**Berkeley DB** fait partie de la famille des bases de données **« clé-valeur »** et est sans doute le produit de cette famille le plus utilisé au monde avec plusieurs dizaines de millions de déploiements revendiqués par l'éditeur. La première version de BDB remonte à 1986 et est la propriété d'Oracle depuis février 2006.

* La famille de Berkeley DB :

1. Oracle BDB Core : Cette version de Berkeley DB est écrite en C ANSI et peut être utilisée comme une bibliothèque pour accéder aux données persistées. Oracle BDB fournit de nombreuses interfaces pour différents langages de programmation (C, Perl, PHP, Python…).
2. Oracle BDB XML Edition : Cette version est une surcouche de BDB. Cette version XML permet de stocker et récupérer facilement des documents XML. L'utilisation de XQuery permet d'accéder aux documents XML stockés.
3. BDB Java Edition : Cette version de Berkeley DB, 100 % Java est une base embarquée adoptant le format de stockage clé-valeur. De plus, BDB JE est très performante et flexible. Cette base supporte les transactions, permet de stocker des objets Java en utilisant seulement quelques annotations.

* La famille de Berkeley DB :

Chaque membre de la famille BDB possède différentes fonctionnalités. Dans toutes les familles de BDB, nous pouvons remarquer les fonctionnalités suivantes :

|  |  |
| --- | --- |
| **Feature Set** | **Description** |
| Data Store (DS) | 1 Writer and n reader |
| Concurrent Data Store (CDS) | n writers and n snapshot reader |
| Transactional Data Store (TD) | Full ACID support on top of CDS |
| High Availability (HA) | Replication for fault tolerence. Fail over recovery support |

Le tableau suivant décrit les différences entre BDB "core" et BDB JE :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **DS** | **CDS** | **TS** | **HA** |
| BDB/BDB XML | X | X | X | X |
| BDD JE |  | X | X |  |

* Introduction à Berkeley DB Java Edition :

1. Installation :

Tout d'abord, débutons par le commencement. Pour pouvoir utiliser BDB Java Edition, vous devez l'installer à l'adresse suivante :

<http://www.oracle.com/technetwork/products/berkeleydb/downloads/index.html>. L'installation de Berkeley DB est assez simple, il suffit de décompresser l'archive que vous venez de télécharger ! Le seul fichier dont vous aurez besoin pour faire fonctionner les exemples que nous verrons plus tard est le : "je-\*.jar".

1. Les API d'accès aux données :

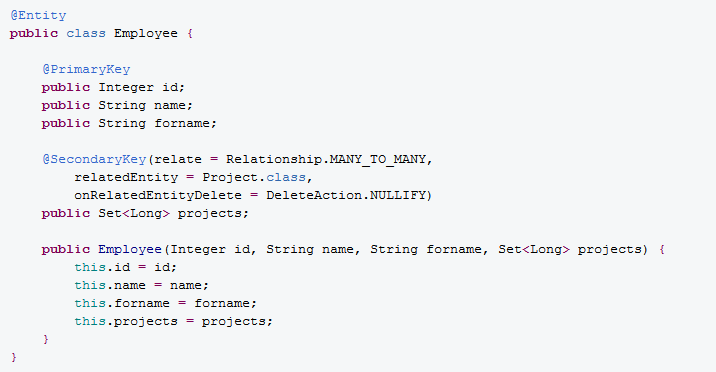
* l'API de base fournit un simple modèle clé-valeur pour stocker et récupérer les données.
* l'API DPL (Direct Persistence Layer) qui vous laisse persister n'importe quelle classe Java comportant un constructeur par défaut en base de données. Cette API vous permet de récupérer les données avec des fonctionnalités de recherche assez riches.
* la Collection API qui étend la Java Collection API avec le support de la persistance et des transactions en plus de l'accès aux données.

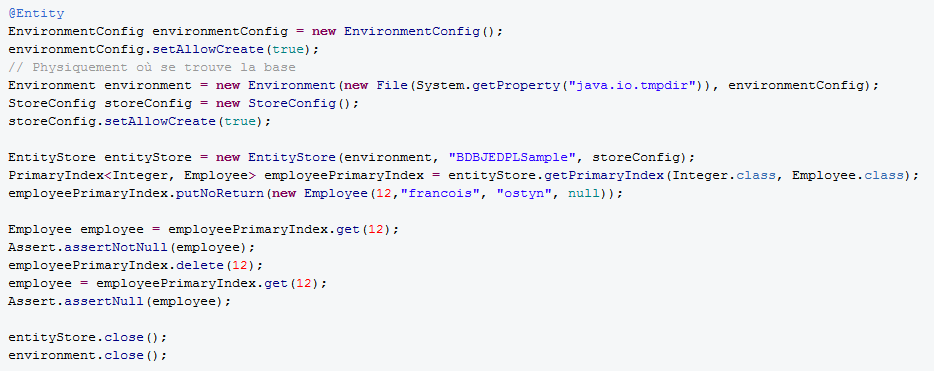
**L'API de base:**

Tout d'abord, avant de commencer à vous parler de DPL, même si nous sommes dans l'univers « NoSQL », il est toujours possible de faire une analogie avec le SQL :

|  |  |
| --- | --- |
| **Terme SQL** | **Équivalence Oracle DPL** |
| Database | Environment |
| Table | Primary Index and its related secondary indices |
| Primary Index | Primary Index |
| Secondary Index | Secondary Index |
| Tuple/Row | Entity |

L'exemple que je vais vous présenter se décompose en deux parties. Premièrement, un entity bean et une classe de management qui gère le CRUD (Create-Replace-Update-Delete) au-dessus de l'entity bean.



Le code ci-dessous nous montre comment accéder aux données présentes dans le POJO Employee :

Dans l'exemple ci-dessus nous ne gérons ni les transactions, ni des relations complexes.

Description de l'environnement de Berkeley DB Java Edition :

Une base de données BDB JE est composée d'un ou plusieurs fichiers de logs qui sont situés dans le répertoire d'environnement.

Par défaut, leur taille est de 10 Mb et peut être modifiée (par exemple, augmenter la taille des logs permet de limiter le nombre d'I/O).

Le support des transactions :

Pour gérer les transactions avec DPL, il faut faire quelques modifications dans son code.