

Übungsaufgabe #6: Library - JXTA

17.01.2008

Ziel dieser Aufgabe ist es die bisherige Bibliotheksverwaltung durch einen einfachen JXTA-Service zu realisieren. Hierzu kann auf den Ergebnissen der Aufgabe 1 oder 4 aufgebaut werden. Die eigentliche Verwaltung der Medienobjekte erfolgt nach wie vor in einer Client/Server Architektur. Für das Lokalisieren und Binden eines Servers werden jedoch Peer-to-Peer Mechanismen der JXTA Plattform verwendet.

a) JXTA-Client

Implementieren sie zuerst einen Client der sich mit der *Net Peer Group* verbindet. Danach soll der *Discovery Service* dieser Gruppe genutzt werden um *Service Advertisements* zu entdecken die einen Dienst mit dem Namen `LibraryService` beschreiben. Stellen sie hierzu eine entsprechende Anfrage an den Discovery Service und spezifizieren sie die Anfrage nach Tag: Name und Value: `LibraryService`.

Realisieren sie ein Kommando `servers` im `LibraryFrontend` welche die Discovery Prozedur aufruft und die gefundenen Instanzen auflistet. Zur Unterscheidung kann der Parameter `Creator (Ctrr)` des *Module Specification Advertisements* genutzt werden. Im Folgenden soll nun das `LibraryFrontend` durch das Kommando `connect` erweitert werden. Dieses ermöglicht es eine Verbindung zu einem der gefundenen Server-Instanzen aufzubauen. Extrahieren sie hierzu das *Pipe Advertisement* aus dem ausgewählten *Module Specification Advertisement*. Binden sie anschließend das Advertisement mit einer bidirektionalen `JxtBiDiPipe`.

b) Protokoll

Der Nachrichtenaustausch zwischen Client und Server erfolgt über Messages nach einem einfachen zeichenbasierten Protokoll. Die Message-Objekte werden hierfür mittels Instanzen der Klasse `StringMessageElement` befüllt. Implementieren sie auf Client-Seite das folgende Protokoll:

Registrieren

```
Anfrage:      Tag: operation      Value: register
              Tag: title      Value: <Title>
Antwort:      Tag: status      Value: success | failed
```

Liste

```
Anfrage:      Tag: operation      Value: list
Antwort:      Tag: title      Value: <Title>
```

Die Antwort kann eine beliebige Anzahl an title-Tags beinhalten.

Ausleihen

```
Anfrage:      Tag: operation      Value: borrow
              Tag: title      Value: <Title>
Antwort:      Tag: status      Value: success | failed
```

Rückgabe

```
Anfrage:      Tag: operation      Value: return
              Tag: title      Value: <Title>
Antwort:      Tag: status      Value: success | failed
```

c) Einfacher JXTA-Service

Nun sollen sie einen einfachen JXTA-Service implementieren. Hierzu ist es zuerst nötig ein *ModulClass Advertisement* und ein *ModuleSpec Advertisement* zu erstellen. Erzeugen sie für beide Advertisements entsprechende IDs und verwenden sie sie für alle zukünftigen Programmläufe. Der Name des *ModulClass Advertisements* soll `JXTAMOD:LibraryService` lauten. Das *ModuleSpec Advertisement* ist mit folgenden Angaben zu befüllen:

```
Name: LibraryService
Version: 1.0
Creator: <login>
SpecURI:
    http://www4.informatik.uni-erlangen.de/Lehre/WS07/V_MW/Ueb-
    ung/aufgaben/a6.pdf
```

Um das *Module Specification Advertisement* zu vervollständigen benötigen sie noch ein *Pipe Advertisement*. Implementieren sie hierzu die Klasse `PipeAdvertisementCreator` welche ein Advertisement erzeugt und in eine Datei schreibt (vgl. `/proj/i4mw/pub/aufgabe6/pipe.adv`). Diese Datei kann dann durch den Service ausgelesen werden. Publizieren sie anschließend ihren Service durch den Discovery Service.

Abschließend muss nun der Service noch das *Pipe-Advertisement* binden und auf ankommende Verbindungen warten. Verwenden sie hierzu eine `JxtaServerPipe` und implementieren sie das Protokoll von Teilaufgabe b) für die Server-Seite.

Anmerkung: Die Aufgabenstellung verwendet Advertisements konzeptionell nicht völlig richtig. Dies geschieht aus Gründen des Aufwandes und des praktischen Nutzens.

Bearbeitung: bis zum 29.01.2008/20:00 Uhr

Alle notwendigen Quelldateien müssen im SVN-Repository eingechekkt sein (2er Gruppen).