

**Bewertung von**  
**Objektmanagementsystemen**  
**für Software-Entwicklungsumgebungen\***

**Lothar Schöpe**

Universität Dortmund  
Lehrstuhl Software-Technologie  
Fachbereich Informatik  
Postfach 500 500  
4600 Dortmund 50



\*Diese Untersuchung wurde im Rahmen des EUREKA-Projektes ESF gefördert.

## Inhaltsverzeichnis

(1)

- Methoden
- Benchmarks für Objektmanagementsysteme
- Praktikabilität der Objektmanagementsysteme

(2)

- konventionelle Benchmarks
- Standard für Objektmanagementsysteme

(3)

- abstrakter Benchmark
- komplexer Benchmark
- einfacher Benchmark

(4)

- Granularität des Datenmodells

(5)

## Methoden

- keine konventionellen Benchmarks
  - ausführbare Programme
  - Programmiersprache
  - Abfragesprache
  
- kein Standard für Objektmanagementsysteme
  - uneinheitliche Funktionalität bzgl.
    - Programmiersprache
    - Abfragesprache
  - uneinheitliche Architektur

## Erkenntnisziele

- **Methoden**
  - konventionelle Benchmarks
  
- **Benchmarks für Objektmanagementsysteme**
  - Abstraktionsniveau
  - Komplexität
  
- **Praktikabilität der Objektmanagementsysteme**
  - Performance
  - Administration

## Benchmarking für OMS

### Abstraktionsniveau

- abstrakter Benchmark
  - "abstrakte" Datenstruktur
  - vielfach implementierbar
  - OMS unabhängig
  
- komplexer Benchmark (Hypermodel)
  - komplexe Datenstruktur
  - hoher Aufwand
  - mögliche Implementierungsvarianten
  
- einfacher Benchmark
  - einfache Datenstruktur
  - geringerer Aufwand

## Komplexität

- Bottom-Up-Vorgehen
  - Reduzierung der Komplexität von OMS-Benchmarks
    - einfacher Benchmark
    - Messung der Performance von elementaren Datenmanipulationsoperationen
  - Ergebnisableitungen
  - komplexer Benchmark

## Abwägungspunkte

### Performance

- initiale Datenbank
  - Verwaltungsdaten
  - Nutzdaten
  
- Attributgröße
  - kleine Attribute
  - große Attribute
  
- verschiedene DB-Zustände
  - kalt
  - warm
  
- verschiedene Zugriffsschutzmechanismen
  - mit Transaktionsschutz
  - ohne Transaktionsschutz

## **OMS-Administration**

- **Administrationsfunktionen**
  - Benutzerinterface / Programmierschnittstelle
  - Integrierbarkeit
  
- **Recovery / Archivierung**
  - große Datenmengen
  - Versionsunterstützung
  - automatisches Recovery / Archivierung
  
- **Browser**
  - textuell
  - graphisch
  
- **Datenaustausch**

## Zusammenfassung

- unterschiedliche Anwendungsbereiche für Objektmanagementsysteme
- grobe Werkzeugdatenmodellierung
  - geringer Integrationsgrad
- geringe Performance
  - CASE- Anwendungen
  - Petri-Netz-Simulatoren
- integrierte Administrations-Werkzeuge
- "intelligentes" Recovery / Archivierung
  - Granularität der Daten
  - Semantik der Daten (Schemata)
  - Versionierung

kein optimales Objektmanagementsystem