

JAVA MESSAGE SERVICE (JMS)

Veerle Ongenae

Overzicht

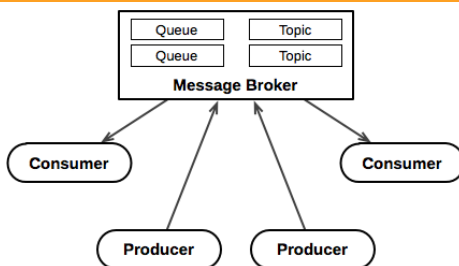
2

- Wat is JMS?
- JMS API
 - Configuratie
 - Producers
 - Consumers
 - Listeners
 - Subscriptions
 - Messages
 - Browsers

Industrieel Ingenieur Informatica, UGent

Messaging

3



<http://torquebox.org/builds/html-docs/messaging.html>

Industrieel Ingenieur Informatica, UGent

JMS

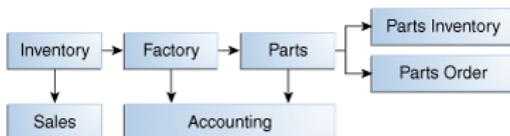
4

- Java Message Service
 - Berichten
 - Asynchroon
 - Betrouwbaar

Industrieel Ingenieur Informatica, UGent

Gebruik JMS

5



<https://docs.oracle.com/javase/7/tutorial/jms-concepts001.htm#BNCDR>

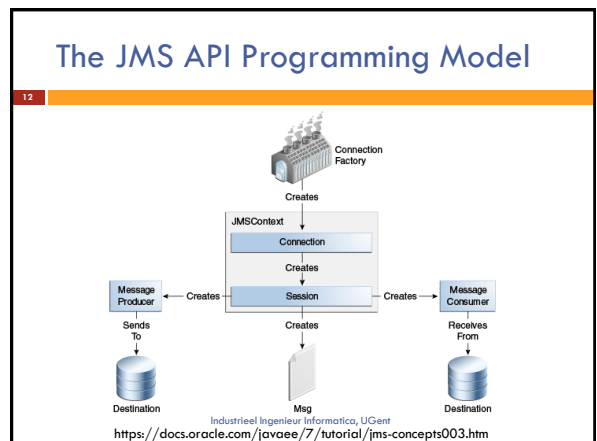
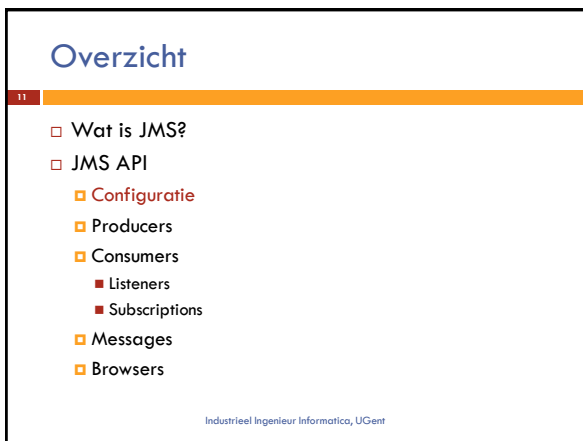
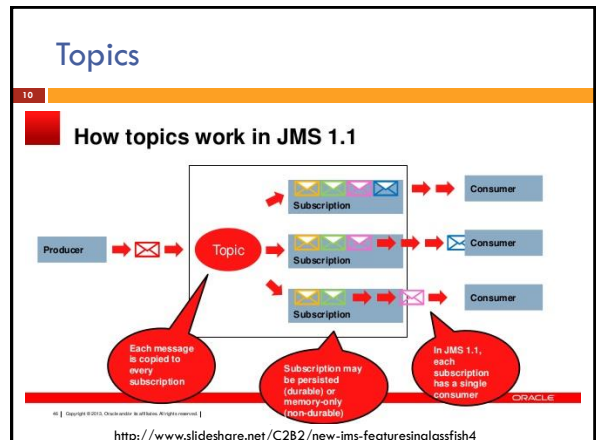
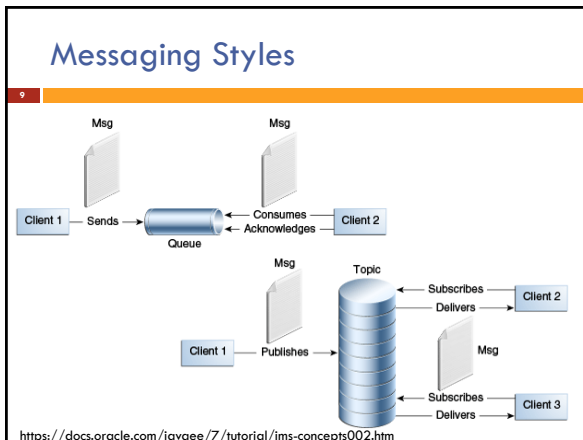
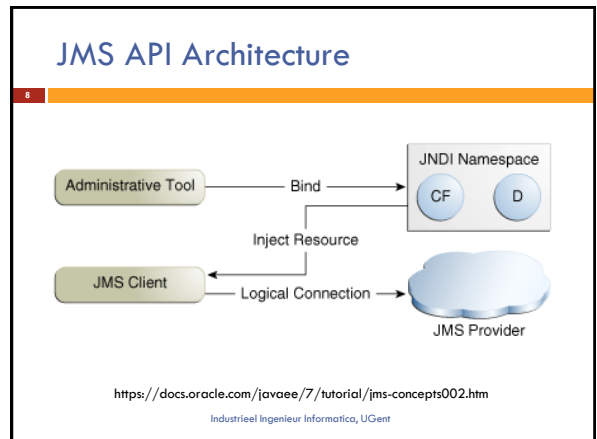
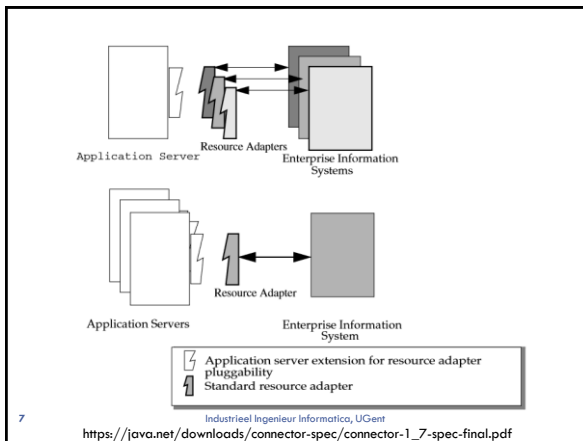
Industrieel Ingenieur Informatica, UGent

JMS versus J2EE

6

- JMS API deel van J2EE
- Verschillende componenten kunnen berichten sturen en ontvangen
- Clients kunnen luisteren naar berichten
- Combineerbaar met JTA (Java Transaction API)

Industrieel Ingenieur Informatica, UGent



Overzicht

19

- Wat is JMS?
- JMS API
 - Configuratie
 - Producers
 - Consumers
 - Listeners
 - Subscriptions
 - Messages
 - Browsers

Industrieel Ingenieur Informatica, UGent

JMS Message Consumers

20

```
try (JMSContext context = connectionFactory.createContext();) {
    JMSConsumer consumer = context.createConsumer(dest);
    while (true) {
        Message m = consumer.receive(1000);
        if (m != null) {
            if (m instanceof TextMessage) {
                System.out.println("Reading message: " +
                    m.getBody(String.class));
            } else {break;}
        }
    }
} catch (JMSRuntimeException e) {
    ...
}
```

Consumer: voorbeeld

21

```
C:\Users\vongenae\Documents\Gedistribueerde toepassingen\voorbeelden\jms\SynchCo
nsumer>appclient -client dist\SynchConsumer.jar queue
pkt 26, 2015 2:17:08 PM org.hibernate.validator.internal.util.Version <clinit>
INFO: HV000001: Hibernate Validator 5.0.0.Final
pkt 26, 2015 2:17:09 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter: Version: 5.1 (Build 9
-b) Compile: July 29 2014 1229
pkt 26, 2015 2:17:09 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter starting: broker is REMO
TE, connection mode is TCP
pkt 26, 2015 2:17:09 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter Started:REMOTE
Destination type is queue
Reading message: This is message 1 from producer
Reading message: This is message 2 from producer
Reading message: This is message 3 from producer
Messages received: 3
```

Industrieel Ingenieur Informatica, UGent

Queue met 2 consumers

22

```
C:\Users\vongenae\Documents\Gedistribueerde toepassingen\voorbeelden\jms\SynchCo
nsumer>appclient -client dist\SynchConsumer.jar queue
pkt 26, 2015 2:17:08 PM org.hibernate.validator.internal.util.Version <clinit>
INFO: HV000001: Hibernate Validator 5.0.0.Final
pkt 26, 2015 2:17:09 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter: Version: 5.1 (Build 9
-b) Compile: July 29 2014 1229
pkt 26, 2015 2:17:09 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter starting: broker is REMO
TE, connection mode is TCP
pkt 26, 2015 2:17:09 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter Started:REMOTE
Destination type is queue
Reading message: This is message 1 from producer
Reading message: This is message 2 from producer
Reading message: This is message 3 from producer
Reading message: This is message 4 from producer
Reading message: This is message 5 from producer
Reading message: This is message 6 from producer
Reading message: This is message 7 from producer
Reading message: This is message 8 from producer
Reading message: This is message 9 from producer
Reading message: This is message 10 from producer
Reading message: This is message 11 from producer
Reading message: This is message 12 from producer
Messages received: 12
```

Industrieel Ingenieur Informatica, UGent

Overzicht

23

- Wat is JMS?
- JMS API
 - Configuratie
 - Producers
 - Consumers
 - Listeners
 - Subscriptions
 - Messages
 - Browsers

Industrieel Ingenieur Informatica, UGent

Asynchronous consumers

24

```
try (JMSContext context = connectionFactory.createContext();) {
    JMSConsumer consumer = context.createConsumer(dest);
    TextListener listener = new TextListener();
    consumer.setMessageListener(listener);

} catch (JMSRuntimeException e) {
    ...
}
```

JMS Message Listeners

```

25
public class TextListener implements MessageListener {
    @Override
    public void onMessage(Message m) {
        try {
            if (m instanceof TextMessage) {
                ...
            } else {
                ...
            }
        } catch (JMSException | JMSRuntimeException e) {
            ...
        }
    }
}

```

Consumer: voorbeeld

```

26
C:\Users\vongeneae\Documents\Gedistribueerde toepassingen\voorbeelden\jms\AsynchC
onsumer>apclient -client dist\AsynchConsumer.jar topic
okt 26, 2015 2:29:37 PM org.hibernate.validator.internal.util.Version <clinit>
INFO: HV000001: Hibernate Validator 5.0.0.Final
okt 26, 2015 2:29:37 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter: Version: 5.1 (Build 9
-b) Compile: July 29 2014 1229
okt 26, 2015 2:29:37 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter starting: broker is REMO
TE, connection mode is TCP
okt 26, 2015 2:29:37 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter Started:REMOTE
Destination type is topic
To end program, enter Q or q, then <return>

```

Industrieel Ingenieur Informatica, UGent

```

To end program, enter Q or q, then <return>
Reading message: This is message 1 from producer
Reading message: This is message 2 from producer
Reading message: This is message 3 from producer
Reading message: This is message 4 from producer
Reading message: This is message 5 from producer
Reading message: This is message 6 from producer
Reading message: This is message 7 from producer
Reading message: This is message 8 from producer
Reading message: This is message 9 from producer
Reading message: This is message 10 from producer
Reading message: This is message 11 from producer
Reading message: This is message 12 from producer
Reading message: This is message 13 from producer
Reading message: This is message 14 from producer
Reading message: This is message 15 from producer
Reading message: This is message 16 from producer
Reading message: This is message 17 from producer
Message is not a TextMessage
Reading message: This is message 1 from producer
Reading message: This is message 2 from producer
Reading message: This is message 3 from producer
Reading message: This is message 4 from producer
Reading message: This is message 5 from producer
Reading message: This is message 6 from producer

```

27

Message Properties en selectors

```

28
Message msg = ...

```

```

// Setting message properties
msg.setStringProperty("title", "Thriller");
msg.setIntProperty("releaseYear", 1982);

```

```

JMSContext ctx = ...
MessageConsumer consumer
    = ctx.createConsumer(queue, "releaseYear < 1980");

```

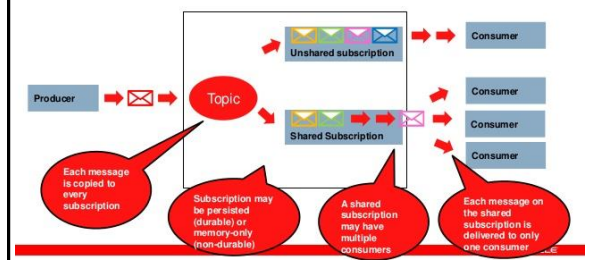
Overzicht

- Wat is JMS?
- JMS API
 - Configuratie
 - Producers
 - Consumers
 - Listeners
 - Subscriptions
 - Messages
 - Browsers

Industrieel Ingenieur Informatica, UGent

Consuming Messages from Topics

Shared subscriptions in JMS 2.0



© 2015 Oracle and/or its affiliates. All rights reserved.

<http://www.slideshare.net/C282/new-jms-featuresin-glassfish4>

Shared Subscriptions

```
try (JMSContext context = connectionFactory.createContext()) {
    consumer = context.createSharedConsumer(topic, "SubName");

    listener = new TextListener();
    consumer.setMessageListener(listener);

    ...
} catch (JMSRuntimeException e) { ... }
```

```
try (JMSContext context = connectionFactory.createContext()) {
    consumer
        = context.createSharedDurableConsumer(topic, "MakeItLast");

    listener = new TextListener();
    consumer.setMessageListener(listener);
    ...
} catch (JMSRuntimeException e) { ... }
```

Overzicht

37

- Wat is JMS?
- JMS API
 - Configuratie
 - Producers
 - Consumers
 - Listeners
 - Subscriptions
 - Messages
 - Browsers

Industrieel Ingenieur Informatica, UGent

Header Field	Description	Set By
JMSDestination	Destination to which the message is being sent	JMS provider <code>send</code> method
JMSDeliveryMode	Delivery mode specified when the message was sent (see Specifying Message Persistence)	JMS provider <code>send</code> method
JMSDeliveryTime	The time the message was sent plus the delivery delay specified when the message was sent (see Specifying a Delivery Delay)	JMS provider <code>send</code> method
JMSExpiration	Expiration time of the message (see Allowing Messages to Expire)	JMS provider <code>send</code> method
JMSPriority	The priority of the message (see Setting Message Priority Levels)	JMS provider <code>send</code> method
JMSMessageID	Value that uniquely identifies each message sent by a provider	JMS provider <code>send</code> method
JMSTimestamp	The time the message was handed off to a provider to be sent	JMS provider <code>send</code> method
JMSCorrelationID	Value that links one message to another, commonly the <code>JMSMessageID</code> value is used	Client application
JMSReplyTo	Destination where replies to the message should be sent	Client application
JMSType	Type identifier supplied by client application	Client application
JMSRedelivered	Whether the message is being redelivered	JMS provider prior to delivery

Message Body

39

Message Type	Body Contains
<code>TextMessage</code>	A <code>java.lang.String</code> object (for example, the contents of an XML file).
<code>MapMessage</code>	A set of name-value pairs, with names as <code>String</code> objects and values as primitive types in the Java programming language. The entries can be accessed sequentially by enumerator or randomly by name. The order of the entries is undefined.
<code>ByteMessage</code>	A stream of uninterpreted bytes. This message type is for literally encoding a body to match an existing message format.
<code>StreamMessage</code>	A stream of primitive values in the Java programming language, filled and read sequentially.
<code>ObjectMessage</code>	A <code>Serializable</code> object in the Java programming language.
<code>Message</code>	Nothing. Composed of header fields and properties only. This message type is useful when a message body is not required.

Berichten aanmaken

40

```
TextMessage message = context.createTextMessage();
message.setText(msg_text);    // msg_text is a String
context.createProducer().send(dest, message);
```

```
String message = "This is a message";
context.createProducer().send(dest, message);
```

```
context.createProducer().send(dest,
    context.createMessage());
```

Industrieel Ingenieur Informatica, UGent

Berichten ontvangen

41

```
Message m = consumer.receive();
if (m instanceof TextMessage) {
    String message = m.getBody(String.class);
    System.out.println("Reading message: " + message);
} else {
    // Handle error or process another message type
}
```

```
String message = consumer.receiveBody(String.class);
```

Overzicht

42

- Wat is JMS?
- JMS API
 - Configuratie
 - Producers
 - Consumers
 - Listeners
 - Subscriptions
 - Messages
 - Browsers

Industrieel Ingenieur Informatica, UGent

JMS Message Browsers

```
43
try (JMSContext context = connectionFactory.createContext()); {
    QueueBrowser browser = context.createBrowser(queue);
    Enumeration msgs = browser.getEnumeration();

    if (!msgs.hasMoreElements()) {
        System.out.println("No messages in queue");
    } else {
        while (msgs.hasMoreElements()) {
            Message tempMsg = (Message) msgs.nextElement();
            System.out.println("\nMessage: " + tempMsg);
        }
    }
} catch (JMSRuntimeException e) { ... }
```

```
44
C:\Users\vongenae\Documents\Gedistribueerde toepassingen\voorbeelden\jms\Message
Browser>appclient -client dist\MessageBrowser.jar
okt 26, 2015 2:37:23 PM org.hibernate.validator.internal.util.Version <clinit>
INFO: HV000001: Hibernate Validator 5.0.0.Final
okt 26, 2015 2:37:23 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter: Version: 5.1 (Build 9
-b) Compile: July 29 2014 1229
okt 26, 2015 2:37:23 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter starting: broker is REMO
TE, connection mode is TCP
okt 26, 2015 2:37:23 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter Started:REMOTE
No messages in queue
```

Industrieel Ingenieur Informatica, UGent

```
45
C:\Users\vongenae\Documents\Gedistribueerde toepassingen\voorbeelden\jms\Message
Browser>appclient -client dist\MessageBrowser.jar
okt 26, 2015 2:38:28 PM org.hibernate.validator.internal.util.Version <clinit>
INFO: HV000001: Hibernate Validator 5.0.0.Final
okt 26, 2015 2:38:28 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter: Version: 5.1 (Build 9
-b) Compile: July 29 2014 1229
okt 26, 2015 2:38:28 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter starting: broker is REMO
TE, connection mode is TCP
okt 26, 2015 2:38:28 PM com.sun.messaging.jms.ra.ResourceAdapter start
INFO: MQJMSRA_RA1101: GlassFish MQ JMS Resource Adapter Started:REMOTE
Message:
Text: This is message 1 from producer
Class: com.sun.messaging.jmq.jmsclient.TextMessageImpl
getJMSMessageID(): ID:9-192.168.56.1(b2:3:ea:b1:f2:e5)-7388-1445866678820
getJMSTimestamp(): 1445866678820
getJMSCorrelationID(): null
JMSReplyTo: null
JMSDestination: PhysicalQueue
getJMSDeliveryMode(): PERSISTENT
getJMSRedelivered(): false
getJMSType(): null
getJMSExpiration(): 0
```