

# Optimieren von Solaris für Oracle Datenbanken

Christian Wolbert  
AS-SYSTEME Unternehmensberatung GmbH  
Stuttgart

## Schlüsselworte:

Infrastruktur, Solaris, Performance, Optimierung, Oracle Datenbank

## Abstract

Optimale Prozessverarbeitung erhält man, wenn das Betriebssystem und die Anwendung aufeinander abgestimmt sind. Die Interaktionen von einer Oracle-Datenbank mit dem Betriebssystem umfassen komplexe Parameter in vielen Bereichen. Kritisch ist häufig der I/O-Bereich, aber auch bezüglich der Arbeitsspeicherverwaltung, dem Netzwerk und dem Prozessmanagement gibt es Konfigurationsanforderungen und Potential zur Optimierung.

Da durch den jeweiligen Einsatzzweck einer Datenbank viele interne Parameter der Datenbank vordefiniert sind, besteht für den Systemadministrator die Herausforderung darin, die Konfiguration, Überwachung und Optimierung von der Betriebssystemseite aus abzustimmen und zu verwalten. Kerntechnologien von Solaris wie ZFS, Virtualisierung in Kombination mit Ressourcenmanagement wollen gut genutzt werden. Der Vortrag ist ein Best-Practice Guide, ein Leitfaden für Systemadministratoren, der typische Problemfälle aufzeigt und Ansätze zur Optimierung liefert.

## Einleitung: Performance-Analyse und Optimierung

Eine Voraussetzung für die Optimierung des Betriebssystems ist eine fehlerfreie Konfiguration und Funktion der Datenbank. Interne Parameter der Datenbank sollten bezüglich des beabsichtigten Verwendungszweckes der Datenbank ausgelegt sein. So ist zum Beispiel die Standardgröße vom *db\_block\_size* Parameter mit 8KB für viele Zwecke die optimale Größe. Hingegen kann im Datawarehouse-Bereich, wo große Datenbestände zusammengeführt und analysiert werden, ein höherer Wert besser sein.

Stehen die internen Abläufe und Konfigurationsparameter der Datenbank fest, geht es bei der Performance-Analyse darum, die Interaktion zwischen Betriebssystem und Anwendung unter die Lupe zu nehmen.

Solaris stellt eine Vielfalt an Werkzeugen und Befehlen - unter anderem *dtrace* - zur Verfügung, mit der man die Interaktion der Datenbank mit dem Betriebssystem untersuchen kann, sowie Engpässe und Ressourcenauslastung erkennen und überwachen kann. Nach der kurzen Vorstellung der entsprechenden Werkzeuge und allgemeinen Richtlinien und Empfehlungen, betrachtet der Vortrag die vier kritischen Bereiche (siehe Abbildung 1), in denen eine Optimierung stattfinden kann: I/O, Memory, Netzwerk und CPU.

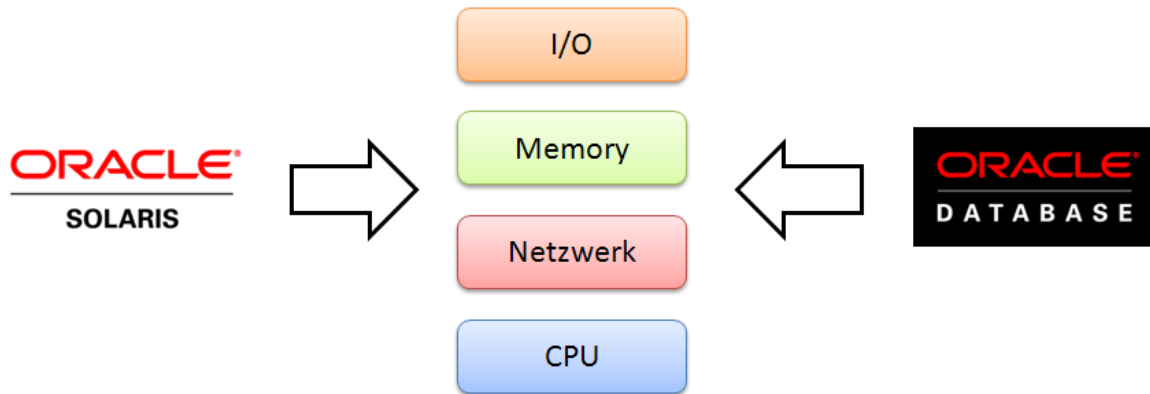


Abb. 1: Bereiche der Interaktion zwischen Betriebssystem und Datenbank

## Optimierungsbereiche

### Teil 1 – Das I/O - Subsystem

Was den I/O betrifft, so gehört das I/O-Subsystem zu den kritischsten Bereichen. Hier besitzt man auch die größten Möglichkeiten optimierend einzugreifen. Fokus der Betrachtung ist das ZFS-Dateisystem. Bezüglich der Datenbankkomponenten (Database Files, Redo Logs, Archive Logs, ...) gibt es verschiedene Anforderungen an das darunter liegende Dateisystem.

Neben der generellen Empfehlungen über die Anzahl und Struktur der verwendeten ZFS-Pools, der Frage, wo es sinnvoll ist, den ZFS-Pool mit *Cache Devices* oder einem *Intent Log* auszustatten, gibt es einen Überblick, welche ZFS-Parameter wie eingestellt werden sollten, um einen optimalen I/O-Transfer zu erreichen. ZFS-Parameter wie zum Beispiel *recordsize* oder *compression*.

### Teil 2 – Memory

Arbeitsspeicher wird in zwei Bereichen benötigt, der SGA und dem PGA. Typischerweise wird in der SGA benötigt. Da Intimate Shared Memory unter Solaris dynamisch ist, wird eine Begrenzung mittels Resource Management benötigt. Es folgen Empfehlungen über die Verwendung des Resource Managements und dann gibt es einen Einblick über die Performanceoptimierung mittels *Multiple Page Sizes*.

### Teil 3 – Netzwerk

Datentransferraten, Nicht verarbeitete Pakete, Retransmissionen mit Duplikaten sind nur einige Stichwörter, die auftreten. Jumbo-Frames und Aggregate sind ein probates Mittel für eine Leistungssteigerung im Netzwerk. Modulparameter vom TCP-Protokoll können in bestimmten Situationen auch Abhilfe schaffen. Neu ist unter Solaris 11 die Umgestaltung des kompletten Netzwerkstacks (Crossbow). Hier gibt es zusätzliche Möglichkeiten, die Gesamtpformance bei der Verarbeitung von Paketen zu steigern (Stichwort: Hardware Lanes, siehe Abbildung 2).

Deutlich reduzierte *Interrupts*, *Context-Switches* und *Lock Contention* führen zu einer deutlichen Leistungssteigerung. Wichtig ist auch die Möglichkeit über eine Bandbreitenregulierung den Datenfluss zu kontrollieren.

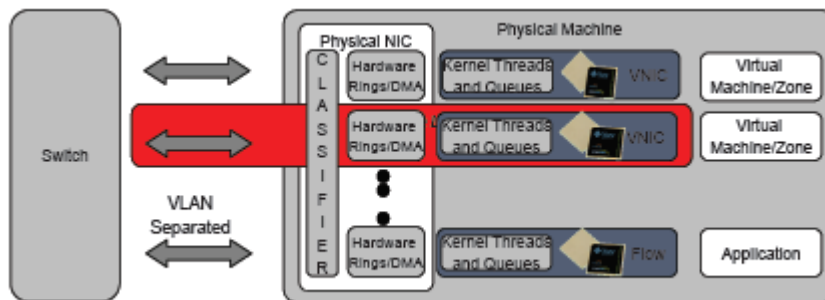


Abb. 2: Solaris 11 – Crossbow Hardware Lanes (Bildquelle: Oracle)

#### Teil 4 – CPU

CPU's sollen bezüglich der Anwendung ungestört den spezifischen Aufgaben nachgehen. Dies lässt sich nur durch dedizierte Ressourcen wirksam erreichen. Es gibt einen Überblick über die Best-Practice Vorgehensweise bei der Implementierung des Ressourcenmanagements. Ein anderer Aspekt besteht darin, den Scheduling-Mechanismus abzuändern: Prioritisierung, garantierte CPU-Anteile können zur internen Verbesserung bei der Verarbeitung von Threads helfen.

Kontaktadresse:

Christian Wolbert  
 AS-SYSTEME Unternehmensberatung GmbH  
 Wankelstraße 1  
 D-70563 Stuttgart

Telefon: +49 (0) 711-90146-15  
 Fax: +49 (0) 711-90146-46  
 E-Mail: christian.wolbert@as-systeme.de  
 Internet: www.as-systeme.de