MQSeries v6.x Application Programming (4 Days)

AUDIENCE:
System administrators, network professionals and advanced web developers that will be maintaining MQSeries environments.

COURSE ABSTRACT:
This course will use both instructor-led discussions and hands-on workshops to demonstrate the capabilities of programming COBOL applications using IBM's MQSeries. Within this course, each student will be exposed to the following topics; the major components of an MQSeries installation (both local and remote), install the MQSeries client, develop MQSeries programs, connect to queue manager, MQI API, both read and send MQSeries messages, perform exception handling, understand the role of messaging middleware, transactions and MQ syncpoints, security in the MQ environment and MQ triggering.

COURSE OBJECTIVES:
Upon conclusion participants will have acquired these skills:

- Understand the capabilities of MQSeries and WebSphere MQ
- Depict the architecture of the MQ environment: Queue Managers, Channels, Local and Remote Queues and MQ clustering
- Design, build COBOL applications for MQSeries environment
- Connect and disconnect to MQ queue Manager
- Perform following tasks with messages; getting/putting, replies and reports and use of context variables
- Highlight the role of the MQI API for application development
- Depict the role of all MQI commands for accessing messages (MQGET/MQPUT)
- Understand the use of transactions with MQSeries
- Illustrate the process for performing data conversions
- Understand the role of triggering in MQSeries environment
- Depict the use of Message groups
- Illustrate the role of security in MQSeries environment
- Define usage of Message segmentation
- Depict the role of distribution lists
- Demonstrate the role of MQ security

Course Content

I. Messaging Overview

A. Messaging Basics
Illustrating Middleware
Messages
Defining a Queue
Queue Utilization
Point-Point Model
Publish/Subscribe
Messaging Simplification
Reliability and Data Integrity
Single Delivery
Units of Work
Failure Handling
Quality Assurance

B. Overview
Defining MQSeries
Advanced Features
Application Environments
Messaging Fundamentals
Application Communications
Message Types
Message Structure
Message Descriptor
MQSeries Runtime
Messaging Segmenting
Distribution Lists
Types of Queues
Local Queue Types
Queue Manager Functions
Program to Program
Two Systems
Support Platforms

C. Application Programming Interface
MQSeries Programming APIs
Using MQSeries MQI
Application Messaging Interface
Java Messaging Service
MQSeries Programming APIs
JMS and JMS Providers

D. Bridges, Connectors and Adapters
MQSeries Adapters
EDI Processing

E. Message Brokers
The Message Flow
Message Brokering Example
Customer Benefits of MQSeries

II. Message Oriented Middleware

A. Overview
Message-Oriented Middleware (MOM)
Asynchronous Messaging
Publish/Subscribe Messaging
Publish/Subscribe Model
Point-to-Point Model
Point-to-Point Messaging
Enterprise Application Integration
Message Broker
Message Broker – Motivation
Message Broker – Structure
Message Broker vs MOM

B. WebSphere MQ Integrator
WMQI – WebSphere MQ Integrator
WMQI Architecture
Message Broker Configuration
WMQI - Message Dictionary
WMQI – Rules Engine
WMQI – Message Flow Example

C. MQSeries Workflow
Workflow Definition
Workflow Example – Purchase Order
Key Benefits of Workflow
Workflow Management Systems
Workflow – Basic Architecture
Workflow – Build Time Functions
WebSphere MQ Workflow

III. Distributed Objects

A. Messaging Basics
Distributed Objects
Parameter Passing
Distributed-Object Model

B. Message-Oriented Communication

Persistence and Synchronicity
Message-Queuing Model
Message-Queuing Architecture
Architecture of Message-Queuing
Message Brokers
Example: IBM MQSeries
MQSeries Channels
Message Transfer

IV. Designing MQSeries Applications

A. Overview
Types of Messages
Report Messages

B. Message Patterns
Message Patterns
One-to-One Message Flow
One-to-Many Message Flow
Many-to-One Message Flow
Publish/Subscribe

C. Messaging Components
Important Messaging Components
Message Queuing Features
MQSeries Cluster
Distributed Queuing
Planning Your Design
WebSphere MQ Objects
Designing Messages

IV. Programming with MQI

A. MQI Function Calls
MQI Abilities
Platforms and Languages
MQI Sequence
MQI Components
MQCONN Usage
Connecting to Queue
Disconnecting from Queue
Using MQOPEN
Using MQCLOSE
MQCLOSE Option Parameters
Using MQGET
Using MQPUT
Using MQPUT1
Using MQINQ
Using MQSET
MQSET Example
MQBACK Units of Work
MQCMIT Usage
Basic Elementary Data Types
MQI Pointer Elements
Pointer Data Types

V. MQI Data Structures

A. MQI Data Structures
MQI Structure Type Rules
MQI Special Data Structures
MQAIR Structure Example
MQBO Structure Example
MQCNO Structure Example
MQDH Structure Example
MQGMO Structure Example
MQMD Structure Example
MQMDE Structure Example
MQOD Structure Example
MQOR Structure Example
MQPMR Structure Example
MQPMO Structure Example
MQTM Structure Example
MQTMC2 Structure Example

VI. MQSeries Transactions

A. Overview
What is a Transaction?
Motivation for Transactions
Atomic Operations
Network or Machine Failure
Shared Data Philosophy
Transactional Model
ACID Properties

B. Transaction Model
Transactional Models
Illustrating Flat Transactions
Nested Transactions
Chained Transactions
Transaction Aborts
Transaction Commit
Transaction Rollback
Transaction Demarcation

C. Transaction with MQI
MQSeries Transactions
Single vs. Two Phase Commit
MQI Transactions
Units of Work
Global Unit of Work
MQCMIT Usage
Distributed Transactions
Two-Phase Commit Protocol
MQBACK Usage
MQBACK Example
MQBACK Units of Work
MQCMIT Usage
Synchronization Points

VII. MQSeries Security

A. MQSeries Security Architecture
Using Remote Queue Manager
MQSeries Administration
Security Checks
MQ Security Architecture
Retrieve MQ User ID
Using setmqaut Command
Grant Access with setmqaut

Dumping MQ Authorizations
Dump Profile Settings
Displaying MQ Authorizations

B. Channel Security Exits
Channel Agents
Channel Security Exits
Processing Overview
Channel Exit Programs
Using MQ_CHANNEL_EXIT
Invoking Channel Exit

VIII. Programming with COBOL

A. Overview
Connection Modes
MQSeries Classes

B. Establishing Connections
MQSeries Connections
Using hostname
Using Channel
Using port
Using userid/password
MQQueueManager
Queue Interaction
MQSeries Messages

C. MQSeries Patterns
Point-to-Point Pattern
Pattern: Point-to-Point Steps
Sender Application
Connecting and Accessing Queue
Build Message
Message Receiver Steps
Receiver Application
Connecting to Queue
Retrieve Message
Pattern: Request/Reply Pattern
Connect to Queue Manager
Set Message Options
Retrieve the Message
Responding to Messages
Pattern: Message Grouping
Logical Messages
Message Grouping Steps
Retrieve Grouped Messages

IX. MQSeries Triggers

A. Overview
Client to Server Communication
Request & Reply
Trigger Applications
Trigger Components
Application and Trigger Message
Trigger Monitor: z/OS
MQTM Structure
Trigger Data
Trigger Type
Process Definition
Shared Transmission Queue

X. Using MQ Clustering

A. Overview
Defining Clusters
MQ Clusters Benefits
Reduced MQSeries Administration
Without Cluster
Cluster Solution
Goals of Clustering
Cluster Concepts
Achieving Clustering
Cluster Components
Remote Queue Access
Client View or Remote View
How Clusters Work
Message Routing
New Clusters Components
Cluster Configuration

Appendix A: MQSeries Reason Codes