**Rapport Projet OrientDB**

1. Présentation du logiciel
2. Objectif et approche

OrientDB est un système de gestion de base de données (SGBD) **NoSQL** orienté document-graphe.

Ce système est plus flexible que les systèmes relationnels car le paradigme relationnel régissant le système SQL n’a pas à être respecté intégralement. En effet dans le cadre d’une base SQL, les données sont rangées en table avec un ensemble de clefs qui rendent sa structure très rigide. Au contraire le système NoSQL est beaucoup plus flexible car les données n’ont pas à être homogènes et indexées.

L’intérêt d’une base de données orientée graphe est la flexibilité et la rapidité de parcours des données.

Premièrement les données sont parcourues sans avoir recours à un index (parcours d’une table) mais en utilisant les arcs (liens) existant entre les nœuds (individus).

Deuxièmement, il n’est pas nécessaire de créer des jointures lors des requêtes qui peuvent être très couteuses surtout lorsqu’on a beaucoup de données.

Enfin, les données pouvant être hétérogènes, le contenu de la base de données peut évoluer selon son utilisation sans nécessairement avoir eu besoin de prévoir ces changements en amont. Par exemple on peut décider d’enregistrer un nouveau type d’individus dans la base sans que l’ensemble de la structure soit remis en cause.

Pour conclure, un système NoSQL orienté document-graphe permet une gestion efficace de données massives et hétérogènes tout en jouissant d’une structure flexible.

1. Type de données

Une base de données orientée graphe est formée de noyaux (vertex) et d’arcs (edge).

Chaque noyau est un document définit par :

* Une classe
* Un identifiant
* Un ensemble d’attribut

1. Langage de requête
2. Manuel d’utilisation

* Installation
* Mis en route
* …

Objectif : Base de données type document et graphes

Type de données : données avec des relations entre les éléments. Format de fichier : fichier json

Langage de requête : SQL

1. Jeu de données

Le jeu provient de la base jeu de données publiques du logiciel OrientDB.

Il s’appelle 'Tolkien-Arda'. Il décrit l’univers des romans de Tolkien (Trilogie du *Seigneur des Anneaux* et *Le Hobbit*)

Database containing Tolkien (Middle-earth) related information like Characters, Locations and Events from his different works (<http://arda-maps.org>)

1. Python

Il existe un package de Python permettant l’utilisation de OrientDB directement sur Python. Ce package est extrêmement complexe et requiert de bonnes connaissances de programmation. Malgré de nombreuses heures de recherche passée à découvrir le package nous avons décidé de ne pas l’utiliser. En effet, le package présente deux modules : « Client » et « OGM ». Le module « Client est le module de « base ». Il permet la réalisation de taches simple : se connecter à un serveur, manipuler les bases de données et faire des requêtes SQL. Il y a plusieurs reproches que nous pouvons faire à PyOrient. Tout d’abord, il est très compliqué de se connecter à un serveur. En effet, il existe un problème de version du serveur rendant très compliqué la connexion. Il faut modifier des paramètres de classe des objets ??? . Le « bug » est reporté à de nombreuses reprises sur le web. Nous avons tenté plusieurs stratégies de résolution du bug. Malgré tout, aucune n’a été 100% efficace sur l’ensemble de nos machines.

La deuxième difficulté que présente PyOrient est la non-lisibilité du résultat des requêtes sur Python directement. Une fois le résultat stocké dans une variable, il est nécessaire d’utiliser des outils de code comme Flask ou Django pour lire les résultats. Cela nous semblait ambitieux et de plus, pour un résultat incertain car nous n’aurions certainement pas obtenus de graphes mais des lignes de résultats ?. L’utilisation de l’ORM est encore plus complexe car réservée aux bases de taille importante et aux grands graphes.