

# Разработка сетевых приложений на Java

Алексей Владыкин

5 декабря 2012

1 Сокеты

2 URL и URI

3 Netty

- 1 Сокеты
- 2 URL и URI
- 3 Netty

- Низкоуровневый API для пересылки байтов по сети
- Поддерживаются протоколы TCP и UDP
- Поддерживается адресация IPv4 (213.180.204.3) и IPv6 (2001:0db8:11a3:09d7:1f34:8a2e:07a0:765d)

# java.net.Socket

- Клиентский сокет, устанавливает TCP соединение с сервером

```
Socket socket = new Socket("localhost", 11111);

OutputStream os = socket.getOutputStream();
os.write(requestBytes);
os.flush();

InputStream is = socket.getInputStream();
is.read(responseBytes);
```

## java.net.ServerSocket

- Серверный сокет, ожидает подключений от клиентов

```
ServerSocket server = new ServerSocket(11111);  
Socket socket = server.accept();  
  
InputStream is = socket.getInputStream();  
is.read(requestBytes);  
  
OutputStream os = socket.getOutputStream();  
os.write(responseBytes);  
os.flush();
```

- 1 Сокеты
- 2 URL и URI
- 3 Netty

# java.net.URI

- Uniform Resource Identifier
- Примеры:
  - `mailto:java-net@java.sun.com`
  - `urn:isbn:096139210x`
  - `http://java.sun.com/j2se/1.3/docs/guide/collections/designfaq.html#28`
- Синтаксические операции: `resolve`, `relativize`

# java.net.URL

- Uniform Resource Locator
- Примеры:
  - `http://java.sun.com/j2se/1.3/`
  - `file:/home/av/projects/`
- Поддерживает операции доступа:  
`openConnection()`, `openStream()`

- 1 Сокеты
- 2 URL и URI
- 3 Netty**

# Channel

- `org.jboss.netty.channel.Channel`
- Аналог `java.net.Socket`
- Представляет конкретное подключение
- Позволяет писать и читать данные (асинхронно)

# ChannelPipeline

- `org.jboss.netty.channel.ChannelPipeline`
- Цепочка обработчиков `ChannelHandler`, связанная с каналом
- `Upstream` — направление из сети в программу,  
`downstream` — направление из программы в сеть

# ChannelEvent

- `org.jboss.netty.channel.ChannelEvent`
- Событие, связанное с каналом
- Upstream: `messageReceived`, `exceptionCaught`, `channelOpen`, `channelClosed`
- Downstream: `write`, `connect`, `disconnect`, `close`

# ChannelFuture

- `org.jboss.netty.channel.ChannelFuture`
- Аналог `java.util.concurrent.Future`
- Представляет результат асинхронной операции