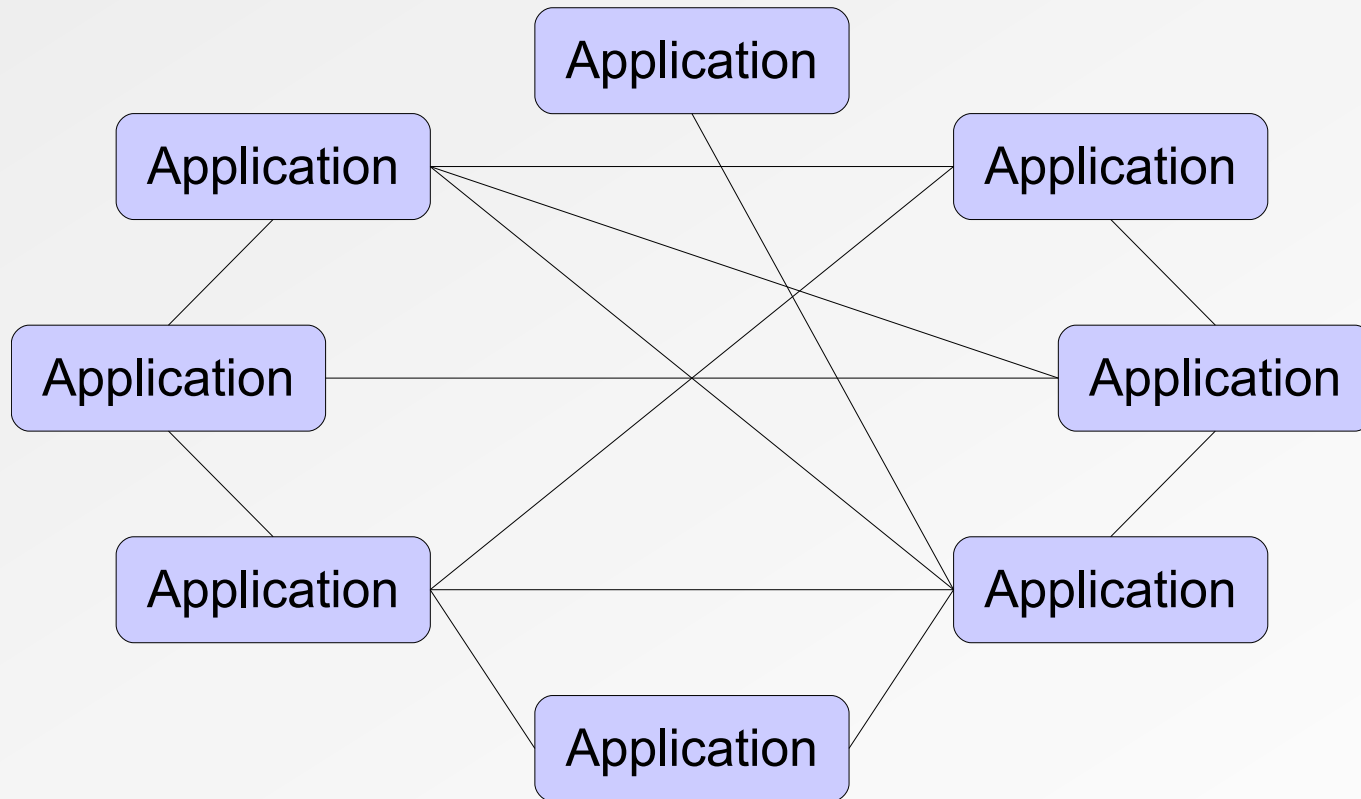
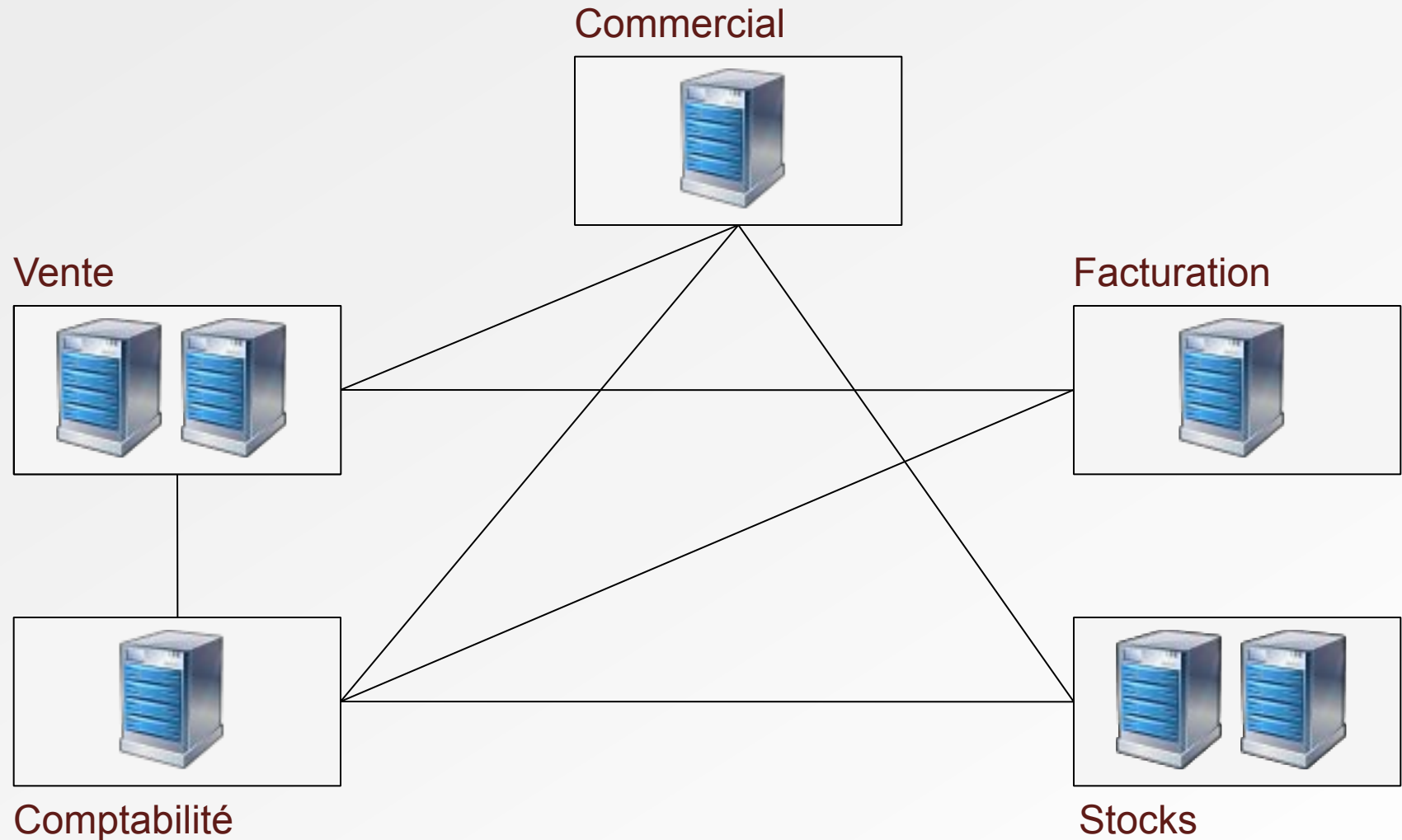


Les problématiques d'intégration

- L'architecture des systèmes d'information
- Un monde complexe et hétérogène
- Synchronisme et asynchronisme
- Le couplage faible / couplage fort



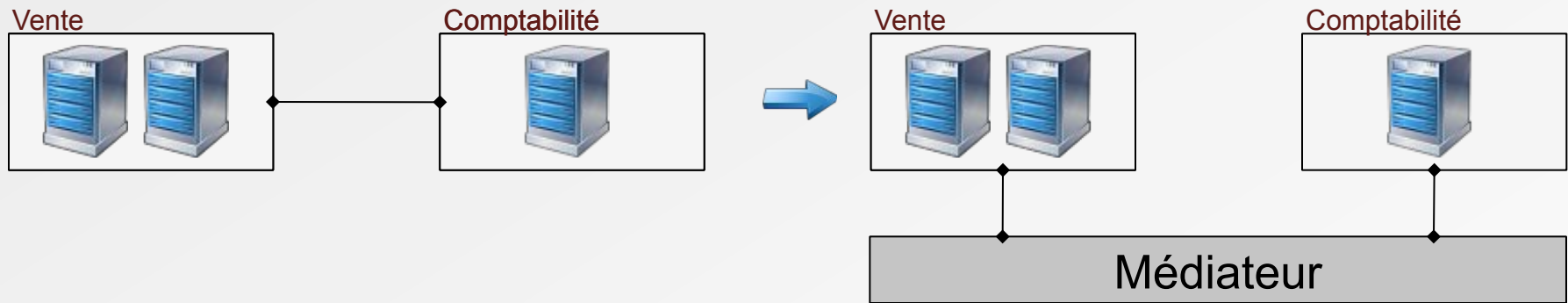
La problématique d'intégration



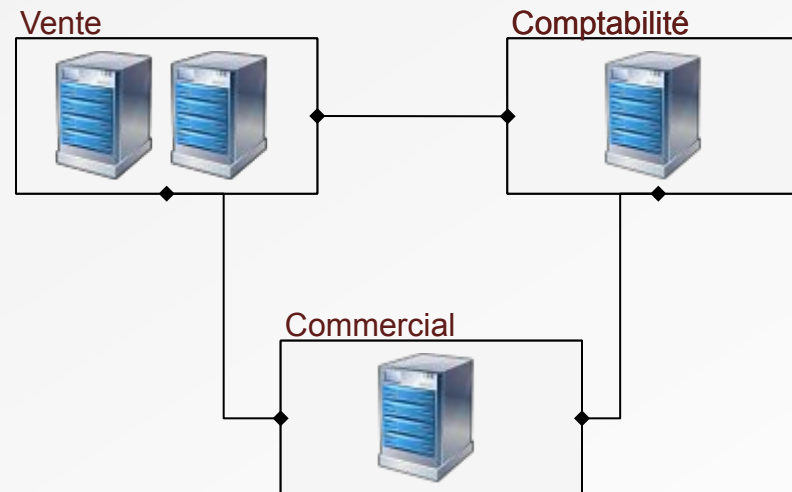
- Les réseaux ne sont pas fiables
 - Système distribué : contraintes fortes
- Les réseaux sont lents
 - Système distribué \approx Système asynchrone
- Les applications sont différentes à tous les niveaux
 - Système d'exploitation
 - Implémentation
 - Format de données

- Le changement est inévitable
 - Changement métier (stratégie, règlement...)
 - Changement technique
 - Changement humain
 - Changement de « mode » (exemple : EJB)

- Médiation – Réduction des adhérences entre applications



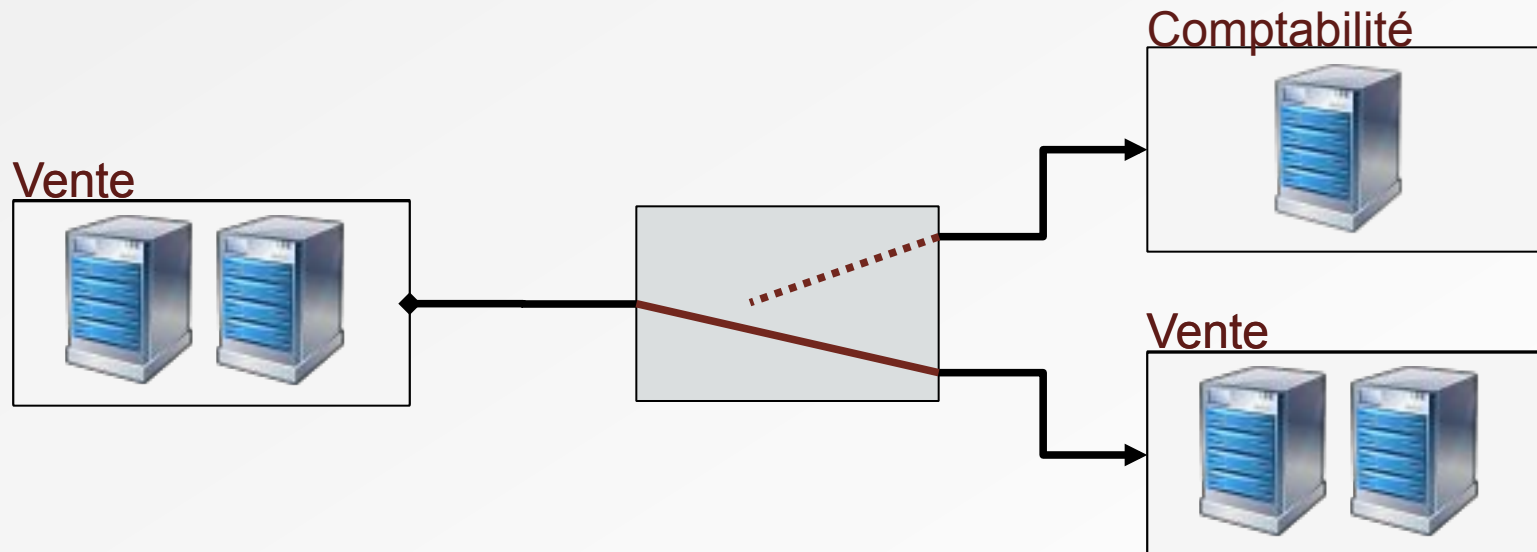
- Connectivité – support de multiples protocoles de transport



- Transformation – Support de différents types de données



- Routage – Routage des messages entre applications



- Exemples : FTP, CFT, Copie locale, etc.
- Avantages
 - Découplage des applications
 - Simple à mettre en place
 - Pas besoin d'outils d'intégration
- Inconvénients
 - Format des données
 - Temps de latence → biais de synchronisation
 - Gestion des fichiers complexes

- Exemples : Oracle, PostgreSQL, etc.
- Avantages
 - SQL → format des données unifié
 - Schéma partagé → pas de conflit sémantique
- Inconvénients
 - Couplage fort au schéma → forte sensibilité au changement
 - Progiciel : impossibilité de changement du schéma

- Exemples : CORBA, RMI, etc.
- Avantages
 - Risque de problème sémantique faible
 - Méthode « naturelle » : intégration facilitée
- Inconvénients
 - Méthode « naturelle » : contraintes comparée à un appel local (performances, robustesse)
 - Couplage applicatif fort

- Exemples : MOM, SMTP, Web Services, etc.
- Avantages
 - Découplage des applications
 - Robuste
 - Performant
- Inconvénients
 - Légère latence
 - Adapté aux « petits » paquets de données

- Liaisons entre applications
 - Directes
 - Non standardisées

Avantages

- Facile et rapide à mettre en place (⚠ dans un premier temps)
- Ne nécessite pas d'installer une solution tierce

Inconvénients

- Forte adhérence entre applications
- Complexité maintenance / supervision

- MOM – Message-Oriented Middleware
- JMS – API Java supportée par une grande majorité des éditeurs de MOM, devient un standard de fait
- Exemples : Apache ActiveMQ, IBM WebSphere MQ

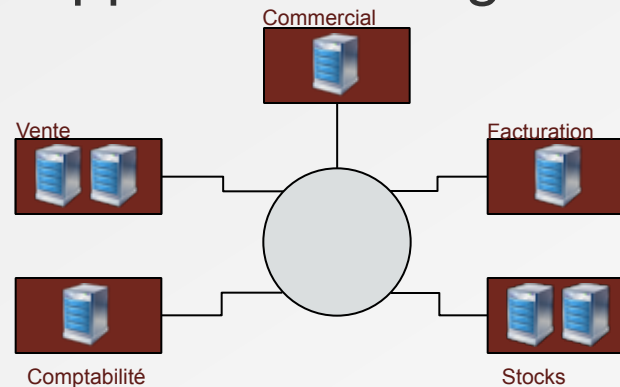
Avantages

- Solution simple

Inconvénients

- API hétérogènes
- Ne couvrent pas l'ensemble des problématiques d'intégration (exemple : connectivité, routage, etc.)

- EAI – Enterprise Application Integration



Avantages

- Solution complète d'intégration

Inconvénients

- Propriétaire
- Prix
- Complexe
- Centralisé → SPOF : « Single Point Of Failure »

- Besoin d'une solution d'intégration
 - Distribuée
 - S'appuyant sur les standards
- Opposition frontale à la vision EAI (centralisée et propriétaire)
- Technologie s'appuyant sur les bonnes pratiques d'intégration et sur l'écosystème existant
 - MOM
 - Patterns d'intégration : routage / validation / transformation
- L'ESB se compose de deux éléments principaux
 - Le bus de messages
 - Les conteneurs de services