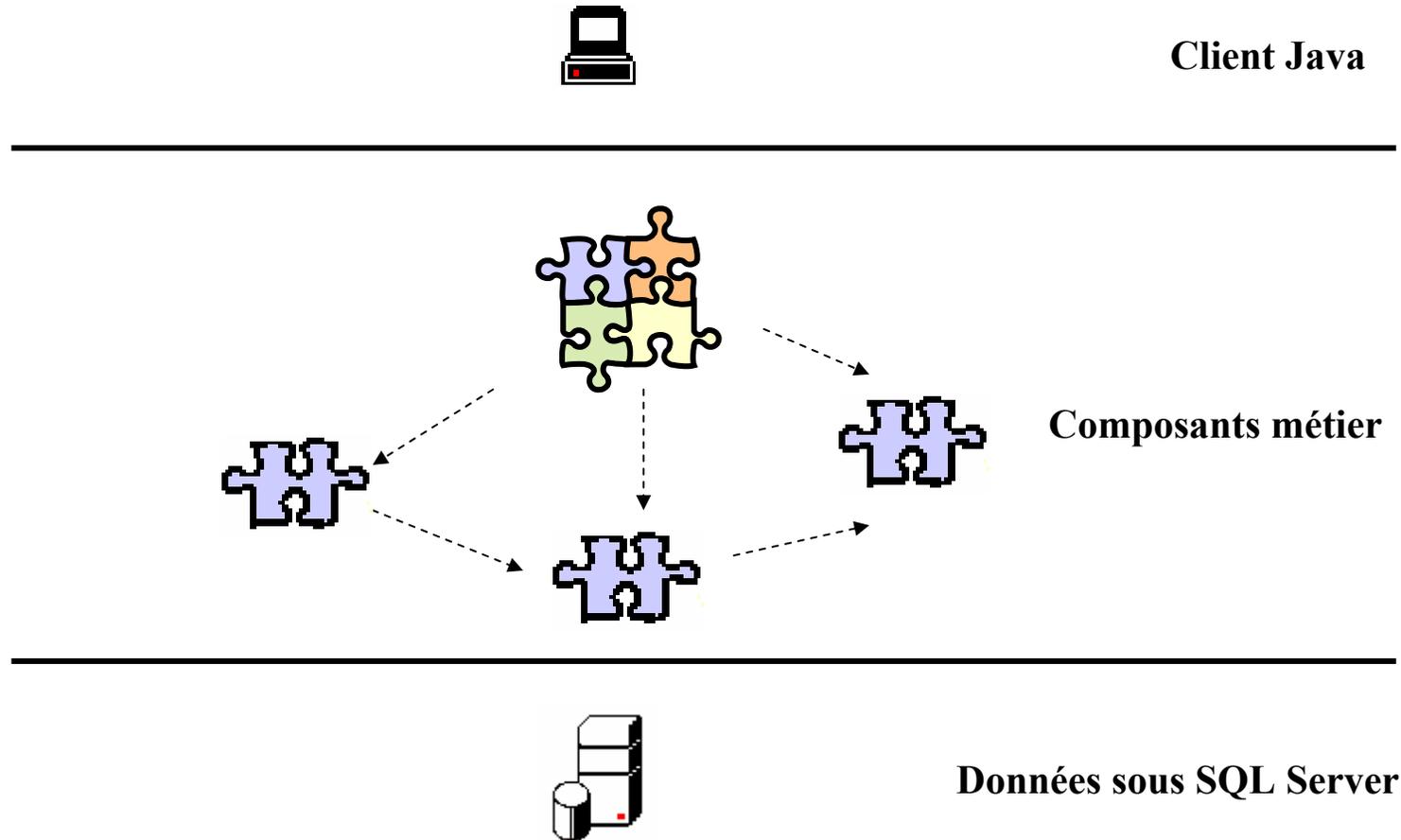


Définition de l'architecture logicielle

(application répartie en CORBA)

Introduction

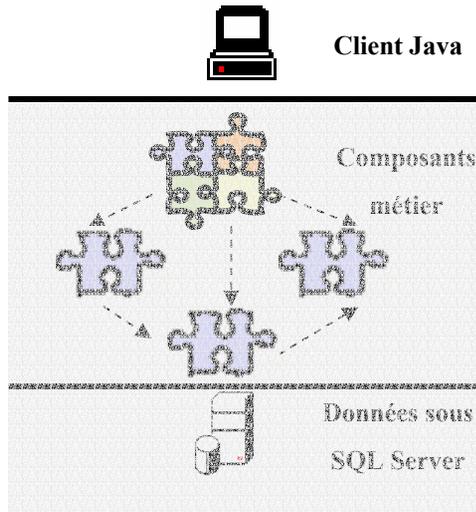
- L'architecture logicielle de l'application est basée sur le modèle 3-tiers



Client (1/2)

□ Le partie cliente de l'application correspond à la couche Présentation (selon le découpage du Gartner Group)

- la communication avec la partie composants métier se fait grâce à un ORB sur le poste client
- l'interface est développée en Java (bibliothèque Swing)

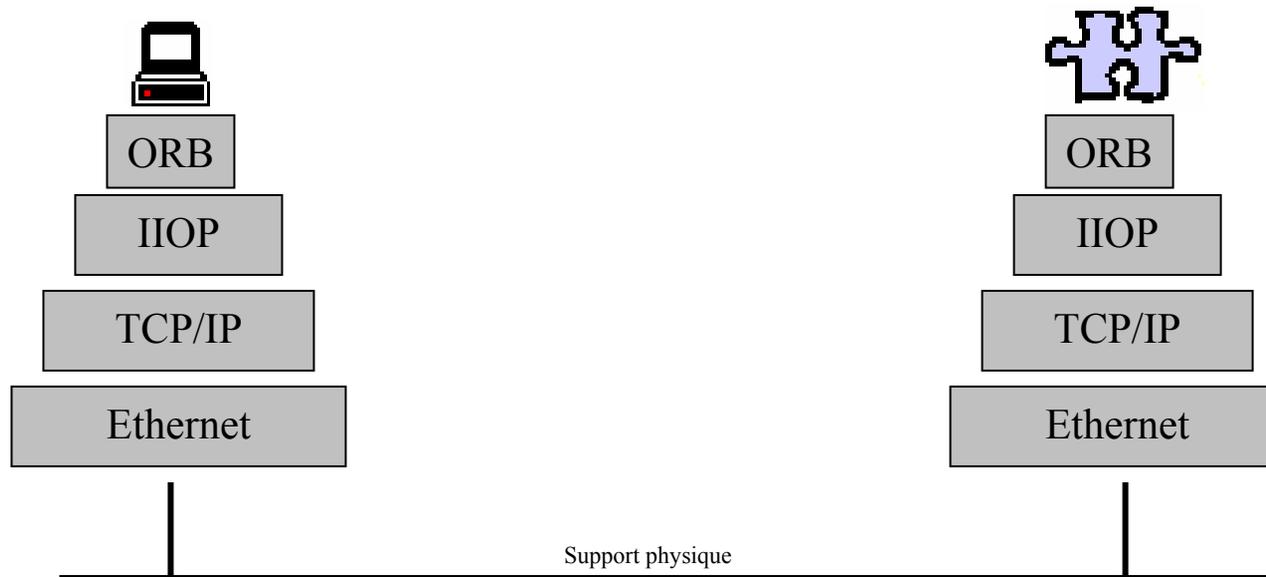


□ Les éléments logiciels suivants doivent être installés sur le poste client

- Inprise VisiBroker 3.4
- JDK 1.3
- souches client des composants métier (stubs, ...)

Client (2/2)

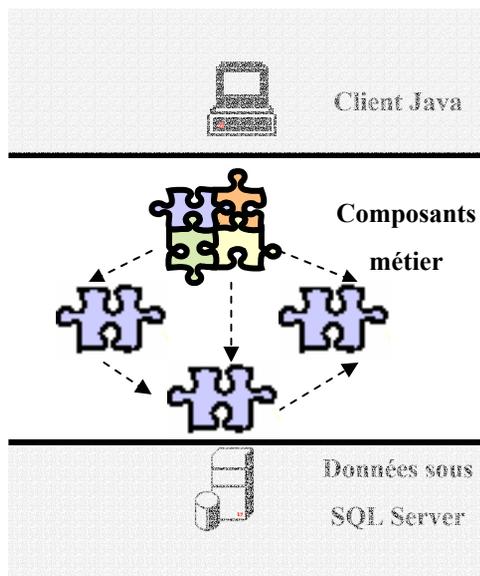
- Le client communique avec la partie métier selon le schéma en couches suivant



Composants métier (1/2)

□ La couche métier implémente la logique métier de l'application via des composants

- l'architecture utilisée est CORBA : plus précisément, on utilise VisiBroker 3.4 comme implémentation de CORBA
- les composants métier communiquent entre eux via le bus logiciel de CORBA
- le langage d'implémentation des objets est Java



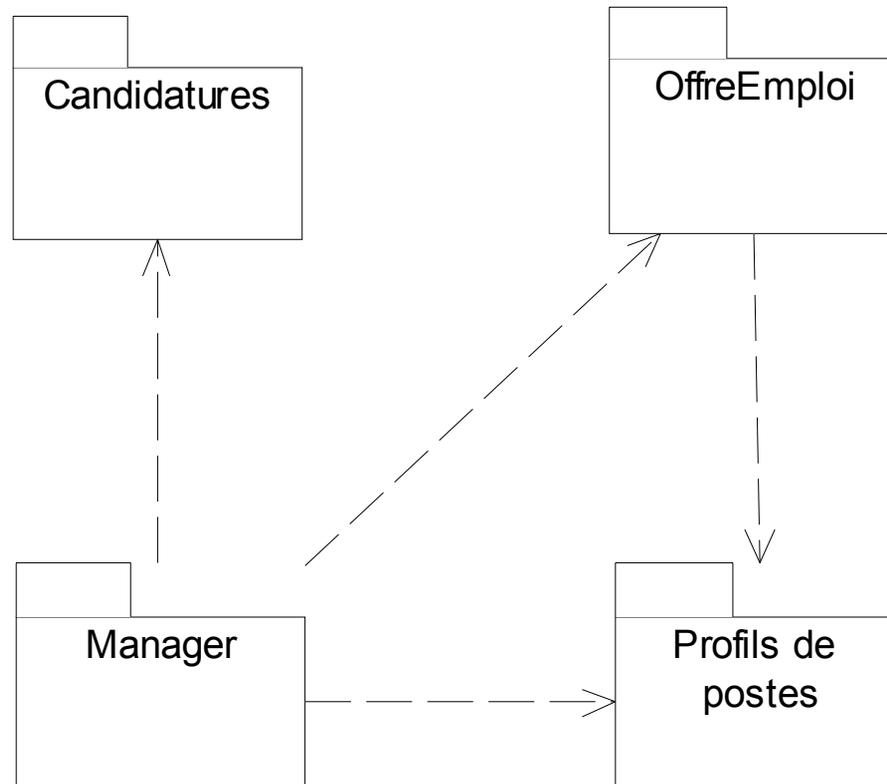
□ Les éléments logiciels suivants doivent être installés sur le serveur d'application

- Inprise VisiBroker 3.4
- JDK 1.3
- souches serveur des composants métier (skeletons, ...)
- drivers d'accès aux données (JDBC au dessus d'ODBC et ODBC pour SQL Server 7.0)

□ L'architecture de la couche métier est détaillée dans des documents annexes (voir **Références**)

Composants métier (2/2)

- ❑ La vue logique des composants métier se présente ainsi

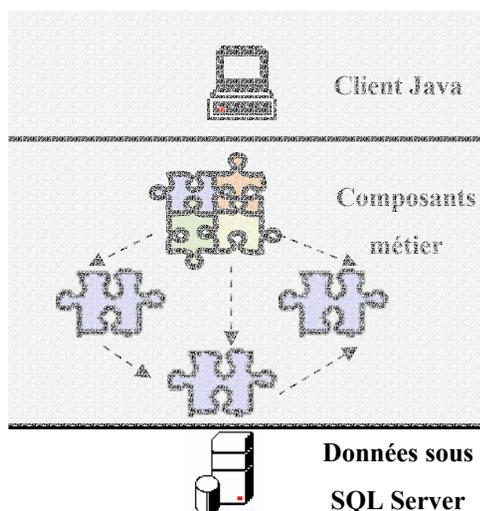


Base de données (1/2)

□ Les données sont gérées par le SGBDR *Microsoft SQL Server 7.0*

- la couche métier accède aux données par le biais d'un composant d'accès aux données

- le composant d'accès aux données, réalisé en Java, utilise l'API JDBC (Java DataBase Connectivity)



□ JDBC est un middleware d'accès aux données écrit en Java, composé de deux parties, l'API JDBC et l'API JDBC Driver

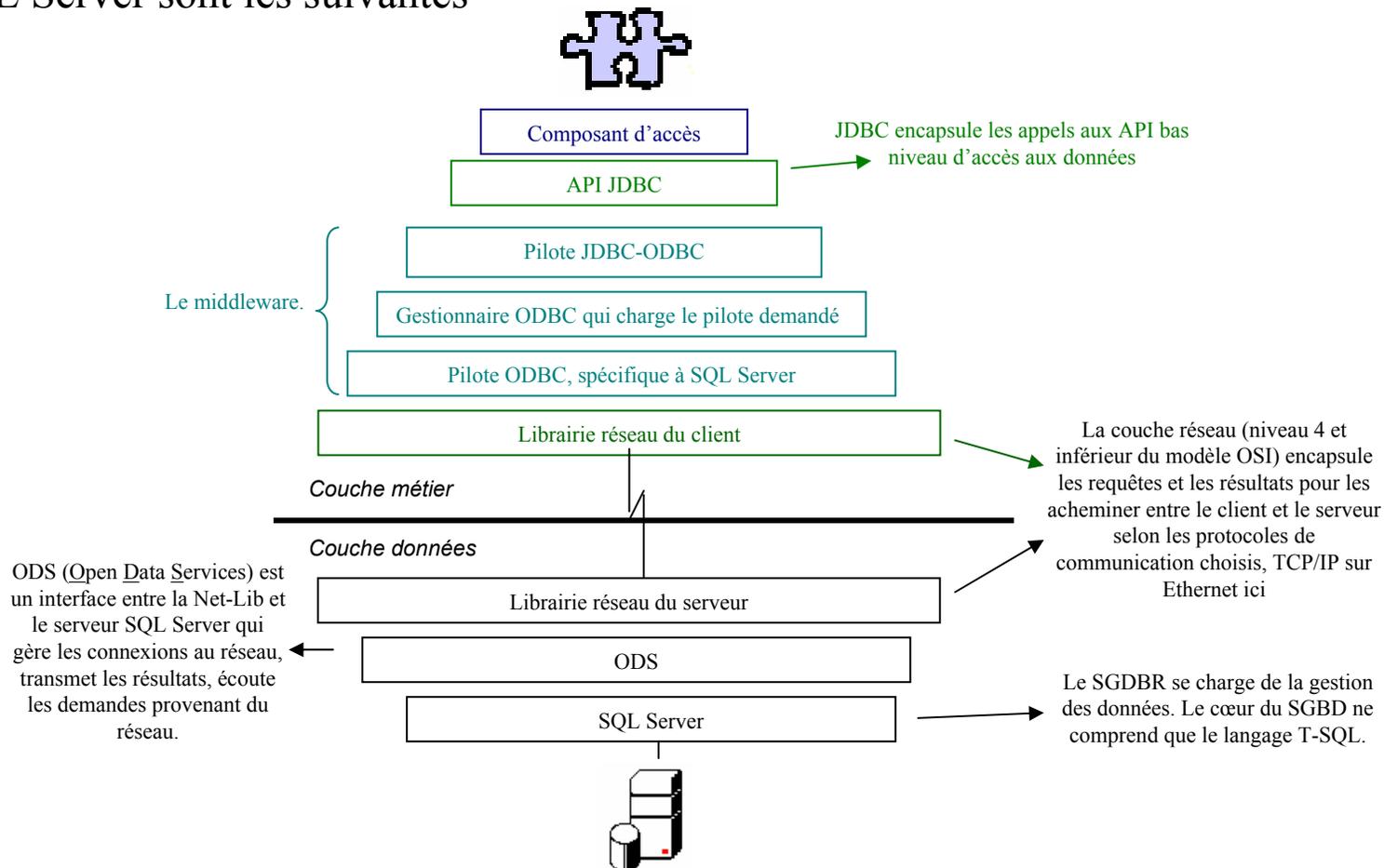
- l'API JDBC est destinée aux développeurs désirant accéder à un SGBDR : les structures de cette API se trouvent dans le package `java.sql`

- l'API JDBC Driver est l'abstraction d'un pilote JDBC. Il faut disposer de l'implémentation des interfaces de `java.sql.Driver` afin d'accéder à un SGBR particulier

□ Javasoftware fournit une implémentation de JDBC s'appuyant sur ODBC, identifiée comme étant le pilote JDBC-ODBC

Base de données (2/2)

- Les différentes couches utilisées entre le composant d'accès aux données et SQL Server sont les suivantes



Références

□ Les documents liés à la définition de l'architecture sont les suivants

- `archi_generale.ppt` : ce document
- `overview_corba.ppt` : description de l'architecture CORBA
- `etapes_corba.ppt` : cycle de développement d'une application CORBA
- `jbuilder_corba.ppt` : de l'IDL à la compilation avec JBuilder 4.0
- `regle_passage.ppt` : règles de passage du modèle à l'implémentation Java en passant par l'IDL