

Modèle à agents mobiles

Sara Bouchenak

Sara.Bouchenak@inria.fr

<http://sardes.inrialpes.fr/~bouchena>

Exemples de code mobile (1/4)

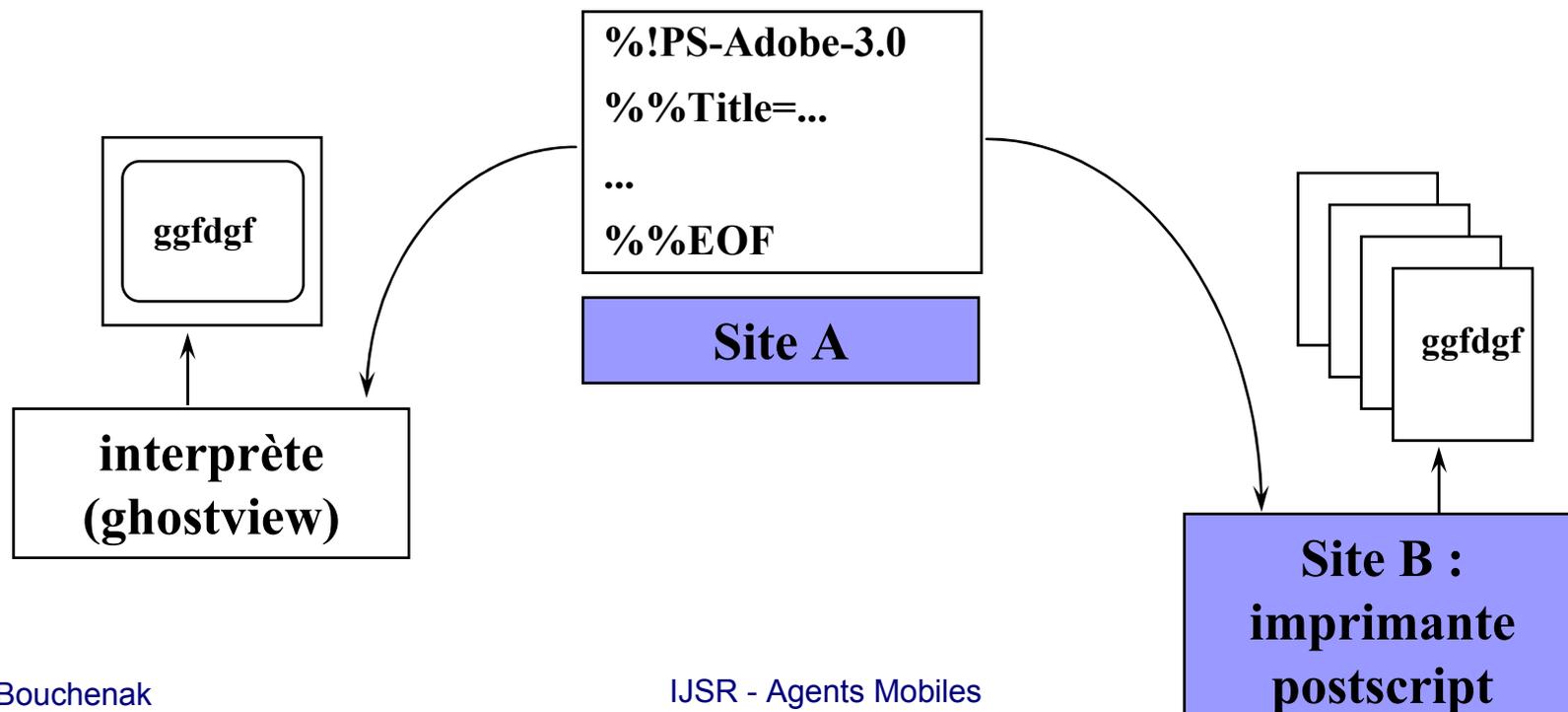
- Code mobile : programme pouvant se déplacer d'un site à un autre sur un réseau

- Exemples :
 - Postscript
 - SQL
 - Applets Java

Exemples de code mobile (2/4)

■ Postscript

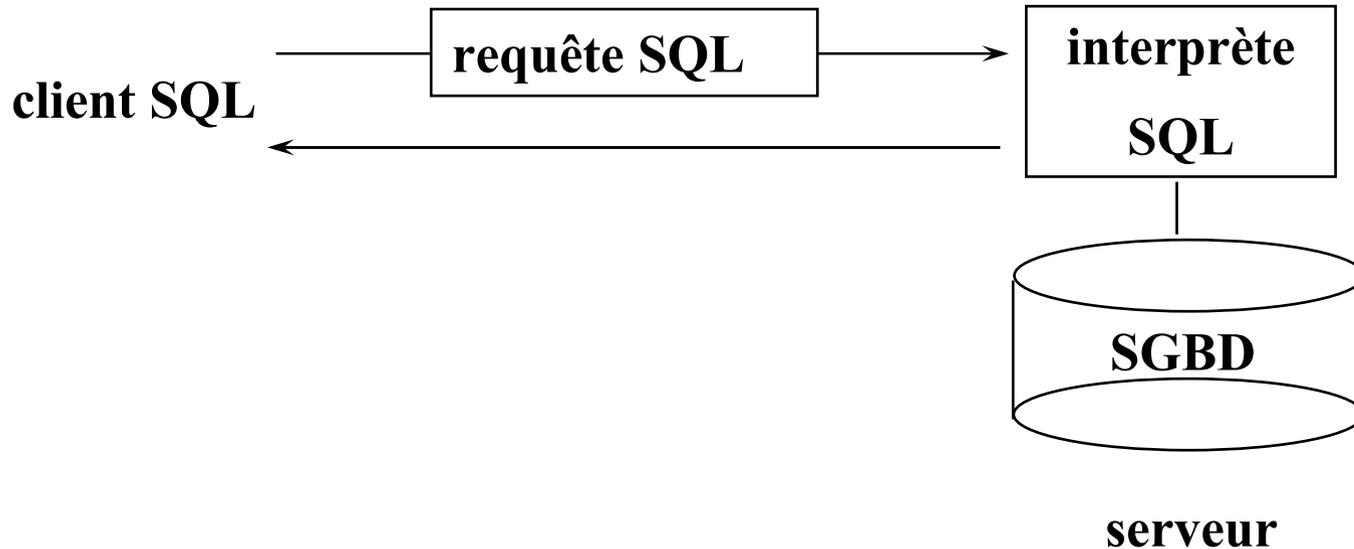
- code exécuté par machine physique ou interprète
- distribue le travail sur différents sites



Exemples de code mobile (3/4)

■ SQL

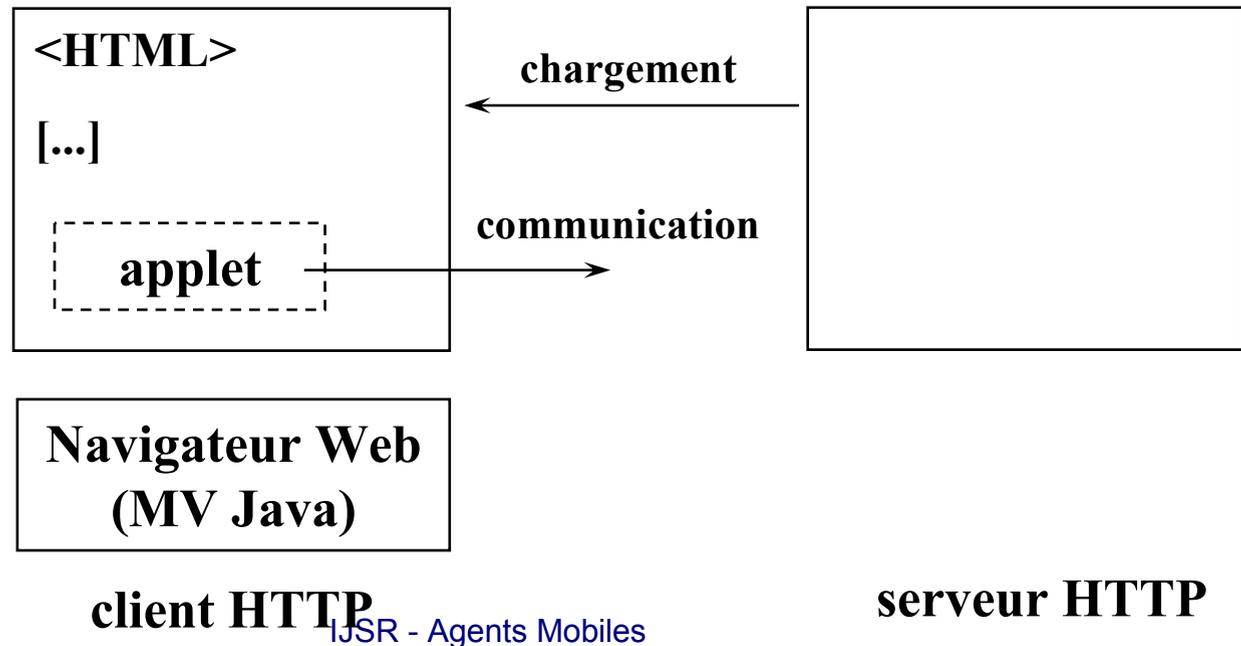
- code exécuté par un interprète SQL
- code requête déplacé vers le serveur de base de données



Exemples de code mobile (4/4)

■ Applets Java

- programme exécutable inclus dans une page HTML
- exécution sur le site qui télécharge la page HTML



Plan

- Agents mobiles : qu'est-ce que c'est et à quoi ça sert?
- Quelques définitions
- Problèmes de mise en œuvre
- Solutions existantes
- Agents mobiles dans Java

Agents mobiles: qu'est-ce que c'est ?

- Agent Mobile : processus, incluant du code et des données, pouvant se déplacer entre des machines pour réaliser une tâche

■ Agent

- objet actif
- communique potentiellement avec d'autres agents

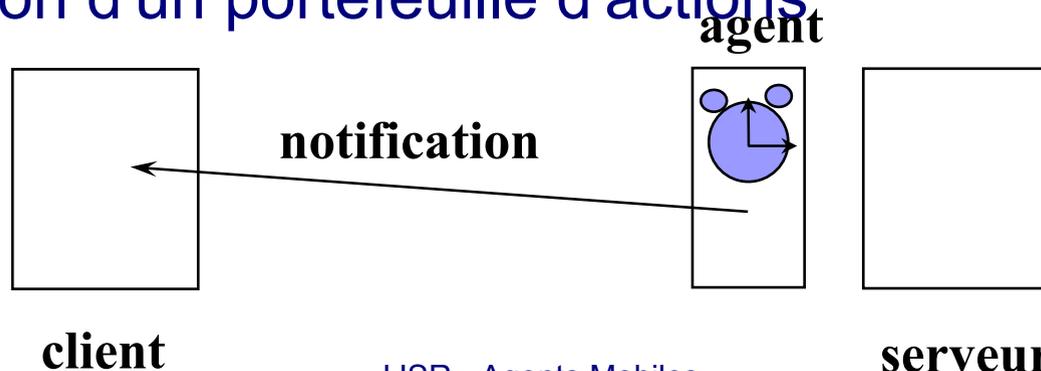
■ Mobile

- se déplace en fonction de ses besoins
- suit parfois un itinéraire

Agents mobiles: à quoi ça sert ? (1/5)

■ Agent notificateur/réactif

- attend une information ou un événement
- prévient un usager ou déclenche une action
- exemples
 - recherche d'annonces dans des journaux
 - gestion d'un portefeuille d'actions



Agents mobiles: à quoi ça sert ? (2/5)

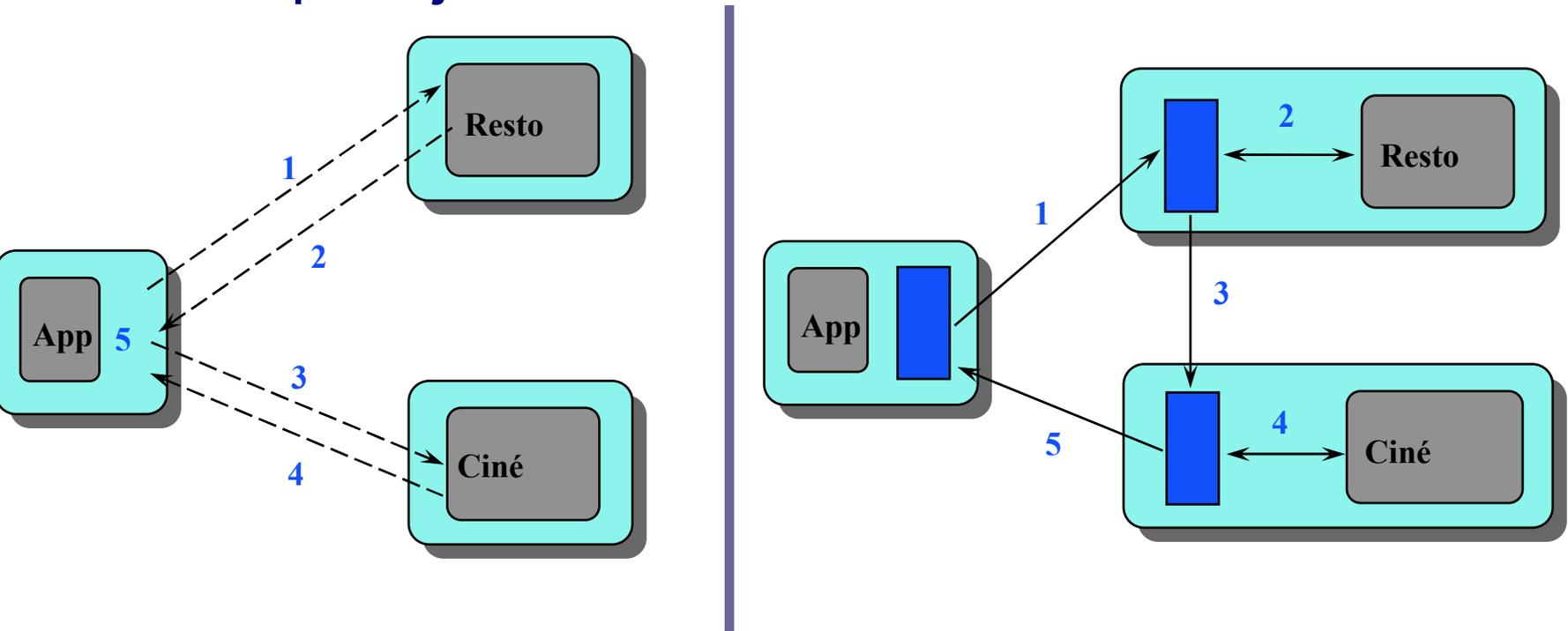
■ Agent itinérant

- réalise une suite d'actions sur des serveurs
- privilégie les accès locaux aux accès distants
- exemple : jointure entre deux sources de données
 - un serveur qui gère des restaurants
 - un serveur qui gère des cinémas
 - recherche des séances de cinéma les plus appropriées aux restaurants sélectionnés

Agents mobiles: à quoi ça sert ? (3/5)

■ Agent itinérant

- exemple : jointure entre deux sources de données



Agents mobiles: à quoi ça sert ? (4/5)

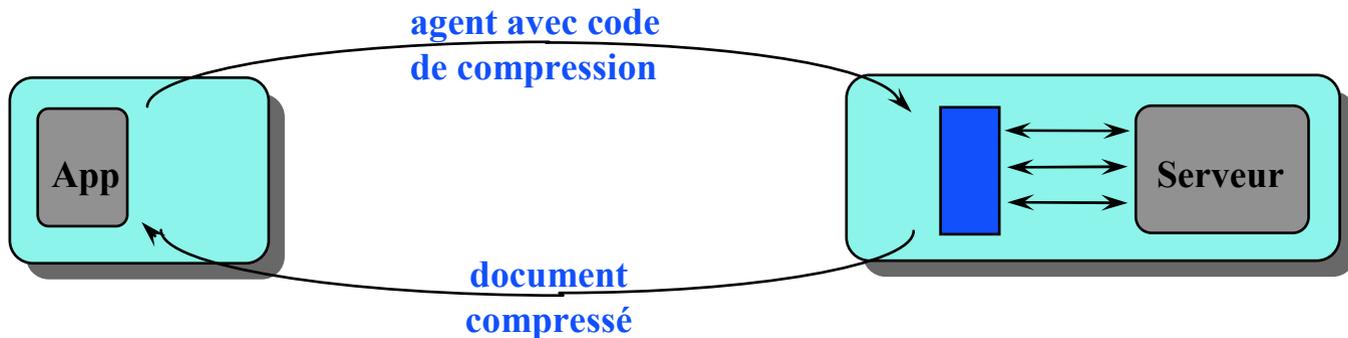
■ Agent d'adaptation

- extension des fonctions du service
- adaptation aux besoins spécifiques des clients
- exemple : compression/chiffrement de documents
 - un client demande un document à un serveur
 - le client transmet un algorithme de compression ou de chiffrement des données
 - le code de cet algorithme est propre au client

Agents mobiles: à quoi ça sert ? (5/5)

■ Agent d'adaptation

- exemple : compression/chiffrement de documents



Plan

- Agents mobiles : qu'est-ce que c'est et à quoi ça sert?
- Quelques définitions
- Problèmes de mise en œuvre
- Solutions existantes
- Agents mobiles dans Java

Quelques définitions (1/3)

■ Agent

- unité de structuration des applications
- ressources allouées à l'agent (contrôlée par l'agent)
- agent mobile ou stationnaire

■ Place

- endroit que peut visiter un agent
- plus fin que le site (contrôle d'accès)
- certaines places prédéfinies pour un agent (ex : *home*)

Quelques définitions (2/3)

■ Déplacement d'un agent

- entre des places
- parfois notion d'itinéraire
- permet de co-localiser des agents pour des interactions locales

■ Interactions entre agents

- entre des agents distants : messages
- entre des agents co-localisés (*meetings*)

Quelques définitions (3/3)

■ Contrôle d'accès

□ autorité

- associée à un agent ou une place
- permet une authentification d'un agent ou d'une place

□ permis

- des capacités associées à des agents ou à des places
- permet de limiter des droits d'accès

Plan

- Agents mobiles : qu'est-ce que c'est et à quoi ça sert?
- Quelques définitions
- Problèmes de mise en œuvre
- Solutions existantes
- Agents mobiles dans Java

Problèmes de mise en œuvre (1/3)

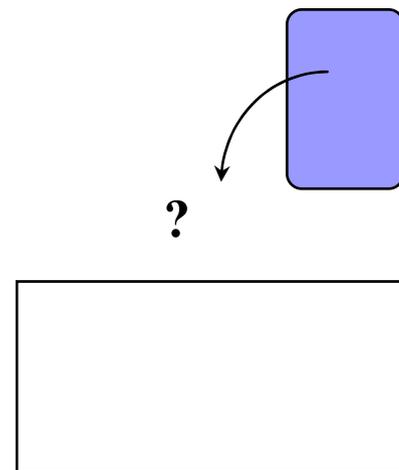
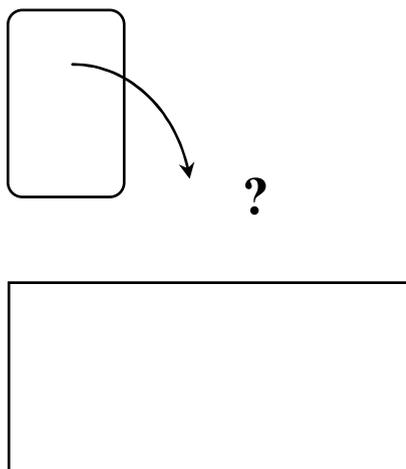
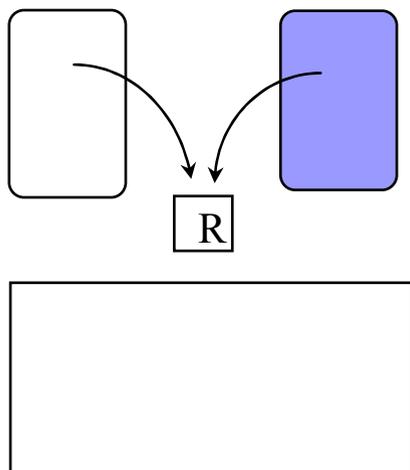
■ Mobilité

- hétérogénéité du code, des données et du système
 - restriction à un environnement homogène
 - gestion par une machine virtuelle
- capture des données d'un agent
 - délimiter l'ensemble des objets à déplacer avec l'agent
- capture de l'état d'exécution
 - migration faible
 - migration forte

Problèmes de mise en œuvre (2/3)

■ Ressources partagées

- référence à distance (lien de poursuite)
- copie (mise en cohérence)
- liaison par équivalence



Problèmes de mise en œuvre (3/3)

■ Sécurité

- attaque d'un hôte par un agent mobile
 - confinement de l'exécution
 - par le matériel
 - par le logiciel
 - authentification de l'agent pour délimiter ses droits d'accès
- attaque d'un agent par un hôte
 - idée : brouillage de code
 - A veut faire calculer $F(x)$ sur le site B sans révéler F à B (F linéaire)
 - A calcule $Z = S.F$ (S matrice inversible choisie par A)
 - A envoie Z à B et B calcule $y = Z(x)$ et envoie y à A
 - A calcule $S^{-1}(y) = F(x)$

Plan

- Agents mobiles : qu'est-ce que c'est et à quoi ça sert?
- Quelques définitions
- Problèmes de mise en œuvre
- Solutions existantes
- Agents mobiles dans Java

Solutions existantes (1/3)

■ Sur Java

- Aglets (IBM)
 - <http://www.trl.ibm.co.jp/aglets>
- Odyssey (General Magic Inc.)
 - http://genmagic.com/technology/mobile_agent.html
- Concordia(Mitsubishi)
 - <http://www.meitca.com/HSL/Projects/Concordia>
- Voyager (Object Space)
 - <http://www.objectspace.com/voyager>
- MOA (OSF/OpenGroup)
 - <http://www.osf.org/RI/java/moa>

Solutions existantes (2/3)

- Autres environnements (Tcl, Python ...)
 - AgentTcl (Dartmouth College)
 - <http://www.cs.dartmouth.edu/~agent>
 - Ara (Université de Kaiserslautern)
 - <http://www.uni-kl.de/AG-Nehmer/Projekte/Ara>
 - Tacoma (Université de Tromsø et Cornell)
 - <http://www.cs.uit.no/DOS/Tacoma>

Solutions existantes : limitations (3/3)

- La mobilité faible impose une programmation fastidieuse
- La localisation des agents reste un problème difficile sur un réseau à grande échelle.
 - doit-elle être entièrement prise en charge par le système ?
- Peu de retour sur l'intérêt de la technologie
 - stratégique : intérêt certain, mais peu de grandes applications

Plan

- Agents mobiles : qu'est-ce que c'est et à quoi ça sert?
- Quelques définitions
- Problèmes de mise en œuvre
- Solutions existantes
- Agents mobiles dans Java

Agents mobiles dans Java (1/3)

■ Avantages de Java

- Diffusion sur toutes les machines (homogénéité)
- Mobilité du code et des objets
 - chargement dynamique de classes (ClassLoader)
 - sérialisation des objets Java
- Sécurité assurée par l'environnement
 - confinement : pas d'adresses virtuelles
 - contrôle d'accès aux ressources (SecurityManager)

Agents mobiles dans Java (2/3)

■ Site de départ

- Envoi du code de l'agent
- Sérialisation et envoi de l'état (données/objets) de l'agent
- Destruction de l'agent sur le site origine

■ Site d'arrivée

- Réception et chargement du code de l'agent
- Dé-sérialisation et réception de l'état de l'agent
- Création d'un *thread*
- Lancement de l'exécution de l'agent par le thread créé

Agents mobiles dans Java (3/3)

- Conclusion: plate-forme *Mobilet* (*let)
 - agents mobiles : nouveau modèle basé sur la mobilité
 - Java fournit les outils pour le support d'agents mobiles
 - intéressant pour des réseaux lents ou lors de déconnexions
- Limites et perspectives
 - transparence de la programmation
 - migration faible/forte
 - utilisation de ressources partagées
 - intégration avec d'autres modèles

Quelques références

- **Mobility: Processes, Computers, and Agents.** D. Milojić, F. Douglass, Wheeler, Addison-Wesley, 1999.
- **Mobile Agents.** G. P. Picco, Proceedings of the 5th IEEE Int. Conference on Mobile Agents (MA'2001), Lecture Notes in Computer Science, Volume 2240, Atlanta, Georgia, USA, December 2001
<http://www.cs.dartmouth.edu/MA2001/>
- **Mobile Agents: are they a good idea?** D. Chess, C. Harrison, Kershenbaum, IBM Research Report, RC 19887, March 1999
<http://www.research.ibm.com/iagents/publications.html>