Agent Mobility

Overview

mobile agents
  and mobile computing

technical issues
  agent languages, distributed execution,
  environment, security

multi-agent systems
  cooperation between agents to solve a task
Mobile Agents

emphasis on electronic agents

**purpose**
release the user from mundane tasks

**approach**
utilize autonomous, mobile programs ("agents")

**advantages**
dealing with information overload, increased efficiency, better results

**problems**
new technology, loss of control, security

mobility is clearly important for autonomous robots, but the emphasis here is on software agents
Mobile Computing

vs. mobile agents

mobile computing equipment
laptops, hand-held computers

usage
not stationary, but usually off-line (not connected to other computers or networks)

limitations
activities requiring network access

potential solution
mobile agents perform the requested activities while the user is off-line, and report the results when the user reconnects
Distributed Applications

execution of subtasks on different machines

distributed programming
distribution-aware implementation

distributed operating systems
provide essential services like task allocation,
load balancing, remote procedure calls, . . .

network services
communication, synchronization

transportation infrastructure
LAN, WAN, Internet, Intranet
provides the physical interconnection between
the agent’s starting and end points
Mobile Code Systems

general architecture

user interface
  communication between agent and user

agent execution environment
  "living space" for agents on computers

services
  local and mobility services
  basic functions provided for the execution and movement of agents
Mobility Services

main obstacle to pervasive use of mobile agents

generic mobility module
  provides most of the support various types of mobile agents need
  good for agent designers
  difficult to implement
  somewhat rigid: extensions need to be compatible with the full module, and changes may affect the infrastructure as a whole

minimal mobility modules
  specific modules for different types of agents
  provide only the minimum support needed
  good for infrastructure providers
  development more difficult for agent designers
  more flexible: extensions can be implemented on top of minimal services
Agent Implementation Languages

cross-platform execution

platform independent
  often converted into an abstract instruction set
  (virtual machine), such as Java, Tcl, Telescript

standard set of services
  libraries, CGI, ActiveX, SOAP

user interface
  generic user interface capabilities
  Java AWT/Swing, Tcl/Tk
Host Security

is it a virus or an agent?

**alien code problem**
- remote host has to execute unknown code

**authentication**
- agents must carry identification and authentication information
- possibly third-party certification

**“padded cell” security**
- isolation layer between the code to be executed and the sensitive parts of the system

**permissions**
- access restrictions for certain activities and types of agents

some of these measures are not technical, but organizational (policies)
Agent Security

is the agent safe out there?

internal workings
should not be fully accessible to foreign hosts

valuables
agents may carry electronic cash, copyrighted materials, important data, . . .
agents must be protected from robbery

shared resources
agents may be prevented from utilizing resources by other careless, greedy, or malicious agents

destruction
agents’ lives must be protected accidents or deliberate destruction
Agent and Resource Identification

Who are you?

agent identification
agents must be identifiable and distinguishable from one another
owner, origins of an agent

resource identification
uniform way to identify and access agent-specific resources
independent of the underlying platform
example: URI (uniform resource identifier) in XML

inter-agent communication
communication protocol
communication language
arrangement of (virtual) meetings between agents
exchange of information
name space conventions
  uniform or at least compatible naming schemes
  for agents and resources
Resource Control

competition for scarce resources
   CPU time, memory, data base access, network connections, bandwidth, . . .

permissions and restrictions
   priorities for agents
   restrictions on operations

remunerations
   agents pay for the utilization of resources

consumption limits
   agents have only a certain amount of currency to spend on resources

complex and difficult task, but very important
Programming Support

program development
  specific requirements and constraints due to the
  mobile and distributed nature
  platform-independence, behavior in systems
  under load, vulnerability

program execution
  the agent’s execution may be temporally and
  spatially inaccessible to the owner
  monitoring, exception handling, incomplete
  execution

remote control
  steering of an agent’s activities
  cancellation of a task, modification, requests
  from hosts visited by the agent
Efficiency

**costs of code mobility**
preparation, packaging, transfer of an agent
authentication, setup of the environment,
execution of the agent’s code

**niches** for agents
in the near future, agents may be restricted to
specific applications: more complex than
client-server or Web-based applications, but
limited by infrastructure, complexity

**scalability**
worldwide use may imply millions of agents
popular services may be hit by thousands of
agents simultaneously