Large-Scale Systems: Globe

- Supporting Large-Scale Wide-Area Applications
  - –Uniform Model for Distributed Computing
  - –Flexible Implementation Framework
  - –Worldwide Scalability
- The Globe Object Model: Distributed Shared Objects
- Example: Scalable World-Wide Web Objects
- *Reading:*
  - *– M. van Steen, P. Homburg, A. S. Tanenbaum. "The Architectural Design of Globe: A Wide-Area Distributed System"* **(ftp://ftp.cs.vu.nl/pub/papers/globe/IR-422.97.ps.Z)**

Distribution Transparency

- *Access Transparency*
  - ... hides differences in data representation and invocation mechanisms
- *Failure Transparency*
  - ... hides failure and possible recovery of objects
- *Location Transparency*
  - ... hides where object resides
- *Migration Transparency*
  - ... hides from an object the ability of a system to change that object's location
- *Relocation Transparency*
  - ... hides from a client the ability of a system to change the location of an object to which the client is bound
- *Replication Transparency*
  - ... hides the fact that an object or its state may be replicated and that replicas reside at different locations
- *Persistence Transparency*
  - ... hides the fact that an object may be (partly) passivated by the system
- *Transaction Transparency*
  - ... hides the coordination of activities between objects to achieve consistency at a higher level

Visibility of Underlying Resources

- How to deal with different computers, operating systems, networks.
  - Differences should be transparent to achieve portability.
  - What about use of special-purpose resources (parallel machines, multimedia workstations, ...)
- Likewise, underlying network should not be entirely hidden.
  - Why not make applications/framework resource aware?
- Example: multicast service for shared whiteboard vs. video multicast
  - reliable vs. non-reliable
  - small multicast group vs. large group
  - many-to-many vs. one-to-many
- Provide simple service, with easy-to-configure extensions.

Globe Object Model: Distributed Shared Objects

- Globe objects are passive, with one or more interfaces.
- Multiple processes may simultaneously access same object.
- Changes to object's state made by one process are madae visible to the others.
- Objects are physically distributed: state is partitioned and replicated across multiple machines.
- Processes are unaware of this: operations and state are encapsulated by object.
- Implementation aspects (communication protocols, replication strategies, distribution and migration of state) are part of object and hidden.