

# ECOOP 2000 Workshop on Quality of Service in Distributed Object Systems

<http://www.vsb.informatik.uni-frankfurt.de/misc/QoS DOS>

June 13, 2000  
Cannes, France

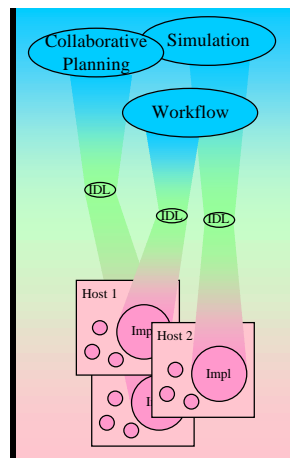
**Christian Becker**

becker@informatik.uni-frankfurt.de

**John Zinky**

JZinky@bbn.com

## Distributed Object Middleware has Emerged to Solve Heterogeneity and Distribution Problems



Applications

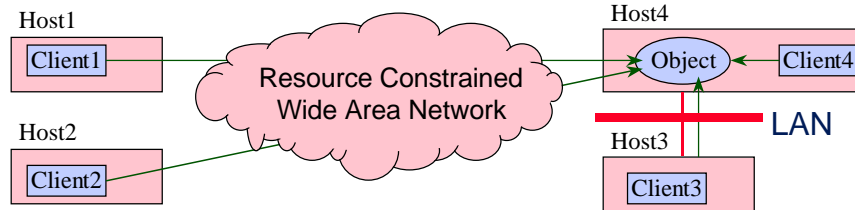
Middleware makes programming distributed applications easier

- Standard programming interfaces hide platform and system dependencies
- Standard protocols, e.g., message formats, allow applications on different systems to interoperate
- Middleware provides higher level, application oriented programming building blocks

Distributed Object Middleware

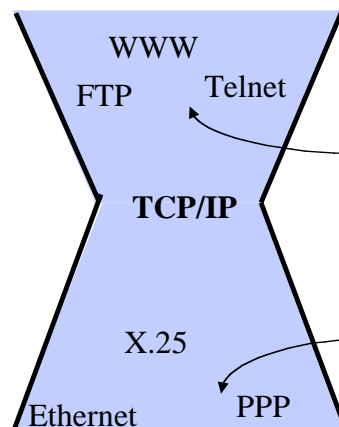
Hosts/Systems

## The Problem: Wide-Area Distributed Applications Are (Still) Hard to Build and Maintain



- WANs resources are dynamic, unpredictable, and unreliable
- Different user communities have different usage patterns
- Quality of Service (QoS) requirements change
- Administrative constraints on resources change
- The same function can be done in multiple ways that use different resource mix

## The Simple Abstraction of TCP/IP Networks Allowed Networks and Applications to Grow Independently

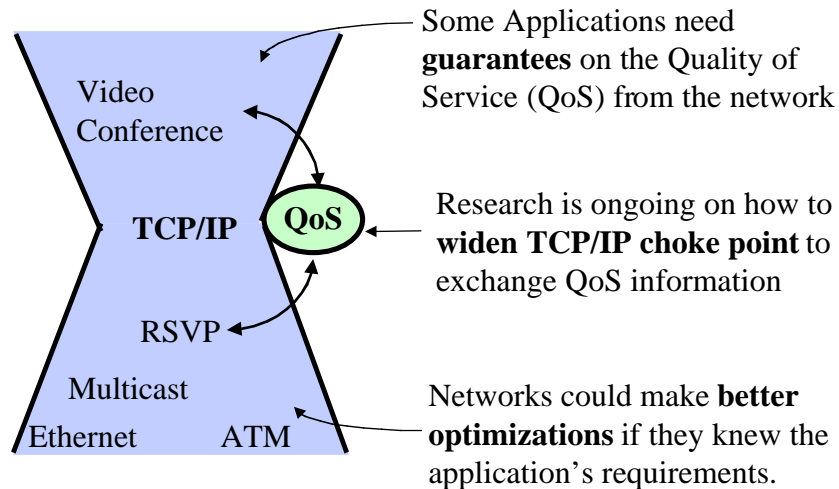


**New Types of Applications** were added without the need to understand anything about networks beyond the Sockets API

**New Networking Technologies** were added without the need to understand anything about applications beyond their use of TCP/IP

\*Dave Clark from a talk at MIT circa 1996

## But Applications and Networks could be Implemented More Efficiently, if More Information were Available



5 06/00 Questions 9/11.00

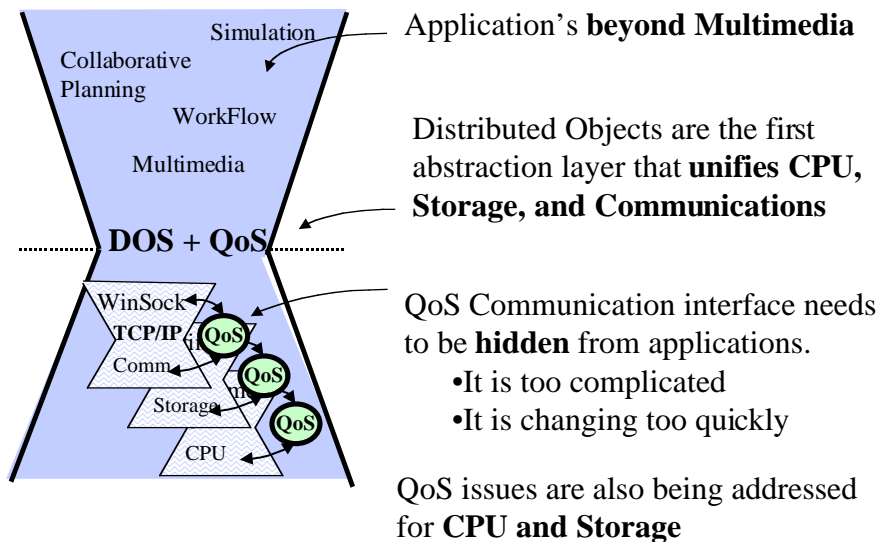
QoS in DOS ECOOP 2000

John Zinky

BBN Technologies



## Distributed Objects with QoS Extensions is a Powerful Abstraction Layer on which to Build Applications



6 06/00 Questions 9/11.00

QoS in DOS ECOOP 2000

John Zinky

BBN Technologies



## Who is Responsible for the The QoS Problem?

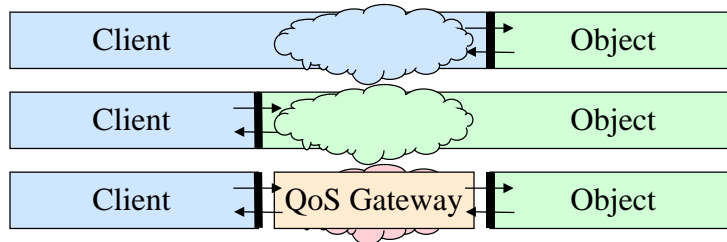
- QoS is well-controlled Client and Object are at the same location



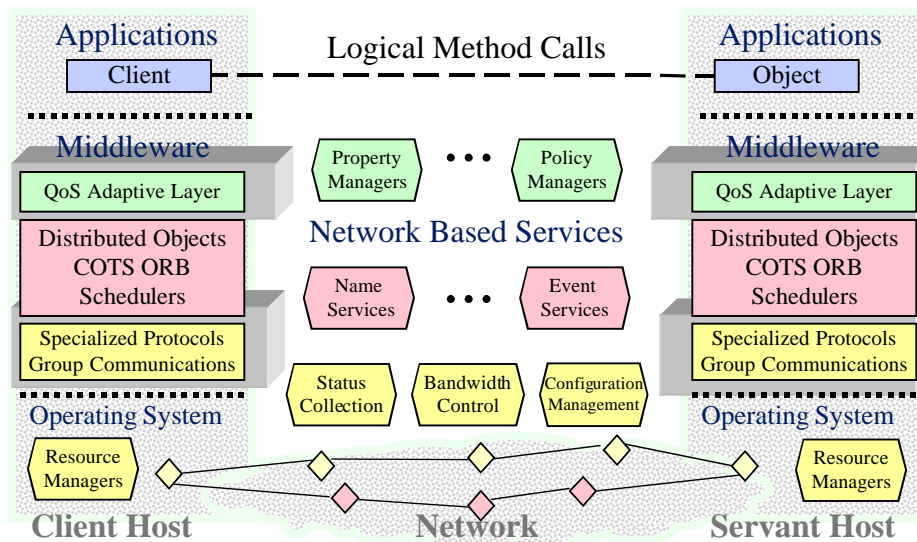
- QoS is unruly across the network



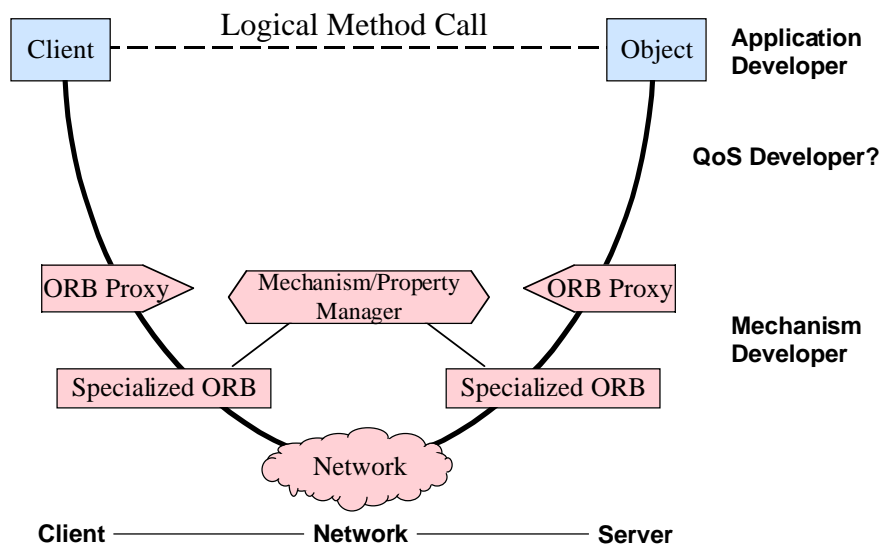
- How can we resolve this?



## Current ORBs Must be Extended for Adaptive QoS Interface and Control



## What Support Should be Given to Different Roles



9 06/00 Questions 9/11/00

QoS in DOS ECOOP 2000

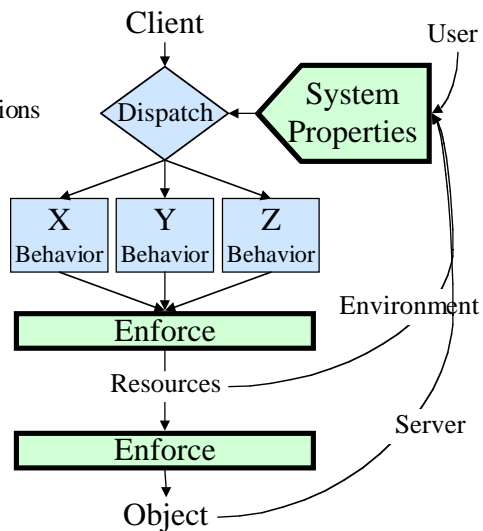
John Zinky

BBN Technologies



## Managing QoS is a Feedback Loop for Changing Behavior Based on System Properties

- Organize System Properties
  - Observe
  - Summarize imperfect observations
  - Translate
  - Actively negotiate
- Alternative Behavior
  - Change policy
  - Control resources
- Emit Events
  - Reevaluate observations
  - Change reservations
- Recover From Errors
  - Mask undesirable behavior



10 06/00 Questions 9/11/00

QoS in DOS ECOOP 2000

John Zinky

BBN Technologies



## Questions

---

- What **kind of QoS** is being handled?
  - Realtime, Dependability, Security, Resource Management
- What **roles** are supported?
  - Application programmer, Mechanism Designer, System Administrator
- How is QoS adaptability **reused**?
  - Service, Library, OS extension, Programming Language
- How is QoS **requirements specified**?
  - Specialized Language, QoS Translation,
- **Where** does QoS adaptability occur?
  - Above, In, or Below the Middleware/ORB
- What **time horizons** does Adaptability occur?
  - Design time, System Configuration, Connection time, Invocation time

## Workshop Objectives

---

- Collect a common **understanding of QoS**
- Discuss dependencies of QoS on **different layers**,
  - e.g. below, in, above ORBs
- **The QoSDOS Reference Model** for Provisioning QoS in Middleware?

## Sessions

---

9:00 Overview

**9:15 QoS Support in and below ORBs**

10:00 Break

**10:15 QoS Specifications**

12:00 Lunch

**1:00 QoS Frameworks "above the ORB"**

2:15 Break

**2:30 QoS Frameworks "above the ORB" (cont)**

4:00 Break

**4:15 QoS Streaming in OO, QoS enabled applications**

5:15 Close