



Programowanie komputerów I

Programowanie sieciowe

Willy Picard

Katedra Technologii Informacyjnych
Akademia Ekonomiczna w Poznaniu

<picard@kti.ae.poznan.pl>

Agenda

- ▶ Cel(e) wykładu
- ▶ Protokoły
- ▶ `java.net.URL`
- ▶ Gniazda (*Sockets*)
- ▶ Przykłady
- ▶ Podsumowanie

Cel(e) wykładu



Przegląd wykładu

Java

- ▶ 8: Przykład podsumowujący
- ▶ 9: Pakiet standardowy
- ▶ 10: Interfejsy graficzne – AWT
- ▶ 11: Interfejsy graficzne – Swing
- ▶ 12: Programowanie We/Wy
- ▶ 13: Programowanie sieciowe
- ▶ 14: JAR & Refleksja
- ▶ 15: Podsumowanie

Cel na dziś

Wprowadzić
programowanie
sieciowe w Javie

Protokoły



Definicja standardu

Model lub przykład
ustanowiony przez instytucję,
zwyczaj, lub powszechne
stosowanie

Przykłady standardów

- ▶ Jakiś pomysł?
- ▶ Postscript
- ▶ PDF
- ▶ HTTP
- ▶ HTML
- ▶ JPEG
- ▶ MPEG
- ▶ ...

Dwa rodzaje standardów

- ▶ De facto
 - ▶ Powszechne stosowanie
- ▶ Ustanowione przez instytucje standaryzujące
 - ▶ IETF (*Internet Engineering Task Force*)
 - ▶ W3 Consortium
 - ▶ ITU (*International Telecommunication Union*)
 - ▶ EBU (*European Broadcasting Union*)
 - ▶ ANSI (*American National Standards Institute*)

Dlaczego standardy są ważne?

- ▶ Dokładna specyfikacja
- ▶ Interoperacyjność
- ▶ Niezależność od sprzedawcy

Definicja protokołu

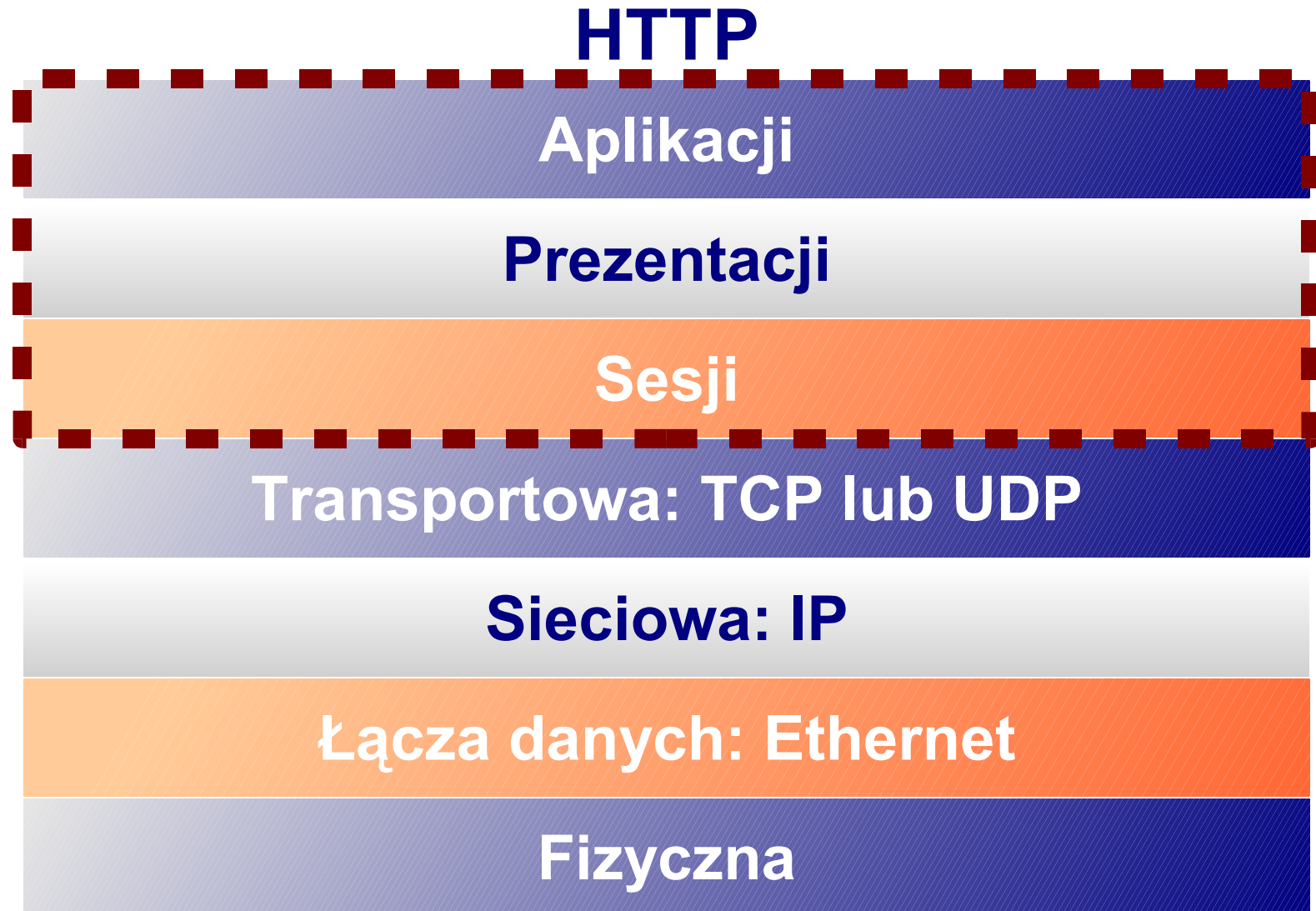
Język komputerowy pozwalający
komunikować się połączonym
ze sobą komputerom

[Cambridge Dictionary]

RFC

- ▶ IETF
(*Internet Engineering Task Force*)
- ▶ RFC
(*Request for Comments*)
- ▶ Standardy Internetu
 - ▶ IP RFC 791
 - ▶ TCP RFC 793
 - ▶ UDP RFC 768
 - ▶ HTTP RFC 2068

Model sieciowy ISO/OSI



`java.net.URL`



Definicja URL

URL jest skrótem od
Uniform Resource Locator
i jest odwołaniem (adres)
do zasobu w Internecie

Struktura URL

- ▶ Dwie części
 - ▶ Identyfikator protokołu
 - ▶ Nazwa zasobu
- ▶ Przykład
 - ▶ `http://www.google.com/`
 - ▶ `http`
 - ▶ `//www.google.com/`

Identyfikator protokołu
Nazwa zasobu

Struktura URL HTTP

- ▶ `http://<nazwaKomputera>:<port><zasób><ref>`
- ▶ Wymagane
 - ▶ Nazwa komputera
 - ▶ Zasób (“/” skrót dla `/index.html`)
- ▶ Opcjonalne
 - ▶ Port (domyślnie 80)
 - ▶ Ref
- ▶ Przykład
 - ▶ `http://www.myServ.org:8080/long.html#toc`

Konstruktory `java.net.URL`

► Bezwzględne

```
new URL("http://www.google.com")  
new URL("http", "www.google.com", "/")  
new URL("http", "www.google.com", 80, "/")
```

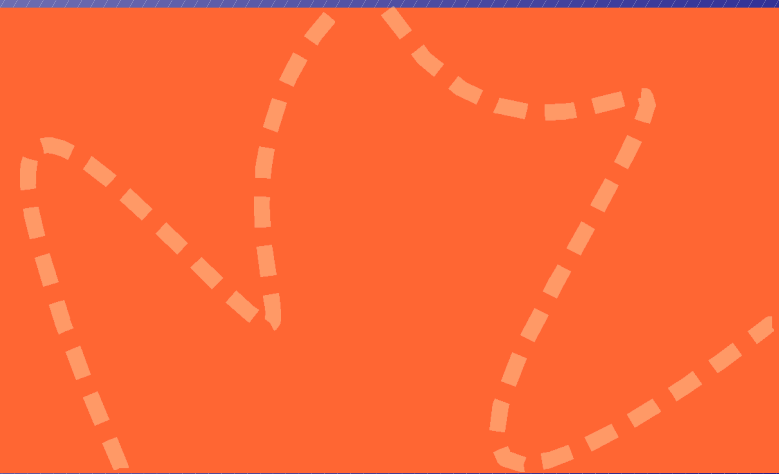
► Względne

```
new URL(googleURL, "/ads/overview.html")
```

Dostęp do `java.net.URL`

- ▶ Bezpośrednio
 - ▶ Metoda `URL.openStream()`
- ▶ Zaawansowane
 - ▶ Metoda `URL.openConnection()`
 - ▶ Klasa `java.net.URLConnection`

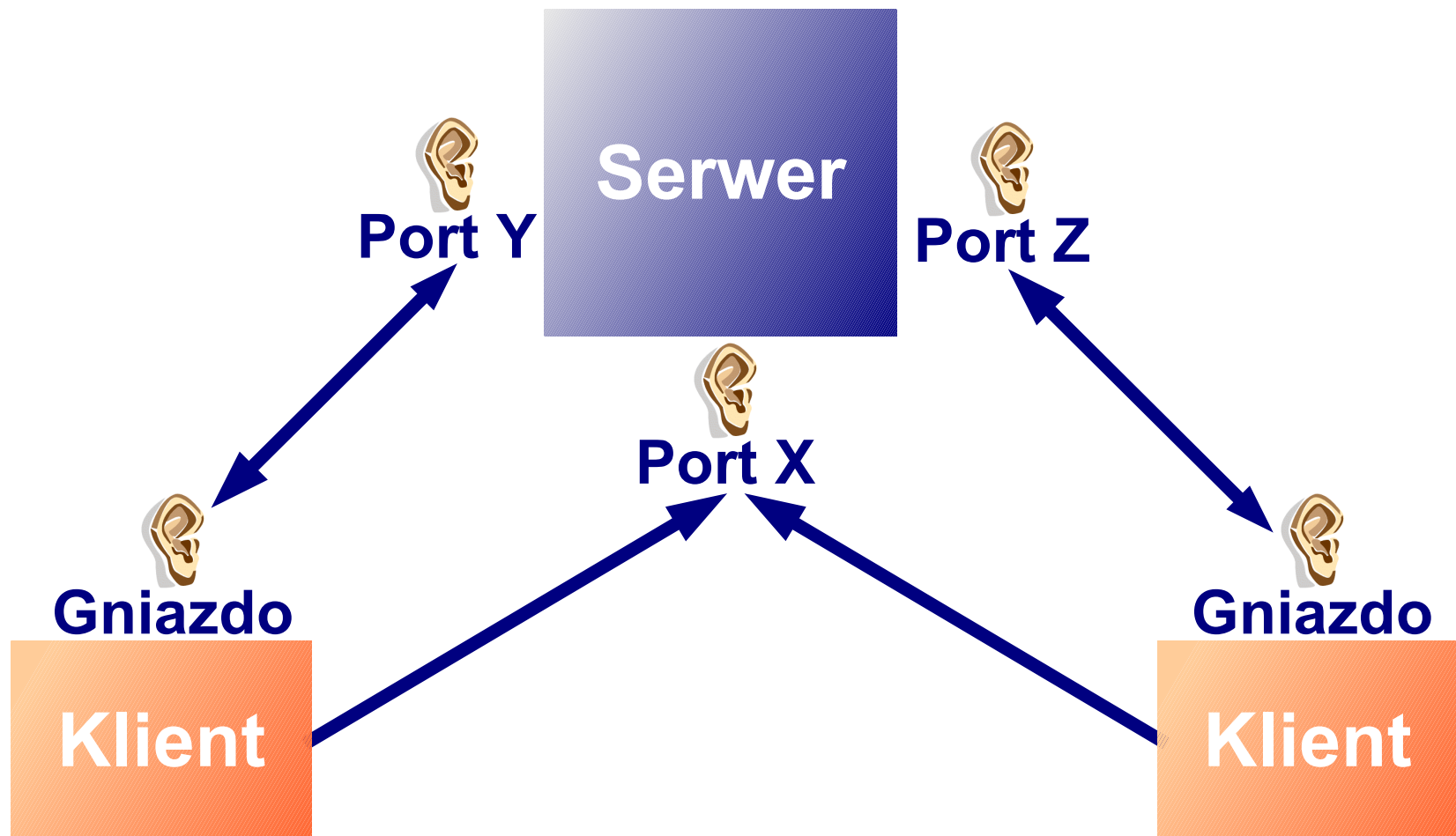
Gniazda *Sockets*



Definicja gniazda

Gniazdo jest końcówką
dwukierunkowego
połączenia między
dwoma programami
uruchomionymi
w sieci

Połączenia za pomocą gniazd



Programowanie gniazd w Javie

- ▶ **Po stronie klienta**

- ▶ `java.net.Socket`
- ▶ `Socket.getInputStream()`
- ▶ `Socket.getOutputStream()`

- ▶ **Po stronie serwera**

- ▶ `java.net.ServerSocket`
- ▶ `Socket serverSocket.accept()`
- ▶ **Jeden wątek dla każdego klienta**

Przykłady



Przykład URL

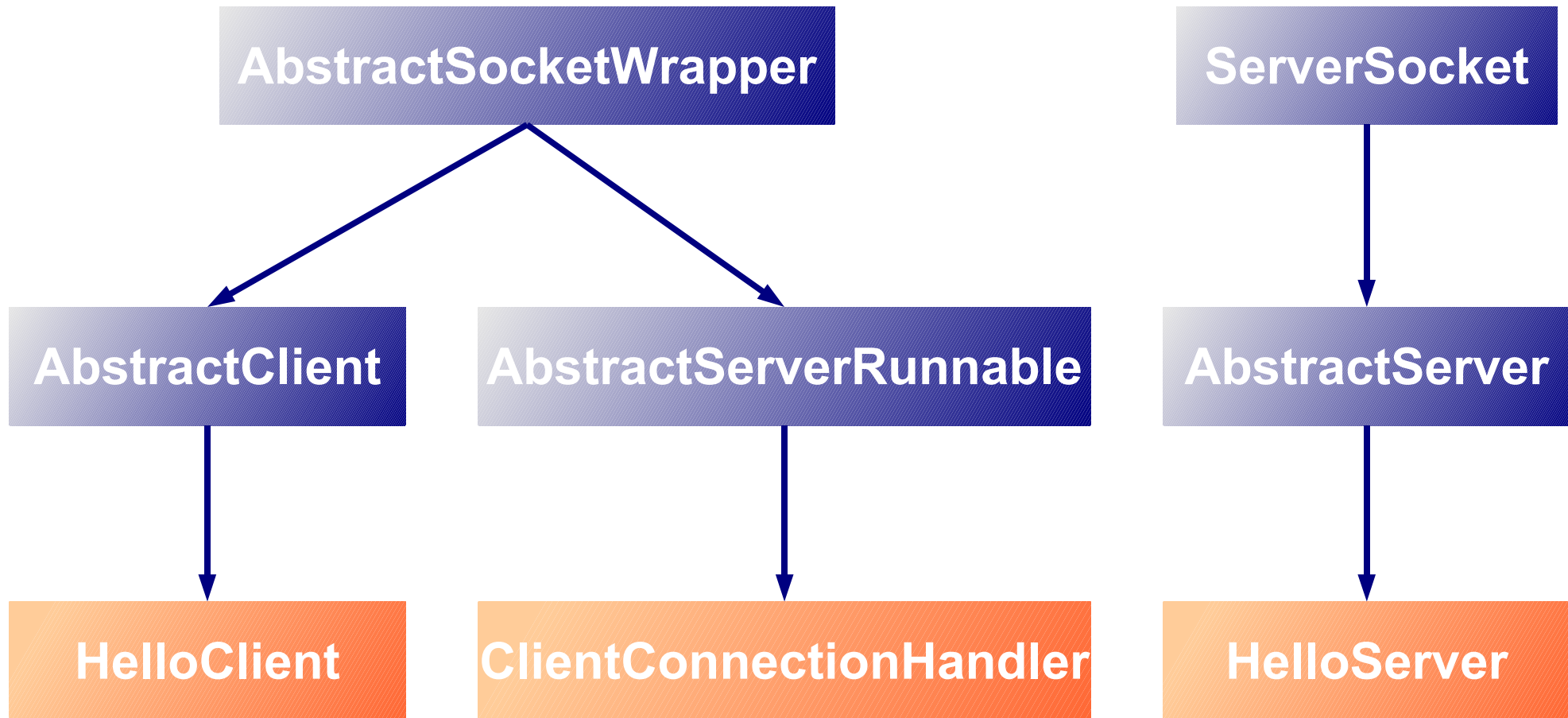
Przeglądarka RFC

klasa **RFCReader**

Przykład gniazda I

Przykład „Hello Server”

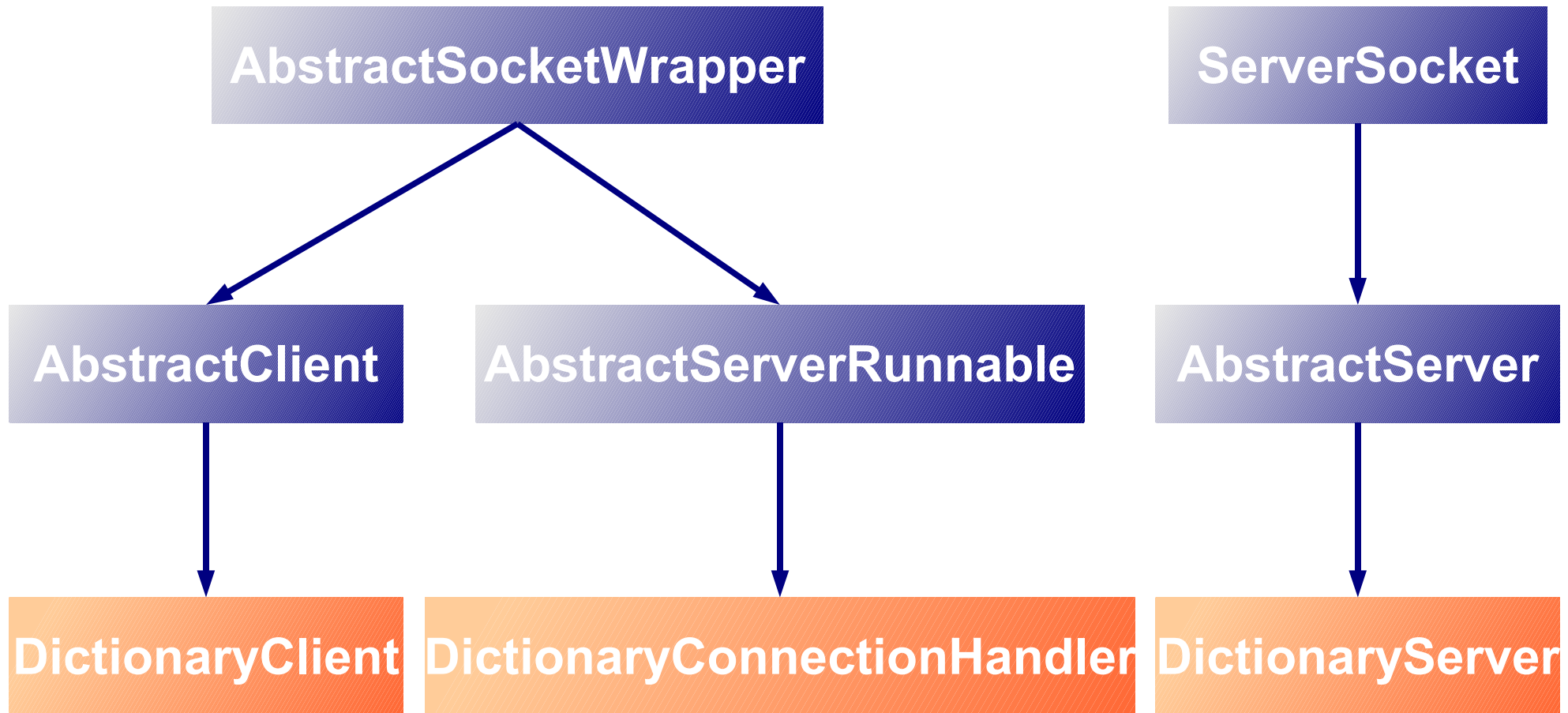
Hierarchia klas



Przykład gniazda II

Przykład „Dictionary Server”

Hierarchia klas



Podsumowanie



Programowanie sieciowe

- ▶ Różne poziomy abstrakcji
 - ▶ Od gniazd
 - ▶ Do URL
- ▶ Inne technologie sieciowe
 - ▶ CORBA
 - ▶ RMI
 - ▶ Servlets
 - ▶ itd.



Do zobaczenia za tydzień