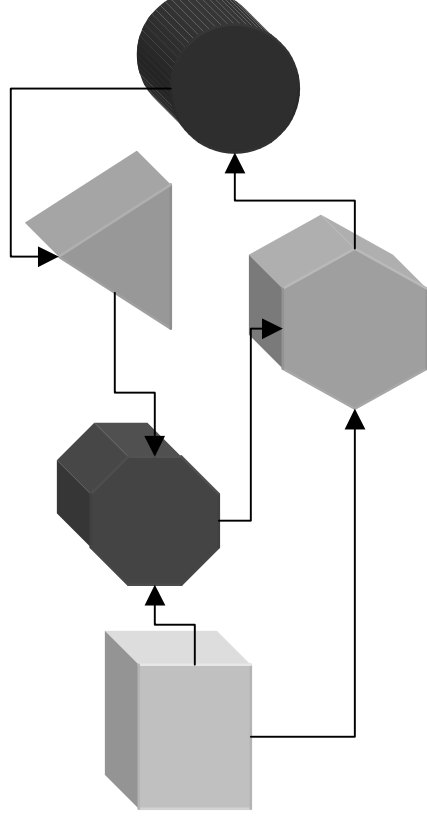


Service Discovery in CORBA



Markus Aleksy
aleksy@wifo3.uni-mannheim.de

Gliederung

- Einleitung
- Die Basiselemente der Interoperabilität
- Problematik der Einstiegsreferenz
- Naming Service
- Trading Service
- Zusammenfassung

Interoperabilität

Die Interoperabilität von CORBA basiert auf

zwei Elementen:

- Definition eines Kommunikationsprotokolls
- Einführung einer eindeutigen Objektreferenzierung

Protokolle

Für die Kommunikation zwischen verteilten

Objekten definiert CORBA zwei Protokolle:

- General Inter-ORB Protocol (GIOP) und das
- Internet Inter-ORB Protocol (IIOP)

Protokolle

GIOP

- ist ein abstraktes Protokoll
- definiert 8 verschiedene Nachrichtenformate
(Request, Reply,...)

IIOP

- spezifiziert, wie GIOP-Nachrichten über TCP/IP-
Verbindungen ausgetauscht werden können
- ist das Standardprotokoll von CORBA

Interoperable Object Reference

Eine Interoperable Object Reference (IOR)

besteht aus folgenden Elementen:

- einer Typidentifizierung
- einer Anzahl von *Interoperability Profiles* (IOPs) von denen jedes
 - eine Internet-Adresse,
 - eine Portnummer und
 - eine Objektidentifikation

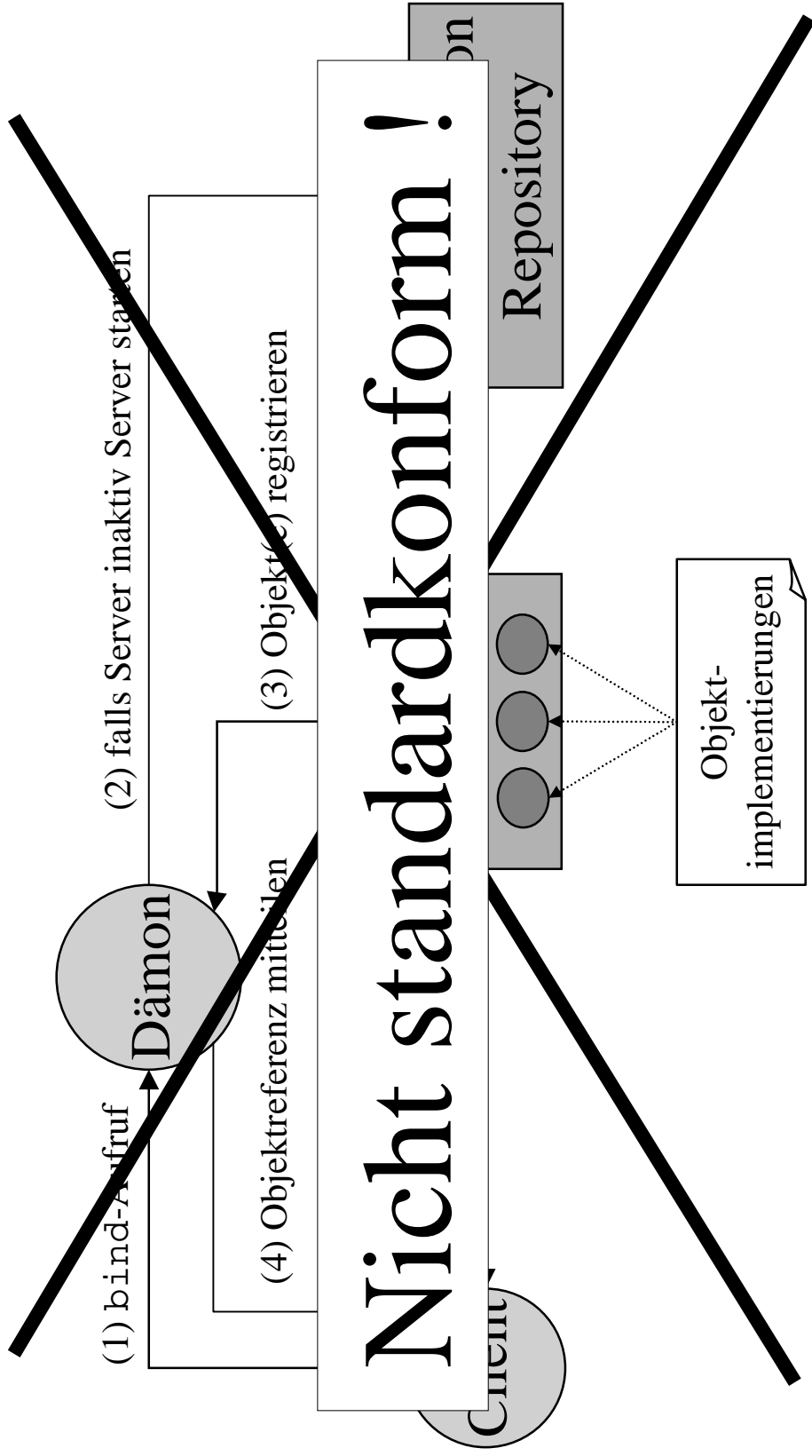
enthält.

Interoperable Object Reference

Beispiel für eine IOR in Klartext:

```
TypeID: "IDL:Counter:1.0"  
Profiles: 1. IIOP 1.0 123.11.76.2 5001  
"OB/ID+NUM.IDL:Counter:1.0.0"
```

Ermittlung der Einstiegsreferenz



Ermittlung der Einstiegsreferenz

CORBA definiert hierfür zwei Wege:

- die ORB-Operation `resolve_initial_references()`
- die ORB-Operationen `string_to_object()` und `object_to_string()`

Ermittlung der Einstiegsreferenz

Eingeführte Verbesserungen:

- Standardisierung von Kommandozeilenoptionen (ORBInitRef bzw. ORBDefaultInitRef)
- Spezifikation neuer - herstellerunabhängiger - Adressierungsarten (corbaLoc, corbaName)

Ermittlung der Einstiegsreferenz

Aufbau der corbaloc-Adressierung für das IOP-Protokoll:

corbaloc:[iiop]:[version@]host[:port] /
object-key

Protokoll - standardmäßig wird IOP benutzt und muß nicht extra angegeben werden

Protokollversion (standardmäßig wird 1.0 eingesetzt)

Objektidentifizierung (Details kennt nur der ORB)

Name oder Adresse des Rechners auf dem der Server läuft

Portnummer

Ermittlung der Einstiegsreferenz

corbaname-Adressierung:

- weitgehend gleicher Aufbau, aber eine andere Semantik
- enthält Informationen über den Ort des Namensdienstes
- enthält Informationen über den Pfad innerhalb des Namensdienstes, welcher zu der gewünschten Objektreferenz führt

Naming Service

Aufgabe:

Abbildung von Namen auf Objektreferenzen

Funktionalität:

- Verwaltet Namen und Kontexte
- ein Kontext kann beliebig viele andere Kontexte und Namen enthalten (hierarchischer Aufbau)
- ein Name ist in einem Kontext eindeutig
- ein Name führt zu genau einer Objektreferenz
- eine Objektreferenz kann unter verschiedenen Namen angemeldet werden

Naming Service

Vorteile:

- symbolische (verständlichere) Namen ersetzen Referenzen
- Entkopplung von Client und Server
- zentrale Anlaufstelle

Probleme:

- Ausfall des Namensdienstes (*single point of failure*)
- Skalierbarkeit

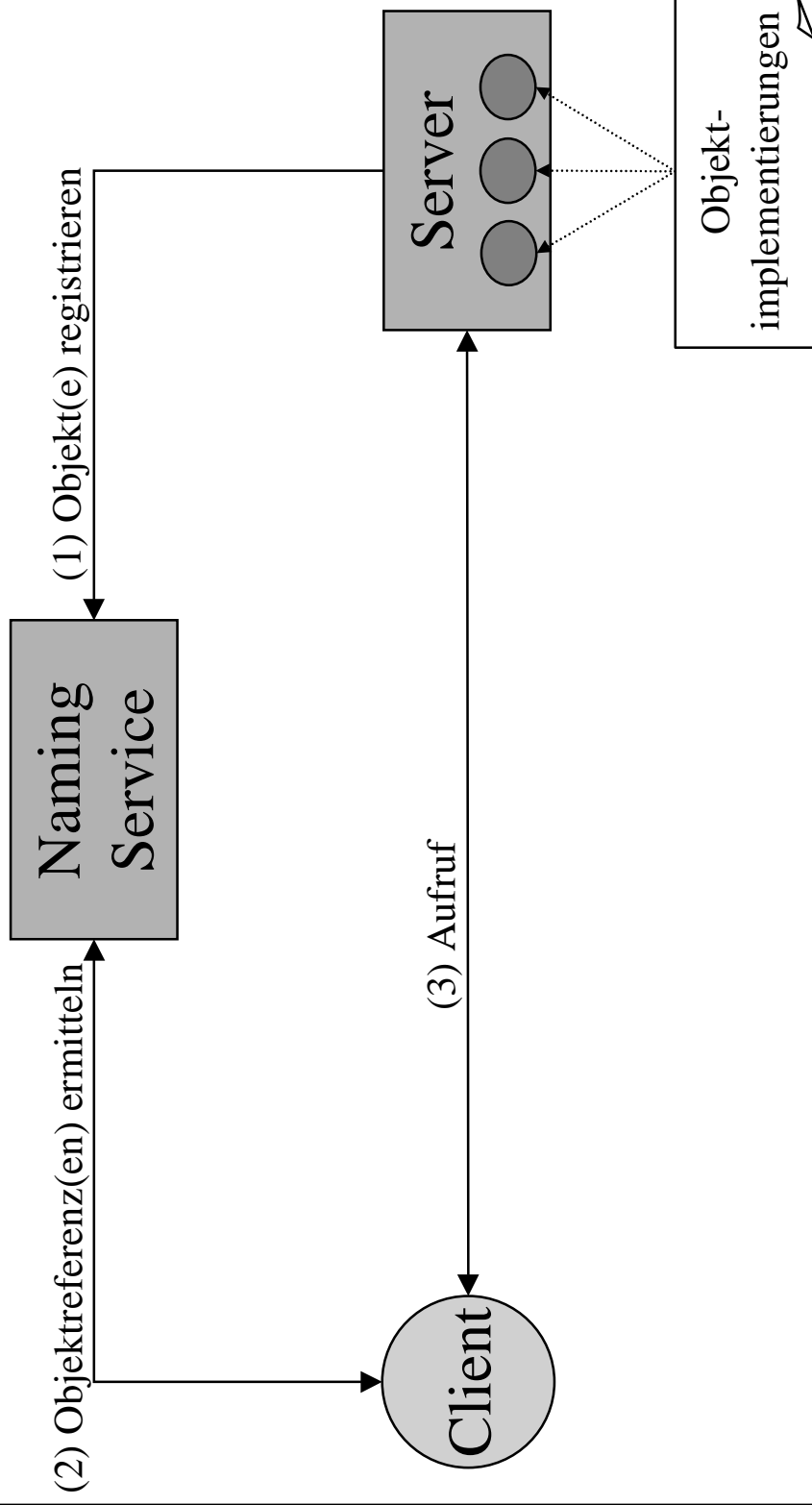
Naming Service

Welche Dienstanbieter gibt es?

- Auflistung aller Bindungen (Traversierung des Namensgraphen)
 - ☞ Zeitaufwendig bei vielen Bindungen
- Einsatz von Iteratoren, die einen Abruf von Teilmengen ermöglichen

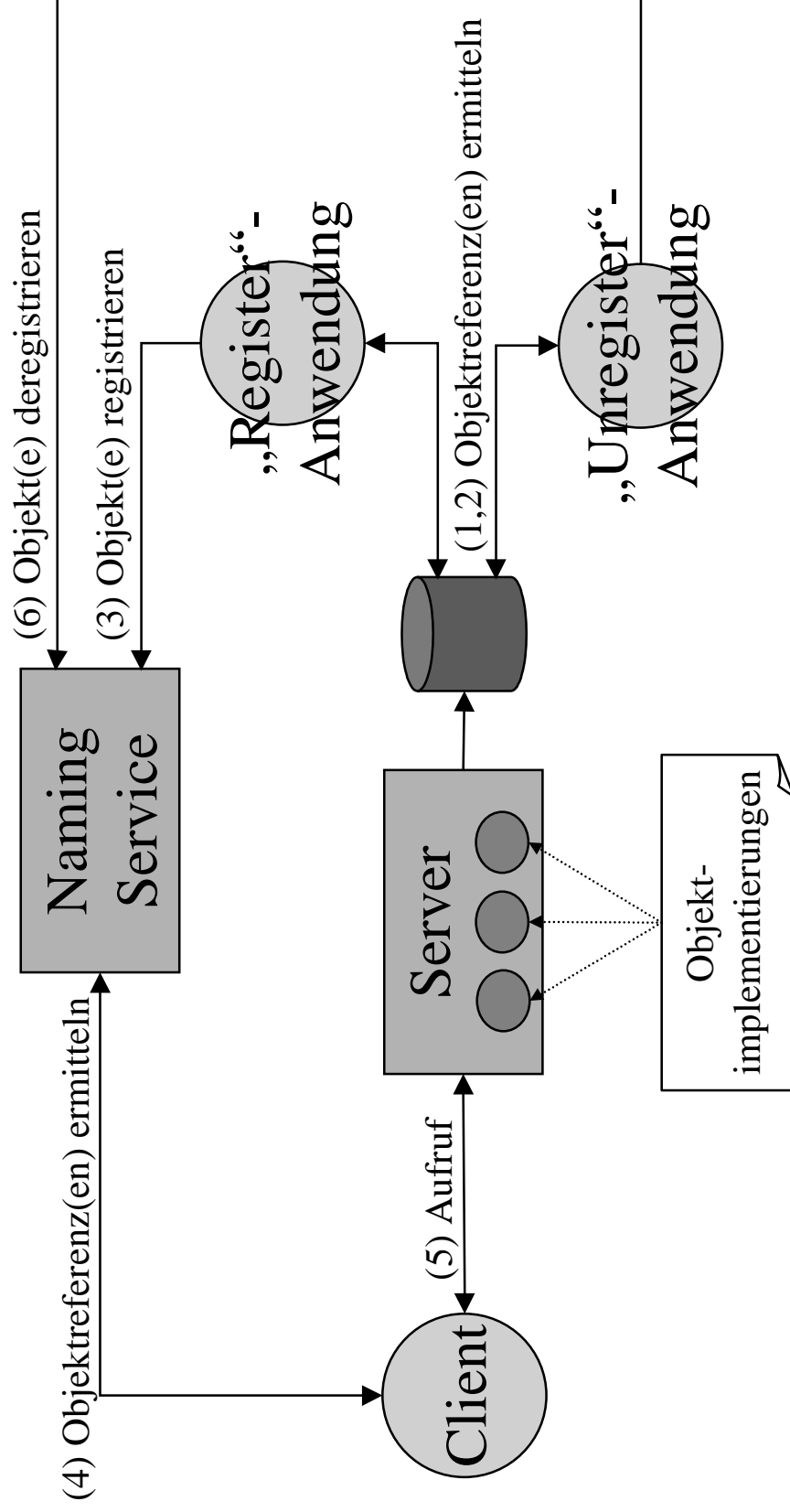
Naming Service

Grundsätzliche Vorgehensweise (1):



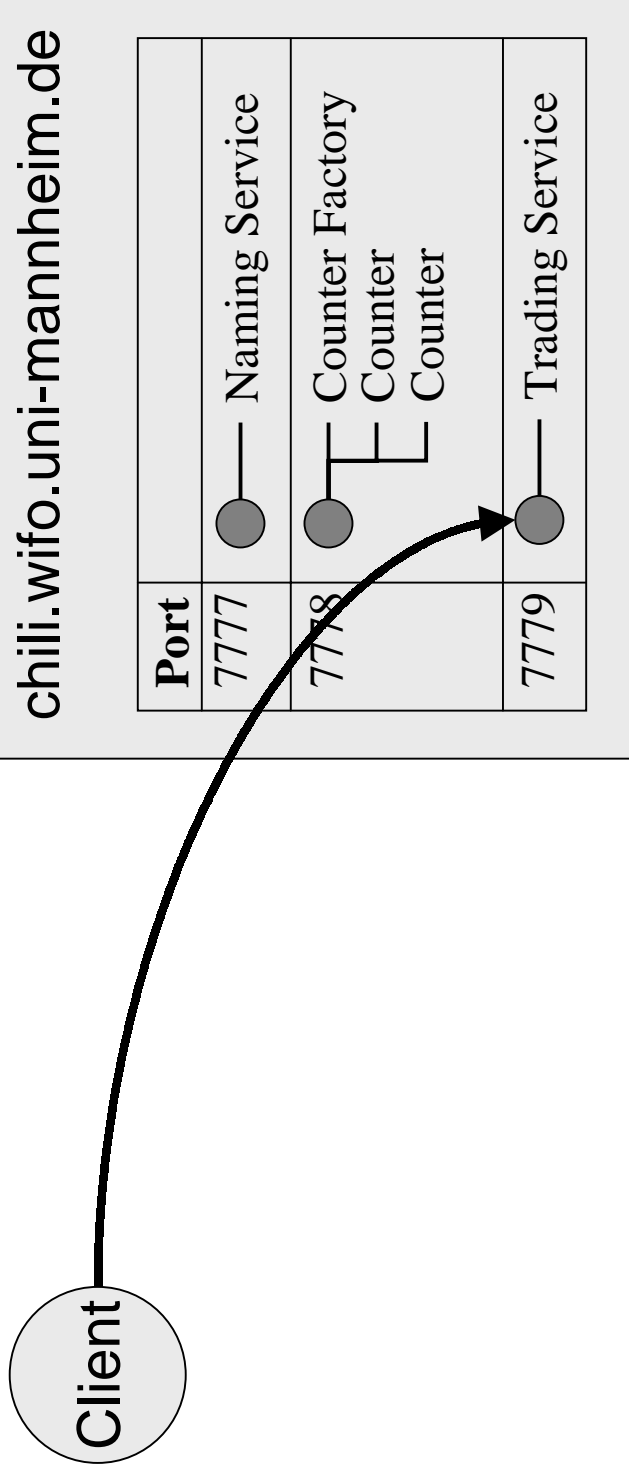
Naming Service

Grundsätzliche Vorgehensweise (2):



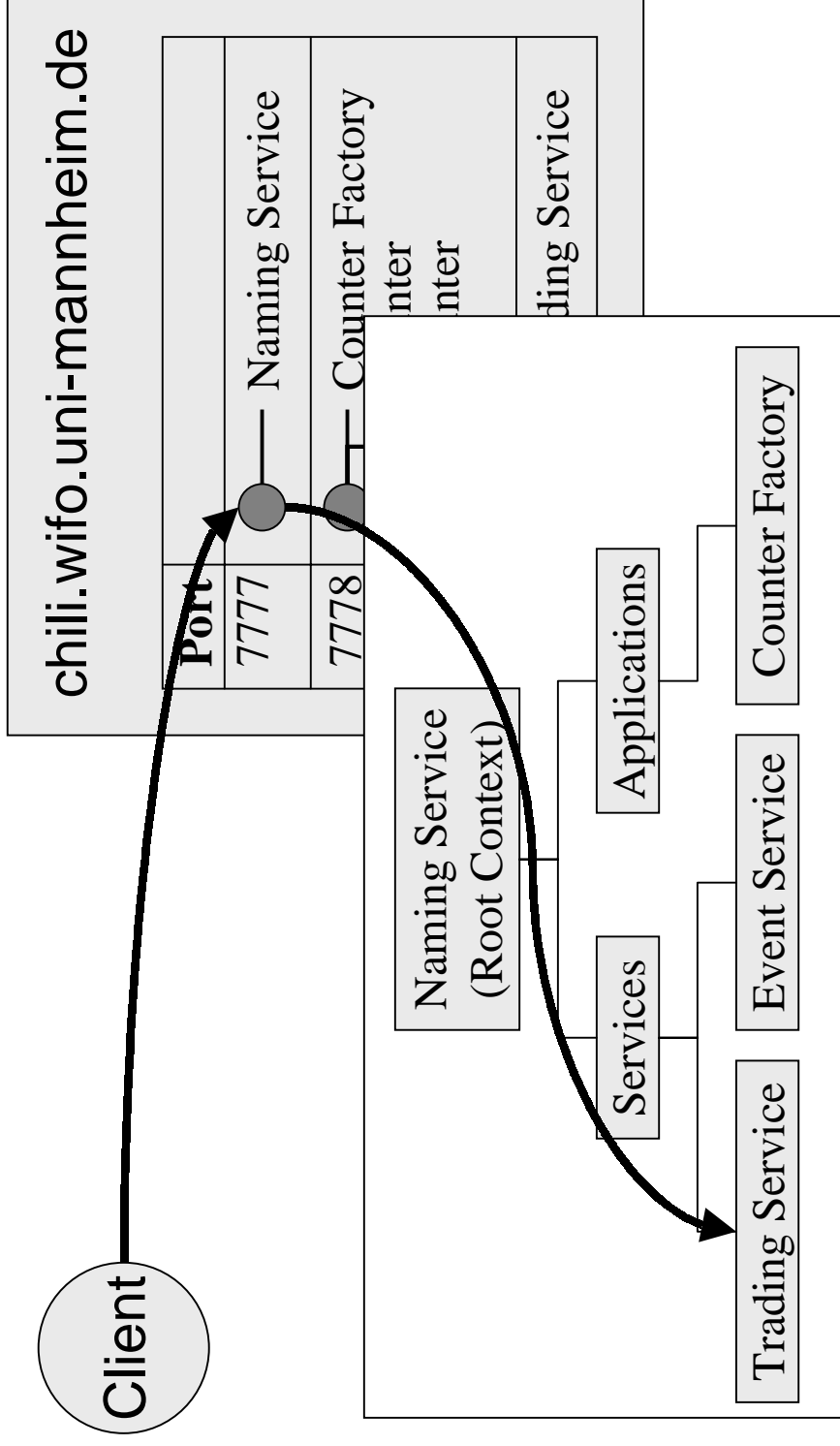
Ermittlung der Einstiegsreferenz

corbaloc::chili.wifo.uni-mannheim.de:7779/TradingService



Ermittlung der Einstiegsreferenz

corbaname::chili.wifo.uni-mannheim.de:7777/Services#TradingService



Trading Service

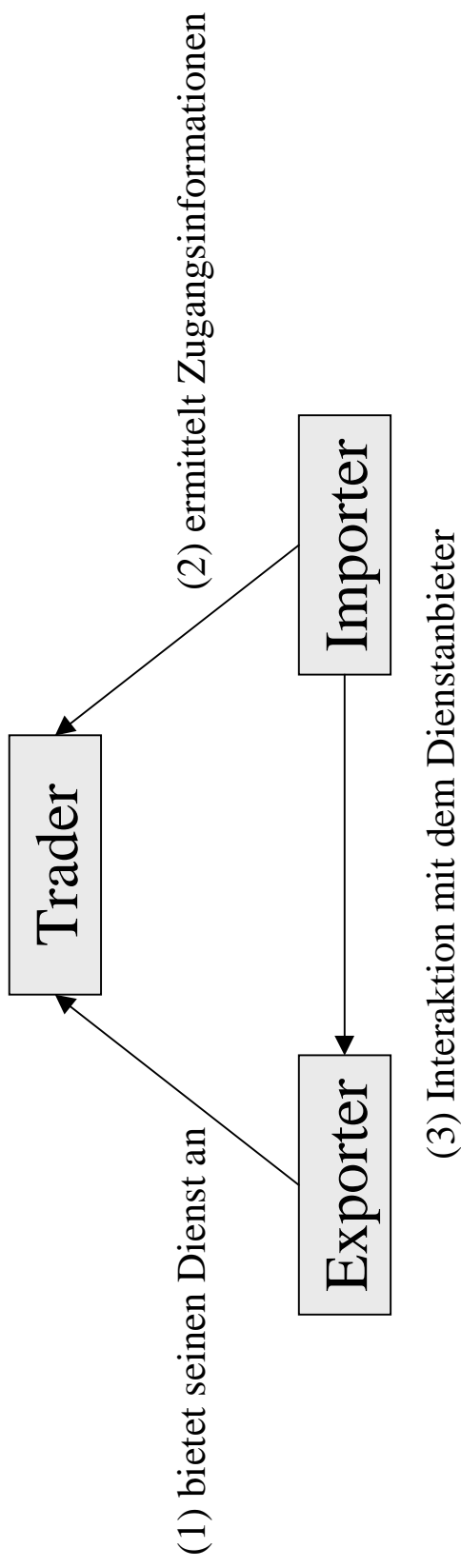
Aufgabe:

Vermittlung eines Dienstangebots an einen Dienstnutzer, wobei eine auf Dienstleistungsbasierende Suche eingesetzt wird.

Die Nutzer des Trading Service lassen sich in zwei Kategorien einteilen:

- Dienstanbieter (*Exporter*)
- Dienstnutzer (*Importer*)

Trading Service



Trading Service

Den Kern der Trader-Spezifikation bilden die folgenden Schnittstellen:

- Lookup
- OfferIterator
- Register
- Link
- Proxy
- DynamicPropEval

Zusammenfassung

Verbesserungen:

- Standardisierung der Kommandozeilenoptionen
- neue Adressierungsarten erleichtern die Lokalisierung der Server / Dienste

Aber:

- aus der Sicht der Anwender immer noch unzureichend
- Probleme beim Ausfall / Migration

Nutzung des Namensdienstes

```
try {
    ORB orb = ORB.init();
    org.omg.CORBA.Object obj =
        orb.resolve_initial_references("NameService");
    NamingContext nc =
        NamingContextHelper.narrow(obj);
    NameComponent[] aName = new NameComponent [1];
    aName[0] = new NameComponent();
    aName[0].id = "PrintManager";
    aName[0].kind = "Color";
    org.omg.CORBA.Object aObj = nc.resolve(aName);
    // ...
}
catch (Exception ex) {
}
```