Nancy Lynch, Idit Keidar, (MIT) Kiyoshi Ueda (NTT)

Project Overview

- Developing models, analysis methods for distributed systems, focusing on cooperative group activities in networks.
- Agent communication, group communication.
- Dynamic:
- Participants come and go, change location.
- Network topology changes, components fail and recover.
- Implementations are complex; difficult to build, understand, and analyze.
- Addressing problems using formal modeling, verification methods
- Input/output automaton, process algebraic, knowledge-based methods.
- Extending, combining methods.
- Applying method to case studies, in computer-supported cooperative work, e-commerce, distributed databases.

Nancy Lynch, Idit Keidar, (MIT) Kiyoshi Ueda (NTT)

Progress Through December 1999

- Agents:
  - Dynamic I/O automaton (DIOA) model
    - Extends I/O automata with automaton creation and destruction.
  - Travel agent case study
    - Simple e-commerce example
    - Using dynamic I/O automata, NePi2 process algebra, knowledge-based programs.
- Group Communication
  - Comprehensive set of specifications for guarantees of GC services.
  - Formal design for virtually synchronous group multicast service.
  - Uses client-server architecture
    - Membership managed by dedicated membership servers.
    - Multicast implemented at clients.
  - Scalable group membership algorithm for WANs.
  - Dynamic configuration service.

Research Plan for the Next Six Months

- Travel agent case study
  - State correctness, performance properties, carry out analysis.
  - Evaluate, compare, combine the three methods used.
- Further develop the dynamic I/O automaton (DIOA) model.
  - Include timing-dependence, liveness.
  - Support for modeling mobility.
- Fault-tolerant agent programming case study.
- Implement our group communication algorithms, tune for performance.
- Consider alternative forms of middleware support for cooperative computing in dynamic WANs.
  - For applications that require consistency, like shared white-board, shared text editor.
  - E.g., totally ordered multicast services that preserve QoS.