Mobile Agents

Introduction

- *Mobile agent* is a *distributed computing* paradigm.

- It has become viable, with recent technologies such as those provided by Java.

- It has great potential for network applications.

- It has **not** been widely deployed.

Mobile (transportable) agents

- An **agent** is "*an independent software program, which runs on behalf of a network user*".

- A mobile agent is a program that, once it is launched by a **user**, can travel from node to node *autonomously*, and can continue to function even if the user is disconnected from the network.
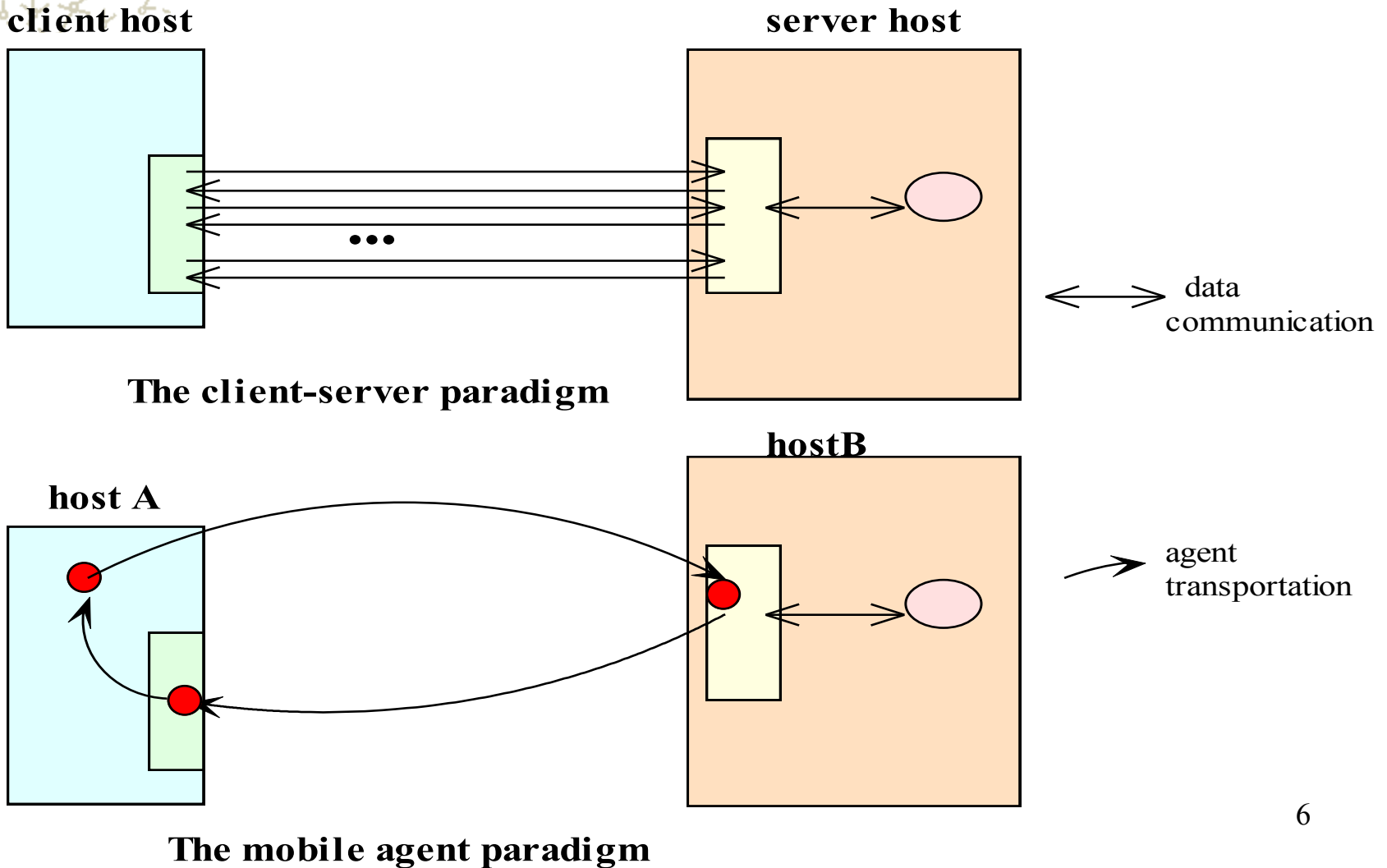
user

File server

Agent server

Database server

Agent Server
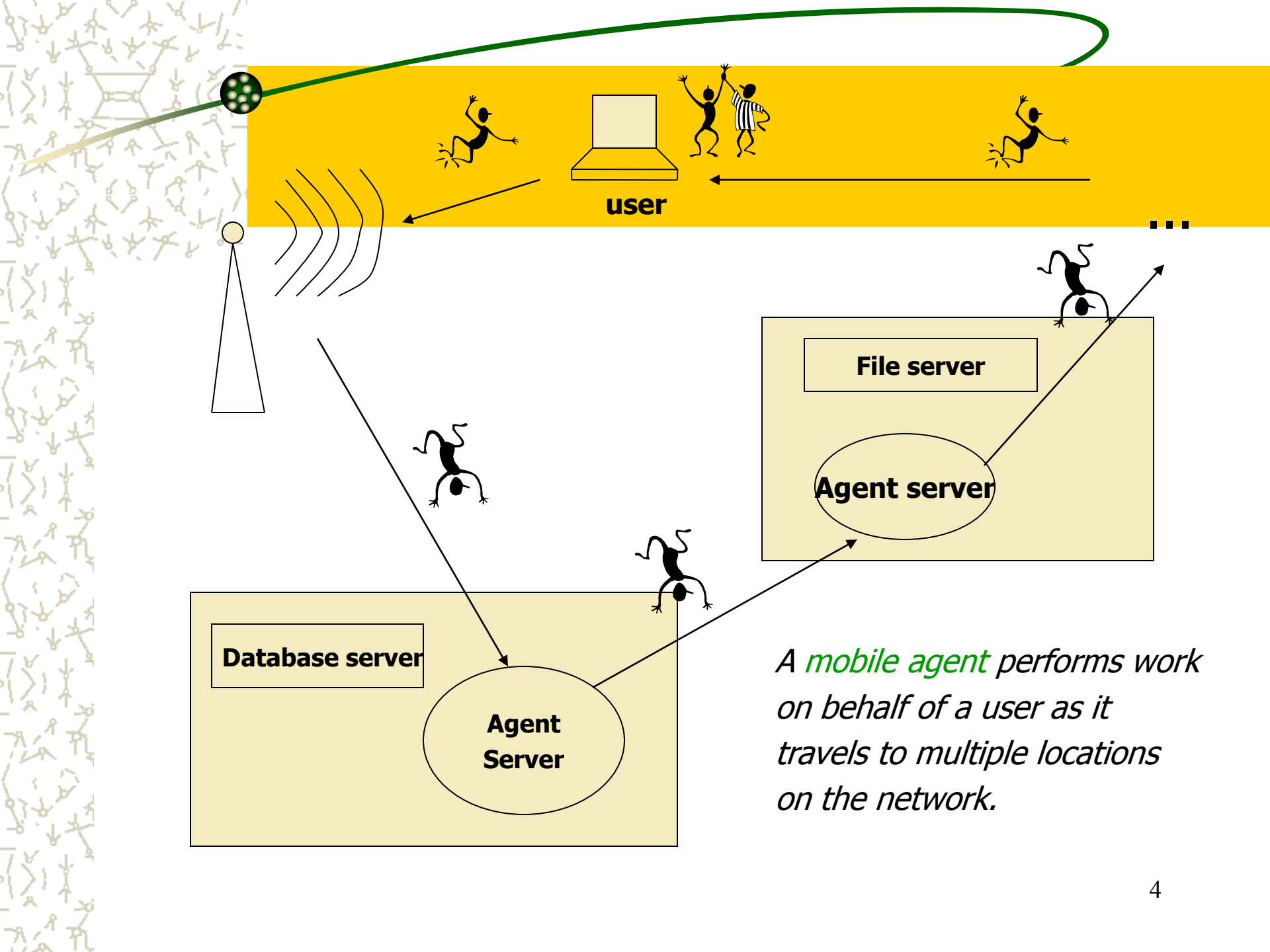
A *mobile agent* performs work on behalf of a user as it travels to multiple locations on the network.

4

Advantages of Mobile Agents

- They allow efficient and economical use of communication channels that may have **low bandwidth**, **high latency**, and may be error-prone.

- They enable the use of **portable**, **low-cost**, personal communications devices to perform complex tasks even when the device is disconnected from the network.

- They allow **asynchronous** operations and true **decentralization**

client host

server host

The client-server paradigm

$\longleftrightarrow$ data communication

hostB

host A

$\longrightarrow$ agent transportation

The mobile agent paradigm

6

Basic Architecture

- An agent server process runs on **each participating host**.

- Participating hosts are networked through links that can be low-bandwidth and unreliable.

- **An agent is a serializable object** whose **execution state** can be frozen for transportation and reconstituted upon arrival at a remote site.

Basic Architecture

mobile agent

agent
server

agent server
host
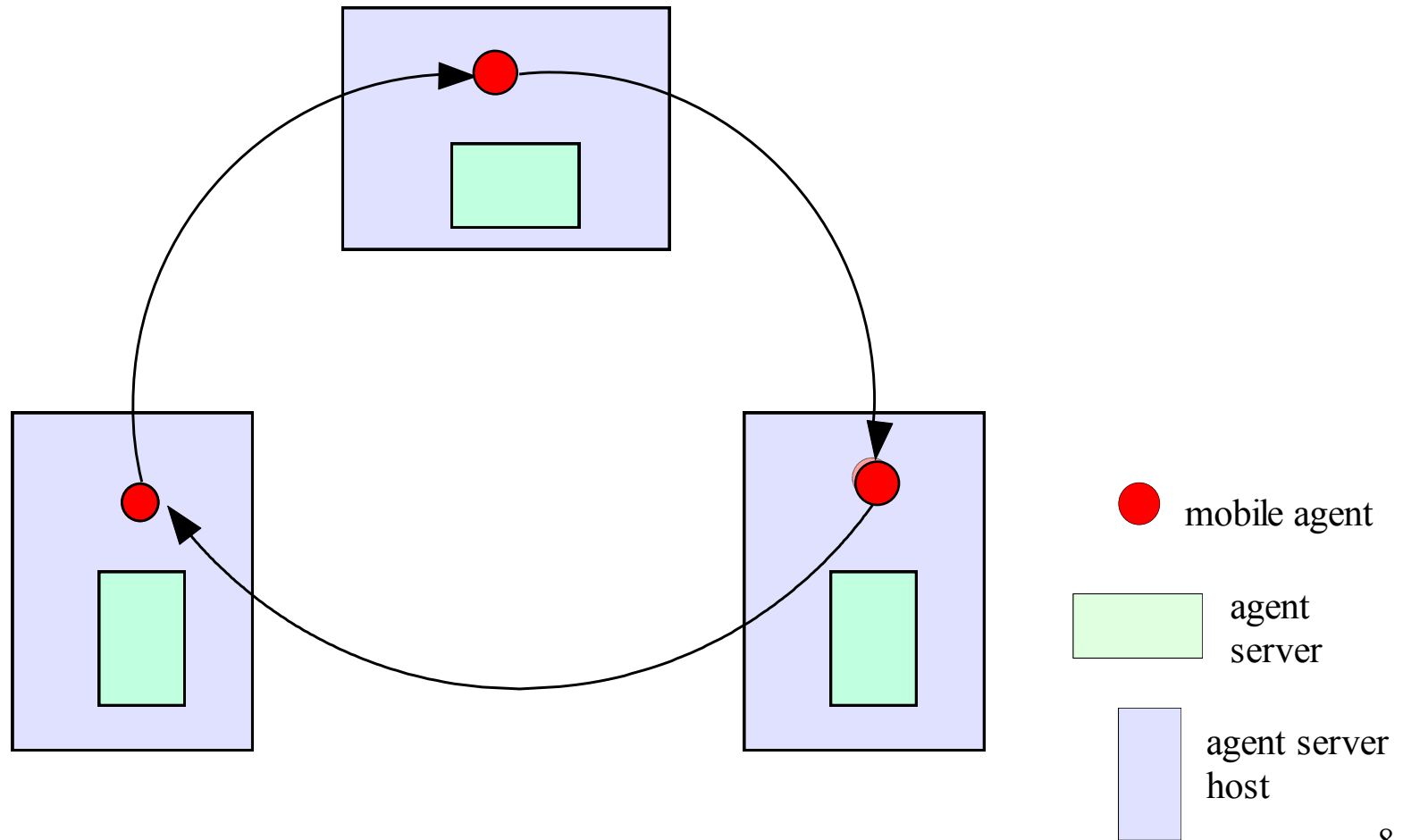
What's in the Agent?

- An agent is an *object*, hence it contains *state data* and *methods*.

- Among the instance data is an *itinerary* of the sites to be visited, which may be *dynamically constructed* or *adjusted*.

- Other data may include an **agent ID** or other **authentication data**.

- The **agent's behavior** at each stop can be **pre-programmed** and **dynamically adjusted**.

A simple demo

- See the mobileAgents folder in the program samples.

- RMI is used to implement the agent, the server, and the agent-launching client.

- The agent is a serialized object with an "execute" method, which is invoked by a **server** that receives the agent.

Mobile-agent applications

- Information retrieval
- Monitoring
- Virtual market-place/ meeting room
- Shareware
- Personal Mobile Agent white paper,
  http://www.x-fetch.com/common/X-Fetch_Personal_Mobile_Agent_White_Paper.pdf
- **IEEE Network Magazine** special issue on Applicability of Mobile Agents to Telecommunications, May-June 2002

11

Security in Mobile Agent Systems

Security concern is the primary deterrent of deploying the mobile-agent technology.

There are **concerns** for both the agent hosts and the mobile agents.

- Agent host concerns:

    Malicious/unauthorized agents can misuse/destroy system resources (e.g., worms).

- Agent concerns:

    Malicious hosts can destroy or alter an agent's logic, (e.g., Mobile agent's route can be altered.)

Security in Mobile Agent Systems

http://mole.informatik.uni-stuttgart.de/security.html

Measures:

- **Authentication** – an agent must authenticate itself to the host, and an agent server must authenticate itself to the agent.

- **Encryption** – an agent encrypts its sensitive data.

- **Resource access** – a host enforces strict access control to its resources.

Mobile-agent framework systems

- **_Using RMI_** to implement a mobile agent application is not generally recommended
  (**http://developer.java.sun.com/developer/onlineTraining/rmi/RMI.html - MobileAgentArchitectures**)

- "The solution to the mobile computing agent using RMI is, at best, a work-around. Other distributed Java architectures have been designed to address security concerns and other issues.

- These are collectively called **_mobile agent architectures_**. Some examples are IBM's **Aglets Architecture** and **ObjectSpace's Voyager System**. These systems are specifically designed to allow and support the movement of Java objects between JVMs, carrying their data along with their execution instructions."

Existing Mobile-agent framework system

✦ IBM Aglet: http://www.trl.ibm.com/aglets/

✦ Mitsubishi Concordia

- http://www.concordiaagents.com/
- white paper

The Mobile Agent System Interoperability Facility (MASIF)

- From the **OMG** (The Object Management Group): "**Mobile agent platforms** have been developed, built on top of **different operating systems**, based on **different programming languages** and **technologies**.

- Even new languages have been realized, exclusively designed for the support of mobile agents.

- However, common trends can be noticed: **Interpreter-based programming languages** like **Java build the basis for most of today's agent platforms**, and **several approaches are associated with the integration** of mobile agents and **RPC-based middleware** like CORBA."

MASIF

Several fundamental requirements have been identified due to experiences that have been made during research and development activities. These requirements cover the following topics:

- **Management Support**
- **Security Support**
- **Mobility Support**
- **Support for Unique Identification**
- **Transaction Support**
- **Communication Support**

MASIF

- Due to the considerations mentioned above, the OMG issued a Request for Proposal (Common Facilities RFP3) for a mobile agent standard in November 1995.

- The corresponding Mobile Agent System Interoperability Facility (MASIF) submission, developed by Crystaliz, General Magic, GMD FOKUS, IBM, and The Open Group, has been adopted by the OMG in February 1998.

MASIF

- The idea behind the MASIF standard is to achieve a certain degree of interoperability between mobile agent platforms of different manufacturers without enforcing radical platform modifications.

- MASIF is not intended to build the basis for any new mobile agent platform. Instead, the provided specifications shall be used as an "add-on" to already **existing systems**.

MASIF

The following list comprises the mandatory requirements that were identified within the MASIF RFP:

- Marshalling and un-marshalling of agent programs
- Encoding of agent containers for transport
- Transport of agents from one agent facility (i.e. execution engine) to another
- Runtime registration and invocation of agent facilities
- Runtime query of a named agent facility by agents
- Runtime security of agents

Ongoing Research

- D'Agents: Mobile Agents at Dartmouth College – overview; position paper
- The MAP system (Italy)
- Gypsy (Austria)
- Grasshopper (Germany)

Sources of Information

- **Mobile Agents Introductory**

  http://www.infosys.tuwien.ac.at/Research/Agents/intro.html

- **The Mobile Agent List**

  http://mole.informatik.uni-stuttgart.de/mal/mal.html

- **Mobile Agent Applications**

  http://www.computer.org/concurrency/pd1999/pdf/p3080.pdf

- **Software Engineering Concerns for Mobile Agent Systems**

  http://www.elet.polimi.it/Users/DEI/Sections/Compeng/GianPietro.Picco/ICSE01mobility/papers/cook.pdf