

A. Les entrepôts

Afin de découvrir des tendances, les analystes doivent disposer d'une grande quantité de données. Ceci est en opposition avec des systèmes OLTP pour lesquels les données historiques doivent être archivées à intervalles réguliers pour ne pas engorger les bases de données.

Les outils Oracle Business Intelligence peuvent interroger directement des systèmes transactionnels, mais comme la plupart des interrogations nécessitent de conserver un historique, les données doivent être structurées en vue des analyses futures.

1. La consolidation des données

Un entrepôt de données (*data warehouse*) est une base de données architecturée pour des requêtes et des analyses plutôt que pour un traitement transactionnel des données. Les entrepôts de données séparent les opérations analytiques des opérations transactionnelles.

a. Architecture des entrepôts

Il existe différentes architectures pour les entrepôts en fonction de l'organisation mise en place. Les schémas suivants représentent les architectures les plus communes.

Schéma 1

Les utilisateurs accèdent directement à l'entrepôt, qui rassemble, dans le schéma suivant, des données OLTP, des métadonnées et des données déjà agrégées pour diminuer le temps d'exécution des requêtes.

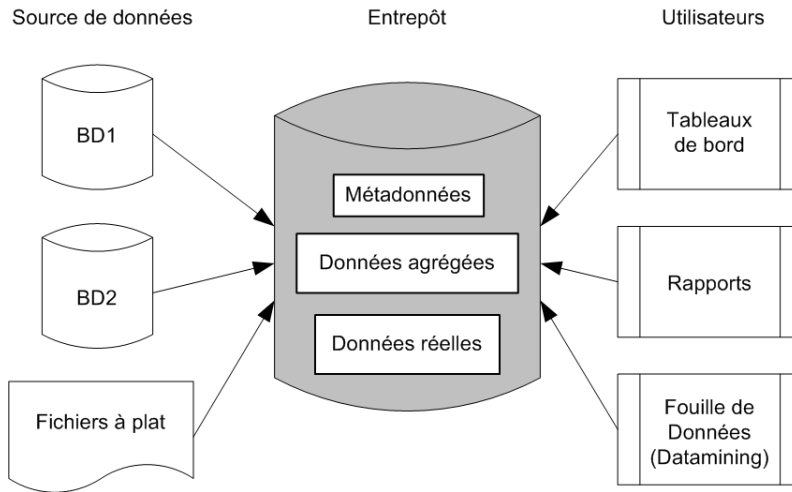


Schéma 2

Mais lorsque les données ne peuvent pas être épurées ou directement mises en forme dans l'entrepôt, elles sont au préalable stockées dans une zone tampon (*staging area*). L'architecture de l'entrepôt est alors similaire au schéma suivant.

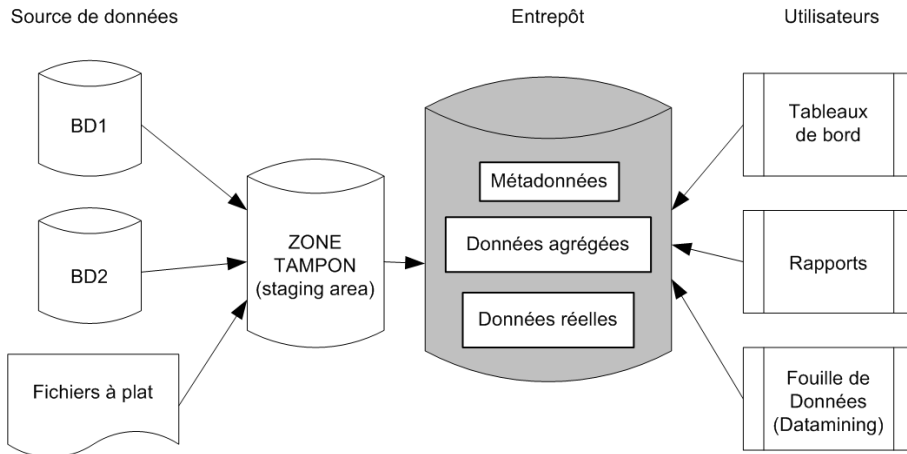
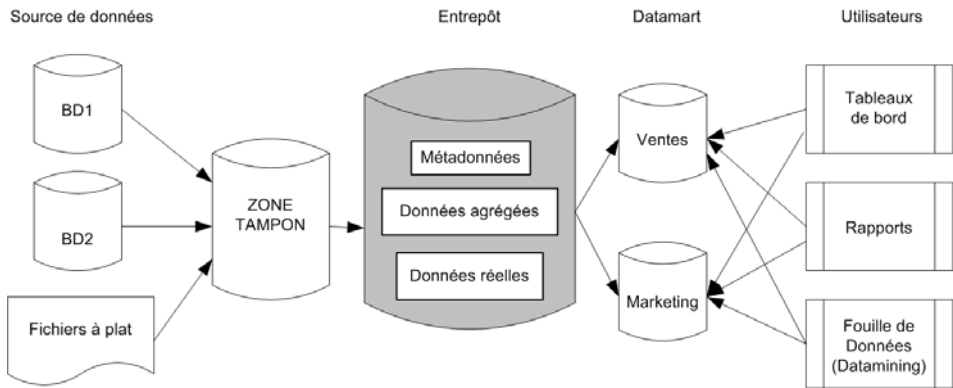


Schéma 3

Lorsque les utilisateurs des différents services de l'entreprise ont des besoins spécifiques, distincts les uns des autres, l'administrateur crée un data mart par service afin d'isoler les données métier.



b. Différence entre systèmes OLTP et entrepôts

La liste suivante résume quelques différences entre des systèmes OLTP et des entrepôts.

Débit

Un entrepôt de données est organisé pour supporter des requêtes très variées.

Modification des données

Les données d'un entrepôt ne sont pas modifiées manuellement mais mises à jour par un processus ETL.

Architecture de la base de données

Les entrepôts ont souvent une architecture en étoile ou en flocons. La plupart des entrepôts utilisent un modèle dimensionnel. Les bases de données multidimensionnelles n'obéissent pas aux mêmes règles que les bases de données OLTP.

c. Architecture matérielle et entrées/sorties

Il n'existe pas une mais plusieurs approches pour déployer une solution Business Intelligence.

Côté matériel, le dimensionnement de l'espace disque est accessoire car l'ajout d'espace disque ne pose aucun problème particulier. Cependant, l'administrateur doit être particulièrement attentif au débit du sous-système d'entrées/sorties du serveur et s'assurer que la configuration disque retenue sera compatible avec les besoins.

L'architecture **OMF** (*Oracle Managed Files*) décharge l'administrateur de la gestion des noms de fichiers lorsque leur nombre (plusieurs centaines de fichiers) devient trop important.

L'architecture **ASM** (*Automatic Storage Management*) décharge l'administrateur de la gestion des disques (*striping*, miroir, équilibrage de charge) lorsque leur nombre devient trop important.

Le partitionnement

Avec des tables de plusieurs millions d'enregistrements, l'administrateur devra partitionner les tables.

Lorsque l'entrepôt conserve une fenêtre de données sur n années, les partitions **BY RANGE** optimisent la gestion des ensembles.

```
SQL> CREATE TABLE VENTE_CL
 2  (idvente NUMBER(15),
 3  idrepresentant VARCHAR2(30),
 4  montant NUMBER(10),
 5  datecommande DATE)
 6  COMPRESS
 7  PARTITION BY RANGE(datecommande)
 8  (
 9  PARTITION v_04 VALUES LESS THAN(TO_DATE('01/01/2005','DD/MM/YYYY')),
10  PARTITION v_05 VALUES LESS THAN(TO_DATE('01/01/2006','DD/MM/YYYY')),
11  PARTITION v_06 VALUES LESS THAN(TO_DATE('04/01/2007','DD/MM/YYYY'))
12  );
```

```
Table créée.
SQL >
```

L'instruction **ALTER TABLE** associée aux clauses suivantes permet une gestion simple et très efficace des données partitionnées :

- **ADD PARTITION, MOVE PARTITION, MODIFY PARTITION ;**
- **TRUNCATE PARTITION, DROP PARTITION ;**
- **RENAME PARTITION ;**
- **SPLIT PARTITION, MERGE PARTITIONS ;**
- **EXCHANGE PARTITION.**

```
-- pour ajouter de nouvelles données
SQL> ALTER TABLE VENTE_CL ADD PARTITION v_07 VALUES LESS THAN (MAXVALUE);
Table modifiée.
SQL>

-- pour retirer des données périmées
SQL> ALTER TABLE VENTE_CL DROP PARTITION vente_2004;
Table modifiée.
SQL>
```

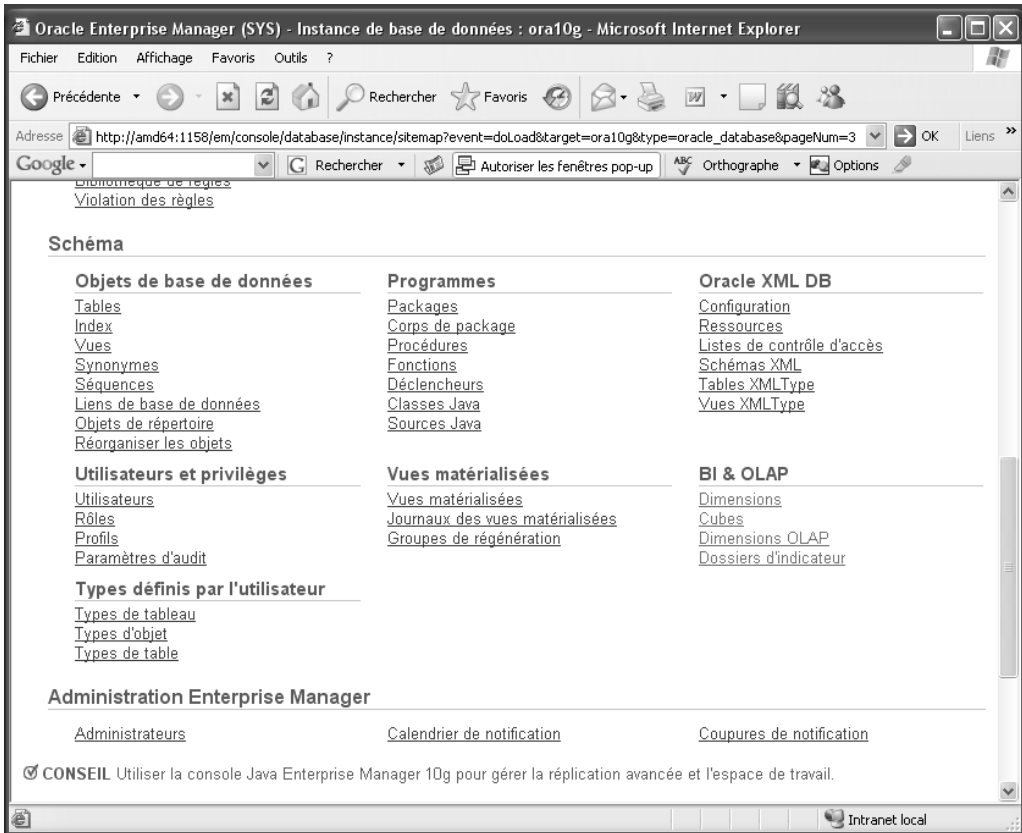
B. OLAP et ORACLE

Le moteur analytique **OLAP**, inclus dans le moteur des bases de données Oracle 10g, prend en charge les données multidimensionnelles.

Avec Oracle 10g, vous n'avez pas besoin d'installer une base relationnelle d'un côté et une base OLAP de l'autre car les bases de données Oracle 10g supportent les deux environnements à la fois. L'administrateur Oracle (**DBA**) dispose des mêmes outils pour administrer une base relationnelle ou une base multidimensionnelles.

1. Oracle Enterprise Manager Database Console

Dans la page d'accueil d'**Oracle Enterprise Manager Database Console**, cliquez dans l'onglet **Administration** pour faire apparaître les liens disponibles dans la zone **BI & OLAP**.



L'option **OLAP** dans une base Oracle 10g représente la première pierre pour l'édifice BI car cette option apporte un moteur multidimensionnel et offre aux bases Oracle :

- une interface de programmation SQL (**CREATE DIMENSION, ALTER DIMENSION, DROP DIMENSION...**) ;
- la gestion graphique des objets **Dimension**, des **cubes...** ;
- une API OLAP ;
- un catalogue OLAP.