

Leistungsoptimierung der HfB-Recherche

Tim Paehler, IFGI WWU Münster

17. 11. 1997

1 Problematik

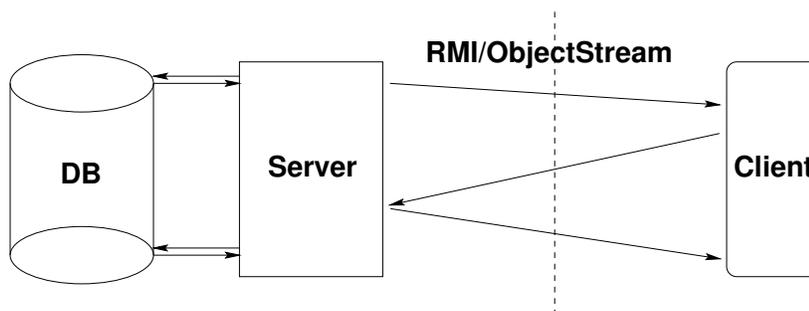
Größtes Performanzproblem: Hohe Dauer beim Aufbau des Metadatenbaums `DefTree`.
Bedingt durch:

- Große Anzahl an zu übermittelnden Daten
- Große Anzahl an zu erzeugenden Java-Objekten
- Nichtsequentieller Zusammenhang (Baumstruktur) der Daten

⇒ Großteil der Zeit wird für die Umwandlung der (readonly-)Daten von RDBMS-Format in Java-Objekt-Format aufgewandt.

2 Lösungsansätze

1. **Verwendung von prepared Statements:** Vorkompilierte SQL-Anweisungen, die variable Parameter aufnehmen.
2. **Verwendung eines dreistufigen Client/Server-Konzeptes:** Daten werden durch Server als Java-Objekte vorgehalten. Übermittlung durch:
 - Object Serialization
 - RMI (Remote Method Invocation)



Dreistufiges Client/Server-Konzept

3 Messungen

Verglichen wurde die Initialisierung des Metadatenbaums bei Verwendung der obigen Ansätze. Da die Messungen unter (unterschiedlicher) Last gemessen wurden, geben ihre Zahlenwerte lediglich Größenordnungen wieder. Gemessen wurde exemplarisch auf folgenden Rechnern:

- SPARC: Sun Ultra-2, 512 MB, Solaris 2.6
- PC/Linux: P166 PC, 64 MB, S.u.S.e Linux 5.1
- PC/NT: Hardware wie oben, Windows NT 4.0

3.1 Ausgangssituation: 2-Tier C/S, normale Statements

Rechner	$t[s]$
SPARC	144
PC/Linux	194
PC/NT	148

3.2 2-Tier C/S, prepared Statements

Rechner	$t[s]$
SPARC	47
PC/Linux	46
PC/NT	36

3.3 3-Tier C/S: Object Serialization

Rechner	$t[s]$
SPARC	14
PC/Linux	28
PC/NT	20

3.4 3-Tier C/S: RMI

Rechner	$t[s]$
SPARC	16
PC/Linux	30
PC/NT	16