



SOA :
une brique de
la 4^{ème} génération de l'architecture informatique
?

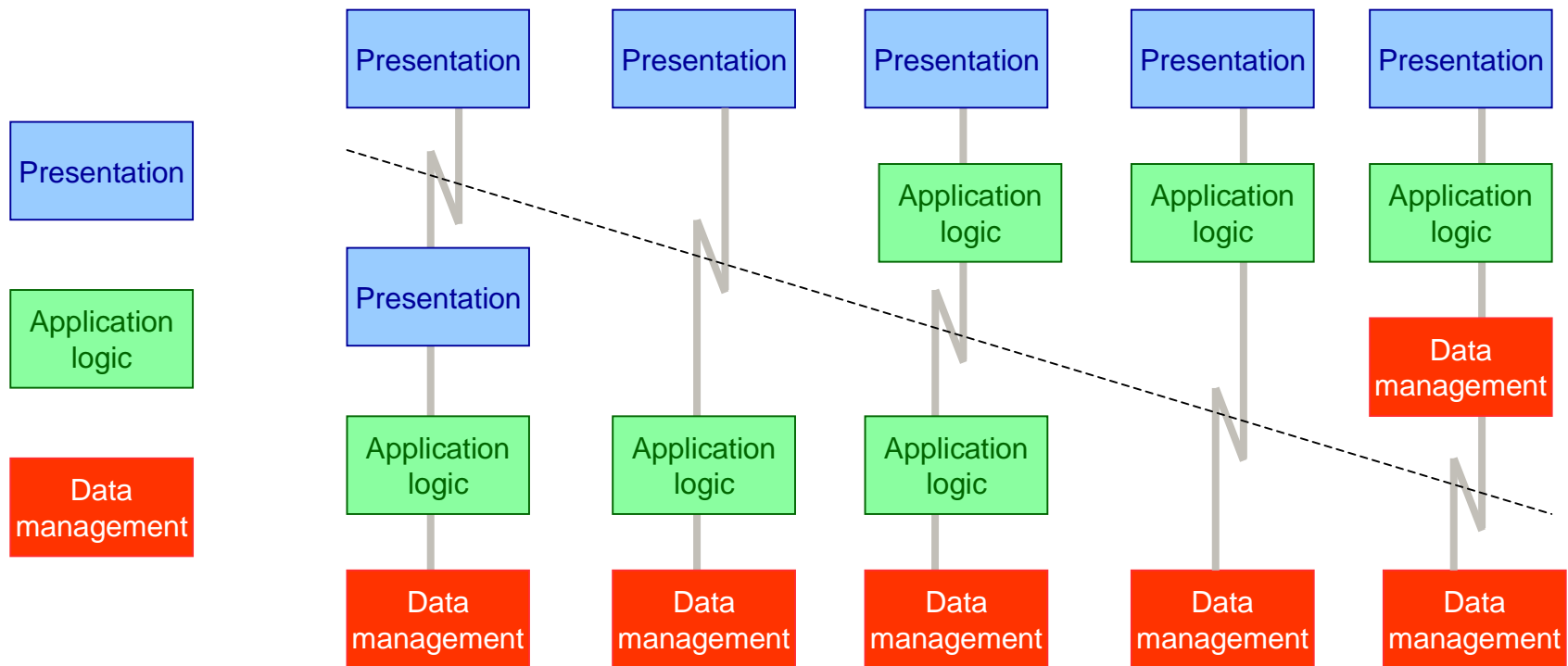


Hervé Crespel
Président du club urba-EA



Gartner 1992 : styles of client-server computing

L'origine du SOA ?

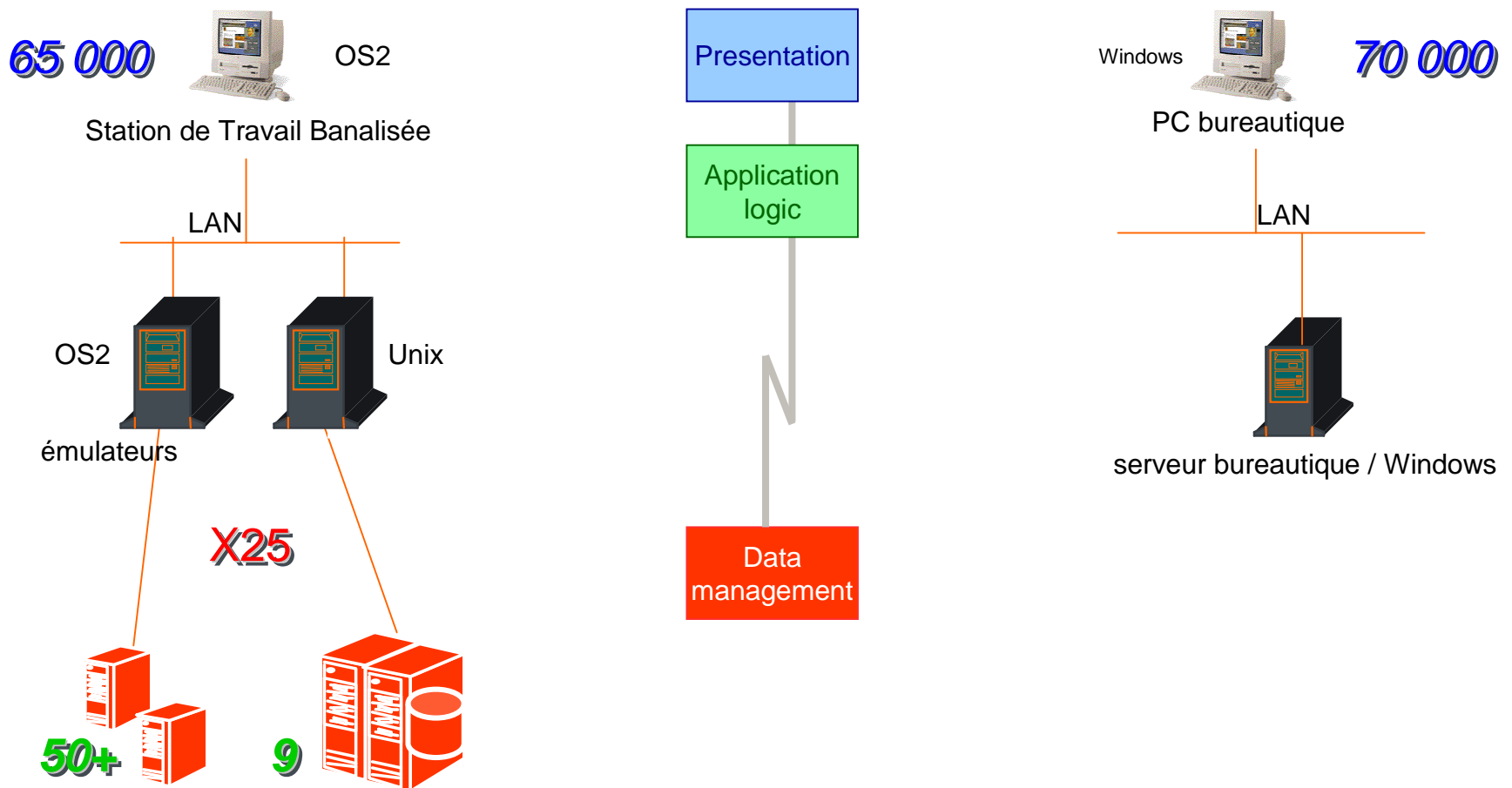


Sommaire

- 2 expériences de transformation de système d'information,
 - ⇒ 1991-1997 : l'architecture technique nationale
 - ⇒ 1998-2002 : archimède et aristote
- pour comprendre l'architecture à 3 niveaux,
- et anticiper la prochaine transformation,
 - ⇒ la nouvelle donne du web 2.0
 - ⇒ qui prépare l'architecture informatique 4.0
- que va embrasser chaque entreprise pour créer du revenu avec ses services.

1991-1997 : Architecture Technique Nationale

Une même infrastructure pour toutes les applications

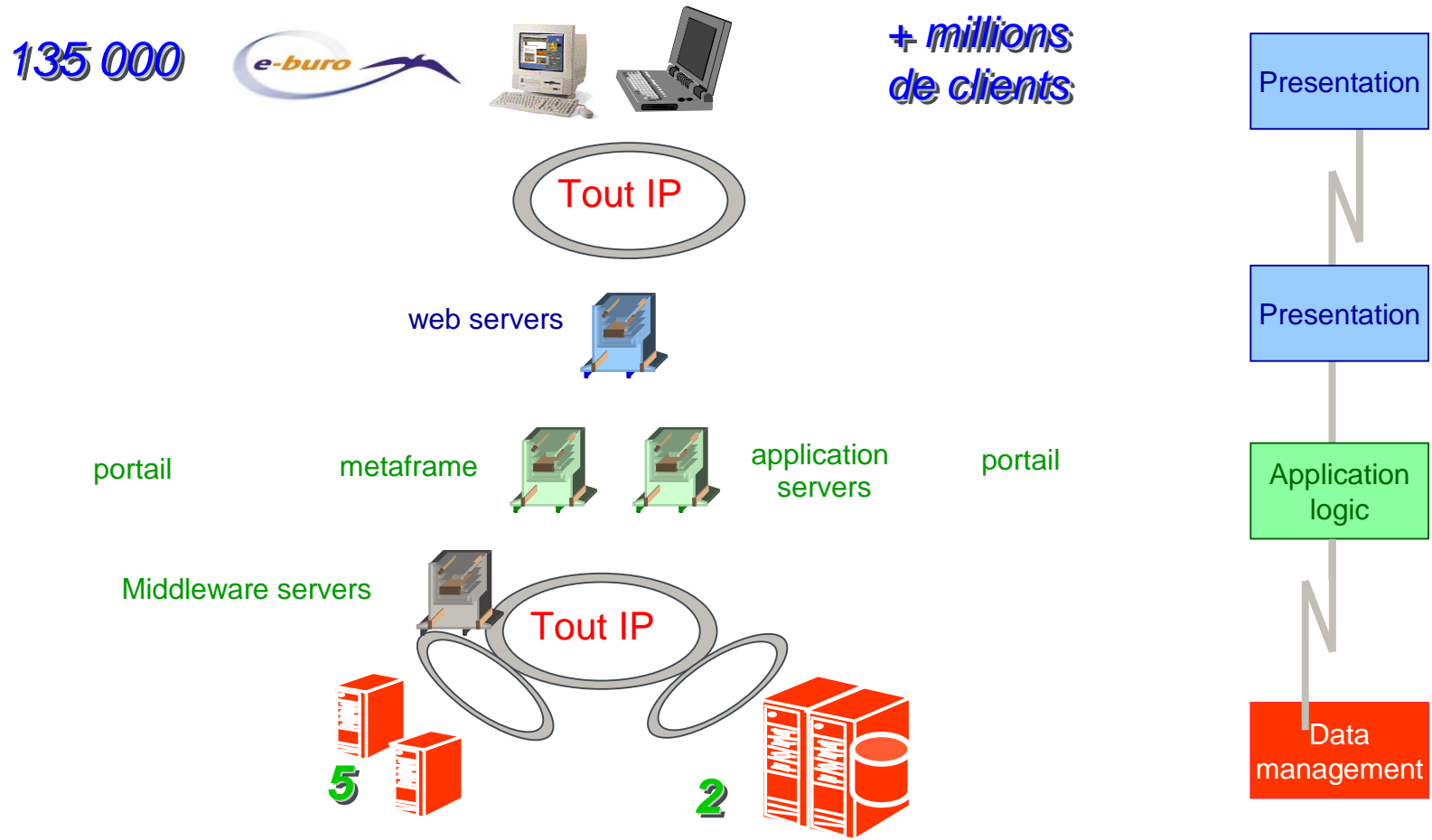


SOA dans l'ATN = banaliser les tuyaux

- L'API-FT
 - ⇒ Un tuyau pour messages bruts entre client et serveur
 - ⇒ Refus d'un format structuré
- L'API-T
 - ⇒ Un tuyau pour appels de programmes
- CFT
 - ⇒ Normalisation du transfert de fichier
- Le RPC est interdit

Le problème de l'explosion combinatoire => rejet du client-serveur

1998-2002 - Archimède : ouvrir le SI sur le monde



SOA = standardiser les tuyaux avec le middleware

- Infrastructure = serveurs techniques + middleware

- Middleware orienté messages (MOM)
 - ⇒MQ-series : véhicule de services
- Middleware orienté transactions (MOT)
 - ⇒Tuxedo : véhicule de services

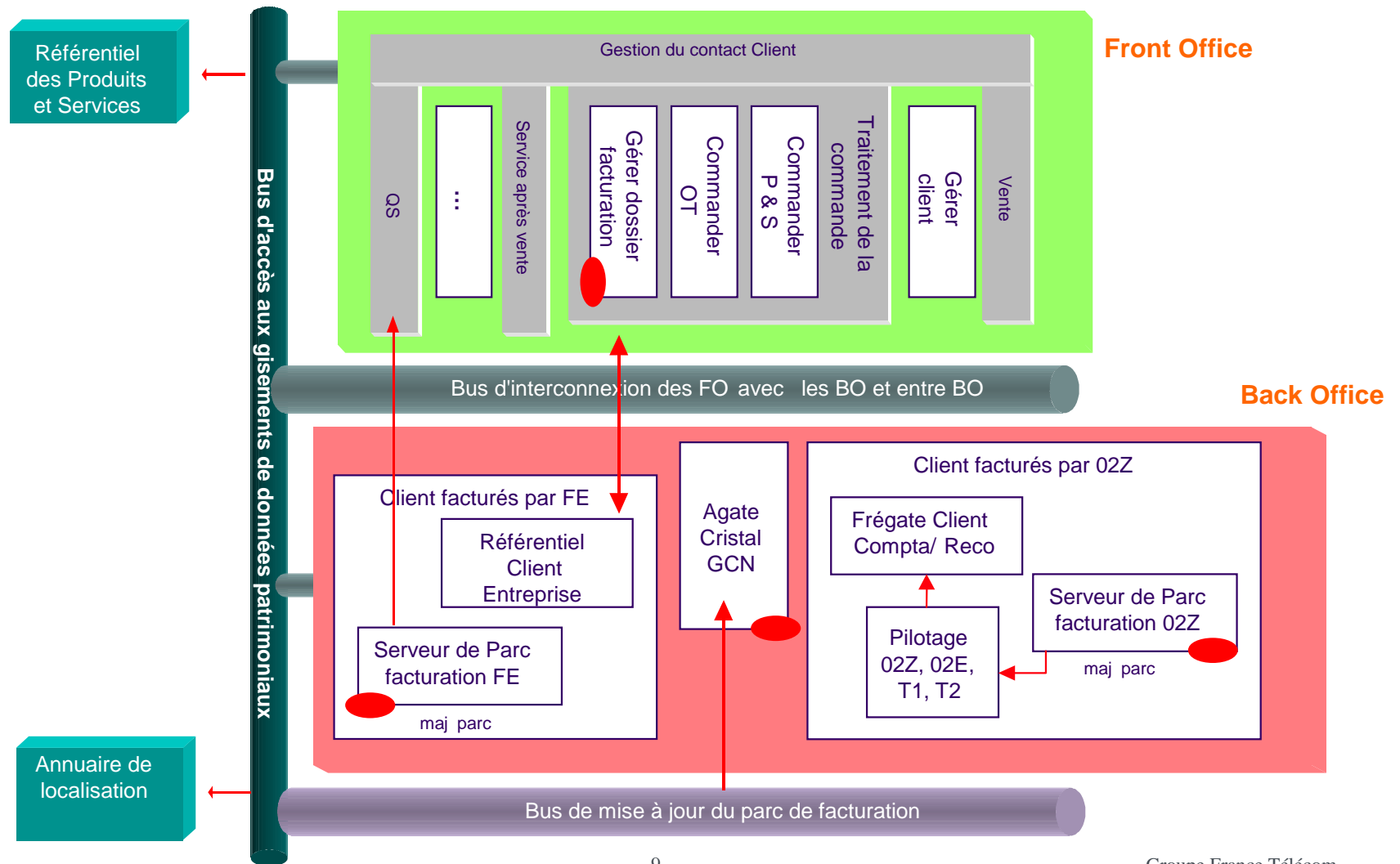
- Les questions qui persistent :
 - ⇒Le mirage de Corba
 - Extensibilité et faible couplage ?
 - ⇒Le purisme de la transaction !
 - A quoi sert le commit à 2 phases ?
 - ⇒Synchrone ou asynchrone ?

2000-2003 - Aristote : la vision d'un chef

- Programme de transformation fonctionnelle :
 - ⇒ Déployer une **architecture de services** : EAI + web services
 - ⇒ Définir ce qu'est un service
 - ⇒ Créer un référentiel de services
- Résistance au changement
 - ⇒ Qui finance ?
 - ⇒ Plus compliqué, plus risqué et plus cher !!!
- Le succès d'un projet phare entraîne les autres



SOA = urbaniser les échanges et rationaliser le middleware

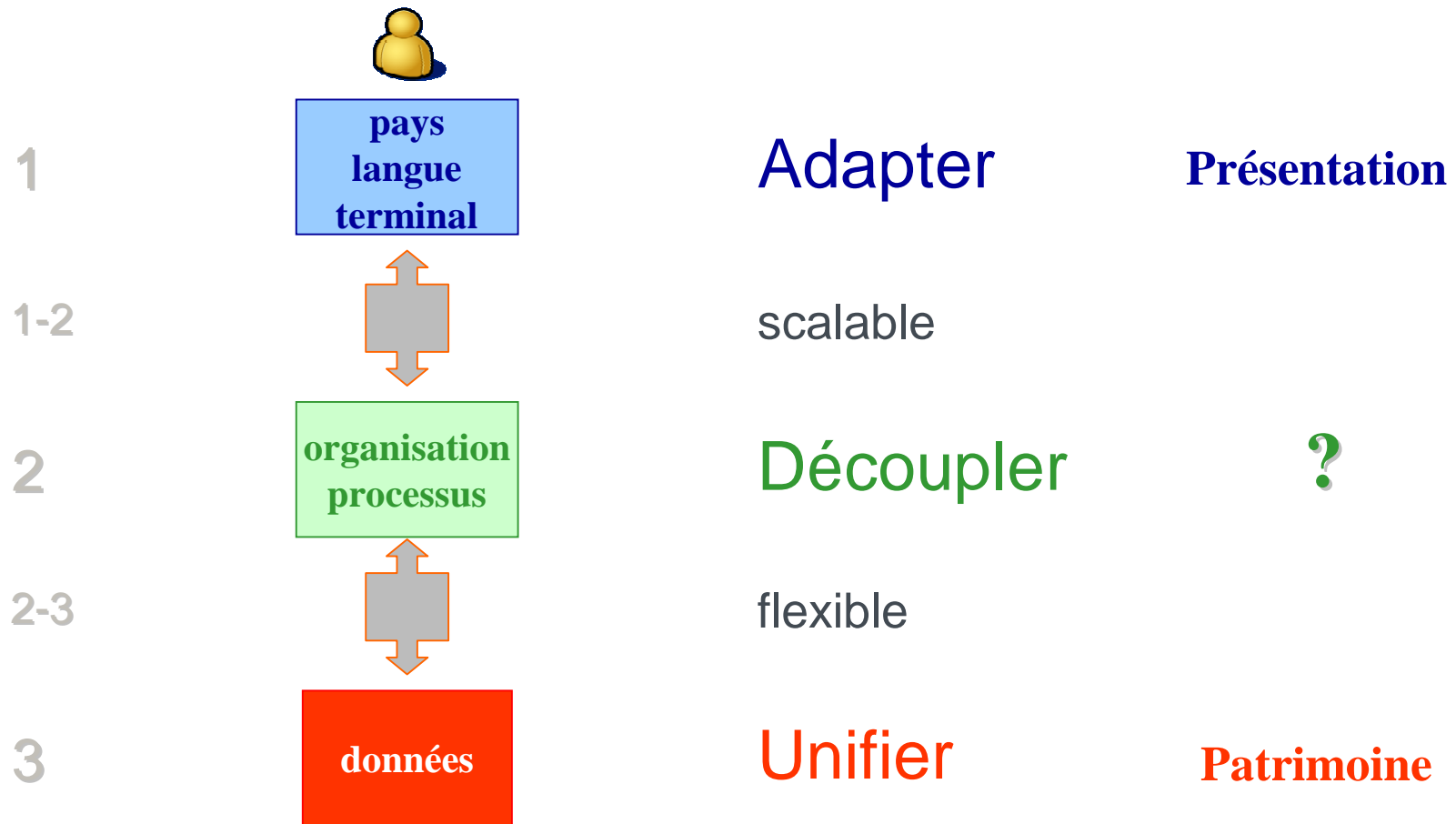


Les clés du succès d'une transformation : archimède + aristote

- Définir deux objectifs de **performance** : réduire les coûts **et** améliorer la QS.
- Le futur sera moins coûteux si chaque projet a **moins** de budget pour faire **en plus** la transformation.
- **Etablir les bonnes pratiques** des nouvelles technos.
- Le tableau de bord de QS est un BAM.
- La QS sera meilleure si chaque personne est intéressée.
- Récompenser pour rendre visible : le prix Performance du SI.

- Résultat :
 - ⇒ **12 milliards** de francs économisés en 4 ans
 - ⇒ Disponibilité annuelle : SI/mainframe : **99,97%** - le reste : **99,2%**

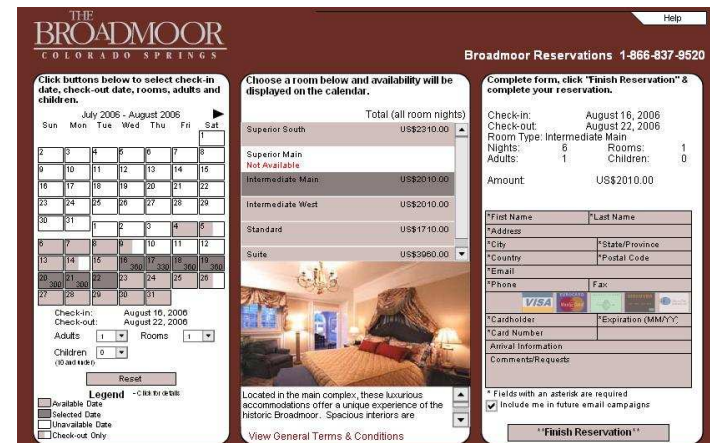
Comprendre l'architecture à 3 niveaux



3 niveaux fonctionnels (pas plus)
5 niveaux techniques (ou plus)

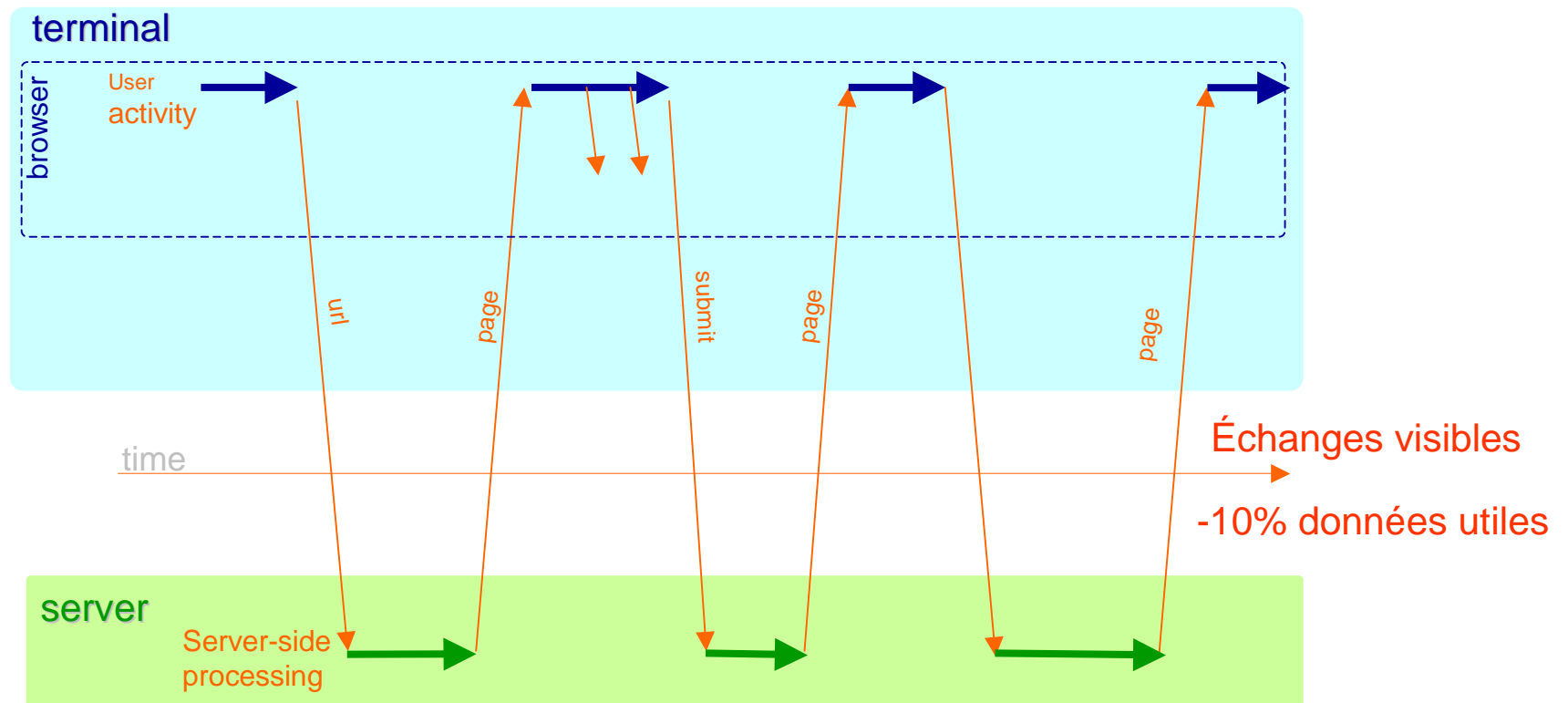
1- Interface riche (RIA)

- Les interfaces homme-machine du web 2.0 résultent de la **fusion** des interfaces graphiques de l'informatique d'entreprise et des arts graphiques numériques.
- L'image devient support de l'interaction.**
- Les flux multimédia s'ajoutent aux flux de données.



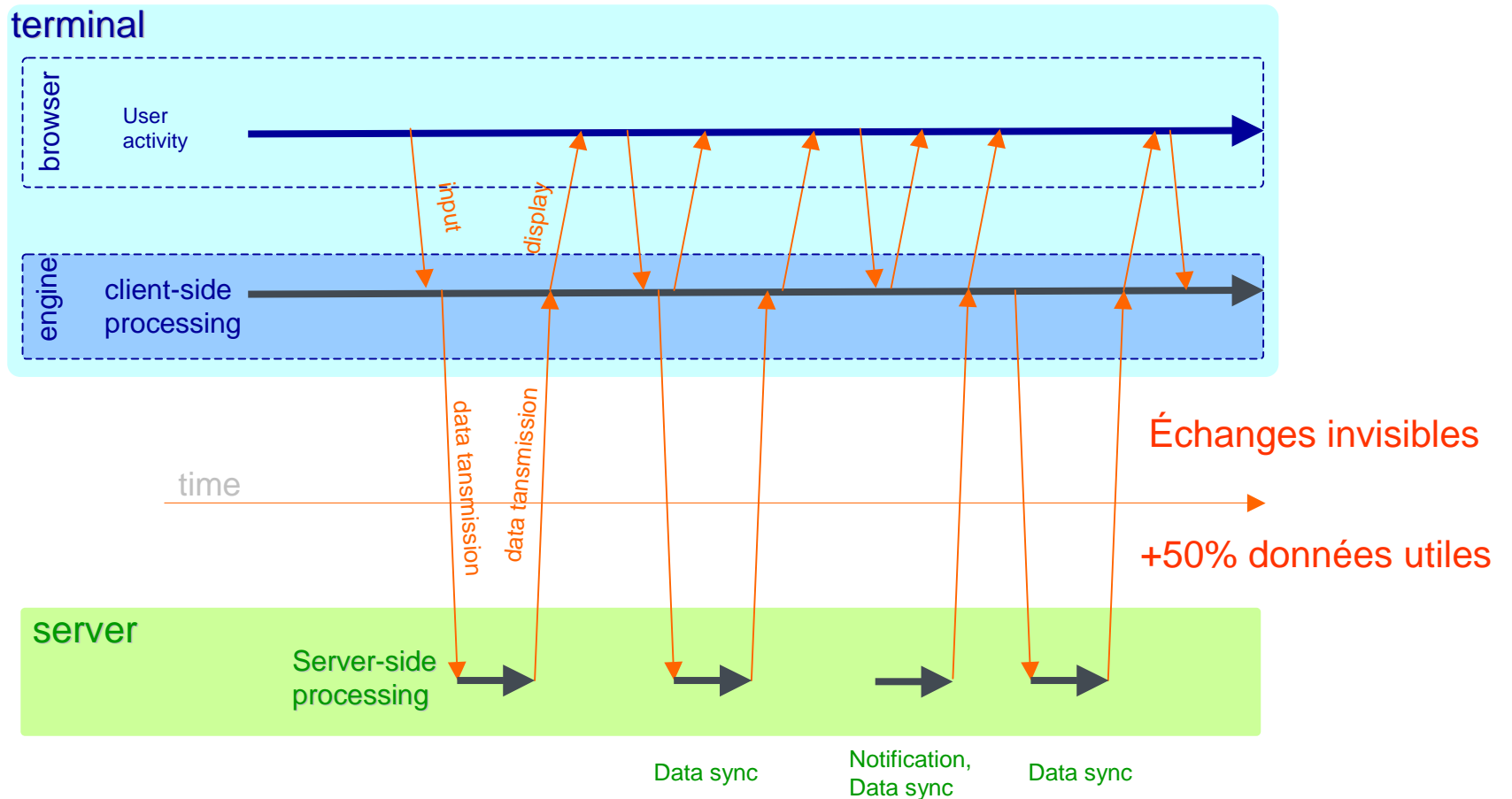
2- Remplacer la communication requête-page ...

Attente, scintillement, pagination



... par des échanges asynchrones structurés

Fluidité, multimédia, tout en un



Asynchrone

- La plupart des actions de l'utilisateur ne remontent pas vers le serveur
- Lorsqu'une action remonte vers le serveur, elle demande une information du serveur.
- L'attente de la réponse ne doit pas dégrader la fluidité de l'interaction avec l'utilisateur

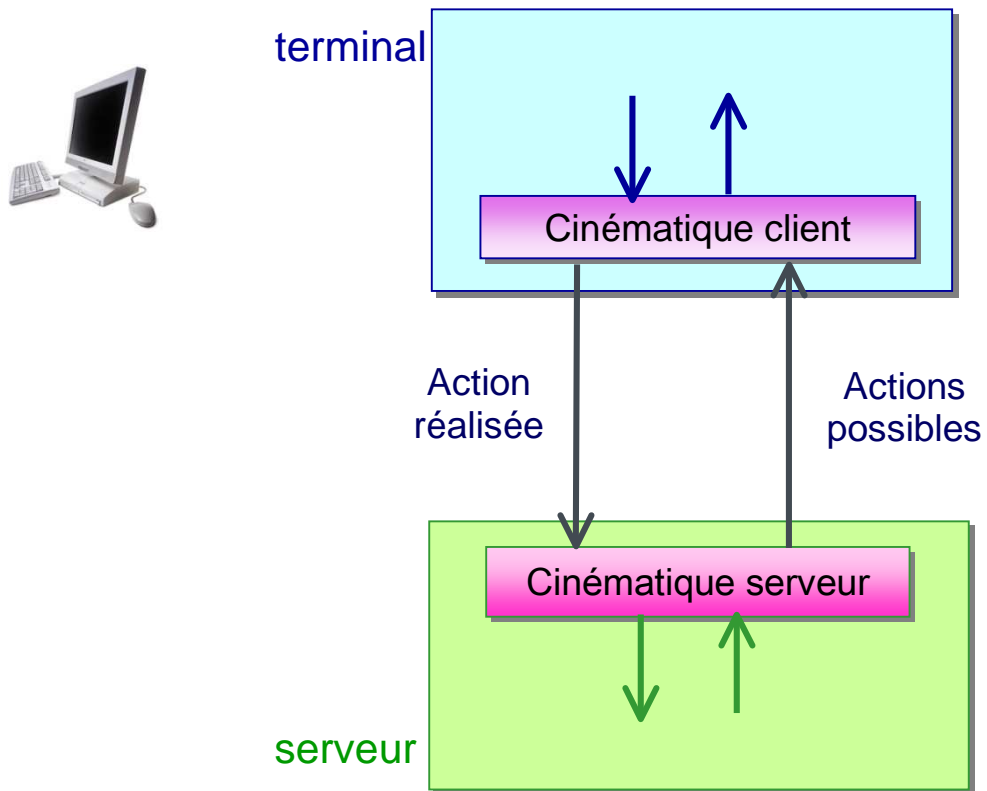
Questions d'architecture (SOA – BPM) :

Quelles actions doivent remonter vers le serveur ?

Quelles relations entre ces actions et les processus métiers ?

Comment garantir les temps de réponse ?

Une cinématique client-serveur



3- Le mash-up

housingmaps



GoogleMaps

Craig's list

Mash-up : nouvelle technique de SOA ?

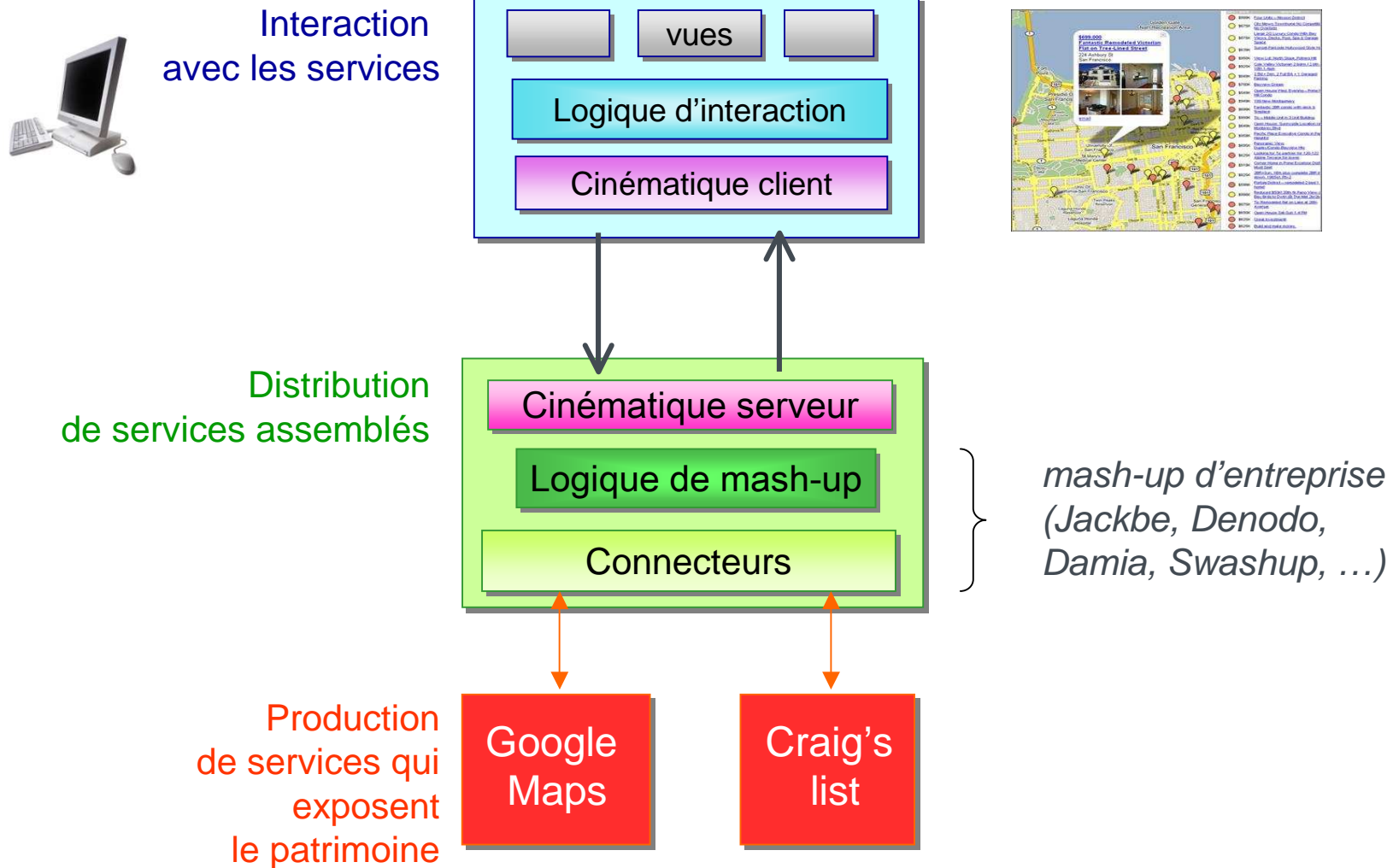
- Le mash-up entrelace des flux de toutes natures
 - ⇒ qui peuvent être des services fournis par le SOA
- Le mash-up peut être fait sur le terminal ou sur le serveur
- Le mash-up peut être une solution d'attente au SOA :
 - ⇒ Le mash-up expose des services construits sur l'existant pendant que les projets fabriquent les services web.

Questions d'architecture :

Où faire les mash-up ? terminal ou serveur ?

Quelle complémentarité entre mash-up et SOA ?

Une architecture de mash-up - côté serveur



4- Loading

- Le serveur télécharge le logiciel sur le terminal
 - ⇒ le déploiement est automatique
- Mais :
 - ⇒ si les programmes sont conservés en cache dans un proxy ou dans le terminal, le changement de version est laborieux.
 - ⇒ le temps de chargement peut être important.
 - ⇒ que se passe-t'il si 2 serveurs différents :
 - téléchargent 2 programmes ayant le même nom ?
 - téléchargent des programmes qui entrent en conflit sur le terminal ?

A l'avenir, il est probable que

Le logiciel reste caché dans le terminal et soit télémanagé par le serveur

Web 2.0 = 4^{ième} génération d'architecture informatique

- 1.0 : le programme fonctionne dans le serveur – le terminal est passif.
- 2.0 : le programme fonctionne dans le terminal – le terminal est un vrai ordinateur (PC). Le serveur en fait le moins possible, pour offrir un maximum d'accès simultanés.
- 3.0 : le programme fonctionne dans le serveur. Le serveur fait tout. Le terminal doit être léger et ne rien faire.
- 4.0 : le programme est réparti entre terminal et serveur. Toute la présentation est faite sur le terminal. **Le serveur ne connaît pas la présentation et expose et consomme des services.**

Character

+

Graphical

+

Illustration

+

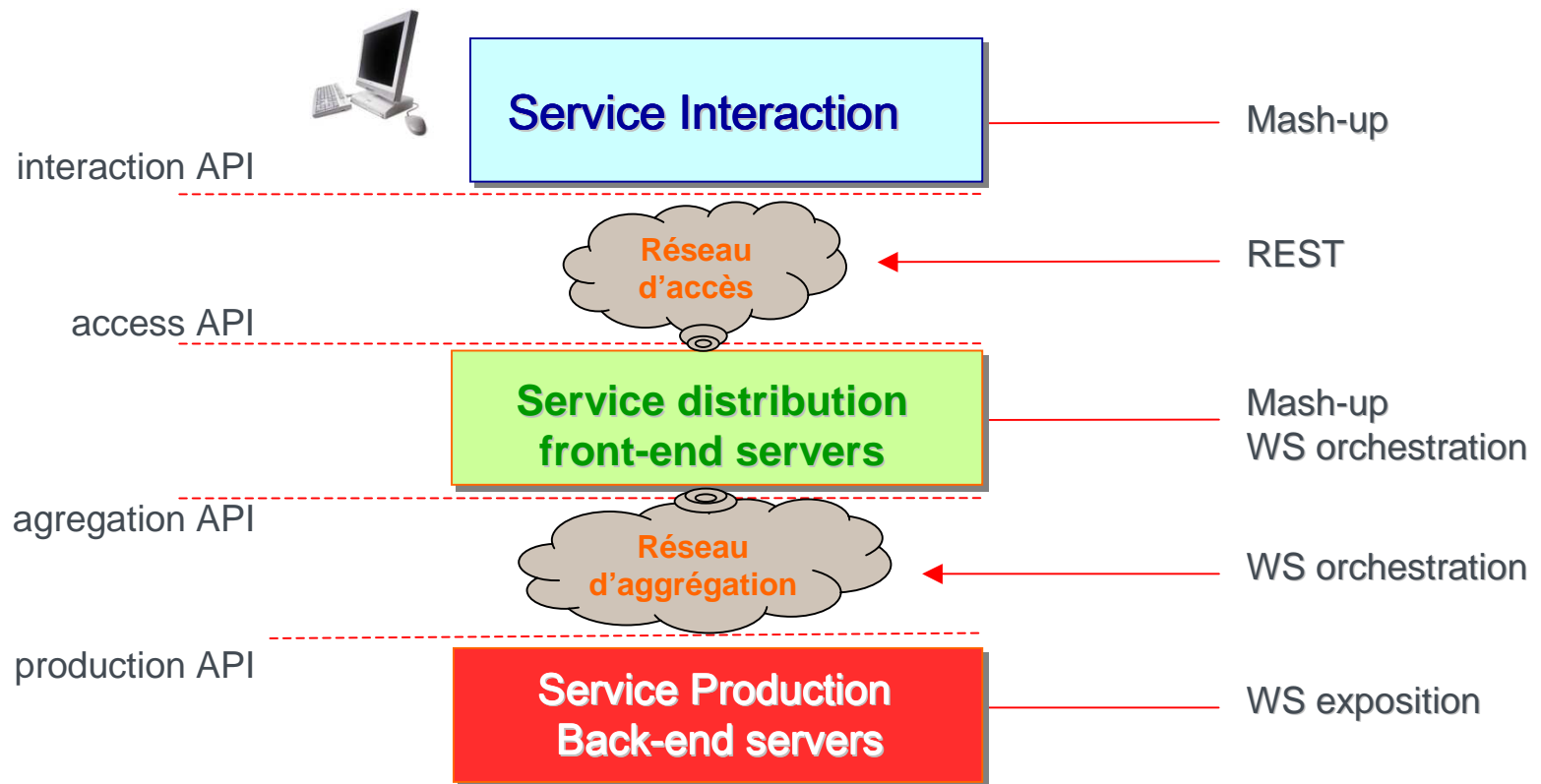
Animation

Architecture 4.0 = RIA + SOA

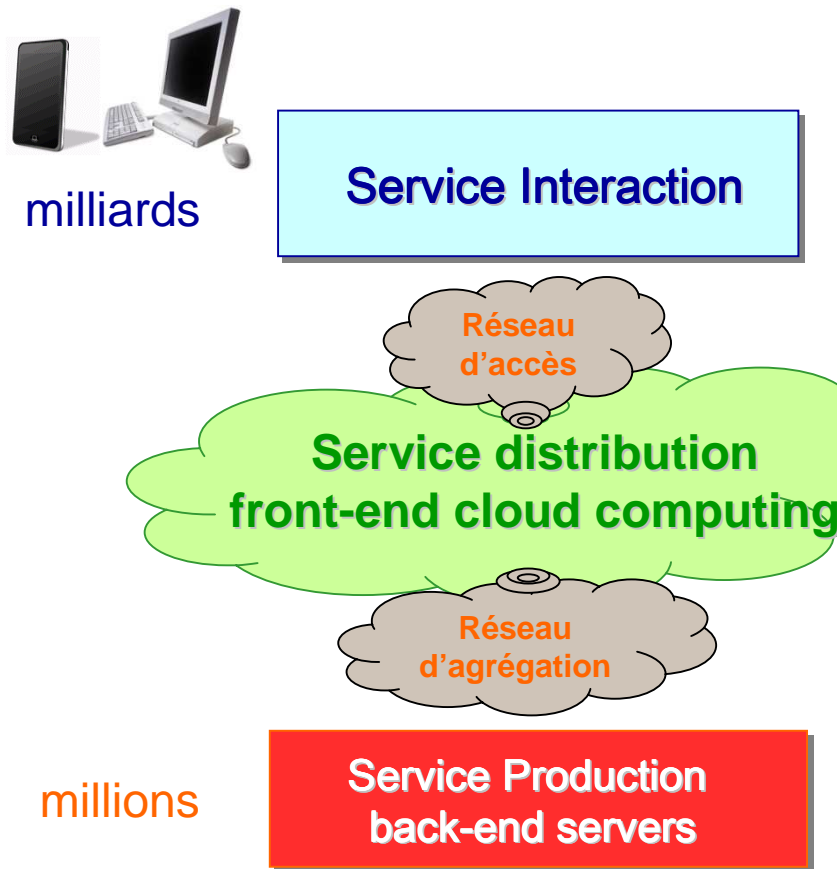
Architecture 4.0 = APIs ouvertes sur Internet

APIs de services

SOA



5- Le cloud : infrastructure de l'architecture 4.0



Très peu d'entreprises auront les moyens de posséder l'architecture 4.0, de bout en bout

Questions d'urbanisme :

- Quelles briques mon entreprise doit détenir, voire fournir ?
- Quelles briques doit-elle utiliser comme SaaS ?
- Quels contrats de services avec les tiers vont garantir la satisfaction des clients et l'efficacité des employés ?

Qui garantit le fonctionnement de bout en bout ?

Service Oriented Enterprises

- Les entreprises vont fournir les services qui leur créent des revenus.
- Elles seront interconnectées avec leur concurrents et partenaires dans la même **architecture 4.0, multi-entreprises**.

Questions d'**Architecture d'Entreprise** :

Quel patrimoine informationnel de l'entreprise mérite d'être développé pour créer des revenus ?

Comment les partenariats vont entrelacer les SI ?

Comment des concurrents peuvent partager la même architecture ?

Urbanisme 1.0 = Service Oriented Enterprise Architecture



www.urba-ea.org

Questions ?

Web 2.0 => Architecture 4.0

=> Service Oriented Enterprises

=> Urbanisme 1.0 (Enterprise Architecture)

herve.crespel@orange-ftgroup.com

Président du club urba-EA

