



*Les nouvelles technologies*





**Architectures C/S Java :**  
**de JDBC à CORBA**



**Sommaire**

- Le langage Java
- Les APIs Java
- Client/Serveur
- Architectures 2-Tiers
- Architectures 3-Tiers



**Caractéristiques**

- Objet
- Interprété (Portable)
- Modulaire (\*.class, JavaBeans)
- Distribué
  - Modèle de distribution de composants logiciels (applets)
  - Intégration simple aux architectures à objets distribués



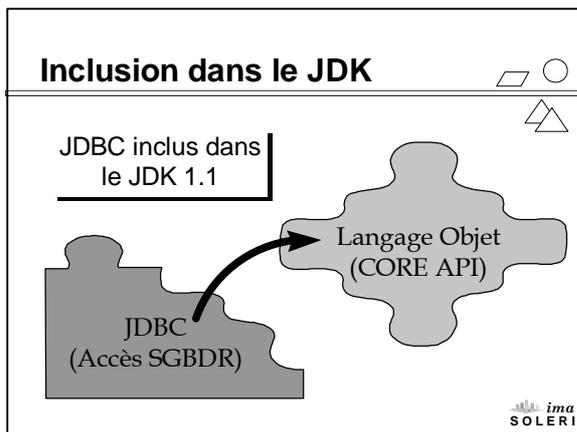
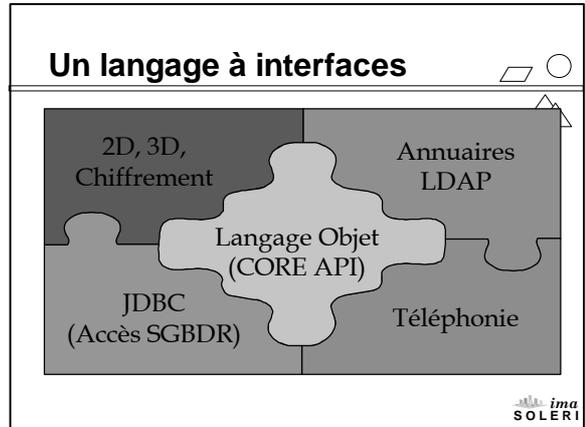
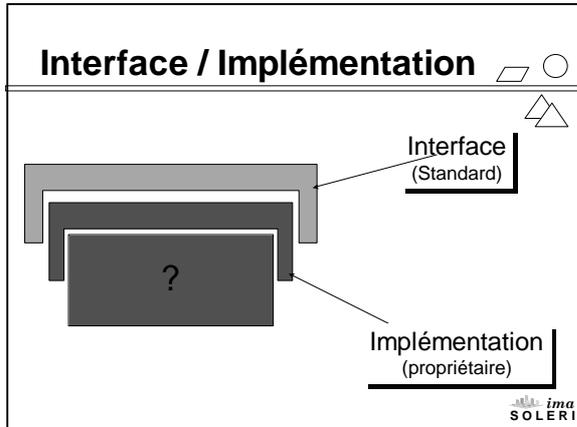
**Java Everywhere**



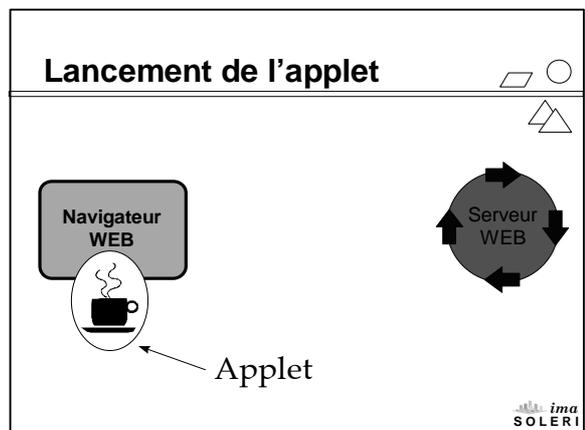
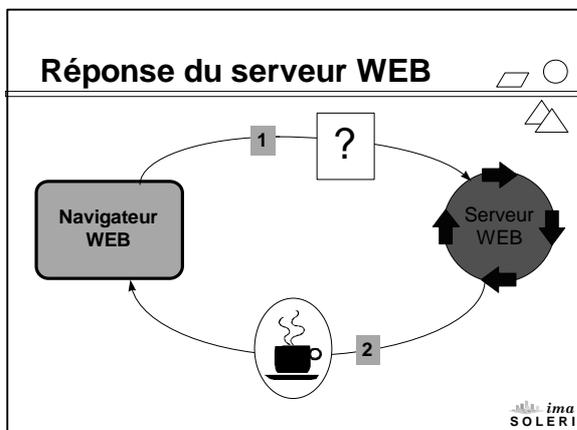
**Sommaire**

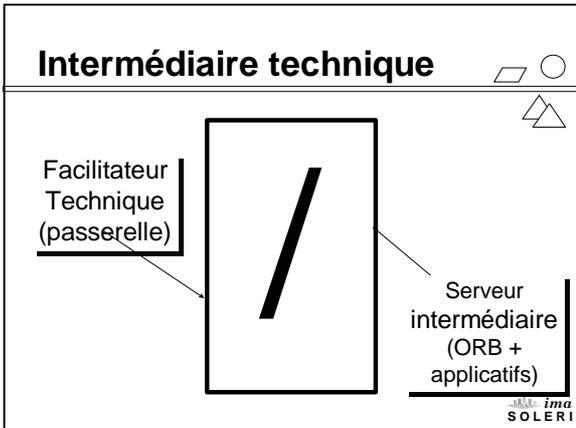
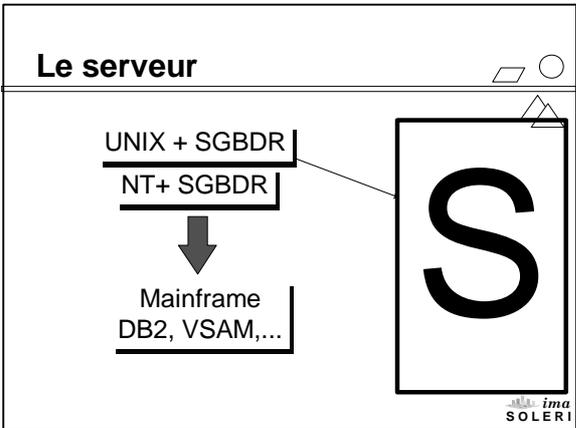
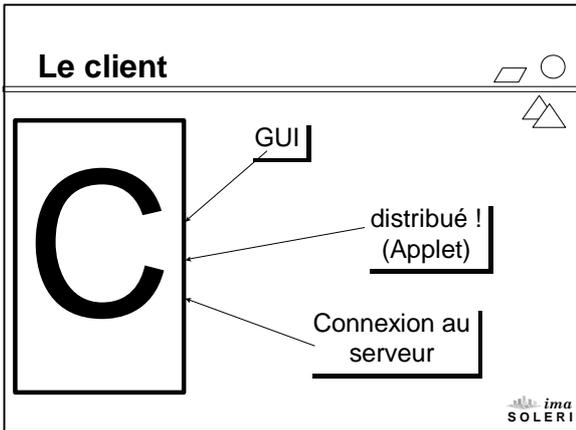
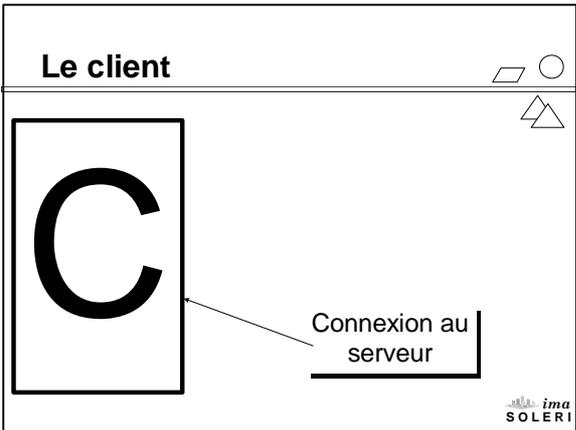
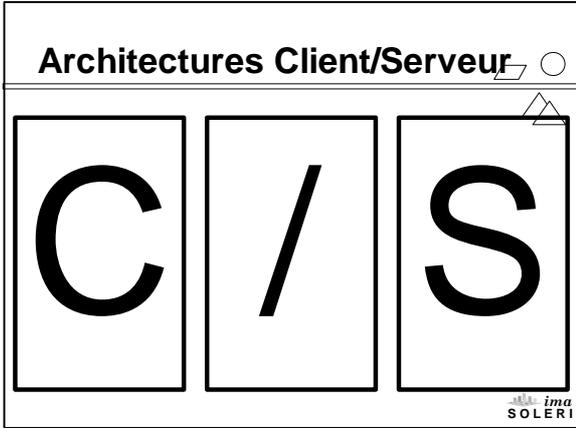
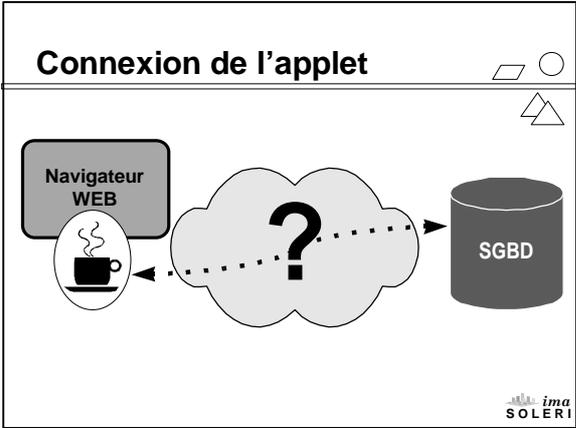
- Le langage Java
- Les APIs Java
- Client/Serveur
- Architectures 2-Tiers
- Architectures 3-Tiers





- ### Sommaire
- Le langage Java
  - Les APIs Java
  - **Client/Serveur**
  - Architectures 2-Tiers
  - Architectures 3-Tiers
- The SOLERI logo is in the bottom right corner.





**Sommaire**

- Le langage Java
- Les APIs Java
- Client/Serveur
- **Architectures 2-Tiers**
- Architectures 3-Tiers

ima SOLERI

**Architectures Client/Serveur**

C

/

S

ima SOLERI

**Architectures Client/Serveur**

C

/S

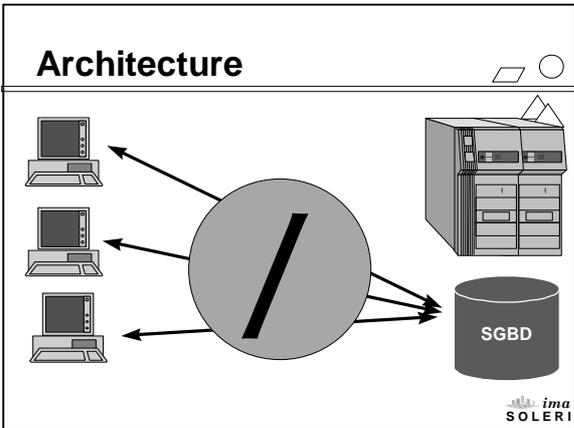
ima SOLERI

**Architectures Client/Serveur**

C/

S

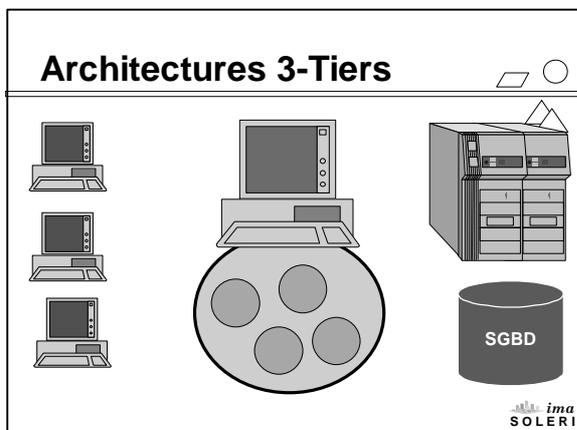
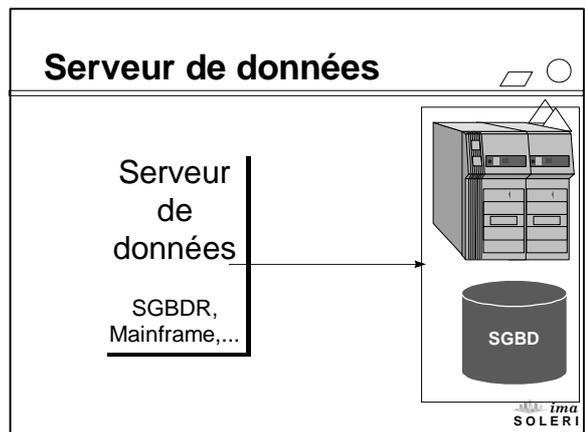
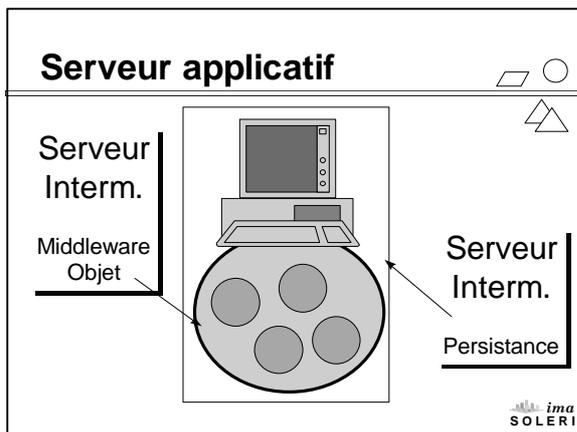
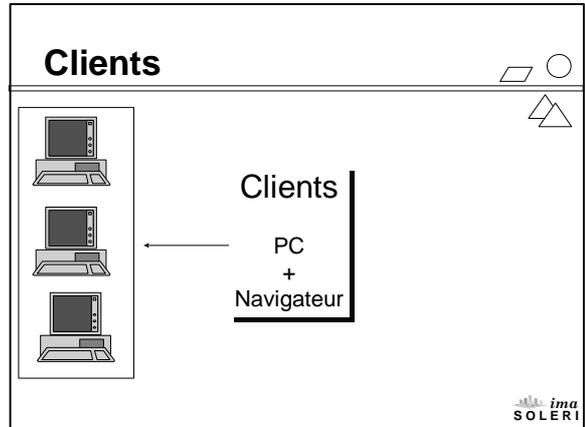
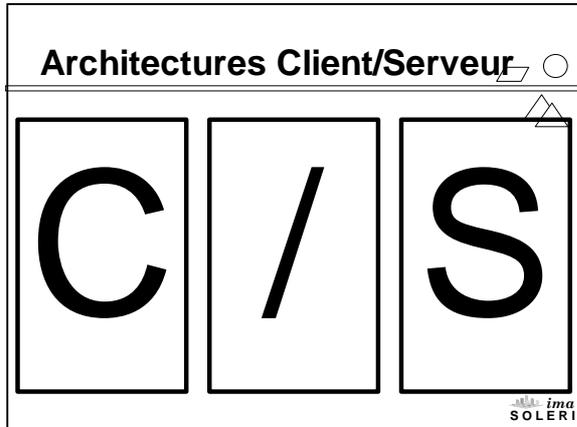
ima SOLERI



**Sommaire**

- Le langage Java
- Les APIs Java
- Client/Serveur
- Architectures 2-Tiers
- **Architectures 3-Tiers**

ima SOLERI



## Sommaire

- L'API JDBC
- Principe de fonctionnement
- JDBC et sécurité Java
- Les drivers JDBC
- Accès aux données en C/S

ima  
SOLERI

## Mars 1996 : JDK 1.0 (Solaris, Windows 95 et NT)

- Une première livraison avec
  - un compilateur
  - une machine virtuelle d'exécution
- De nombreux AGL annoncés
- Les principaux navigateurs internet intègrent Java
- Spécifications initiales de JDBC

ima  
SOLERI

## API JDBC

- JDBC est une API pour serveur SQL
- Basé sur le standard X/Open SAG CLI (Call Level Interface)
- Comme toutes les API de serveurs SQL
- Comme ODBC...

ima  
SOLERI

## Que fait JDBC ?

- JDBC permet :
  - la connexion au SGBD
  - la définition puis l'envoi d'ordres SQL au serveur
  - la récupération (et la lecture) des résultats.
- JDBC est une interface de bas niveau (de même niveau que CLI)

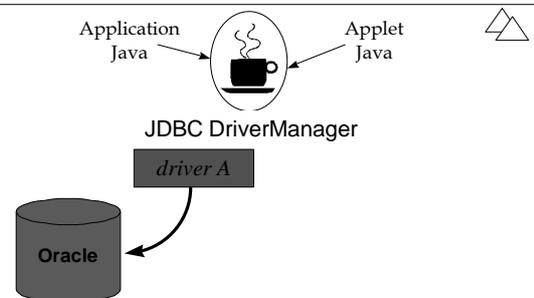
ima  
SOLERI

## Sommaire

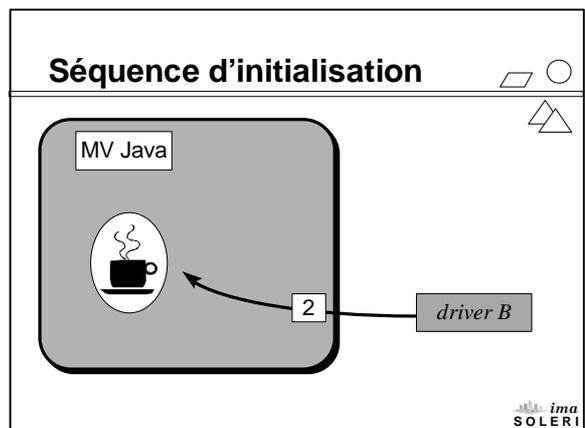
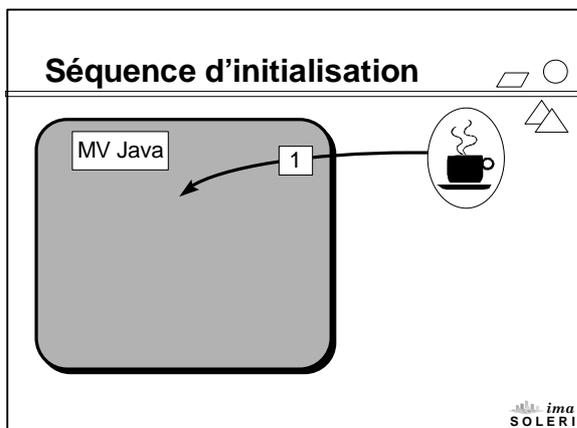
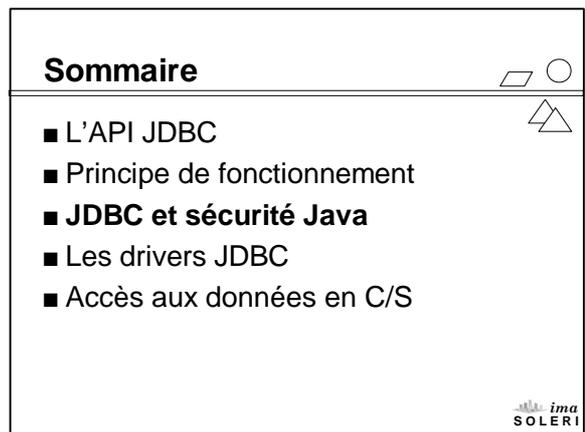
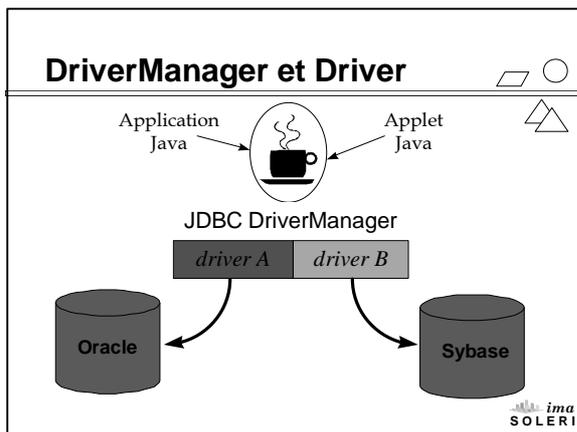
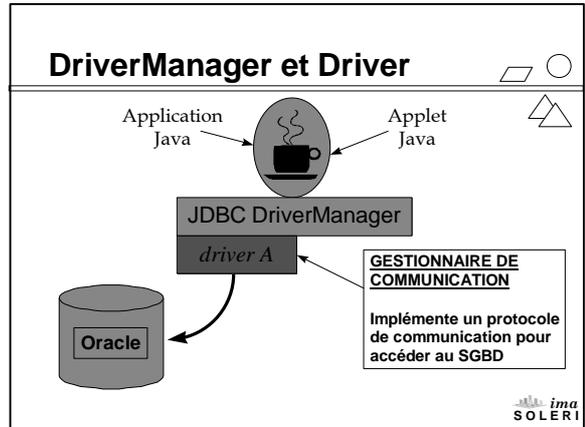
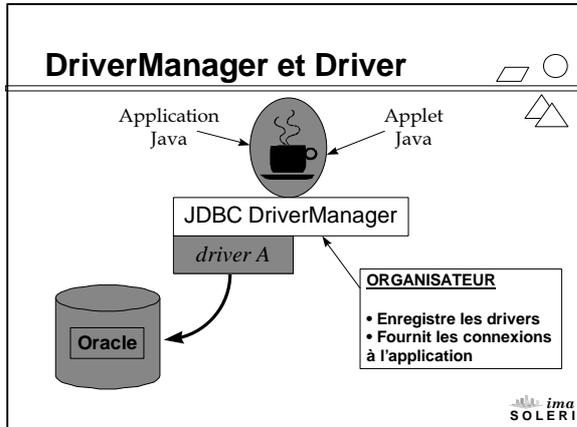
- L'API JDBC
- Principe de fonctionnement
- JDBC et sécurité Java
- Les drivers JDBC
- Accès aux données en C/S

ima  
SOLERI

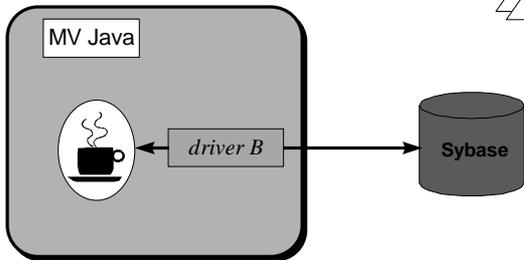
## JDBC, DriverManager et Driver



ima  
SOLERI



## Séquence d'initialisation



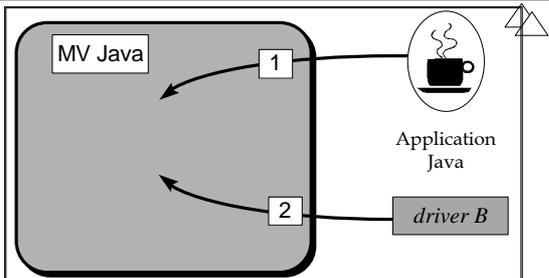
ima  
SOLERI

## Deux cas !

- Une seule machine
  - Driver et programme sur la même machine
- Deux machines
  - Communication Client/Serveur

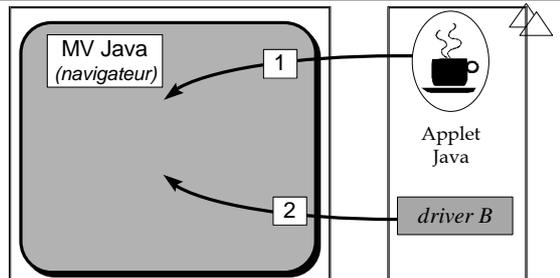
ima  
SOLERI

## Cas 1 : Une seule machine



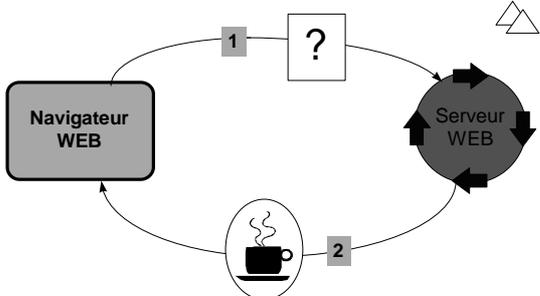
ima  
SOLERI

## Cas 2 : Deux machines (C/S)



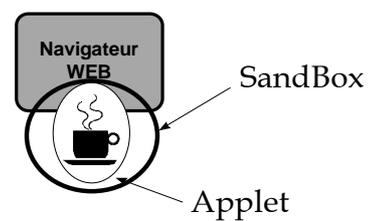
ima  
SOLERI

## Réponse du serveur WEB



ima  
SOLERI

## Lancement de l'applet



ima  
SOLERI

## Modèle SandBox

Du point de vue d'une machine virtuelle Java donnée, toute classe Java chargée au travers du réseau devra être contrôlée dans le cadre du modèle SandBox !

ima  
SOLERI

## Contraintes de sécurité

Une applet ne doit pas :



- ⊗ charger de code natif (dépendant du système)
- ⊗ accéder aux ressources du poste client (disque dur, imprimantes,...)
- ⊗ se connecter à un hôte différent de celui duquel elle provient

ima  
SOLERI

## Relaxer la sécurité

- Depuis le JDK 1.1
- Signature électronique des applets
- Définition d'applets *trusted* et *untrusted*
- Certificats internationaux sur Internet
- Mise en place de serveurs de certificats dans le cadre d'un Intranet

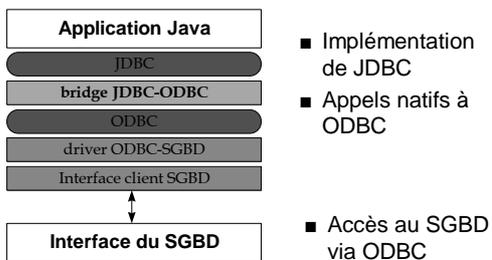
ima  
SOLERI

## Sommaire

- L'API JDBC
- Principe de fonctionnement
- JDBC et sécurité Java
- **Les drivers JDBC**
- Accès aux données en C/S

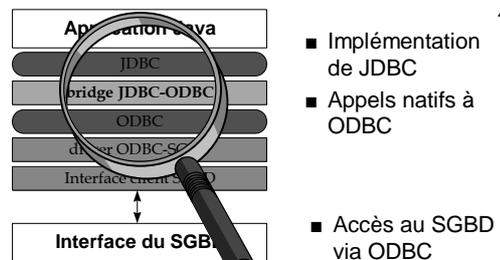
ima  
SOLERI

## Type I : pont JDBC-ODBC



ima  
SOLERI

## Type I : pont JDBC-ODBC



ima  
SOLERI

**Type I : pont JDBC-ODBC**

ima SOLERI

**Type I : pont JDBC-ODBC**

- Applet Java exécutée dans un navigateur Web

ima SOLERI

**Type I : pont JDBC-ODBC**

- Applet Java exécutée dans un navigateur Web
- Interdiction de charger du code Natif (modèle SandBox)

ima SOLERI

**Type I : pont JDBC-ODBC**

- Bridge JDBC-ODBC d'Intersolv livré en standard avec le JDK 1.1
- Offre logicielle ODBC étendue
- Interdit aux applets Java
- ODBC = Microsoft = Windows
- «ODBC is slow» !

ima SOLERI

**Type II : pont JDBC-API SGBD**

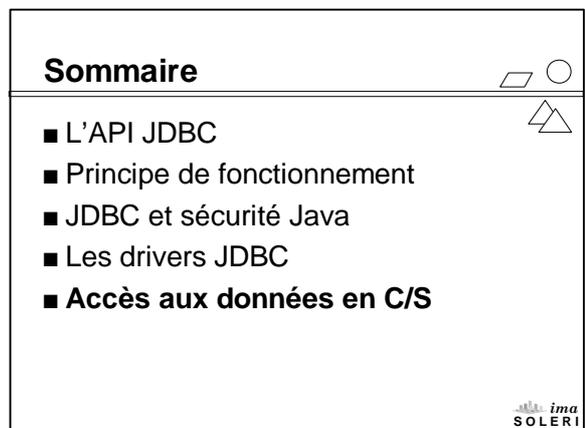
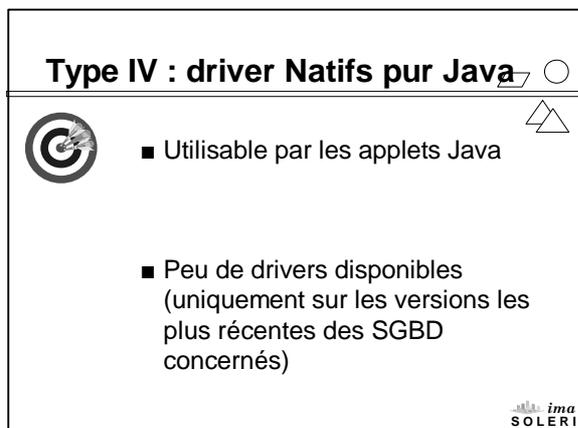
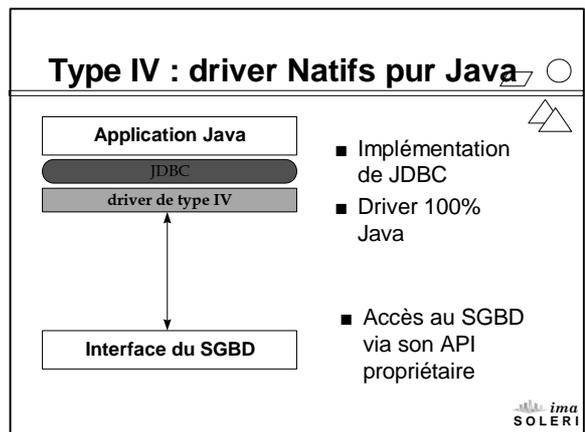
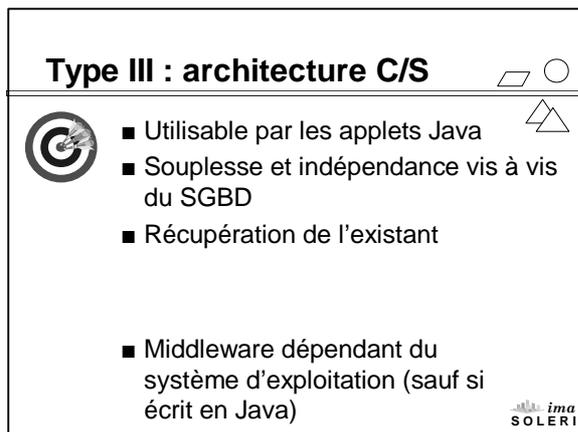
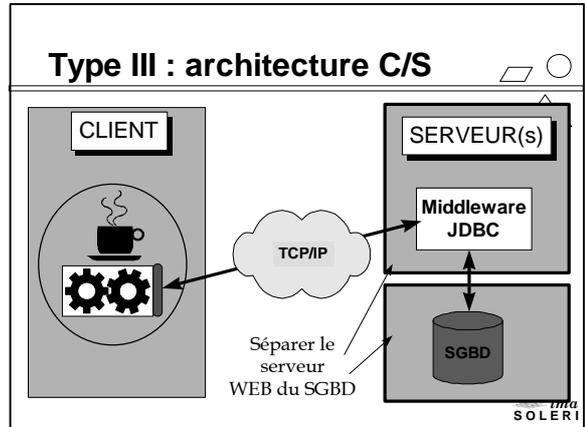
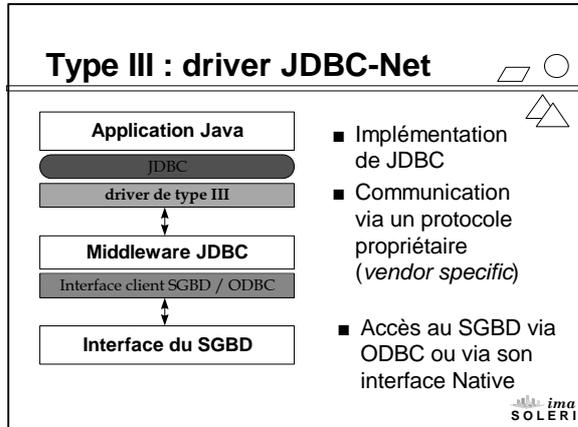
- Implémentation de JDBC
- Appels natifs à un driver natif
- Accès au SGBD via son API propriétaire

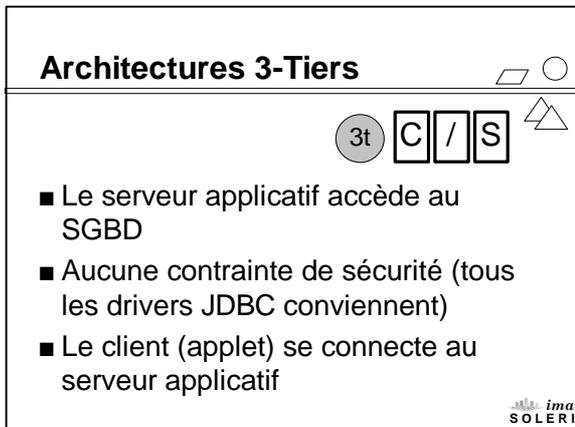
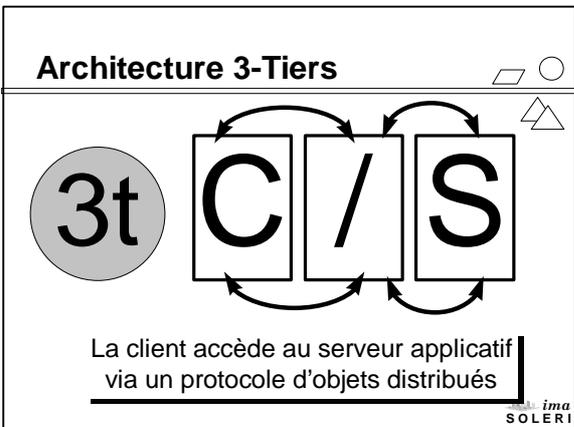
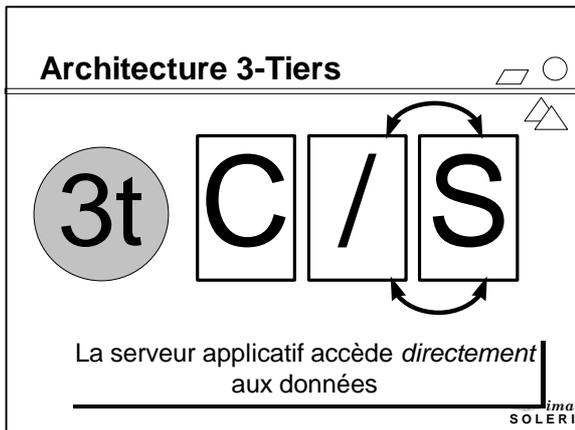
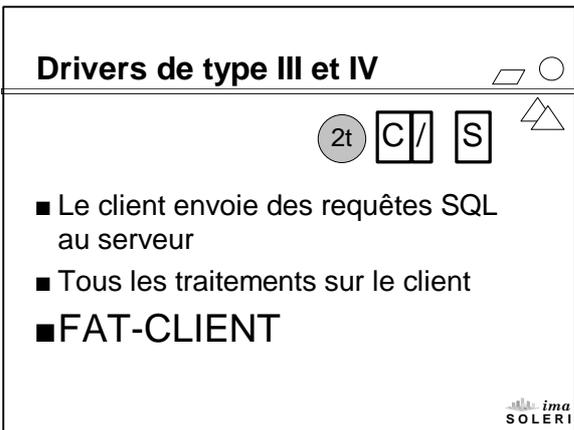
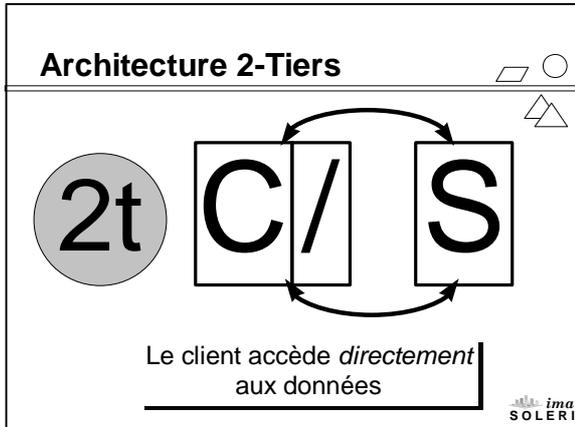
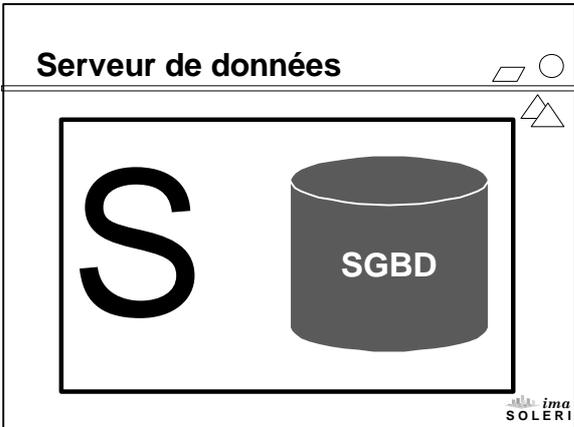
ima SOLERI

**Type II : pont JDBC-API SGBD**

- Meilleures performances qu'avec le bridge JDBC-ODBC
- Interdit aux applets Java
- Dépendant du système d'exploitation

ima SOLERI







## Objets distribués : CORBA, RMI, DCOM et Java



## Sommaire



- 3 systèmes
- Objets distribués ?
- Invocation à distance
- l'ORB
- RMI
- DCOM
- CORBA

## 1 langage = 3 systèmes



- 3 solutions :
  - RMI (*Remote Method Invocation*)
  - DCOM (*Distributed Component Object Model*)
  - CORBA (*Common Object Request Broker Architecture*)
- 3 lobbies (SUN, OMG et Microsoft)

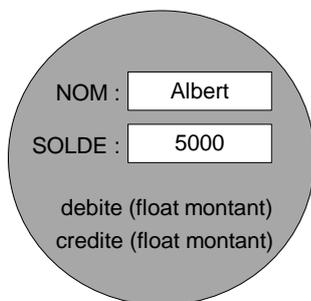


## Sommaire

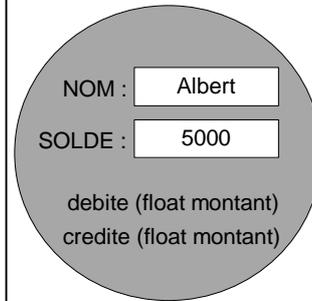


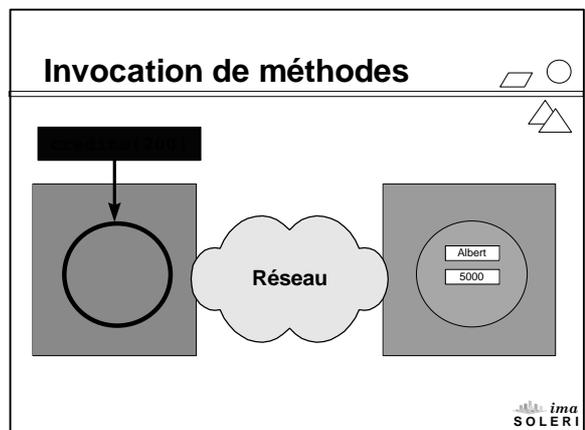
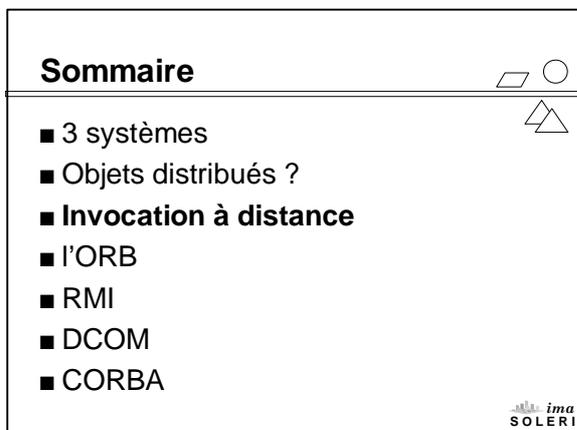
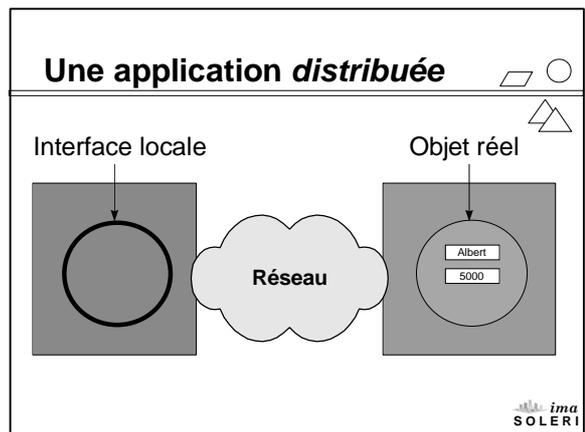
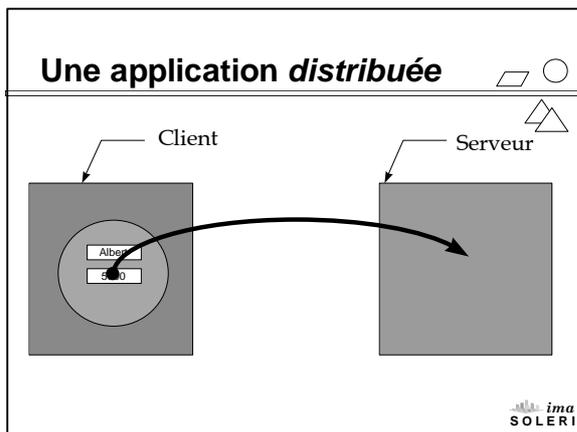
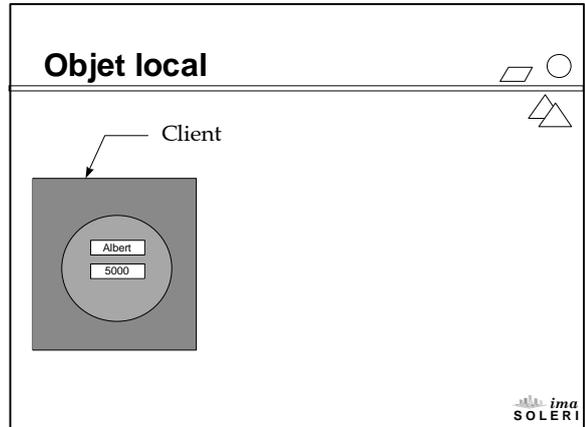
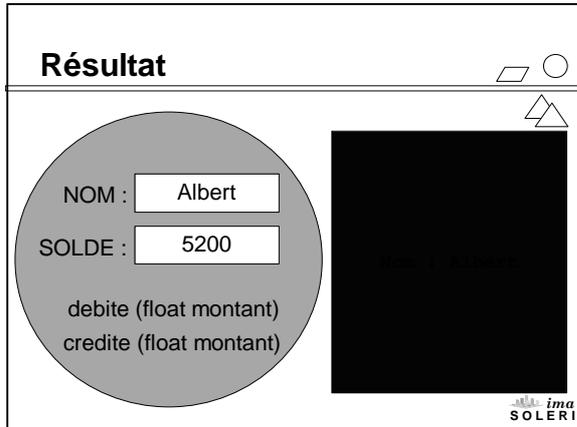
- 3 systèmes
- **Objets distribués ?**
- Invocation à distance
- l'ORB
- RMI
- DCOM
- CORBA

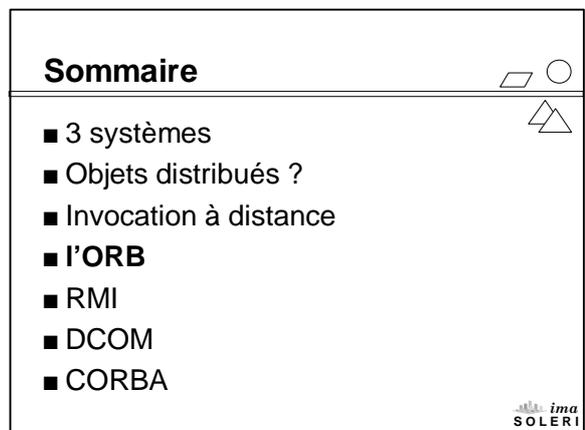
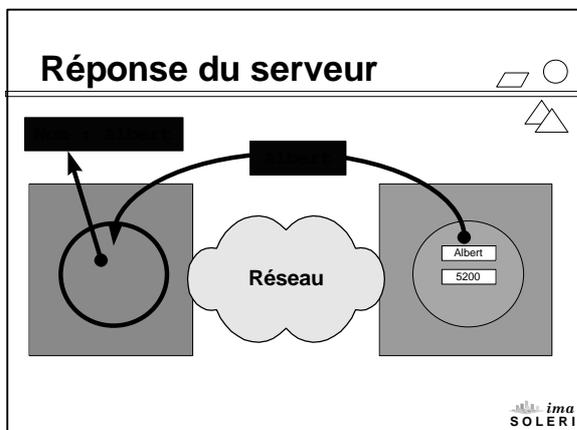
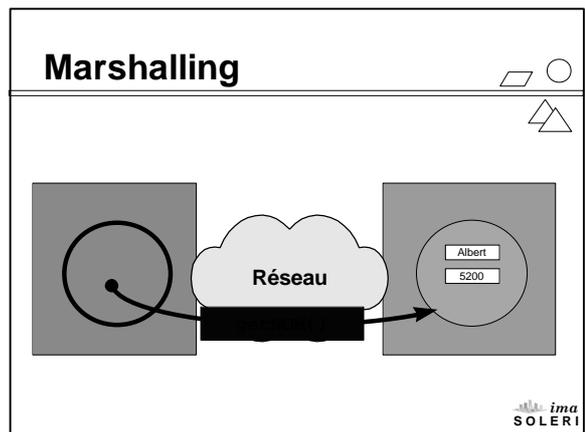
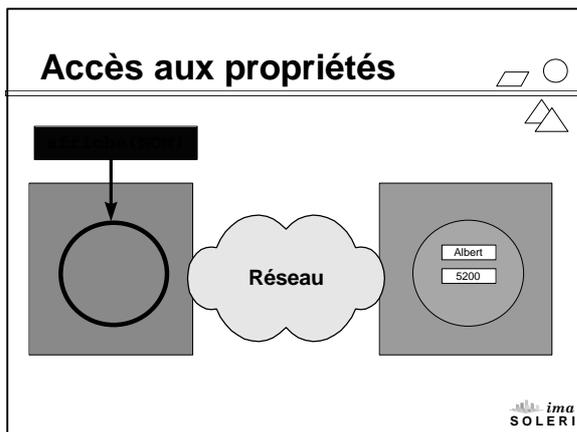
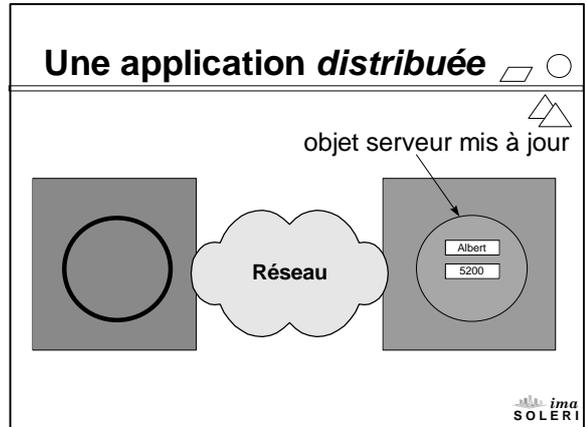
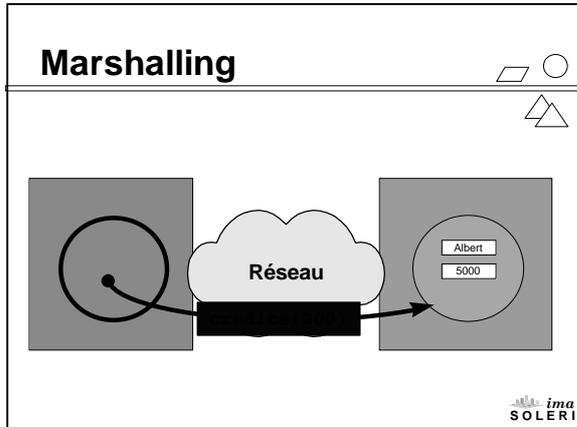
## Une instance de l'objet

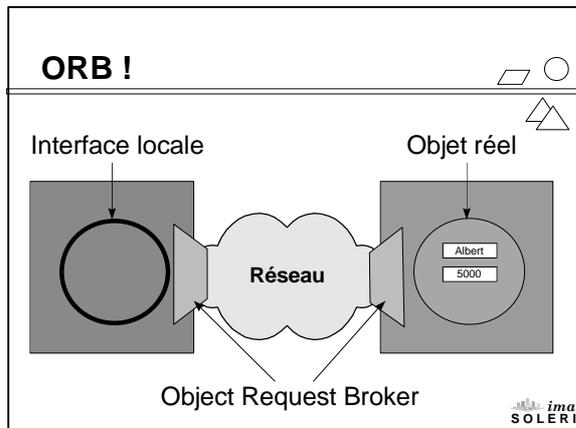


## Manipulation de l'objet









- ### Un middleware : l'ORB
- ORB : *Object Request Broker*
  - Le courtier en objet
  - Reçoit les requêtes
  - Renvoie des références d'objets
  - Gère la communication entre client et serveur
- ima  
SOLERI

- ### ORB et plus encore
- Gère l'adressage des objets (localisation)
  - Gère la persistance
  - Gère la sécurité des communications
  - **ACTIVATEUR**
    - Instancie les objets à distance à la demande
- ima  
SOLERI

- ### Sommaire
- 3 systèmes
  - Objets distribués ?
  - Invocation à distance
  - l'ORB
  - **RMI**
  - **DCOM**
  - **CORBA**
- ima  
SOLERI

- ### Java to Java
- API du JDK 1.1.x
  - Basé sur la sérialisation
    - transformation d'un objet en une représentation binaire minimale
    - utilisée pour la distribution d'objets et la persistance
  - Simple à mettre en œuvre
  - Robuste (car simple)
  - Fonctionnalités limitées (-> JDK1.2)
- ima  
SOLERI

- ### ORB = MV Java
- L'ORB est inclus dans le JDK
  - Très bas niveau
  - Pas d'activateur
  - Un registre de nommage (associe une référence à une chaîne de caractère)
- ima  
SOLERI

## Protocole et implémentation

- Implémentation en Java uniquement
- Protocole = RMI Protocol
- Mode de transfert propriétaire (SUN)
- Mais IIOF pour bientôt !

 ima  
SOLERI

## Résumé

- Intégré au JDK 1.1
- Protocole de communication propriétaire au dessus de TCP (IIOF).
- Simple à mettre en œuvre
- Robuste et pauvre fonctionnellement
- En évolution...

 ima  
SOLERI

## Sommaire

- 3 systèmes
- Objets distribués ?
- Invocation à distance
- l'ORB
- RMI
- **DCOM**
- CORBA

 ima  
SOLERI

## DCOM et Windows

- ORB intégré à Windows 32bits depuis NT4.0 (et patch pour Windows 95)
- Ensemble de modules assez peu homogène
- Complexe à appréhender
- Peu d'outils de bas niveau

 ima  
SOLERI

## Windows to Windows

- Intéressant dans le cas d'un réseau 100% pur Windows
- Evolution vers d'autres systèmes
- Au cœur de la solution : COM (OLE, OCX, ActiveX)
- Microsoft Transaction Server

 ima  
SOLERI

## Résumé

- *Distributed Component Object Model*
- Géré par l'X/Open (????)
- Nombreux langages d'implémentation
  - C++ (Visual C++), Visual Basic
  - Produits L4G (PowerBuilder, Delphi,...)
- WINDOWS !

 ima  
SOLERI

**Sommaire**

- 3 systèmes
- Objets distribués ?
- Invocation à distance
- l'ORB
- RMI
- DCOM
- **CORBA**

ima SOLERI

**CORBA 2.0**

- Instance de normalisation concertée :  
OMG
- *Object Management Group*
- 850 Entreprises actrices dans le domaine des technologies Objet
- Norme v2 : CORBA 2.0
- Intégration de Java à la norme

ima SOLERI

**ORB**

- CORBA
- Activateur (CORBA Daemon)
- Nombreux services (pas tous implémentés)
  - Naming Service (Trading)
  - Event Service
  - Transaction Service
  - Security

ima SOLERI

**IIOp**

- Internet Inter-ORB Protocol
- Permet de faire dialoguer des ORB de fournisseurs différents
- Construit au dessus de TCP/IP
- Implémenté par la plupart des ORB du marché

ima SOLERI

**Résumé**

- *Common Object Request Broker Architecture*
- Standardisé par l'OMG
- Indépendant du langage
  - C, C++, Smalltalk, ADA95, Java
- Produits riches
- Bien adapté à Java
- IIOp

ima SOLERI

**1997, deux publications *ima***

ima SOLERI

