Implementing the OMG Real-Time CORBA 1.0 Specification in TAO

Marina Spivak, Irfan Pyarali, and Chris Gill
Center for Distributed Object Computing
Washington University, St. Louis, MO

Monday December 11, 2000
Real-Time CORBA Priority Models

- **PriorityModelPolicy**
  - Determines priority at which requests are serviced

- **SERVER_DEPRECATED**
  - Server handles requests at the priority declared at the time of object creation

- **CLIENT_PROPAGATED**
  - Request is executed at the priority requested by client (priority encoded as part of client request)
POA Thread Pools

Standard API for creating and configuring multiple threads for servicing CORBA requests

- Different levels of service
- Overlapping computation and I/O

Interface RTORB {...
ThreadpoolId
create_threadpool_with_lanes
(in unsigned long stacksize, in ThreadpoolLanes lanes, ...);
Both server side and client side policies supported
- Some policies control protocol selection, others configuration
- Order of protocols indicates protocol preference
- Some policies are exported to client in object reference

Interface TCPProtocolProperties:

```plaintext
ProtocolProperties
{
    attribute long send_buffer_size;
    attribute long recv_buffer_size;
    attribute boolean keep_alive;
    attribute boolean dont_route;
    attribute boolean no_delay;
};
```
Other Real-Time CORBA Features

Implemented:

• Priority banded connections and explicit binding
  – Allow end-to-end priority preservation, reduce jitter
• Private connections – reservation prevents multiplexing
• Priority transforms
• Standard synchronizers

In progress:

• Optional global scheduling service

Some issues:

• Request buffering and thread borrowing
Use Case: Avionics Mission Computing

- Mission Computing System
  - Display updates
    - low priority
  - Control events, e.g., actuators
    - high priority
- End-to-end priority preservation required
  - Thread pools with lanes
  - CLIENT_PROPAGATED priority model
  - Explicit binding
  - Priority banded connections
Concluding Remarks

- Real-Time CORBA 1.0 introduces important facilities for applications with QoS requirements
- We expect that such new facilities will increase interest and applicability on CORBA for distributed real-time systems
- This work directly contributes to advancing the use of middleware in real-time distributed systems by implementing the specification in an open-source ORB, and providing feedback to the OMG