Dealing with Hybrid Streams of Text and Bytes

José Paumard
PHD, JAVA CHAMPION, JAVA ROCK STAR

@JosePaumard https://github.com/JosePaumard

Agenda

How to write text on binary streams

How to write both text and bytes on binary streams

Using the Java I/O API to use sockets

Introducing Hybrid Streams

Two new classes:

- InputStreamReader that extends Reader
- OutputStreamWriter that extends Writer

InputStreamReader is a Reader

So it can read chars and Strings

But it is created on an InputStream

And can take a CharSet as a 2nd parameter

OutputStreamWriter is a Writer

So it can write chars and Strings

But it is created on an OutputStream

And can take a CharSet as a 2nd parameter

```
InputStream is = ...;
InputStreamReader isr = new InputStreamReader(is, StandardCharsets.UTF_8);

LineNumberReader lnr = new LineNumberReader(isr);

String line = lnr.readLine();
int lineNumber = lnr.getLineNumber();
```

Since an input stream reader is a reader…

It can be used to create other readers, using the decorator pattern

And these readers can be used as usual

```java
OutputStream os = ...;
OutputStreamWriter osw = new OutputStreamWriter(os, StandardCharsets.UTF_8);

PrintWriter pw = new PrintWriter(osw);

pw.printf("Hello %s\n", person.getFirstName());
```

And the same goes for writers

It can be used to create other writers, using the decorator pattern

And these readers can be used as usual

Those two classes InputStreamReader and OutputStreamWriter

Enable the writing / reading of chars

On streams of bytes

And since several streams can be opened on the same file, in-memory array, etc…

It allows for the creation of hybrid structures

Mixing strings / chars and raw bytes on the same stream

Hybrid Streams Case Study

We have a list of all the Aesop’s fables:

- each fable has a title

- and a text, which can be a little long

- there is a lot of fables

```
Aesop's Fables
291
   1235     123 The Wolf and the Lamb
   3271     245 The Bat and the Weasels
```

The format is the following

- a magic ID

- the total number of fables in plain text

- then for each fable: its position (aligned), length (aligned) and title

```
Aesop's Fables
291
   1235     123 The Wolf and the Lamb
   3271     245 The Bat and the Weasels
<binary texts>
```

Then, in binary compressed format, the text of each fable, one by one

Demo

Let us see some code!

Let us create a writer method that takes a list of fables, and produces this file

Creating Streams Over the Network

Passing data from one machine to another across a network can be done with sockets

A socket listens for incoming requests on a machine

When a request arrives, it can create two streams:

- an input stream, to read data from it

- an output stream, to send data to the other machine

```java
ServerSocket serverSocket = new ServerSocket(12345);
Socket socket = serverSocket.accept();

InputStream  inputStream  = socket.getInputStream();
OutputStream outputStream = socket.getOutputStream();
```

First we create a server socket, that listens to a given port

Then we put this socket on a listen state, the accept call is blocking

The rest is pure Java I/O

```java
Socket socket = new Socket();
InetSocketAddress address = new InetSocketAddress("127.0.0.1", 12345);
socket.connect(address);

InputStream  inputStream  = socket.getInputStream();
OutputStream outputStream = socket.getOutputStream();
```

Then we create a client socket, that connects to the same port

Then we can send requests and get responses from the server socket using pure Java I/O

... and maybe a little Java Thread

Demo

Let us see some code!

Let us create a server and a client socket and show how they can communicate together

There are many parts of the JDK that relies on Java IO to communicate:

- network (socket)
- HTTP support with servlets (Java EE)
- database (JDBC)

Module Wrap Up

What did you learn?

Hybrid Streams!

How to communicate through the network using Socket and Java I/O

How to create complex file formats mixing text and binary information

This kind of format is used in JPEG, MPEG and many other standard file formats

Course Wrap Up

The Java I/O API is at the basis of many other API

It is very useful to write low level code

There are hurdles when using it, the use of charsets is one of them

It can be use to transfer objects from one application to another

It is also used to generate XML or JSON

There are extensions to it, NIO and NIO2

Course Wrap Up

Thank you!

@JosePaumard

https://github.com/JosePaumard