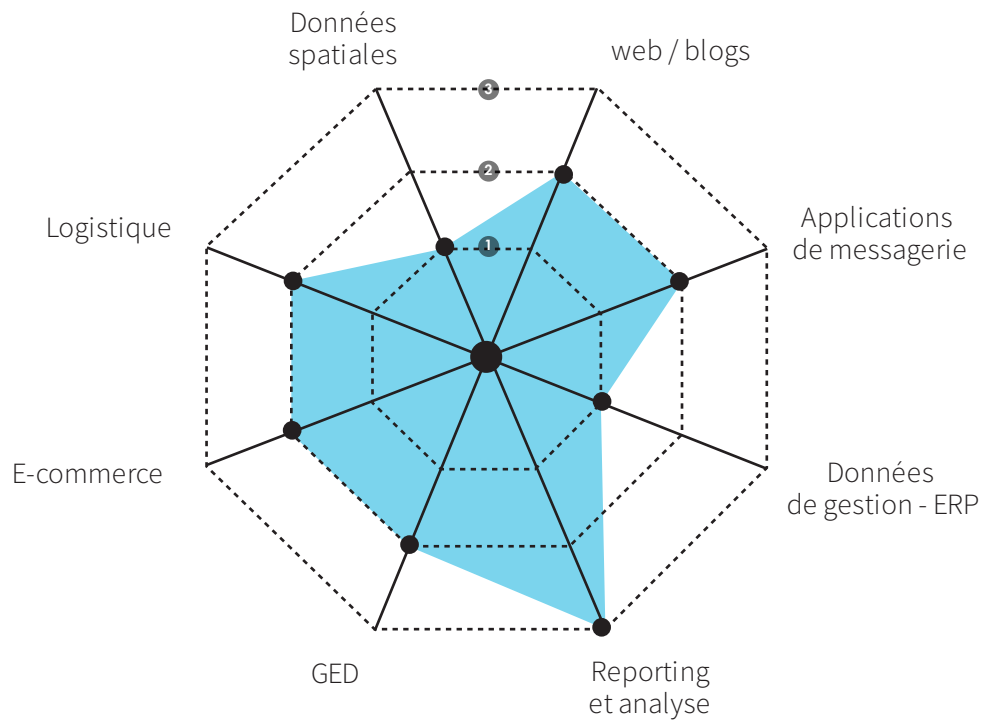
Au même titre que le modèle orienté graphe, les bases de données NoSQL orientées colonne peuvent être une bonne solution pour des usages de type reporting et applications BI.

Les besoins de reporting impliquent de faire des recherches à travers de grandes quantités de données bien structurées, afin d'effectuer des calculs d'agrégation sur plusieurs axes d'analyse. Pour ce type de tâches, on se tournera naturellement vers une base relationnelle capable de manipuler directement des structures multidimensionnelles.



LES BASES DE DONNÉES NOSQL ORIENTÉES COLONNE

TYPE D'APPLICATION



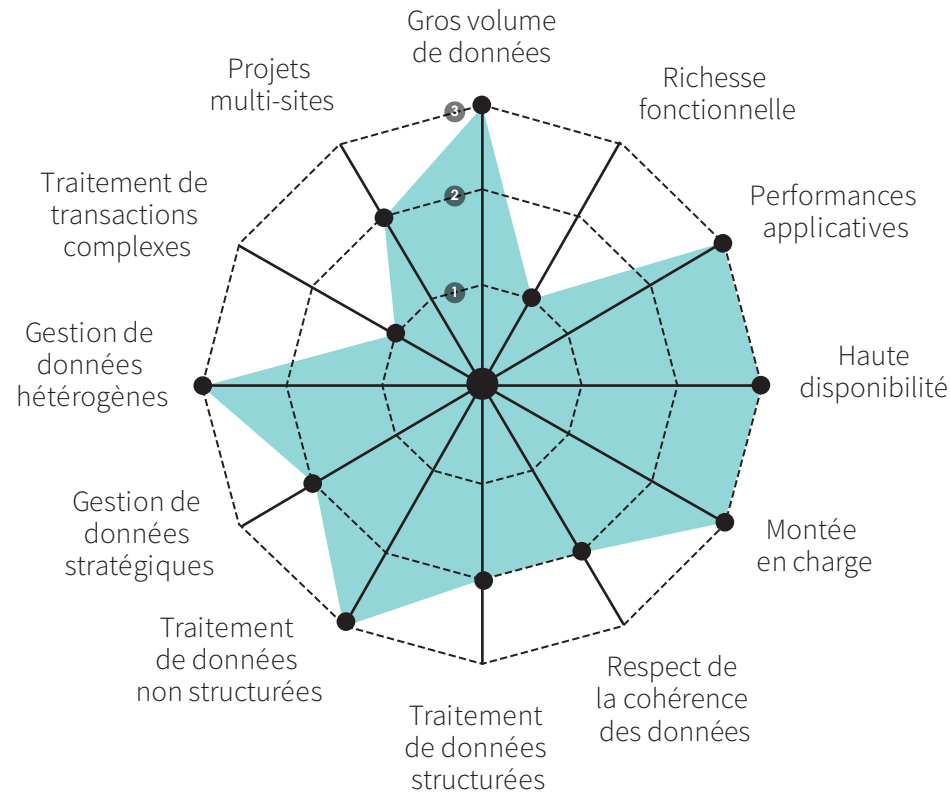
Cela dit, lorsque les volumes à traiter sont importants, la nécessité de distribuer le traitement conduira à se tourner vers le NoSQL. Dans une telle hypothèse, Rudi Bruchez conseille dans son ouvrage « Les bases NoSQL et le big data » d'opter pour « un moteur orienté colonnes, comme HBase ou Cassandra... quitte à stocker des données précalculées et dupliquées selon plusieurs clés correspondant aux axes d'analyse, et à profiter de traitements en MapReduce. »

Dans les cas d'usage les plus courants, on peut évoquer la recherche sur Internet, les applications Web à grande échelle ainsi que les applications analytiques capables de traiter des péta-octets de données.



LES BASES DE DONNÉES NOSQL ORIENTÉES GRAPHE

CARACTERISTIQUES DU PROJET



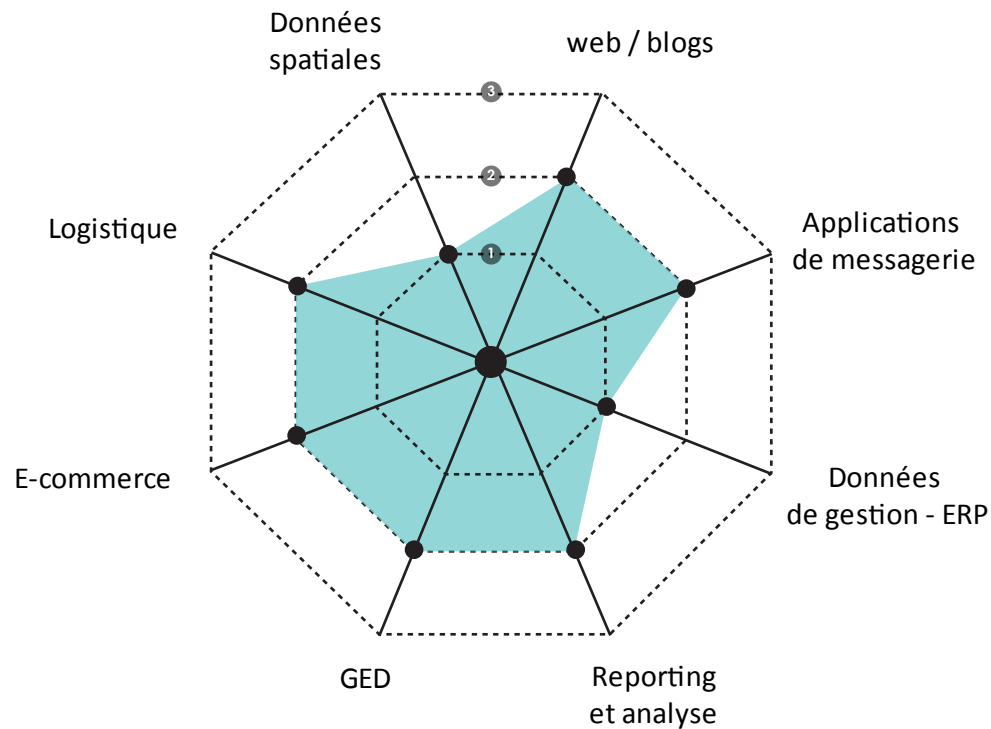
Les bases NoSQL orientées graphe sont un peu plus généralistes et peuvent convenir à des projets impliquant la gestion de données stratégiques car elles sont davantage orientées complexité que volume.

Elles répondront mieux aux besoins des applications BI et analytiques qu'aux problématiques de gestion opérationnelle, dès lors que de gros volumes de données entrent en jeu.

Par contre, si les volumes sont peu importants, une base relationnelle pourra être privilégiée. Cela s'explique par les besoins de reporting, qui impliquent de faire des recherches à travers de grandes quantités de données bien structurées afin d'effectuer des calculs d'agrégation sur plusieurs axes d'analyse.

LES BASES DE DONNÉES NOSQL ORIENTÉES GRAPHE

TYPE D'APPLICATION



C'est donc uniquement lorsque le volume de données devient important que la base NoSQL sera préférée au modèle relationnel dans le domaine de l'analytique.

Pour rester sur les usages, les bases de données orientées graphe conviendront bien aux réseaux sociaux ou aux moteurs de recommandations pour leur faculté à créer des associations entre des éléments de données.

Autre exemple, elles trouveront leur place dans le cadre du développement d'une plateforme permettant d'associer un grand nombre de requêtes et de mots clés à des informations techniques, juridiques, médicales...

TABLEAU COMPARATIF

LEGENDE : ★★★ SGBD très adapté ★★ SGBD moyennement adapté ★ SGBD peu adapté

SYSTÈME DE GESTION DE BASE DE DONNÉES	Relationnelles - SQL	NoSQL clé/valeur	NoSQL document	NoSQL colonne	NoSQL graphe
Exemples :	Oracle DB, SQL Server, MySQL	Aerospike, Redis, Riak	MongoDB, Couchbase, CouchDBServe	Solutions basées sur Hadoop	Neo4J, InfiniteGraph
CARACTERISTIQUES DU PROJET					
Gros volume de données à traiter (Big Data)	★	★★★	★★★	★★★	★★★
Besoin de richesse fonctionnelle	★★★	★	★★	★	★
Besoin de performances applicatives	★★	★★★	★★★	★★★	★★★
Haute disponibilité des données / tolérance aux pannes	★★	★★★	★★★	★★★	★★★
Capacités de montée en charge	★★	★★★	★★★	★★★	★★★
Respect de la structure / cohérence des données (attributs Acid)	★★★	★	★★	★	★★
Traitement de données structurées	★★★	★	★★	★	★★
Traitement de données non structurées	★	★★★	★★★	★★★	★★★
Gestion de données stratégiques	★★★	★	★	★	★★
Gestion de données hétérogènes	★★	★★★	★★★	★★★	★★★
Traitement de transactions complexes	★★★	★	★	★	★
Projets multi-sites / multiples datacenters	★★	★★	★★	★★★	★★
TYPE D'APPLICATION					
web / blogs	★	★★	★★★	★★	★★
Applications de messagerie	★	★★	★★★	★★	★★
Données de gestion - ERP	★★★	★	★	★	★
Reporting et analyse	★★★	★★	★★	★★★	★★
GED	★★	★★	★★	★★	★★
E-commerce	★★	★★	★★	★★	★★
Logistique	★★	★★	★★	★★	★★
Données spatiales	★★★	★	★★	★	★