

Realisierung einer Client/Server-Anwendung mit CORBA und Java unter Berücksichtigung bestehender C++-Komponenten

**Klaus Beschorner
Wolfgang Rosenstiel**



**Universität Tübingen
Technische Informatik**



**iT media Consult GmbH
Stuttgart**

- **Motivation**
- **Problemstellung**
- **Aufgaben, Probleme und Lösungen**
- **Anwendung und ihr Leistungsverhalten**
- **Bewertung von CORBA/Java**
- **Zusammenfassung**

- **Heterogene und kostenintensive Systemumgebungen**
- **Altanwendungen**
- **Neue Anforderungen: World Wide Web, Java, ECommerce**

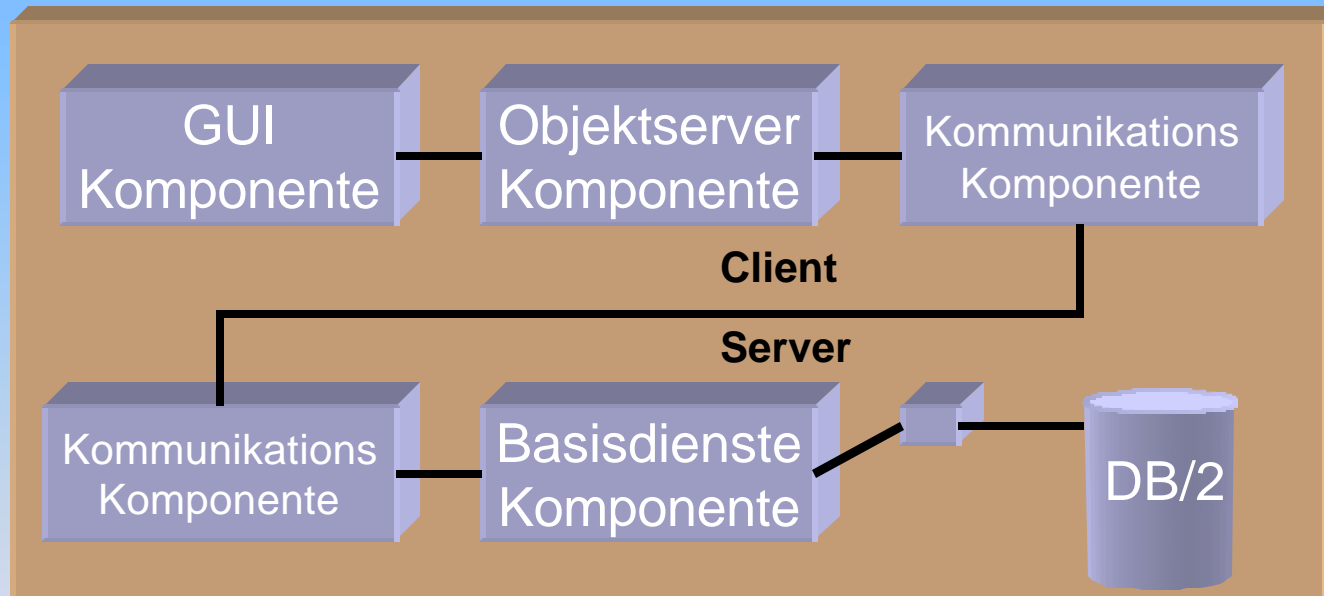
Integration?

Browser-Anwendungen im Internet/Intranet?

**Mögliche Lösung: Common Object Request Broker
Architecture (CORBA)**

Versicherungsbranche: Agentur-Informationssystem

- Realisiert mit „IBM Visual Age for C++“
- Kommunikation auf Socket-Basis

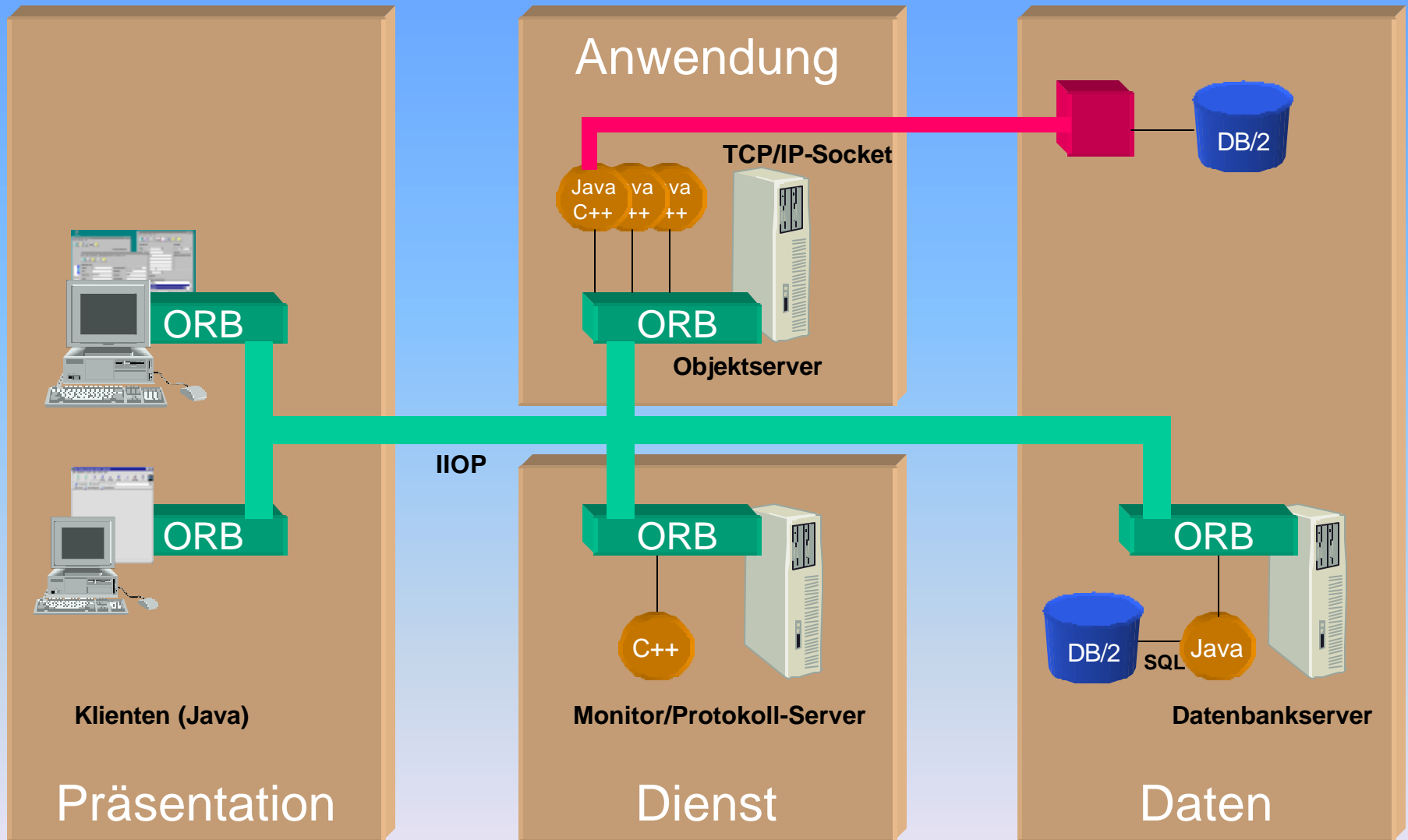


Ziel: Umsetzung mit CORBA, Wiederverwendung Objektserver

Resultierende Aufgaben

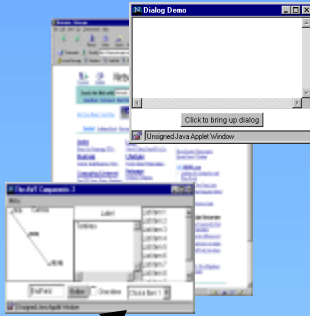
<i>Aufgabe</i>	<i>Probleme</i>
<i>ORB-Integration</i>	- Systemarchitektur
<i>C++-GUI in Java</i>	- mehrere Masken in Applets - Wiederverwendung in Applikationen
<i>C++-Code in CORBA-Objekte kapseln</i>	- Compiler-Inkompatibilität
<i>Java-Code in CORBA-Objekte kapseln</i>	- vorhandene Datenstrukturen

Systemarchitektur



Applet- und Applikationsproblematik

„Pop-out“-
Frames



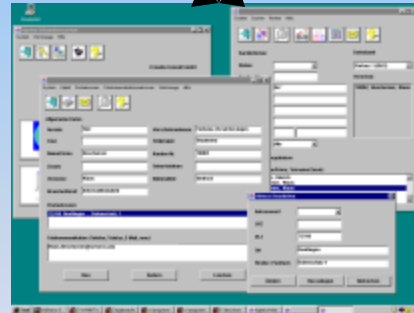
mehrere Browser-
Fenster



ein Browser-
Fenster

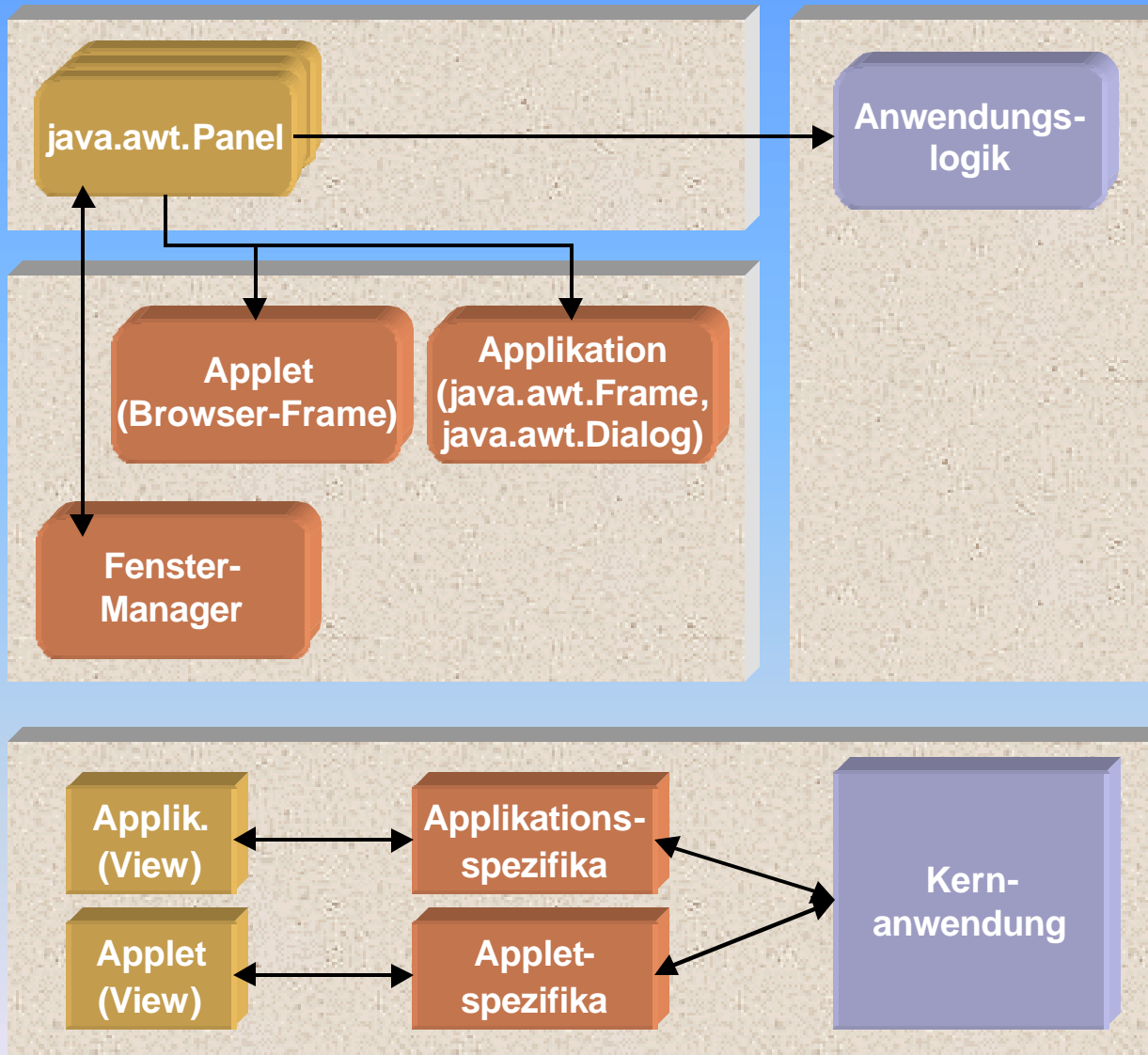


Wiederverwendung



Problem: Unterschiede zwischen Applet und Applikation

Applet & Applikationsprogrammierung



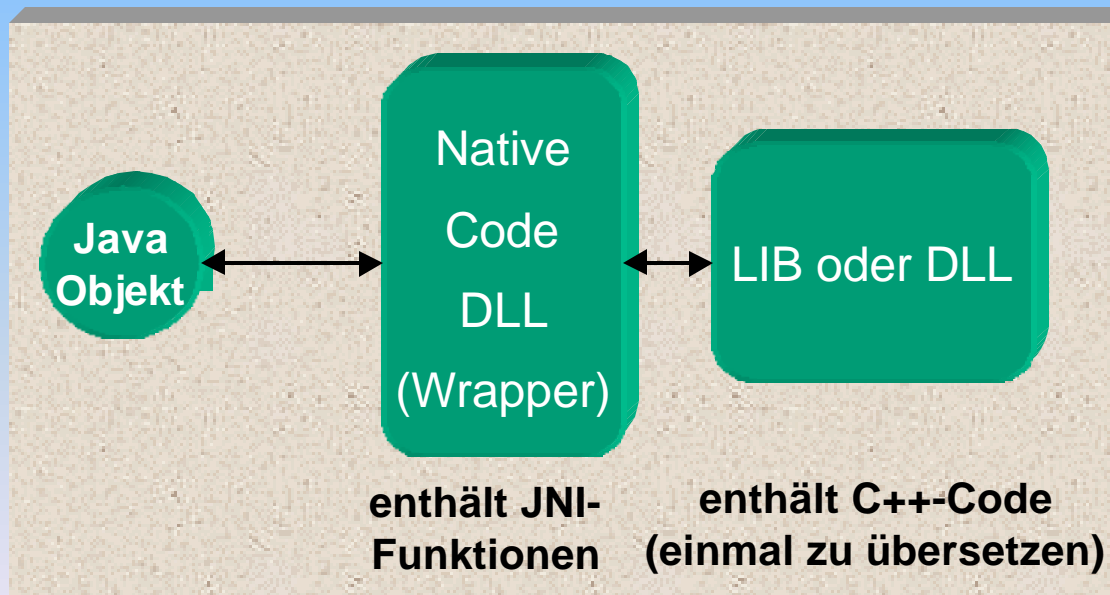
Verfahren

Design

Java Native Interface zur Einbindung von C++-Code (1)

- **Grund** : **Wiederverwendung**
- **Probleme** :
 - Abbildungen zwischen C++ und Java
 - Technische Umsetzung

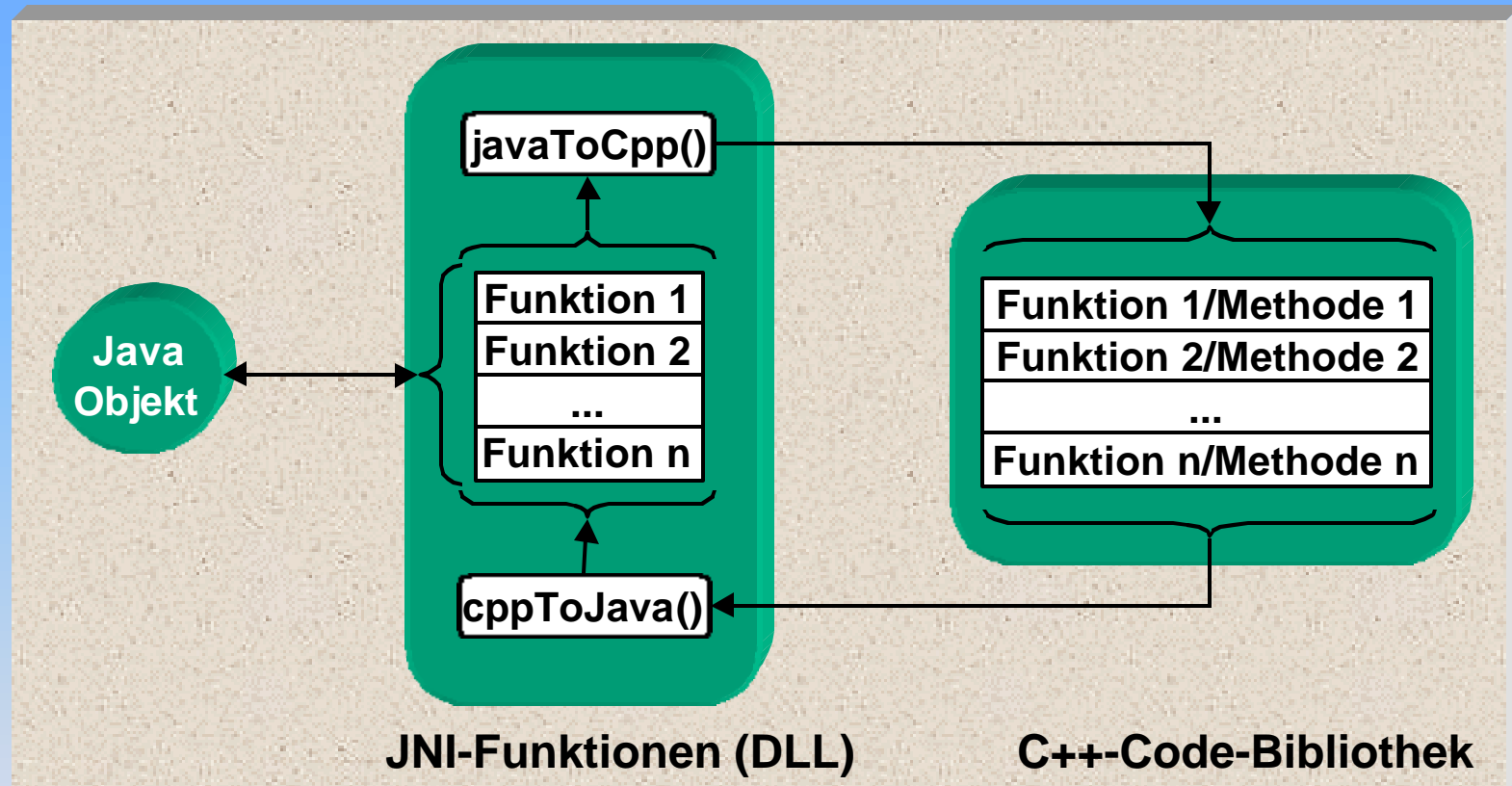
Technische Umsetzung: Wrapper



Vorteile:

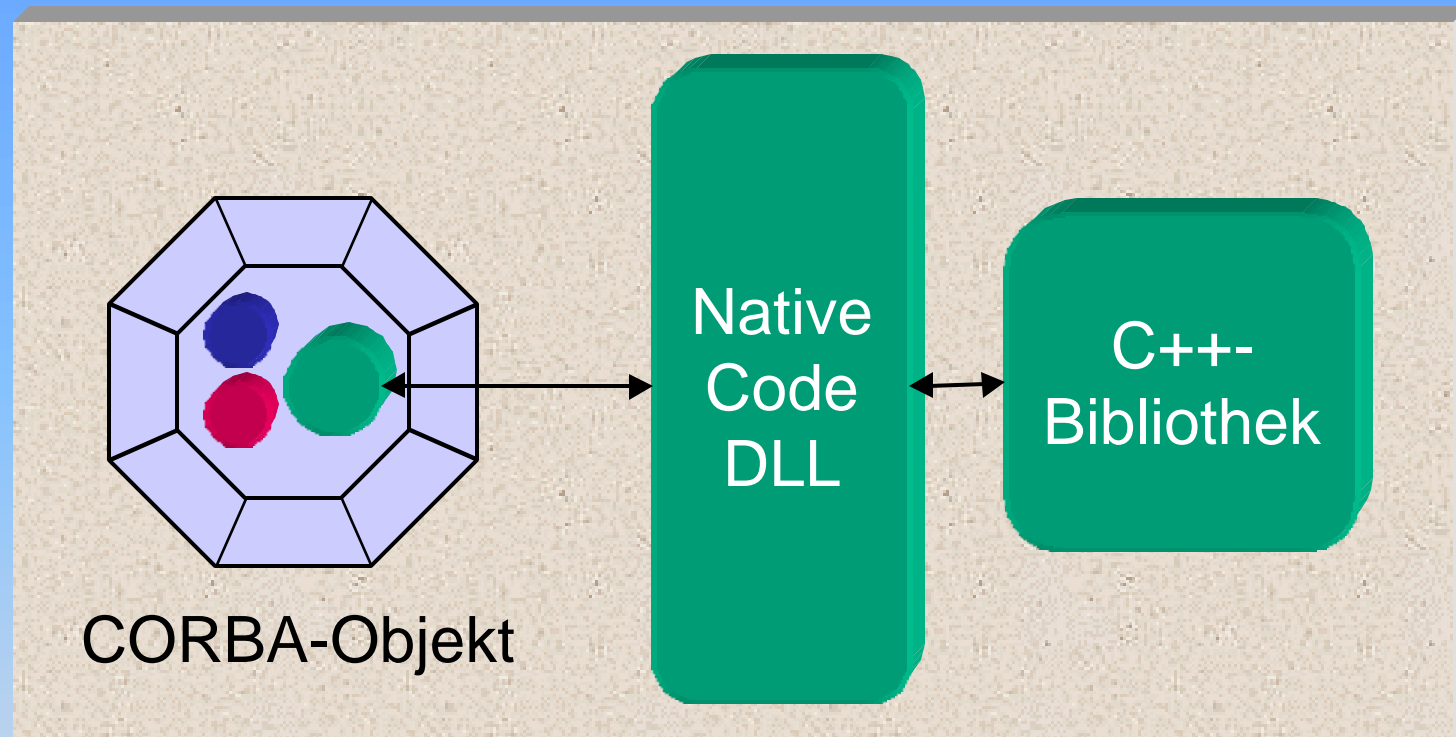
- C++-Code als Black-Box
- keine fehleranfälligen Änderungen
- kein Quellcode nötig

Beispiel für den Aufbau einer Wrapper-DLL



eine Datenstruktur **D** ein Paar Konvertierungsfunktionen

Kapselung von bestehendem Code in CORBA-Objekte (1)



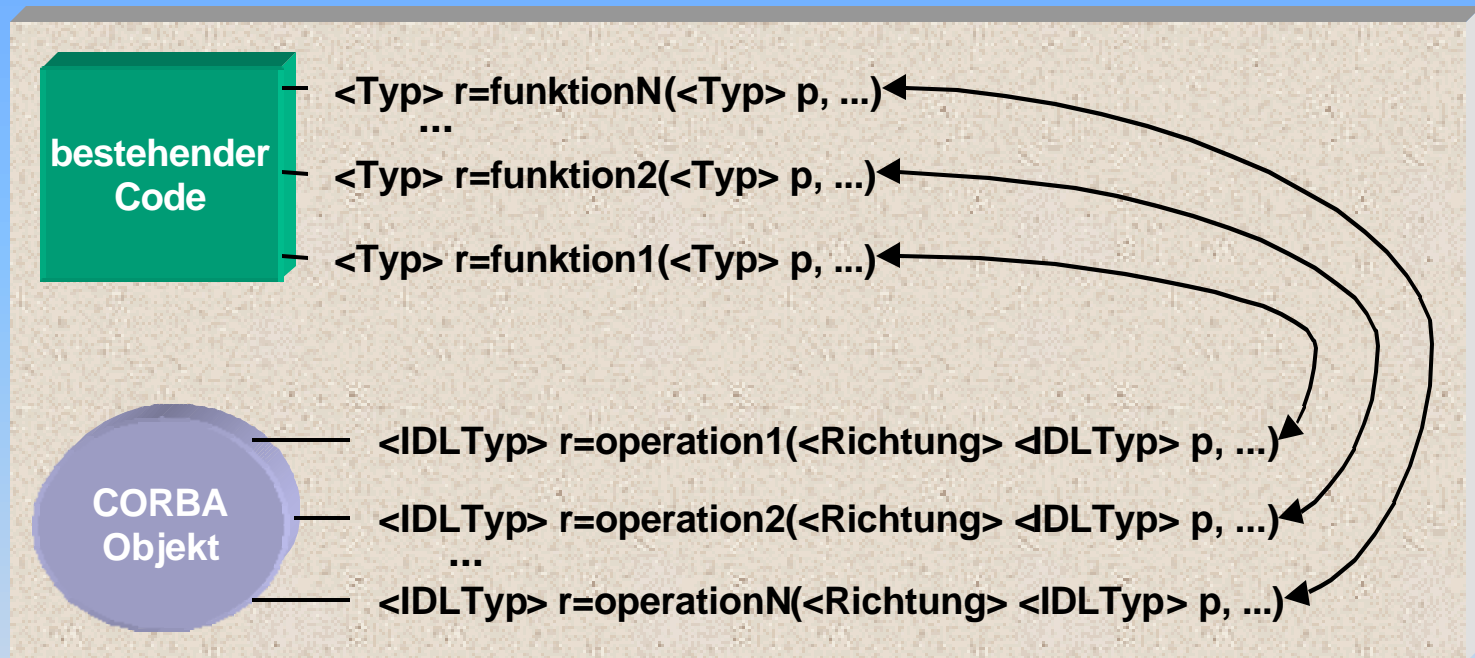
Allgemein:

Bestehenden Code mit einer IDL-Schnittstelle versehen.

Kapselung von bestehendem Code in CORBA-Objekte (2)

Aufgabe:

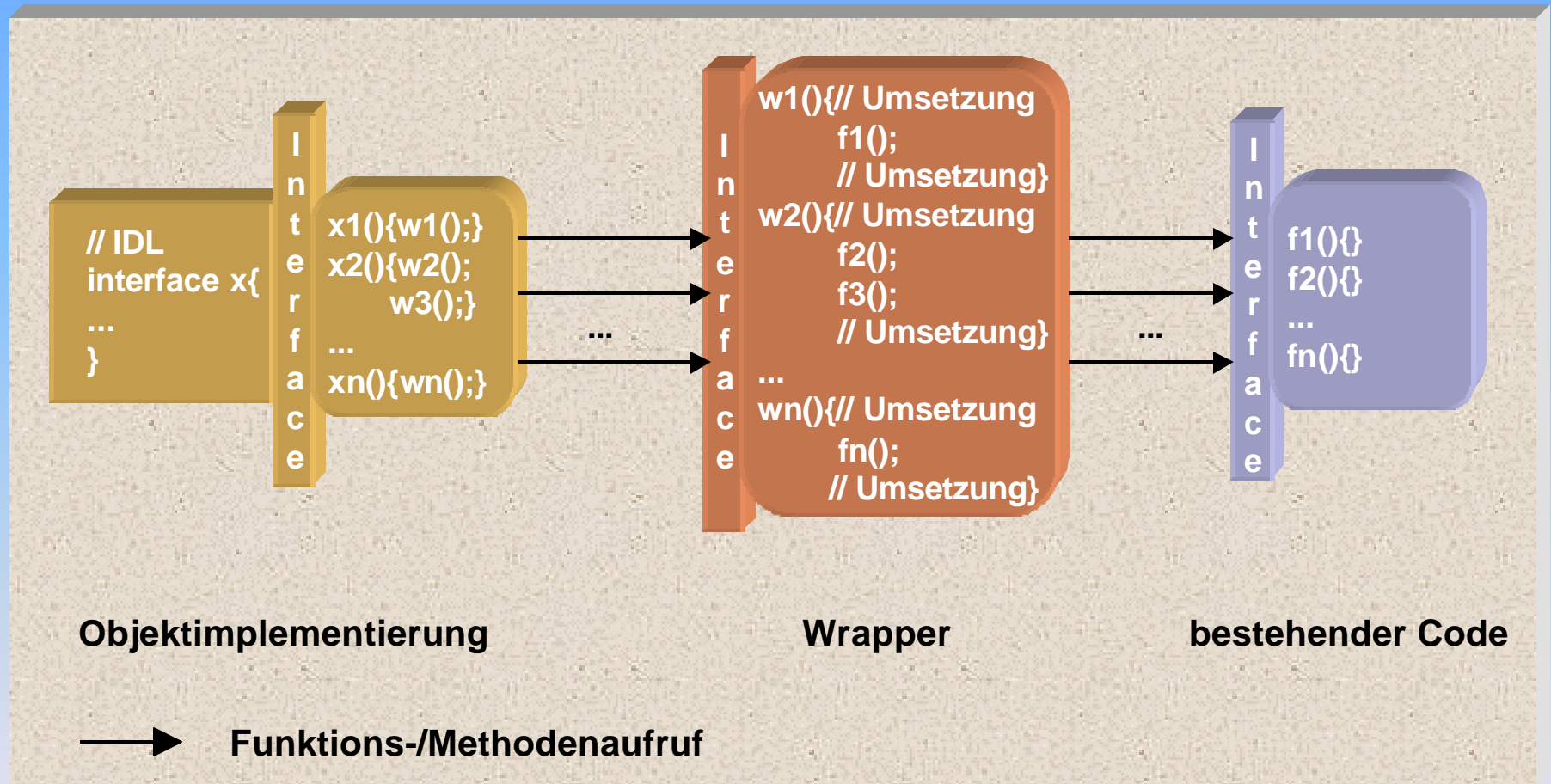
Abbildung zwischen Programmierschnittstelle und IDL-Schnittstelle



Faustregel:

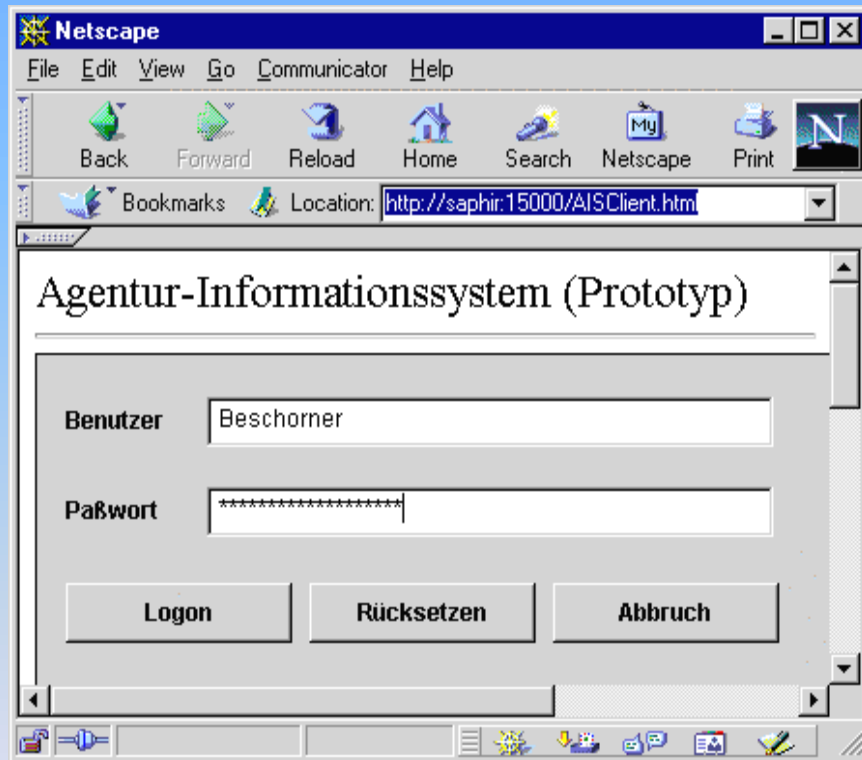
Je mehr Grunddatentypen der jeweiligen Sprache verwendet werden, desto einfacher die Abbildung.

Technische Umsetzung: Wrapper

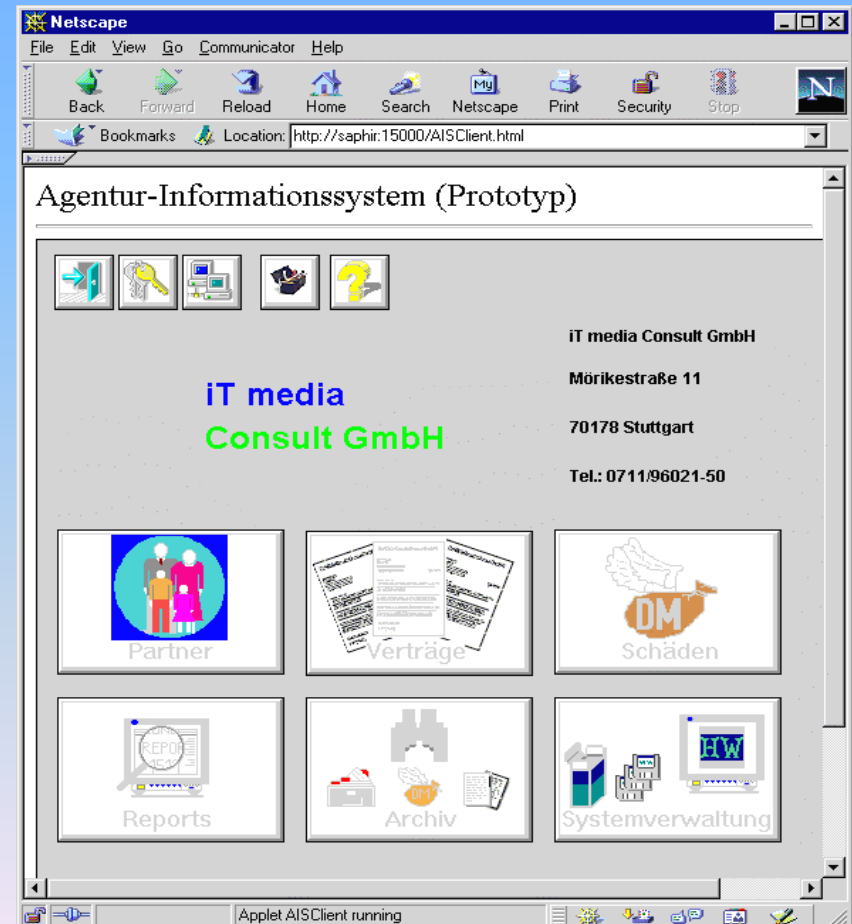


Browser-Anwendung: Masken werden ausgetauscht.

1. Anmelden



2. Hauptauswahl



3. Suchen

Netscape
File Edit View Go Communicator Help

Back Forward Reload Home Search Netscape Print Security Stop

Bookmarks Location: <http://saphir.15000/AISClient.html>

Agentur-Informationssystem (Prototyp)

Suchkriterien

Status: Datenbank: Partner 1 (DB/2)

Kunden-Nr.:

Name/Firma: Vorschau: 10001, Beschorner, Klaus

Zusatz:

Vorname:

Straße:

PLZ / Ort:

Zielgruppe:

29 Suchergebnisse

Partner-ID, Name/Firma, Vorname/Zusatz

7695	Best, Jakob
7719	Best, Frido
7721	Best, Frido
7853	Best, Menno u. Johannes
10001	Beschorner, Klaus

4. Bearbeiten

Netscape
File Edit View Go Communicator Help

Back Forward Reload Home Search Netscape Print Security Stop

Bookmarks Location: <http://saphir.15000/AISClient.htm>

Agentur-Informationssystem (Prototyp)

Allgemeine Daten

Anrede	<input type="text" value="Herr"/>	Vers.Unternehmen	<input type="text" value="Fantasie-Versicherung"/>
Titel	<input type="text" value="Dipl.-Inform."/>	Zielgruppe	<input type="text" value="Akademiker"/>
Name/Firma	<input type="text" value="Beschorner"/>	Kunden Nr.	<input type="text" value="10001"/>
Zusatz	<input type="text"/>	Geburtsdatum	<input type="text"/>
Vorname	<input type="text" value="Klaus"/>	Nationalität	<input type="text" value="deutsch"/>
Branche/Beruf	<input type="text" value="EDV"/>		

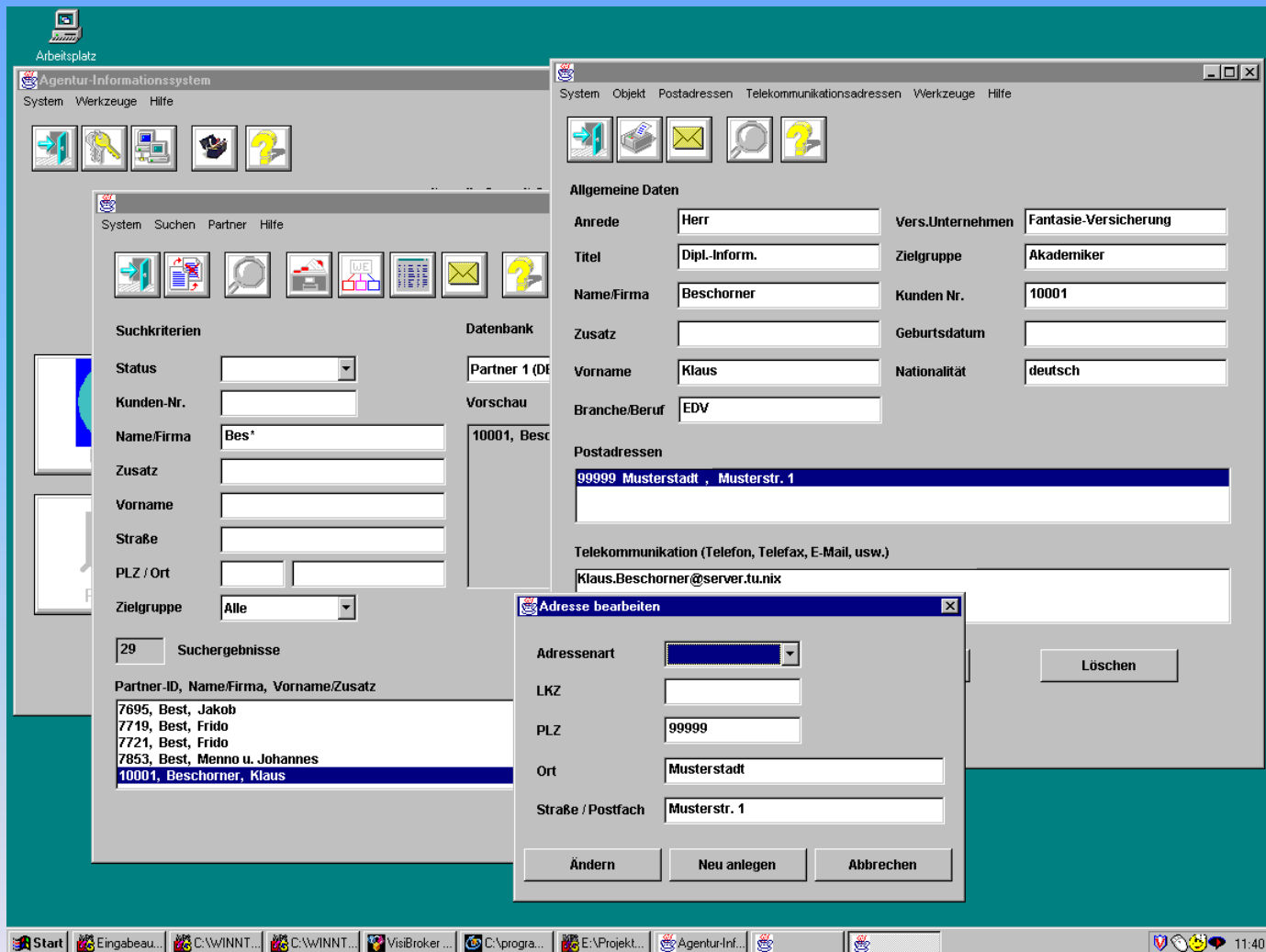
Postadressen

Telekommunikation (Telefon, Telefax, E-Mail, usw.)

Neu Ändern Löschen

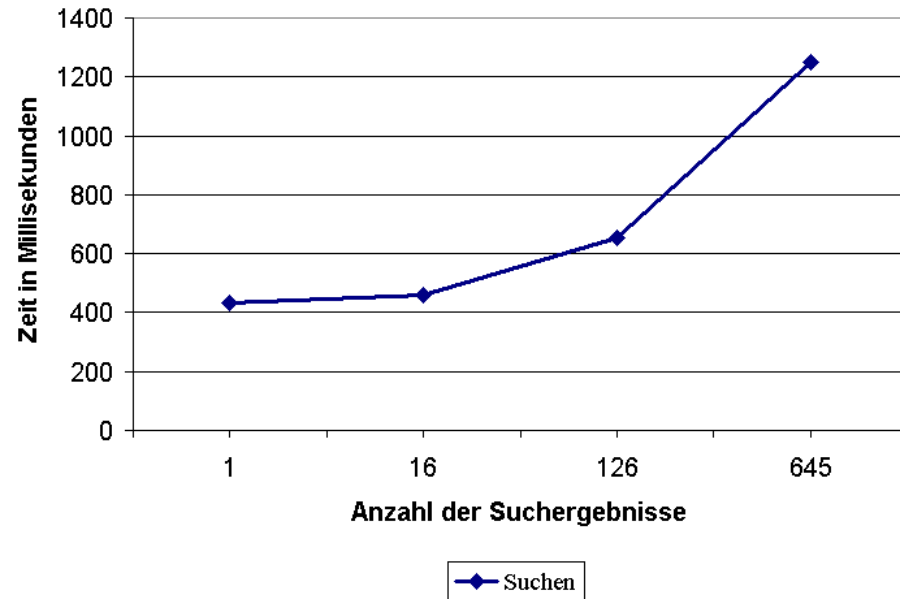
Applet AISClient running

Anwendung (Java-Applikation)

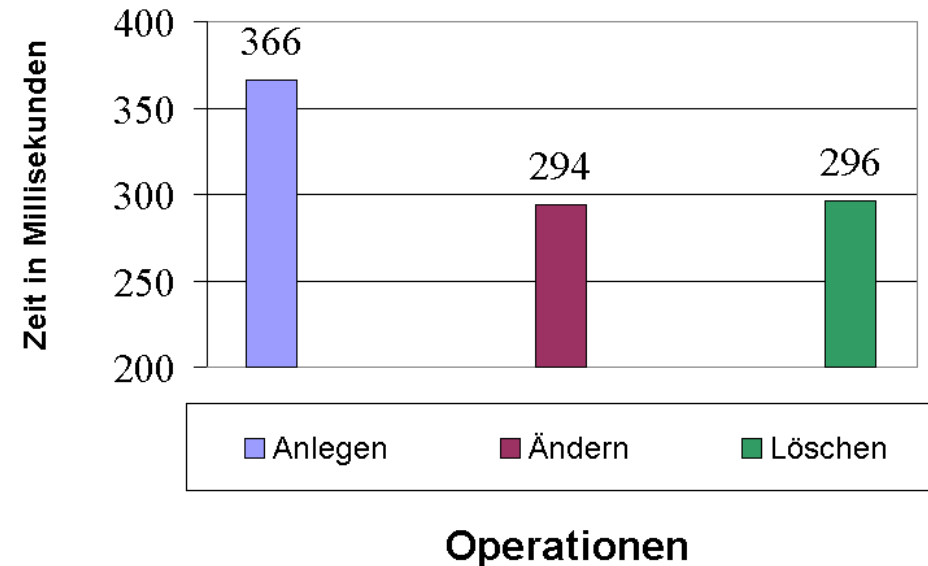


Zeitverhalten

Antwortzeiten für Suchanfragen



Antwortzeiten für Anlegen, Ändern, Löschen



Server (alle Komponenten): Pentium II/266 MHz

Client: Pentium 60 MHz

Datenbank: ca. 10000 Datensätze

Bewertung (CORBA & Java-Sprachanbindung)

- + keine Netzwerkprogrammierung**
- + projektübergreifende Auslegung von Objekten**
- + Kapselung von bestehendem Code**
- + W W W-Fähigkeit durch Java-Anbindung**
- + Java-Anbindung ist einfach**
- Einarbeitungszeit, Dynamik**
- ORB-Verfügbarkeit für Zielplattform (z.B. C++)**
- in Wirklichkeit kein Plug & Play (unterschiedliche ORBs!)**
- proprietäre Features von ORBs**

- **Migration eines bestehenden Systems in eine auf CORBA basierende Internet/Intranet-Anwendung**
- **Applet- und Applikationsprogrammierung**
- **Wiederverwendung von bestehendem Code**
- **Prototyp einer Anwendung und ihr Zeitverhalten**