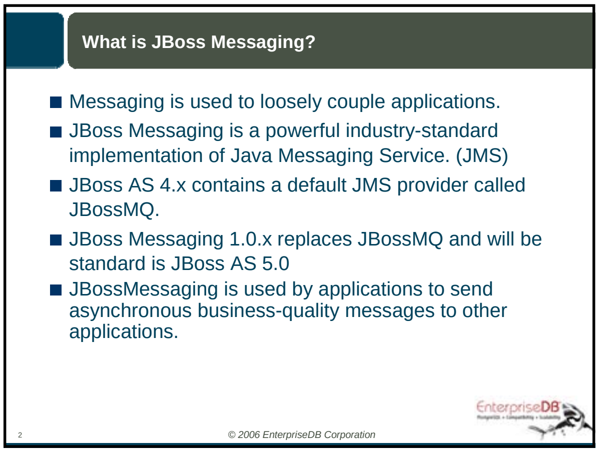
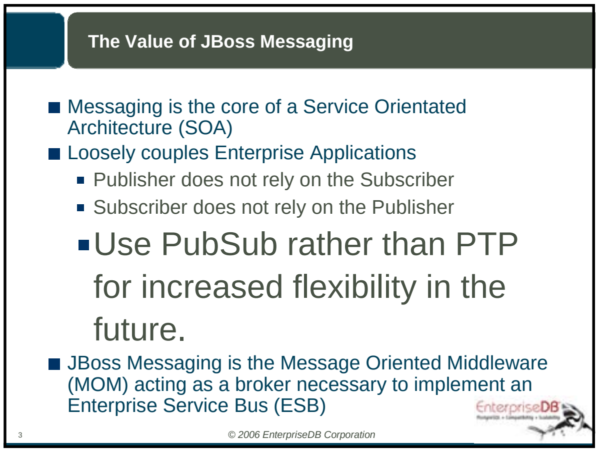
Using JBoss Messaging in the Enterprise v2

Denis Lussier



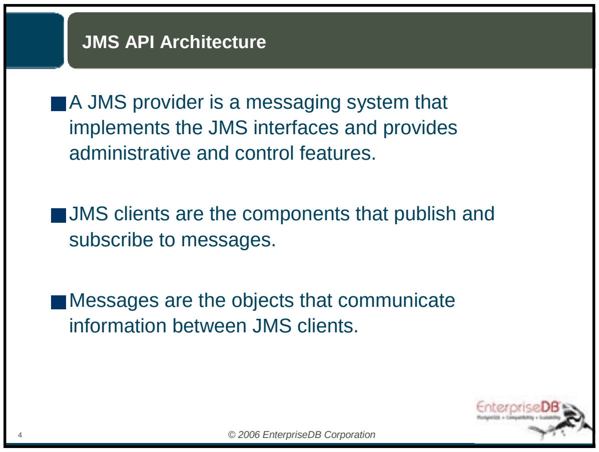
What is JBoss Messaging?

- Messaging is used to loosely couple applications.
- JBoss Messaging is a powerful industry-standard implementation of Java Messaging Service. (JMS)
- JBoss AS 4.x contains a default JMS provider called JBossMQ.
- JBoss Messaging 1.0.x replaces JBossMQ and will be standard is JBoss AS 5.0
- JBossMessaging is used by applications to send asynchronous business-quality messages to other applications.



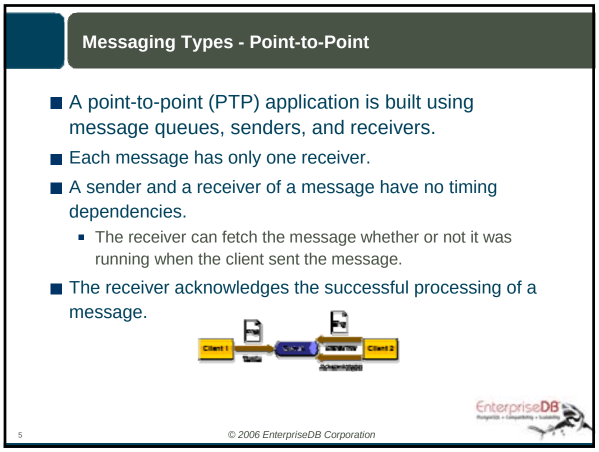
The Value of JBoss Messaging

- Messaging is the core of a Service Orientated Architecture (SOA)
- Loosely couples Enterprise Applications
  - Publisher does not rely on the Subscriber
  - Subscriber does not rely on the Publisher
  - Use PubSub rather than PTP for increased flexibility in the future.
- JBoss Messaging is the Message Oriented Middleware (MOM) acting as a broker necessary to implement an Enterprise Service Bus (ESB)



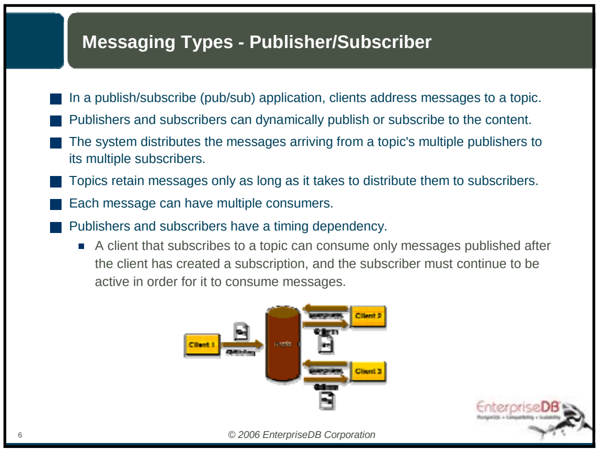
JMS API Architecture

- A JMS provider is a messaging system that implements the JMS interfaces and provides administrative and control features.
- JMS clients are the components that publish and subscribe to messages.
- Messages are the objects that communicate information between JMS clients.



Messaging Types - Point-to-Point

- A point-to-point (PTP) application is built using message queues, senders, and receivers.
- Each message has only one receiver.
- A sender and a receiver of a message have no timing dependencies.
  - The receiver can fetch the message whether or not it was running when the client sent the message.
- The receiver acknowledges the successful processing of a message.

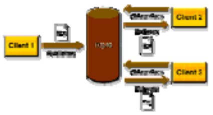


Messaging Types - Publisher/Subscriber

- In a publish/subscribe (pub/sub) application, clients address messages to a topic.
- Publishers and subscribers can dynamically publish or subscribe to the content.
- The system distributes the messages arriving from a topic's multiple publishers to its multiple subscribers.
- Topics retain messages only as long as it takes to distribute them to subscribers.
- Each message can have multiple consumers.
- Publishers and subscribers have a timing dependency.
  - A client that subscribes to a topic can consume only messages published after the client has created a subscription, and the subscriber must continue to be active in order for it to consume messages.

Messaging Types – Pub-Sub with Durable Topic

- Cross between P2P and Pub-Sub
- Receive Client registers topic with an ID & password
- Send Client does not change for durable topic
- Durable Topic Receive Client receives messages even when it was not actively listening at time of message publication



Enterprise Class Messaging Features

- Scalability
  - New High Performance Messaging Core
  - Built for Enterprise grade message traffic
  - New “Pageable Channels” efficiently support large queues that may overflow memory
  - Designed for Clustering from the ground up
  - Production Quality Clustering later this year
  - Support for EJB 2.1 & EJB 3.0
- Guaranteed Delivery
  - Message Autonomy
  - Message Persistence
  - Message Acknowledgement

Scalability

- Message Driven Beans
  - An enterprise bean that handles messages asynchronously
  - Behaves like a JMS listener
  - Can consume and process messages concurrently
    - Can receive hundreds of messages from various applications and process them all at the same time

Guaranteed Delivery

- Message Autonomy
  - Messages are independent units of work that can be resent multiple times
- Message Persistence
  - Messages are stored to disk in an enterprise class RDMS such as EnterpriseDB
- Message Acknowledgement
  - The JMS Provider manages the message as the receiver consumes the message

JBoss JMS implementation

- Invocation Layer
- Security Manager
- Destination Manager
- Message Cache
- State Manager
- Persistence Manager

Invocation Layer

- Handles the communication protocol with clients to send and receive messages
  - Unified Invocation Layer version 2 (UIL2 IL)
    - A multiplexing layer that creates 2 virtual sockets over a single physical to provide bidirectional communication
  - Java Virtual Machine (JVM IL)
    - Eliminates the TCP/IP overhead when the client resides in the same JVM as the server
  - HTTP (HTTP IL)
    - Access over HTTP and HTTPS protocols for use through a firewall

Security Manager

- Applies a role based Access Control List (ACL) to a destination
- Permissions
  - Read
    - A client can receive messages from the destination
  - Write
    - A client can send or publish messages to the destination
  - Create
    - A client can create durable subscriptions from a destination

Destination Manager

- The main service for JBossMQ
- Manages all of the destinations created on the server
- Handles configuration of other key services
  - Message Cache
  - State Manager
  - Persistence Manager

Message Cache

- Handles the pushing of messages to disk when memory resources become scarce
- Works in conjunction with the Persistence Manager
- Determines the least recently used messages when choosing the messages to push to disk

State Manager

- Configures Users
- Configures Roles
- Configures Durable Topic Subscriptions

Persistence Manager

- Stores messages when marked as persistent
- Utilizes a JDBC data source
- Has a low memory overhead
- Highly integrated with the Message Cache

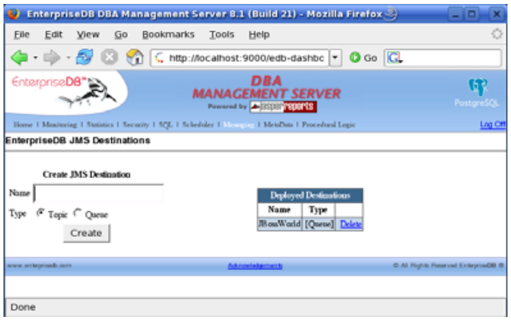
Enterprise JBoss Messaging

- The EnterpriseDB Management Server ships preconfigured with JBoss as a enterprise class JMS provider
  - EnterpriseDB as the default data source of the Persistence Manager
- Simplifies the creation of destinations through the use of an easy to use user interface

Why Isn’t JBoss Messaging enterprise-class

- The short (and only) answer is HSQLDB
- HSQLDB is a great and compact little database that actually inspired the EnterpriseDB compatibility idea.
- Its got some pretty decent ANSI & Defacto compatability.
- It falls down under heavy load and/or with more than 1 GB of data.

Creating Queues and Topics


EnterpriseDB DBA Management Server 8.1 (Build 21) - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://localhost:9000/edb-dashbc
Go
EnterpriseDB
DBA
MANAGEMENT SERVER
Powered by jasperreports
PostgreSQL
Home | Monitoring | Statistics | Security | SQL | Scheduler | Messaging | MetaData | Procedural Logic
Log Off

EnterpriseDB JMS Destinations

Create JMS Destination

Name

Type ● Topic ○ Queue

Create

Deployed Destinations		
Name	Type	
JBossWorld	[Queue]	Delete

www.enterprisedb.com
Acknowledgements
© All Rights Reserved EnterpriseDB ®
Done