

Base de données orientée graphe

Une **base de données orientée graphe** est une base de données orientée objet utilisant la théorie des graphes, donc avec des nœuds et des arcs, permettant de représenter et stocker les données.

Sommaire

- 1 Qu'est ce qu'une base de données orientée graphe ?
- 2 Avantages des bases de données orientées graphe
- 3 Comparaison avec les bases de données relationnelles
- 4 Articles connexes
- 5 Liens externes
- 6 Articles connexes

Qu'est ce qu'une base de données orientée graphe ?

Par définition, une base de données orientée graphe correspond à un système de stockage capable de fournir une adjacence entre éléments voisins : chaque voisin d'une entité est accessible grâce à un pointeur physique.

C'est une base de données orientée objet adaptée à l'exploitation des structures de données de type graphe ou dérivée, comme des arbres.

Exemple d'illustration :

En **base de données relationnelle traditionnelle** :

Table Client contient un identifiant, un nom, un prénom, une adresse, un numéro client, correspondant aux noms des colonnes respectives. Considérons le client "id01 Dupont Jean 96 rue des lilas 010203"

Cette table client est rattachée à une table entreprise, avec un numéro de SIRET, un nom, et un domaine d'activité par exemple. Considérons l'entreprise "1221 EntrepriseD automobile".

La relation entre les tables se nomme a_pour_client, et bien évidemment EntrepriseD a pour client Jean.

En **base de données orientée graphe**, la table client sera représentée par un ensemble de nœuds pour chaque instance, donc le même client sera représenté par un nœud suivant :

```
"Client :
identifiant : id01
nom : Dupont
prénom : Jean
adresse : 96 rue des lilas
numéro client : 010203".
```

L'entreprise sera représentée de la même manière par un nœud :

```
"Entreprise :
siret : 1221
nom : Entreprise D
domaine : automobile"
```

Et la relation entre les deux sera matérialisée par un arc partant du nœud Entreprise EntrepriseD vers le nœud Dupont Jean, nommée "a_pour_client".

Aussi, l'entreprise aura autant de pointeurs que de clients (chaque arc partant de l'entreprise vers le nœud correspondant avec comme nom a_pour_client).

De la même manière, le nœud client pourra pointer vers un nœud véhicule, lequel véhicule pourra pointer vers un nœud typePneu, par exemple.

Intérêt : il sera ainsi beaucoup plus facile d'accéder au type de pneu du client n° id01 par le biais du graphe puisqu'en parcourant les arcs on retrouvera directement le client, puis son véhicule, et enfin son type de pneus. En base de données relationnelle traditionnelle, en revanche, il aurait fallu faire des jointures entre la table Client, la

Système de gestion de Base de données Modèles:
▪ Base de données hiérarchique
▪ Base de données réseau
▪ Base de données relationnelle
▪ Base de données orientée objet
▪ Base de données orientée documents
▪ Base de données orientée graphe
▪ Base de données XML

table Véhicule, et la table Pneus ce qui aurait été très couteux.

L'intérêt de cette structure devient donc évident dans le cas de données complexes. C'est ainsi une structure idéale pour des recherches du type "partir d'un nœud et parcourir le graphe" plutôt que "trouver toutes les entités du type X", plus adaptées aux bases de données relationnelles traditionnelles. Il est cependant possible en base de données orientées graphes d'effectuer des recherches de ce dernier type, en utilisant un système d'indexation qui pourra selon les cas être interne au graphe (super-nœuds servant à l'indexation) ou au-dessus du graphe (via Apache Lucene par exemple).

Elle est particulièrement appropriée lorsqu'il s'agit d'exploiter les **relations** entre les données (par exemple des connaissances entre des personnes, la description de l'ensemble des pièces d'une machine industrielle et de la manière dont elles sont liées entre elles).

Les bases de données orientées graphes sont utilisées aujourd'hui dans la modélisation des réseaux sociaux : LinkedIn utilise par exemple ce système avec un graphe représentant les personnes et leur relations, et parvient ainsi facilement à afficher le degré de séparation entre chaque contact, qui n'est finalement que la distance entre les nœuds.

Elles sont de même utilisées dans le stockage de masse de données ou Big data, avec ainsi un enjeu important à l'heure actuelle dans l'exploitation des données par leur structure adaptée.

Il existe différents types de graphes permettant le stockage des données tels que :

- ▮ le graphe non orienté « single-type » ;
- ▮ l'hypergraphe ;
- ▮ les Graph Property ([en](#)) ;
- ▮ etc.

Une telle base de données se caractérise donc par les critères suivants :

- ▮ stockage des données représentées sous forme d'un graphe, avec des nœuds et des arcs ;
- ▮ lecture et parcours des données sans recours à un index, en utilisant les arcs pour passer d'un nœud à l'autre ;
- ▮ flexibilité du modèle de données ; il n'est pas nécessaire de créer une entité pour les nœuds ou les arêtes contrairement au modèle d'une base de données relationnelle ;
- ▮ intégration d'une API utilisant des algorithmes classiques de la Théorie des graphes (plus-court-chemin, Algorithme de Dijkstra, calcul de centralité...) permettant une exploitation différente des bases de données relationnelles.

Avantages des bases de données orientées graphe

Les bases de données orientées graphes apportent des avantages non négligeables :

Performances accrues :

- ▮ Traiter des données fortement connectées en évitant les multiples jointures très coûteuses qu'il faudrait mettre en œuvre dans les bases de données relationnelles traditionnelles et ainsi permettre des mises à jour très performantes, même pour un très grand ensemble de données.
- ▮ Offrir des performances exceptionnelles en termes de rapidité de temps de réponse pour les lectures locales, par parcours de graphe.

Développements simples :

- ▮ L'utilisation de langages de requêtes tels que Cypher ou Gremlin dédiés au traitement des données connectées facilite les développements. Par exemple la recherche d'amis d'amis prends une seule ligne de code en Cypher.

Modélisation facile :

- ▮ Gérer facilement un modèle complexe puisque la base de données ne s'appuie pas sur un schéma rigide.
- ▮ Permettre une modélisation parfois plus naturelle et plus lisible selon le cas d'utilisation.
- ▮ Découverte de nouveaux cas d'usages par une représentation naturelle des données.

Comparaison avec les bases de données relationnelles

Les bases de données relationnelles sont plus adaptées à des requêtes de type "trouver toutes les entités de type X" grâce aux structures internes des tables. Cela est d'autant plus vrai lorsqu'il s'agit de réaliser des opérations d'agrégation sur toutes les lignes d'une table.

En revanche elles sont moins efficaces sur l'exploitation des relations car ceci nécessite une mise en place d'index. Comme cela a été dit plus tôt, les bases de données orientées graphes permettent de parcourir les relations grâce à des **pointeurs physiques** ce qui n'est pas le cas des clefs étrangères (bases relationnelles) qui offrent uniquement des pointeurs logiques.

Les bases de données graphes sont donc plus adaptées en termes de performance lorsqu'il s'agit de manipuler des volumes de données importants et une profondeur de graphe élevée.

Articles connexes

- Neo4j
- Blazegraph
- FlockDB (en), , Base de données hiérarchique développée en Scala, s'appuie sur MySQL (initialement développé par Twitter)
- DEX (en)

Liens externes

- Neo4j (http://neo4j.com/)
- InfiniteGraph (http://wiki.infinitegraph.com/3.0/w/index.php?title=InfiniteGraph_Developer_Site)
- InfoGrid (http://infogrid.org/trac/)
- OrientDB (http://www.orientdb.org/)

Articles connexes

- Base de données relationnelle
- Base de données orientée objet
- Base de données orientée documents

Ce document provient de « https://fr.wikipedia.org/w/index.php?title=Base_de_données_orientée_graphe&oldid=119905640 ».

Dernière modification de cette page le 27 octobre 2015, à 09:41.

Droit d'auteur : les textes sont disponibles sous licence Creative Commons paternité partage à l’identique ; d’autres conditions peuvent s’appliquer. Voyez les conditions d’utilisation pour plus de détails, ainsi que les crédits graphiques. En cas de réutilisation des textes de cette page, voyez comment citer les auteurs et mentionner la licence.

Wikipedia® est une marque déposée de la Wikimedia Foundation, Inc., organisation de bienfaisance régie par le paragraphe 501(c)(3) du code fiscal des États-Unis.