**Acteurs :**

**Recensement des contraintes :**

## Contraintes exprimées :

* 1. **Besoins techniques**

**Acteurs :**

* *Courtiers :*
* *Clients individuels:*Connexion avec des PC.
* *Clients mobiles :* Clients légers avec connexion très rapide : accès direct aux cours.
* *“Gros clients” :* Contrainte critique ; délai d'exécution des transaction.
* *Clients traditionnels :* pas de contraintes techniques, leurs transactions sont assurées a travers les agences régionales.
* *Les courtiers de la banque* vont assurer les transactions des clients traditionnels.
* *Agents “à la corbeille” :* accès direct a la centrale.
  1. **Besoins métier**
* Fournir des outils d’aide a la décision.
  1. **Contraintes structurantes**
     1. **Architecture centralisée répartie**

Architecture répartie dans les agences régionales et locales qui est centralisée pour l’aspect Business Intelligence dans l’agence centrale.

* Agence centrale (Paris) :
  + Reliée directement à la place de marché
  + Obtenir les actualités du marché (carnets d’ordre)
  + Doit recevoir tous les ordres des agences régionales
  + Passe les ordres sur le marché
  + Mettre a jour le carnet d’ordres
  + Stocker les historiques d’ordres
* Agence régionales :
  + Reliée à l’agence central
  + Transmets les ordres à cette dernière.
* Agence Locales :
  + \*\*\* Reliée à l’agence régionales la plus proche. (Définir les régions)
  + Transmets les ordres à cette dernière.
    1. **Qualité de service**

Le service de passage d’ordres doit être assuré à tout moment lors des heures d’ouverture de la place financière.

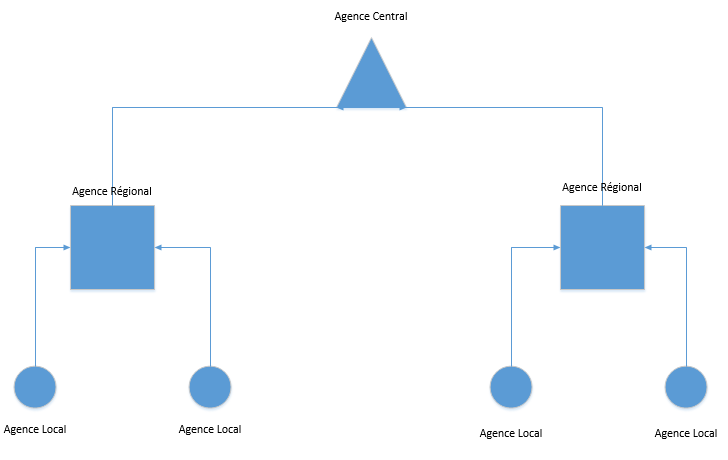
1. **Contraintes existantes :**
   1. **Contraintes organisationnelles**
   2. **Contraintes liées à l’environnement technique**

**Architecture applicative**

**www.ksvali.com/wp-content/uploads/2009/08/fixed\_income\_trading\_system\_architecture.pdf**

[**http://www.cisco.com/web/strategy/docs/finance/TradingFlrArch110707.pdf**](http://www.cisco.com/web/strategy/docs/finance/TradingFlrArch110707.pdf)

**1. Description fonctionnelle**



**3 blocs applicatifs :**

Architecture répartie dans les agences régionales et locales qui est centralisée pour l’aspect Business Intelligence dans l’agence centrale.

Description détaillée des blocs applicatifs

* Agence centrale (Paris) :

***Exigences fonctionnelles:***

* + Reliée directement à la place de marché
  + Obtenir les actualités du marché (carnets d’ordre)
  + Doit recevoir tous les ordres des agences régionales
  + Passe les ordres sur le marché
  + Mettre a jour le carnet d’ordres
  + Stocker les historiques d’ordres
  + **Le Carnet d'ordres**
  + Le Carnet d'ordres propose les 5 meilleures limites à l'achat et à la vente sur un titre ainsi que le nombre d'actions et d'ordres positionnés à chaque niveau. Dans un respect réglementaire, Boursorama diffuse sur son site visiteur/membre le carnet d'ordre décalé de 15 minutes.
  + Boursorama vous propose également un indicateur permettant d'identifier de façon plus facile la tendance du carnet d'ordre (achat ou vente) en affichant où se situe l'équilibre entre l'ensemble des titres à la vente et à l'achat. Ainsi, si la ligne verte de l'indicateur est plus longue que la ligne rouge, cela signifie que la quantité de titres à l'achat est supérieure à celle à la vente.
  + **Bid & Ask**
  + Le Bid and Ask (ou fourchette de cotation) représente la fourchette de prix achat-vente fournie en continu par les émetteurs. Les émetteurs partenaires fournissent un prix à l'achat et à la vente de façon continue et pendant les mêmes horaires d'ouverture que sur le marché réglementé de NYSE Euronext.

***Exigences techniques:***

Haute disponibilité, supervision

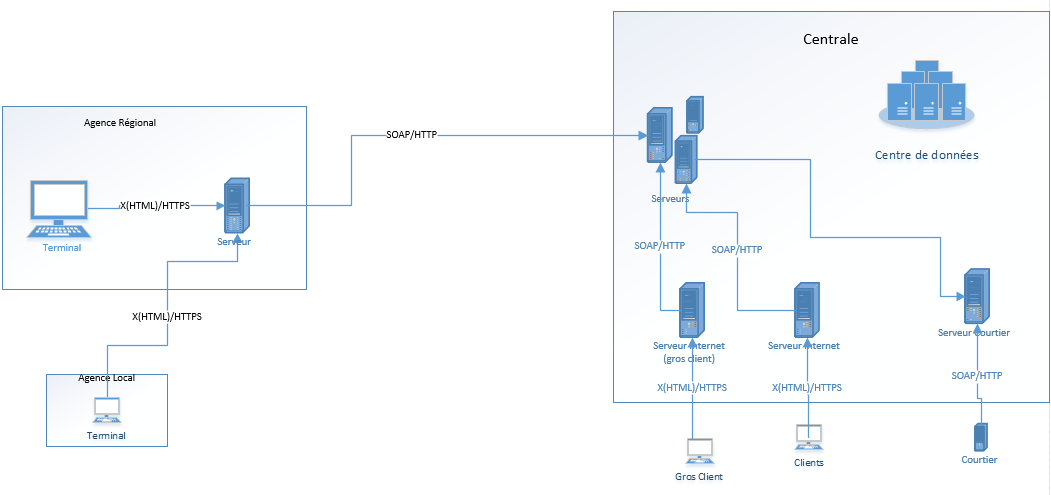
Développement spécifique.

# Matériel : IBM CICS Transaction Server

Système : Linux / Z

* Agence régionales :
  + Reliée à l’agence central
  + Transmets les ordres à cette dernière.
* Agence Locales :
  + \*\*\* Reliée à l’agence régionales la plus proche. (Définir les régions)
  + Transmets les ordres à cette dernière.

**2. Cartographie applicative des flux**



**3. Matrice des flux**

* **“**[Trading applications trade-offs : Databases and low-latency](http://stackoverflow.com/questions/7281391/trading-applications-trade-offs-databases-and-low-latency)
  + Caching: Cache the rules in your application, so you don't need to hit the database for every trade
  + Threading: Put the code that stores the transaction data in the database into another thread. Like this, you can route the trade to the exchange immediately and safe the data in the database parallel to that.”

**4. Cinématique représnetative**

**Architecture logicielle**

**1. Modèle d’architecture**

Définir pour chaque blocs applicatifs

**2. Motifs de conception**

**3. Outils utilisés : Librairies, composants, frameworks**

**Architecture technique**

Ce que Philippe a raconté à la fin du cours :

Courtier : La banque fournit aux courtiers un système clef-en-mains à ses clients (d’autres banques) qu’ils pourront offrir à leur propre clientèle. 2 solutions extrêmes soit on mutualise notre architecture soit on propose à chaque courtier un architecture séparée. Bah il faut justement trouver un compromis non? c’est ce qu’il disait non? oui. Ah d’accord.

Agent à la corbeille : Salariés de notre banque qui place de l’argent qu’on lui confie selon une stratégie contractée par un client friqué. Il doit être averti de tout changement du carnet d’ordre et doit pouvoir réagir le plus rapidement possible.

Clients traditionnels (mamie) : Envoie un recommandé avec accusé de réception pour qu’on opère pour eux selon leurs demandes