



# Performance Evaluation Of Middleware Bridging Technologies

---

*Rod Fatoohi, Vandana Gunwani, Qi Wang  
& Charlton Zheng*

Computer Engineering, San Jose State U., San Jose, CA

ISPASS-2000, April 24, 2000, Austin, TX



# Outline

---

- Background
- Middleware technologies: DCE, CORBA, DCOM
- Bridging properties
- DCOM/CORBA Bridges: results, observations
- DCE/CORBA Bridges: results, observations
- Concluding remarks



# Background

---

- Middleware: set of common services that enable applications & end users to exchange information across networks
  - Mask differences in OS & network services from end users
  - Problem: middleware services from different technologies are incompatible
- Bridges needed to enable interoperability



# Middleware Technologies: DCE

---

- Distributed Computing Environment by The Open Group: reference implementation
- Based on RPC
- Main services: Cell Directory Service (CDS), Security Service, Distributed Time Service
- Mainly support for C
- Available on most platforms

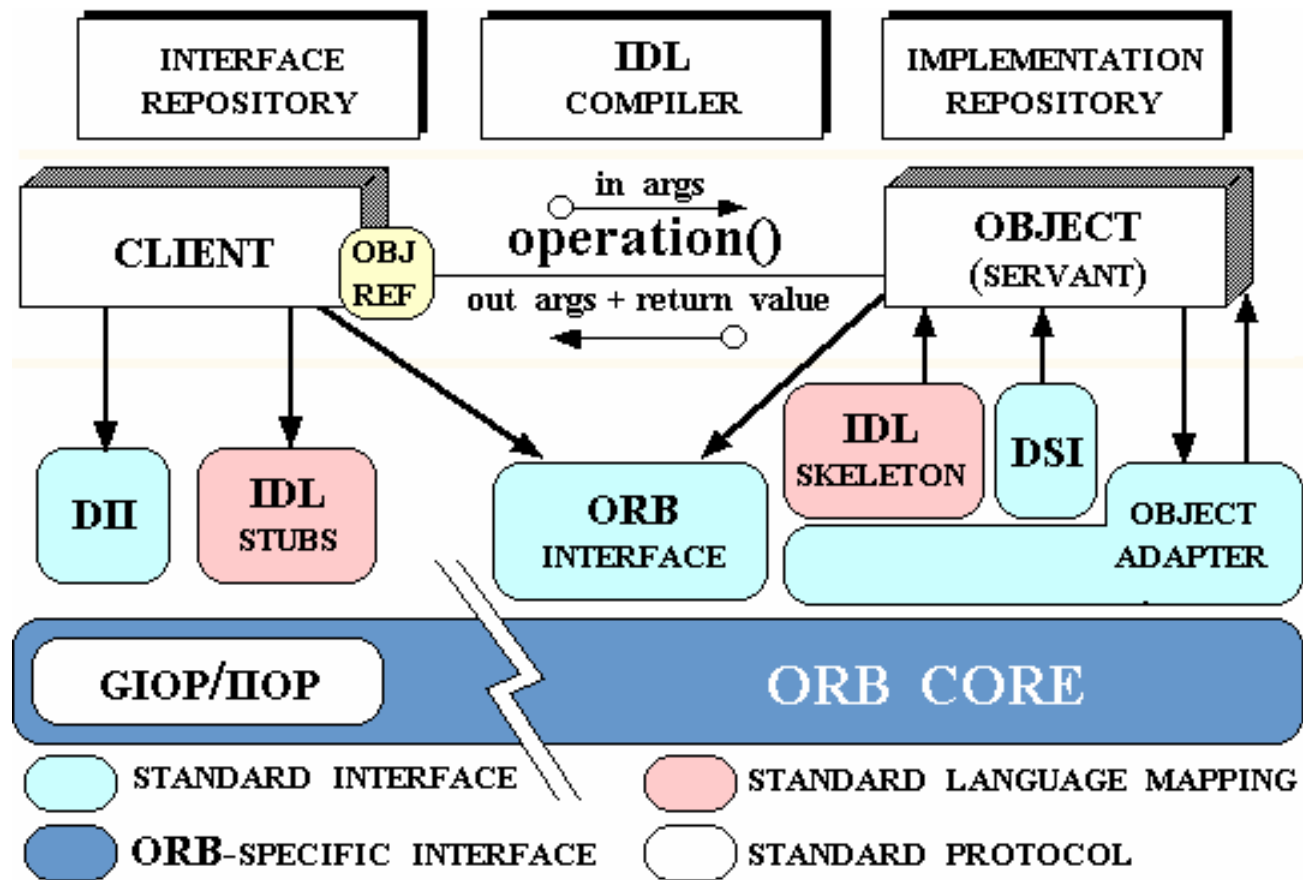


# Middleware Technologies: CORBA

---

- Common Object Request Broker Architecture by Object Management Group (OMG): specifications only
- Uses standard communication protocol (IIOP) & standard interface (IDL)
- Over dozen services specified
- Different languages: Java, C++, C, ...
- Available on multiple platforms

# CORBA Architecture





# Middleware Technologies: DCOM

---

- Distributed Component Object Model by Microsoft: implementation & specification
- Extension of COM to communicate w/ remote objects
- Uses Object RPC for communication
- Many services implemented
- Uses IDL w/ Java, C++, C
- Mainly on Windows platform (NT & '98)



# Bridge

---

- process that allows a client in one middleware domain to make requests to, and receive replies from, a server in another middleware domain.



# Bridge Properties

---

- **Unidirectional vs. Bi-directional**
- **Static vs. Dynamic**
- **Customized vs. Commercial**
- **ORB-specific vs. ORB-neutral**



# Test Problems

---

- *Count*
  - Invokes method to increment variable 1000 times & measures average response time
- *BankATM*
  - Simulates simple bank application

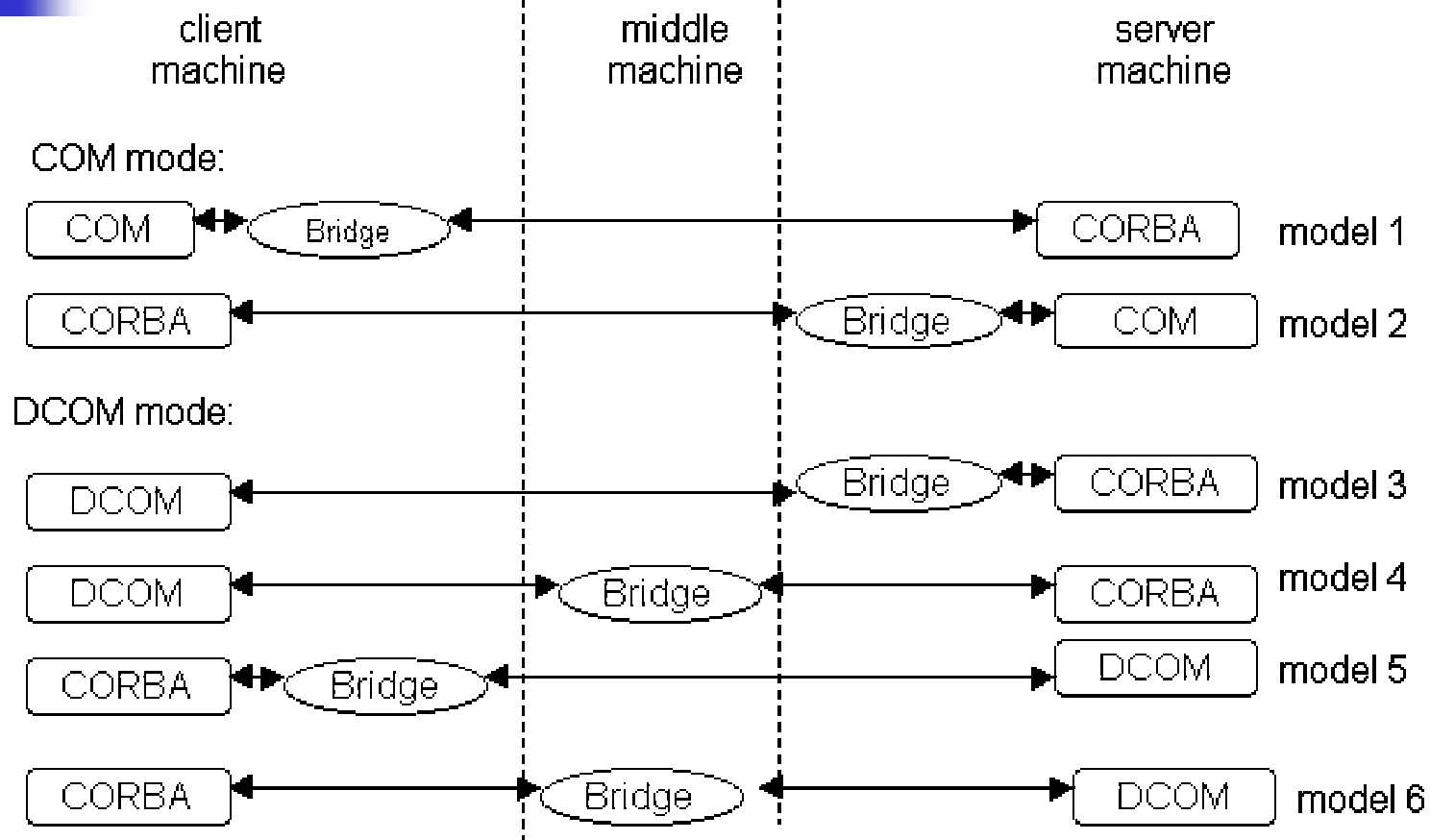


# DCOM/CORBA Bridges

---

- Based on OMG Inter-working Spec
  - 2 parts: A – between COM & CORBA  
B – between DCOM & CORBA
  - Specifies 6 configurations

# Bridge Configurations





# Bridge Location

---

- Bridge on the client side:
  - server's communication protocol is used
  - Bridging computation on the client side
  - Installing bridge on each client machine
- Bridge on the server side:
  - Bridging computation on the server side
- Bridge on 3<sup>rd</sup> machine
  - Additional process needs to be configured



# DCOM/CORBA Bridges: Results

---

- 2 bridges: ObjectBridge by Visual Edge & OrbixCOMet by IONA
  - Both bi-directional & dynamic
- NT 4.0 w/ Pentium II 233 MHz, 128 MB
- Language:
  - C++ for both bridges
  - Also Java & Java Applet for ObjectBridge
  - Same language for both client & server
- **C**: Client, **S**: Server, **B**: Bridge; **X**, **Y**, **Z**: 3 machines; all results in milliseconds.



# OrbixCOMet C++ Client & Server

	<b>CORBA Server</b>	<b>DCOM Server</b>
<b>CO RB A  C</b>	<b>C &amp; S on X: 2.470</b>	<b>C, B &amp; S on X: 250.80</b>
	<b>C on X, S on Y: 2.664</b>	<b>C &amp; B on X, S on Y: 250.87</b>
		<b>C on X, B &amp; S on Y: 250.87</b>
		<b>C on X, B on Y, S on Z: 251.73</b>
<b>DC OM  C</b>	<b>C, B &amp; S on X: 3.021</b>	<b>C &amp; S on X: 0.23</b>
	<b>C &amp; B on X, S on Y: 3.164</b>	
	<b>C on X, B &amp; S on Y: 3.899</b>	<b>C on X, S on Y: 1.262</b>
	<b>C on X, B on Y, S on Z: 4.183</b>	



# ObjectBridge C++ Client & Server

	<b>CORBA Server</b>	<b>DCOM Server</b>
<b>CORBA</b>	<b>C &amp; S on X: 0.397</b>	<b>C, B &amp; S on X: 1.525</b>
	<b>C on X, S on Y: 0.991</b>	<b>C &amp; B on X, S on Y: 1.604</b>
		<b>C on X, B &amp; S on Y: 1.702</b>
<b>C</b>		<b>C on X, B on Y, S on Z: 1.834</b>
<b>DCOM</b>	<b>C, B &amp; S on X: 0.841</b>	<b>C &amp; S on X: 0.23</b>
	<b>C &amp; B on X, S on Y: 1.141</b>	
	<b>C on X, B &amp; S on Y: 1.082</b>	<b>C on X, S on Y: 1.262</b>
<b>C</b>	<b>C on X, B on Y, S on Z: 1.191</b>	



# ObjectBridge Java Client & Server

	<b>CORBA Server</b>	<b>DCOM Server</b>
<b>CO RB A  C</b>	<b>C &amp; S on X: 1.432</b>	<b>C, B &amp; S on X: 2.113</b>
	<b>C on X, S on Y: 1.602</b>	<b>C &amp; B on X, S on Y: 1.772</b>
		<b>C on X, S &amp; B on Y: 1.842</b>
		<b>C on X, B on Y, S on Z: 1.983</b>
<b>DC OM  C</b>	<b>C, B &amp; S on X: 0.982</b>	<b>C &amp; S on X: 0.091</b>
	<b>C &amp; B on X, S on Y: 1.332</b>	
	<b>C on X, B &amp; S on Y: 1.352</b>	<b>C on X, S on Y: 1.922</b>
	<b>C on X, B on Y, S on Z: 1.402</b>	



# ObjectBridge Java Server & Applet Client

	<b>CORBA Server</b>	<b>DCOM Server</b>
<b>CO RB A  C</b>	<b>C &amp; S on X: 1.573</b>	<b>C, B &amp; S on X: 2.383</b>
	<b>C on X, S on Y: 22.342</b>	<b>C &amp; B on X, S on Y: 2.073</b>
		<b>C on X, S &amp; B on Y: 22.451</b>
		<b>C on X, B on Y, S on Z: 22.885</b>
<b>DC OM  C</b>	<b>C, B &amp; S on X: 1.031</b>	<b>C &amp; S on X: 0.401</b>
	<b>C &amp; B on X, S on Y: 1.372</b>	
	<b>C on X, B &amp; S on Y: 1.392</b>	<b>C on X, S on Y: 1.973</b>
	<b>C on X, B on Y, S on Z: 1.472</b>	



## DCOM/CORBA Bridges: Observations

---

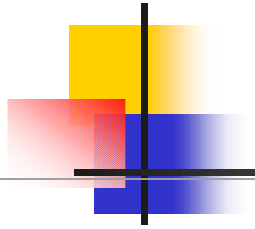
- ObjectBridge is faster than OrbixCOMet
- Each bridge adds ~ 0.5 ms overhead
- In general, C++ outperforms Java
- Bridge location impact is insignificant
- Java Applet is slow



# DCE/CORBA Bridge: Results

---

- DCE-CORBA bridge by Inprise
- Unidirectional: CORBA client -> DCE server
- Acts as CORBA server (<- CORBA client) & DCE client (-> DCE server)
- Supports VisiBroker ORB only
- Generates CORBA IDL from DCE IDL
- Solaris 2.5, UltraSPARC Iii 333 MHz, 128 MB
- Client language: C++, Java, Java Applet



# DCE-CORBA Bridge

	DCE Server	CORBA Server
DCE C	C & S on X: 0.698	N/A
	C on X, S on Y: 1.089	N/A
C++ CORBA C	C, B & S on X: 7.832	C & S (C++) on X: 0.583
	C & B on X, S on Y: 7.996	
	C on X, B & S on Y: 7.856	C on X, S (C++) on Y: 0.842
	C on X, B on Y, S on Z: 7.475	
Java CORBA C	C, B & S on X: 8.334	C & S (Java) on X: 1.472
	C & B on X, S on Y: 8.271	
	C on X, B & S on Y: 7.569	C on X, S (Java) on Y: 1.401
	C on X, B on Y, S on Z: 7.768	
Java applet CORBA C	C, B & S on X: 11.326	C & S (Java) on X: 4.373
	C & B on X, S on Y: 10.909	
	C on X, B & S on Y: 28.161	C on X, S (Java) on Y: 25.807
	C on X, B on Y, S on Z: 29.661	



## DCE/CORBA Bridge: Observations

---

- Delay through bridge: 7 ms
- CORBA IIOP is bit faster than DCE RPC
- C++ faster than Java & a lot faster than Java Applet
- Bridge location impact is insignificant



# OODCE/CORBA Bridging

---

- Object-Oriented DCE: wrapper to support O-O development using C++
- Uses DCE IDL to define services
- Results: similar to DCE/CORBA results



# OODCE-CORBA Bridge

	<b>OODCE Server</b>	<b>CORBA Server</b>
<b>OODCE C</b>	<b>C &amp; S on X: 0.797</b>	N/A
	<b>C on X, S on Y: 1.127</b>	N/A
<b>C++ CORBA C</b>	<b>C, B &amp; S on X: 7.461</b>	<b>C &amp; S (C++) on X: 0.583</b>
	<b>C &amp; B on X, S on Y: 8.041</b>	
	<b>C on X, B &amp; S on Y: 7.510</b>	<b>C on X, S (C++) on Y: 0.842</b>
	<b>C on X, B on Y, S on Z: 7.805</b>	
<b>Java CORBA C</b>	<b>C, B &amp; S on X: 7.847</b>	<b>C &amp; S (Java) on X: 1.472</b>
	<b>C &amp; B on X, S on Y: 7.973</b>	
	<b>C on X, B &amp; S on Y: 7.626</b>	<b>C on X, S (Java) on Y: 1.401</b>
	<b>C on X, B on Y, S on Z: 8.083</b>	



# Concluding Remarks

---

- 3 bridges performed well using 2 simple programs
- Bridge overheads are not significant
- ObjectBridge is faster than OrbixCOMet
- Bridges have different properties
- Bridge location impact is not in performance but in deployment



# Acknowledgement

---

- Inprise, Visual Edge & IONA
- NASA