Jim White, General Magic, 11 March 1996
jim_white@genmagic.com

Submitted to the Joint W3C/OMG Workshop on
Distributed Objects and Mobile Code

In the last five years General Magic has developed and deployed Telescript, a technology for mobile agents. This paper sketches a common agent platform intended to allow any number of such technologies to coexist and interoperate.

Agent

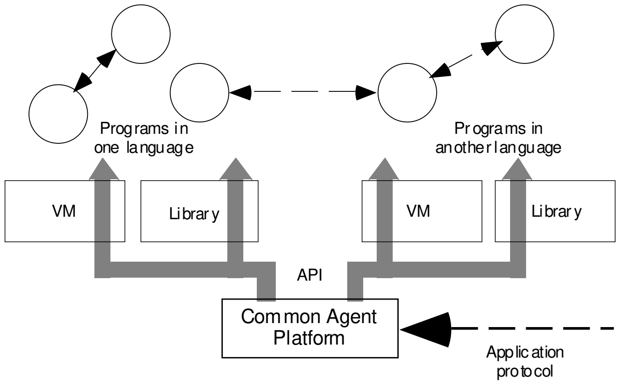
An agent is a program that a person or organization vests with its authority, that can run unattended for a long time (e.g., a week), and that can meet and interact with other agents. The person or organization is the agent's authority.

One can distinguish two kinds of agent. A stationary agent executes on one computer system. A mobile agent can execute on different computer systems at different times of its life.

Agent platform

Agents will be written in a variety of programming languages. This creates a need for a common agent platform, a language-independent means for agent interaction and mobile agent migration. A server's implementation of the platform might offer a full range of services; a PC's might offer a subset. An application protocol would interconnect the implementations of different systems so that their agents can interact and so that their mobile agents can migrate between systems.

The platform would offer agent services to its clients. These clients would include language environments, the virtual machines and libraries for particular programming languages. A language environment would offer some or all of the agent services to programs written in that language.



Agent services

The platform could provide services like these:

Program naming. Assign names to agent programs so as to distinguish one from another.

- Program authentication. Authenticate the implementors of agent programs.
- Program migration. Transport agent programs between computers. The name of a program might be used to detect that it was already present at its destination.
- Agent naming. Assign names to agents so as to distinguish one from another.
- Agent authentication. Authenticate the authorities of agents.
- Agent migration. Transport agents between computers. The mobile agent's destination might be specified indirectly as the computer that hosts a specified stationary agent.
- Agent permissions. Assign names to permissions, enforce an agent's assigned permissions, and let one agent grant permissions to other agents or re-negotiate its own.
- Agent relationships. Assign names to relationships, let agents begin and end relationships, and enforce rules of conduct for relationships.
- Agent creation. Let agents create other agents locally and remotely. A new agent might have the authority of the existing agent and either the same permissions or a subset of them.
- Agent termination. Terminate agents in an orderly fashion, e.g., by notifying other agents with which they have relationships.
- Agent waiting. Write to disk agents that must wait for long periods of time for events to occur (e.g., for relationships to be established).
- Agent checkpointing. Checkpoint agents to disk so that they survive crashes of their host systems. A transaction service might coordinate the checkpointing of related agents.
- Agent interaction. Let related agents interact. The means of interaction (e.g., events, pipes, and RPC) might depend upon whether the agents occupy the same or different systems.
- Agent management. Let network management and debugging agents monitor other agents. A debugging agent, e.g., might be able to suspend and resume agents under development.