Konzepte objektorientierter Programmierung

Prof. Dr. Peter Müller

Werner Dietl

Software Component Technology

Exercises 11: Concurrency and Distribution

Wintersemester 06/07

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

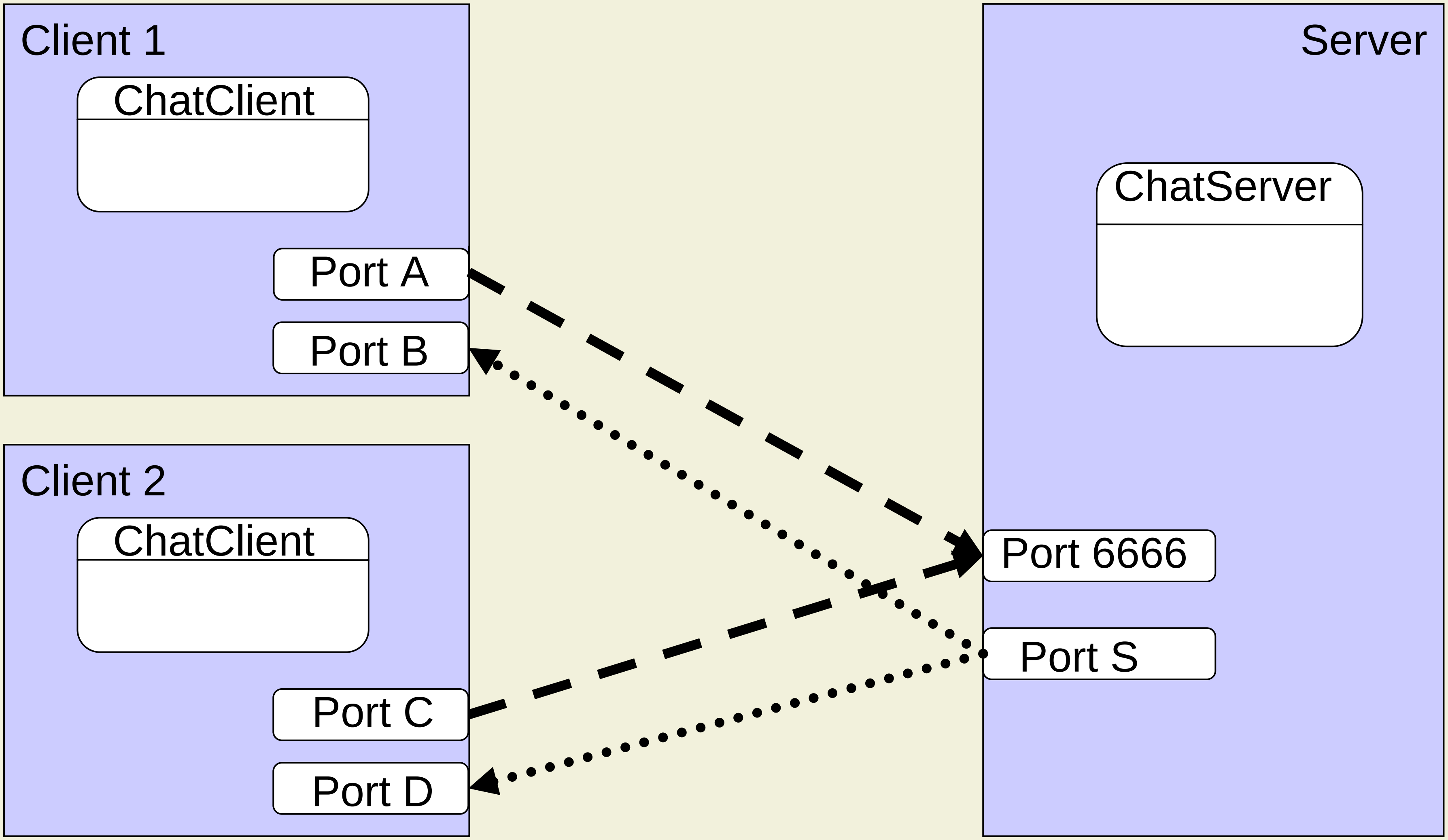
Concurrency

- New package since Java 1.5: `java.util.concurrent`

- Adds: Executors, Locks, ReadWriteLocks, Atomicity, ThreadPools, Synchronizers, …
- Easier to use and less error prone

- Sun Tech Days: “Java SE – Beyond the Basics: Java Concurrency, Type Checking Verifier, …” by Sridhar Reddy and Wei Tao

Homework 10 – Chat Application



Messages

- Register: Client → Server
  Client host and port
- Bcast: Client → Server
  Message
- Msg: Server → Client
  Message
- Deregister: Client → Server
  Client host and port

Which Threads do we need?

- Client:
  - Keyboard handler
  - Connections from server

- Server:
  - Multiple connections from clients

Ex. 2. a. Entering the shower

```
class Dusche {
  static int frauen=0, maenner=0;

  synchronized static void rein(
                          Person p) {
    while (maenner+frauen == 10)
      Dusche.class.wait();

    if (p instanceof Mann)
      maenner = maenner+1;
    else frauen=frauen+1;
  }
```

Ex. 2. a. Leaving the shower

```
synchronized static void raus(Person p)
{
  if (p instanceof Mann)
    maenner=maenner-1;
  else
    frauen=frauen-1;

  Dusche.class.notifyAll();
}
```

Ex. 2. b. Entering the shower

```
synchronized static void rein(Person p)
{
  while ( maenner+frauen == 10 ||
     (p instanceof Mann && frauen>0) ||
     (p instanceof Frau && maenner>0)
        )
    Dusche.class.wait();

  if (p instanceof Mann)
    maenner = maenner+1;
  else frauen=frauen+1;
}
```

Questions?