

System Integration

Mini Case Studies © 2010

Middleware

Shawn A. Butler, Ph.D.
Senior Lecturer, Executive Education Program
Institute for Software Research
Carnegie Mellon University

Objectives

- Understand the difference between different types of middleware

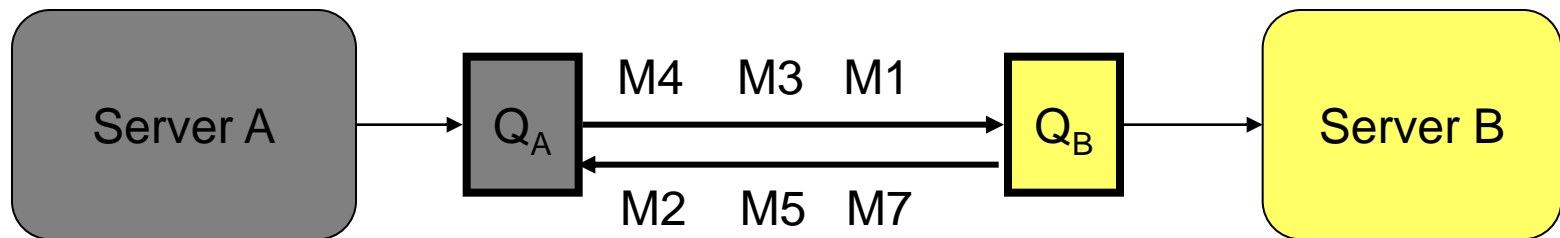
Agenda

- Message Oriented Middleware
- Object Architectures
- Transaction Processing

Messaging

- Messaging is one of the most popular middleware technologies for integrating disparate applications
- Asynchronous communication, but can be designed to do synchronous communication
- Different types of MOM
 - Point to point message queuing
 - Broadcasting
 - Quality of Service, CORBA implemented messaging services

Basic Message Queuing



In request/reply messaging, the request and reply occur within a single transaction

Publish/Subscribe Messaging

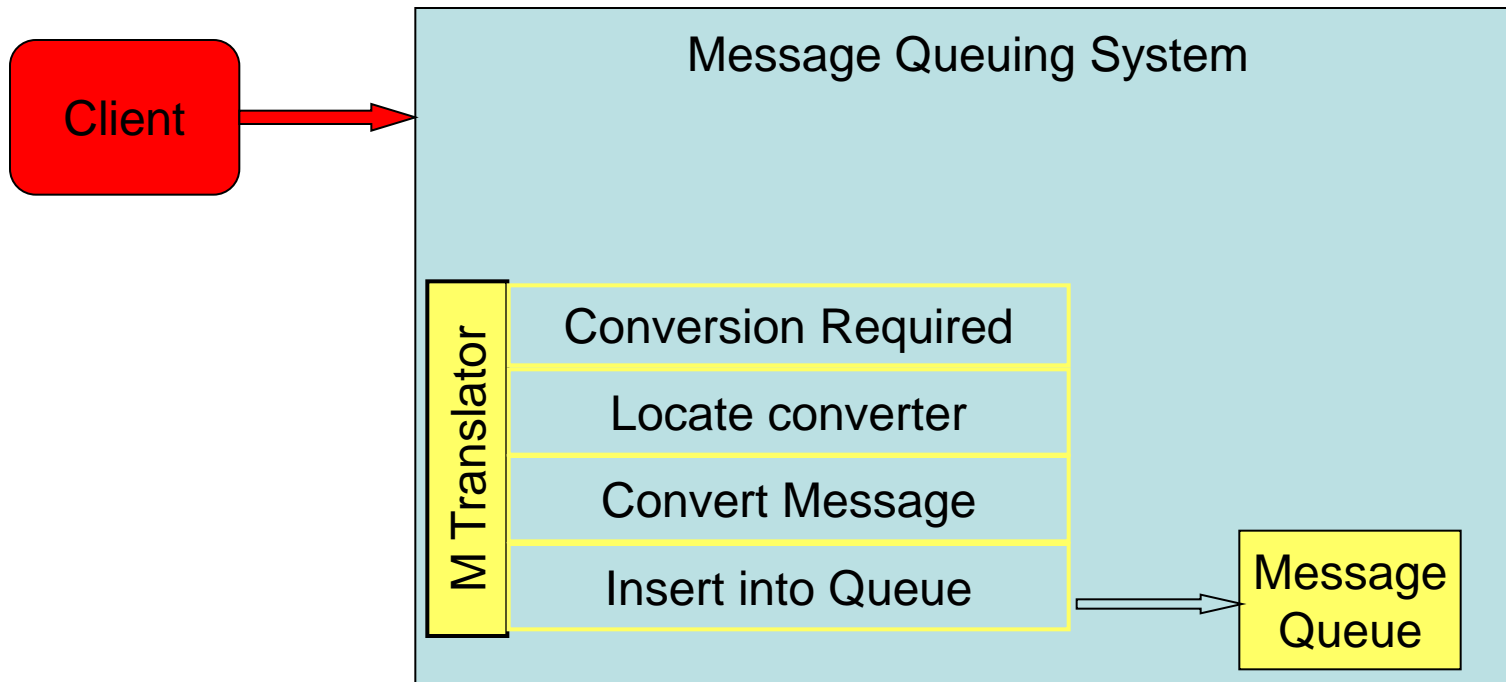
- Point-to-point Option
 - Clients 'subscribe' to information
 - Server sends a message when a match is found
- Broadcast Option
 - Clients 'subscribe' to information
 - Server broadcasts a message to all subscribed clients

Message Translation

Message translation converts data in a form that can be transmitted in a message and converts the data back into its native form when it reaches the target

- Fixed point translation –
 - Distribute translation software to powerful processors
 - Best when messaging follows a set route
- En-route translation –
 - Takes advantage of spare computing power
 - Routes vary

Message Queuing System



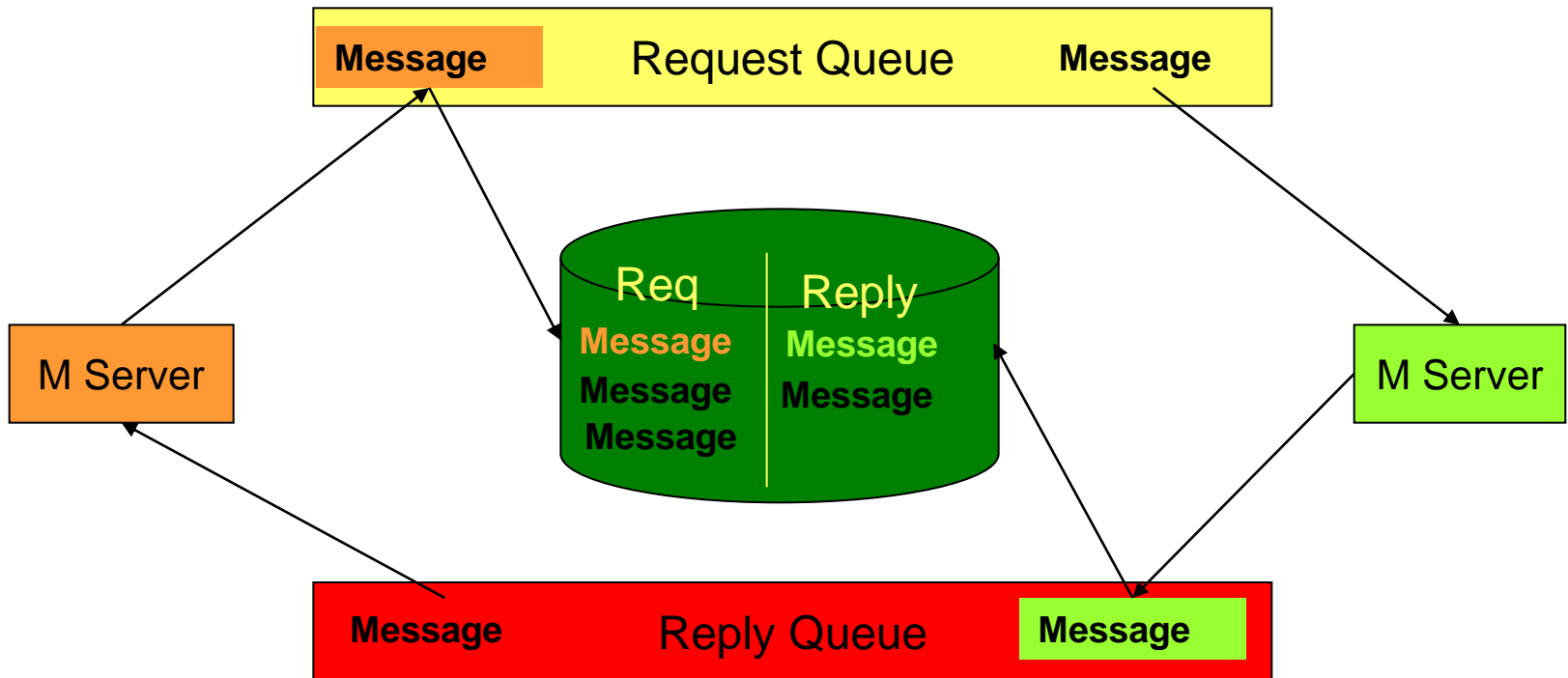
Queue Management

- Synchronicity – synchronize functions
- Response time – time to reply to a message
- Message content – format and data structures within a message
- Message size – physical size of the message in bytes
- Message priority – level of importance
- Queue volume – total number of messages in a queue
- Queue timeouts – length of time a queue can sustain messages
- Queue persistence - Ability to store messages in a persistent form
- Queue priority – Message queue priority

Why Message Persistence?

- Queue failure
- Request message is corrupted
- Reply message is corrupted
- Recipient server fails before finishing process
- Requesting client fails before receiving the reply

Persistent Queue



Object Architectures

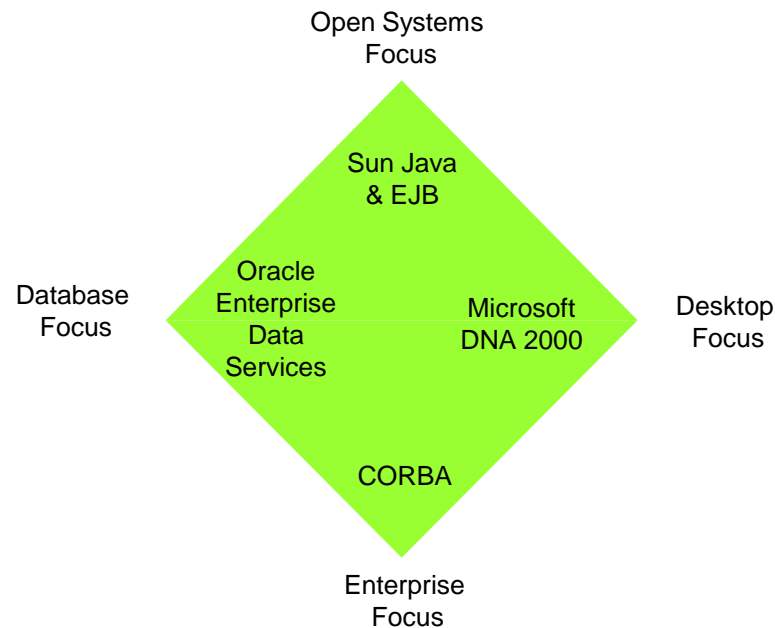
A technology architecture will always affect the application architecture within which it is used. When a major technology product, such as an ORB, COM+, or EJB, is used for systems development, its architecture will shape the remainder of the architecture around it.

“Enterprise Application Integration”

Ruh, Maginnis, and Brown, Wiley Publisher, Page 84, Chapter 5

Commonly Used Object Architectures

- .Net (formerly DCOM/COM+)
- CORBA
- Enterprise Java Beans



CORBA

- Object Management Architecture is the specification for CORBA
- Object Request Broker (ORB) a distributed mechanism for transparently communicating client requests to various target object implementations
- ORBS provide three functions:
 - Operating system transparency
 - Remote object life cycle services
 - Remote object method calls

CORBA Services

- Information management services – data management services
- Task management service – event, concurrency, and transaction services
- System management services – naming, lifecycle, licensing, and trader services
- Infrastructure services – security, time, and messaging services

CORBA Facilities and Domains

- Application level services
 - Distributed document component facility
 - Internet facility
 - Print facility
 - Workflow management facility
 - Calendar facility
- Domains – business-specific standard groups
 - Electronic commerce
 - Finance
 - Medical
 - Manufacturing

Previous Limitations to CORBA

- Previous limitations
 - Lack of real-time capability
 - Lack of robust asynchronous mechanism
- Current limitations
 - Few basic services are actually implemented

Enterprise Java Beans

Web Clients	Java applets, servlets, server pages
Database Access	Java Database Connectivity (JDBC)
Distributed Components	Java Beans Enterprise Java Beans
Distributed Objects	Remote Method Invocation
Object Directory	Java Naming and Directory (JNDI)
Transaction Services	Java Transaction Services (JTS)
Message Service	Java Message Service (JMS)

CORBA Components Package

- Container environment that packages transactionality, security, and persistence with interface and event resolution
- Integration with Java Beans (EJB)
- Software distribution format
- Containers?
 - Track event types
 - Provide channels for events
 - EJBs act as CORBA components

Transaction Processing

- Enforcement of ACID properties for all intersystem transactions
- MOM doesn't guarantee ACID
- Example:
 - Customer wants airplane, hotel and rental car reservations
 - If everything is in one database, then no problem if the transaction has to roll back
 - What if there are three different databases, three different applications, in three different locations, with long delays between parts
 - TP can handle management of transaction activities in a distributed, multi-application environment

Summary

- Message Oriented Middleware is one of the most popular applications for distributing information among different applications and systems
 - It provide asynchronous delivery of messages among all system components
 - Relies on a naming service to find all components
- Many products are available to support integration of system components
- Transaction Process Monitors (TP Monitors) ensures ACID properties in distributed environments