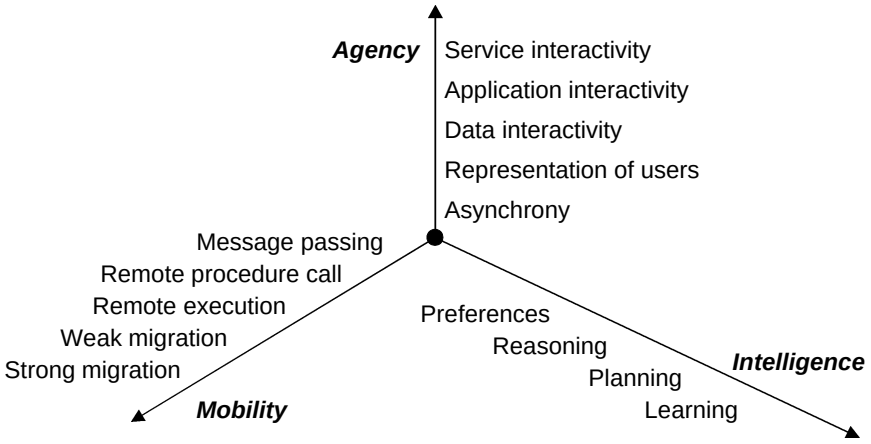
Mobile Agents

Haiping Xu
Computer Science Department
The University of Illinois at Chicago

Introduction

- The term agent comes from greek “agein”, which means to drive or to lead.
- Today the term agent denotes something that producing an effect, e.g., drying agent, a shipping agent.
- It is suitable to describe current trends in computer science: active instruments (to which work can be delegated) vs. passive tools.
- The term agent in computer science refers to software agent.

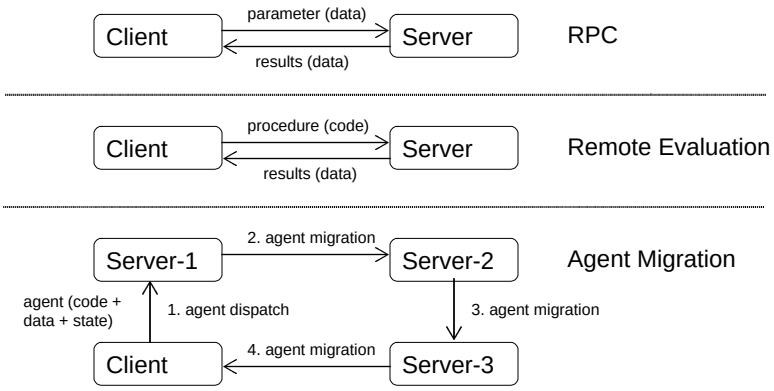
Space of Software Agents



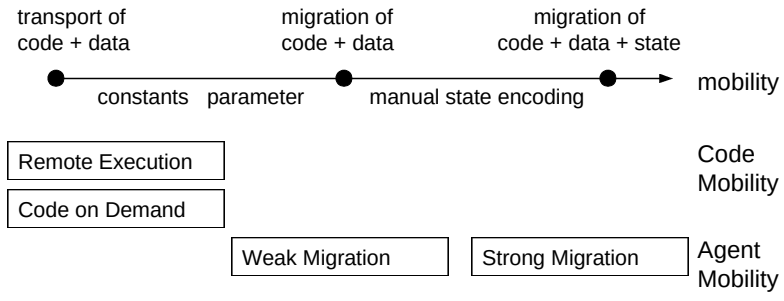
Current Researches on Agents

- Do not exploit all the capabilities classified by these three dimensions.
- Multi-agent systems (MAS)
  - Execute a given task.
  - Use distributed but static agents.
  - Collaborate and cooperate in an intelligent manner.
- Mobile agents (MA)
  - Model agent mobility and agent coordination.
  - Assume very limited or even no intelligence.

Evolution of the Mobile Agent Paradigm



Degrees of Mobility



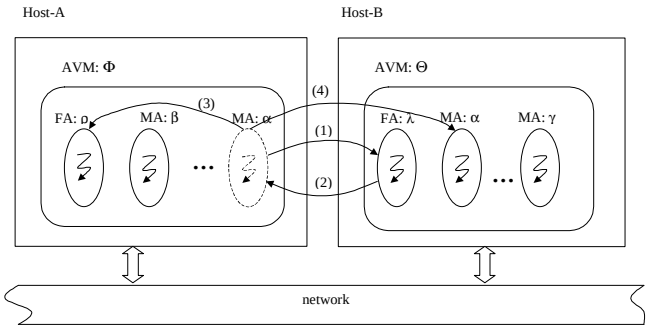
Why Mobile Agents ?

- Software-Distribution on Demand
  - Easy to transport code and install packages automatically.
  - Code mobility simplify the management of an existing structure.
- Reduction of Communication Costs
  - The number of interactions.
  - The amount of data communicated over the network.

Why Mobile Agents ? (continue)

- Asynchronous Tasks
  - Asynchronous processing of requests.
  - Mobile device can be disconnected and reconnected.
- Scalability Due to Dynamic Deployment
  - A hierarchy of mobile agents can be set up.
  - The structure of agent hierarchy can change dynamically.

A Mobile Agent System



(1) move-request, (2) grant, (3) notify, (4) move

Agent world architecture and an example of agent migration

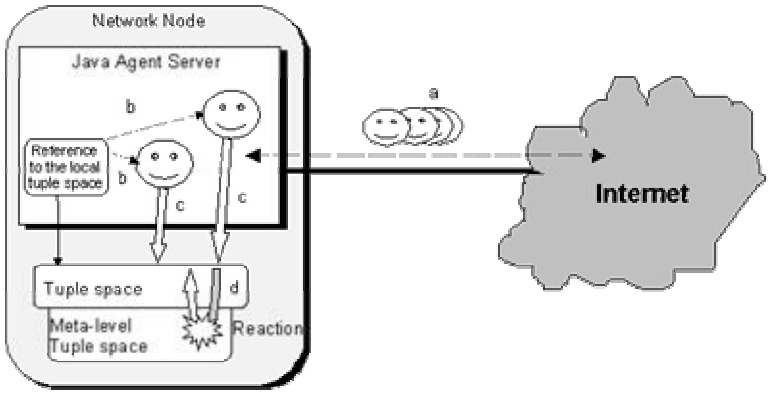
Examples of Mobile Agent Systems

- Mole (University of Stuttgart)
  - First Java-based of a mobile agent system.
  - Use Java as the agent programming as well as the implementation language.
- Aglets (IBM)
  - An aglet is a mobile java object and corresponds to mobile agents.
  - Support synchronous and asynchronous message passing.
  - Agent mobility is implemented by weak migration.

Examples of Mobile Agent Systems (continue)

- Agent Tcl (Dartmouth College)
  - An extension of the Tool Command Language (Tcl), which is a scripting language on UNIX.
  - Implements strong mobility.
- MARS (Univ. of Modena and Reggio Emilia)
  - An architecture for mobile agent coordination.
  - Based on a reactive tuple space model.
  - The tuple space is programmable.

The MARS Architecture



Academic Research Work

- Model agent mobility (physical vs. logical)
  - Distributed join-calculus: an extension of π-calculus that introduce the explicit notions of named localities and distributed failure.
  - Mobile UNITY: a programming notation that captures the notion of mobility and transient interaction among mobile nodes.
  - MobiS: an extended version of PoliS, which is a specification language based multiple tuple spaces.
  - LIME: a middleware based on tuple spaces.

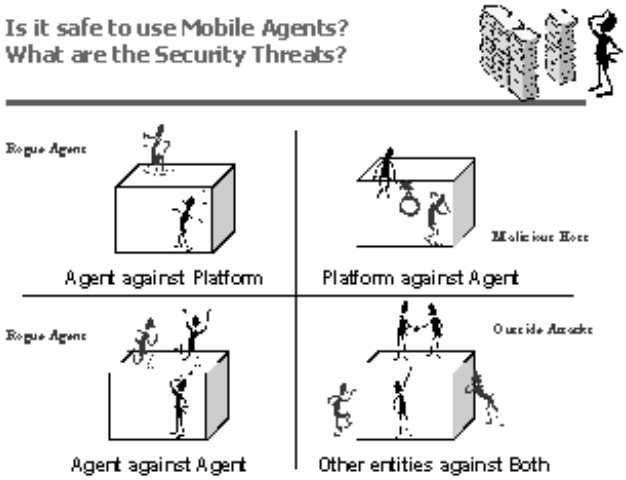
Academic Research Work (continue)

- Model agent communication
  - Knowledge Query Manipulation Language (KQML)
  - Foundation for Intelligent Physical Agents (FIPA)
  - Mobile Agent System Interoperability Facility (MASIF)
- Model agent coordination
  - Inter-agent coordination vs. agent-environment coordination.
  - Example: context-dependent coordination in MARS.

Challenges

- Security
  - Four areas: (1) inter-agent security (2) agent-host security (3) inter-host security (4) hosts and unauthorized third party.
  - Agent-host security:
    - Protect hosts from malicious (visiting) agents.
    - Protect agents from malicious hosts.

Challenges (continue)



Challenges (continue)

- Control structures
  - Primitives: create, clone, and terminate agents.
  - To terminate agents is more complicated in the context of a hierarchy of agents.
- Transactional support
  - To guarantee that the agent is performed exactly once, independent of communication and node failures.
  - Upon failure, agent states must be made recoverable.

My Current Research Work

- Proposed a general model for intelligent software agents.
  - Explicitly model asynchronous message passing.
  - Introduce inheritance mechanism into agent-oriented software design.
  - Exercise behavioral analysis and verification.
- Model intelligent mobile agents (IMA).
  - Introduce mobility into agent-oriented software model.
  - Provide a framework for intelligent mobile agent.

References

- N. R. Jennings, K. Sycara and M. Wooldridge, “A Roadmap of Agent Research and Development,” *International Journal of Autonomous Agents and Multi-Agent Systems*, 1(1): 7-38, 1998.
- K. Rothermel and M. Schwehm, “Mobile Agents,” In: A. Kent and J. G. Williams (Eds.): *Encyclopedia for Computer Science and Technology*, Volume 40 - Supplement 25, New York: M. Dekker Inc., 1999, pp. 155-176.
- N. M. Karnik and A. R. Tripathi, “Design Issues in Mobile-Agent Programming Systems,” *IEEE Concurrency*, July-September 1998.
- Tim Finin, Yannis Labrou and Yun Peng, “Mobile Agents can Benefit from Standards Efforts in Inter-agent Communication,” *IEEE Communications Magazine*, Vol. 36, No. 7, pp. 50-56, July 1998.
- G. Cabri, L. Leonardi, F. Zambonelli, “Engineering Mobile-Agent Applications via Context-dependent Coordination,” In *Proceedings of the 23rd International Conference on Software Engineering (ICSE 2001)*, Toronto, Canada, 2001, pp.371-380.

Website for Mobile Agent Systems

- Mole (University of Stuttgart)
  - http://mole.informatik.uni-stuttgart.de/
- Aglets (IBM)
  - http://www.trl.ibm.com/aglets/
- Agent Tcl (Dartmouth College)
  - http://agent.cs.dartmouth.edu/
- MARS (Univ. of Modena and Reggio Emilia)
  - http://sirio.dsi.unimo.it/MOON/MARS/index.html