

MobilityTools

A Toolbox for Agent Mobility and Interoperability Based on OMG Standards

bruno.dillenseger@francetelecom.fr

France Télécom R&D (ex-CNET), BP98, F-38243 Meylan CEDEX

Why MobiliTools?

Usual drawbacks of mobile agent platforms:

- Mobile agent platforms usually enforce an agent framework with a dedicated mix of activity model and communication model;
- Unfortunately, there is no universal agent framework meeting every requirement (scalability, reactivity, robustness, reliability, pro-activity, adaptability...);
- Standards are still lacking for making heterogeneous agent inter-operate (at both model and implementation levels).

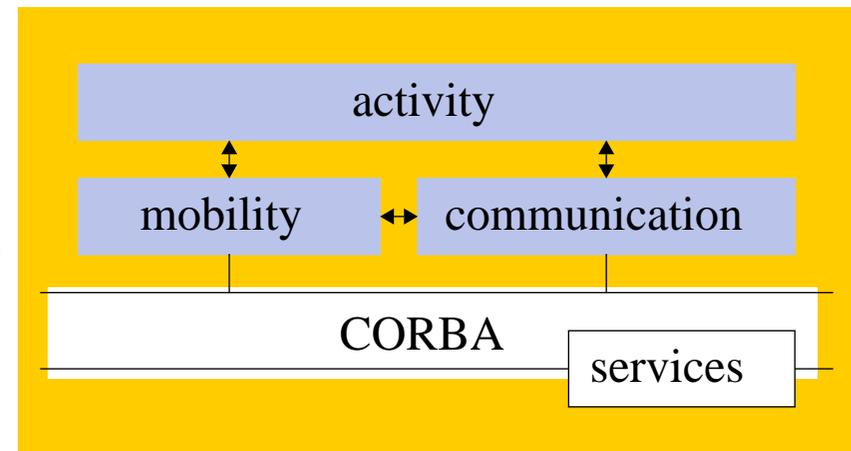
MobiliTools specific approach

Middleware approach

Use distributed system standards (OMG) to help interoperability;

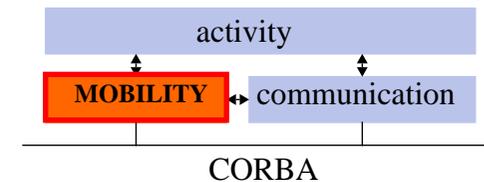
Modular approach

Separate mobility, communication and activity support into independent generic/customizable components to provide a flexible toolbox.

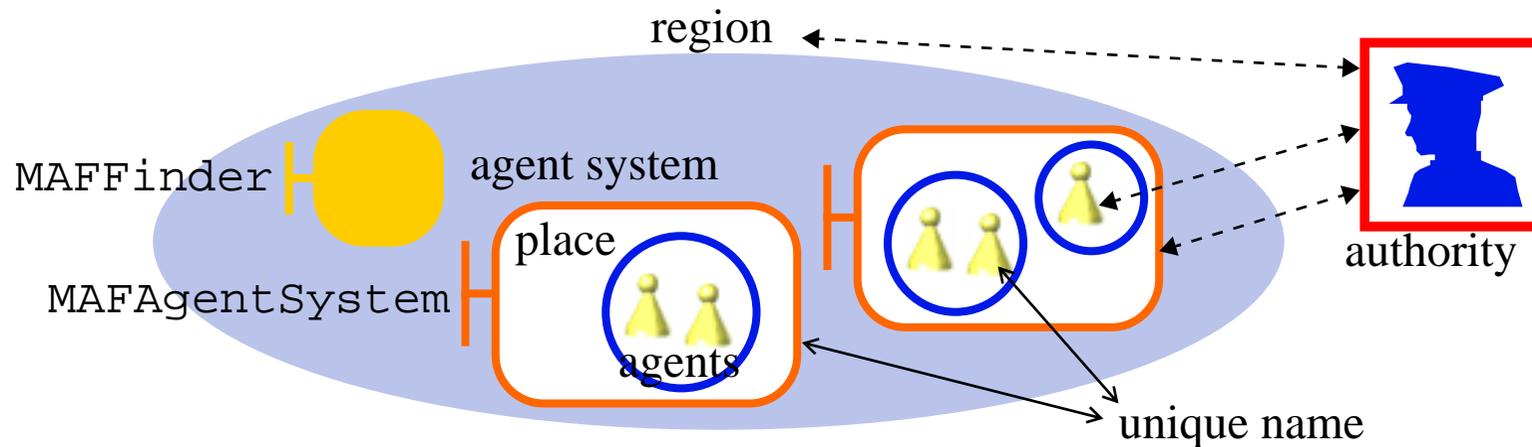


MobiliTools = extended middleware for mobile objects.

Mobility support with SMI



SMI implements OMG's Mobile Agent System Interoperability Facilities specifications



- interface `MAFFinder` registers and looks up agents.
- interface `MAFAgentSystem` manages an agent execution environment and local agents;

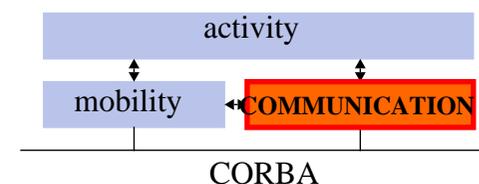
Simple MASIF Implementation

Simple and generic

- classes `Agency` and `Finder` implement MASIF's interfaces and provides a customizable Java based environment for agent execution and management;
- No dedicated agent framework but MASIF's concepts and a generic life cycle defined by interface `MobileObject`: `afterBirth`, `beforeMove`, `afterMove`, `afterMoveFailed`, `beforeSuspend`, `beforeResume`, `beforeDeath`...

MASIF interfaces have been slightly completed with a couple of operations to enhance management features.

Communication support with ACTS



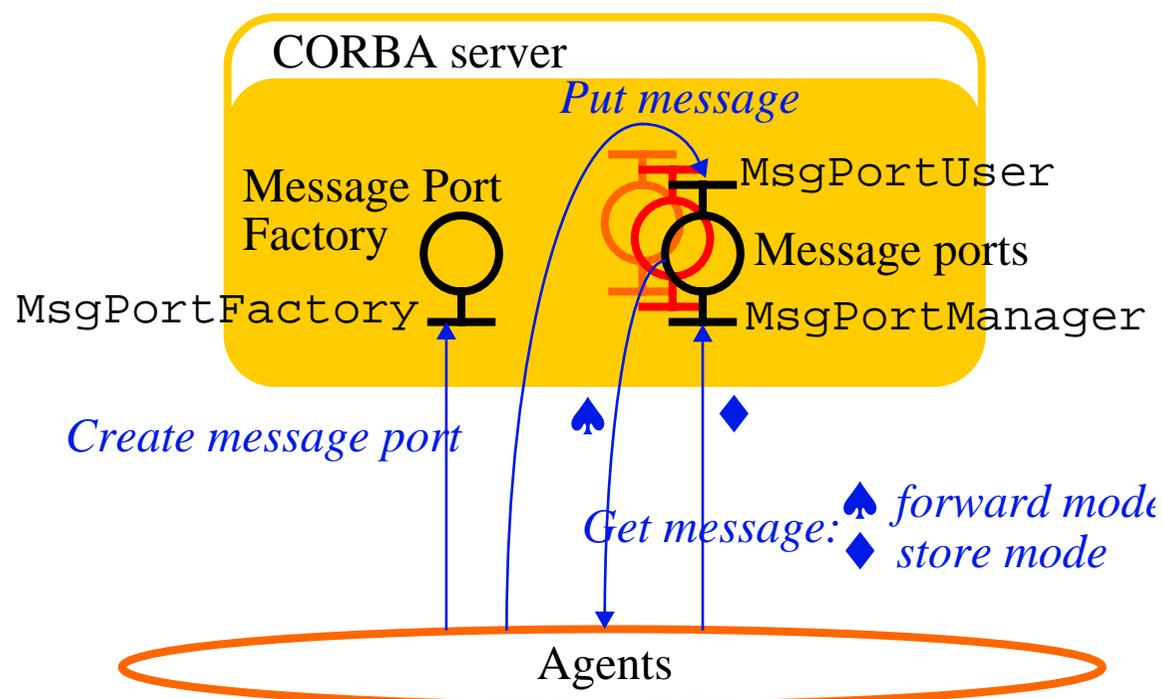
Why a CORBA service for Agent Communication Transport?

Issue	ACTS answer
MASIF doesn't address agent communication interoperability.	ACTS defines interfaces to transport communication between heterogeneous agents via CORBA.
Agent mobility can make communication delivery dramatically complex.	ACTS's pragmatic approach is to spread an infrastructure of static message buffers to store messages. Mobility awareness makes it reliable.
There is a variety of agent communication styles.	ACTS is highly versatile and customizable: "any" type messages, both store and forward delivery modes.
Should agents using ACTS be CORBA objects?	Only the forward mode requires agents to be, or at least to provide, CORBA objects.

Agent Communication Transport Service

How does it work?

- static *message ports* store (MsgPortUser interface) and deliver (MsgPortManager interface) messages;
- *message port factories* create message ports (MsgPortFactory interface) on demand.



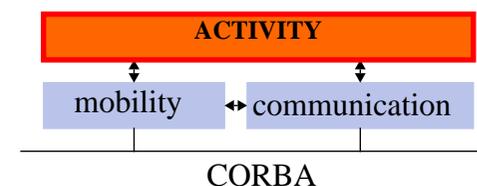
ACTS customization

Our Mailbox and FIPA'98 “personalities” show advanced utilizations of ACTS for transporting agent communication.

For example, ACTS Mailbox personality wraps message ports and CORBA interfaces to provide:

- user-friendly Java mailbox objects, with high-level addresses in-line with MASIF's region concept (agent_name@region);
- multicast and unicast features;
- mobility-transparent message forwarding management.

What about agent activity?



Agent activity management has to be defined by the application programmer through the implementation of SMI's MobileObject interface.

⇒ apply SMI's generic lifecycle to one particular agent model.
(e.g. assign one autonomous thread of activity to each agent)

Activity may also be tightly coupled with communication.

e.g. event-driven "reactive" agents, synchronous programming model.

Availability

Requirements

Java 2; CORBA-compliant ORB: e.g. Java2, open source Jonathan/David (<http://www.objectweb.org/>).

acts.jar = 88K, smi.jar = 195K, smi+acts.jar = 256K (incl. GUI tools)

Licence

MobiliTools belongs to France Télécom R&D.

It is available **free** of charge for **research** purpose on signed official demand (ask bruno.dillenseger@francetelecom.fr for details).

Conclusion

MobiliTools provides a set of INDEPENDENT and CUSTOMIZABLE Java tools for mobility support, based on OMG STANDARDS:

- SMI for object/agent mobility, based on MASIF — interoperability for agent creation and mobility is not fulfilled.
- ACTS for (mobile) object/agent communication interoperability.

On-going and future plans include agent activity engines for SMI (e.g. synchronous programming model), miscellaneous ACTS and SMI enhancements (... and interoperability investigations?)