# Top Ten ways to improve your MQ/IIB environment

**Suganya Rane**

**Solution Architect**

**Prolifics**

# Agenda – Top 10 MQ & IIB Tips

1. Business Requirements
2. Sizing
3. Naming Standards
4. Logging
5. Application Design Standards
6. High Availability
7. Disaster Recovery
8. Monitoring
9. Role Requirements
10. Skill Development

# 1. Business Requirements

- Define the requirements
- Document interfaces & Stakeholders
- Collect Interface Objects
- Document the interchange specifications
- Define the mappings
- Define the SLA requirements
- Testing
  - Environments
  - Requirements
- Implementation requirements

# 2. Sizing Requirements

## Server Sizing

WebSphere MQ requires:

- Memory
- CPU
- Fast Disk
- Network
- IBM Techline
  - Server Estimates
    - Memory, CPU, DISK
  - Workload Based

## Queue Manager Sizing

- Log File directory space
- Log File Type
- Log File Size - **http://tiny.cc/MQsizing**
- Log File Placement
- Data Directory Size
- Queue Buffers
- Queue Max Size
- Z/OS  (Bufferpools, Logs, PageSets)

# 3. Naming Standards

- Define a naming standard that works
- Select simple names, KISS it
- Avoid object or location identifiers
- Define cluster naming standards
- Use Alias names for local Cluster Queue access
- Keep QMGR Names short
- SDLC identifier (i.e. MQDAPP01)
- Adhere to standards
- Support Pack

# 4. Logging

- Log Types – Circular vs Linear
- Log Sizing is important, for distributed, download and use the log file size estimator @ **http://tiny.cc/MQsizing**
- Running out of log space is bad.
- Keep logs on spindles separate from other file systems
- DEV and SIT use Circular
- UAT, PROD and DR use linear
- Move archived logs off file system
- Delete after successful restart
- Usually 14 days on weekly restart

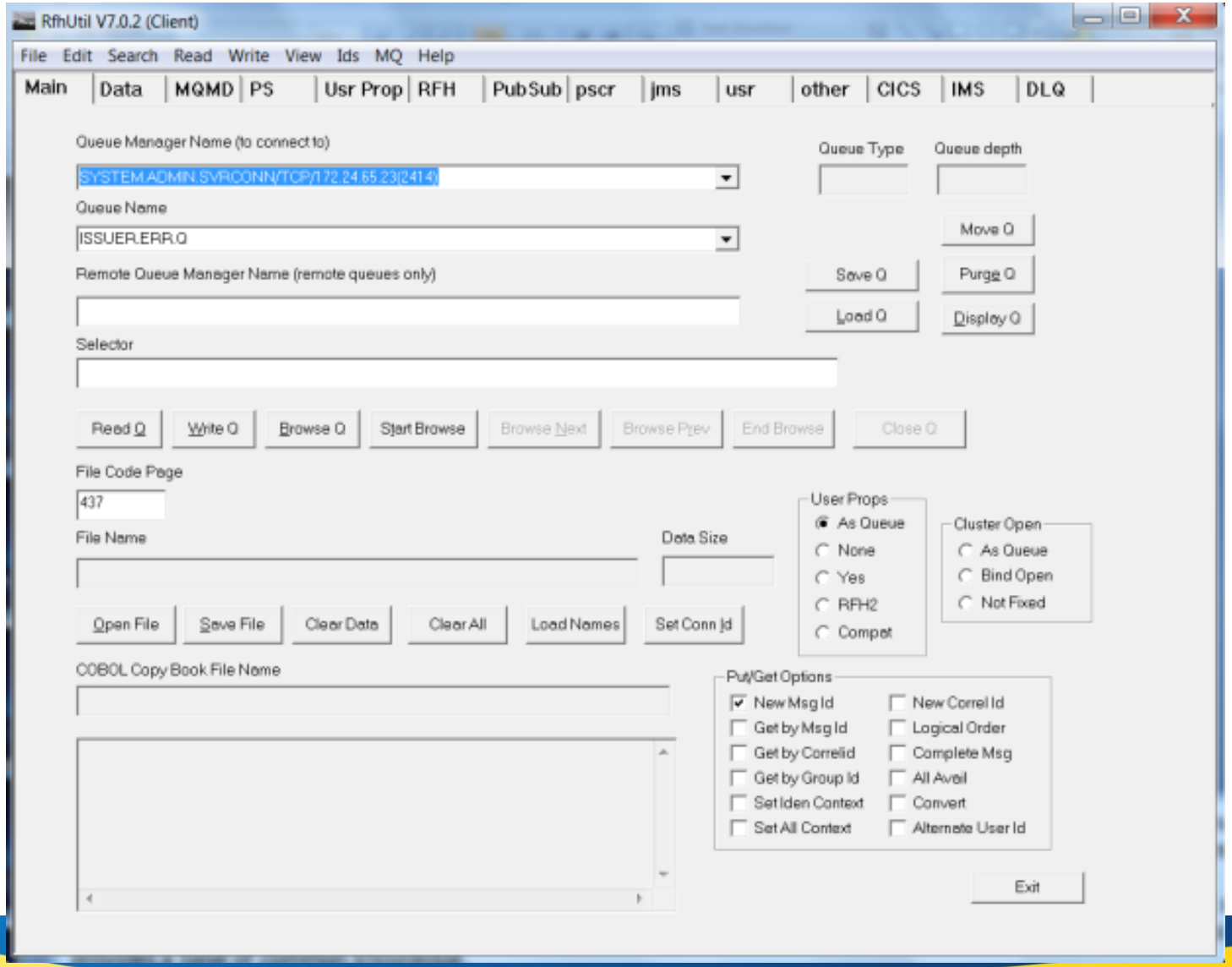
# 5. Application Design Standards

- Create Application programming standards
- Should work with Naming Standards
- Create patterns and detail QMGR interactions
  - Request/Reply
  - Triggering
  - Error Processing
  - Reconnection
  - Poison messages
  - Audit / Logging
- Language Specific Interactions
- Create program specification templates
- Create Test Script Templates
- Perform Code Reviews prior to implementation
- Security
- Wrappers

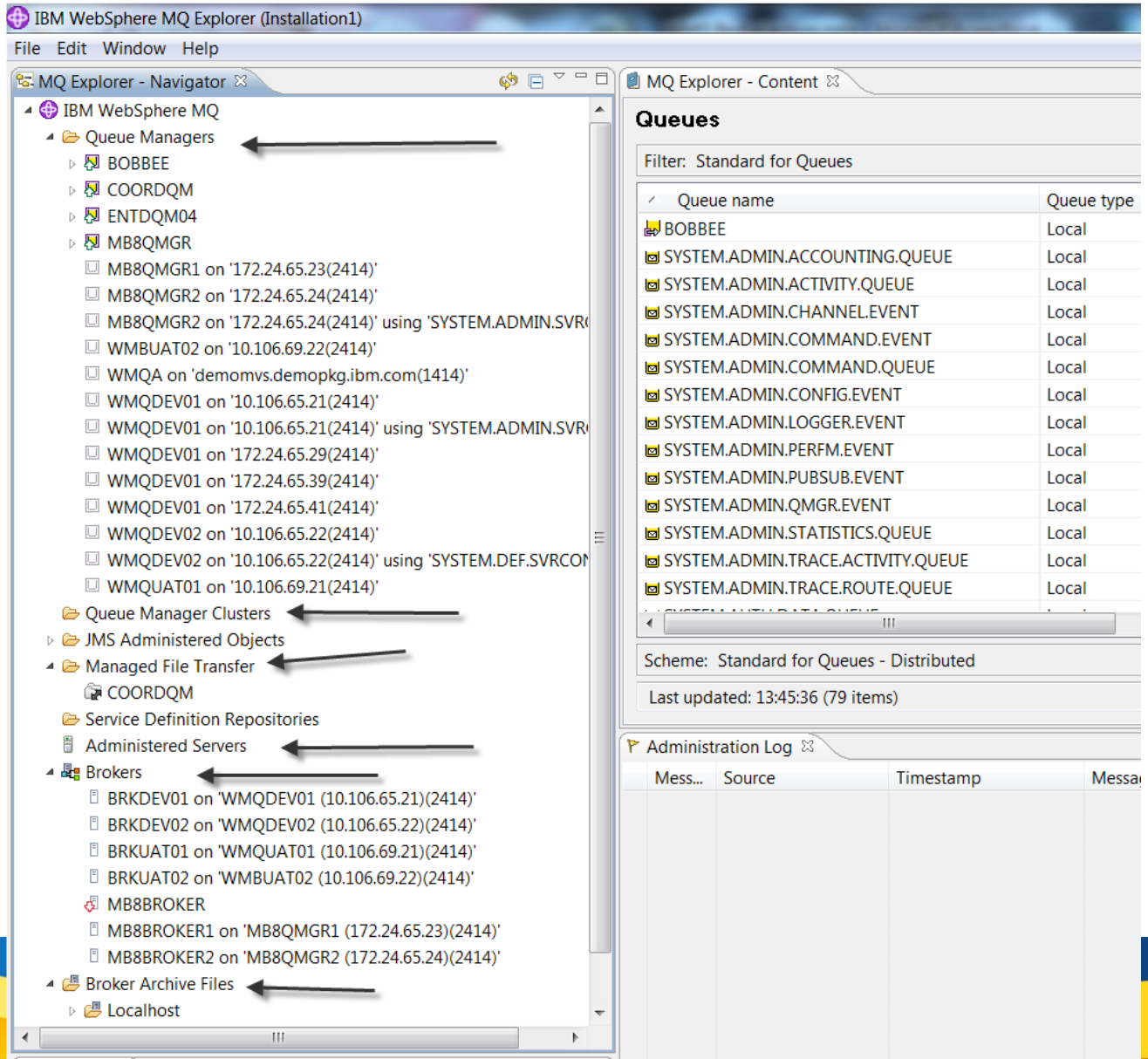
# Tools

- What tools do you need
- Do you standardize or give free hand
- Do you buy or borrow?
- What Tools do your application programmers use
- What tools do your administrators use
- What tools do your Business users use
- What tools do your support team use
- Support Packs
- MQ out-of-the-box

# Tools - RFHUTIL



# Tools – MQ Explorer



# 6. High Availability

- What does High Availability mean to you and your business?
- Different people, different requirements.
- What is the driving force for the availability of a system
- What is my approach?
- How do I implement it.
- What am I really getting.
- Active/Passive
- Active/Active

# 7. Disaster Recovery

- End of the world approach
- SLA's again as determining factor
- Placement of datacenter is key.
- A lot of activity after 9/11 and power grid loss
- File systems are replicated to a secondary datacenter
- Primary activity can be sync or async
- Secondary can be passive or active
- Moving data ?

# 8. Monitoring

- DO NOT WAIT TILL THE LAST MINUTE.
- Pick you Monitoring Approach.
  - simple or complex
  - Feature rich or basic
- Decide the key factors that provide enterprise wide capabilities
- Centralize your Monitoring Console
- Know what to monitor
- MQ Monitoring - http://tiny.cc/MQMonitoring

# 9. Role Definition

- Who are the actors
- What groups control what
- Who do you turn to
- Clear definition
- Multiple Hats
- Training

# 10. Skill Development

- Skilled employees are an asset not a liabillity
- Training is important for the health of the system
- Changing technologies require training
- In-house, remote, Self Paced
- IBM offers training in all features and components of their software
- External sites are available for training
- Hire vs Train
- Certification Testing

# Clustering – Why do we use it

- Simplified administration
- Large WMQ networks require many object definitions (Channels, Transmit queues, Remote queues)
- Workload balancing
  - Spread the load
  - Route around failures
- Flexible connectivity
  - Overlapping clusters
  - Gateway Queue managers
- Pub/sub Clusters

# Clustering

- Don’t have too many cooks
  - Treat a single cluster as a single point of administration
  - Have well defined policies for the gateway queue managers

- Treat all overlapping clusters as a single Namespace
  - Channel names, and therefore queue manager names, should be unique throughout.

# Questions & Answers