

**Verteilte Systeme
WS 2015/16**

Übungsblatt 11

In dieser Übung wird aktuelles Messaging am Beispiel der Message-Oriented Middleware (MOM)-Plattform ActiveMQ vertieft. Diese realisiert einen Message Broker, der über die Java Message Service Schnittstelle (JMS) entsprechend dem Publish/Subscribe-Modell genutzt werden soll.

Web-Ressourcen:

[1] ActiveMQ

<http://activemq.apache.org/>

[2] Java Message Service

Einführung: http://de.wikipedia.org/wiki/Java_Message_Service

Tutorial: <http://docs.oracle.com/javase/6/tutorial/doc/bncdq.html>

[3] JMS 1.1 Spec und API Dokumentation

<http://www.oracle.com/technetwork/java/docs-136352.html>

Aufgabe 11.1 (Installation von ActiveMQ)

Von der ActiveMQ-Website [1] können Sie die aktuelle Version (5.13.0) als .jar herunterladen und unmittelbar als Server nutzen. Beachten Sie die Informationen unter Getting Started und die gegebenen Installationshinweise. Sehen Sie den Transport über TCP vor. Eine Persistenz der Nachrichten oder Transaktionsverarbeitung werden für das hier betrachtete Beispiel nicht benötigt. Beachten Sie, dass Sie mit der bereitgestellten Oberfläche (Web-Konsole) den aktuellen Zustand des Brokers beobachten können.

Aufgabe 11.2 (JMS Pub/Sub-Modell)

Die MOM-Plattform ActiveMQ besitzt in der verwendeten Version eine JMS 1.1 – konforme Programmierschnittstelle. Machen Sie sich z.B. anhand der angegebenen Dokumentation [2] und [3] kundig über JMS und seiner Nutzung auf der Basis des Topic-basierten Publisher/Subscriber-Modells. Wesentliche Klassen (vgl. Dokumentation) sind dabei `Topic`, `Session`, `Connection`, `MessageProducer`, `MessageSubscriber`. Für den Empfang einer Nachricht implementiert man einen `MessageListener.onMessage(Message)` EventHandler oder nutzt `MessageConsumer.receive()`. Beachten Sie zudem die Möglichkeiten, JMS-Nachrichten insbesondere vom Typ `MapMessage` zu erzeugen und zu verarbeiten. Für die Nutzung von JMS mit ActiveMQ ist die Verwendung von JNDI nicht erforderlich, die `ActiveMQConnectionFactory` kann direkt instantiiert werden.

Aufgabe 11.3 (Primzahlen)

- (a) Publisher: Implementieren Sie in einem eigenen Projekt als ersten Publisher einen Primzahlerzeuger. Dieser erzeuge die Folge der Primzahlen und generiere daraus MapMessages mit dem JMS Header Feld JMSTimeStamp und mit den folgenden Message Properties: laufende Nummer der Primzahl, Primzahl, Rechenzeit zur Bestimmung der Primzahl. Der genaue Aufbau der Nachrichten ist selbst festzulegen. Der Publisher veröffentliche die Folge der Primzahlen unter dem Topic "Primzahlen". Zwischen zwei erzeugten Primzahlen schlafe der Publisher für eine konfigurierbare Zeit (z.B. 1 Sekunde als Default-Wert).
- (b) Subscriber: Implementieren Sie einen Subscriber für "Primzahlen", der, wenn immer eine neue Primzahl-Message verfügbar ist, diese mittels des Listeners mit allen Attributen auf Standardausgabe ausgabe.

Aufgabe 11.4 (Primzahlzwillinge)

Unter Primzahlzwillingen versteht man ein Paar von Primzahlen p_1 und p_2 , wobei $p_2 = p_1 + 2$ ist. Die ersten Primzahlzwillinge sind also (3,5), (5,7), (11,13), Eine kleine Einführung finden Sie unter <http://de.wikipedia.org/wiki/Primzahlzwillig>. In dieser Aufgabe soll die Bestimmung von Primzahlzwillingen dadurch erfolgen, dass der Abstand zwischen je zwei aufeinander folgenden Primzahlen in der Folge aller Primzahlen beobachtet wird. Ist dieser 2, so ist ein neuer Primzahlzwillig gefunden.

- (a) Publisher: Implementieren Sie als weiteren Publisher in einem eigenen Projekt einen Primzahlzwillinge-Erzeuger. Dieser sei Subscriber des Topics "Primzahlen" und gleichzeitig Publisher des Topics "Primzahlzwillinge". Er generiere ebenfalls MapMessages mit dem Header Feld JMSTimeStamp und mit den folgenden Message Properties: laufende Nummer des Primzahlzwillings, Primzahl p_1 , Primzahl p_2 , Rechenzeit zur Bestimmung der Primzahlzwillinge als Summe der Rechenzeiten der beiden Primzahlen p_1 und p_2 .
- (b) Subscriber: Implementieren Sie einen Subscriber für "Primzahlzwillinge", der, wenn immer eine neue Message verfügbar ist, diese mit allen Attributen auf Standardausgabe ausgabe.
- (c) Konfiguration und Test: Testen Sie eine lokale Konfiguration und eine verteilte mit jeweils mehreren Subscribern, in der die Publisher nicht lokal zum Server laufen. Beobachten Sie Ihre Konfiguration mit der in ActiveMQ eingebauten Web Console. Geben Sie jeweils einen Screenshot mit ab.

Aufgabe 11.5 (optional, sichere Verbindungen)

Stellen Sie Ihre Konfiguration so um, dass alle Verbindungen zum Message Broker nur noch über SSL erfolgen.