The Design and Performance of Real-time CORBA Event Services

Carlos O’Ryan

June, 2001
Motivation: Applying CORBA to Real-time Avionics

• Typical interactions:
  – I/O arrives
  – Proxies demarshasal data
  – Facades process data

• Advantages
  – Anonymous consumers and suppliers
  – Asynchronous event delivery
  – Centralized filtering and scheduling
Motivation: Applying CORBA to Distributed Interactive Simulation

- Typical interactions:
  - Large number of hosts
  - Widely distributed
  - Large number of events
- Why CORBA?
  - Reduced development costs
  - Standards based
Limitations of Traditional Event Services

- High Latency:
  - Two network traversals per message
- Poor scalability:
  - Event Service host must process all requests
- No support for Real-time
  - No filtering
  - No support for priorities
  - Implementation can have unbounded priority inversions
Extensions to Support Real-time Requirements

- Extend event types to support filtering
  - Consumers declare types of interest
  - Suppliers declare event types published
- Consumers can register for periodic events
- Dispatching is done in priority order
- Internal data structures avoid unbounded priority inversions
Using Federations to Reduce Latency and Improve Scalability

- Multiple Event Services are connected to for a Federation
  - Minimal latency for collocated consumer-supplier pairs
  - Worst case latency: single network traversal
  - Improved scalability due to reduced network traffic
- Replacing IIOP with a multicast protocol can improve scalability even further
Empirical Results: Testbed

- Hardware configuration:
  - Dual Pentium III @ 700Mhz
  - 256Kb Cache
  - 512Mb RAM
  - 100BaseT NIC and Hub

- Software configuration
  - TAO v.1.1.15
  - gcc-2.95.2 (-O3, static)
  - RedHat Linux 6.2 (2.2.14 kernel)
Event Channel Latency

- Main results:
  - Latency 784 usecs
  - Jitter 48 usecs
  - 99.9% below 3380 usecs
Identifying Sources of Overhead

- Contrast with
  - TCP/IP
  - CORBA Requests
  - CORBA Callbacks

<table>
<thead>
<tr>
<th></th>
<th>Latency (usecs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP/IP</td>
<td>224</td>
</tr>
<tr>
<td>CORBA</td>
<td>315</td>
</tr>
<tr>
<td>Callback</td>
<td>641</td>
</tr>
<tr>
<td>Event Service</td>
<td>784</td>
</tr>
</tbody>
</table>
The roundtrip latency is comparable to a normal event channel (803 usecs).

However the oneway latency is significantly smaller:
- Estimated 650 usecs.
Concluding Remarks

- The Real-time properties of a CORBA Event Service are largely an implementation detail
  - But the interface lacks a number of fundamental features
  - Both the Notification Service and our extensions overcome those problems
- CORBA Event Services can minimize network overhead using:
  - Federations
  - Multicast protocols
- More information available from
  - http://doc.ece.uci.edu/~coryan/EC/