

Introduction aux applications réparties

Sacha Krakowiak

Université Joseph Fourier
Labo. Sirac (INPG-INRIA-UJF)

Plan de la présentation

Applications réparties : applications s'exécutant sur plusieurs sites reliés par un réseau de communication

- † L'évolution technologique récente et son impact sur les applications réparties
- † Quelques exemples d'applications réparties
 - † Flots de données
 - † Édition coopérative
 - † Télévision interactive
 - † Commerce électronique
 - † Vidéoconférence
- † Caractéristiques et besoins des applications réparties
- † Plan de l'École

L'évolution technologique récente et son impact sur les applications réparties

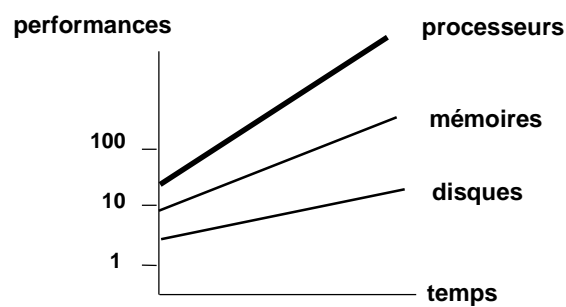
† Influence de l'évolution technologique sur les applications

- † Extension du domaine de faisabilité
 - † Amélioration des applications existantes
 - † Nouveaux champs d'applications
- † Modifications du poids relatif des coûts et contraintes
 - † Choix de nouvelles solutions techniques
 - † Nouveaux "points d'équilibre" (coût-efficacité, etc.)

† Aspects examinés

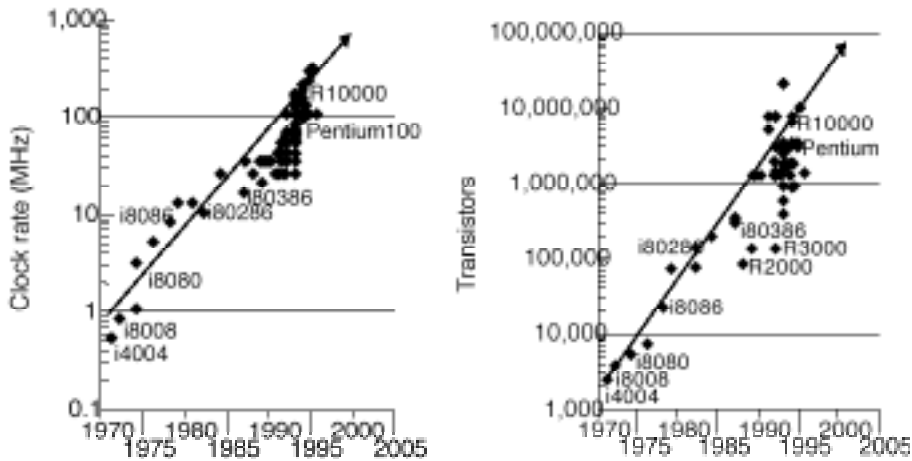
- † Composants informatiques
- † Réseaux et communications

Évolution des composants informatiques Processeurs, mémoires, disques



- † Performances absolues des processeurs
- † Différentiel processeurs-mémoires-disques

Évolution des processeurs



Source : D. Culler, J. P. Singh, A. Gupta, *Parallel Computer Architecture: A Hardware/Software Approach*, Morgan Kaufmann, 1998.

Évolution des performances des processeurs Conséquences

- † **Accroissement de la vitesse**
 - † traitements élaborés (images, recherche d'info., etc.)
 - † regain des techniques interprétatives
 - † liaison dynamique
- † **Accroissement de la densité + baisse coût unitaire**
 - † processeurs autonomes (portables, cartes, etc)
- † **Rapidité d'évolution + rapport performances/prix**
 - † effacement des machines parallèles spécialisées
 - † limites du parallélisme interne => parallélisme inter-processeurs
 - † extension du domaine d'application des multiprocesseurs

Différentiel processeurs-mémoires-disques Conséquences

- † **Les caches prennent une importance croissante**
 - † Multiples niveaux entre processeur et mémoire
 - † Caches logiciels
- † **Le parallélisme aggrave la différence de performances**
 - † Limites des MPPs (performances des bus)
 - † Intérêt des machines à mémoire distribuée
- † **Le parallélisme s'étend à la mémoire secondaire**
 - † RAIDs (motivation principale : disponibilité)
 - † Disques sur réseau (extension répartie du principe des RAIDs)

Évolution des réseaux

Accélération du progrès des fonctions et performances des réseaux

- † **Réseaux à grande distance et réseaux "métropolitains"**
 - † L'Internet (+ intranets, etc.)
- † **Réseaux locaux**
 - † Réseaux "départementaux"
 - † Réseaux à couplage étroit (grappes de machines)
- † **Communications mobiles**

Évolution de l'Internet un exemple : les applications multimédia

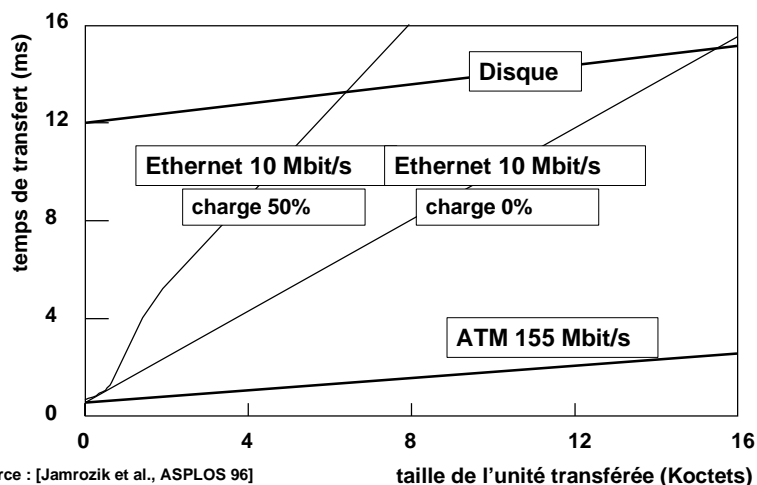
† Caractéristiques des applications multimédia

- † diffusion sélective (plutôt que point à point)
- † exigences de qualité de service (contraintes sur bande passante, stabilité, etc.)

† Solutions envisagées

- † généralisation du routage multipoint (à terme Mbone = Internet)
- † adaptation du réseau aux applications
 - † réservation de ressources (ex : RSVP)
- † adaptation des applications au réseau
 - † contrôle de délai, de débit et de pertes, éventuellement mode dégradé (ex : IVS)

Réseaux locaux : performances en fonction de la taille de l'unité de transfert



source : [Jamrozik et al., ASPLOS 96]

$$t = t_0 + N/D$$

t_0 = latence, D = débit, N = taille

Conséquences de l'évolution des réseaux

- † **Gestion globale de l'information sur un réseau**
- † **Développement de l'usage des caches**
 - † internes aux systèmes et logiciels de base...
 - † ... mais aussi contrôlés par les applications
 - † Caches coopératifs
 - † Préchargement guidé par les applications
- † **Le réseau comme support d'information**
 - † NC vs PC
 - † Caches et systèmes de stockage attachés au réseau
 - † Réduction de la latence d'accès
 - † Capacité de croissance

Techniques d'interconnexion

- † **Réseaux locaux à hautes performances**
 - † Réseaux commutés
 - † ATM, Myrinet (transfert explicite)
 - † Réseaux à capacités d'adressage
 - † SCI (écriture directe en mémoire distante)
- † **Serveurs en grappes (*clusters*)**
 - † Bon rapport performances-prix (composants standard)
 - † Applications potentielles
 - † Calcul intensif (parallélisme)
 - † Gestion d'information (grand volume, partage, haut débit)
 - † Nouveaux outils de programmation

Communications mobiles : caractéristiques

- † Faible bande passante
- † Bande passante variable
- † Déconnexions fréquentes
- † Taux d'erreur élevé
- † Réseaux hétérogènes
- † Changement d'adresse et d'environnement
- † Ressources limitées pour les mobiles
- † Risques accrus pour la sécurité

Communications mobiles : conséquences pour les applications

- † **Au niveau des communications**
 - † Changements possibles de l'adresse du mobile
 - † recherche de protocoles à adresse invariante
 - † Gestion des informations dépendantes de la localisation
 - † données de configuration : serveur de noms, imprimantes, etc.
- † **Au niveau de la gestion des applications**
 - † Nécessité d'*adaptation* des applications aux conditions variables (QoS, taux d'erreurs, etc)
 - † modes dégradés
 - † Gestion du mode déconnecté
 - † gestion de caches
 - † protocoles de reconnexion (cohérence)

Conclusion sur l'influence de l'évolution technologique des réseaux

- † **Interaction nécessaire entre application et système (pour une utilisation efficace des ressources)**
- † **Adaptation (dynamique) des applications aux ressources disponibles**
 - † exemple 1 : communications multimédia
 - † exemple 2 : communications mobiles
- † **Adaptation (dynamique) du système aux besoins des applications**
 - † exemple 1 : stratégies (caches, préchargement) pilotées par les applications
 - † exemple 2 : réseaux actifs (code téléchargé dans les routeurs)
 - † exemple 3 : systèmes configurables (chargement dynamique de modules)

Pourquoi des applications réparties ?

La plupart des applications informatiques sont, de fait, réparties.

- † **Besoins propres des applications**
 - † Communication, coopération
 - † Partage d'information, accès à des ressources distantes
 - † Répartition "intrinsèque" des applications
- † **Possibilités techniques**
 - † Coût et performances des machines et des communications
 - † Interconnexion généralisée
 - † Exemple : interpénétration informatique-télécom-télévision
 - † Les applications informatiques utilisent les techniques des télécom (réseau téléphonique, mobiles)
 - † Les réseaux de télécom sont des systèmes informatiques
 - † La télévision devient numérique et interactive

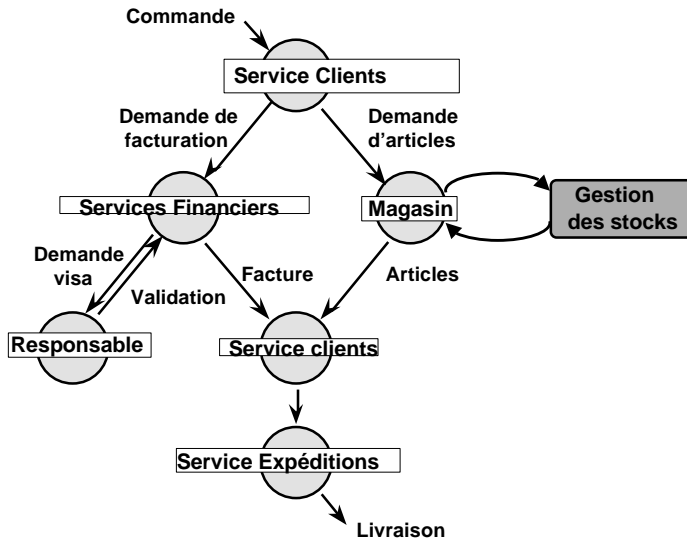
Quelques classes d'applications réparties

- † Coordination d'activités
 - † Systèmes à flots de données ("workflow")
 - † Systèmes à "agents"
- † Communication et partage d'information
 - † Bibliothèques virtuelles
- † Collecticiels
 - † Édition coopérative
 - † Téléconférence
 - † Ingénierie concourante
- † Applications "temps réel"
 - † Contrôle de procédés
 - † Avionique, etc.
 - † Localisation de mobiles
- † Nouveaux services grand public
 - † Presse électronique
 - † Télévision interactive
 - † Commerce électronique

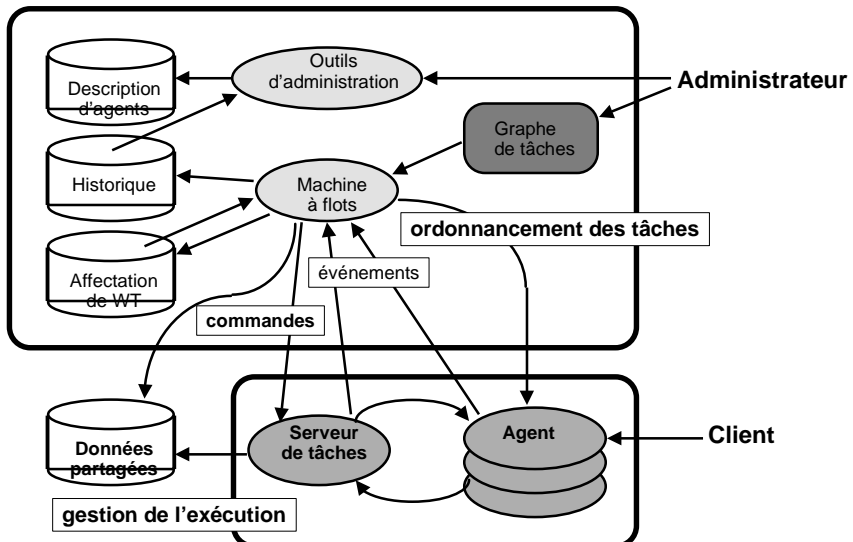
Applications à flots de données ("workflow")

- † **Fonction : exécution coordonnée d'un ensemble de tâches**
 - † Exécution sur différents organes de traitement
 - † Coordination = séquençement + échange de données
 - † Tâche : procédure spécifiée (souvent application existante)
 - † Modèle courant : événements -> réactions
- † **Applications**
 - † Procédures administratives (circulation de documents)
 - † Procédés industriels (conception, fabrication, transport)
 - † Fonctionnement de services (santé, etc.)
 - † Développement de logiciel

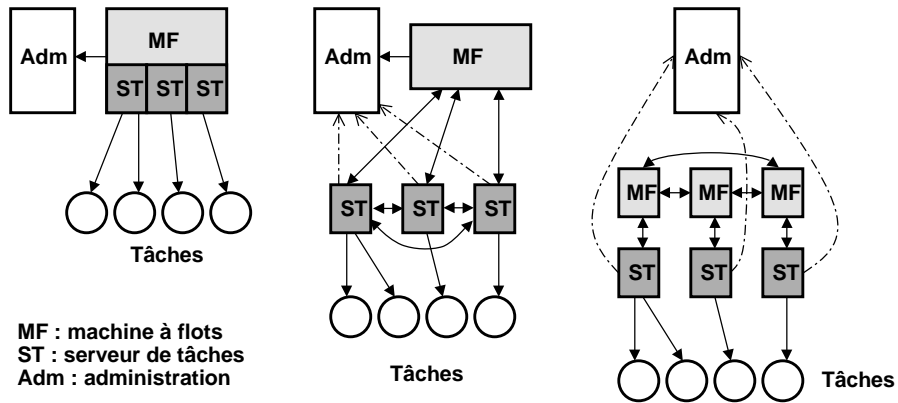
Exemple d'application à flots de données



Structure logique d'un système à flots de données



Organisation physique d'un système à flots de données



Applications à flots de données Commentaires sur l'étude de cas

- † **Coordination des tâches**
 - † Synchronisation et communication
 - † Mécanismes à base d'événements
- † **Gestion de l'exécution**
 - † Association tâche - agent
 - † Rôles, groupes de processus
- † **Adaptation à l'environnement**
 - † Correspondance structure logique -> structure physique
 - † Support (système centralisé, grappe, LAN, WAN, ...)
 - † Protocole (objets partagés, RPC, CORBA, bus, WWW, ...)
- † **Réutilisation de l'existant**

Télévision interactive

† Fonctions : fourniture d'un ensemble de services aux clients

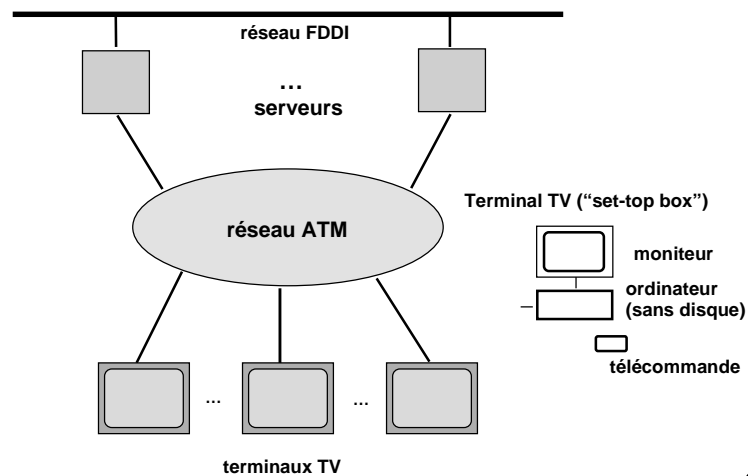
- † Diffusion de programmes à la demande (individuelle)
- † Télé-achat
- † Jeux interactifs

† Contraintes

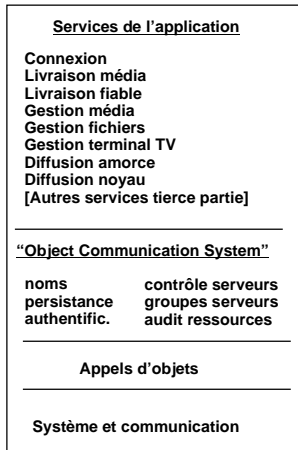
- † Interface client familière (télécommande)
- † Disponibilité
- † Performances
- † Coût (terminal client simple)
- † Extensibilité

Source : M. N. Nelson, M. Linton, S. Owicki, "A Highly Available, Scalable ITV System", *Proc. 15th ACM Symp. on Operating Systems Principles*, December 1995 [expérience Time-Warner, Orlando]

Télévision interactive : architecture du réseau



Service de télévision interactive : services et objets



Serveur

† Spécification des services

† Interface (IDL)

† Réalisation des services

† 1 service = 1 objet

† désignation par référence

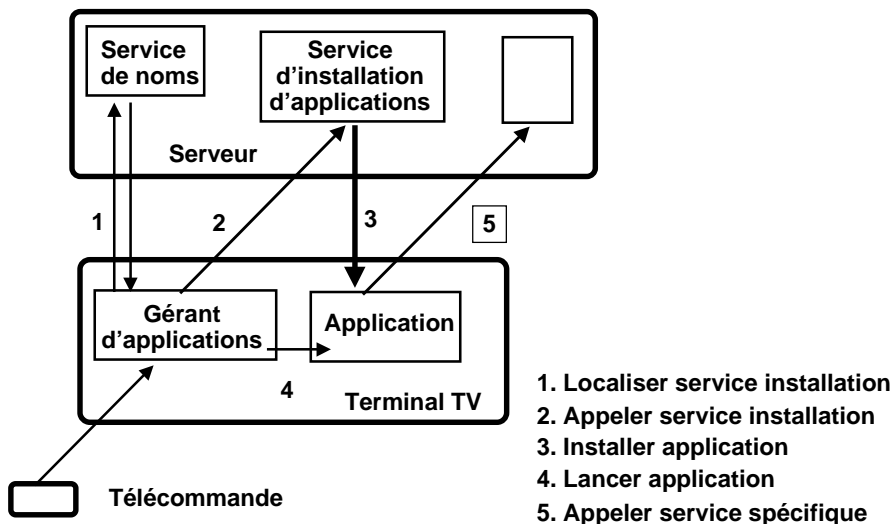
† enregistrement dans service de noms

† Disponibilité

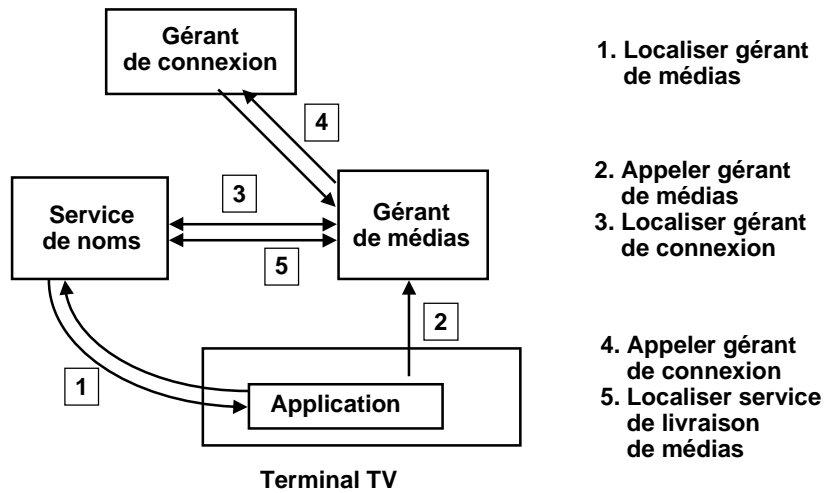
† Duplication active

† Serveur primaire/secondaire

Télévision interactive : lancement d'une application



Télévision interactive : gestion des ressources



1. Localiser gérant de médias

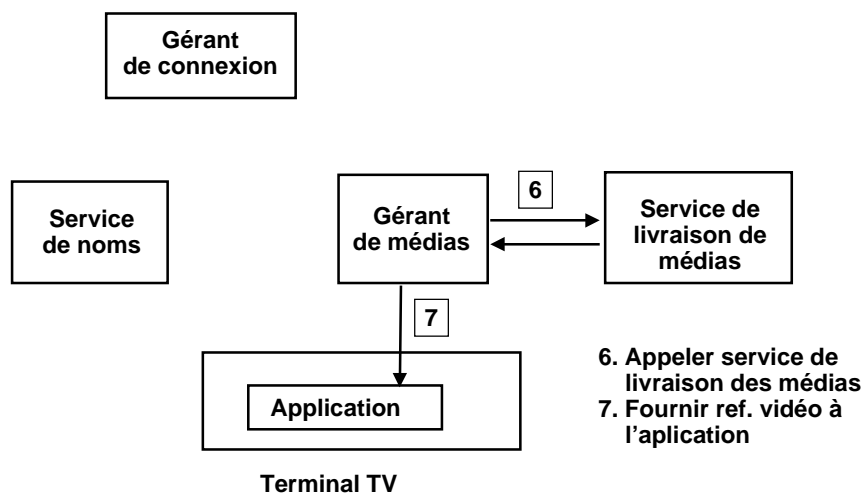
2. Appeler gérant de médias

3. Localiser gérant de connexion

4. Appeler gérant de connexion

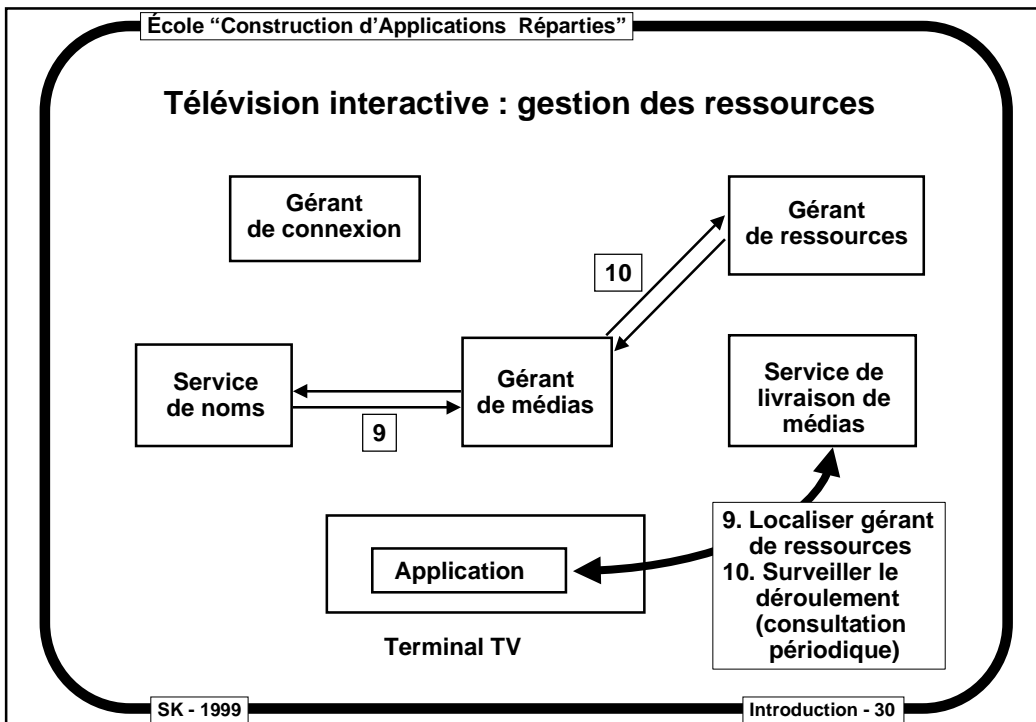
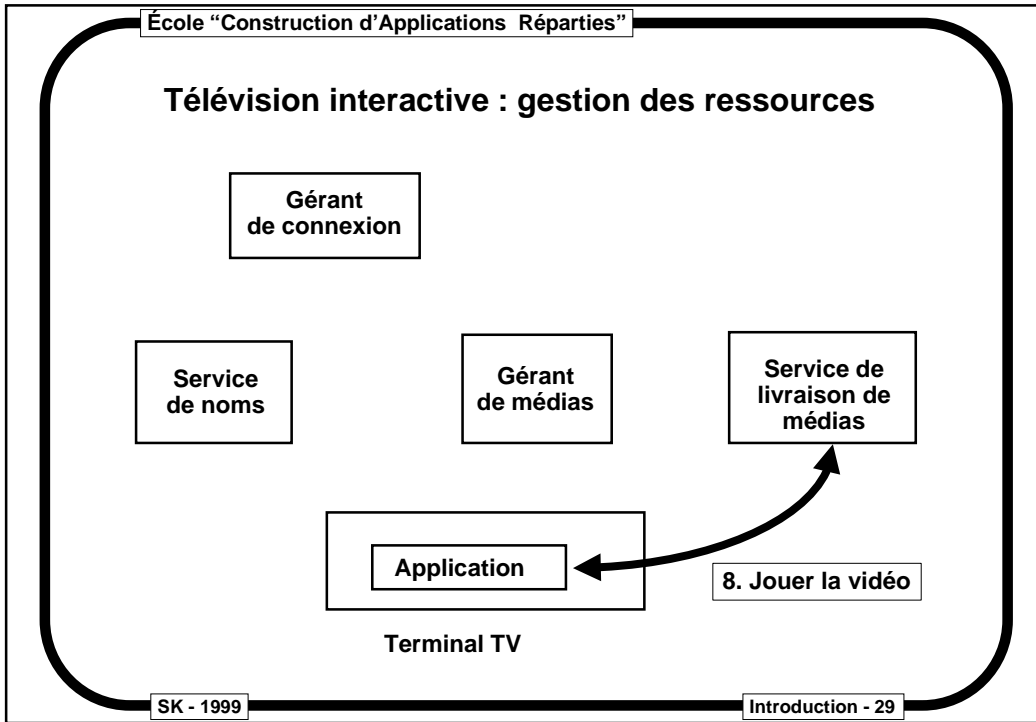
5. Localiser service de livraison de médias

Télévision interactive : gestion des ressources



6. Appeler service de livraison des médias

7. Fournir ref. vidéo à l'application



Télévision interactive Commentaires sur l'étude de cas

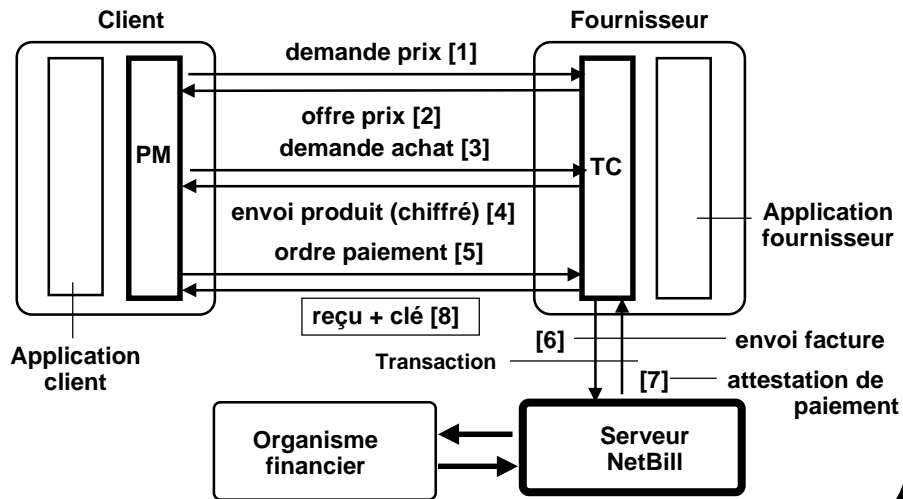
- † **Contraintes d'une application "grand public"**
 - † Disponibilité => duplication active/passive, reprise
 - † Performances => bon dimensionnement ; répartition de charge
 - † Simplicité de l'interface => terminal TV
 - † Passage à l'échelle => évolution incrémentale
- † **Génie logiciel d'une grande application répartie**
 - † Organisation client-serveur
 - † Modèle d'objets
 - † Utilisation systématique de l'IDL
 - † Service de noms évolué
 - † Capacité d'évolution

Commerce électronique

- † **Fonction : exécution de transactions commerciales entre clients et fournisseurs**
 - † Recherche de produits et services (catalogues)
 - † Commande simple ou groupée
 - † Paiement
 - † Livraison
- † **Contraintes**
 - † Protection des informations confidentielles (client et fournisseur)
 - † Respect des règles de concurrence
 - † Respect des garanties fournisseur->client (offre sincère, exécution du contrat)
 - † Respect des garanties client->fournisseur (identité, paiement)
 - † Respect des droits de propriété (licences, droits d'auteur)
 - † Disponibilité permanente du service

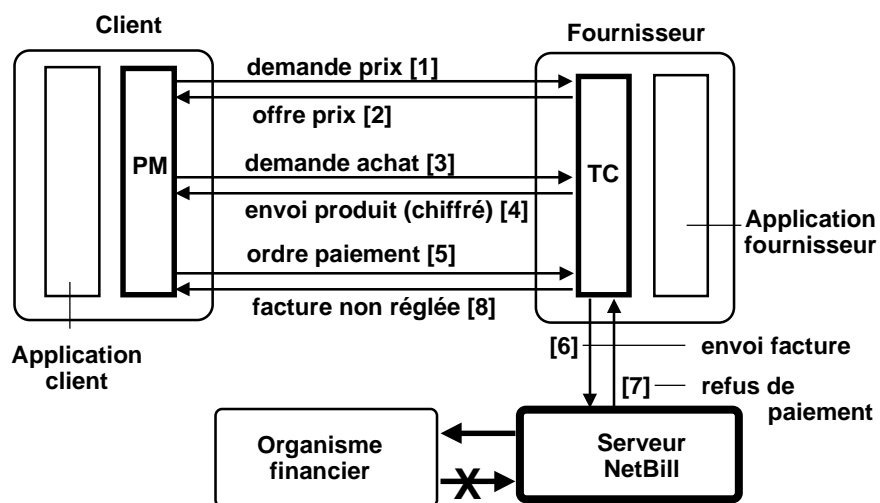
Un service de commerce électronique (NetBill)

Cas spécial : produit disponible sous forme électronique



Un service de commerce électronique (NetBill)

Cas spécial : produit disponible sous forme électronique



Tolérance aux fautes dans NetBill Point de vue du client

- † **Panne dans les étapes 1 à 4**
 - † aucune transaction financière n'a eu lieu
 - † pas de paiement, pas de livraison
- † **Panne après émission de [5] (ordre de paiement)**
 - † le client a accepté le produit
 - † pas de réponse du serveur
 - † le client a l'initiative de la reprise (contacte fournisseur ou serveur NetBill pour connaître l'état de la requête)
- † **Erreurs possibles**
 - † Ordre de paiement non transmis au serveur : annuler (délai de garde)
 - † Ordre de paiement transmis et accepté (transaction) : le client finira par recevoir la clé (si besoin, du serveur)

Tolérance aux fautes dans NetBill Point de vue du fournisseur

- † **Panne dans les étapes 1 à 5**
 - † aucune transaction financière n'a eu lieu
 - † pas de paiement, pas de livraison
- † **Panne après émission de [6] (facture)**
 - † le fournisseur finira par obtenir une réponse du serveur NetBill (au besoin en renvoyant la facture [6])
 - † propriété transactionnelle sur [6]+[7] (envoi facture et résultat du règlement) : annulation possible si panne durable

Source pour NetBill : <http://www.ini.cmu.edu/NETBILL/>

Commerce électronique Commentaires sur l'étude de cas

† Sécurité

- † Confidentialité (secret des informations) : chiffrement
- † Intégrité (pas de modifications non désirées)
- † Authentification
 - † des partenaires (signature électronique)
 - † du contenu des messages

† Tolérance aux fautes

- † Atomicité des transactions commerciales (paiement+livraison)
- † Garanties assurées par le serveur (état défini, opérations transactionnelles)
- † Pas d'hypothèses sur sites extérieurs au serveur

Un système expérimental de vidéoconférence

† Fonctions

- † Réaliser une réunion virtuelle, le flot audio et vidéo produit par chaque participant étant diffusé aux autres.

† Contraintes

- † Permettre un déploiement sur un réseau hétérogène (bande passante différente d'un participant à l'autre)
- † Permettre l'usage de programmes personnalisés pour chaque participant
- † Permettre l'adaptation à des conditions variables d'exploitation
- † Assurer la capacité de croissance du système

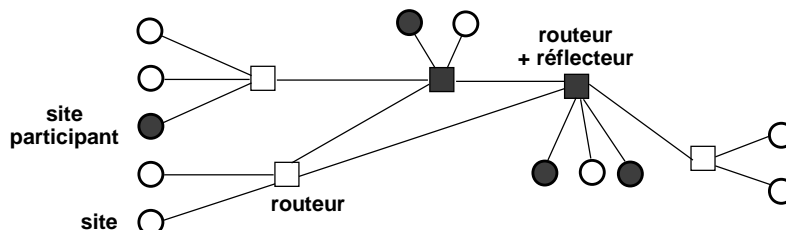
Source : M. Baldi, G. P. Picco, F. Risso, "Designing a Videoconference system for Active Networks", *Proc. 2nd Int. Workshop on Mobile Agents (MA98)*, Stuttgart, Sept. 1998.
<http://www.elet.polimi.it/Users/DEI/Sections/Compeng/GianPietro.Picco/papers/ma98.anet.ps.gz>

Choix de conception

- † **Système à diffusion (*multicast*)**
 - † Utiliser un réseau à diffusion existant (exemple Mbone)
 - † Bonne capacité de croissance
 - † Peu de capacités d'adaptation
- † **Architecture centralisée (serveur)**
 - † Gestion simplifiée, bonne adaptabilité
 - † Capacité de croissance limitée, faible tolérance aux fautes
- † **Choix proposé : serveur(s) adaptable(s)**
 - † Serveur pouvant
 - † être modifié dynamiquement
 - † être dupliqué ou changer de place
 - † Combine adaptabilité et capacité de croissance
 - † Utilise la technique des réseaux actifs

Organisation du système

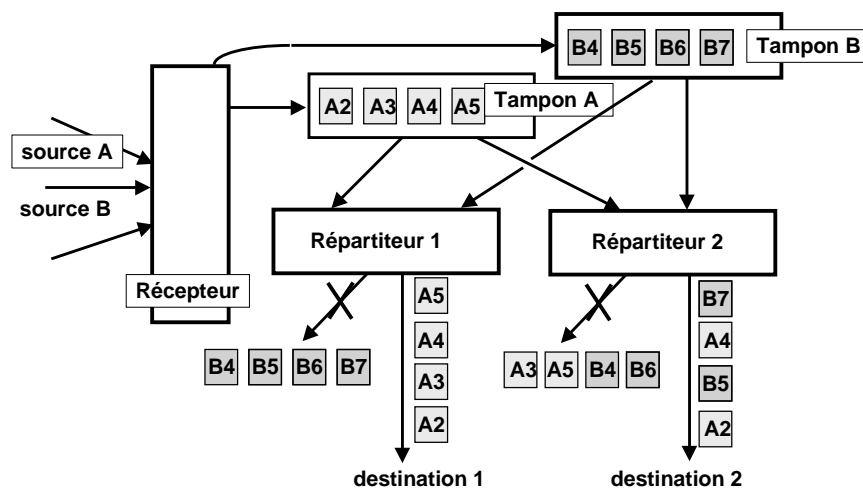
- † **"Réflecteurs" = serveurs adaptables**
 - † Implantés dans les routeurs
 - † Code téléchargeable
 - † Duplication et migration possibles



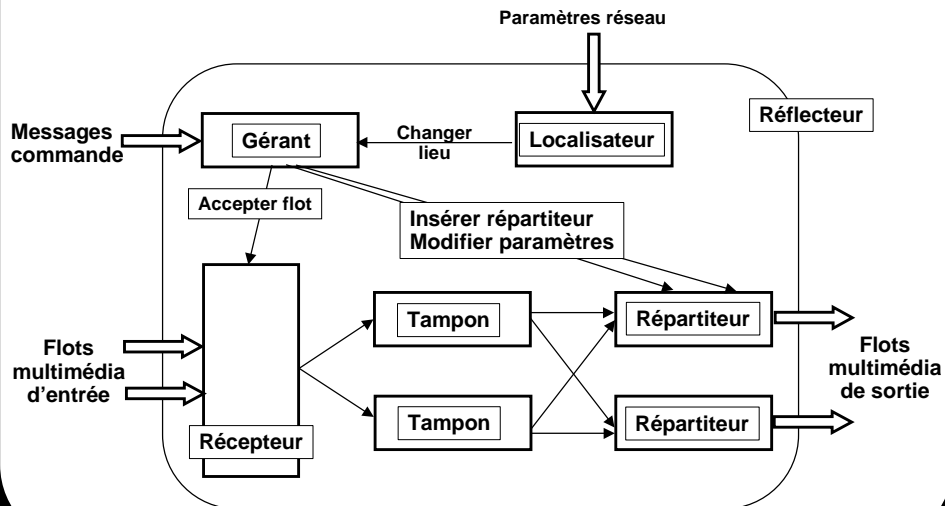
Structure d'un réflecteur (1/2)

- † Récepteurs (1 audio, 1 vidéo)
 - † Reçoit les flots de paquets arrivants et les stocke dans des tampons (1 par participant)
- † Répartiteur
 - † Redistribue les paquets des tampons aux participants
 - † Réalise une politique particulière de distribution
 - † selon la provenance du paquet
 - † selon les capacités du récepteur
 - † sur demande spécifique (transcodage, etc.)
 - † Téléchargeable, plusieurs peuvent coexister

Structure d'un réflecteur (2/2)



Architecture du système de vidéoconférence



Vidéoconférence Commentaires sur l'étude de cas

- † **Adaptation dynamique**
 - † Aux conditions variables d'exploitation
 - † Aux besoins des utilisateurs
 - † À la croissance du système
- † **Techniques pour l'adaptation**
 - † Structure évolutive (duplication ou migration)
 - † Code mobile (+ liaison dynamique)
 - † Mesures de charge

Domaine de recherche actif, expériences en cours.

Caractéristiques et besoins des applications réparties

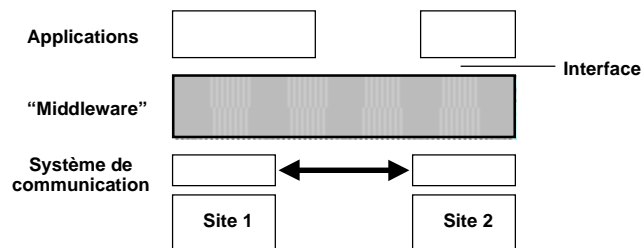
- † Organisation
- † Coordination
- † Communications
- † Sécurité
- † Disponibilité
- † Capacité de croissance
- † Génie logiciel
 - † Construction
 - † Administration

Organisation des applications réparties

- † Client-serveur
 - † Exécution "synchrone" requête-réponse ; base : RPC
 - † Extensions : serveurs coopérants ; service discontinu ; ...
- † Objets partagés (organisations diverses)
- † Flots de communication
 - † Discrets (messages) ou continus (multimédia)
- † Code mobile
 - † Machine virtuelle (masque l'hétérogénéité)
 - † Problèmes de sécurité
- † "Agents"
 - † Programme agissant pour le compte d'une entité cliente
 - † Agents fixes ou mobiles, statiques ou adaptatifs
 - † Coopération entre agents

Organisation des applications réparties Un schéma commun : le "middleware"

- † **"Middleware" : couche de logiciel (réparti) destinée à**
 - † masquer l'hétérogénéité des machines et systèmes
 - † masquer la répartition des traitements et données
 - † fournir une interface commode aux applications (modèle de programmation + API)

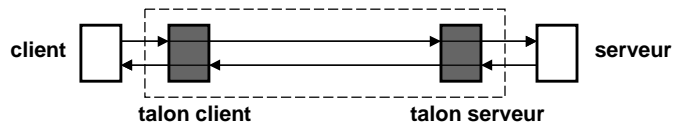


Organisation des applications réparties Importance de la normalisation

- † **Le développement du "middleware" impose une normalisation des interfaces**
 - † Logiciel de base
 - † Domaines spécifiques d'applications
- † **Nombreux consortiums et standards**
 - † Open Group (ex-OSF) : systèmes, outils de base
 - † Web Consortium (W3C) : Web et outils associés
 - † OMG : objets répartis (CORBA, IIOP, etc.)
 - † ODMG : bases de données à objets
 - † ODP : organisation "ouverte" des applications
 - † Workflow Management Coalition : applications à flots de données
 - † ...

Construction d'applications réparties Un outil de structuration : le mandataire (*proxy*)

- ◆ Initialement : réalisation de l'appel de procédure à distance

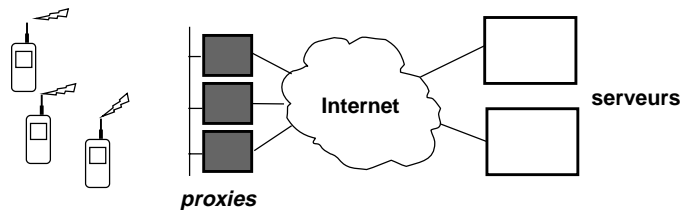


- ◆ Fonction : réalisation de la communication
synchronisation
adaptation de protocole
conversion de données
- ◆ Évolution ultérieure
fonctions propres à l'application
gestion d'objets
sécurité
adaptation à des conditions variables de fonctionnement

Utilisation de *proxy* pour l'adaptation : applications pour clients légers mobiles

Objectif : porter une application répartie complexe sur des clients "légers" (PDAs)

Méthode : reporter les fonctions coûteuses dans un ensemble de *proxies*



- Fonctions des *proxies* :
- filtrage et compression (avec perte) des images
 - filtrage de texte (HTML)
 - agrégation de réponses aux requêtes
 - gestion de caches
- Programmation indépendante du serveur, adaptée aux caractéristiques des clients

Coordination

- † **Coordination : permet aux composants d'une application d'interagir en respectant des règles fixées**
 - † Englobe synchronisation et communication
 - † Divers niveaux d'abstraction
 - † Divers mécanismes élémentaires (messages, événements, etc.)
- † **Caractéristiques de la communication**
 - † Synchrone : schéma prédéfini (ex. : RPC)
 - † Asynchrone : réaction à un événement
 - † Flots discrets ou continus - QoS
- † **Informations partagées**
 - † Espace unique ou multiples
 - † Divers modes de réalisation (centralisé, réparti, dupliqué, ...)

Sécurité

- † **Confidentialité**
- † **Intégrité**
 - † Protection par contrôle des droits d'accès
 - † Isolation (pare-feux)
- † **Authentification, signature électronique**
 - † Identification des partenaires
 - † Non-déni d'envoi ou de réception
 - † Messages authentifiés
 - † Respect possible de l'anonymat
- † **Une méthode de base : la cryptographie**
 - † Aspects techniques
 - † Aspects sociaux, juridiques et législatifs

Disponibilité des informations et des services

- † Séparation entre machines "contrôlables" (serveurs) et non contrôlables (la plupart)
- † Technique de base n°1 : la duplication
 - † des serveurs
 - † des services
 - † des informations
- † Technique de base n°2 : les protocoles de groupe
 - † protocoles d'appartenance
 - † protocoles de diffusion (fiable, causale, atomique)
- † Technique de base n°3 : les transactions

Capacité de croissance

- † Propriétés souhaitables : éviter la dégradation des performances lorsqu'augmentent
 - † le nombre de sites
 - † le nombre d'utilisateurs
 - † le volume de données
 - † la fréquence des interactions
- † Quelques heuristiques
 - † éviter la centralisation des données
 - † éviter les algorithmes nécessitant la participation de tous les sites
 - † déléguer et décentraliser les décisions
- † Un exemple réussi : DNS

Génie logiciel des applications réparties

- † **Réutilisation de l'existant**
 - † Réutilisation en général non prévue à l'avance
- † **Adaptation à l'environnement**
 - † Support matériel (machines et réseaux)
 - † Hétérogénéité
 - † Conditions variables d'utilisation (charge, ressources, etc)
- † **Quelques voies ...**
 - † Méthodes à base d'objets ("composants")
 - † Encapsulation, interfaces séparées
 - † Spécialisation de classes
 - † Interfaces d'administration
 - † Code mobile

Outils et services pour applications réparties

- † **Infrastructures de systèmes**
 - † Support d'objets répartis
 - † Services divers
 - † Gestion de groupes (collaboration, tolérance aux fautes)
 - † Gestion de ressources (QoS, performances)
- † **Outils de développement**
 - † Langages et environnements
 - † Outils de connexion
- † **Outils d'administration**
 - † Description globale
 - † Interfaces d'administration

Plan de l'École

- † **Modèles de structuration des applications réparties**
 - † Client-serveur, bus logiciel, objets répartis, ...
- † **Environnements pour applications réparties**
 - † DCOM-ActiveX
 - † CORBA
 - † Web
 - † Environnements Java (Beans, EJB)
- † **Techniques spécifiques pour applications réparties**
 - † Code et agents mobiles
 - † Composants
 - † Carte à puce (JavaCard)
- † **Modèles de description d'applications réparties**