

Programación Distribuida

Java Message Service

JAIME SALVADOR



AGENDA

- **Introducción**
- Middleware Orientado a Mensajes (MOM)
- JMS
- ActiveMQ

Introducción

- Todas las formas de comunicación (entre aplicaciones) vistas hasta el momento son sincrónicas
 - Sockets
 - RMI
 - HTTP Invoker
 - Hessian/Burlap

Introducción

- Todas las formas de comunicación (entre aplicaciones) vistas hasta el momento son sincrónicas
 - Sockets
 - RMI
 - HTTP Invoker
 - Hessian/Burlap

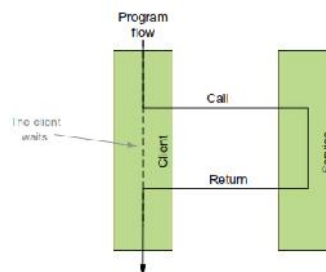


Figure 1.7.1. When communicating synchronously, the client must wait for the operation to complete.

Introducción

- Existe otra forma de comunicación: ASINCRONICA:

Envío de mensajes de una aplicación a otra aplicación sin esperar una respuesta

- JMS: Java Message Service
- AMQP: Advanced Message Queuing Protocol

AGENDA

- Introducción
- **Middleware Orientado a Mensajes (MOM)**
- JMS
- ActiveMQ

MOM

- MOM es un software que envía y recibe mensajes de forma confiable entre varios sistemas
- El uso más común es para la integración de sistemas
- Cuando un cliente envía un mensaje, MOM almacena el mensaje en una ubicación específica indicada por el que envía el mensaje e informa que el mensaje fue recibido

MOM

- El que envía el mensaje se denomina **Producer** (*Productor*)
- El que recibe el mensaje se denomina **Consumer** (*Consumidor*)
- El lugar en el que deposita el mensaje se denomina **Destination** (*Destino*)

MOM

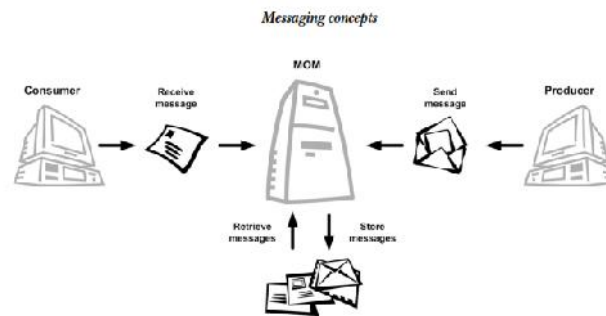


Figure 4.1 Basic MOM flow. When the producer sends a message to the middleware, it's stored immediately and later collected by the consumer.

MOM

- Existen varias implementaciones de MOM
- Todos los servidores Java EE incluyen un Sistema de Mensajería (MOM)
- Existen servidores autónomos: ActiveMQ

MOM

	Tomcat	TomEE	TomEE JAXRS	TomEE+	TomEE PluME	OpenEJB
Java Servlets	✓	✓	✓	✓	✓	
Java ServerPages (JSP)	✓	✓	✓	✓	✓	
Java ServerFaces (JSF)		✓	✓	✓	✓	
Java Transaction API (JTA)		✓	✓	✓	✓	✓
Java Persistence API (JPA)		✓	✓	✓	✓	✓
Java Contexts and Dependency Injection (CDI)		✓	✓	✓	✓	✓
Java Messaging Service (JMS)				✓	✓	✓
EclipseLink					✓	
Mojara					✓	

MOM

- Mensaje
Unidades de información de negocio las cuales son enviadas desde un Sistema a otro Sistema
- Protocolos
HTTPS/S, TCP/IP
- Clientes: múltiples lenguajes
- API: para envío y recepción de mensajes

MOM

Modelos de mensajería

- Define cómo los productores y consumidores intercambian mensajes
- JMS: Existen dos modelos estándar: *point-to-point* (PTP) y *publish-subscribe*

MOM

Point-to-Point

- Un mensaje viaja desde un productor (punto A) a un solo consumidor (Punto B)
- Pueden existir varios productores y consumidores, pero solo un consumidor consume el mensaje

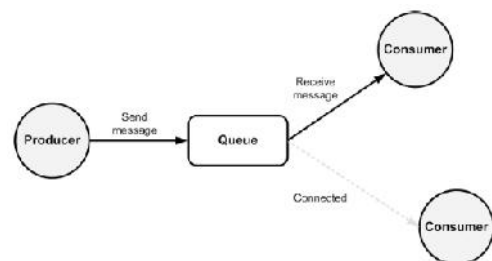


Figure 4.4 PTP messaging model with one producer and two consumers

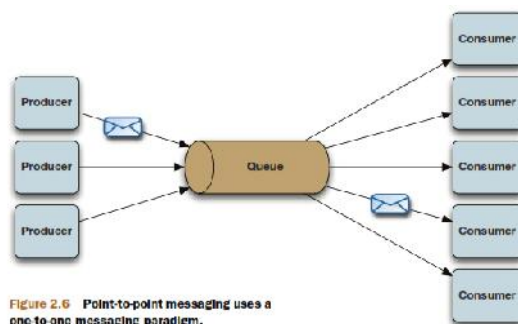
MOM

Point-to-Point

- Los destinos PTP se denominan **colas** (*colas*)
- Los productores escriben en la cola y los consumidores leen de la cola
- No se garantiza el orden en el que se consumen los mensajes

MOM

Point-to-Point



MOM

Point-to-Point

- Message in a bottle
 - Productor: naufrago
 - Queue: océano
 - Consumidor: bañista



MOM

Publish/Subscribe (pub-sub)

- Un productor genera un mensaje el cual es recibido por varios consumidores

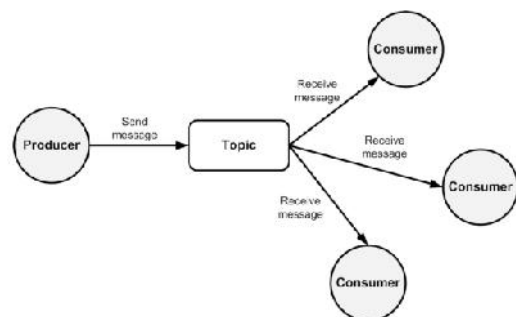


Figure 4.5 The pub-sub messaging model with one producer and three consumers. Each topic subscriber receives a copy of the message.

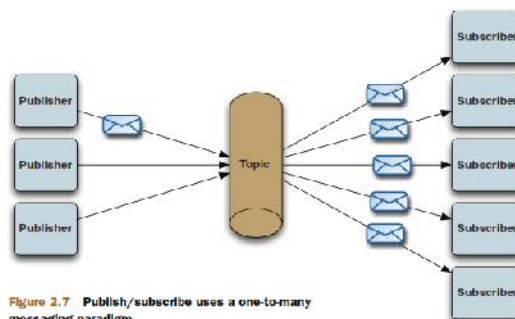
MOM

Publish/Subscribe (pub-sub)

- Los destinos pub-sub se denominan **topic** (tópico)
- Un consumidor se denomina **subscriber** (suscriptor)

MOM

Publish/Subscribe



AGENDA

- Introducción
- Middleware Orientado a Mensajes (MOM)
- **JMS**
- ActiveMQ

JMS

- Java Message Service
- Proporciona un API para acceso estándar a sistemas MOM
- API estándar para enviar y recibir mensajes utilizando Java
- **JMS no es un MOM!!!**

JMS

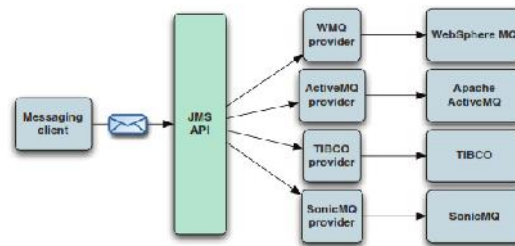


Figure 2.4 JMS allows a single client to easily connect to many JMS providers.

JMS

JMS define varios conceptos

- JMS client
- Non-JMS client
- JMS producer
- JMS consumer
- JMS provider
- JMS message
- JMS domains
- Administered objects
- Connection factory
- Destination

JMS

JMS client

- Los clientes utilizan el API JMS para interactuar con el MOM (similar a JDBC)
- `MessageProducer` : para enviar mensajes a los destinos
- `MessageConsumer`: para consumer mensajes de los destinos

JMS

Non-JMS client

- Los clientes utilizan el API proporcionado por el MOM (no estándar)
- Es posible utilizar el protocolo CORBA IIOP o RMI

JMS

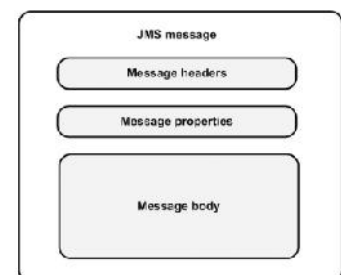
JMS provider

- Es el fabricante que implementa el MOM e implementa el API JMS
- Proporciona acceso al MOM a través del API JMS
- Ej. ActiveMQ: <http://activemq.apache.org>
- Ej. ArtemisMQ: <https://activemq.apache.org/artemis/>

JMS

JMS message

- Es la forma en la cual se transmite información del negocio
- Permite el envío de texto, datos binarios, etc.
- Contiene dos partes: header y body



JMS

JMS domains

- Point-to-point (PTP)
- Publish/Susbcribe (pub-sub)

JMS

Administered Objects

- Connection Factory: permite crear conexiones al JMS provider

`ConnectionFactory`

- Destination

AGENDA

- Introducción
- Middleware Orientado a Mensajes (MOM)
- JMS
- **ActiveMQ**

ActiveMQ

ActiveMQ es un MOM open source el cual implementa la especificación JMS



<http://activemq.apache.org>

PREGUNTAS?



Bibliografía

- Craig Walls. Spring in Action, Fourth Edition. Manning, 2015.
- Bruce Snyder, Dejan Bosanac, Rob Davies. ActiveMQ in Action. Manning, 2011.
- <http://activemq.apache.org> , consultada el 2015-06-18,