

Les SGBD

■ Introduction au cours

- Guillaume Moulard
Architecte Technique
DVSI/DPS/DPI
France Telecom
- guillaume.moulard@francetelecom.com

Pour en savoir plus ...

- Introduction aux BD de C.J. Date Ed.VUIBERT
- Ingénierie des systèmes d'information Merise
D. NANCI et B. ESPINASSE Ed.SYBEL
- Comprendre MERISE J.P. MATHERON
Ed. EYROLLES.
- Documentation Oracle : Concepts Volume 1&2
<http://www.oracle.com/> & <http://docs.oracle.com/>
- les ouvrages de CODD, GARDARIN, TARDIEU,
MIRANDA, ...

Thème 1/2

- Historique des SGF/SGBD
 - La gestion de fichier
 - Les objectifs des SGBD
 - Algèbre relationnelle et Indexation (btree, bitmap,...)
- L'état de l'art :
 - SGBD/R -> Oracle
- La modélisation des données
Merise et UML

Thème 2/2

- Les structures internes d'un SGDB
SQL présentation du langage
- TP SQL (create, insert, update, select, delete).
- Correction TP
Evaluation

- 3 cours + 2 TP

Historique : des SGF aux SGBD

- Les principes théoriques du modèle relationnel furent proposés en 1970 par le mathématicien E.F. Codd
 - **1. Représentation des informations**
 - Les informations sont représentées au niveau **logique**, et non physique, ce qui signifie que l'on ne se préoccupe pas de l'implémentation réelle des données. Elles sont décrites par des valeurs contenues dans des tables, organisées en lignes et colonnes. (*donc presque comme dans un tableur...*)
 - **2. Accès aux données**
 - Une donnée est accessible logiquement, c'est à dire sans connaissance de son implantation physique, grâce à la combinaison du nom de la table, de la clé primaire (ou d'un numéro d'article) et du nom de la colonne (*du nom du champ*).

Codd

- **3. Gestion des valeurs absentes**
 - Cette règle précise que la valeur NULL (celle qui correspond à une absence d'information) est interprétable, et de valeur différente d'une chaîne de caractères vide (ou composée de "blancs") ou encore d'une valeur numérique égale à zéro.
- **4. Dictionnaire de données**
 - La description de la base de données est représentée par des informations accessibles comme s'il s'agissait de données ordinaires. Le langage relationnel permet donc en principe de manipuler indifféremment des données du système d'information ou des données décrivant la base elle-même. La description de la base est donc stockée dans des tables faisant partie de ce que l'on nomme le dictionnaire de données.

Codd

■ 5. Le langage

- Le SGBD doit inclure au moins un langage comportant l'ensemble des fonctionnalités suivantes : définition des données, définition des vues, manipulation des données, contraintes d'intégrité, autorisations, gestion des transactions.

■ 6. La mise à jour à travers une vue

- Une vue est un mode de représentation logique de la base de données. Si une vue peut être mise à jour, elle peut aussi l'être par le système.

■ 7. La mise à jour des tables

- Le langage relationnel doit disposer d'ordres de haut niveau s'appliquant non seulement à la lecture des données, mais aussi à la création, la mise à jour ou la suppression d'informations.

Codd

■ 8. L'indépendance physique

- Les programmes d'applications et les transactions interactives sont indépendants de la représentation physique des données et des méthodes d'accès sous-jacentes. Cela garantit la souplesse d'évolution du système d'information et exige que le SGBD dissocie efficacement la représentation logique d'une part et les aspects d'organisation physique d'autre part.

■ 9. L'indépendance logique

- Cette règle stipule que les programmes ne sont pas remis en cause lorsque des modifications (sans perte d'informations structurelles) sont opérées sur les relations de la base. Citons par exemple l'éclatement d'une table en deux, ou, à l'opposé, la fusion de deux tables.

Codd

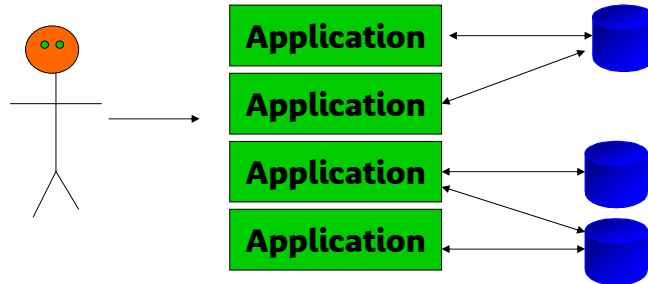
- **10. L'indépendance vis-à-vis des contraintes d'intégrité**
 - Les contraintes d'intégrité susceptibles d'évoluer dans le temps doivent pouvoir être formulées en dehors de tout programme applicatif et être référencées dans le dictionnaire des données. Une application informatique constitue une réponse à un problème qui se pose à un moment précis. Si l'entreprise et son environnement évoluent, le système d'information doit par conséquent pouvoir s'adapter à cette évolution, sans que soit remis en cause l'acquis applicatif.
- **11. L'indépendance vis-à-vis de la répartition des données**
 - L'environnement applicatif n'est pas affecté par la répartition des données sur des sites et des supports physiques distincts, ou toute modification de cette éventuelle répartition.

Codd

- **12. La non-subversion**
 - Cette dernière règle stipule que si le système dispose d'un langage de bas niveau, ce langage ne peut pas contourner ou remettre en cause les contraintes de sécurité et les règles d'intégrité énoncées au plus haut niveau.

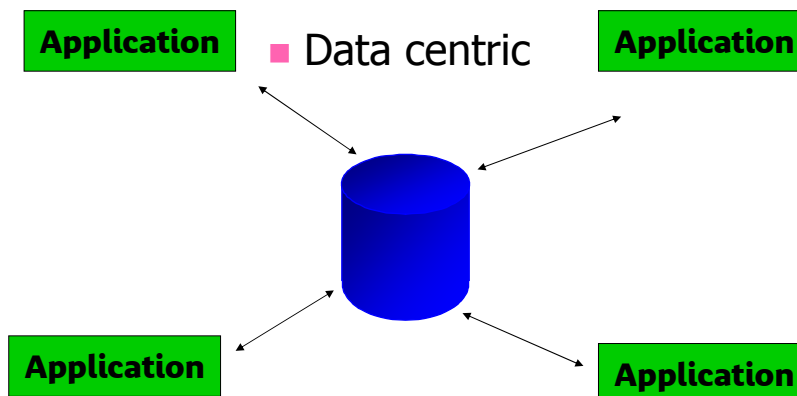
Place des SGBD dans les systèmes d'informations

■ Services centric



Place des SGBD dans les systèmes d'informations

■ Data centric



La gestion de fichier

- Organisation des fichiers
 - Séquentielle: stockage dans l'ordre d'arrivée
 - Relative : enregistrement de taille fixe
 - Aléatoire : accès par une fonction de hachage
 - Indexée : accès direct à l'enregistrement par l'index

Les objectifs des SGBD

- Gain en productivité et en performance
- Indépendance des programmes vis à vis des données
- Non redondance des données
- Accès par des langages de haut niveau
- Administration des données
- Cohérence et sécurité des données
- Optimisation des accès
- Partageabilité des données

Quelques chiffres

	Années hommes de développement	Nombre de ligne de code en millier
Compilateur	10 → 200	20 → 300
SGBD	300 → 500	300 → 600
Soft de GPAO	300 → 500	500 → 1 000
OS	2 500 → 5 000	5 000 → 10 000

Le marché des Base de donnée selon IDC

WORLDWIDE RDBMS SOFTWARE REVENUE BY VENDOR, 2000-2002 (\$M)

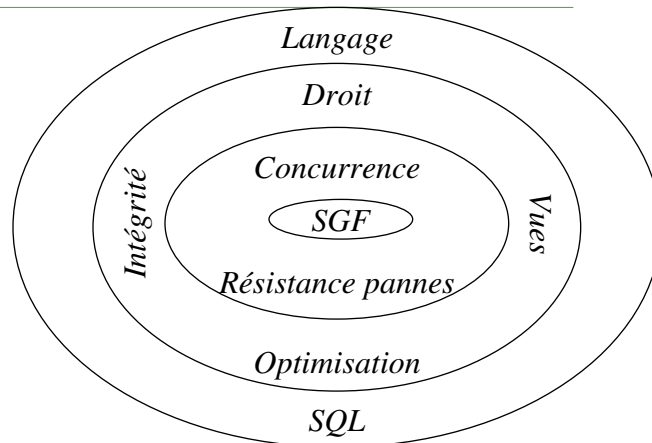
	2000	2001	2002	2001 Share (%)	2002 Share (%)	2001-2002 Growth (%)
Oracle	5,667.2	5,365.0	5,096.2	41.7	39.4	-5.0
IBM	3,087.0	3,991.0	4,350.2	31.0	33.6	9.0
Microsoft	983.9	1,250.7	1,438.3	9.7	11.1	15.0
Sybase	521.5	495.0	464.0	3.8	3.6	-6.3
NCR Teradata	205.7	207.2	222.5	1.6	1.7	7.4
Other	2,210.2	1,550.5	1,379.5	12.1	10.7	-11.0
Total	12,675.5	12,859.3	12,950.8	100.0	100.0	0.7

Source: IDC, 2003

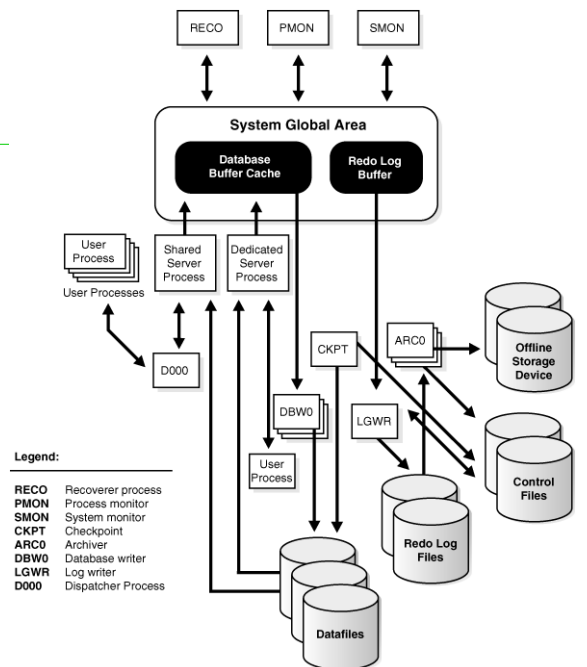
Différent type de SGBD

- LDAP → Pour la gestion des personnes et des organisation
- SGBD-OO (Orienté Objet) → Pour les applications qui gère un grand nombre de type de données différents.
- SGBD-R → pour les applications de gestion

L'architecture d'un SGBD



L'architecture d'un SGBD (ii)

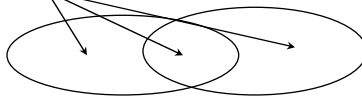


Algèbre relationnelle

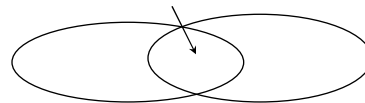
- C'est la base théorique des SGBD
- C'est une implémentation de la théorie des ensembles
- Elle se compose :
 - Opérateurs ensemblistes
 - Union, Intersection, différence, produit cartésien
 - Opérateurs relationnels
 - Restriction, Projection, Jointure, Division.

Opérateurs ensemblistes

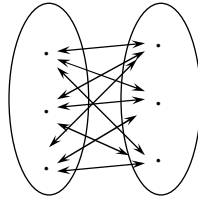
Union



Intersection



produit cartésien



différence



Opérateurs relationnels

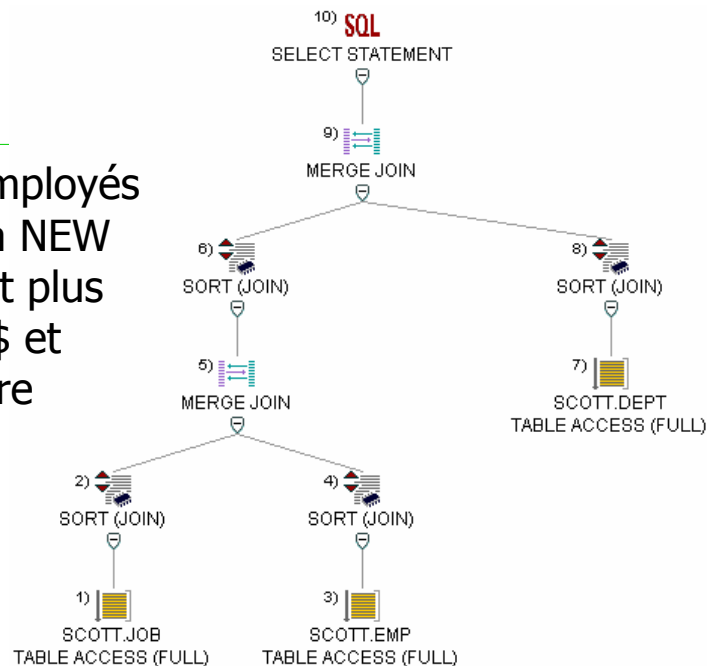
- Restriction
 - Obtention des tuples satisfaisants le critère
- Projection
 - Elimination des attributs non désirés
- Jointure
 - Composition de deux relations sur un domaine commun
- Division
 - Ensemble des tuples qui ont l'ensemble des éléments d'un ensemble

Algèbre relationnelle : Exemple

- Nœuds feuilles = relation de base
 - EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, COMM, DEPTNO)
 - DEPT (DEPTNO, DNAME, LOC)
 - JOB (NAME, ADVANTAGE)
- Nœuds intermédiaires = opérateurs
- Nœuds racines = résultats
- Arc = flux de données

Graphe

- Liste des employés qui sont a NEW YORK, ont plus de 800 K\$ et une voiture



Plan exécution par Oracle (1/2)

- 1 Cette étape du plan extrait toutes les lignes de la table JOB.
- 2 Cette étape du plan accepte un ensemble de lignes (son unique enfant) et le trie en vue d'une opération de fusion/jointure.
- 3 Cette étape du plan extrait toutes les lignes de la table EMP.
- 4 Cette étape du plan accepte un ensemble de lignes (son unique enfant) et le trie en vue d'une opération de fusion/jointure.
- 5 Cette étape du plan accepte deux ensembles de lignes triés en fonction de la clé de jointure. En parcourant les deux ensembles de lignes dans l'ordre de la clé de jointure, chaque paire de lignes distincte remplissant la condition de jointure figurant dans la clause WHERE est détectée au cours d'un seul passage des ensembles de lignes.

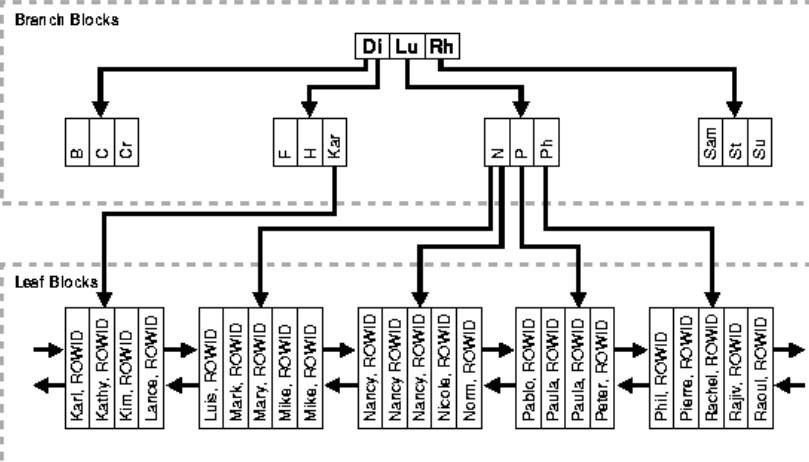
Plan exécution par Oracle (2/2)

- 6 Cette étape du plan accepte un ensemble de lignes (son unique enfant) et le trie en vue d'une opération de fusion/jointure.
- 7 Cette étape du plan extrait toutes les lignes de la table DEPT.
- 8 Cette étape du plan accepte un ensemble de lignes (son unique enfant) et le trie en vue d'une opération de fusion/jointure.
- 9 Cette étape du plan accepte deux ensembles de lignes triés en fonction de la clé de jointure. En parcourant les deux ensembles de lignes dans l'ordre de la clé de jointure, chaque paire de lignes distincte remplissant la condition de jointure figurant dans la clause WHERE est détectée au cours d'un seul passage des ensembles de lignes.
- 10 Cette étape du plan définit cette instruction comme instruction SELECT.

Langage : SQL

```
SELECT  ENAME
FROM    EMP E, DEPT D, JOB J
WHERE   E.DEPTNO = D.DEPTNO
AND     E.JOB = J.NAME
AND     E.SAL > 800
AND     D.LOC = 'NEW YORK'
AND     J.ADVANTAGE = 'CAR'
```

Indexation : btree



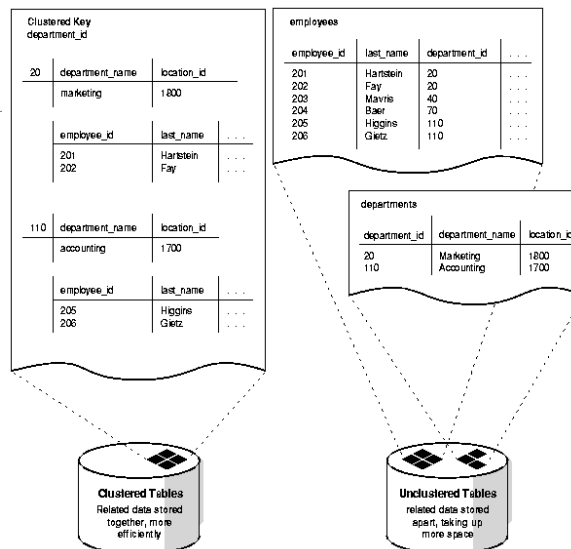
Indexation : bitmap

CUSTOMER #	MARITAL STATUS	REGION	GENDER	REGION='east'	REGION='central'	REGION='west'
101	single	east	male	1	0	0
102	married	central	female	0	1	0
103	married	west	female	0	0	1
104	divorced	west	male	0	0	1
105	single	central	female	0	1	0
106	married	central	female	0	1	0

MARITAL_STAT = 'married' AND REGION IN ('central', 'west');

status = 'married'	region = 'central'	region = 'west'				
0	0	0		0	0	0
1	1	0		1	1	1
1	0	1		1	1	1
0	0	1	=	0	1	0
0	1	0		0	1	0
1	1	0		1	1	1

Indexation Clusters

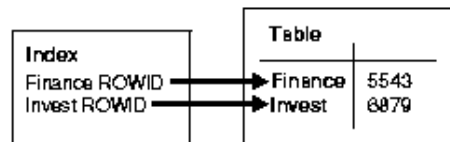


Function-Based Indexes

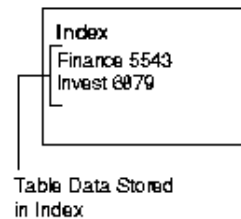
- CREATE INDEX uppercase_idx ON emp (UPPER(empname));
- SELECT * FROM emp WHERE UPPER(empname) = 'MARK';

Index-Organized Tables

Regular Table and Index

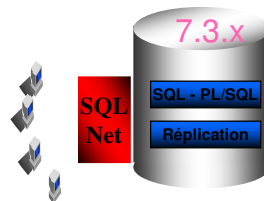
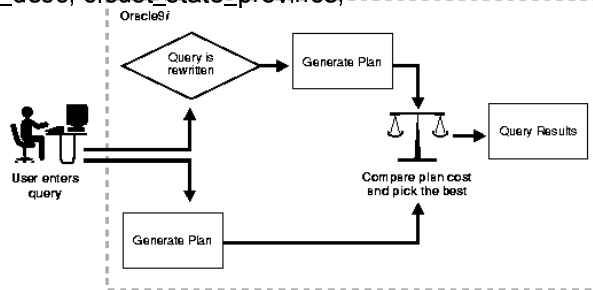


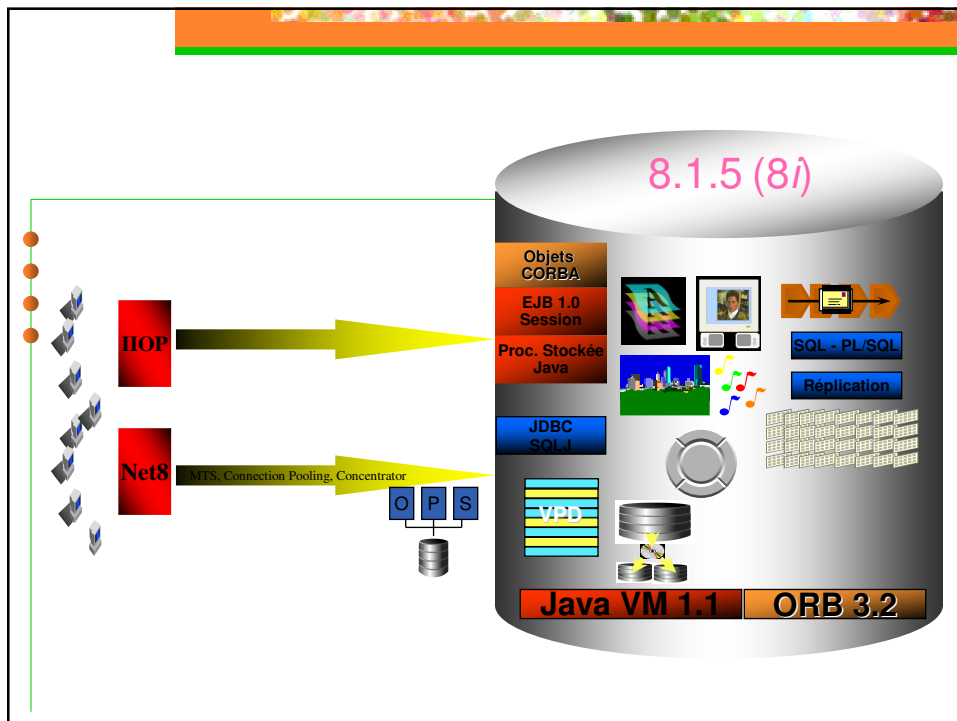
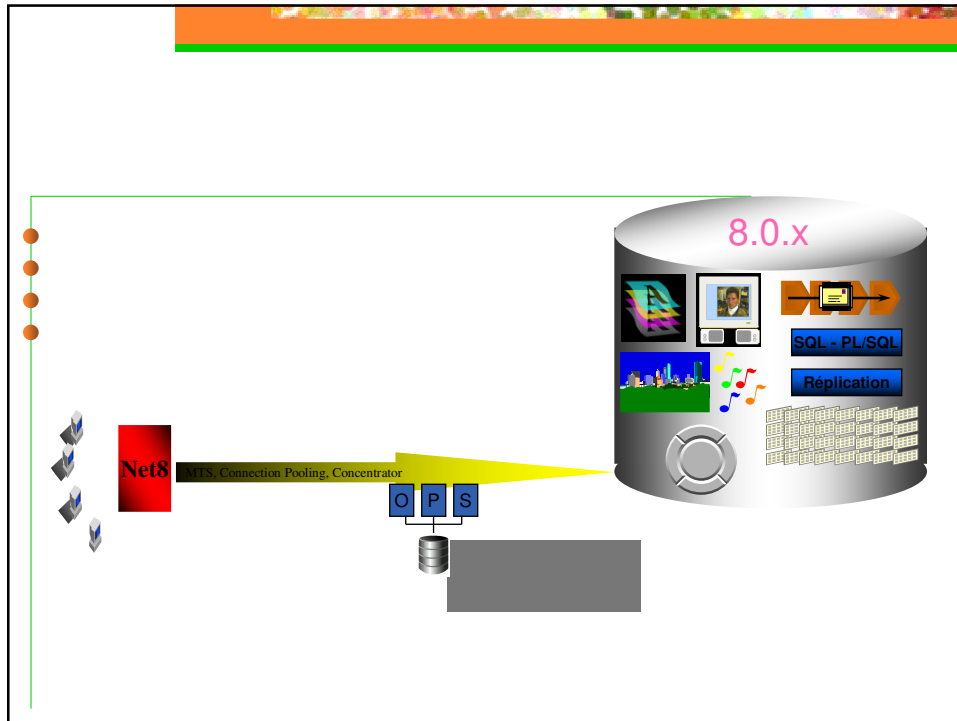
Index-Organized Table

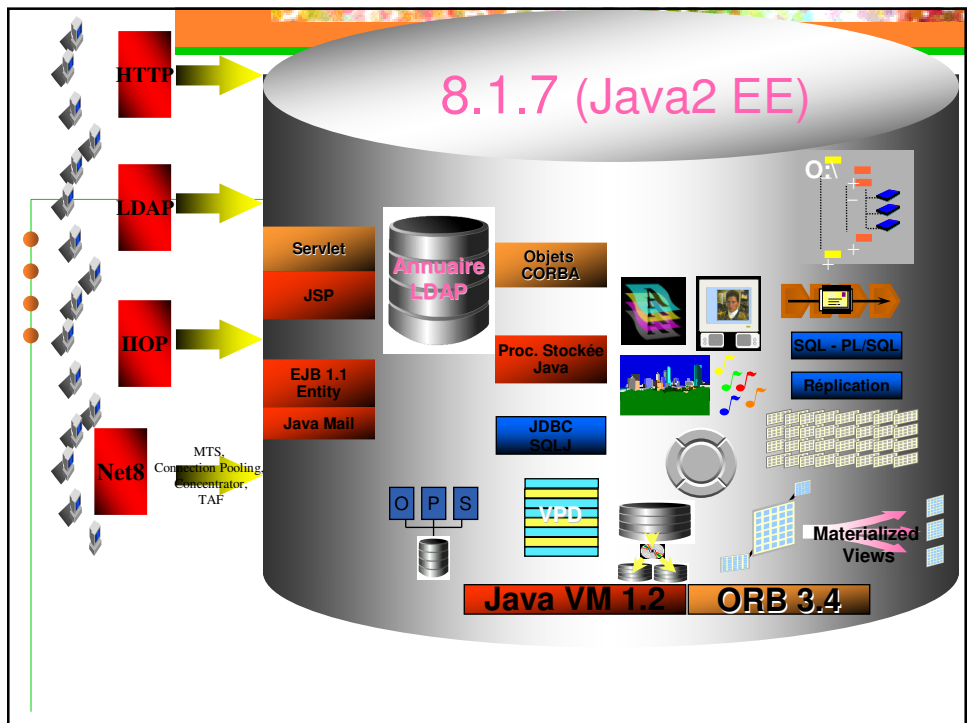
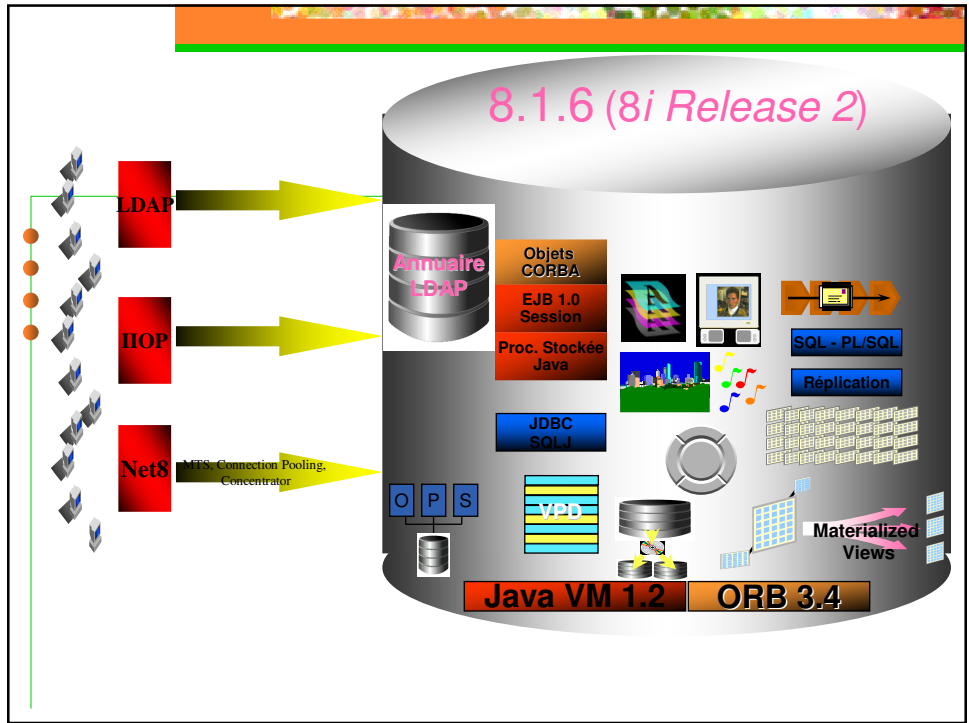


Materialized Views

```
CREATE MATERIALIZED VIEW sales_sum_table  
ENABLE QUERY REWRITE AS  
SELECT t.calendar_month_desc AS month, c.cust_state_province  
AS state,  
SUM(s.amount_sold) AS sales  
FROM times t, customers c, sales s  
WHERE s.time_id = t.time_id AND s.cust_id = c.cust_id GROUP  
BY t.calendar_month_desc, c.cust_state_province;
```







Oracle 9i

- Data Encryption and Label Security
- Real Application Clusters
- XML DB a XML TOOLS
- Public Key Infrastructure
- Database Globalization Support
- Data Guard-Logical Standby
- Flashback Query

SGBD Objet – Versant Les Règles d'Or

- | ➤ Règles orientées-objet | ➤ Règles base de données |
|-----------------------------------|--------------------------|
| ○ Identité d'objet | ○ Persistance |
| ○ Objet complexe | ○ Gestion de mémoires |
| ○ Encapsulation | ○ Langage de requêtes |
| ○ Type ou Classe | ○ Multi-usager |
| ○ Héritage | ○ Fiabilité |
| ○ Surcharge et résolution tardive | ○ Sécurité |

SGBD Objet – Versant TRANSPARENCE DES SGBDOs

- Transparence des fonctionnalités de stockage et d'accès aux objets
- Pas de transcodage entre l'interface Objet et la Base

