



# Remote Method Invocation

- Einleitung
  - Einordnung von RMI
  - Ziele von RMI
- Grundlagen
  - RPC
- Funktionsweise von RMI
  - Kommunikations-Architektur
  - Elemente und Ablauf
- Implementierung

# RMI - Einleitung



## Einordnung von RMI

- neben der reinen Datenübertragung im Netzwerk, ist der Zugriff auf Dienste, die sich auf entfernten Rechnern befinden wichtig
- RMI Werkzeug/Mechanismus zur verteilten Anwendungsprogrammierung
- d.h. Zugriff auf Objekte, die im Kontext einer anderen JVM instanziiert sind
- diese JVM kann (muß aber nicht) auf einem entfernten Rechner laufen
- Plattformunabhängigkeit von Java

# RMI - Einleitung



## Ziele von RMI

- Zugriff auf die Dienste eines entfernten Rechners erfordert ein aufwendiges und fehleranfälliges Kommunikationsprotokoll (Datenerstellung, Kommunikationssteuerung)
- RMI implementiert dieses Kommunikationsprotokoll und entlastet den Anwendungsprogrammierer
- Skalierbarkeit von Anwendungen erhöhen
- Lastausgleich ermöglichen
- *z.B. Middleware*

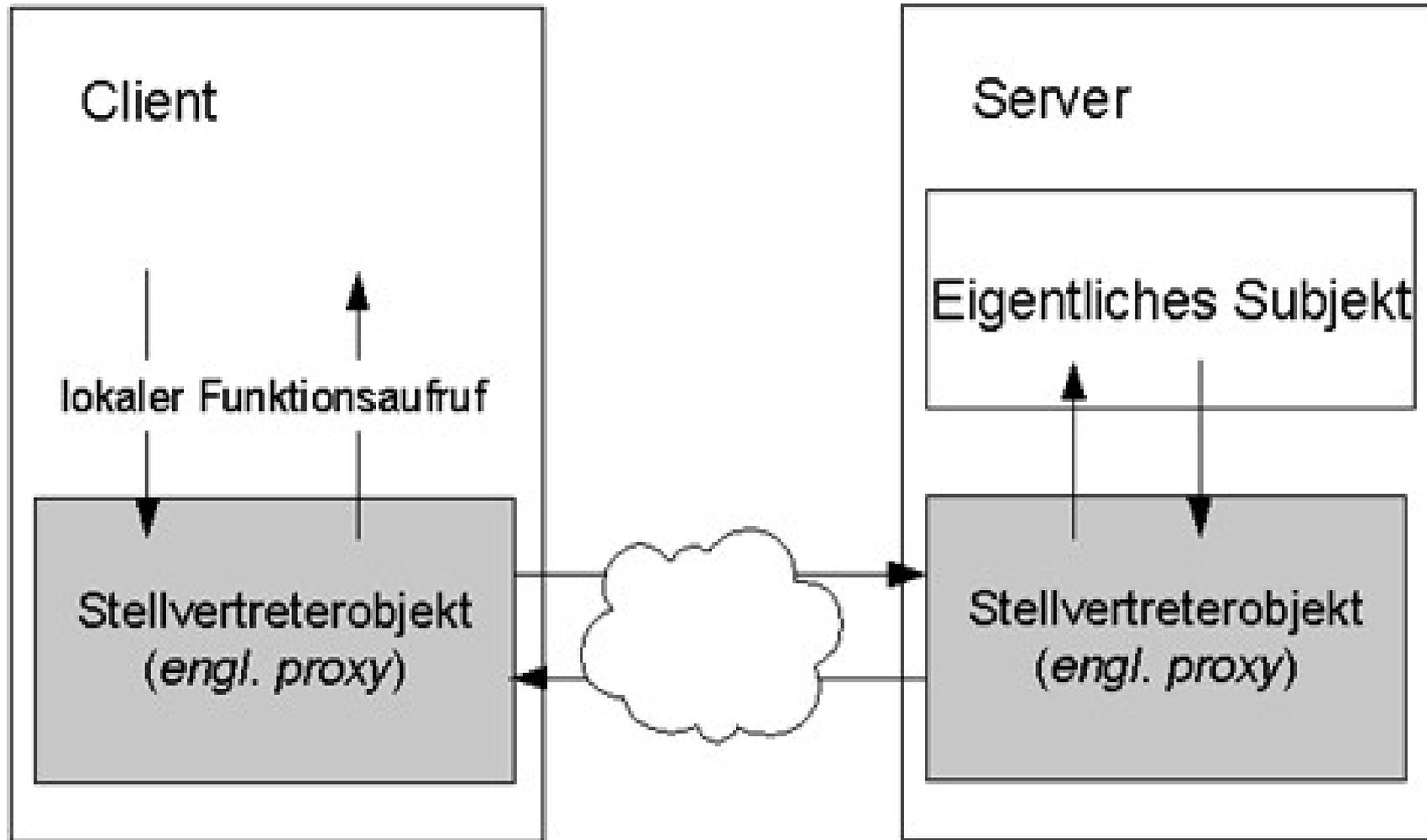
# RMI - Grundlagen



## **RPC – *Remote Procedure Call***

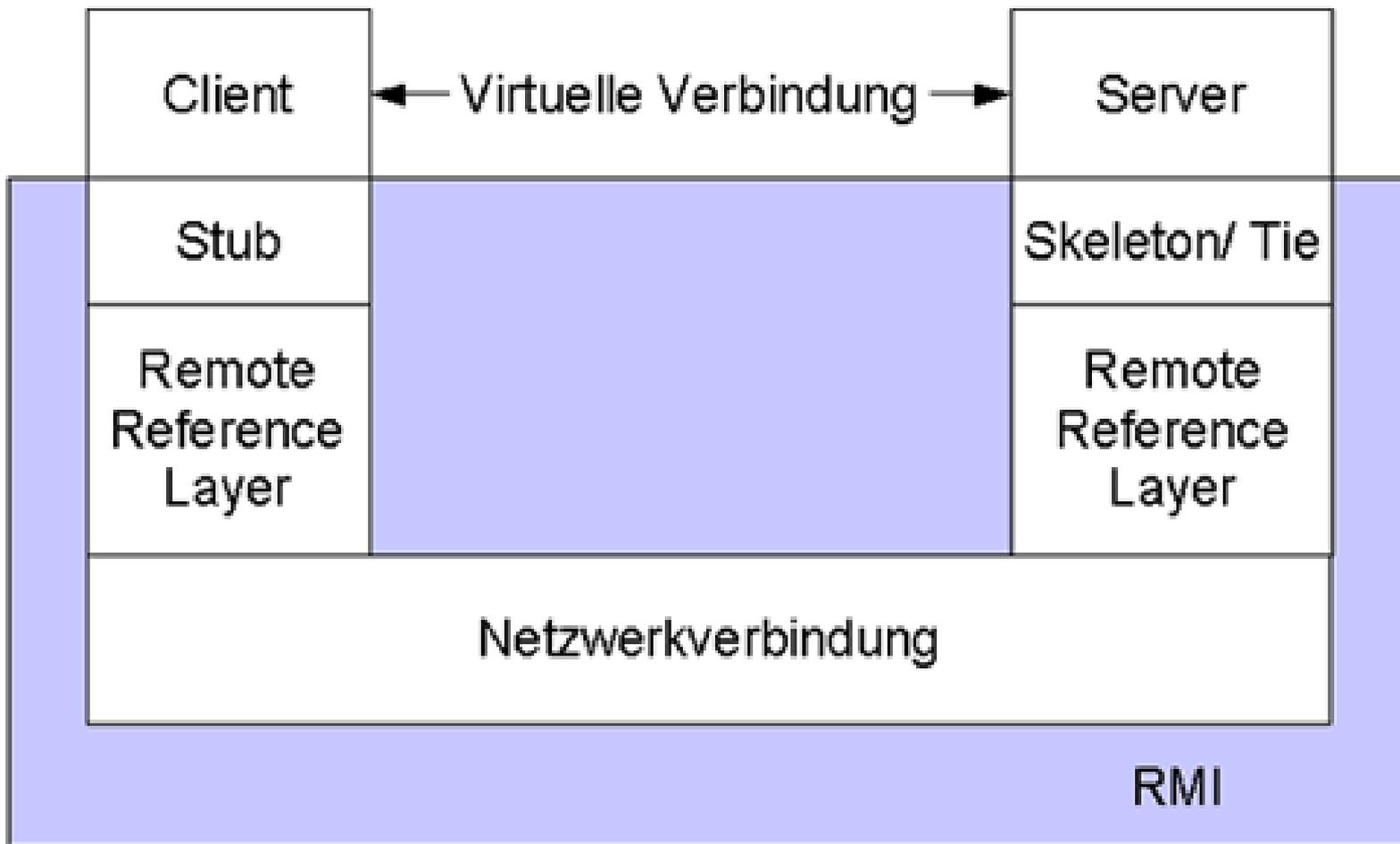
- erstmals 1983 von Birrel und Nielson [Birrel, 1983] vorgestellt
- **RPC Prinzip:** Zugriff auf entfernte Serverfunktion sieht wie ein lokaler Prozeduraufruf aus
- verbirgt so die Komplexität der Netzwerkprogrammierung vor dem Anwendungsentwickler
- eigentliche Verbindung läuft über Stellvertreterobjekte (*engl. proxies*)
- synchrones Kommunikationsschema

# RMI – Grundlagen: RPC



# RMI – Funktionsweise

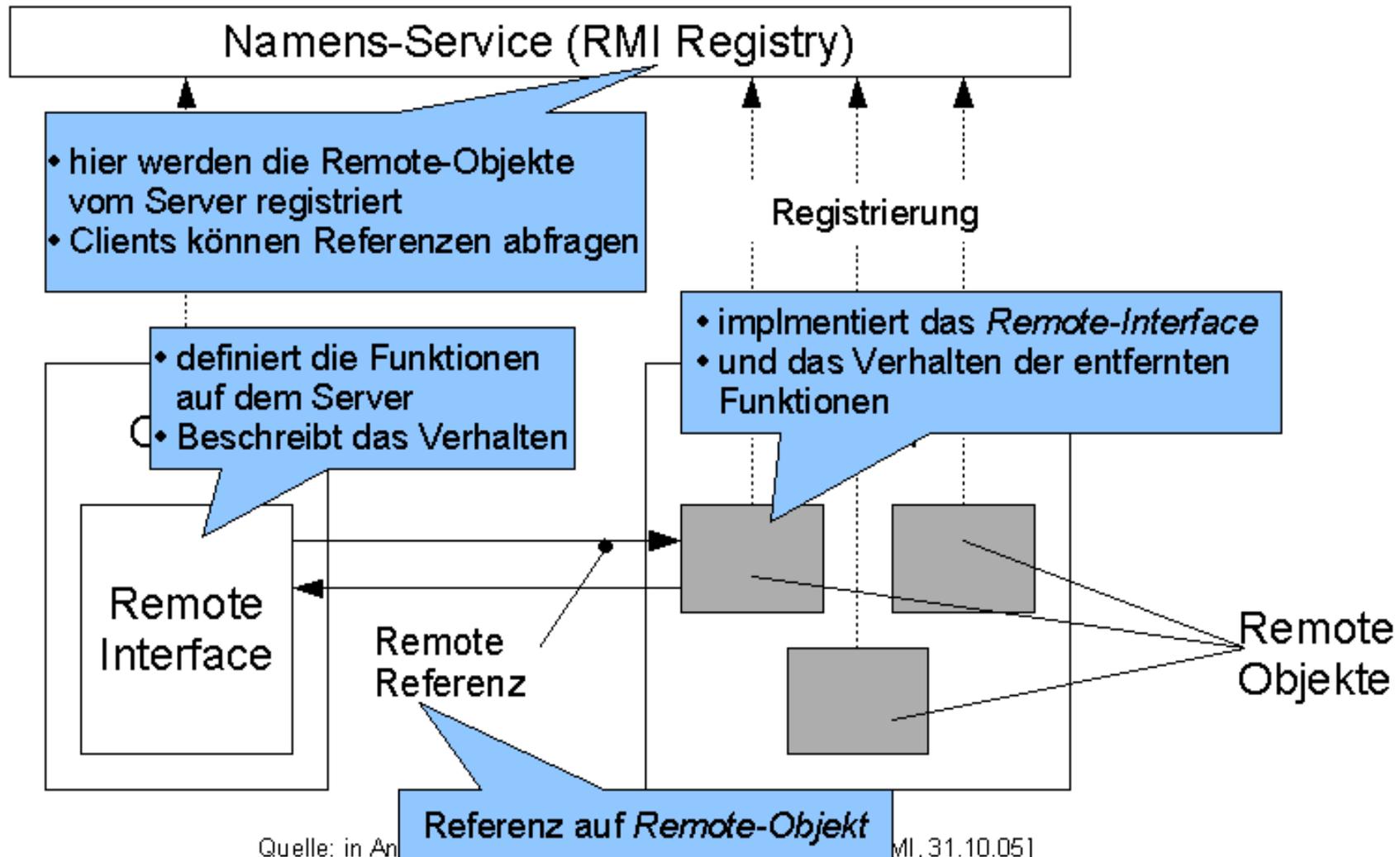
## Kommunikations-Architektur



Quelle: Anlehnung an [Krüger, 2004]

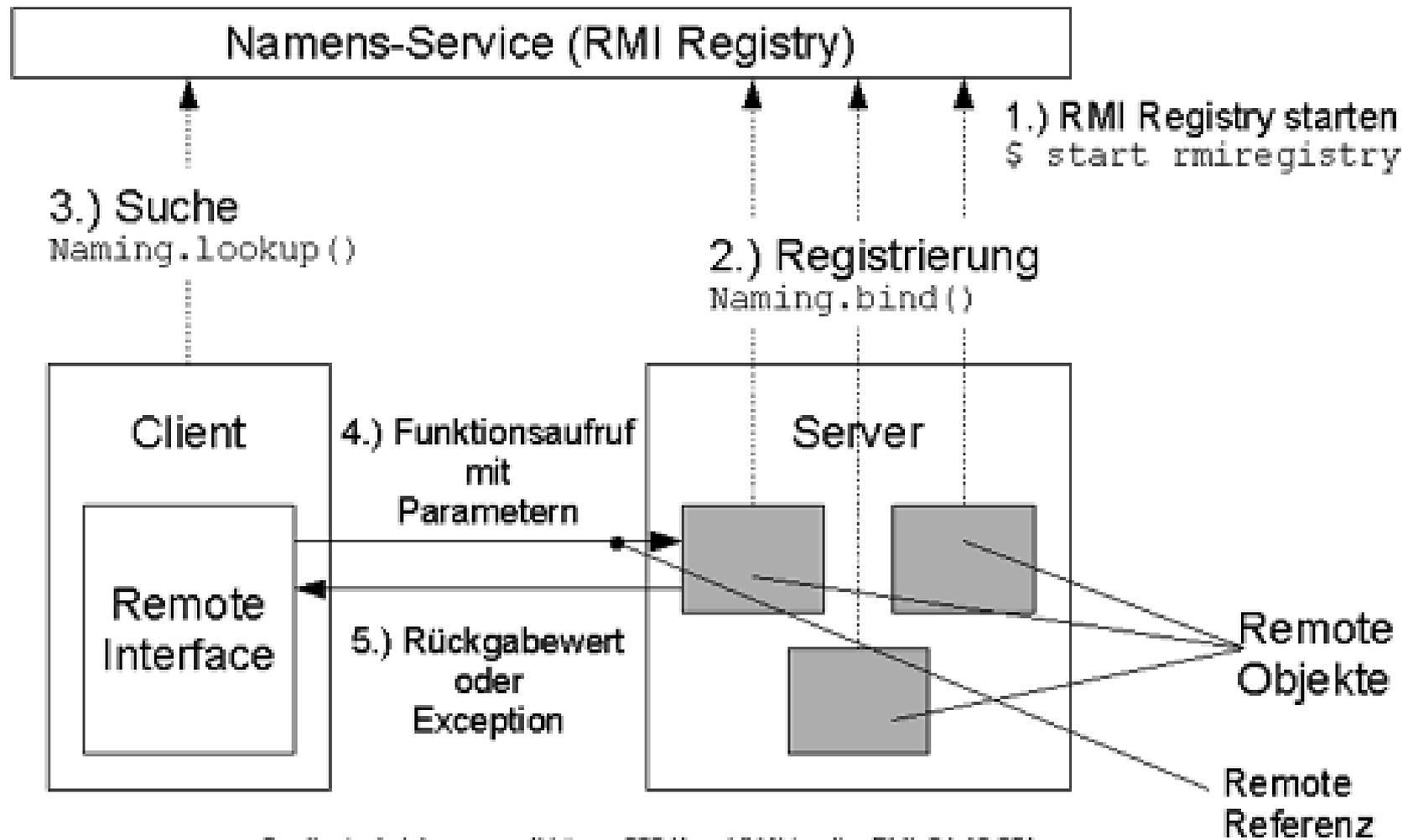
# RMI – Funktionsweise

## Elemente



# RMI – Funktionsweise

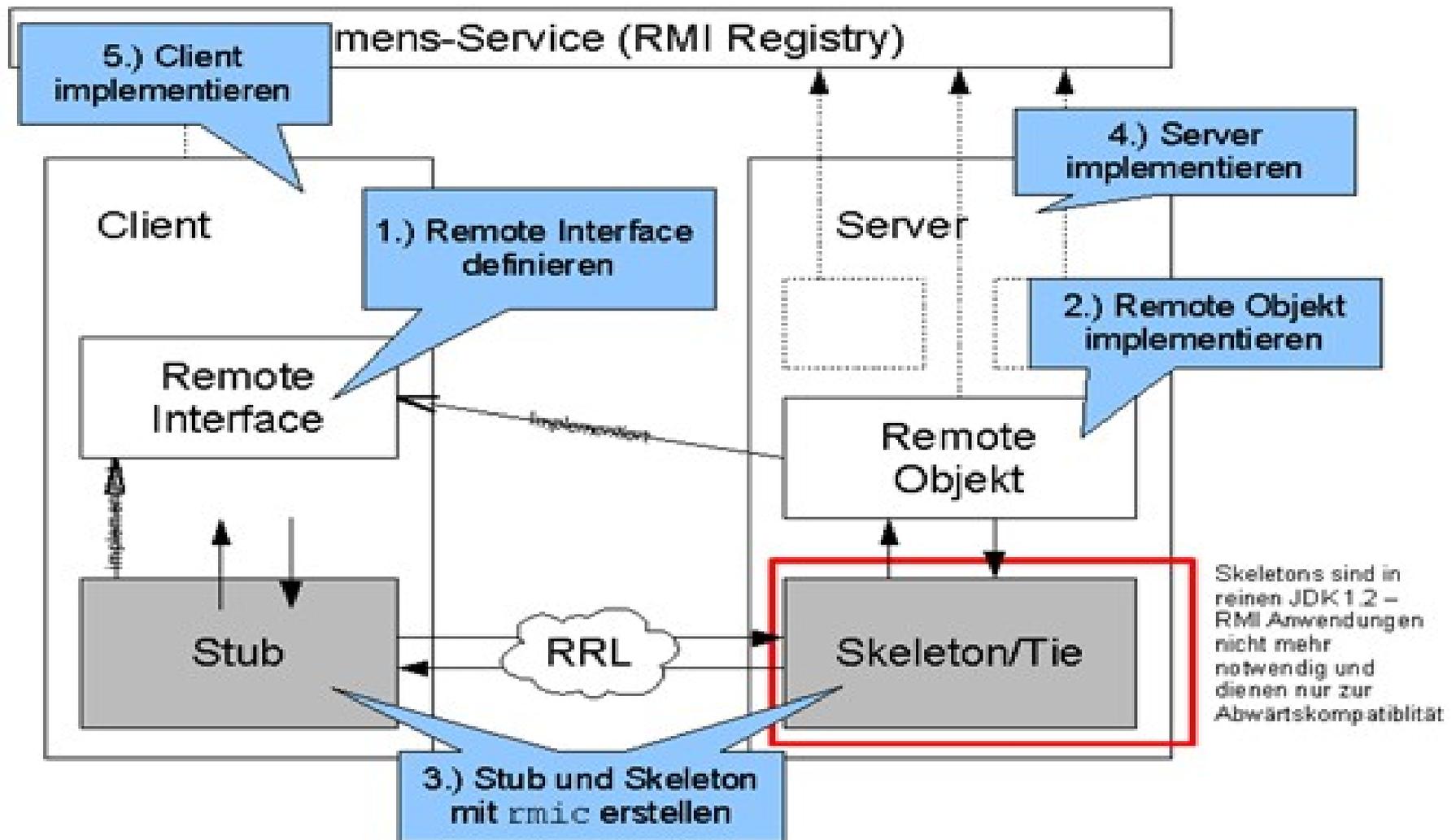
## Ablauf



Quelle: in Anlehnung an [Krüger, 2004] und [Wikipedia, RMI, 31.10.05]

# RMI – Implementierung

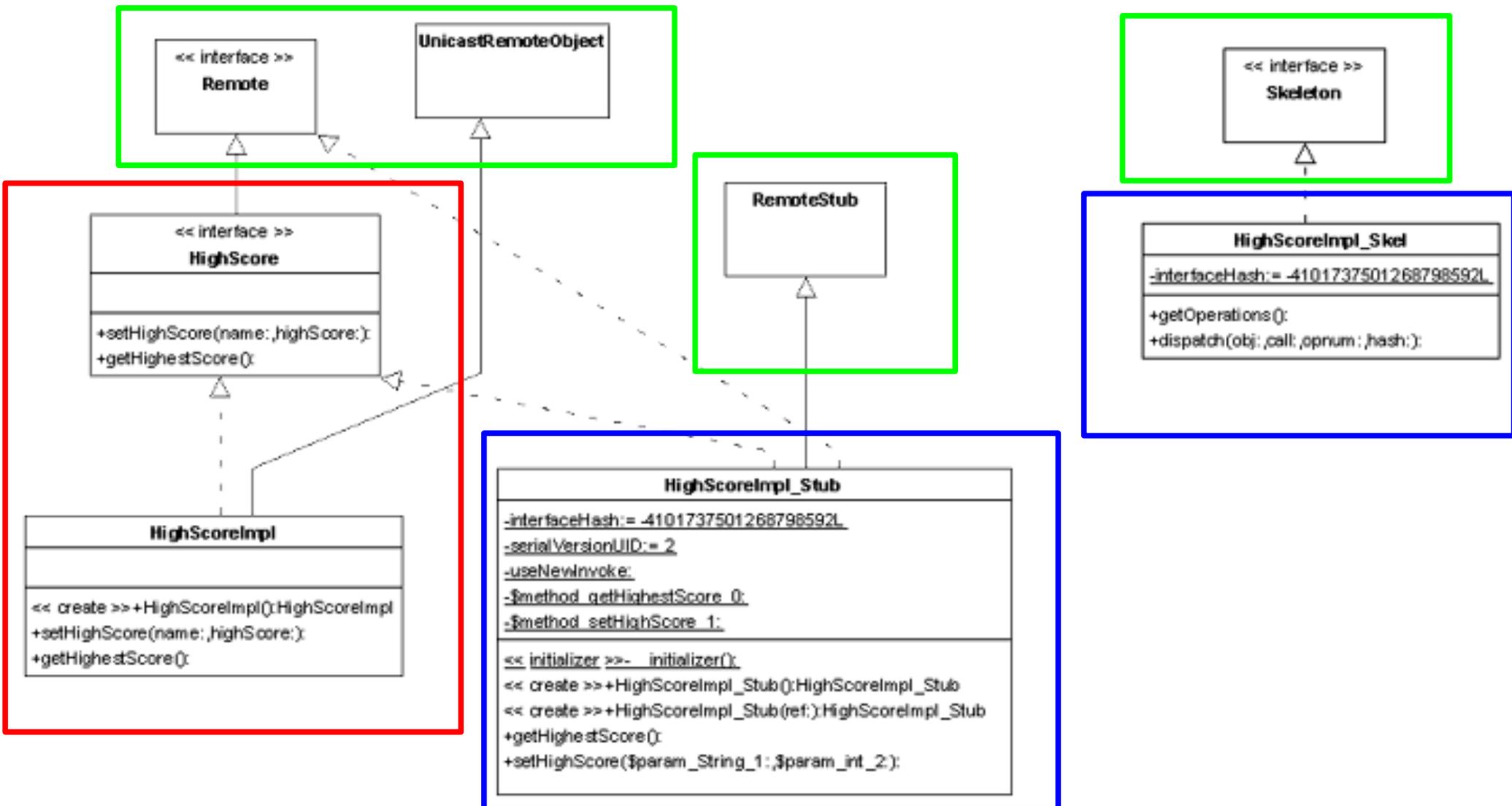
## Vorgehen



Quelle: Eigene Darstellung in Anlehnung an [Krüger, 2004]

# RMI – Implementierung

## Zusammenfassung



# RMI – Ausblick



- Granularität der entfernten Methoden
  - Aufruf einer entfernten Methode kostet wesentlich mehr Zeit als der Aufruf einer lokalen Methode
  - Laufzeitprobleme
- Nachladen von Programmcode
- Distributed Garbage Collection
- Die verschiedenen RMI Protokolle
  - Sichere Übertragung mit SSL
  - Umgehen von Firewalls mit HTTP

# RMI – Literatur



- [Birrel, 1983] Birrell, A. D. & Nelson, B. J., "Implementing Remote Procedure Calls", XEROX CSL-83-7, October 1983
- [Darwin, 2002] Java Kochbuch, Ian F. Darwin, 1. Aufl. 2002, O'Reilly
- [Gamma, 2004] Erich Gamma, Richard Helm, Ralph E. Johnson u. a., Entwurfsmuster, 2004, ADDISON-WESLEY, MÜNCHEN
- [Krüger, 2004] Krüger, Handbuch der Java Programmierung, 4. Aufl., Addison-Wesley
- [Middendorf, 2002] Java - Programmierhandbuch und Referenz, 3. Auflage 2002  
Stefan Middendorf, Reiner Singer, Jörn Heid
- [RFC 1057, 1988] RPC: Remote Procedure Call – Protocol Specification  
Version 2, June 1988, <http://www.ietf.org/rfc/rfc1057.txt>
- [Sun1, 2005] Sun Microsystems, JDK 5.0 Documentation,  
<http://java.sun.com/j2se/1.5.0/download.jsp#docs>
- [Sun2, 2005] Sun Microsystems, White Paper 'Java Remote Method Invocation - Distributed Computing for Java'
- [Sun3, 2000] Sun Microsystems, jGuru: Remote Method Invocation,  
<http://java.sun.com/developer/onlineTraining/rmi/index.html>
- [Ullenboom, 2004] Java ist auch eine Insel, Galileo Computing, 4. Aufl.
- [Wikipedia, RMI, 30.10.2005] Wikipedia, Artikel RMI, vom 30.10.2005,  
<http://de.wikipedia.org/wiki/RMI>