

Einfluss von Programmierschnittstellen auf Anwendungsperformance

Dierk Lenz



DOAG 2008 Jahreskonferenz
Nürnberg im Dezember 2008



Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Die Firma Herrmann & Lenz wurde 1995 gegründet und hat aktuell 13 Mitarbeiter.
- Firmensitz: Burscheid (bei Köln).
- Beratung, Schulung und Fernwartung für das Thema *Oracle Datenbanken*.
- Schwerpunktthemen: Hochverfügbarkeit, Tuning, Migrationen und Troubleshooting.
- Viele DOAG-Aktivitäten (Leitung der Regionalgruppe NRW, Vorträge für Regios, SIGs und Konferenzen).



Weitere Aktivitäten

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung





Inhalt

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- 1 Einführung
- 2 Basics ...again
- 3 Die Qual der Wahl
- 4 Mögliche Performanceprobleme
- 5 Schlußbemerkung



Einführung

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung



Herausforderungen moderner Softwareentwicklung

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- In Quality
- In Budget
- In Time
- Time To Market

Aufgrund der beiden *Time*-Anforderungen werden die anderen Qualitätsmerkmale oft geopfert!



Konzepte und Planung sind wichtig!

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Die Auswahl der Basis für die Softwareentwicklung wird oft willkürlich getroffen.
- Die Art und Weise des Datenbankzugriffs wird nur nebenbei behandelt.
- Oft findet man auch eine Entwicklung mit unterschiedlichsten Werkzeugen – jeder Entwickler nimmt das, womit er/sie am besten klarkommt.



Basics ...again

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung



Die fünf Gebote der Datenbankprogrammierung

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- 1 Benutze Bindevariablen!
- 2 Verarbeite SQL-Befehle immer so, dass sie ohne erneutes Parsen wiederholt ausgeführt werden können!
- 3 Beachte, dass der Serverparameter `cursor_sharing` das Problem **NICHT** für Dich löst!
- 4 Benutze das Array Interface sowohl für das Holen der Daten (Array Fetch) als auch in Richtung Datenbank (Array INSERT usw.)!
- 5 Wenn Deine Entwicklungsumgebung, Deine Schnittstelle oder Dein O/R-Mapper dies nicht leisten kann – benutze das Ding nicht!



Die Qual der Wahl

**Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance**

Dierk Lenz

Einführung

Basics ...again

**Die Qual der
Wahl**

Mögliche
Performance-
probleme

Schluß-
bemerkung



Welche Schnittstellen gibt es z.B. unter Windows?

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

Der *Oracle Data Access Components* (ODAC) Download aus dem Oracle Technology Network enthält:

- Oracle ODBC Driver
- Oracle Objects for OLE
- Oracle Data Provider for .NET
- Oracle Providers for ASP.NET
- ...nicht zu vergessen JDBC Thin und JDBC OCI für Java, OCI, OCCI, PRO*C/C++ usw.

Und wo ist der Unterschied?



(Fast) Alles ist OCI

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Das *Oracle Call Interface* ist „am nächsten“ an der eigentlichen internen Schnittstelle.
- Alle anderen Schnittstellen setzen auf OCI auf!
- Ausnahme: Der JDBC Thin Driver, der per Definition nicht auf einem installierten Oracle Client aufsetzt und die Datenbankschnittstelle in Java „emuliert“.
- Jede Schnittstelle versucht, den Funktionsumfang des OCI „nach oben“ optimiert weiterzugeben, z.B. das *Oracle C++ Call Interface* OCCI und der Oracle Provider for .NET in objektorientierter Form.
- In den entsprechenden Sprachen (C++, C# usw.) hat man trotzdem die Wahl, entweder die spezialisierte Schnittstelle oder auch OCI direkt zu benutzen.



Noch mehr Auswahl

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Betrachtet man z.B. den Provider for .NET, so gibt es diesen in einer Microsoft- und einer Oracle-Variante.
- Vom Microsoft-Provider kann man sicherlich nicht die Unterstützung spezieller Oracle-Features erwarten – er implementiert lediglich die Methoden, die im Namespace `System.Data.SqlClient` definiert sind.
- Im Gegensatz dazu hat der .NET Provider aus dem Hause Oracle den Anspruch, sowohl in Performance als auch Funktionsvielfalt zu punkten.
- Auf der Microsoft-Seite ist natürlich auch der Anspruch der Programmierstandards zu verbuchen.



Einfacher Performancetest für 10000 INSERTs

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Mit einem C#-Programm werden über den Oracle Data Provider for .NET 10000 Datensätze in eine Tabelle ohne Index eingefügt.

| Test | Laufzeit | Faktor |
|----------------------------------|-----------|--------|
| 10000x Parse, Autocommit | 48,36 sec | |
| 10000x Parse | 14,59 sec | 3,3 |
| 1x Parse, 10000x INSERT | 6,82 sec | 7,1 |
| 1x Parse, 1000x Array-INSERT(10) | 0,79 sec | 61,2 |
| 1x Parse, 100x Array-INSERT(100) | 0,24 sec | 201,5 |

- Messergebnisse auf virtualisierter Client/Server-Umgebung gemessen – absolute Zahlen sind nicht repräsentativ!



Erstaunliche Ergebnisse

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Macht man das gleiche mit C und OCI, C++ und OCCl oder Java, so sehen die Ergebnisse verblüffend ähnlich aus!
- Die Schwankungen durch Seiteneffekte in der virtualisierten Umgebung sind größer als die Unterschiede zwischen den Sprachen.
- Vorsicht vor eingebauten Features wie dem automatischen COMMIT!



Was sagt uns das über Anwendungsperformance?

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Man kann viel falsch machen!
- Die Einhaltung der *fünf Gebote* ermöglicht immense Performanceverbesserungen.
- Aber: Die meisten Anwendungen bestehen aus mehr als einer Tabelle mit zwei Spalten!



Komplexität von Anwendungen

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Bereits einfache Anwendungen haben Relationale Datenmodelle mit 100 oder mehr Entitäten – sprich: Tabellen.
- Da Anwendungen heute meist objektorientiert programmiert werden, wird das Datenmodell zweimal erstellt.
- An dieser Stelle setzen O/R-Mapper an (Hibernate, TopLink, LINQ to SQL, ...).
- Da der Datenbankzugriff dann abstrakt im Hintergrund abläuft, wird über Details nicht mehr nachgedacht.



Mögliche Performanceprobleme

**Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance**

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

**Mögliche
Performance-
probleme**

Schluß-
bemerkung



Mögliche Performanceprobleme: Zuviel Daten

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Wenn die Objekte zu groß dimensioniert werden, entspricht ein Objekt vielen Tabellen in der Datenbank.
- Benötigt man einzelne Informationen, so wird per Automatismus das gesamte Objekt geladen.
- Noch „schlimmer“ wird es, wenn durch bestimmte Aktionen ganz Datenbereiche in der Datenbank gesperrt werden.



Mögliche Performanceprobleme: Sperrstrategien unterschiedlicher Werkzeuge

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Je abstrakter die eingesetzten Werkzeuge sind, desto weniger Einfluss hat man auf Dinge wie Sperrstrategien.
- Auch Entwickler, die sich nicht absprechen, verfolgen oft unterschiedliche Sperrstrategien.
- Dies kann zu Deadlocks und/oder langen Wartezeiten für Anwender führen.



Mögliche Performanceprobleme: Falscher Fokus

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Bei Performanceproblem ist *der langsame Server* ein dankbares Opfer.
- Es wird viel Energie in Serveranalysen gesteckt.
- Die Anwendung wird als problemfrei angenommen.
- Hier steckt oft das größte Potential.



Mögliche Performanceprobleme: Programmierstandards

Einfluss von
Programmier-
schnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Grundsätzlich sind Standards selbstverständlich positiv zu bewerten.
- Allerdings werden aus vielen Dingen Prinzipien abgeleitet, deren Auswirkungen dramatisch sein können.
- Beispiel datenbankunabhängige Programmierung: ein Killer-Argument für Softwarehäuser. Egal, was der Kunde möchte, unsere Anwendung unterstützt es, solange es einen xyz-Treiber gibt.
- Beispiel Multi-Threading: Sehr beliebt, insbesondere seit es Multi-Core-Prozessoren gibt. Erlaubt es aber auch, die Langsamkeit des Datenbankzugriffs in einem Hintergrund-Thread zu verstecken...



Mögliche Performanceprobleme: „Skalierbare“ Konzepte

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Der ADO.NET Disconnected Layer wird im Rahmen von .NET als Grundlage für Anwendungen auf Basis der *Windows Presentation Foundation* (WPF) bereitgestellt. Oracle liefert hierzu mit den *Developer Tools for Visual Studio* Erweiterungen, die speziell auf Oracle angepaßt sind.
- Trotzdem bleibt das Konzept des Disconnected Layers bestehen: Nach dem Laden der Daten in die Anwendung wird die Verbindung zur Datenbank **getrennt!**
- Bei Bedarf können die lokal durchgeführten Änderungen in die Datenbank zurücksynchronisiert werden.
- Dies ist natürlich lediglich ein Beispiel; viele Frameworks und O/R-Mapper arbeiten nach diesem Prinzip.



Mögliche Performanceprobleme: „Skalierbare“ Konzepte

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Problematisch ist lediglich die Konsistenz in der Datenbank. Werden die Daten vor dem Zurückschreiben auf *Staleness* geprüft, erfordert dies zusätzlichen Aufwand – und eine zusätzliche Logik für den Fall, dass sich die Daten zwischen Laden und Schreiben geändert haben. Wird keine Prüfung gemacht, so können Inkonsistenzen die Folge sein.



Schlußbemerkung

**Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance**

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

**Schluß-
bemerkung**



Was ist also zu tun?

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Datenmodellierung und Datenbankprogrammierung sind keine Nebenprodukte.
- Die beiden Teile der Softwareentwicklung entscheiden später über Performance und Skalierbarkeit der Anwendung.
- Es ist durchaus legitim, sich von Frameworks, O/R-Mappern und weiteren Werkzeugen helfen zu lassen.
- Es sollte jedoch Know-How über die Arbeitsweise der Komponenten sowie die Bereitschaft, für spezielle Anforderungen spezielle Techniken einzusetzen, vorhanden sein.



Herrmann & Lenz – weitere Vorträge bei der DOAG-Jahreskonferenz

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- Susanne Jahr: **Active Data Guard**
Mittwoch 3.12., 13:00 Uhr, Tokio
- Wilhelm Breßer: **Unsichere Bindevariablen**
Mittwoch 3.12., 14:00 Uhr, Neu Delhi
- **Performance-Aspekte bei der
Anwendungsprogrammierung für Oracle Datenbanken**
Schulungstag – Donnerstag 4.12.



Weitere Fragen?

Einfluss von
Programmierschnittstellen
auf
Anwendungs-
performance

Dierk Lenz

Einführung

Basics ...again

Die Qual der
Wahl

Mögliche
Performance-
probleme

Schluß-
bemerkung

- `dierk.lenz@hl-services.de`
- `http://www.hl-services.de`
- Hier gibt es nach der Konferenz auch unsere Vorträge in der aktuellsten Fassung zum Download!