

Locate and Track

When Location Matters



copyright 2000 GemStone Systems, Inc.
All rights reserved.

Locate and Track

The Future of Distributed Computing

- Many more IP nodes than People
- More machine-to-machine than machine-to-people communication (agents work for other agents)
- Highly dynamic relationships between agents
- Agents with physical attributes
 - Computing has physical side-effects
 - Physical action has computing side-effects
- Entity lifetimes vary wildly



How Might Agents Interact?

- Computationally: across some network topology
 - Synchronizing actions is affected by latency
 - Predicting effects requires consistency
- Physically: across some distance in space
 - Exchanging objects requires propinquity
 - Coordinating movements requires orientation



Examples of Interaction

- Agents communicate locally
 - Routing agent in vehicle gets conditions from local traffic agents
 - Mobile browser gets local point of interest information and ads
- Agents interact physically
 - Robot crew handles and moves objects



More Examples

- Agents are controlled remotely in realtime
 - Teleoperation by human operator
 - Subagents control parts of realtime processes (e.g., demolition of structures)



Moreover, Locations can Change

- Vehicle moves outside of agent's local area
- Communications channel changes latency
- Agent separation exceeds limits of physical contact



Forces

- Availability - client must be able to find service based on location, distance, orientation, or communication channel characteristics
- Restriction of service - service must be able to refuse request if location of client invalidates service constraints
- Termination of service - client and service must be able to recognize if location invalidates continued use of service



Problem:

- Location may matter in the relationships among agents, so location-transparency may not be desirable

Solution:

- Make location one of the criteria for discovery of services, and one of the criteria for maintaining a lease

Note that location may not be just a quality-of-service issue: a service in the wrong place may not be usable at all by a particular client.



Alternative Solutions:

- Record locations of services in a central registry
 - Doesn't scale to millions or billions of services
 - May not be able to update location fast enough for clients to make correct decisions
 - Lose other benefits of distributed architecture
- Build location into higher-level protocols
 - Lower performance may impair tracking
 - Makes discovery and lease-management harder to implement



Known Uses of the Pattern

- IP routing (uses distributed routing tables)
- Cellular communication location and tracking (locality determined by signal strength)
- Wireless network roaming (locality determined by signal strength)
- Shipping container tracking (locality determined manually or automatically as container moves)



Benefits

- Filtering - Client can find a properly-located service without searching through many that are obviously unsuitable
- Decentralization - Obviates the need for a central location registry
- Simplifies Application - Location code is independent of application behavior
- Flexible Negotiation - Location behavior can be different for instances of the same service, and can vary with time



Liabilities

- Overhead - May require extra message traffic, especially for tracking
- Complicated implementation - Lease handling is more complicated, which may bubble up to client or service code



Conclusion

- If location is a factor, the nature of distribution and the topology of the network as seen by clients and services changes
- Just as there are situations where remoteness can't be hidden, so there are situations where location can't be ignored
- Adding location as a basic concept to service discovery can enable a new class of large-scale, highly dynamic agent orientations

