

# Realisierung eines OMG-konformen Tradingdienstes

Vortrag zum Großen Beleg

Alexander Schlett : [as14@inf.tu-dresden.de](mailto:as14@inf.tu-dresden.de)

Betreuer: M. Fischer

Verantwortlicher Hochschullehrer: Prof. Hußmann

# Gliederung

- Einführung Trading-Dienst
- Ziel, Umfang von VTrader
- Architektur
- Aktueller Stand und Ausblick
- Beispielanwendung

# Einführung: Übersicht

- Trading-Dienst:
  - CORBA-Dienst, von OMG spezifiziert
  - „gelbe Seiten“ für Objekte im verteilten System
  - Entdeckung von Diensten zur Laufzeit
  - Funktionalität für Anbieter und Verbraucher
  - Können vernetzt sein

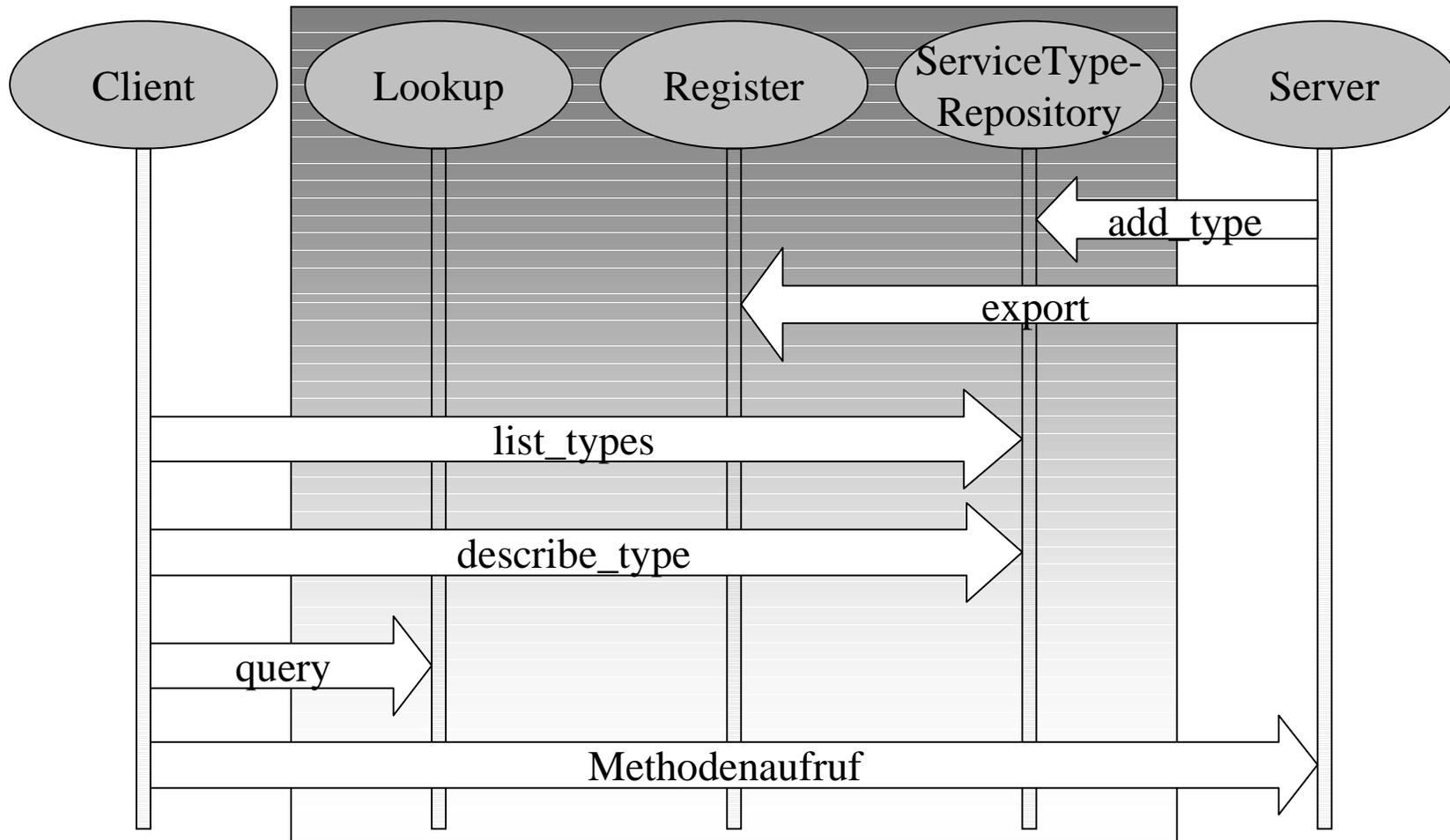
# Einführung: Trader-Schnittstellen

- Dienst-Schnittstellen:
  - ServiceTypeRepository
    - erzeugen und verwalten von Diensttypen
    - Diensttyp ist: Name, Schnittstelle, Eigenschaften
  - Register
    - anbieten, ändern, zurücknehmen von Diensten (export)
    - Angaben: Objektreferenz, Diensttyp, Eigenschaften
    - liefert Beschreibung von Diensten
  - Lookup
    - entdecken von Diensten
    - Parameter bestimmen Suchergebnis (z.B. Diensttyp, Eigenschaften)

# Einführung: Trader-Schnittstellen

- Admin
  - konfigurieren der Trader-Parameter
- Link
  - verwenden anderer Trader (Vernetzung)
- Proxy
  - ermöglicht Ermittlung der Objektreferenzen zur Anfragezeit
- DynamicPropEval
  - Auflösung von dynamischen Eigenschaften zum Anfragezeitpunkt

# Einführung: Szenario



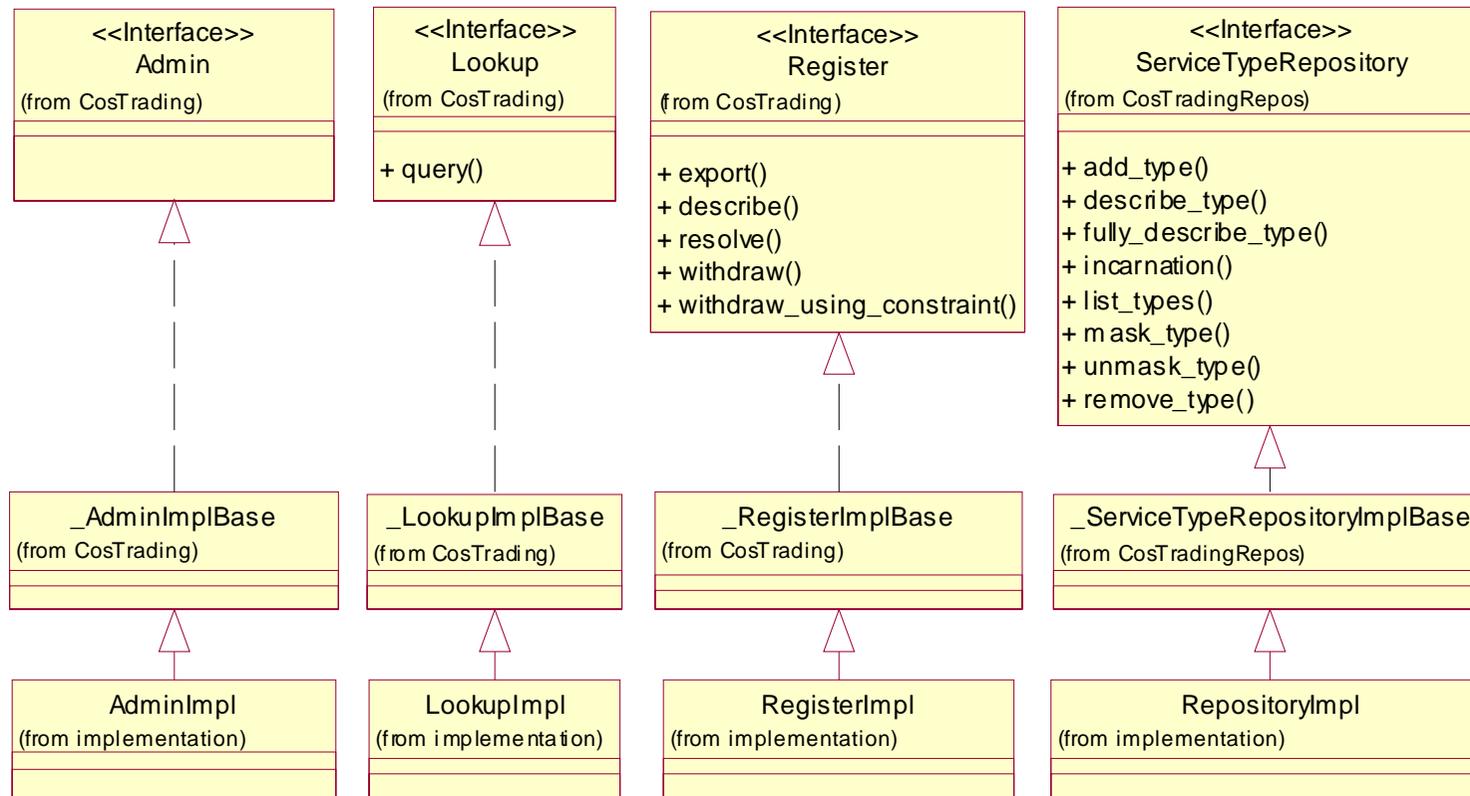
# Einführung: Trader-Arten

- **query-Trader:** Lookup
- **simple Trader:** Lookup, Register
- **stand-alone Trader:** Lookup, Register, Admin
- **linked Trader:** Lookup, Register, Admin, Link  
verwendet als Client: Lookup, Register
- **proxy Trader:** Lookup, Register, Admin, Proxy  
verwendet als Client: Lookup
- **full-service Trader:** Lookup, Register, Admin, Link, Proxy  
verwendet als Client: Lookup, Register

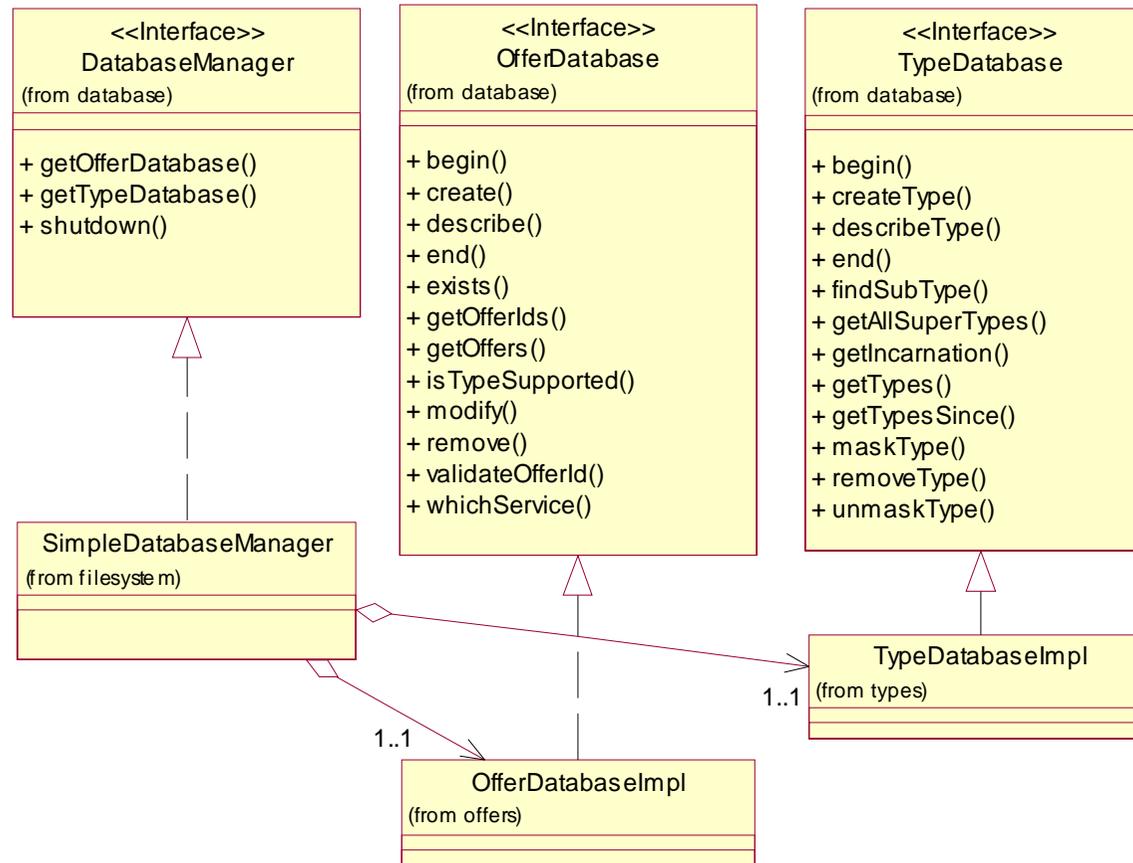
# Ziel/Umfang

- Implementierung eines einfachen Trading-Dienstes mit Java
- anschauliche Beispielanwendung
  
- Umfang für GB: stand-alone Trader
- Projektname: VTrader
- dabei Erweiterbarkeit im Vordergrund
- Datenbank einfach - über Serialisierung

# Architektur: Trader-Interfaces



# Architektur: Datenbankschnittstelle



# Aktueller Stand

- Zustand: lauffähiger Trader, TypeManager
- Datenbankschnittstelle:
  - Implementation für Serialisierung:
    - ServiceTypes werden als Hashtable in eine Datei serialisiert
    - Offers in einzelne Dateien serialisiert + Index extra
- Offene Teile:
  - komfortablere Fehlerbehandlung
  - GUI für Start des Traders
  - Vergleich mit anderen Implementierungen
  - Anwendungsbeispiel

# Ausblick

- Mögliche Erweiterungen:
  - Link, Proxy - Schnittstellen
  - Vorlieben, Restriktionen bei Anfragen  
(preferences, constrains)
  - Anbindung eines Datenbanksystems

# Beispielanwendung

- Zur Veranschaulichung der Trader-Funktionalität
- geplant:
  - Internet-by-call least-cost-router
  - Provider bieten ihren Dienst an
  - User können mit bestimmten Parametern günstigste Internet-Einwahl ermitteln
  - Parameter z.B. Ort, Uhrzeit, Bandbreite etc.