

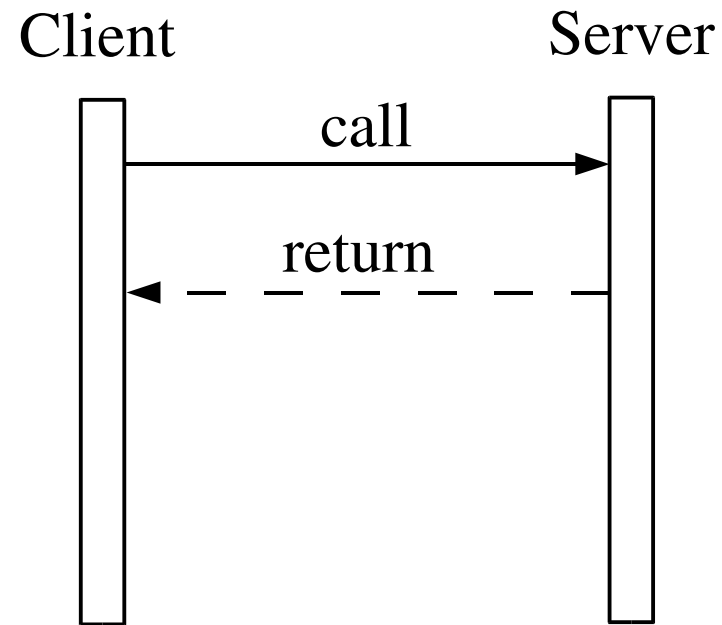
Einführung in RMI

Axel Rauschmayer,
2003-11-14

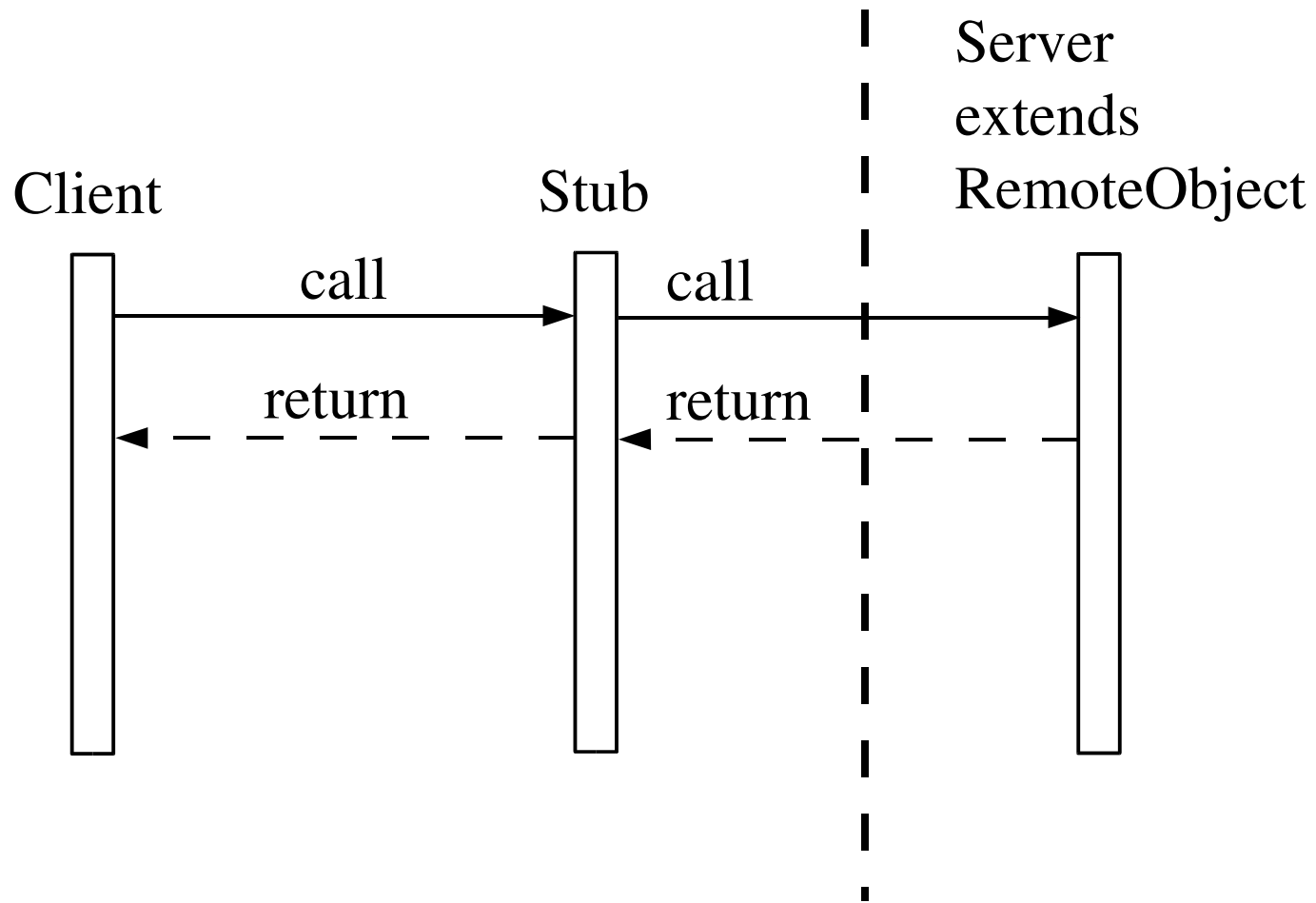
Problem

- Wie verteile ich Software möglichst transparent?
- Lösung: Virtualisierung – ich tue so, als existiere die Trennung nicht

Client-Server-Kommunikation



Implementierung per RMI



Schwierigkeiten

- Wie finde ich das Server-Objekt?
 - Anbieten und abholen bei zentraler Registry
- Wie transportiere ich Argumente über den Draht?
 - ggf. automatischer Serialisierung
- Sicherheit bei Argument-Objekten?
 - SecurityManager

Was muß der Client vom Server wissen?

- Interface (separat von der Implementierung)
 - extends `java.rmi.Remote`
 - Methoden: throws `RemoteException`
- Parameterklassen
 - müssen `Serializable` sein
- Stub
 - Speziell für die Implementierung angefertigt

Implementierung des Servers

- extends `UniCastRemoteObject`
- implements `ServerInterface`
- Konstruktor: Ohne Argumente, ruft `super()` auf
- Implementierung der Interface-Methoden
- optional:
 - `SecurityManager` konfigurieren
 - Instanz bei Registry binden

Implementierung des Clients

- SecurityManager konfigurieren
- Hole Referenz auf Server-Objekt
 - (bzw. auf dessen lokalen Vertreter, den Stub)
- Verwende die Referenz wie üblich für Methodenaufrufe

Security Policy

```
% cp $JRE_HOME/lib/security/java.policy \  
myserver.policy
```

```
% java \  
-Djava.security.policy=myserver.policy
```

```
----->8
```

```
grant {  
    permission java.net.SocketPermission \  
        "localhost:1024-", \  
        "accept,connect,listen,resolve";  
};
```

Kompilieren

```
% cd server
% javac *.java
% rmic-1.4.1_01 -v1.2 RemotePrinterImpl
% cp Message.class RemotePrinter.class \
  RemotePrinterImpl_Stub.class ../client
% cd ../client/
% javac Client.java
```

Ausführung

Server

```
% rmiregistry-1.4.1_01 1099 &  
% java \  
-Djava.security.policy=../my.policy \  
RemotePrinterImpl
```

Client

```
% java \  
-Djava.security.policy=../my.policy \  
Client "Howdy!"
```

Dynamisches Laden von Code

- Stubs und Hilfs-Klassen können dynamisch geladen werden
- Genaueres unter:
 - <http://sirius.ps.uci.edu/~smichael/rmi.htm>
 - <http://www.javaworld.com/javaworld/jw-12-1998/jw-12-enterprise.html>

Weitere Informationen

- <http://java.sun.com/docs/books/tutorial/rmi/>
- <http://java.sun.com/j2se/1.4.2/docs/guide/rmi/>
 - (besonders die FAQ)
- <http://java.sun.com/j2se/1.4.2/docs/guide/security/PolicyFiles.html>