

# ORDIX news

Das IT-Magazin der ORDIX AG



## Oracle: Buffer Pool Advisor

18 | Vorschläge für die Dimensionierung des Database Buffer Cache

- 6 | Architektursicht - Dokumentation der Softwarearchitektur
- 26 | Neue Reihe: Hibernate für Fortgeschrittene - Die Batch-Fetching-Strategie für Relationen
- 34 | Stakeholder Management – aber richtig! - Projektmanagement in der Praxis
- 37 | Der Panther hält Wache - Erweiterte Benutzerverwaltung unter Informix



# HIBERNATE, die richtige BIBLIOTHEK für ihre Datenbank!

Besuchen Sie unser  
**Seminar** zu Hibernate  
und Java Persistence API!

## Hibernate und die Java Persistence API

Die Hauptaufgabe von Hibernate ist das Object Relational Mapping. Es ermöglicht, gewöhnliche Objekte mit Attributen und Methoden in relationalen Datenbanken zu speichern und aus entsprechenden Datensätzen wiederum Objekte zu erzeugen.

In unserem Seminar werden Sie mit der Hibernate-Implementierung von Java Persistence API vertraut gemacht.

**Die nächsten Termine:**      26.03.-30.03.2012    Seminarzentrum Wiesbaden  
   25.06.-29.06.2012    Seminarzentrum Wiesbaden

Weitere Informationen und Anmeldung unter:  
<http://training.ordix.de/>

**ORDIX AG**

# Clouds ...

Paderborn, März 2012



Cloud-Computing soll der Renner auf der CeBIT (die gibt es tatsächlich noch) gewesen sein. Bei diesem nebulösen Gebilde, das Wikipedia als „Ansatz, abstrahierte IT-Infrastrukturen ... dynamisch an den Bedarf angepasst über ein Netzwerk zur Verfügung zu stellen“ beschreibt, handelt es sich sehr häufig darum, dass IT wieder zur Geheimnis umwobenen „Wissenschaft“ wird. Rechnerkapazität, Daten etc. werden ja nun schon seit mehr als einem Jahrzehnt über Netzwerke zur Verfügung gestellt. Wird uns also Altbekanntes wieder als nagelneu verkauft?

Selbst der nicht immer korrekt zitierte bzw. fremd verwendete<sup>1</sup> wissenschaftliche Dienst des Bundestages hat sich damit beschäftigt<sup>2</sup>, was sich ganz nett lesen lässt („Daten nicht mehr lokal ... gespeichert (z.B. auf einem Tischrechner“). Haben Sie auch einen Tischrechner? Na gut, eigentlich ist das ja für unsere Politikelite geschrieben.

Cloud auf Mallorca, wo ich gerade für eine Woche bin, hat hier leider eine ganz andere Bedeutung. Während in Deutschland schönsten Wetter ist, dreht schon seit zwei Tagen eine riesige Wolke ihre Runde über die Balearen. Dies veranlasste den englischen Golflehrer zu sagen, dass es die Woche davor bestes Wetter gab und jetzt eben „because of the cloud“ schlecht ist.

Zurück zur anderen Cloud: Nahezu jeder bietet dazu mittlerweile etwas an, ein schicker Zug auf den es scheinbar gilt aufzuspringen. Doch aufgepasst, nebulöse Sachen verschwinden einfach. Ich hoffe, es sind weder Ihre Daten dabei noch sonst irgendetwas, das Ihnen wichtig erscheint.

Etwas weniger nebulös sind hoffentlich unsere Artikel (abgesehen vielleicht vom „Buffer Pool Advisor“ 🍌). Oracle ist mit neuen „Produkten“ und Themen gleich vier Mal vertreten (GoldenGate, APEX, Buffer Pool Advisor, Time Zone Files), vom Administrator über den Programmierer bis hin zum Management werden bei diesen Artikeln alle bedient.

IBM ist immerhin mit Informix (letzter Artikel in der Reihe „Neues zu 11.70“) und DB2 Work Load Manager vertreten, der helfen soll, die Last auf DB2 Datenbanken besser zu verteilen.

Neues zum Thema Lastverteilung gibt es auch von Java 7 zu berichten. Mit Hilfe eines Frameworks sollen Hardwareressourcen (mehrere Prozessoren oder Kerne) besser ausgelastet werden können. Sicher mehr als eine gute Idee, denn vielfach werden Ressourcen nicht gut oder nur einseitig ausgelastet.

Für die „Datenbank“ unter den Java Programmierern gibt es auch etwas Neues zu Hibernate. Beide Java Artikel passen sicher gut zum Thema Java und Performance, was, wie wir aus einer Reihe von Projekten und Projektanfragen wissen, immer häufiger Bestandteil von manch einer Kundenunzufriedenheit ist.

Und nochmal Datenbank: Wir beschreiben den Tungsten Replicator, den man im Umfeld von MySQL auch gut und gerne als Konkurrenz zu Oracle GoldenGate sehen kann. Womit sich der Kreis der Artikel schon fast schließt. Last but not least haben wir einen weiteren Artikel in der Reihe Softwarearchitekturen und ebenfalls aus dem Bereich der Weichware (Softskills / Projektmanagement) unseren Artikel zum Thema Stakeholder Management.

Jetzt hoffe ich darauf, dass sich die Cloud bald in Luft auflöst – natürlich nur die über Mallorca. Ich möchte ja den IBMern, HPIern und wer noch alles etwas im Cloud-Computing anbietet, nicht ihr neues Thema madig machen. Sonst haben die ganzen Cloud Executives nichts mehr zu tun. Wobei der einzige richtige Cloud Executive, den ich kenne, ist Steve Jobs. Er sitzt vermutlich da oben auf seiner Cloud und denkt sich seinen Teil.

Ihr

Wolfgang Kögler

<sup>1</sup> Zu Guttenberg

<sup>2</sup> [http://www.bundestag.de/dokumente/analysen/2010/cloud\\_computing.pdf](http://www.bundestag.de/dokumente/analysen/2010/cloud_computing.pdf)



Java 7: Neuigkeiten aus dem parallelen Universum



Oracle: Der Buffer Pool Advisor

## Datenbanken

### 18..... Oracle: Der Buffer Pool Advisor

Mittlerweile werden die Administratoren von vielen Helfern bei der Verwaltung ihrer Datenbanken unterstützt. Wie nützlich sind solche Helfer? Dieser Artikel stellt den Buffer Pool Advisor vor und gibt eine Einschätzung, wie verlässlich seine Prognosen wirklich sind.

### 21..... IBM DB2: Workload Manager (WLM)

Sie kennen mit Sicherheit die Beschwerden der Anwender, wenn die Systemantwortzeiten zu lang sind. Wir stellen Ihnen mit dem Workload Manager ein Werkzeug für die Steuerung und das Monitoring der Datenbanklast vor.

### 30..... Oracle: APEX-Administration „Light“

Administration muss nicht immer umfangreiche Programmierkenntnisse voraussetzen. Wir zeigen, wie die Entwicklungs- und Laufzeitumgebung einfach administriert werden kann.

### 37..... IBM Informix 11.70 (Teil V): Der Panther hält Wache

Wer darf was? Was darf wer? Dies sind nur zwei Fragen, die ein Administrator beantworten muss, wenn er eine Benutzerverwaltung aufbaut. Informix bietet mit der aktuellen Version attraktive Funktionen, um die Benutzerrechte effektiv zu steuern.

### 40..... Upgrade der Time Zone Files: Über die Kosten des Lichts...

Was hat ein amerikanischer Präsident mit der Sommerzeit zu tun? Diese Frage beantworten wir in dem Artikel zu dem Upgrade der Time Zone Files in Oracle Datenbanken.

### 43..... Oracle GoldenGate – Datenintegration in heterogenen Datenbanklandschaften (Teil II): Jetzt im Angebot: Heterogener Initial Load

Oracle GoldenGate bietet die Möglichkeit, Daten aus einer Oracle Datenbank in andere Datenbanken zu transferieren. Die Migration von und zu anderen heterogenen Datenbanken wird in diesem Artikel vorgestellt.

## Projektmanagement

### 34..... Projektmanagement in der Praxis (Teil IV): Stakeholder Management - aber richtig!

Der Faktor Mensch ist in Projekten nicht zu unterschätzen. Die Stakeholder müssen entsprechend ihren Anforderungen und ihrem Einfluss im Rahmen des Projektmanagements beteiligt werden. Dieser Artikel zeigt Ihnen, wie wichtig ein Stakeholder Management ist und welche Chancen es bietet.

## Softwarearchitektur

### 6..... Softwarearchitektur (Teil IV): Architektursichten

Die Sicht auf Softwarearchitekturen erfolgt aus verschiedenen Blickwinkeln, daher ist eine umfassende Dokumentation zwingend erforderlich. Mit diesem Artikel stellen wir die Unified Modeling Language (UML) vor, die ein standardisiertes und weltweit anerkanntes Mittel zur Dokumentation von Analyse- und Architekturmodellen ist.

26

Hibernate: Batch-Fetching-Strategie für Relationen

## Java/JEE

### 10..... Java 7 - Was bringt die neue Version? (Teil II):

#### Neuigkeiten aus dem parallelen Universum

Java 7 bringt einige Neuerungen im Bereich des Concurrency-Umfeldes. Besondere Aufmerksamkeit bekommt in diesem Artikel das Fork/Join Framework. Es bietet effiziente Algorithmen, um Rechnersysteme mit mehreren Prozessoren bzw. Kernen gut und gleichmäßig auszulasten.

### 26..... Neue Reihe: Hibernate für Fortgeschrittene (Teil I)

#### Die Batch-Fetching-Strategie für Relationen richtig anwenden

Optimierungsmaßnahmen können sowohl positive als auch negative Auswirkungen auf die Performance haben. Dieser Artikel erklärt die Wirkungsweise der sogenannten Batch-Fetching-Strategie von Hibernate und was bei der Anwendung zu berücksichtigen ist.

## OpenSource

### 14..... OpenSource Tungsten Replicator: Replication for Free

Es gibt eine Vielzahl an Replikationswerkzeugen für Datenbanken. In diesem Artikel stellen wir den Tungsten Replicator vor, der eine gute OpenSource-Alternative zu Oracle GoldenGate ist.

## Aktuell | Standards

### 17..... Larry Ratlos

### 24..... Seminarübersicht: März 2012 bis Juni 2012

### 47..... Rückblick auf den ersten Treffpunkt RheinMainIT

40

Oracle: Über die Kosten des Lichts...

## Impressum

- Herausgeber:** ORDIX AG Aktiengesellschaft für Softwareentwicklung, Beratung, Schulung und Systemintegration, Paderborn
- Redaktion:** Jens Pothmann, Evelyn Ernst
- V.i.S.d.P.:** Benedikt Georgi, Wolfgang Kögler
- Anschrift der Redaktion:** ORDIX AG | Westernmauer 12 - 16 | 33098 Paderborn  
Tel.: 05251 1063-0 | Fax: 0180 1673490
- Gestaltung/Layout:** Jens Pothmann
- Auflage:** 9.200 Exemplare
- Druck:** Druckerei Bösmann, Detmold
- Bildnachweis:** © NASA/courtesy of nasaimages.org  
© sxc.hu | Carpathian Mountains | geo cristian  
© istockphoto.com | Magical world of reading | Lobke Peers  
© istockphoto.com | Video Wall | Erik Reis  
© iconfinder.com | search-icon  
© shutterstock.com | Evolution of lighting light | Vasyil Helevachuk  
© istockphoto.com | stack of books in notebook | pagadesign  
© sxc.hu | Boxes | satty4u  
© istockphoto.com | Hmm... | sdominick
- Autoren:** Dr. Hubert Austermeier, Evelyn Ernst, Andreas Flügge, Martin Hoermann, Carsten Hummel, Benedikt Georgi, Klaus Günther, Wolfgang Kögler, Sven Loer, Andreas Maßmann, Stefan Mertens, Thomas Schwarz, Werner Wellmann
- Copyright:** Die ORDIX news erscheint vierteljährlich. Alle Eigentums- und Nachdruckrechte, auch die der Übersetzung, der Vervielfältigung der Artikel oder von Teilen daraus, sind nur mit schriftlicher Zustimmung der ORDIX AG gestattet. ORDIX® ist ein eingetragenes Markenzeichen der ORDIX AG.
- Warenzeichen:** Einige der aufgeführten Bezeichnungen sind eingetragene Warenzeichen ihrer jeweiligen Inhaber. ORDIX® ist eine registrierte Marke der ORDIX AG.
- Haftung:** Eine Haftung für die Richtigkeit der Veröffentlichungen kann trotz sorgfältiger Prüfung durch die Redaktion vom Herausgeber nicht übernommen werden.
- Sie können die Zusendung der ORDIX news jederzeit ohne Angabe von Gründen schriftlich (z.B. Brief, Fax, E-Mail) abbestellen.

# Architektursichten

Die Dokumentation von Softwarearchitekturen macht einen großen Teil der Tätigkeiten eines Softwarearchitekten aus. Art und Umfang der Dokumentation sind dabei ebenso wichtig, wie die Wahl der richtigen Werkzeuge. Obwohl längst nicht völlig unumstritten, ist und bleibt die Unified Modeling Language (UML) eines der wichtigsten Standardmittel des Architekten. In diesem Artikel betrachten wir, wie UML zur angemessenen Dokumentation von Softwarearchitekturen verwendet werden kann.

## Dokumentation

Die Dokumentation von Softwarearchitekturen dient vornehmlich der Kommunikation zwischen den unterschiedlichen Stakeholdern und dem Architekten. Er muss also darauf achten, dass er gegenüber allen Beteiligten eine geeignete Form und eine verständliche Darstellung findet, da die Interessen von sehr unterschiedlichen Aspekten und Sichtweisen geprägt sind.

So sind beispielsweise betriebliche Belange, wie Installationsprozeduren oder Wartungsvorgänge eines Systems für das Management von geringer Bedeutung, während dies für die Administratoren zum Tagesgeschäft gehört.

## Sichten

Die Vielschichtigkeit und Komplexität eines Softwaresystems ist in der Regel nicht durch eine einzelne Darstellung zu dokumentieren. Aus diesem Grund bedient man sich spezieller Formen, die sich auf bestimmte Aspekte des Gesamtsystems konzentrieren und so die Komplexität der Darstellung reduzieren. Die Idee dieser Sichten ist noch relativ neu. Sie geht unter anderem auf Philipp Kruchten zurück, der sie 1995 bei der Entwicklung des Rational Unified Process (RUP) vorstellte. Er prägte den Begriff des „4+1-Sichtenmodells der Softwarearchitektur“ (siehe Abbildung 1).

Die vier Sichten repräsentieren dabei die Blickwinkel unterschiedlicher Stakeholder, wie Endanwender, Entwickler, Administratoren oder Manager. Sie beziehen sich auf die Szenarien (Use Cases), die die eigentliche Funktionalität des Systems beschreiben.

Bis heute haben sich die vier Sichten etabliert. Im deutschsprachigen Raum haben sich dafür die Begriffe Systemkontext oder Kontextabgrenzung (Logical view), Bausteinsicht (Development view), Laufzeit-sicht (Process view) und Verteilungssicht (Physical view) etabliert.

## Unified Modeling Language (UML)

Wie in vielen anderen Ingenieursdisziplinen gibt es auch in der Informatik Standards zur Kommunikation von speziellen Sachverhalten. Seit 1997 ist die Unified Modeling Language (UML) von der Object Management Group (OMG) standardisiert und weltweit als Mittel zur Dokumentation von Analyse- und

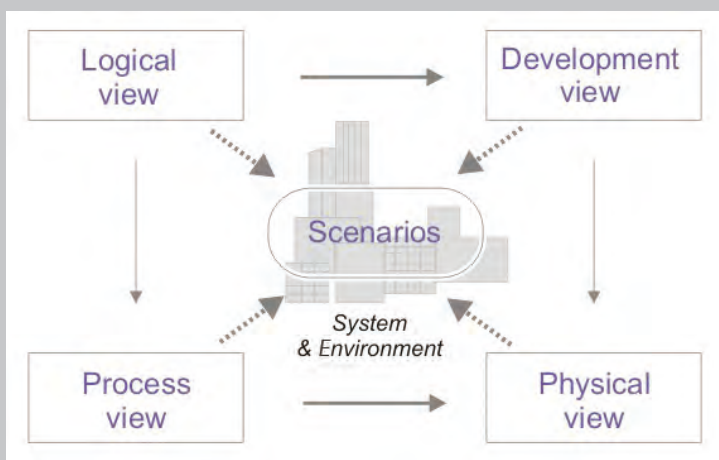


Abb. 1: „4+1-Sichtenmodell der Softwarearchitektur“ (Quelle: Wikipedia).

Architekturmodellen anerkannt. Obwohl sie wegen ihrer Komplexität nicht unumstritten ist, steht sie als Standard zurzeit konkurrenzlos dar.

Es ist also naheliegend, die UML auch als Dokumentationsmittel für die oben genannten Sichten zu verwenden. Für jede Sicht stellt die UML eine oder mehrere geeignete Diagrammart zu Verfügung. Dabei ist natürlich zu berücksichtigen, dass für die einzelnen Sichten je nach Umfang, Teilaspekte der Architektur über mehrere Diagramme verteilt dargestellt werden können.

### Systemkontext (Logical view)

Der Systemkontext hat eine etwas übergeordnete Stellung. Er zeigt, wie das System in seiner Umgebung eingebettet ist. Als Blackbox wird es aus der Vogelperspektive betrachtet und ermöglicht so die Abgrenzung zu umgebenden Systemen zu finden. Dadurch können Schnittstellen und Infrastrukturkomponenten erkannt und Nutzer des Systems identifiziert werden.

Da der Systemkontext häufig zur Kommunikation mit den nicht technischen Stakeholdern verwendet wird, stellt er das System auf einer relativ hohen Abstraktionsstufe dar (siehe Abbildung 1 aus dem Teil III dieser Reihe [3]).

Zu diesem Zweck hat sich das Komponentendiagramm aus der UML bewährt. Denkbar sind aber auch durchaus andere Darstellungen wie zum Beispiel das Kommunikationsdiagramm oder das Klassendiagramm.

### Bausteinsicht (Development view)

Im Gegensatz zum Systemkontext ist die Bausteinsicht eine wesentlich speziellere Darstellung des Systems. Sie dient vornehmlich den Entwicklern und beschreibt, aus welchen Architekturbausteinen, also Subsystemen, Komponenten und Schnittstellen das System besteht. Die Bausteinsicht befasst sich dabei nur mit dem statischen Aufbau des Systems, sie berücksichtigt keinerlei Informationen bezüglich der internen Verarbeitungsprozesse.

In der Regel verwendet man zur Darstellung einen Top-Down-Ansatz ausgehend von der Blackbox des Gesamtsystems aus dem Systemkontext. Diese wird über mehrere Verfeinerungsstufen in kleinere Bausteine zerlegt, bis alle architekturrelevanten Details

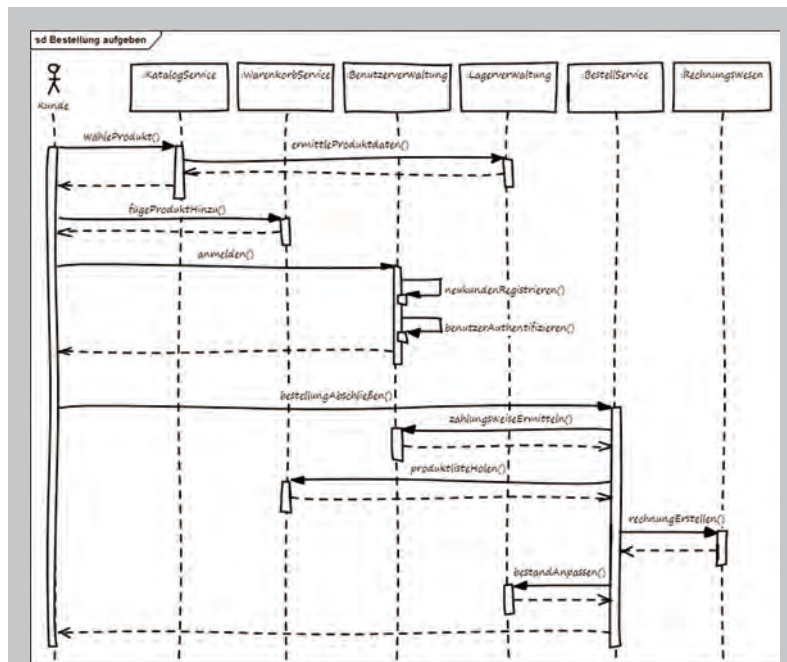


Abb. 2: Sequenzdiagramm eines Teilaspekts der Laufzeitsicht.

herausgearbeitet sind (siehe Abbildung 3 aus dem Teil III dieser Reihe [3]).

Bausteine bilden einzelne Funktionalitäten ab. Die Bausteinsicht macht Strukturen und Abhängigkeiten zwischen den einzelnen Bausteinen durch die Schnittstellen sichtbar. Zur Darstellung verwendet die Bausteinsicht Klassensymbole, Komponenten und Pakete aus der UML. Dabei handelt es sich um Abstraktionen, die im detaillierten Architekturentwurf weiter verfeinert werden können.

So bietet sich folgerichtig für die Darstellung der Bausteinsicht die Verwendung derjenigen UML-Diagramme an, die sich vornehmlich mit der Abbildung der Strukturierung des Gesamtsystems befassen. Dies sind neben dem Komponentendiagramm auch das Klassen- und Paketdiagramm.

### Laufzeitsicht (Process view)

Die Laufzeitsicht dokumentiert das System so, wie es sich zur Laufzeit darstellt. Sie beantwortet Fragen zur Existenz und zum Verhalten von Instanzen bestimmter Bausteine zur Laufzeit, deren Zusammenspiel und Konfiguration.

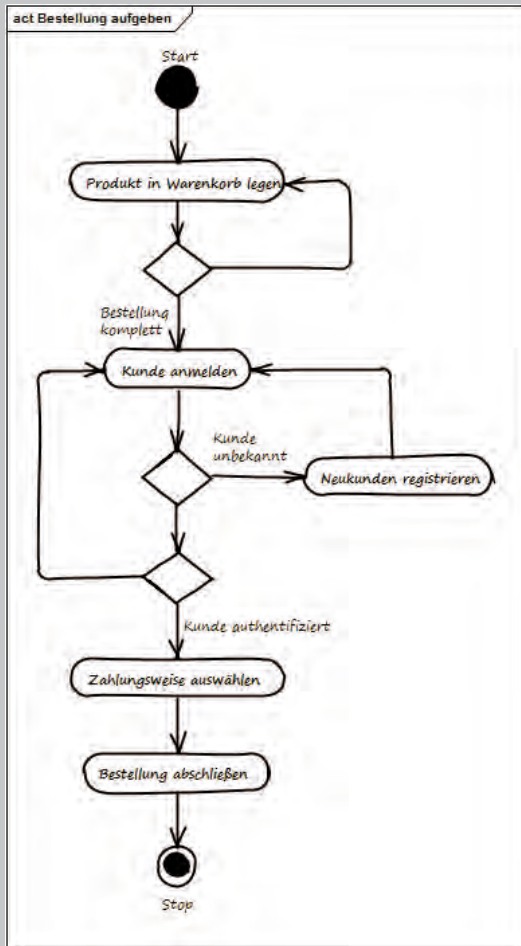


Abb. 3: Aktivitätendiagramm.

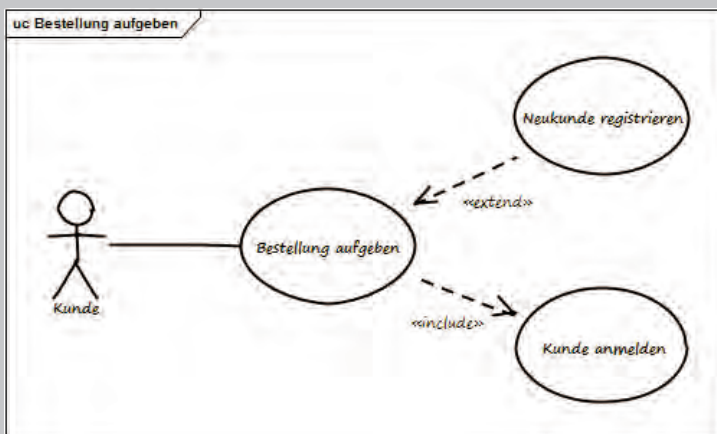


Abb. 4: Use Case.

Anders als bei der Bausteinsicht werden hier also vorwiegend die dynamischen Aspekte eines Systems beschrieben. Es interessieren dementsprechend diejenigen Bausteine, von denen zur Laufzeit auch tatsächlich Exemplare erzeugt werden. Sie werden als Einheit ausgeführt und aufgerufen, ihr Verhalten implementiert spezielle Funktionalitäten.

Das Zusammenspiel dieser Bausteine manifestiert sich in deren Beziehungen untereinander. Dabei kann es sich sowohl um Kontroll- als auch um Datenflüsse handeln.

Zur Notation der Laufzeitsicht bietet die UML mit dem Aktivitäts- und dem Sequenzdiagramm zwei sehr gut geeignete Darstellungsformen an. Zur Erstellung der Laufzeitsicht muss die Dynamik der statischen Bausteine aus der Bausteinsicht beschrieben werden. Die Use Cases sind dabei eine sehr nützliche Hilfe, denn sie beschreiben im Wesentlichen das Verhalten des Gesamtsystems.

Das (stark vereinfachte) Sequenzdiagramm aus Abbildung 2 verdeutlicht sehr gut die Abhängigkeiten der einzelnen Komponenten durch die Darstellung der gegenseitigen Aufrufe.

Die Erstellung solcher Sequenzdiagramme kann durch die Betrachtung der dazugehörigen Use Cases und Aktivitäten erleichtert werden, wie dies die Diagramme in den Abbildungen 3 und 4 verdeutlichen.

### Verteilungssicht (Physical view)

Mit der vierten noch fehlenden Sicht, der Verteilungssicht, wird die Ablaufumgebung des Systems beschrieben. Hierbei handelt es sich um eine Art Landkarte, in der alle Hardwarekomponenten und die dazugehörigen Protokolle eingezeichnet werden.

Oft auch als Infrastruktursicht bezeichnet, stellt die Verteilungssicht also physikalische Geräte wie Server, Prozessoren, Netzwerke, Router, Firewalls, Speicher oder Datenbanken dar. Diese werden oft um Leistungsdaten, Kapazitäten, Mengengerüste, Geschwindigkeiten oder Angaben zu Betriebssystemen und Laufzeitumgebungen ergänzt.

Die UML stellt für die Darstellung der Verteilungssicht das Deployment-Diagramm zur Verfügung (siehe Abbildung 5). Der Knoten wird für beliebige technische Elemente verwendet. Für die Laufzeitelemente können auch Komponenten- oder Paketsymbole verwendet werden.



Weitere Sichten?

Die vorgestellten Sichten sind ein wichtiger Bestandteil der Softwarearchitekturdokumentation. Dennoch stellen sie nur einen Teil der Gesamtdokumentation dar. In einem späteren Artikel werden wir eine weitere Methode vorstellen, mit der Architekturen systematisch in einem sinnvollen Umfang dokumentiert werden können. Zudem zeigen wir auf, welche weiteren Aspekte es noch zu berücksichtigen gilt.

Grundsätzlich sollte jedoch auch die Dokumentation von Softwarearchitekturen einem pragmatischen Ansatz folgen. Das bedeutet, dass wenn bestimmte Informationen zum Verständnis der Architektur benötigt werden oder überflüssig sind, diese entsprechend hinzugefügt oder weggelassen werden können.

Werden möglicherweise Informationen zu Datenhaltung oder Datenstrukturen benötigt, so sollte unbedingt auch eine entsprechende Sicht, beispielsweise in Form von Entity-Relationship-Diagrammen, zur Dokumentation hinzugefügt werden.

Andere Formen

UML stellt ein mächtiges und sinnvolles Instrumentarium zur Dokumentation von Softwarearchitekturen bereit. Im Sinne eines pragmatischen Ansatzes sollte der Architekt aber auch andere Formen in Betracht ziehen. So reichen in manchen Fällen auch einfache Zeichnungen, Tabellen, Checklisten, Aufzählungen oder andere textuelle Darstellungen aus, deren Verwendung durchaus legitim ist.

Fazit

Das erläuterte Sichtenmodell stellt einen geeigneten Ansatz dar, um die zentralen Aspekte einer Softwarearchitektur zu dokumentieren. UML hält, alle wichtigen Mittel dafür bereit. Allerdings fällt auch auf, dass dafür nur ein Bruchteil des mächtigen Instrumentariums tatsächlich benötigt wird.

Ein Softwarearchitekt muss also kein spezialisierter UML-Experte sein. Er ist im Gegenteil gut beraten, die Dokumentation immer den Anforderungen seiner Zielgruppe anzupassen und allzu technische oder spezialisierte Darstellungsformen zu vermeiden. Eine einfache Zeichnung, Tabelle oder simpler Text ist zur Dokumentation manchmal mindestens genauso gut geeignet, wie ein detailliertes UML-Diagramm.

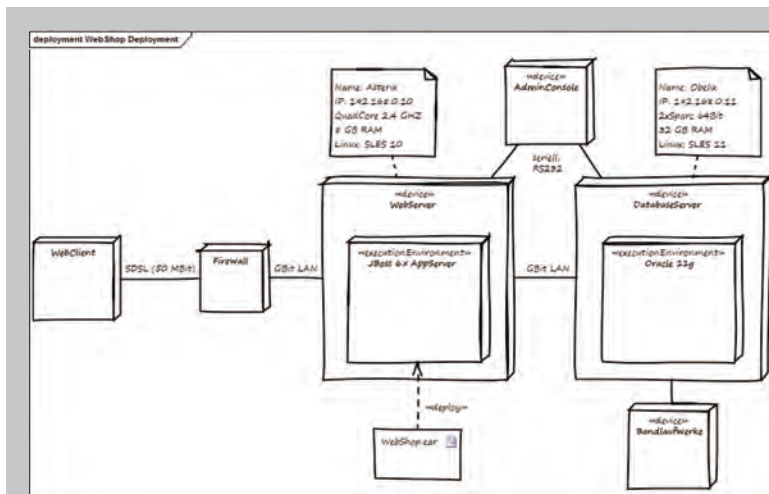


Abb. 5: Deployment-Diagramm für die Verteilungssicht.

Links

- ▶ [1] ORDIX news Artikel 1/2011 „Softwarearchitektur (Teil I) - Die Grundsteinlegung“: [http://www.ordix.de/ORDIXNews/1\\_2011/softwarearchitektur.html](http://www.ordix.de/ORDIXNews/1_2011/softwarearchitektur.html)
- ▶ [2] ORDIX news Artikel 2/2011 „Softwarearchitektur (Teil II) - Der Softwarearchitekt“: [http://www.ordix.de/ORDIXNews/2\\_2011/softwarearchitektur.html](http://www.ordix.de/ORDIXNews/2_2011/softwarearchitektur.html)
- ▶ [3] ORDIX news Artikel 4/2011 „Softwarearchitektur (Teil III) - Die Architekturentwicklung“: [http://www.ordix.de/ORDIXNews/3\\_2011/architekturentwicklung.html](http://www.ordix.de/ORDIXNews/3_2011/architekturentwicklung.html)
- ▶ [3] ORDIX Seminarempfehlung „Softwarearchitekturen“: <http://training.ordix.de/siteengine/action/load/kategorie/Java-JEE/nr/1116/index.html>



Andreas Flügge (info@ordix.de).



Java 7 – Was bringt die neue Version? (Teil II)

## Neuigkeiten aus dem parallelen Universum

Lange hat es gedauert, bis die Version 7 von Java endgültig zur Verfügung stand. Die wesentlichen Neuerungen stellen wir in dieser Artikelreihe vor, die im ersten Teil bereits die Erweiterungen im Java-Sprachkern zum Thema hatte [1]. Substantielle Verbesserungen finden wir auch im Bereich der Parallelverarbeitung oder des Multithreading, die als Neuerungen in dem Concurrency Framework zusammengefasst sind und den Gegenstand dieses Artikels bilden.

### Einzelkämpfer vs. Teamwork

Die Kapazitäten von modernen Rechnern wachsen weiter rasant an und die Ausstattung mit mehreren Prozessoren bzw. mehreren Kernen pro Prozessor gehört inzwischen längst zum Standard. War es bis vor Jahren noch üblich, die Rechnerleistung hauptsächlich durch höhere Taktgeschwindigkeiten von Prozessoren zu steigern, hat sich dieser Trend aufgrund rein physikalischer Gesetzmäßigkeiten deutlich verändert. Um weiterhin das Moore'sche Gesetz [2]

einhalten zu können, demzufolge sich die durchschnittliche Rechnerleistung alle zwei Jahre verdoppelt, treiben die Computerhersteller in zunehmendem Maße den Einbau von Mehrkern-CPU's voran.

Um auch weiterhin die enormen Kapazitäten von modernen Rechnersystemen in vollem Umfang nutzen zu können, braucht es neuartige Ansätze, was unabdingbar Programmiersprachen mit den Möglichkeiten der Parallelverarbeitung erfordert. Das Ausführen von parallelen Ausführungspfaden – das

Multithreading – war bei Java von Beginn an (Java 1.0, 1995) möglich, wenn auch anfangs eher rudimentär. Mit der Weiterentwicklung der Java-Plattform wurde auch dieser Teil sukzessive modernisiert, wobei allerdings erst mit Java 5 ein eigenes Concurrency Framework Einzug hielt [3].

Dieses Framework setzt sich aus einer Vielzahl von neuen Datenstrukturen (**BlockingQueue**, **ConcurrentMap**, ...) zusammen, die sich alle im neuen Paket `java.util.concurrent` befinden und dazu passende, effiziente Mechanismen zur Synchronisation und zum Umgang mit Konkurrenzsituationen beinhalten.

### Neuerungen, die es in sich haben

In Java 7 hat es nun mehrere Erweiterungen in diesem Bereich gegeben, von denen das Fork/Join Framework klar heraussticht. Es verfolgt den Ansatz, einen hochwertigen Parallelisierungsalgorithmus („Teile und herrsche“; auch bekannt als Divide and conquer) als Java-Konstrukt anzulegen [4].

Aber auch die **TransferQueue** bringt neue Möglichkeiten für geregelte Übergabeszenarien, genauso wie die kleine aber feine Klasse **ThreadRandomLocal**, die eine zuverlässige und effiziente Erzeugung von Zufallszahlen in Multithread-Anwendungen unterstützt. Diese drei Neuerungen wollen wir im Folgenden etwas näher beleuchten.

### Fork/Join Framework

Das neue Fork/Join Framework in Java SE 7 ist eine Implementierung des `ExecutorService` Interface aus dem Concurrency Framework und kann besonders bei der effizienten Auslastung von Mehrprozessorsystemen helfen.

Es wurde konzipiert für Lösungsansätze von Problemen mit großem Umfang, die sich aber dennoch gut aufteilen lassen [5]. Das Ziel ist es, die verfügbare Rechnerkapazität unter größtmöglicher Vermeidung von Prozessorwartezeiten gut und effizient zu nutzen.

Wie bei `ExecutorService`-Implementierungen üblich, arbeitet auch das Fork/Join Framework mit Aufgaben, die Worker Threads in einem Thread Pool zugeordnet werden. Eine Besonderheit stellt der interne Algorithmus dar, nach dem die Zuteilung der Threads mit Rechenkapazität erfolgt. Es kommt ein „work-stealing“-Algorithmus zum Einsatz, der Threads effizient mit Arbeit versorgt und den wir später etwas genauer betrachten.

```

Ergebnis loese(Problem problem) {
    if (problem.groesse < SEQUENTIELL_SCHWELLE)
        return problem.loeseSequentiell();
    else {
        Ergebnis links, rechts;
        INVOKE-IN-PARALLEL {
            links = loese(problem.extrahiereLinkeHaelfte());
            rechts = loese(problem.extrahiereRechteHaelfte());
        }
        return kombiniere(links, rechts);
    }
}

```

Abb. 1: Pseudo-Code für das „Teile und herrsche“-Prinzip.

```

class MaxSolver extends RecursiveAction {
    private final MaxProblem problem;
    int result;
    protected void compute() {
        if (problem.size < THRESHOLD)
            result = problem.solveSequentiell();
        else {
            int m = problem.size / 2;
            MaxSolver left, right;
            left = new MaxSolver(problem.subproblem(0, m));
            right = new MaxSolver(problem.subproblem(m,
                problem.size));
            forkJoin(left, right);
            result = Math.max(left.result, right.result);
        }
    }
}

ForkJoinExecutor pool = new ForkJoinPool(nThreads);
MaxSolver solver = new MaxSolver(problem);
pool.invoke(solver);

```

Abb. 2: Java Code für eine Umsetzung der Maximumsuche mit den neuen Mitteln des Concurrency Framework aus Java 7.

### Teile und herrsche

Wenn Probleme zu groß werden, muss man sie auf mehrere Schultern verteilen. Diese Lektion lernt man im täglichen Leben und sie lässt sich auch auf die Informatik übertragen. Ein seit langem bekannter und dabei sehr eleganter Ansatz stellt das „Teile und herrsche“-Prinzip dar, welches nach einem rekursiven Prinzip funktioniert (siehe Abbildung 1).

Dahinter steckt die Idee, dass Problemgrößen unterhalb einer bestimmten Schwelle gut in Eigenregie zu lösen sind, d.h. sequentiell. Andernfalls ist das Problem möglichst gleichmäßig aufzuteilen und an unabhängig operierende Arbeiter zu delegieren. Dieser Ansatz erweist sich deshalb als so praktikabel und elegant, weil er nur eine numerische Größe braucht – nämlich die Schwelle – und nicht weitere zahlenmäßige Angaben, wie z.B. eine feste Anzahl von Workern, die die parallele Arbeit ausführen. Das zentrale Konstrukt in diesem Ansatz verbirgt sich im **INVOKE-IN-PARALLEL**, das gewährleisten muss, die nachfolgenden Aufgaben (Tasks) parallel auszuführen.

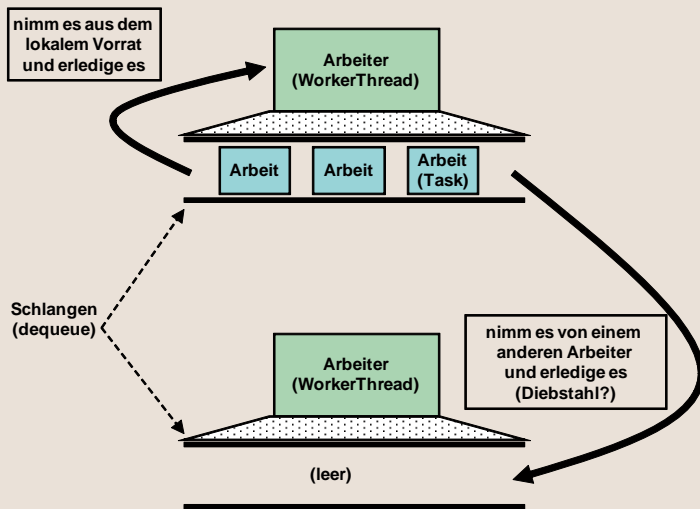


Abb. 3: Prinzipielle Funktionsweise des „work stealing“-Verfahrens. Sobald der Arbeitsvorrat eines Arbeiters leer ist, wird der eines anderen „geplündert“.

Genau an dem Punkt greifen die neuen Klassen und Interfaces aus Java 7:

- ForkJoinPool
- ForkJoinTask
- RecursiveAction
- RecursiveTask<V>
- u.a.m.

Ein konkretes Beispiel, anhand dessen sich das Prinzip gut nachvollziehen lässt, stellt der nachfolgende Algorithmus zur Maximumsuche in einem großen Array dar (siehe Abbildung 2). Die lokale Variable `problem` hat einen speziellen, einem Array ähnlichen Typ (Quellcode stammt aus [6]). Dieser Typ erlaubt mit der Methode `subproblem()` die Aufteilung in kleinere Teilarrays.

Der Algorithmus hat viele Aspekte, deren genauere Betrachtung sich lohnen würde, allerdings wollen wir uns auf den Aufruf von `forkJoin(left, right)` konzentrieren. Dieser beinhaltet einen rekursiven Aufruf, obwohl man es nicht sofort sieht. Denn wie der Name schon andeutet, vollführt die Methode zwei elementare Aufgaben. Zunächst „fork“ sie alle Objekte, die sie bekommt, was nichts anderes heißt, als dass sie deren Methode `compute()` ausführt (aha – da ist sie ja, unsere Rekursion).

Aber das geschieht keineswegs sequentiell, denn im Hintergrund haben wir das Concurrency Framework, welches dafür sorgt, dass die `compute()`-Methoden

von Worker Threads eines Pools in paralleler Manier ausgeführt werden. Dann folgt noch der „join“-Schritt, der hauptsächlich aus Warten besteht und die Zusammenführung der Teilergebnisse verantwortet.

### Diebstahl soll tugendhaft sein – wo kommen wir denn dahin?

Hinter der Implementierung des Fork/Join Framework steckt das Prinzip des „work stealing“, was auf den ersten Blick etwas merkwürdig anmutet. Es beruht auf der Idee, dass Worker Threads, die ihre Arbeit erledigen, diese Arbeit in sogenannten „Dequeues“ (double ended queues) vorhalten und sich aus diesen auch mit frischer Arbeit versorgen. Nun kommen die doppelten Enden ins Spiel. Die Eigenversorgung geschieht vom Kopf her, d.h. Arbeit aus der eigenen dequeue wird von vorn heraus genommen.

Hat ein Worker Thread aber seinen Vorrat aufgebraucht und eine leere Dequeue, dann schaut er sich nach einer anderen, gefüllten um und bedient sich dort aber am anderen Ende, was einer Entnahme von hinten entspricht (siehe Abbildung 3). Das ist ganz klar Diebstahl, der in diesem Kontext allerdings nicht nur erlaubt sondern auch höchst willkommen ist, zeigt doch diese effiziente Strategie viele positive Aspekte.

### Die TransferQueue bietet eine geregelte Übergabe

In einem vorherigen Artikel [3] wurde schon die `BlockingQueue` aus Java 5 vorgestellt, mit deren Hilfe sich das bekannte „Producer and Consumer“-Entwurfsmuster umsetzen lässt. Dabei gelangt auf Produzentenseite ein Objekt per Methode `put(...)` in die Schlange. Konsumiert wird auf der anderen Seite über die Methode `take()` und so erhält man eine wunderschön symmetrische Produktionslandschaft.

Allerdings zeichnet sich eine derart aufgebaute Kommunikationskette dadurch aus, dass ihre Partner (Producer and Consumer) nahezu völlig voneinander entkoppelt sind, d.h. sie haben keine Kenntnis und Kontrolle darüber, wann der jeweilige Partner etwas hinein getan bzw. heraus genommen hat. Diese Entkoppelung ist zwar häufig gewollt, nichtsdestotrotz braucht man hin und wieder aber auch eine etwas genauere Kontrolle. Da kann das neue Konstrukt `TransferQueue` aus Java 7 weiter helfen, welches mit den zusätzlichen Methoden `transfer(...)` und `tryTransfer(...)` die Kontrolle zumindest auf der Produzentenseite erlaubt.

Die Namen sind sehr sinnstiftend, denn bei `transfer(...)` wartet der Produzent so lange, bis das Objekt konsumiert, also aus der Queue entnommen wurde. Bei `tryTransfer(...)` findet lediglich der Versuch statt, ein Objekt zur Übergabe bereitzustellen. Gibt es keinen Abnehmer, wird im Programmablauf fortgefahren. Diese Methode kann auch mit einem Timeout `x` aufgerufen werden, was eine gewisse Wartezeit `x` erzeugt. Das heißt, ein Konsument hat `x` Sekunden Zeit, das Objekt zu entnehmen.

Es gibt eine Reihe weiterer, hilfreicher Methoden wie die Abfrage `hasWaitingConsumer()` oder eine Methode, um an die Anzahl von wartenden Konsumenten zu kommen `getWaitingConsumerCount()`.

### Schnelle Produktion von Zufallszahlen

Zufallszahlen und deren Erzeugung spielen u.a. in der Kryptographie und in computergestützten Simulationen eine große Rolle. In Java kann dafür `Math.random()` genutzt werden. Diese Methode produziert sogenannte Pseudozufallszahlen, deren Werte sich jeweils auf einen Vorgängerwert beziehen. Diese Vorgängerwerte müssen gespeichert werden, was in Multithread-Anwendungen zu Konkurrenzsituationen führen kann, wenn mehrere Threads gleichzeitig speichern wollen. Dieser Wert ist geschützt und zu einer Zeit nur von einem Thread manipulierbar. Das aber kann in hochparallelen Systemen, wie man sie in Simulationsumgebungen häufig antrifft, zu empfindlichen Engpässen führen.

Hier rückt die neue Klasse `ThreadRandomLocal` in das Blickfeld, welche in `java.util.concurrent` lokalisiert ist [7]. Die Wahl fällt auf die Methode `ThreadRandomLocal.current().nextX(...)` wobei das `X` für `Int`, `Long`,... steht. Sie verwendet nur Thread-lokale Speicherorte für die Vorgängerwerte und dies führt dazu, dass konkurrierende Zugriffe unterschiedlicher Threads prinzipiell ausgeschlossen sind. Die Klasse `ThreadRandomLocal` eignet sich daher besonders in Kontexten, wie sie z.B. bei `ForkJoinTasks` anzutreffen sind, die üblicherweise in großen Thread Pools abgearbeitet werden.

### Fazit

Man sieht sehr deutlich, dass es Oracle sehr ernst ist mit der konsequenten Weiterentwicklung der Java-Plattform, denn die Neuerungen im Concurrency-Bereich gehen tief unter die Oberfläche und stellen sehr effiziente Algorithmen und angepasste Datenstrukturen für die Parallelverarbeitung zur Verfügung. Wir haben uns stellvertretend das Fork/Join Framework

## Glossar

### Rekursion

Die Rekursion ist u.a. ein Prinzip in der Informatik, das „Selbstbezug“ ausdrückt. Eine Methode wird z.B. als rekursiv bezeichnet, wenn sie sich selbst aufruft. Dabei ist besondere Vorsicht geboten, weil Abbruchkriterien sicherstellen müssen, dass keine endlosen Aufrufketten entstehen können (was zu einem Stack Overflow führen würde).

## Links

- ▶ [1] ORDIX news Artikel 4/2011 „Project Coin“:  
[http://www.ordix.de/ORDIXNews/4\\_2011/java7.html](http://www.ordix.de/ORDIXNews/4_2011/java7.html)
- ▶ [2] [http://de.wikipedia.org/wiki/Mooresches\\_Gesetz](http://de.wikipedia.org/wiki/Mooresches_Gesetz)
- ▶ [3] ORDIX news Artikel 4/2010 „Concurrency - Lästige Konkurrenz (I)“:  
[http://www.ordix.de/ORDIXNews/4\\_2010/concurrency.html](http://www.ordix.de/ORDIXNews/4_2010/concurrency.html)
- ▶ [4] [http://de.wikipedia.org/wiki/Teile\\_und\\_herrsche\\_%28Informatik%29](http://de.wikipedia.org/wiki/Teile_und_herrsche_%28Informatik%29)
- ▶ [5] Java 7 Tutorial zum Thema:  
<http://docs.oracle.com/javase/tutorial/essential/concurrency/forkjoin.html>
- ▶ [6] Brian Goetz, Sun Microsystems, 2008 JavaOneSM Conference:  
<http://java.sun.com/javaone>
- ▶ [7] Definition der Klasse `ThreadLocalRandom`:  
<http://docs.oracle.com/javase/7/docs/api/java/util/concurrent/ThreadLocalRandom.html>

**Bild:** © NASA/courtesy of nasaimages.org  
© sxc.hu | Carpathian Mountains | geo cristian

etwas intensiver angeschaut, das eine substantielle Verbesserung darstellt und auch in dem ORDIX Seminar „Java Programmierung Aufbau“ behandelt und vermittelt wird.



*Dr. Hubert Austermeier  
(info@ordix.de).*

# Replication for Free

Das OpenSource-Werkzeug „Tungsten Replicator“ ist derzeit in der Version 2.0.5 verfügbar. Der Tungsten Replicator ist eine Data Replication Engine für MySQL. Außer MySQL werden aber auch andere Datenbanksysteme unterstützt. Dadurch wird eine Vielfalt an Replikationsmöglichkeiten auch zwischen verschiedenen Datenbanksystemen geschaffen. Zudem ermöglicht es den Datenbankadministratoren die einfache Erstellung von Datenbankkopien.

## Voraussetzungen

Die Software des Tungsten Replicator ist auf der Homepage [1] zu finden. Dort befinden sich sowohl allgemeine Informationen, ein Download-Bereich als auch diverse Tutorials für die Benutzung des Werkzeuges [2], [3]. Neben der OpenSource-Variante gibt es auch eine Enterprise-Variante des Produktes. Diese ist aber in jedem Fall kostenpflichtig, bietet dafür jedoch Support der Firma Continuent [4].

Für die OpenSource-Variante bestehen einige Grundvoraussetzungen, die für eine funktionierende Installation geschaffen werden müssen. Zunächst werden folgende Komponenten zwingend vom Tungsten Replicator gefordert:

- MySQL (ab Version 4.1 bis Version 5.5)
- Java (ab Version 1.6)
- Ruby (ab Version 1.8.5)

Betriebssystemseitig werden alle gängigen Unix-Derivate unterstützt. Sollte eine andere Distribution

als Redhat oder Debian verwendet werden, so muss die Validierung des Betriebssystems ausgeschaltet werden. Ansonsten kann es zu Fehlern kommen. Folgende Datenbanken werden neben MySQL von dem Tungsten Replicator unterstützt:

- PostgreSQL
- Oracle
- MariaDB
- XtraDB
- MongoDB

Im Weiteren wird ein genauer Blick auf den Tungsten Replicator und seine Verwendung mit MySQL geworfen.

## Master und Slave

Seit der MySQL-Version 3.23.33 verfügt die OpenSource-Datenbank über eine Möglichkeit zur Replikation. Um die Replikation zu aktivieren, müssen lediglich zwei Konfigurationsparameter geändert werden. Diese Änderung wird von dem Datenbankadministrator betriebssystemabhängig in der Datei **my.cnf** (Unix, Linux) oder in der Datei **my.ini** (Windows) gepflegt.

- **server-id = 1**
- **log-bin=mysql-bin**

Der Parameter **server-id** dient zur eindeutigen Identifizierung des MySQL Server im Tungsten Replicator. Der Tungsten Replicator bedient sich ebenfalls dieser Parameter, um die an einer Replikation beteiligten Server eindeutig zu identifizieren und die entsprechenden Transaktionsinformationen auf dem Master auszuwerten.

```
HOME=/opt/tungsten/replication355
./tungsten-installer \
--master-slave \
--cluster-hosts=tungsten1.ordix.de,tungsten2.ordix.de \
--master-host=tungsten1.ordix.de \
--datasource-user=tungsten \
--home-directory=$HOME \
--thl-directory=$HOME/thl \
--relay-directory=$HOME/relay \
--service-name=replication355 \
--start-and-report \
--no-validation
```

Abb. 1: Installation eines Master-Slave-Cluster.

Wichtig ist, dass die Binary Logs für den angelegten Systembenutzer des Tungsten Replicator nutzbar sind. Daher muss auf die Berechtigungen des Binary Log ein besonderes Augenmerk gelegt werden, da es ansonsten zu Fehlern bei der Replikation kommen kann.

### Installation

Das heruntergeladene Archiv des Tungsten Replicator muss in ein dafür vorgesehenes Verzeichnis entpackt werden. In unserem Beispiel wurde dafür der Pfad `/opt/tungsten` verwendet. In dem Verzeichnis `tools` befinden sich die empfohlenen Installations-Skripte:

- `./configure`
- `./configure-service`
- `./tungsten-installer`

Es wird empfohlen ausschließlich mit den Skripten `./tungsten-installer` und `./configure-service` zu arbeiten. Ein Beispiel für eine Tungsten-Installation zeigt Abbildung 1. Das angegebene Zielverzeichnis wird angelegt und der definierte Service gestartet. Nach dieser Initialisierung können weitere Services mit dem Skript `./configure-service` angelegt oder auch gelöscht werden.

Der Tungsten Replicator benutzt bei keiner spezifischen Angabe Standardwerte für die MySQL-Datenbank (Port 3306, Verzeichnis `/var/lib/mysql`). Aber auch diese Werte können modifiziert und bei der Installation mit angegeben werden. Eine Übersicht über sämtliche Befehle kann mittels der Option `--help` generiert werden.

### Replikationsmechanismen

Generell können bei der Installation eines Service die folgenden Replikationsmechanismen genutzt werden:

- `direct`
- `master-slave`

Der Replikationsmechanismus `direct` ist eine schnelle Methode, um eine Replikation von einem Master auf einen Slave zu erstellen. Dabei werden die Binary Logs lokal auf dem Slave abgelegt und daraus ein Transaction History Log (THL) Stream erstellt. Der zusätzliche Aufwand, der durch das Anlegen der Log-Dateien entsteht, wird durch die Verwendung einer parallelen Replikation ausgeglichen. Ein Beispiel zeigt die Abbildung 2. Es wird auf zwei verschiedene Slaves parallel repliziert, wodurch eine bessere Verfügbarkeit der Daten erreicht wird.

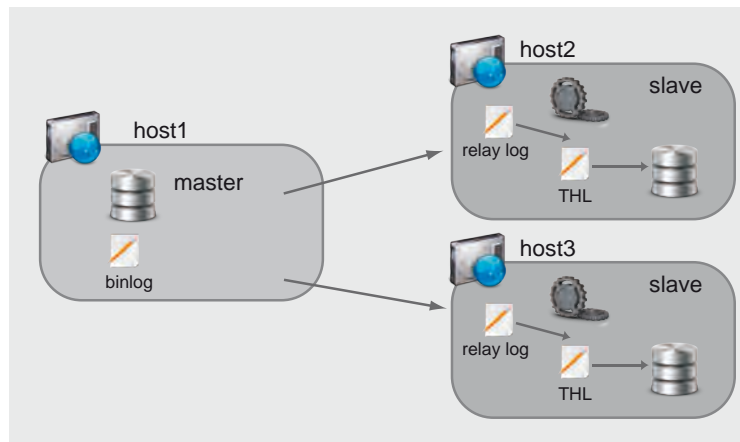


Abb. 2: Funktionsweise einer Slave-Direct-Replikation.

```

+-----+
| Tables_in_tungsten_replication355 |
+-----+
| consistency |
| heartbeat   |
| trep_commit_seqno |
| trep_shard  |
+-----+
    
```

Abb. 3: Tabellen der Tungsten-Datenbank.

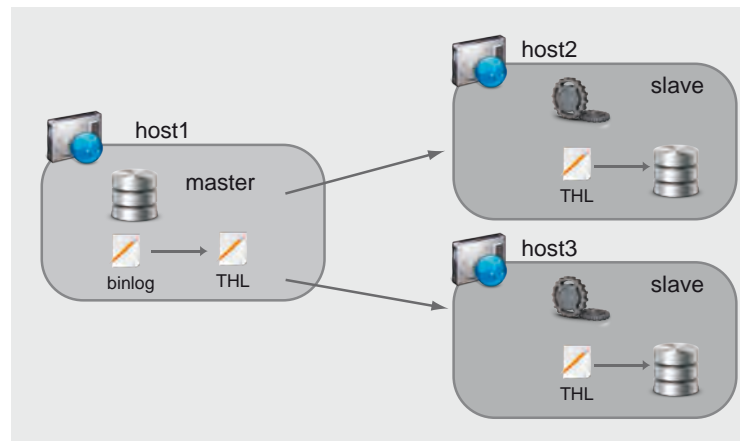


Abb. 4: Funktionsweise einer Master-Slave-Replikation.

Der Replikationsmechanismus `master-slave` ermöglicht die Konfiguration eines Master und der dazugehörigen Slaves. Es können ein oder mehrere Hosts in den Cluster aufgenommen werden. Dabei erzeugt der Tungsten Replicator eine eigene Datenbank mit dem Namen „tungsten\_servicename“. Die Tabellen der Datenbank werden in Abbildung 3 gezeigt.

Statusinformationen der Replikation werden dort hinterlegt und können bei Bedarf abgerufen werden. Der

```
Usage: ./replicator { console | start | stop | restart |
condrestart | status |dump }
```

Abb. 5: Benutzung des Replicator.

```
./trepctl services
Processing services command...
NAME          VALUE
----          -
appliedLastSeqno : 25
appliedLatency  : 0.499
role           : master
serviceName     : replication355
serviceType    : local
started        : true
state          : ONLINE
Finished services command...
```

Abb. 6: Ausgabe von ./trepctl services.

## Glossar

### Binary Logs

Beinhalten die Transaktionsprotokolle von MySQL.

### Tungsten Replicator

OpenSource Replication Engine für MySQL und andere Datenbanken.

### Transaction Historical Log

Stream der replizierten nachgefahrenen Transaktionen.

## Links

- ▶ [1] Homepage des Tungsten Replicator:  
<http://code.google.com/p/tungsten-replicator/>
- ▶ [2] Wiki für den Tungsten Replicator:  
<http://code.google.com/p/tungsten-replicator/w/list>
- ▶ [3] Download-Bereich:  
<http://code.google.com/p/tungsten-replicator/downloads/list>
- ▶ [4] Tungsten Enterprise:  
<http://www.continuent.com>

Master verarbeitet die Binary Logs und versendet die THL an die jeweiligen Slaves. Diese fahren die Transaktionen nach und halten ihren Datenbestand auf diese Weise up-to-date. Ein Beispiel dieser Replikation ist in Abbildung 4 zu sehen.

## Administration

Die Administration des Tungsten Replicator erfolgt durch diverse mitgelieferte Werkzeuge:

- `replicator`
- `trepctl`
- `thl`

Das Werkzeug `replicator` ermöglicht die eigentliche Replikation. Er lädt sämtliche Einstellungen wie z.B. Filter oder sonstige Skripte (siehe Abbildung 5). Die zugrunde liegende Konfiguration des Replicator findet in der Konfigurationsdatei `static."servicename".properties` statt. An dieser Stelle können z.B. weitere Skripte, Filter o.ä. für den Tungsten Replicator eingebunden werden. Diese können zusätzliche Aufgaben, wie das Ignorieren von bestimmten Tabellen oder den Failover des Master auf einen Slave, übernehmen.

Das Werkzeug `trepctl` ermöglicht eine genaue Administration der definierten Services. So ermöglicht es, sowohl einen Service online / offline zu schalten als auch das Überspringen von bestimmten Sequenzen. In der Abbildung 6 ist eine Ausgabe der aktuellen Services dargestellt. Darüber kann nachvollzogen werden, welcher Service offline bzw. online ist, welche Rolle er inne hat und viele weitere Informationen. Dieses Werkzeug bietet eine große Vielfalt an Möglichkeiten zur Administration. Das Werkzeug `thl` ermöglicht eine genaue Analyse des Transaction History Log. Mit dem Befehl `thl -service Servicename info` bekommt der Anwender eine Information über die Logs und die Angabe der minimalen und maximalen Sequenznummern, die in den Logs geschrieben wurden. Eine detaillierte Ausgabe der THL erhält der Anwender mit dem Befehl `thl -service servicename list`. Dabei werden sämtliche Sequenzen und die dazu gehörigen SQL-Befehle gelistet.

## Fazit

Der Tungsten Replicator stellt eine gute OpenSource-Alternative zu Oracle GoldenGate dar, um in heterogenen Datenbanklandschaften Replikationen zu betreiben. Die Anwendung lässt sich schnell und problemlos installieren. Der administrative Aufwand ist bei vorhandenen Kenntnissen des jeweiligen Datenbankproduktes verhältnismäßig gering.

Im Vergleich zu der herkömmlichen Replikation mit MySQL ermöglicht der Tungsten Replicator mehr Möglichkeiten bzgl. der Administration und des Monitoring mit zusätzlichen Werkzeugen und Skripten. Darüber hinaus bietet er eine wesentlich größere Flexibilität. Im Vergleich zu MySQL lässt sich beispielsweise eine Multi-Master-Slave-Konfiguration realisieren oder es können eigene Filter und Skripte eingebunden werden.



Carsten Hummel  
([info@ordix.de](mailto:info@ordix.de))



Rätsel

# Larry und das Passwort-Problem



Larry ist mal wieder für ein Projekt eingeteilt. Er ist für den Betrieb der Oracle-Datenbanken bei dem Kunden Herr Wichtig zuständig. Aber wie so häufig, hat Larry ein Problem. Können Sie ihm helfen?

## Der Benutzer Alice

Auf der Datenbank ORCL1 wurde in dem Schema Alice ein private DB-Link „ordix“ angelegt. Mithilfe dieses DB-Links greift der Benutzer Alice auf das Schema Bob in der Datenbank ORCL2 zu. Da das Passwort für das Schema Bob auf der Datenbank ORCL2 geändert werden soll, bittet Herr Wichtig Larry darum, dass Passwort für den private DB-Link „ordix“ zu ändern.

Larry meldet sich als Benutzer SYS auf der Datenbank ORCL1 an und erhält bei dem Versuch, den DB-Link „ordix“ mit dem geänderten Passwort neu anzulegen, die Fehlermeldung aus Abbildung 1.

Larry wundert sich, dass er als Benutzer SYS keine hinreichenden Rechte hat. Können Sie Larry sagen, wie er einen private DB-Link als Benutzer SYS mit geändertem Passwort neu anlegen kann? Kleiner Tipp: Schreiben Sie hierzu eine PL/SQL-Routine.

## Können Sie Larry helfen?

Larry freut sich auf Ihren Lösungsvorschlag. Senden Sie bis zum **02. April 2012** Ihre Antwort an [kniffel@ordix.de](mailto:kniffel@ordix.de).

## Das Annotations-Problem wurde gelöst!

Larry bedankt sich für die richtige Lösung aus der letzten Ausgabe (siehe Abbildung 2) bei:

Sascha Arndt  
Klaus Höppner

```
SQL> alter session set current_schema = ALICE;
Session altered.

SQL> drop database link "ordix";
drop database link "ordix"
*
ERROR at line 1:
ORA-01031: insufficient privileges
```

Abb. 1: Fehlermeldung der Datenbank.

Damit Annotation-Informationen in der Klasse zur Laufzeit hinterlegt werden, müssen sie in der folgenden Form deklariert werden:

```
@Retention( RetentionPolicy.RUNTIME )
public @interface MyAnnotation {
    String initialValue();
    int length();
}
```

Wesentlich ist die Angabe:

```
@Retention(RetentionPolicy.RUNTIME).
```

Abb. 2: Lösung des Larry-Rätsels aus der Ausgabe 4/2011.



Ratgeber für Oracle Datenbanken

## Der Buffer Pool Advisor

Oracle Datenbanken verfügen über eine Menge an Ratgebern, die ihre Administration vereinfachen sollen. Aber wie nützlich und aussagekräftig sind diese Helfer wirklich? In diesem Artikel stellen wir den Buffer Pool Advisor vor und bewerten dessen Können.

### Was ist der Buffer Pool Advisor?

Oracle stattet seine Datenbanken mit einer Vielzahl an Ratgebern aus. Eine besondere Stellung nimmt dabei der Buffer Pool Advisor ein. Seine Aufgabe besteht darin, Vorschläge bezüglich der Größe des Database Buffer Cache zu unterbreiten.

Zu diesem Zweck werden verschiedene Zwischenspeichergrößen simuliert und die Auswirkungen der Änderung auf die Anzahl der Festplattenzugriffe (physical I/O) geschätzt.

Das Ergebnis des Ratgebers könnte z.B. sein, dass eine Verdopplung des Database Buffer Cache die Anzahl der Festplattenzugriffe halbiert. Wie verlässlich diese Schätzungen des Buffer Pool Advisor sind, wird am Ende dieses Artikels bewertet.

### Stets zu Ihren Diensten

Der Buffer Pool Advisor wurde bereits in der Oracle Version 9.0.1.0 eingeführt und steht seitdem per Default zur Verfügung.

Die Initialisierungsparameter `DB_CACHE_ADVICE` und `STATISTICS_LEVEL` steuern, ob der Buffer Pool Advisor eingeschaltet ist. Der Parameter `DB_CACHE_ADVICE` ist standardmäßig nicht gesetzt. In diesem Fall bestimmt der Parameter `STATISTICS_LEVEL`, ob der Buffer Pool Advisor aktiv ist. Dieser Parameter verfügt über die folgenden Einstellungsmöglichkeiten:

- **BASIC:** Keine Ratgeber und Statistiken stehen zur Verfügung.

- **TYPICAL:**  
Die meisten Ratgeber und Statistiken stehen zur Verfügung.
- **ALL:**  
Alle Ratgeber und Statistiken stehen zur Verfügung.

Per Default steht der Initialisierungsparameter STATISTICS\_LEVEL auf „TYPICAL“. Bei dieser Einstellung und bei der Angabe von „ALL“ ist der Buffer Pool Advisor im Einsatz. Die Einstellung „BASIC“ sorgt dafür, dass der Buffer Pool Advisor keine Empfehlungen generiert. Sobald der Parameter DB\_CACHE\_ADVICE gesetzt wurde, hat der Parameter STATISTICS\_LEVEL keinen Einfluss mehr auf das Verhalten des Buffer Pool Advisor.

### Alles im Griff

Über den Parameter DB\_CACHE\_ADVICE kann der Buffer Pool Advisor gezielt ein- oder ausgeschaltet werden.

- OFF: Der Buffer Pool Advisor ist ausgeschaltet.
- READY: Der Ratgeber ist ausgeschaltet, gesammelte Statistiken bleiben aber erhalten.
- ON: Der Buffer Pool Advisor ist eingeschaltet.

Eine Aktivierung des Parameters (setzen auf „READY“ oder „ON“) kann laut der Oracle Dokumentationen zu Problemen führen. Die Ursache hierfür ist ein zusätzlicher Speicherplatzbedarf innerhalb der System Global Area (SGA) für den Buffer Pool Advisor. Die Reservierung des Speicherplatzes kann unter Umständen bei laufender Instanz zu Fehlern führen.

### Vorhang auf!

Die Ergebnisse des Buffer Pool Advisor sind ein Bestandteil der AWR- und STATSPACK-Reports. In Abbildung 1 ist die Ausgabe des Buffer Pool Advisor in einem AWR-Report dargestellt. Die Ausgaben in diesen Reports entsprechen letztlich einer Abfrage der View V\$DB\_CACHE\_ADVICE, welche nur bei eingeschaltetem Buffer Pool Advisor Daten beinhaltet.

Die SQL-Abfrage in Abbildung 2 liefert eine ähnliche Darstellung der Ergebnisse, wie sie auch in einem STATSPACK-Report zu finden wäre.

**Buffer Pool Advisory**

- Only rows with estimated physical reads >0 are displayed
- ordered by Block Size, Buffers For Estimate

P	Size for Est (M)	Size Factor	Buffers for Estimate	Est Phys Read Factor	Estimated Physical Reads
D	400	0.10	49,475	8.70	20,073,568
D	800	0.20	98,950	5.43	12,536,611
D	1,200	0.30	148,425	3.96	9,125,240
D	1,600	0.40	197,900	3.33	7,676,259
D	2,000	0.50	247,375	2.90	6,693,021
D	2,400	0.60	296,850	2.44	5,625,409
D	2,800	0.70	346,325	2.18	5,020,389
D	3,200	0.80	395,800	1.63	3,760,155
D	3,600	0.90	445,275	1.10	2,526,761
D	4,000	1.00	494,750	1.00	2,307,011
D	4,400	1.10	544,225	0.90	2,085,815
D	4,800	1.20	593,700	0.86	1,973,952
D	5,200	1.30	643,175	0.83	1,923,361
D	5,600	1.40	692,650	0.82	1,898,576
D	6,000	1.50	742,125	0.81	1,869,467
D	6,400	1.60	791,600	0.80	1,856,545
D	6,800	1.70	841,075	0.78	1,795,548
D	7,200	1.80	890,550	0.76	1,746,426
D	7,600	1.90	940,025	0.74	1,696,009
D	8,000	2.00	989,500	0.70	1,606,065

Abb. 1: Ausgabe des Buffer Pool Advisor in einem AWR-Report.

### Zahlen über Zahlen

Ausgangspunkt zur Interpretation der Ergebnisse des Buffer Pool Advisor ist stets die Zeile, wo der Wert in der Spalte „Size Factor“ gleich 1,0 ist. Diese immer mittig gelegene Zeile beschreibt das Verhalten der Instanz mit den aktuell gewählten Konfigurationsparametern. Alle anderen Zeilen beinhalten Vorhersagen für das Instanzverhalten bei einer Änderung des Database Buffer Cache.

Der Wert in der Spalte „Size for Est (M)“ gibt die Größe des Database Buffer Cache in Megabyte an. In Abbildung 1 sind diesem Bereich der SGA aktuell 4.000 Megabyte zugewiesen. Die Spalte „Size Factor“ gibt die Größenänderung des Database Buffer Cache in Prozent an. Die Veränderung der Festplattenzugriffe (physical reads) im Verhältnis zum Ausgangswert wird in der Spalte „Est Phys Read Factor“ abgelesen. Ein Wert von 0,80 in dieser Spalte bedeutet, dass die Anzahl an Festplattenzugriffen für die in dieser Zeile angegebene Größe des Zwischenspeichers bei 80 % des Ausgangswertes liegen würde.

### Der Ratschlag

Die Ergebnisse des Buffer Pool Advisor in Abbildung 1 sind folgendermaßen zu interpretieren. Seit Instanz-

```
SELECT substr(name,1,1) P
,      size_for_estimate „Size for Est (M)“
,      to_char(size_factor,'990.99') „Size Factor“
,      buffers_for_estimate „Buffers for Estimate“
,      to_char(estd_physical_read_factor,'990.99') „Est
Phys Read Factor“
,      estd_physical_reads „Estimated Physical Reads“
,      estd_physical_read_time „Est Phys Read Time“
,      estd_pct_of_db_time_for_reads „Est dbtime for Rds“
FROM   v$db_cache_advice;
```

Abb. 2: SQL-Abfrage für die Anzeige der Ergebnisse des Buffer Pool Advisor.

## Glossar

### AWR

Automatic Workload Repository - Das AWR besteht aus Tabellen im Schema SYS, in denen laufend Performance-Informationen zur Datenbank und dessen Server zentral gespeichert werden. Diese Funktionalität steht ab Oracle 10g EE zur Verfügung.

### FTS

Full Table Scan - Bei einem Full Table Scan werden zur Ausführung eines SQL-Statement alle Datenbankblöcke einer Tabelle gelesen. Ein Full Table Scan unterscheidet sich deutlich von einem Indexzugriff, bei dem nur selektiv Datenbankblöcke gelesen werden.

### SGA

System Global Area - Die SGA ist der von Oracle allokierte Hauptspeicher auf dem Datenbank-Server.

### STATSPACK

Werkzeug zur detaillierten Analyse von Datenbanken. Kostenfreie Alternative zu AWR-Reports.

### Data Base Buffer Cache

Speicherbereich innerhalb der System Global Area (SGA). Dient zum Zwischenspeichern von Datenbankblöcken.

Bild: © istockphoto.com | Magical world of reading | Lobke Peers

start wurden ca. 2,3 Mio. Festplattenzugriffe durchgeführt. Der Ratgeber schätzt, dass eine Vergrößerung des Database Buffer Cache um 100 % die Anzahl an Festplattenzugriffen um 30 % reduzieren würde.

Eine Halbierung des Database Buffer Cache hätte hingegen 190 % mehr Festplattenzugriffe zur Folge. Eine Verkleinerung des Zwischenspeichers hätte dementsprechend einen größeren Einfluss auf die Anzahl an Festplattenzugriffen als eine Vergrößerung.

## Bewertung

Eine Überprüfung und kritische Betrachtung der Ergebnisse des Buffer Pool Advisor ist nach unseren Erfahrungen vor der Umsetzung der Ratschläge unverzichtbar. Zudem hängt die Aussagekraft der Ergebnisse von dem Lastprofil der jeweiligen Oracle Datenbank ab.

Werden auf der Datenbank viele Full Table Scans (FTS) ausgeführt, sind die Ergebnisse des Buffer Pool Advisor in der Regel unbrauchbar. Full Table Scans führen im Gegensatz zum Lesen einzelner Blöcke zu einer völlig anderen Verwendung des Database Buffer Cache. Bei einem FTS werden alle Datenbankblöcke einer Tabelle für die Ausführung einer SQL-Anweisung gelesen. Die auf diese Weise eingelesenen Datenbankblöcke werden eher aus dem Cache entfernt, als über Indizes gelesene Blöcke. Aufgrund dieses Verhaltensmusters ignoriert der Ratgeber höchst wahrscheinlich die Ausführung von FTS im Rahmen seiner Schätzung.

Bei vielen Zugriffen über Indizes unterliegt die Aussagekraft der Ergebnisse meist erheblichen Schwankungen. Die Präzision der Ergebnisse variiert sehr stark, sodass die Ergebnisse oftmals lediglich richtige Tendenzen aufzeigen. Die Vorhersagen für eine Verkleinerung des Database Buffer Cache sind in der Regel besser als diejenigen für eine Vergrößerung. Auch für den Zugriff über Indizes liefert der Ratgeber manchmal unbrauchbare Ergebnisse.

Nach unseren Erfahrungen liefert der Buffer Pool Advisor keine verlässlichen Vorhersagen. Seine Ergebnisse sollten lediglich als unterstützende Informationsquelle für detaillierte Untersuchungen herangezogen werden.



Sven Loer  
(info@ordix.de).

IBM DB2

# Workload Manager (WLM)

Datenbank-Server müssen aus sehr vielfältigen Quellen Informationen verarbeiten, unter Umständen auch große Mengen von Daten. Dabei kann es passieren, dass in Abhängigkeit von unterschiedlichen Lasten, die Systemantwortzeiten starken Schwankungen unterliegen. Dieser Artikel beschreibt, welche Möglichkeiten der Workload Manager bietet, um die Datenbankleistung zu steuern.

## Wozu braucht man ein Workload Management?

Ein gutes Workload Management dient dazu, die Anfragen an einen Datenbank-Server zu kontrollieren und mit maximaler Effizienz zu steuern. Der DB2 Workload Manager ist ein Werkzeug zum Steuern und Monitoren der Datenbanklast [1]. Der Manager kommt zum Einsatz wenn das Augenmerk der Performance-Optimierung gilt, d. h. dass die Systemantwortzeiten minimal oder berechenbar konstant sein sollen.

Werden z.B. bei Warenwirtschaftsanwendungen die Antwortzeiten bemängelt, wird zunächst nach lastintensiven SQL-Abfragen gesucht, die im Tagesgeschäft während der Hauptlast nichts zu suchen haben. Nach Rücksprache mit den Benutzern müssen diese Anwendungen dann vom Datenbankadministrator von der Datenbank manuell entfernt werden.

Somit stehen dann die Ressourcen für die eigentlich notwendigen Anwendungen wieder zur Verfügung. Diese Vorgehensweise beinhaltet eine zeitintensive Suche nach arbeitslastintensiven Anwendungen und eine radikale Beendigung der Verbindungen. Die Nutzung des Workload Manager bietet hier eine Erleichterung für den Datenbankadministrator. Er steuert die Arbeitslasten automatisch und ordnet diese nach Bedarf und Priorität den vorhandenen Ressourcen zu. Hierzu ermittelt er die Arbeitslast über einen bestimmten Zeitraum, bestimmt die Zuteilung der Last und überwacht diese im Anschluss.

## Die Ermittlung der Workload

Nach der Definition der Ziele erfolgt die Ermittlung der Workload in drei Phasen (siehe Abbildung 1).

- **Phase 1:**  
Ermittlung der Arbeitslast vor dem Eintritt in den Server
- **Phase 2:**  
Die Verwaltung der Arbeitslast bei der Ausführung
- **Phase 3:**  
Die Überwachung der Arbeitslast

## Ermitteln von Aktivitäten der Workload

Eine Workload ist ein Objekt, das die eingehende Arbeit auf einer Datenbank beschreibt. Die Identifikation kann entweder nach dem Ursprung oder dem Typ der Workload erfolgen.

Sie basiert auf den Verbindungsattributen des Benutzers. Attribute sind zum Beispiel die Applikations-ID, die Authentifizierung, die IP-Adresse des Client oder auch verschiedene Kombinationen dieser Attribute.

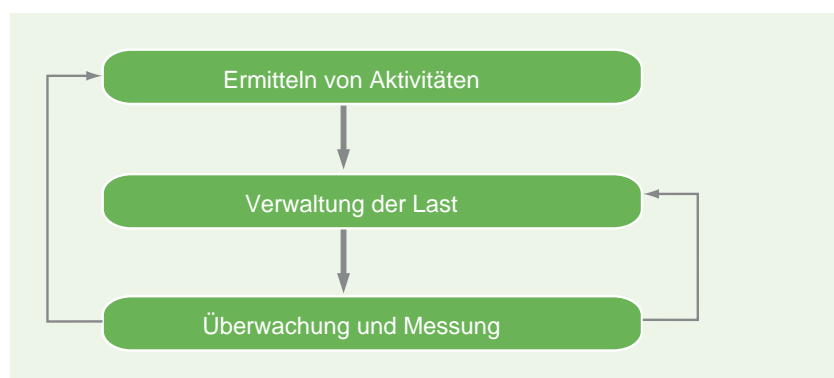


Abb. 1: Die drei Phasen zur Ermittlung der Workload.

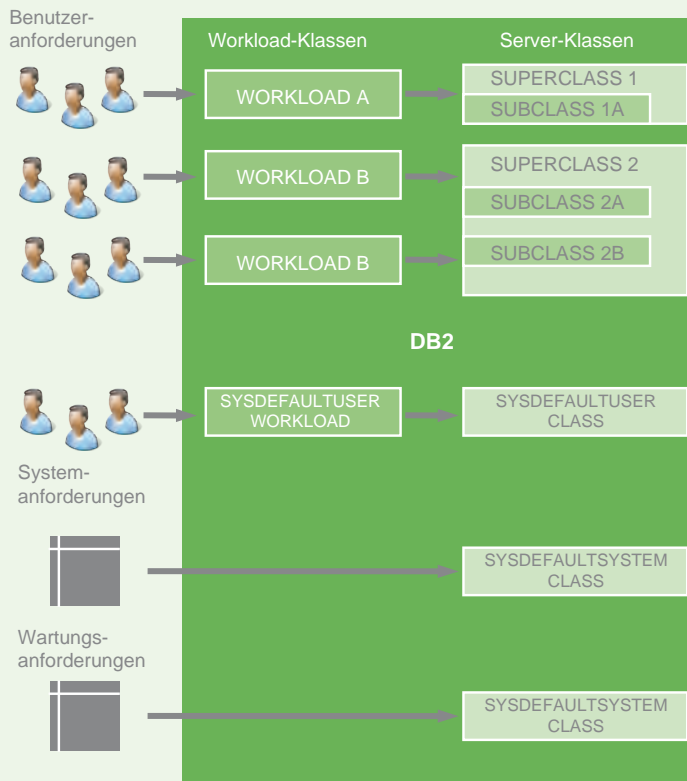


Abb. 2: Serviceklassen und Workloads bei der DB2.

Jeder eingehenden Datenbankverbindung wird eine Workload zugeteilt. Verbindet sich ein Anwender mit der Datenbank, werden die Attribute der Verbindung verglichen und nach der Übereinstimmung zugeteilt. Sollte keine Zuordnung existieren wird diese der `SYSDEFAULTUSERWORKLOAD` zugeteilt. Eine Workload beschreibt die Arbeitslast und routet sie dann an die Serviceklasse weiter.

### Verwaltung der Arbeit (Serviceklassen)

In der Verwaltungsphase werden die Systemressourcen der Workload zugeordnet, damit die definierten Ziele erreicht werden können. Zudem wird beschrieben, welche Maßnahmen ergriffen werden, wenn die Ziele nicht erreicht werden. Die Serviceklasse ist der erste Punkt für die Ressourcenzuweisung. Sie definiert eine Ausführungsumgebung, in der die Arbeit erledigt werden kann. Diese Umgebung weist verfügbare Ressourcen zu und kann Schwellwerte angeben, die bestimmen wie die Arbeit ausgeführt werden soll.

Datenbankabfragen können auch priorisiert werden und bei zu vielen gleichen Abfragen, kann ein Teil automatisch in eine Warteschlange gestellt werden.

In den Serviceklassen können verschiedene Prozesse (z.B. Stammdaten- oder Bewegungsdatenverarbeitung) nach geschäftlichen Anforderungen zusammengefasst werden. Die Serviceklasse bietet eine zweischichtige Hierarchie. Dabei kann eine Superklasse mehrere Unterklassen enthalten, wodurch die Arbeit je nach Anforderung weiter unterteilt werden kann. Die Serviceunterklassen übernehmen, wenn nichts anderes angegeben ist, die Merkmale der Servicesuperklasse.

Die Servicesuperklasse ist zwar die höchste Stufe in der Klassenhierarchie, die eigentliche Arbeit wird allerdings von den Unterklassen ausgeführt. Jede Superklasse enthält eine Standardunterklasse, in der die Aktivitäten ausgeführt werden, falls die Aktivitäten nicht explizit einer anderen Unterklasse zugewiesen wurden. Die Ressourcen der Superklasse werden von den Unterklassen gemeinsam benutzt (siehe Abbildung 2).

### Vordefinierte Superklassen mit den jeweiligen Unterklassen

DB2 bietet standardmäßig drei vordefinierte Superklassen mit den jeweiligen Unterklassen:

- `SYSDEFAULTUSERCLASS (SYSDEFAULTSUBCLASS)` hier werden alle Benutzeraktivitäten der Standardbenutzer ausgeführt.
- `SYSDEFAULTSYSTEMCLASS (SYSDEFAULTSUBCLASS)` enthält alle internen Verbindungen und DB2-Threads.
- `SYSDEFAULTMAINTENANCECLASS (SYSDEFAULTSUBCLASS)` beinhaltet alle Hintergrundverbindungen, ist der Diagnosemonitor für Backup, Reorg, Runstats.

Jeder Datenbankverbindung wird eine Workload zugeteilt. Existiert keine benutzerdefinierte Workload, wird die Last dem Default Workload (`SYSDEFAULTUSERWORKLOAD`) zugeteilt. Dieser wird direkt auf der `SYSDEFAULTUSERCLASS` abgebildet.

Für die Steuerung der Ressourcen gibt es verschiedene Wege. In den Serviceklassen können die Prioritäten für die CPU (Agenten), den Arbeitsspeicher (Bufferpool) und Platten (Vorablesezugriff) gesteuert werden. Im Betriebssystem AIX können die DB2-Serviceklassen in den Serviceklassen des AIX Workload Manager eingebunden werden, um die Leistung zu optimieren.

## Die Überwachungsphase

Der Hauptzweck der Überwachungsphase ist es, den fehlerfreien Ablauf und die effektive Auslastung des Systems zu kontrollieren. Dies kann durch eine Echtzeitüberwachung mit Hilfe der DB2-Tabellenfunktionen oder dem DB2-Event-Monitor für eine Langzeitüberwachung erfolgen.

Hierdurch werden Informationen gewonnen, um Veränderungen am System rechtzeitig zu erkennen. Den Problemen kann auf den Grund gegangen werden und das System für eine optimale Effizienz modifiziert werden.

Für die Echtzeitüberwachung werden folgende Tabellenfunktionen benötigt:

- **WLM\_GET\_SERVICE\_CLASS\_WORKLOAD\_OCCURRENCES\_V97**  
Die Funktion zeigt die Workload-Ereignisse über die Datenbankpartitionen zu einer bestimmten Serviceklasse.
- **MON\_GET\_WORKLOAD**  
Messwerte für Workloads.
- **MON\_GET\_SERVICE\_SUBCLASS**  
Messwerte für eine oder mehrere Unterklassen.
- **WLM\_GET\_SERVICE\_SUPERCLASS\_STATS**  
Basisstatistiken für eine oder mehrere Superklassen.

Der Event-Monitor wird zur Langzeitüberwachung und zur Gewinnung von historischen Daten herangezogen. Er sammelt Informationen über Serviceklassen, Workloads oder Aktivitäten, die bestimmte vorgegebene Schwellwerte verletzt haben und wann dies passiert ist. Es werden Statistiken über die Anzahl von Aktivi-

## Glossar

### Runstats

Dienstprogramm zur Erzeugung von statistischen Daten über die Inhalte von DB2-Tabellen und deren Indizes.

### Reorg

Nicht mehr benötigter Speicherplatz wird freigegeben, die Tabellen mit Indizes werden neu aufgebaut.

## Links

- ▶ [1] Implementierung eines DB2 Workload Management in einem DWH:  
<http://www.ibm.com/developerworks/data/bestpractices/workloadmanagement/>
- ▶ [2] IBM Redbooks DB2 WLM für Linux, Unix und Windows:  
<http://www.redbooks.ibm.com/redbooks/pdfs/sg247524.pdf>

täten ermittelt, wann diese beendet wurden und über die mittlere Ausführungszeit.

## Fazit

Datenbankadministratoren erhalten mit dem DB2 Workload Managereinvielfältiges Werkzeug an die Hand. Mit einer Reihe neuer Monitoring-Elemente kann die Datenbankleistung gesteuert und maximiert werden.



Thomas Schwarz  
([info@ordix.de](mailto:info@ordix.de)).

## Seminarempfehlung: IBM DB2 für Linux/Unix/Windows Monitoring und Tuning

- ▶ **Informationen/Online-Anmeldung:** <http://training.ordix.de/siteengine/action/load/kategorie/Datenbanken/nr/644/index.html>

In diesem Seminar lernen Sie anhand der vorhandenen Werkzeuge, DB2-Datenbanken für Linux/Unix/Windows zu analysieren und Performance-Maßnahmen durchzuführen.

### Seminarinhalte

- Unterschiede zwischen den Datenbanktypen OLTP und OLAP
- Monitoring Tools
- Architektur und Speichermodell inklusive STMM
- Physikalisches Datenbank-Layout
- Parameter, Datenbank-Caches und Bufferpool-Strategien
- Locking/Logging aus der Performance-Perspektive
- Maintenance Utilities zur Performance-Steigerung | Tunen von Utilities
- Tabellen und Datentypen
- Parallelisierung | SQL Tuning
- Vertiefung der Theorie durch praktische Übungen und Beispiele

### Termine

- 02.04. - 04.04.2012 in Wiesbaden
- 16.07. - 18.07.2012 in Wiesbaden
- 08.10. - 10.10.2012 in Wiesbaden

**Seminar-ID:** DB-DB2-05

**Dauer:** 3 Tage

**Preis pro Teilnehmer:** 1.290,00 € (zzgl. MwSt.)

**Frühbucherpreis:** 1.161,00 € (zzgl. MwSt.)

Datenbanken				März - Juni 2012
DB-DB-02	Datenbank-Hochverfügbarkeitslösungen für Entscheider	1 Tag	550,00 €	02.04.2012   01.06.2012
DB-DB-03	Datawarehouse Grundlagen	3 Tage	1.290,00 €	02.04.2012
DB-DB-01	Datenbank-Modellierung	2 Tage	890,00 €	10.04.2012
DB-ORA-01	Oracle SQL	5 Tage	1.890,00 €	07.05.2012   11.06.2012
DB-ORA-01A	Oracle SQL für Experten	3 Tage	1.290,00 €	23.04.2012
DB-ORA-02	Oracle Datenbankprogrammierung mit PL/SQL Grundlagen	5 Tage	1.890,00 €	16.04.2012   25.06.2012
DB-ORA-34	Oracle Datenbankprogrammierung mit PL/SQL Aufbau	5 Tage	1.890,00 €	26.03.2012   18.06.2012
DB-ORA-42	Oracle PL/SQL Tuning	3 Tage	1.890,00 €	18.06.2012
DB-ORA-44	Entwicklung von Web-Anwendungen mit Oracle PL/SQL	2 Tage	1.090,00 €	29.03.2012   21.06.2012
DB-ORA-43	Oracle APEX Programmierung	3 Tage	1.890,00 €	05.03.2012   14.05.2012   11.06.2012
DB-ORA-38	Objektorientierung in Oracle	5 Tage	1.990,00 €	16.04.2012
DB-ORA-03	Oracle Datenbankadministration Grundlagen	5 Tage	1.990,00 €	19.03.2012   23.04.2012   21.05.2012   18.06.2012
DB-ORA-04	Oracle Datenbankadministration Aufbau	5 Tage	1.990,00 €	19.03.2012   07.05.2012
DB-ORA-06	Manuelles Oracle Backup und Recovery & weitere Konzepte	5 Tage	1.990,00 €	auf Anfrage
DB-ORA-32	Oracle Backup und Recovery mit RMAN	5 Tage	1.990,00 €	16.04.2012   18.06.2012
DB-ORA-07	Oracle Tuning und Monitoring	5 Tage	1.990,00 €	12.03.2012   07.05.2012
DB-ORA-41	Oracle AWR und ASH Analyse und Interpretation	3 Tage	1.290,00 €	21.05.2012
DB-ORA-11	Oracle Troubleshooting Workshop	5 Tage	1.990,00 €	19.03.2012   25.06.2012
DB-ORA-08	Oracle 11gR2 RAC und Grid Infrastructure	5 Tage	1.990,00 €	21.05.2012
DB-ORA-15	Oracle 11g Neuheiten	5 Tage	1.990,00 €	26.03.2012   25.06.2012
DB-ORA-33A	Oracle Security	4 Tage	1.590,00 €	10.04.2012
DB-ORA-31	Oracle Data Guard	4 Tage	1.590,00 €	16.04.2012
DB-ORA-35	Oracle Grid Control	3 Tage	1.290,00 €	14.05.2012
DB-ORA-36	Oracle Replikation	3 Tage	1.290,00 €	auf Anfrage
DB-ORA-37	Oracle Streams Advanced Queuing	3 Tage	1.290,00 €	10.04.2012
DB-ORA-39	Oracle Migration und Patching	3 Tage	1.290,00 €	29.05.2012
DB-ORA-40	Oracle Capacity Planning	3 Tage	1.290,00 €	auf Anfrage
DB-INF-01	IBM Informix SQL	5 Tage	1.790,00 €	23.04.2012
DB-INF-02	IBM Informix Administration	5 Tage	1.990,00 €	21.05.2012
DB-INF-04	IBM Informix Dynamic Server Backup und Recovery	3 Tage	1.290,00 €	29.05.2012
DB-INF-03	IBM Informix Tuning und Monitoring	5 Tage	1.990,00 €	26.03.2012   18.06.2012
DB-INF-07	IBM Informix Hochverfügbarkeits-Technologien unter Unix	4 Tage	1.590,00 €	07.05.2012
DB-DB2-01	IBM DB2 für Linux/Unix/Windows SQL Grundlagen	5 Tage	1.890,00 €	23.04.2012   25.06.2012
DB-DB2-02	IBM DB2 für Linux/Unix/Windows Administration	5 Tage	1.990,00 €	07.05.2012
DB-DB2-05	IBM DB2 für Linux/Unix/Windows Monitoring und Tuning	3 Tage	1.290,00 €	02.04.2012
DB-MY-01	MySQL Administration	3 Tage	1.190,00 €	10.04.2012
DB-MSSQL-1	Microsoft SQL Server Administration	5 Tage	1.790,00 €	auf Anfrage
Entwicklung				März - Juni 2012
E-UML-01	Anforderungsanalyse mit UML	3 Tage	1.190,00 €	16.04.2012   04.06.2012
E-SWA-01	Softwarearchitekturen	5 Tage	1.890,00 €	07.05.2012   25.06.2012
P-PHP-01	PHP Programmierung Grundlagen	5 Tage	1.690,00 €	16.04.2012
P-PHP-02	PHP Programmierung Aufbau	3 Tage	1.190,00 €	14.05.2012
P-PERL-01	Perl Programmierung Grundlagen	5 Tage	1.690,00 €	16.04.2012
P-PERL-02	Perl Programmierung Aufbau	5 Tage	1.690,00 €	25.06.2012
P-UNIX-01	Shell   Awk und Sed	5 Tage	1.690,00 €	11.06.2012
P-UNIX-01A	Awk Intensiv-Workshop	3 Tage	1.190,00 €	02.04.2012
P-XML-01	Einführung in XML	3 Tage	1.190,00 €	02.04.2012   04.06.2012
P-XML-02	XML Programmierung unter Java mit DOM und SAX	2 Tage	890,00 €	auf Anfrage
P-XML-03	Oracle und XML	3 Tage	1.190,00 €	auf Anfrage
Web- und Applikations-Server				März - Juni 2012
INT-04	Apache Web-Server Installation und Administration	3 Tage	1.190,00 €	02.04.2012
INT-07	Tomcat Konfiguration und Administration	3 Tage	1.190,00 €	16.04.2012   14.05.2012
INT-08	WebSphere Application Server Installation und Administration	3 Tage	1.390,00 €	04.06.2012
INT-11	Administration und Konfiguration für JBoss	3 Tage	1.190,00 €	16.04.2012

## Informationen und Anmeldung

Für Informationen und Fragen zu individuell zugeschnittenen Seminaren, Ausbildungsreihen oder Inhouse-Schulungen stehen wir Ihnen gerne zur Verfügung. Auf Wunsch senden wir Ihnen auch unser komplettes Seminarprogramm zu.

**Online-Anmeldung,  
aktuelle Seminarinhalte  
und Termine unter:**

<http://training.ordix.de>





Systemmanagement					März - Juni 2012
SM-NAG-01	Systemüberwachung mit Nagios Grundlagen	3 Tage	1.190,00 €	19.03.2012   18.06.2012	
SM-NAG-02	Systemüberwachung mit Nagios Aufbau	2 Tage	890,00 €	22.03.2012   21.06.2012	
Java-JEE					März - Juni 2012
OO-01	Einführung in die Objektorientierte Programmierung	3 Tage	1.190,00 €	10.04.2012	
P-JAVA-01	Java Programmierung Grundlagen	5 Tage	1.690,00 €	23.04.2012	
P-JAVA-03	Java Programmierung Aufbau	5 Tage	1.690,00 €	11.06.2012	
P-JAVA-02	Java GUI Entwicklung mit Swing	5 Tage	1.690,00 €	11.06.2012	
P-JEE-01	JEE für Entscheider	1 Tag	590,00 €	16.04.2012	
P-JEE-02	Einführung in JEE	3 Tage	1.290,00 €	17.04.2012	
P-JEE-03A	JSP und Servlet Programmierung	5 Tage	1.590,00 €	18.06.2012	
P-JEE-04	EJB Programmierung	5 Tage	1.590,00 €	11.06.2012	
P-JEE-05	Web-Anwendungen mit JavaServer Faces (JSF)	5 Tage	1.590,00 €	07.05.2012	
P-JEE-06	Entwickeln mit dem Spring-Framework	3 Tage	1.190,00 €	04.06.2012	
INT-05	Java Web Services	3 Tage	1.190,00 €	02.04.2012	
J-HIB-01	Hibernate und die Java Persistence API	5 Tage	1.690,00 €	26.03.2012   25.06.2012	
P-JEE-08	Java Performance Tuning	3 Tage	1.290,00 €	10.04.2012	
Betriebssysteme					März - Juni 2012
BS-01	Unix/Linux Grundlagen für Einsteiger	5 Tage	1.690,00 €	26.03.2012   18.06.2012	
BS-25	Unix AufbauSeminar für Datenbank- und Applikationsbetreuer	5 Tage	1.790,00 €	25.06.2012	
BS-02	Linux Systemadministration	5 Tage	1.690,00 €	12.03.2012   16.04.2012	
BS-09	Linux Hochverfügbarkeits-Cluster	3 Tage	1.290,00 €	07.05.2012	
BS-19	Linux Cluster mit Pacemaker und Heartbeat 3	3 Tage	1.190,00 €	23.04.2012   29.05.2012	
BS-16	OpenLDAP - Praxiseinsatz im Netzwerk	4 Tage	1.490,00 €	11.06.2012	
BS-03	Solaris Systemadministration Teil I	5 Tage	1.990,00 €	23.04.2012	
BS-04	Solaris Systemadministration Teil II	5 Tage	1.990,00 €	07.05.2012	
BS-06	Solaris kompakt für erfahrene Unix/Linux-Umsteiger	5 Tage	1.990,00 €	25.06.2012	
BS-24	Solaris 11 Administration Neuheiten	5 Tage	1.990,00 €	07.05.2012	
BS-18	Solaris Virtualisierung mit ZFS und Container (Zonen)	5 Tage	1.990,00 €	11.06.2012	
BS-23	Solaris Virtualisierung mit LDOM	3 Tage	1.290,00 €	02.04.2012	
AIX-01	IBM AIX Systemadministration Grundlagen	5 Tage	1.990,00 €	07.05.2012	
AIX-02	IBM AIX Installation, Backup und Recovery mit NIM	3 Tage	1.290,00 €	21.05.2012	
AIX-03	Analyse v. komplexen AIX Performance Problemen	5 Tage	2.300,00 €	21.05.2012	
Projektmanagement					März - Juni 2012
PM-01	IT-Projektmanagement - Methoden und Techniken	5 Tage	1.990,00 €	16.04.2012	
PM-06	System. Projektmanagement - Projektteams souverän führen	4 Tage	1.850,00 €	24.04.2012	
PM-08	Agiles Projektmanagement mit SCRUM	2 Tage	1.100,00 €	12.03.2012   11.06.2012	
PM-09	Capability Maturity Model Integration (CMMI)	2 Tage	1.100,00 €	14.06.2012	
PM-10	Kennzahlen der IT	2 Tage	1.100,00 €	24.05.2012	
PM-07	Krisenmanagement in Projekten	2 Tage	1.100,00 €	29.05.2012	
PM-05	IT-Projektcontrolling	3 Tage	1.290,00 €	10.04.2012	
PM-11	Konfliktmanagement	2 Tage	1.100,00 €	31.05.2012	
IT-Management					März - Juni 2012
MGM-02	IT-Architekturen	3 Tage	1.650,00 €	23.04.2012	
MGM-01	E-Business	2 Tage	1.100,00 €	26.04.2012	
MGM-07	IT-Strategien	2 Tage	1.100,00 €	26.03.2012   25.06.2012	
MGM-08	Strategien effizient entwickeln	3 Tage	1.650,00 €	auf Anfrage	
MGM-03	IT-Management	5 Tage	1.990,00 €	05.03.2012   07.05.2012	
MGM-05	IT-Risikomanagement	3 Tage	1.650,00 €	18.06.2012	
MGM-04	IT-Prozessmanagement	3 Tage	1.650,00 €	21.05.2012	

Zentrale: **ORDIX AG**  
 Westermauer 12 - 16  
 33098 Paderborn  
 Tel.: 05251 1063-0

Seminarzentrum: **ORDIX AG**  
 Kreuzberger Ring 13  
 65205 Wiesbaden  
 Tel.: 0611 77840-00

Unsere Seminarstandorte sind: Wiesbaden, Bielefeld und Hannover.  
 Die Preise gelten pro Seminar pro Teilnehmer in Euro  
 zzgl. ges. MwSt., Inhouse-Preise auf Anfrage.



Neue Reihe: Hibernate für Fortgeschrittene (Teil I)

# Die Batch-Fetching-Strategie für Relationen richtig anwenden

Hibernate bietet eine Menge von Konfigurationsmöglichkeiten, mit denen unter anderem die Performance des Systems wesentlich verbessert werden kann. Allerdings wenn man sie nicht richtig einsetzt, kann der entgegengesetzte Effekt eintreten und das Programm langsamer werden. In diesem Artikel richten wir unser Augenmerk auf die Batch-Fetching-Strategie und zeigen anhand einer Beispielanwendung, was hier zu berücksichtigen ist.

## Das Beispiel

Im Folgenden gehen wir davon aus, dass wir für einen Kunden eine Filmverwaltung realisieren sollen. Zentraler Bestandteil dieses Projektes ist eine Webapplikation für die Recherche und die Ausgabe der Treffer. In der ersten Ausbaustufe sollen nur Filme mit den dazugehörigen Schauspielern berücksichtigt werden. Zwischen diesen beiden Entities gibt es eine m:n-Relation, da an einem Film in der Regel mehrere Schauspieler mitwirken und ein Schauspieler meistens auch in mehreren Filmen mitgespielt hat. Das

entsprechende Hibernate Mapping ist in Abbildung 1 verkürzt dargestellt.

## Der nicht optimierte Versuch

In der Webapplikation kann unter anderem nach einem Schauspieler recherchiert werden, um dessen Filme auszugeben. Da wir uns hier mit der Performance beschäftigen wollen, die in der Regel bei datenbankorientierten Anwendungen durch die Anzahl und Komplexität der SELECT-Anweisungen bestimmt wird,

werden im Folgenden zusätzlich zu der eigentlichen Programmausgabe auch die SQL-Anweisungen angegeben.

So führt eine Suche nach dem Schauspieler „Tom Hanks“ zu der in Abbildung 2 dargestellten Ausgabe. Die Ausgaben sind dabei für die bessere Lesbarkeit etwas überarbeitet worden. Um die generierten Statements besser nachvollziehen zu können, stehen außerdem die IDs der Datensätze in Klammern am Ende der Zeile.

Wie zu erkennen ist, wird das erste Statement abgeschickt, sobald das erste Mal auf die Kollektion der Filme eines Schauspielers zugegriffen wird. Mit jedem Zugriff auf die Schauspieler eines Films wird wiederum diese Kollektion geladen.

Dies bedeutet, dass für die Ausgabe eines Schauspielers, der in 100 Filmen mitgewirkt hat, 101 Statements notwendig sind (weshalb dieser Sachverhalt auch unter dem Namen „N+1 SELECT-Problem“ bekannt ist). Bei einer Webanwendung, die von tausenden Besuchern gleichzeitig genutzt wird, führt diese Strategie schnell zu einer schlechten Antwortzeit. Deshalb ist hier eine Optimierung dringend notwendig.

### Die optimierte Variante

Wird eine Kollektion, die noch nicht initialisiert bzw. eingeladen wurde, verwendet (d.h. es werden die Methoden `size()`, `get()`, `add()`, `remove()` oder eine ähnliche Methode aufgerufen), so wird diese Kollektion per Query aus der Datenbank von Hibernate geladen. Dieses sogenannte lazy loading ist der Default. Es kann aber für jede Kollektion separat ein eager loading eingestellt werden, was in bestimmten Fällen sinnvoll sein kann. In der Regel aber ist lazy loading das vernünftiger Vorgehen, da sich sonst schnell die halbe Datenbank im Speicher befindet.

Durch die Definition einer `BatchSize` für eine Kollektion kann das oben beschriebene Verhalten von Hibernate etwas optimiert werden. Soll eine Kollektion, die mit einer `BatchSize` definiert ist, von Hibernate eingeladen werden, so schaut Hibernate in seinen internen Datenstrukturen nach, welche Kollektionen des gleichen Typs bisher noch nicht geladen wurden. Diese Kollektionen werden bis zur angegebenen `BatchSize` mit geladen. Hierfür ist nur ein Statement nötig!

Da wir die Anzahl der Ladevorgänge für die Schauspieler eines Films reduzieren wollen, passen wir das Mapping in der Klasse `Film` für die Many-to-Many-Relation Schauspieler an und fügen eine `BatchSize` hinzu (siehe Abbildung 3).

```
@Entity
public class Film {
    @Id
    @GeneratedValue(...)
    private int id;
    private String titel;
    @ManyToMany(...)
    @JoinTable(name = "JOIN_FILM_SCHAUSPIELER", ...)
    private Collection<Schauspieler> schauspieler;
...}

@Entity
public class Schauspieler {
    @Id
    @GeneratedValue(...)
    private int id;
    private String name;
    @ManyToMany(mappedBy = "schauspieler")
    private Collection<Film> filme;
...}
```

Abb. 1: Das Hibernate Mapping für die Entities „Film“ und „Schauspieler“.

```
Tom Hanks (2)
Mitwirkungen
select f.* from join_film_schauspieler jfs, film f where jfs.fk_film_id=f.id and jfs.fk_schauspieler_id=2
Apollo 13 (1)
select s.* from join_film_schauspieler jfs, schauspieler s where jfs.fk_schauspieler_id=s.id and jfs.fk_film_id=1
Tom Hanks (2)
Kevin Bacon (9)
Illuminati (10)
select s.* from join_film_schauspieler jfs, schauspieler s where jfs.fk_schauspieler_id=s.id and jfs.fk_film_id=10
Tom Hanks (2)
Ewan McGregor (11)
Sakrileg (12)
select s.* from join_film_schauspieler jfs, schauspieler s where jfs.fk_schauspieler_id=s.id and jfs.fk_film_id=12
Tom Hanks (2)
```

Abb. 2: Ausgaben bei der Recherche nach „Tom Hanks“ (nicht optimiert).

```
@ManyToMany(cascade = CascadeType.ALL)
@JoinTable(name = "JOIN_FILM_SCHAUSPIELER", ...)
@BatchSize(size = 100)
private Collection<Schauspieler> schauspieler;
```

Abb. 3: Hibernate Mapping mit `BatchSize`.

Startet man nun das Programm, so erhält man die Ausgaben aus Abbildung 4. Die gesamte Ausgabe kann nun mit nur zwei Statements erzeugt werden. Erst wenn die Anzahl der Filme eines Schauspielers 100 übersteigt, wird ein weiteres Statement an die Datenbank geschickt. Allgemein ausgedrückt, werden aus N+1-Statements  $(N/\text{BatchSize})+1$ -Statements. Wird also die `BatchSize` den Datenkonstellationen entsprechend gewählt, ist die Anzahl der Statements sehr viel geringer und die Performance unserer Applikation kann dementsprechend deutlich verbessert werden!

```

Tom Hanks (2)
Mitwirkungen
select f.* from join_film_schauspieler jfs, film f where
jfs.fk_film_id=f.id and jfs.fk_schauspieler_id=2
Apollo 13 (1)
select s.* from join_film_schauspieler jfs, schauspieler s where
jfs.fk_schauspieler_id=s.id and jfs.fk_film_id in (1, 12, 10)
Tom Hanks (2)
Kevin Bacon (9)
Illuminati (10)
Tom Hanks (2)
Ewan McGregor (11)
Sakrileg (12)
Tom Hanks (2)

```

Abb. 4: Ausgaben bei der Recherche nach „Tom Hanks“ (optimiert).

```

@Entity
public class Schauspieler {
    ...
    @ManyToMany(cascade = CascadeType.ALL)
    @JoinTable(name = "JOIN_SCHAUSPIELER_PRODUKTIONEN", ...)
    private Collection<Film> produktionen;
    ...
}

```

Abb. 5: Das erweiterte Hibernate Mapping für Schauspieler.

```

Tom Hanks (2)
Produktionen
select f.* from join_schauspieler_produktionen jsp, film f
where jsp.fk_film_id=f.id and jsp.fk_schauspieler_id=2
The Pacific (3)
Band of Brothers (6)
Mitwirkungen
select f.* from join_film_schauspieler jfs, film f where
jfs.fk_film_id=f.id and jfs.fk_schauspieler_id=2
Apollo 13 (1)
select s.* from join_film_schauspieler jfs, schauspieler s where
jfs.fk_schauspieler_id=s.id and jfs.fk_film_id in (1, 6, 10, 12, 3)
Tom Hanks (2)
Kevin Bacon (9)
Illuminati (10)
Tom Hanks (2)
Ewan McGregor (11)
Sakrileg (12)
Tom Hanks (2)

```

Abb. 6: Ausgaben, erweitert um die Filmproduktionen.

### Eine weitere Anforderung

Kaum ist das Programm schnell genug, werden beim Kunden neue Wünsche geweckt. Ein Schauspieler soll nun eine Kollektion all der Filme bekommen, die von ihm produziert wurden (siehe Abbildung 5). Die Filmproduktionen, in denen der Schauspieler mitgespielt hat sollen nun ebenfalls ausgegeben werden, was dann zum Log aus Abbildung 6 führt.

Auf den ersten Blick scheint alles in Ordnung zu sein: drei Statements wurden abgeschickt. Genau mit dem Ergebnis haben wir auch gerechnet. Also wo ist dann das Problem?, werden Sie sich sicher fragen.

Schauen wir uns doch das letzte Statement noch einmal etwas genauer an. Tom Hanks hat laut unserer Datenbank in den Filmen „Apollo 13 (1)“, „Illuminati (10)“ und „Sakrileg (12)“ mitgespielt. Zudem werden nun die Schauspieler aus den Filmen „The Pacific (3)“ und „Band of Brothers (6)“ ebenfalls geladen. Diese Filme hat er zwar produziert, doch von denen wird die Liste der Schauspieler doch gar nicht ausgegeben!

Gehen wir noch mal einen Schritt zurück. Durch das erste Statement werden auch die zuletzt genannten beiden Filme geladen. Die Kollektionen der Schauspieler dieser beiden Filme sind natürlich nicht initialisiert. Für die Ausgabe der produzierten Filme erfolgt die Initialisierung ebenfalls nicht, da die Liste der Schauspieler unangetastet bleibt.

Bei der Ausgabe der Filme, in denen Tom Hanks mitgespielt hat, hingegen werden die mitwirkenden Schauspieler benötigt. Beim ersten Zugriff auf eine Schauspieler-Kollektion ist Hibernate gezwungen, diese zu laden. Da eine BatchSize definiert wurde, wird nicht ein Statement der Form:

```

select s.* from join_film_schauspieler jfs,
schauspieler s where jfs.fk_schauspieler_
id=s.id and jfs.fk_film_id=?

```

verwendet, sondern ein Statement in der Art:

```

select s.* from join_film_schauspieler jfs,
schauspieler s where jfs.fk_schauspieler_
id=s.id and jfs.fk_film_id in (?, ?, ?, ...)

```

Wobei im `in`-Statement die Fragezeichen durch die IDs der Filme ersetzt werden, die sich zur Zeit im Speicher bzw. im Cache befinden. Hierbei werden leider auch die Schauspieler aus „The Pacific (3)“ und „Band of Brothers (6)“ berücksichtigt, da deren Schauspieler-Kollektionen nicht initialisiert sind und die BatchSize noch nicht überschritten wurde.

Dieser Vorgang bedeutet nicht nur eine Verschlechterung der Ladezeit sondern auch einen höheren Bedarf an Speicherplatz. Beides ist natürlich nicht erwünscht.

### Fazit

Mittels einer BatchSize kann die Anzahl der Statements, die Hibernate zum Laden benötigt, drastisch reduziert werden. Doch hier ist Vorsicht geboten. Schon bei relativ einfachen Strukturen kann die Angabe einer BatchSize zu unnötigen Instanzen führen,

was eine Verschlechterung der Antwortzeiten und einen höheren Speicherplatzverbrauch zur Folge hat.

Zur Beurteilung einer Optimierungsmaßnahme ist eine genaue Analyse der generierten SQL-Statements unbedingt notwendig. Bei längeren Traces kann dies zwar zu einer unangenehmen, langwierigen und unübersichtlichen Arbeit werden, doch es gibt hierzu keine Alternative. Dabei ist anzumerken, dass die normalen SQL-Traces, die von Hibernate geliefert werden, für diese Analyse meistens nicht ausreichen. Hier kann z.B. ein spezieller Log-Datenbanktreiber aussagekräftigere Daten erzeugen.

In einem aktuellen Projekt führte die Anwendung einer BatchSize zu einer erheblichen Verbesserung der Performance – in einem Anwendungsfall. Bei einem anderen Anwendungsfall hingegen beobachteten wir den oben beschriebenen Effekt (mehr Speicher, mehr Rechenzeit). So war man gezwungen, teilweise die gute Performance des ersten Anwendungsfalles zu opfern, um die Performance des zweiten zu verbessern. Erst dieser Kompromiss führte für beide Fälle zu einer akzeptablen Lösung.



Andreas Maßmann  
([info@ordix.de](mailto:info@ordix.de)).

## Glossar

### eager loading

Hierunter versteht man das sofortige Laden eines referenzierten Objektes.

### lazy loading

Hierunter versteht man das verzögerte Laden eines referenzierten Objektes. Das referenzierte Objekt wird erst bei Bedarf geladen. Hierdurch werden überflüssige Datenbankzugriffe und Speicherplatz eingespart. Realisiert wird diese Strategie durch den Einsatz von Proxies.

## Links

- ▶ [1] ORDIX news Artikel 03/2009 „Lazy und Eager Loading - ein guter Cache lädt ...“:  
[http://www.ordix.de/ORDIXNews/3\\_2009/Java\\_JEE/hibernate\\_lazy\\_eager\\_loading.html](http://www.ordix.de/ORDIXNews/3_2009/Java_JEE/hibernate_lazy_eager_loading.html)
- ▶ [2] ORDIX news Artikel 02/2009 „Easy Hibernate Cache - mehr PS oder mehr Sicherheit?“:  
[http://www.ordix.de/ORDIXNews/2\\_2009/Java\\_JEE/easy\\_hibernate\\_cache.html](http://www.ordix.de/ORDIXNews/2_2009/Java_JEE/easy_hibernate_cache.html)
- ▶ [3] ORDIX news Artikel 03/2008 „First-/Second Level und Query Cache - schneller als die Polizei erlaubt ...“:  
[http://www.ordix.de/ORDIXNews/3\\_2008/Java\\_J2EE\\_JEE/Performance\\_steigern\\_Query\\_Cache.html](http://www.ordix.de/ORDIXNews/3_2008/Java_J2EE_JEE/Performance_steigern_Query_Cache.html)
- ▶ [4] Leseempfehlung:  
Java Persistence mit Hibernate von Gaving King, Christian Bauer, Jürgen Dubau

**Bild:** © istockphoto.com | Video Wall | Erik Reis  
© iconfinder.com | search-icon

## Seminarempfehlung: Hibernate und die Java Persistence API

**Informationen/Online-Anmeldung:** <http://www.ordix.de/trainingsshop/siteengine/action/load/kategorie/Java-JEE/nr/549/index.html>

Die Java Persistence API ist die durch JEE standardisierte Programmierschnittstelle für objektrelationale Bibliotheken. Hibernate ist die wohl am weitesten verbreitete Implementierung dieser API. In diesem Seminar werden Sie mit der Hibernate-Implementierung von JPA vertraut gemacht. Hibernate ist ein Open Source Persistenz-Framework für die Speicherung von Java Objekten in relationalen Datenbanken. Durch deklaratives Mapping bietet es die Möglichkeit, einfach und effizient auf Basis relationaler Datenbanken zu entwickeln.

### Seminarinhalte

- Einführung
- O/R-Mapping
- Arbeiten mit persistenten Objekten
- Fortgeschrittene Themen
- Vertiefung der Theorie durch praktische Übungen und Beispiele

### Termine

26.03. - 30.03.2012 in Wiesbaden  
25.06. - 29.06.2012 in Wiesbaden  
10.09. - 14.09.2012 in Wiesbaden  
19.11. - 23.11.2012 in Wiesbaden

**Seminar-ID:** J-HIB-01

**Dauer:** 5 Tage

**Preis pro Teilnehmer:** 1.690,00 € (zzgl. MwSt.)

**Frühbucherpreis:** 1.521,00 € (zzgl. MwSt.)

Oracle Application Express (APEX)

# APEX-Administration „Light“

Dieser Artikel zeigt die einfache Administration einer APEX-Anwendung ohne tiefgreifende APEX- und / oder SQL-Kenntnisse. Zur Demonstration werden die aktuellen Versionen von Oracle Application Express (APEX 4.1), APEX Listener (1.1), SQL Developer (3.0) und Oracle Datenbank (11.2) verwendet.

## SQL Developer - Administration leicht gemacht

Der SQL Developer hilft bei der Entwicklung und Administration von Oracle Datenbanken. Als Alternative zum SQLPLUS\* wird er ab der Version 11g bei der Installation des Oracle Client mitgeliefert.

Mit dem SQL Developer können schon in der Standardversion ohne weitere Optionen administrative Aufgaben im APEX-Umfeld durchgeführt werden.

Hierzu zählen:

- Anwendung öffnen
- Anwendung ändern
- Aliasnamen vergeben
- Globale Benachrichtigungen erstellen
- Status der Anwendungen ändern
- Proxy-Einstellungen
- Diverse Berichte
- Anwendungen importieren
- Anwendung bereitstellen
- Anwendung löschen
- Refactoring der Anwendung
- DDL generieren

Im Folgenden werden diese Optionen näher betrachtet.

## Anwendung öffnen

Mit dem Aufruf des Punktes „Anwendung öffnen“, erhält der Administrator umfangreiche Informationen. Die ersten Zeilen zeigen die Applikations-ID und den Namen der Applikation. Genannt werden ihm sowohl der alphanumerische Aliasname als auch der interne Name.

Darüber hinaus wird dem Administrator das genutzte Parsing-Schema, der Ablageort der Bilder der Anwendung inklusive Logo angezeigt. Neben der Versionsnummer und dem Verfügbarkeitsstatus der Anwendung wird die Form der Authentifizierung und auch ein eventuell aktiviertes Debugging angezeigt.

Zusätzlich erhält er noch inhaltliche Informationen zu der letzten durchgeführten Änderung als Zeitstempel inkl. Benutzerkennung. Der Administrator hat die Möglichkeit die allgemeinen Informationen, wie die Anzahl der Seiten pro Anwendung als auch die zugehörigen Workspace-Informationen, hier zu ermitteln.

## Anwendung ändern

Unter „Anwendung ändern“ sind mehrere Unterpunkte subsumiert. So kann der alphanumerische Aliasname statt der Applikations-ID im Quellcode der Anwendung genutzt werden.

Durch diese Unterpunkte können Änderungen in einzelnen Bereichen der Anwendung vollzogen werden. Der Punkt „Umbenennen der Applikation“ führt zu einem neuen Namen einer Anwendung. Das „Erstellen einer globalen Nachricht“ dient dem Administrator dazu, die einzelnen Anwender / Entwickler einer APEX-Applikation zu informieren.

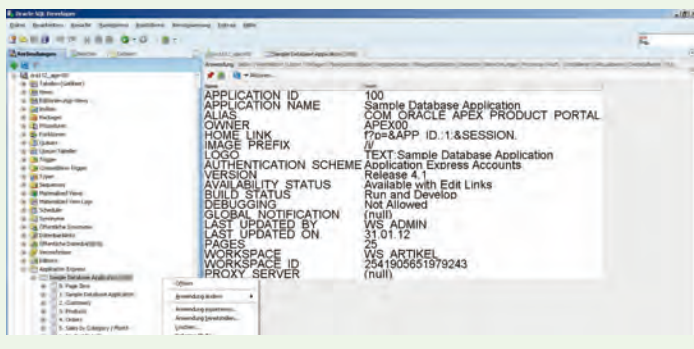


Abb. 1: Anwendungsadministration „Light“.

Mit dem nächsten Punkt „Statusänderung“ legt der Datenbankadministrator fest, für wen und wie die Anwendung nutzbar ist:

- Available (für alle verfügbar)
  - ▶ Sinnvoll in einer produktiven Umgebung
- Available with Edit-Link (für alle verfügbar mit Bearbeitungslinks)
  - ▶ Sinnvoll in der fachlichen Testphase einer Anwendung für eine schnellere Navigation
- Developer Only (nur für Benutzer mit Entwicklerprivilegien verfügbar)
  - ▶ Sinnvoll in der Entwicklungsphase einer Applikation
- Unavailable (nicht verfügbar)
  - ▶ Applikation soll exportiert werden. Hier bietet sich die Überleitung aus der Test- in die Produktionsumgebung an.

Mit dem Punkt „Proxy-Einstellungen“ bestimmt der Administrator einen Proxy Server für die Applikation.

Bei den jeweiligen Popup-Menüs lässt sich unter dem Reiter „SQL“ der zugehörige Quellcode ausgeben. Dadurch können die Änderungen auch in Skriptform mittels des Package WWV\_FLOW\_API durchgeführt werden.

Mit diesem Package und den einzelnen Funktionalitäten erhält der Administrator eine Fülle an Möglichkeiten zur Änderung einer Applikation. Für die weitergehende Nutzung sei auf die Oracle-Dokumentation verwiesen.

### Anwendung importieren

Durch das Einspielen einer entsprechenden Skriptdatei in die Datenbank wird eine Anwendung importiert. Wichtig ist hierbei die Angabe des jeweiligen Parsing-Schemas (siehe Abbildung 2 bis 6).

Die Eingabe der korrekten Applikations-ID ist zu beachten, da diese nicht im Konflikt mit einer bereits im System vorhandenen ID stehen darf. Im Vorfeld sollte dem Administrator bereits klar sein, ob diese ID in allen Umgebungen (Entwicklung/Test/Produktion) identisch sein soll oder muss!

### Anwendung bereitstellen

Diese Funktionalität ist in der hier genutzten Versionskombination der Oracle-Produkte nicht ver-



Abb. 2: Anwendungsimport (Schritt 1).



Abb. 3: Anwendungsimport (Schritt 2).

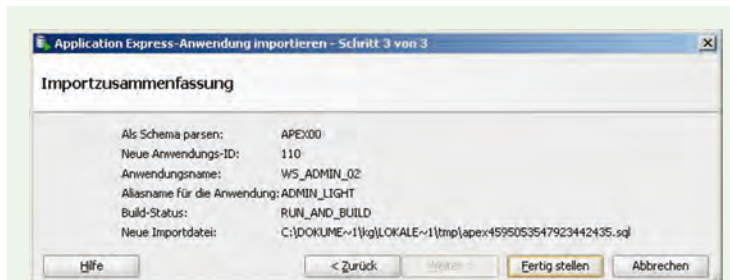


Abb. 4: Anwendungsimport (Schritt 3).

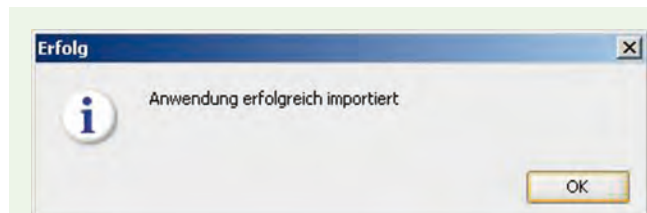


Abb. 5: Anwendungsimport Erfolgsmeldung.

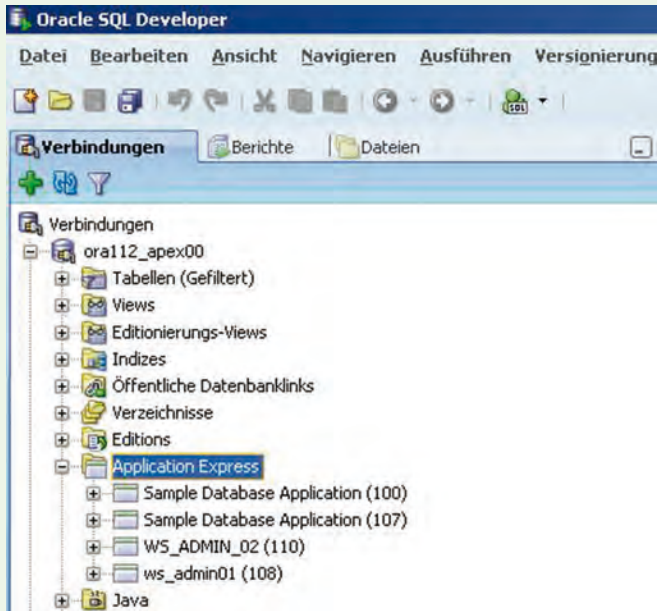


Abb. 6: Anwendungsübersicht nach Import.



Abb. 7: Anwendungsbereitstellung (Schritt 1).

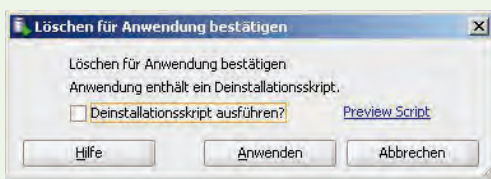


Abb. 8: Deinstallationskriptprüfung.

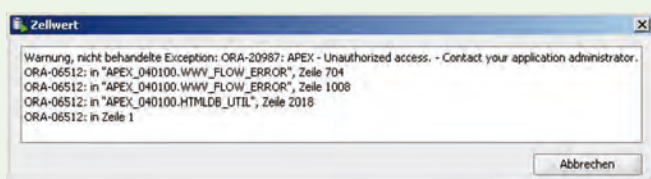


Abb. 9: Exportfehler bei Verwendung APEX Version 4.1.

fügar. Oracle hat hierzu schon den Patch 4.1.1 für APEX angekündigt. Als Alternative wurde die Version APEX 4.0 gewählt, die einwandfrei funktioniert. Der Unterschied zu der beschriebenen Importfunktion besteht in der direkten Auswahl der zu importierenden Anwendung aus der Datenbank heraus und nicht aus einer Importdatei des Dateisystems (siehe Abbildung 7). Ansonsten ist das Vorgehen identisch zur Importfunktion.

## Anwendung löschen

Mit diesem Menüpunkt ist das Löschen einer Anwendung sehr einfach möglich. Hier wird allerdings im Gegensatz zum Löschen aus der APEX-Anwendung heraus nicht mehr geprüft, ob ein Anwender an der Applikation angemeldet ist! Der angemeldete Anwender bekommt die folgende Fehlermeldung bei seiner nächsten Aktion:

### Fehler ERR-7260 Bestimmen des Workspace für die Anwendung (XXX) nicht möglich!

Somit sollte zuvor geprüft werden, ob jemand an der Anwendung eingeloggt ist.

Die Checkbox „Deinstallationskript ausführen“ sollte nicht aktiviert sein (siehe Abbildung 8). Damit werden Objekte auf Datenbankebene gelöscht. Diese können einer weiteren Anwendung über das gemeinsam genutzte Parsing-Schema zugeordnet sein. Somit würden hier Datenbankobjekte gelöscht, die eine weitere Anwendung benötigt.

## Refactoring (Bulk)

Bei „Refactoring“ wird der Quellcode aller anonymen Blöcke analysiert und in Form eines Package ausgegeben. Dies stellt das Refactoring der Anwendung dar, wodurch der Quellcode aus der Anwendung in die Datenbank verlagert werden kann.

## Schnell-DDL

Die unter „Schnell-DDL“ angeführten Optionen

- in Datei speichern
- in Arbeitsblatt speichern
- in Zwischenablage speichern

unterstützen bei der Replikation von Anwendungen auf andere Systeme. Diese Portabilität ist insbesondere bei der Entwicklung und der Übertragung auf verschiedene Testsysteme hilfreich.



Das Exportieren einer Anwendung ist mit dem SQL Developer in der APEX Version 4.1 ohne das Einspielen des aktuellen Patch 4.1.1 aber nicht möglich! Es wird ein Zugriffsfehler generiert (siehe Abbildung 9). Auch hier wurde zu Demonstrationszwecken die Version APEX 4.0 genutzt.

## Fazit

Die Nutzung der aktuellen Versionen von Oracle zur Administration einer APEX-Anwendung ermöglicht ein einfaches Administrieren der Anwendung ohne spezielle Kenntnisse. Die Erweiterung des SQL Developer im APEX-Bereich ist jederzeit durch die Integration weiterer Plugins möglich. Darüber hinaus besteht auch die Möglichkeit über den Bereich „Berichte“ verschiedene Informationen zu der Applikation zu erhalten (siehe Abbildung 10).



*Klaus Günther  
(info@ordix.de).*

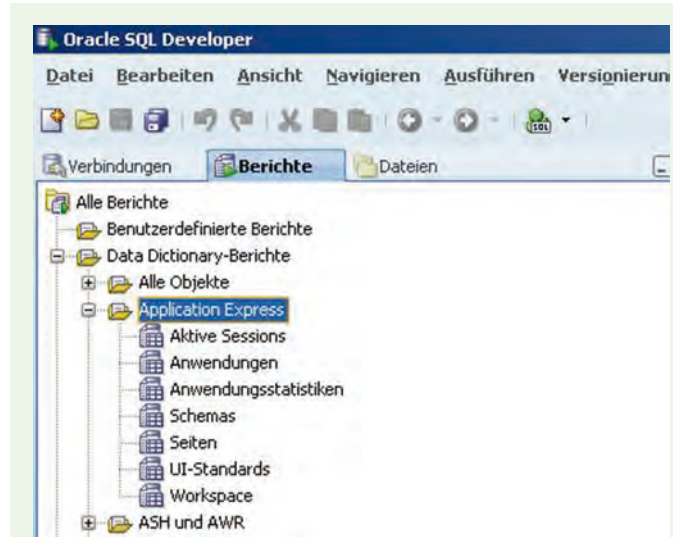


Abb. 10: Berichte im SQL Developer.

## Glossar

### APEX

Oracle Application Express (APEX) – Tool für die Entwicklung von Webapplikationen (ehemals HTML DB).

### APEX Parsing-Schema

Datenbankschema einer APEX-Applikation.

## Seminarempfehlung: Oracle APEX Programmierung

Informationen/Online-Anmeldung: <http://training.ordix.de/siteengine/action/load/kategorie/Datenbanken/nr/1010/index.html>

In diesem Seminar erhalten Sie einen Einblick in die Funktionen von APEX (Oracle Application Express). Oracle Application Express ist eine vollständige, in die Oracle Datenbank integrierte Entwicklungs- und Laufzeitumgebung für Webanwendungen. Mit einem normalen Web Browser werden Anwendungen schnell und einfach entwickelt. In einem Wechsel von Theorie und Praxis lernen Sie, wie Anwendungen aufgebaut, entwickelt und verwaltet werden können. Am Ende des Kurses sind Sie in der Lage selbstständig APEX-Anwendungen zu erstellen.

### Seminarinhalte

- APEX-Architektur
- Websheet Application
- Database Applications
- Benutzerverwaltung
- User-Authentifizierung
- Workspace-Administration
- Anbindung / Nutzung des SQL Developer
- Vertiefung der Theorie durch praktische Übungen und Beispiele

### Termine

- 14.05. - 16.05.2012 in Wiesbaden
- 11.06. - 13.06.2012 in Hannover
- 06.08. - 08.08.2012 in Wiesbaden
- 17.09. - 19.09.2012 in Bielefeld

**Seminar-ID:** DB-ORA-43

**Dauer:** 3 Tage

**Preis pro Teilnehmer:** 1.290,00 € (zzgl. MwSt.)

**Frühbucherpreis:** 1.161,00 € (zzgl. MwSt.)

Projektmanagement in der Praxis: „Wie aus dem richtigen Leben...“ (Teil IV)

# Stakeholder Management - aber richtig!

Ob Scheiterhaufen oder Marterpfahl, für den englischen Begriff „Stake“ gibt es viele Übersetzungen. Im Rahmen des strategischen Managements trifft die Übersetzung risikobehafteter Einsatz aber wohl am besten zu. Im Projektmanagement beschreibt der Begriff Stakeholder Personen bzw. Gruppen, die ein berechtigtes Interesse am Verlauf und dem Ergebnis eines Projektes haben. Doch wie geht man mit den verschiedenen Anspruchsgruppen um? Und wie gestaltet sich ein erfolgreiches Stakeholder Management? Diesen und weiteren Fragen gehen wir in diesem Artikel auf den Grund.

## Was sind eigentlich Stakeholder?

Der Stakeholder-Ansatz geht auf R. Edward Freeman zurück, der 1984 sein Konzept „Strategic Management. A Stakeholder Approach“ [1] als strategisches Managementinstrument veröffentlichte. Dabei orientiert er sich an den zentralen Anspruchsgruppen eines Unternehmens und dem interdisziplinären Zusammenspiel mit dem Management. Dieses muss, unabhängig von den gesetzten Zielen berücksichtigen, wie sich das eigene Handeln auf andere und das Unternehmen in Gänze auswirkt.

Die maßgebliche Idee ist die Gewinnung der Akzeptanz der Unternehmensumwelt, um die für das Unternehmen beste Strategie zu entwickeln.

Der Begriff Stakeholder umfasst dabei alle natürlichen und juristischen Personen, die mit einem Unternehmen in Verbindung stehen. Diese Personen haben ein berechtigtes Interesse an dem Verlauf und den Ergebnissen des Unternehmens. Gemäß der obigen Übersetzung, hat ein Stakeholder einen Einsatz geleistet, an dessen Erhalt er interessiert ist. Die typischen Anspruchsgruppen eines Unternehmens werden in Abbildung 1 dargestellt.

Im Rahmen des Projektmanagements sind die Stakeholder eines Projektes alle Anspruchsgruppen, die direkt oder indirekt an dem Projekt beteiligt bzw. von

dem Projekt betroffen sind. Diese Personen gilt es durch eine Stakeholder-Analyse im Hinblick auf ihr speziellen Projektinteressen zu untersuchen. Die typischen Stakeholder eines Projektes sind in Abbildung 2 dargestellt.

## Erst identifizieren...

Zu Beginn eines jeden Projektes gilt es, die potenziellen Stakeholder zu bestimmen. Dazu sollten folgende Fragen beantwortet werden:

- Wer ist von dem Projekt betroffen?
- Wer hat an dem Projekt ein berechtigtes Interesse?
- Wer gewinnt / verliert, wenn das Projekt erfolgreich ist oder scheitert?
- Wer wird benötigt, damit das Projekt erfolgreich wird?
- Wer kann zusätzliche Unterstützung leisten?
- Welche Beziehungen haben die identifizierten Stakeholder untereinander?

Nach Beantwortung dieser Fragen ist es dem Projektleiter möglich, die direkt oder indirekt am Projekt Beteiligten zu identifizieren und hinsichtlich ihrer speziellen Projektinteressen zu untersuchen.

... dann analysieren

Die Einstellungen der ermittelten Personen zum bzw. ihr Bedeutung für das Projekt können sehr unterschiedlich ausfallen. Daher besteht der nächste Schritt in der systematischen Erfassung der verschiedenen Stakeholder im Hinblick auf folgende Aspekte:

- ihre jeweilige Einstellung zum Projekt
- ihre individuellen Ziele
- ihre potenzielle Bedeutung und ihr Einfluss

Im Rahmen einer solchen Analyse erfolgt daher die präzise Ermittlung der Anspruchsgruppen und ihrer Anforderungen. Eine Hilfestellung kann dabei die Tabelle in Abbildung 3 geben. Eine Stakeholder-Untersuchung im Projekt wird, übrigens auch Projektumfeldanalyse genannt.

In der Praxis ist es ratsam, zu Beginn eines jeden Projektes ein Gespräch mit jedem einzelnen Stakeholder zu führen. Dabei sind die Ziele, Anforderungen und ggf. Bedenken zu ermitteln. Zudem ist eine Vereinbarung über den Informationsfluss, die eigene Rolle, Entscheidungswege und die eigene Prioritäten zu treffen.

Im Detail sind folgende Fragestellungen zu beantworten:

- Welche Ziele, Erwartungen und ggf. Befürchtungen hat die Person an das Projekt?
- Welchen Informationsbedarf hat sie?
- In welcher Tiefe?
- In welchen Zyklen?
- Zu welchem Anlass?
- In welcher Form?
- Wie möchte der jeweilige Stakeholder in Entscheidungen einbezogen werden?
- Welche Prioritäten setzt der Stakeholder?

Auf Grundlage der durchgeführten Stakeholder-Analyse können nun Maßnahmen definiert werden, wie z.B.:

- Kommunikationskonzept für die Stakeholder entwickeln
- Berücksichtigung von individuellen Wünschen bei der Projektdurchführung
- Gezielte Betreuung einzelner Stakeholder planen
- Gewinnung von Promotoren
- Einbindung der Projektbeteiligten bei Grundsatzentscheidungen

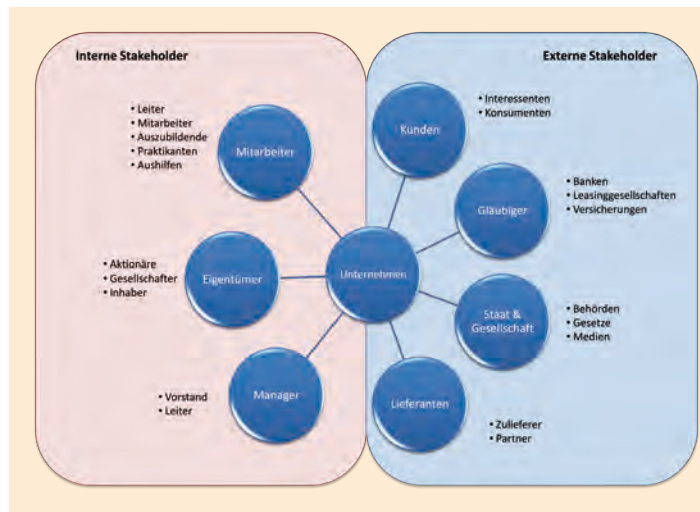


Abb. 1: Typische Anspruchsgruppen eines Unternehmens.



Abb. 2: Typische Anspruchsgruppen eines Projektes.

Person, Interessengruppe	Einstellung zum Projekt (positiv / neutral / negativ)	Projektrelevante Ziele	Bedeutung, Macht, Einfluss	* Erwartungen - Befürchtungen	Maßnahmen, Strategien

Abb. 3: Übersicht zur Stakeholder-Analyse [2].

## Glossar

### Stakeholder

Englischer Begriff für Interessensvertreter. Im Bereich Projektmanagement sind dies alle Personen oder Parteien, die ein berechtigtes Interesse bzw. Anspruch am Verlauf und / oder dem Ergebnis eines Projektes haben, z.B. Auftraggeber, Projektmitarbeiter, zukünftige Nutzer, etc.

## Quellen

- ▶ [1] F. X. Bea / J. Haas, Strategisches Management, S. 113, 5. Auflage, Lucius & Lucius Verlagsgesellschaft mbH, Stuttgart 2009
- ▶ [2] F. X. Bea / S. Scheurer / S. Hesselmann, Projektmanagement, S. 103, 2. Auflage, UVK Verlagsgesellschaft mbH, Konstanz und München 2011

Die Stakeholder-Analyse erfolgt auf einer subjektiven Basis, die durch konkrete Beobachtungen abgesichert wird. Sie ist eine Momentaufnahme, da sich die analysierten Personen und Situationen stetig ändern. Die Stakeholder sollten während des Vorhabens informiert und betreut werden. Sinnvoll ist es außerdem auch kritische Stakeholder direkt mit einzubeziehen (z.B. in das Steering-Committee).

### ... und schließlich managen!

Das Stakeholder Management ist eine fortlaufende Aufgabe. Die Stakeholder können den Erfolg oder Misserfolg eines Projektes maßgeblich direkt oder indirekt beeinflussen. Sie können die Projektdurchführung erschweren oder im positiven Fall vereinfachen.

Oftmals werden Projekten von scheinbar Unbeteiligten wahre Stolpersteine in den Weg gestellt. Auch politische Aspekte und Hintergründe werden häufig nicht offen mitgeteilt.

Das Stakeholder Management hat das Ziel, die relevanten Personen so zu steuern, dass sich ihre Einwirkung auf das Projekt möglichst positiv gestaltet bzw. ein negativer Einfluss unterbleibt, oder mindestens abgeschwächt wird. Dabei sollen die unterschiedlichen Ziele und Erwartungen der Stakeholder möglichst weitgehend unter einen Hut gebracht werden. Dies beinhaltet, dass die Kommunikationsabläufe so gesteuert werden, dass die Bedürfnisse der Projekt-

beteiligten erfüllt und Probleme gemeinsam mit den Stakeholdern gelöst werden können.

Zudem dient es dazu, politische Zusammenhänge besser analysieren und verstehen zu können. Dadurch können die Ursachen von Entscheidungen eher nachvollzogen und die Chancen für den Projekterfolg erhöht werden.

Aus diesen Gründen ist es notwendig, im Anschluss an die Analyse sich ergebende Chancen und Risiken einzuschätzen, mögliche Konsequenzen abzuleiten, Maßnahmen zu identifizieren, einen Handlungsplan zu erstellen und umzusetzen. Abschließend erfolgt die Evaluation der Ergebnisse der Umsetzung.

## Fazit

Ein erfolgreiches Stakeholder Management macht aus Betroffenen Beteiligte. Dabei werden die Stakeholder durch eine, ihren Bedürfnissen entsprechende Kommunikation an dem Projekt beteiligt. Sie wirken direkt an den für sie relevanten Entscheidungen mit. Dabei müssen stets Kompromisse zwischen den unterschiedlichen Interessen der verschiedenen Stakeholder, auf Basis der Projektprioritäten, gefunden werden.

Ein guter Projektleiter sollte sein Projekt genauso steuern wie ein Unternehmer, der sein Unternehmen in der Art und Weise steuert, dass eine gute Balance zwischen den Anforderungen der unterschiedlichen Stakeholder erreicht wird.



*Evelyn Ernst  
(info@ordix.de).*



*Benedikt Georgi  
(info@ordix.de).*

IBM Informix 11.70 (Teil V)

# Der Panther hält Wache

Mit der Version 11.70 hat IBM weitere attraktive Funktionalitäten für einen sicheren Zugriff auf das Datenbanksystem Informix eingeführt. Mit dieser Version wird eine einfache Methode angeboten, Benutzer ohne Betriebssystemkonto zuzulassen und zu verwalten, was bis dato bei Informix unmöglich war.

## Übersicht

Das sogenannte „Usermapping“ bietet dem Datenbankadministrator die Möglichkeit, Benutzer ohne ein Betriebssystemkonto am Datenbanksystem anzumelden. Über eine externe Authentifizierung kann dieser Benutzer dann einem, in der Datenbank bereits definierten Benutzer, zugeordnet werden.

Mit der Definition von „trusted context“ können einfach und ressourcenschonend Endbenutzer eines Anwendungsservers an den Datenbankserver durchgereicht und damit ein differenziertes Berechtigungskonzept aufgebaut werden.

Das Überwachen von Datenbankaktivitäten ist jetzt auch tabellenabhängig möglich, was die Analyse von kritischen Vorgängen erheblich erleichtert.

## Zugeordnete Benutzer

Der Datenbankadministrator kann den Server so konfigurieren, dass auch externen Benutzern der Datenbankzugriff gewährt werden kann. Benutzer müssen sich über eine externe Authentifizierung per PAM (Pluggable Authentication Module) oder SSO (Single Sign on) mit Kerberos anmelden oder über eine interne Authentifizierung eine Verbindung herstellen. Diese können anschließend einem Profil auf Betriebssystemebene zugeordnet werden.

Der Datenbankserver benötigt beim Ausführen einer SQL-Anweisung häufig Berechtigungen auf Betriebssystemebene, z.B. wenn Schreib- oder Lesezugriffe auf eine Datei erforderlich sind. In diesem Fall können dem Benutzer Eigenschaften eines Betriebssystembenutzers zugewiesen werden. Ebenso kann ein externer Benutzer einem nur im Datenbanksystem definierten Benutzer mit seinen Berechtigungen und anderen Eigenschaften zugeordnet werden.

## Zugelassene Ersatzbenutzer

Welche Betriebssystembenutzer und -gruppen als Ersatz für zugeordnete Benutzer zugelassen sind, wird in der Datei `/etc/informix/allowed.surrogates` hinterlegt. Diese Datei kann Benutzer und Benutzer-IDs sowie Gruppen und Gruppen-IDs enthalten, z.B.:

```
USER: user1, user2, 200, 350..400
GROUP: ifx_dbsa, root
```

In diesem Beispiel werden die „konkreten“ Benutzer user1 und user2 sowie die User-IDs 200 bzw. der Bereich der User-IDs von 350 bis 400 und die Gruppen ifx\_dbsa und root zugewiesen werden.

```
CREATE USER user101 WITH PROPERTIES user1;
CREATE USER ext102 WITH PASSWORD ext102 ACCOUNT LOCKED;
GRANT ACCESS TO user102 PROPERTIES USER user2;
GRANT ACCESS TO user103 PROPERTIES UID 200, GROUP (root);
GRANT ACCESS TO user104 PROPERTIES USER user1, HOME
"/home/user2";
GRANT ACCESS TO user105 PROPERTIES USER user1
AUTHORIZATION (dbsa);
```

Abb. 1: Erzeugung interner Nutzer und Zuweisung von Eigenschaften.

Durch eine Erweiterung in der SQL-Syntax ist das Erzeugen von internen Benutzern und das Zuweisen von Eigenschaften möglich. Einige Beispiele zeigt Abbildung 1. Zur Nutzung der Funktionalität der zugeordneten Benutzer muss der Konfigurationsparameter USERMAPPING gesetzt sein. Er kann drei Werte annehmen:

- **OFF**  
Es können sich keine externen Benutzer ohne Betriebssystemkonto anmelden.

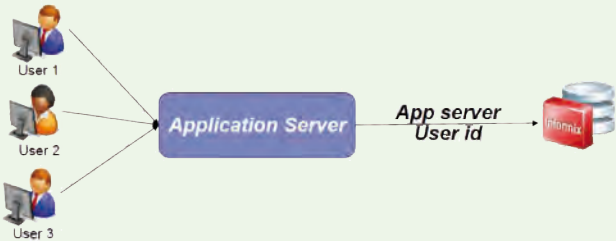


Abb. 2: Konventionelle 3-Schichten-Architektur.

```
CREATE TRUSTED CONTEXT tcx1
  USER jupiter
  ATTRIBUTES (ADDRESS "192.0.2.1",
             ADDRESS "planets.lab.ibm.com")
  DEFAULT ROLE purchase
  ENABLE
  WITH USE FOR user1, user2 WITHOUT
  AUTHENTICATION, user3
  WITH AUTHENTICATION
```

Abb. 3: Definition einer gesicherten Verbindung.

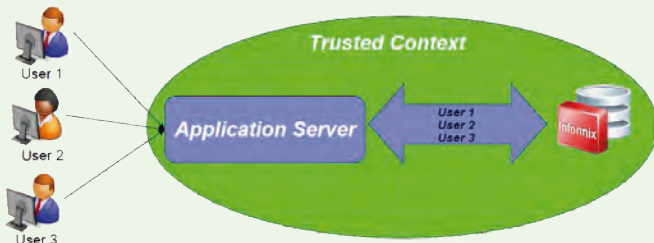


Abb. 4: 3-Schichten-Architektur unter Trusted Context.

- **BASIC**  
Externe Benutzer können sich anmelden, aber keine Operationen privilegierter Benutzer ausführen.
- **ADMIN**  
Externe Benutzer mit entsprechend zugewiesener Berechtigung können auch administrative Operationen (z.B. DBSA) ausführen.

Achtung: Der Wert des Parameters wird beim Erstellen einer Instanz und bei einem Upgrade zunächst auf OFF gesetzt!

Informix legt die zugehörigen Daten in der Datenbank SYSUSER in folgenden Tabellen ab:

- **sysusermap,**
- **syssurrogates**
- **syssurrogategroups**

## Trusted Context

In der traditionellen Konstellation einer 3-schichtigen Architektur (Endbenutzer – Anwendungs – Datenbankserver) werden alle Operationen über einen, dem Anwendungsserver zugeordneten Benutzer getätigt (siehe Abbildung 2). Dieser muss alle Rechte besitzen, die ein beliebiger Endbenutzer benötigt. Eine Differenzierung bzw. Protokollierung von Aktionen des eigentlichen Endbenutzers ist nur dann möglich, wenn der Anwendungsserver in der Lage ist, für jeden Endbenutzer eine eigene Verbindung zum Datenbankserver aufzubauen. Dies bedeutet aber einen erheblichen Aufwand.

Informix bietet jetzt die Möglichkeit, eine gesicherte Verbindung zu definieren. Sie wird durch folgende Eigenschaften beschrieben:

- Die Verbindung wird durch einen bestimmten Benutzer aufgebaut.
- Die Verbindung stammt von einem dedizierten Rechner

Ein Beispiel hierzu zeigt Abbildung 3.

## Benutzerwechsel innerhalb einer Verbindung

Ein Benutzer „jupiter“ kann mittels des Befehls **EXEC SQL CONNECT TO "<datenbasename>" TRUSTED** eine zunächst gesicherte Verbindung aufbauen.

Innerhalb dieser Verbindung kann nun der Benutzer gewechselt werden, sofern dieser im Objekt des Trusted Context explizit als Benutzer aufgeführt oder das Objekt mit der Klausel **WITH USE FOR PUBLIC** definiert wurde. Dies geschieht nun folgendermaßen (siehe Abbildung 4):

```
SET SESSION AUTHORIZATION TO "user1";
SET SESSION AUTHORIZATION TO "user3" USING
"<password>"
```

Auch hier kann zusätzliche noch eine Passwortabfrage erfolgen. Der Endbenutzer wird somit über eine gesicherte Verbindung des Benutzers „jupiter“ an den Datenbankserver durchgereicht. Vor dem Wechsel des Benutzers („jupiter“ auf „user1“ bzw. „user3“) sind alle offenen Transaktionen – falls die Datenbank mit Logging betrieben wird – abzuschließen.

## Selektives Auditing

Das Überwachen von Aktivitäten auf der Datenbank konnte bisher nur generell für eine Datenbank ein- oder ausgeschaltet werden. Dies machte es für einen Administrator schwierig, sicherheitsrelevante Verletzungen von sensiblen Daten zu untersuchen, da alle Veränderungen von Daten auf allen Tabellen protokolliert wurden.

Die Überwachung von Aktivitäten kann jetzt für einzelne Tabellen eingeschaltet werden. Dadurch verringert sich das Aufkommen von Prüfsätzen deutlich und es kann eine Analyse gezielt vorgenommen werden. Die Prüfung weist auch „geswichtete“ Benutzer aus und verhält sich damit transparent gegenüber der Verwendung zugeordneter Benutzer.

## Resümee zur Version 11.70

Dieser Artikel beendet die Reihe über die neue Version von IBM Informix 11.70. Seit Oktober 2010 liegt mittlerweile das dritte Update vor. Die aktuelle Version lautet 11.70xC4. Jedes Update hat weitere Verbesserungen und teilweise erhebliche funktionale Erweiterungen gebracht. Ein Umstieg von 11.50 ist in der Regel völlig problemlos und mit geringem Aufwand möglich.

Insbesondere mit der Ultimate Warehouse Edition hat sich Informix im Bereich Warehousing gegenüber der Konkurrenz einen technologischen Vorteil verschafft. Es wird spannend zu beobachten, ob hiermit wirklich Marktanteile zurückerobert werden können!



Werner Wellmann  
([info@ordix.de](mailto:info@ordix.de)).

## Glossar

### SSO

Single Sign-on – bedeutet, dass ein Benutzer nach einer einmaligen Authentifizierung an einem Arbeitsplatz auf alle Rechner und Dienste, für die er lokal berechtigt (autorisiert) ist, am selben Arbeitsplatz zugreifen kann, ohne sich jedes Mal neu anmelden zu müssen. Wechselt der Benutzer den Arbeitsplatz, wird die Authentifizierung, wie auch die lokale Autorisierung, hinfällig.

### PAM

Pluggable Authentication Modules – Softwarebibliothek, die eine allgemeine Programmierschnittstelle (API) für Authentifizierungsdienste zur Verfügung stellt.

### Kerberos

Ein verteilter Authentifizierungsdienst (Netzwerkprotokoll) für offene und unsichere Computernetze.

## Links

- ▶ [1] ORDIX news Artikel 1/2011  
„Der Panther zeigt die Zähne“:  
[http://www.ordix.de/ORDIXNews/1\\_2011/informix\\_11\\_70.html](http://www.ordix.de/ORDIXNews/1_2011/informix_11_70.html)
- ▶ [1] ORDIX news Artikel 2/2011  
„Der Panther betritt den multidimensionalen Raum“:  
[http://www.ordix.de/ORDIXNews/2\\_2011/informix\\_dwh.html](http://www.ordix.de/ORDIXNews/2_2011/informix_dwh.html)
- ▶ [1] ORDIX news Artikel 3/2011  
„Transaction Survival - der Panther überlebt“:  
[http://www.ordix.de/ORDIXNews/3\\_2011/transaction\\_survival\\_informix.html](http://www.ordix.de/ORDIXNews/3_2011/transaction_survival_informix.html)
- ▶ [4] ORDIX news Artikel 4/2011  
„Der Panther kommt auf leisen Sohlen“:  
[http://www.ordix.de/ORDIXNews/4\\_2011/informix\\_embedding.html](http://www.ordix.de/ORDIXNews/4_2011/informix_embedding.html)



Upgrade der Time Zone Files in Oracle Datenbanken

## Über die Kosten des Lichts...

Der amerikanische Präsident Benjamin Franklin schrieb im Jahre 1784 in einem Brief an das Journal of Paris über die Kosten des Lichts. Dieser Brief wird häufig als Ausgangspunkt für die Einführung der Sommerzeitregelung aufgeführt. Mehr als zweihundert Jahre später müssen sich unzählige Menschen in verschiedenen Berufen mit dieser Regelung auseinandersetzen. Inwiefern diese Regelung die Oracle Datenbankadministratoren unter Ihnen betrifft, klärt dieser Artikel.

### Timestamp with Time Zone

Als Oracle Datenbankadministrator muss man sich prinzipiell lediglich die Frage stellen, ob in den zu betreuenden Datenbanken Daten vom Typ „Timestamp with Time Zone“ verwendet werden. Ist dies der Fall, so muss zumindest eine genaue Analyse vorgenommen werden. Daten dieses Datentyps können prinzipiell in zwei Varianten gespeichert werden. Die eine ist diejenige mit der absoluten Differenz zur Standardzeitzone UTC. Hier könnte z.B. UTC +02:00 der Zeit hinzugefügt werden, um in den Sommermonaten die deutsche Uhrzeit zu speichern.

Bei der zweiten Variante wird die Zeit zuzüglich der Zeitzone z.B. Europe/Berlin gespeichert. Während die

erste Variante nicht interpretierbar ist, kann die zweite durch eine Änderung der Sommerzeitregeln eine Veränderung erfahren.

### Wer ist Arthur David Olson?

In Deutschland gilt die Sommerzeitregelung seit 1980. Danach wird die Uhr am letzten Sonntag im März um eine Stunde vor- und am letzten Sonntag im Oktober wieder zurückgestellt. Zuvor gab es bereits während des ersten und zweiten Weltkrieges verschiedene Sommerzeitregeln in Deutschland. Weltweit gibt es sehr häufig Änderungen an diesen nationalen Regeln. So wurde beispielsweise in den USA im Jahr 2007 die Sommerzeit um drei Wochen verlängert.



Russland hat im letzten Jahr die Zeitumstellung dagegen komplett beendet.

Die Regeln zur Sommerzeitumstellung unterliegen der nationalen Gesetzgebung. Darüber hinaus gibt es keine weltweite Autorität, welche diese Regeln zentral verwaltet. Diese Aufgabe hat stellvertretend die Internet Assigned Numbers Authority (IANA) übernommen [2]. Sie verwaltet diese Regeln in der Time Zone Database. Diese Datenbank wird oft nach einem der Gründer Arthur David Olson auch als Olson-Liste bezeichnet.

Oracle speichert die Zeitzoneneinformationen der Olson-Liste im sogenannten Time Zone File. Diese Datei liegt im Verzeichnis `$ORACLE_HOME/oracore/zoneinfo/`. Sie sollte immer dann aktualisiert werden, wenn die Daten in der Datenbank zukünftigen Regeländerungen unterliegen. Sind beispielsweise ausschließlich Daten mit der Zeitzone Europe/Berlin in der Datenbank enthalten, so ist ein Upgrade zur Zeit nicht zwingend notwendig, da derzeit keine Änderung an der deutschen Sommerzeitregelung geplant ist.

### Time Zone Upgrade

Tritt nun der Fall ein, dass Daten vom Typ „Timestamp with Time Zone“ von aktuellen oder zukünftigen Änderungen betroffen sind, so ist ein Upgrade des Time Zone File dringend zu empfehlen. Das Upgrade besteht dabei aus drei Schritten. Im ersten Schritt muss das Time Zone File zur Verfügung gestellt werden. Hierzu ist der passende Patch vom Oracle Support unter Patches & Updates zu installieren. Im zweiten - optionalen - Schritt kann der Administrator prüfen, wie lange das Upgrade laufen wird und wie viele Datensätze betroffen sind (`dst$affected_tables.row_count`). Die Abbildung 1 zeigt die drei Kommandos für den sogenannten **Prepare**. Die Laufzeit der Prozedur `find_affected_tables` entspricht der Laufzeit für den späteren Aufruf der Prozedur `upgrade_database` und damit in etwa der gesamten Laufzeit für das Upgrade.

Im dritten Schritt wird das Upgrade durchgeführt. Die Kommandos für das Upgrade werden in Abbildung 2 aufgeführt. Die Datenbank muss dabei zweimal gestoppt und gestartet werden. Einmal im exklusiven Modus `migrate` und anschließend im Multi-User-Modus. Während der zweiten Phase kommt es allerdings zu Sperren auf zu ändernde Datensätze, solange die Prozedur `upgrade_database` ausgeführt wird.

Die Prozedur `dbms_dst.begin_upgrade` markiert alle Tabellen, die Spalten vom Typ „Timestamp with Time Zone“ enthalten. Diese Tabellen und deren Status bezüglich des Upgrade sind in `DBA_TSTZ_TABLES` sichtbar. Durch den Aufruf von

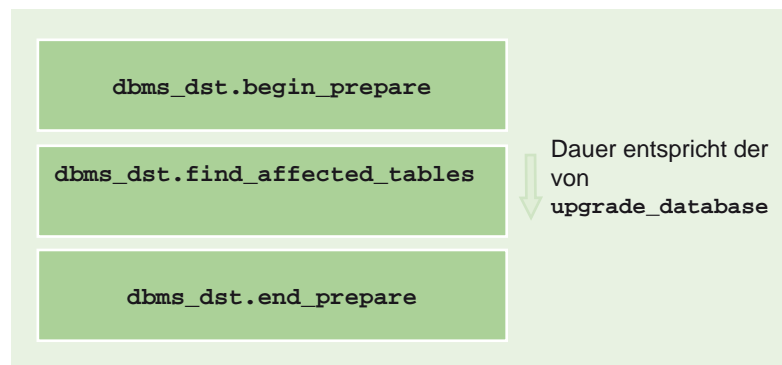


Abb. 1: Prepare Upgrade.

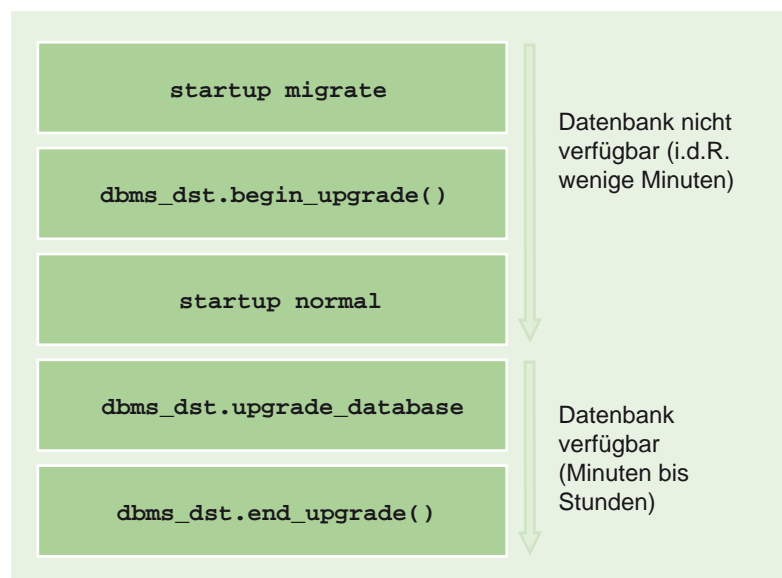


Abb. 2: Upgrade Database.

```
DBMS_DST.UPGRADE_TABLE
( num_of_failures OUT BINARY_INTEGER
, table_list IN VARCHAR2
, upgrade_data IN BOOLEAN := TRUE
, parallel IN BOOLEAN := FALSE
, continue_after_errors IN BOOLEAN := TRUE
, log_errors IN BOOLEAN := FALSE
, log_errors_table IN VARCHAR2 := 'sys.dst$error_table'
, error_on_overlap_time IN BOOLEAN := FALSE
, error_on_nonexisting_time IN BOOLEAN := FALSE
, log_triggers_table IN VARCHAR2 := 'sys.dst$trigger_table'
, atomic_upgrade IN BOOLEAN := FALSE);
```

Abb. 3: DBMS\_DST.UPGRADE\_TABLE.

`dbms_dst.upgrade_database` werden Schritt für Schritt alle Tabellen durchlaufen. Bei diesem Durchlauf wird jeder Zeitstempel geprüft, hierzu wird die gesamte Tabelle mit einem Full Table Scan analysiert.

```
SELECT PROPERTY_NAME, SUBSTR(property_value, 1, 30) value
FROM DATABASE_PROPERTIES
WHERE property_NAME LIKE 'DST_%'
ORDER BY PROPERTY_NAME;
```

PROPERTY_NAME	VALUE
DST_PRIMARY_TT_VERSION	4
DST_SECONDARY_TT_VERSION	0
DST_UPGRADE_STATE	NONE

Abb. 4: DATABASE\_PROPERTIES.

## Glossar

**DST** – Daylight Saving Time, engl. Sommerzeit

**UTC** – Universal Time Coordinated, koordinierte Weltzeit

## Links

- ▶ [1] Benjamin Franklin's essay on daylight saving:  
<http://www.webexhibits.org/daylightsaving/franklin3.html>
- ▶ [2] Webseite der Internet Assigned Numbers Authority:  
<http://www.iana.org/time-zones>
- ▶ [3] Von Oracle referenzierte Olson-Liste:  
<http://www.twinsun.com/tz/tz-link.htm>
- ▶ [4] UPGRADE\_DATABASE Procedure:  
[http://docs.oracle.com/cd/E11882\\_01/appdev.112/e25788/d\\_dst.htm#CIHJEIJC](http://docs.oracle.com/cd/E11882_01/appdev.112/e25788/d_dst.htm#CIHJEIJC)
- ▶ [5] Oracle Support:  
Impact of changes to daylight saving time (DST) rules on the Oracle database [ID 357056.1]  
  
Updated DST transitions and new Time Zones in Oracle Time Zone File patches [ID 412160.1]

**Bild:** © shutterstock.com | Evolution of lighting light | Vasyil Helevachuk

Unterliegt ein Zeitstempel einer Änderung, weil sich die Regeln zur Berechnung der Zeitumstellung geändert haben, so wird dieser Zeitstempel konvertiert. Dies geschieht durch ein Update der entsprechenden Zeile.

## Upgrade von großen Datenbeständen

Ist die zu aktualisierende Tabelle sehr groß oder unterliegen sehr viele Datensätze einer Regeländerung, so kann das Upgrade einer Tabelle entsprechend lange dauern und die zugehörige Transaktion den konkurrierenden Zugriff auf die Sätze verzögern. Oracle bietet mit der Prozedur `dbms_dst.upgrade_table` die Möglichkeit an, einzelne Tabellen parallel zu

verarbeiten (Parameter `parallel=>TRUE`) oder die gesamte Tabelle zu überspringen (Parameter `upgrade_data=>FALSE`). Letzteres sollte natürlich nur gemacht werden, wenn der Administrator sicher ist, dass keine Datensätze aktualisiert werden müssen. Die Abbildung 3 führt alle Parameter dieser Prozedur auf.

## Informationen im Detail

Die Ansicht `DATABASE_PROPERTIES` zeigt die Version des eingesetzten Time Zone File (`DST_PRIMARY_TT_VERSION`), die Version, auf die migriert wird (`DST_SECONDARY_TT_VERSION`) und den Status des Upgrade (`DST_UPGRADE_STATE`). Die Abbildung 4 zeigt exemplarisch die Ausgabe vor der Migration. Die Tabellen `DBA_TSTZ_TABLES` und `DBA_TSTZ_TAB_COLS` enthalten Informationen über den Fortschritt der Migration.

Wer nun der Meinung ist, dass in seiner Datenbank der Datentyp „Timestamp mit Time Zone“ keine Verwendung findet, sollte sich genauer mit dem Paket `dbms_scheduler` beschäftigen. Hier wird beim Kommando `create_job` ein Startdatum von diesem Typ verlangt.

## Ausblick

Beim Upgrade des Time Zone File in großen Umgebungen ist unbedingt darauf zu achten, dass der Oracle Client und die verwendeten Datenbanken jeweils über die selbe Version verfügen. Greift ein Client auf mehrere Datenbanken zu, so führt das zu einem kaskadierenden Problem. Oracle empfiehlt grundsätzlich die verwendete Time Zone File Version auf allen Datenbanken und Clients identisch zu halten.

## Fazit

Wäre Benjamin Franklin heute Datenbankadministrator einer multinationalen hochverfügbaren Datenbank, würde er sicherlich eine etwas andere Kostenrechnung durchführen. Auf jedem Fall würden wir ihm die Teilnahme an unserem Seminar „Oracle Migration und Patching“ empfehlen, bei dem er die hier vorgestellten Verfahren und andere Besonderheiten der Oracle Migration kennen lernen würde.



Martin Hoermann  
([info@ordix.de](mailto:info@ordix.de)).

Oracle GoldenGate – Datenintegration in heterogenen Datenbanklandschaften (Teil II)

## Jetzt im Angebot: Heterogener Initial Load

Als 2009 die GoldenGate Software Inc. aufgekauft und in die Produktmatrix von Oracle eingebettet wurde, wurde nicht nur das Zeitalter der „Zero-Downtime“ beschlossen, sondern auch die Pforten zur Migration von und zu anderen heterogenen Datenbanken aufgestoßen. Diese beiden Themen wollen wir uns nun genauer anschauen.

Vor dem Einsatz von Oracle GoldenGate sollten einige Dinge wohl überlegt sein. Es sollte zum Beispiel die Struktur der Datenbanken sowie der Daten selbst betrachtet werden, um die nötigen Konfigurationsschritte zu planen. Oracle GoldenGate repliziert ganze Datenbankschemata (alle DDL-Befehle) oder einzelne Tabellen (alle DML-Befehle).

Alle in diesem Artikel beschriebenen Vorgehensweisen werden anhand der Replizierung einer Testtabelle dargestellt. Die GoldenGate-Prozesse selbst replizieren ausschließlich sich ändernde Daten. Eine Ausnahme hierzu bildet der Initial Load, der im Folgenden genauer erklärt wird.

### Initial Load

Angenommen, die zu replizierenden Daten befinden sich auf einer produktiven Datenbank (Quelldatenbank). Würde der Extract- sowie Replicat-Prozess eingeschaltet, so werden ausschließlich alle Datenänderungen ab diesem Zeitpunkt übertragen. Was wäre, wenn ein Datensatz geändert werden würde, der als solcher auf dem Zielsystem noch nicht vorhanden ist? Hier taucht ein Problem auf!

Nun gibt es verschiedene Lösungswege um dieses Problem zu beheben. Die Daten können zum Beispiel mit Oracle-Hausmitteln per Export/Import übertragen werden. Ein performanterer Weg insbesondere bei größeren Tabellen, stellt der Oracle Data Pump dar. Beide Varianten können jedoch nur in reinen Oracle-Umgebungen eingesetzt werden.

Sobald es heterogen wird, stoßen beide genannten Werkzeuge an ihre Grenzen. Die Lösung, um diese Grenzen zu umgehen, bringt Oracle GoldenGate in Form des Initial Load mit. Ein Initial Load transferiert

alle Daten einer Tabelle zum Zielsystem. Dies löst das Problem mit der Heterogenität. Der Initial Load, wie alle anderen Prozesse von GoldenGate arbeitet mit Daten in einem einheitlichen datenbankübergreifenden Format. Diese Daten werden in sogenannten Trailfiles gespeichert. Zur Veranschaulichung zeigt Abbildung 1 den zeitlichen Ablauf.

Eine Konfiguration des Initial Load auf dem Quellsystem basiert auf dem Zusammenschluss der Konfigurationen des Extract- und Data-Pump-Prozesses. Ein Zusatz ist der Befehl `SOURCEISTABLE`, der dem Prozess sagt, dass die gesamte Tabelle ausgelesen werden soll statt nur die Datenänderungen (siehe Abbildung 2).

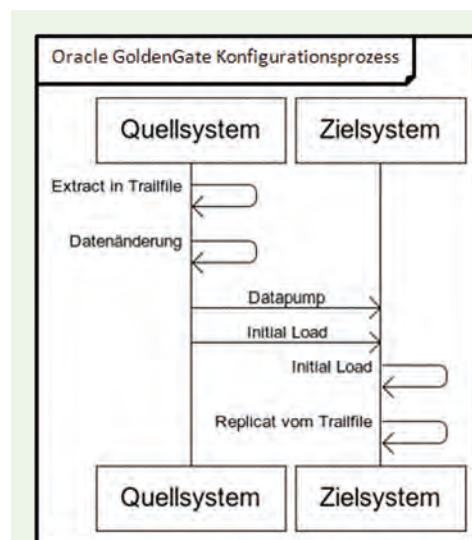


Abb. 1: Oracle GoldenGate – Initial-Load-Prozess.

```
SOURCEISTABLE
SOURCEDB ogg, USERID <benutzername>, PASSWORD <passwort>
RMTHOST <IpDerQuelldatenbank>, MGRPORT <managerPort>
RMTFILE ./dirdat/<nameDerRemoteTrailDatei>, PURGE
TABLE <dbSchema>.<tabelle>;
```

Abb. 2: Quelldatenbankkonfiguration Initial Load.

```
SPECIALRUN
ASSUMETARGETDEFS
USERID <benutzername>@<dbInstance>, PASSWORD <passwort>
EXTFILE ./dirdat/<nameDerLokalenTrailDatei>
DISCARDFILE ./dirrpt/iload.dsc, PURGE
MAP <dbSchema>.*, TARGET <dbSchema>.*;
END RUNTIME
```

Abb. 3: Zieldatenbankkonfiguration Initial Load.

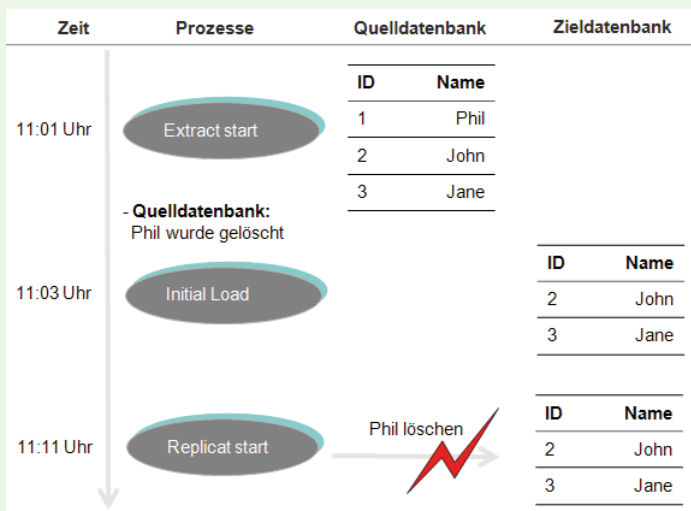


Abb. 4: Initial Load/Replicat-Konflikt.

Bei der Konfiguration des Zielsystems ist zu beachten, dass die Parameter **SPECIALRUN** und **END RUNTIME** enthalten sein müssen. Der Parameter **SPECIALRUN** sorgt dafür, dass der Initial-Load-Prozess als einmaliger Prozess angesehen wird und keine Checkpoints geschrieben werden. Dagegen bedeutet der Parameter **END RUNTIME**, dass - nicht wie beim Replicat-Prozess - auf weitere Datenänderungen gewartet wird. Es wird somit nur einmalig die vorhandene Trail-Datei abgearbeitet. Anschließend wird der Initial-Load-Prozess gestoppt. Eine Beispielkonfiguration ist in Abbildung 3 dargestellt.

Um in der Konfiguration des Initial Load fortzuschreiten, stoßen wir nun auf das Problem, dass alle Datenänderungen im Zeitfenster zwischen dem Starten des Initial Load und des Replicat-Prozesses

verloren gehen. Um auch dies zu lösen, müssen die einzelnen GoldenGate-Prozesse in einer bestimmten Reihenfolge gestartet werden.

Zunächst muss der Extract-Prozess auf dem Quellsystem gestartet werden. Dieser liest alle Datenänderungen aus und schreibt sie in lokale Trailfiles. Anschließend wird der Prozess „Data Pump“ gestartet (nicht zu verwechseln mit dem Oracle Data Pump), sodass die Daten der lokalen Trailfiles auf das Zielsystem als sogenannte Remote Trailfiles transferiert werden. Nachdem diese beiden Prozesse gestartet wurden gehen wir davon aus, dass auf dem Quellsystem weitere Datenänderungen stattgefunden haben.

Folgend wird der Initial Load gestartet, der eine Trail-Datei der gesamten Daten der zu replizierenden Tabelle anlegt und diese direkt auf das Zielsystem transferiert. Die Daten der gesamten Tabelle werden dann in das Zielsystem eingespielt.

Abschließend wird der Replicat-Prozess auf dem Zielsystem gestartet, der alle vom Extract-Prozess-Startpunkt angefallenen Datenänderungen in die Zieldatenbank einspielt.

Wichtig dabei ist, dass die Konfiguration des Replicat-Prozesses den Befehl **handlecollisions** enthält. Wird davon ausgegangen, dass zwischen dem Zeitpunkt des Startens des Extract-Prozesses und dem Starten des Initial Load z.B. ein Datensatz der Tabelle gelöscht wird, so würde Oracle GoldenGate beim Start des Replicat-Prozesses mit einem Fehler beendet (siehe Abbildung 4).

Oracle GoldenGate versucht diesen bestimmten Datensatz zu löschen. Doch auch hier entsteht ein Problem! Der Datensatz ist nicht vorhanden, da er bereits vor dem Start des Initial Load gelöscht und daher gar nicht auf die Zieldatenbank transferiert wurde. Der Befehl **handlecollisions** vermeidet das Beenden des Replikationsprozesses bei dieser Art von Konflikten.

Nach dem Start des Replikationsprozesses sind beide Datenbestände der Datenbanktabellen stimmig. Daraufhin sollte der Befehl **handlecollisions** aus der Konfiguration wieder entfernt werden. Durch diesen Befehl werden eventuell auftretende fehlerhafte Transaktionen nicht mehr in das Discardfile des jeweiligen Prozesses geschrieben. Ein Discardfile ist eine Datei, in die alle abgebrochenen Transaktionen geschrieben werden. Mit ihr kann der GoldenGate-Administrator im Nachhinein die Transaktionen überprüfen bzw. manuell nachpflegen.

Zusammengefasst ermöglicht der Initial Load als solches eine Zero-Downtime-Migration in hetero-

genen Systemlandschaften. Da alle vorhandenen Daten ohne jegliche Ausfallzeit von vorgegebenen Tabellen auf die jeweilige Zieldatenbank transferiert werden können.

## Heterogenität

Oracle GoldenGate ermöglicht die Datensynchronisation in heterogenen Landschaften. Im Folgenden wird die Konfiguration eines unidirektionalen Datentransfers von einer Microsoft SQL-Server-Datenbank zu einer Oracle 11g R2-Datenbank beschrieben.

Im Umfeld von Microsoft SQL Servern muss grundsätzlich die Verbindung zwischen Oracle GoldenGate und der Datenbank per ODBC aufgebaut werden. Dazu muss ein System-DSN-ODBC-Treiber für die vorliegende Microsoft SQL-Server-Datenbank angelegt werden. Der native SQL-Treiber sollte gewählt werden mit aktivierten `ansi_nulls` und `quoted_identifier`, um Fehlern vorzubeugen.

Durch Nutzung des ODBC-Treibers ändert sich die Schreibweise des Datenbank-Login-Parameters von:

```
USERID <benutzername>@<dbInstance>,
PASSWORD <passwort>
```

in diesen betriebssystemunabhängigen ODBC Parameter:

```
SOURCEDB <odbcTreiberName>,
USERID <benutzername> PASSWORD <passwort>
```

Jegliche anderen Konfigurationsparameter von GoldenGate bleiben, wie in der ORDIX news 2/2011 [2] beschrieben, unverändert. Nun besteht eine Verbindung zwischen GoldenGate und der Microsoft SQL-Server-Datenbank. Um die Datenbank für den Extract-Prozess der Daten einzurichten, muss ein Voll-Backup der Datenbank eingerichtet werden. Dies kann über Rechtsklick auf die Datenbank -> Optionen -> Wiederherstellungsmodell geprüft werden (siehe Abbildung 5).

Im Folgenden wird nun eine DDL-Konfiguration anhand einer Testtabelle vorgenommen. Zuerst muss die genannte Testtabelle erstellt werden. Hier ist das Problem, dass je nach verwendeter Datenbank unterschiedliche Datentypen existieren. Es gibt zum Teil kleinere Unterschiede, z.B. gibt es unter einer Oracle-Datenbank den Datentyp `VARCHAR2`, der unter Microsoft SQL-Server-Datenbanken nur `VARCHAR` heißt.

Oracle GoldenGate kann die Daten automatisch in einen anderen Datentyp konvertieren. Sind die Tabellen exakt die gleichen, kann der Parameter `ASSUMETARGETDEFS` in der Konfiguration angegeben

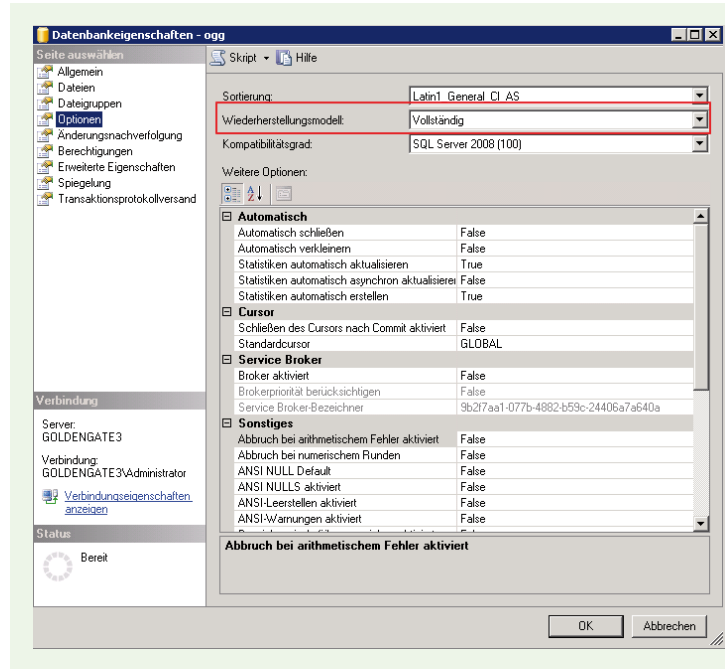


Abb. 5: Microsoft SQL-Server-Datenbank Voll-Backup.

## Quelldatenbank

```
CREATE TABLE emp
( employee_id NUMBER(6)
, first_name VARCHAR2(20)
, last_name VARCHAR2(25)
, hire_date DATE DEFAULT SYSDATE
```

## Zieldatenbank

```
CREATE TABLE emp
( employee_id NUMBER(6)
, last_name VARCHAR2(25)
, first_name VARCHAR2(20)
, hire_date DATE DEFAULT SYSDATE
```

Abb. 6: Data Definition File.

werden. In unserer Artikelkonfiguration benötigen wir eine Defgen-Datei, da die Tabellen nicht identisch sind.

## Data Definition File

Eine Defgen-Datei wird immer dann benötigt, wenn die miteinander zu synchronisierenden Datenbanken nicht exakt die gleichen Tabellendefinitionen aufweisen oder die Daten in andere Tabellenstrukturen abgebildet werden sollen. Auch wenn die Struktur als solches fast identisch ist. Unterschiedlich sind schon geringste Abweichungen, z.B. die Reihenfolge wie

## Verschlüsselung

Im ersten Artikel dieser Artikelreihe [2] wurden einige Sicherheitslücken bemängelt, wie z.B. dass Passwörter im Klartext gespeichert werden. Oracle GoldenGate unterstützt die Verschlüsselung von allen wichtigen Daten. Alle Trailfiles können per 256-Byte-Schlüssel verschlüsselt werden.

Die Datenübertragung der Trailfiles über das TCP-/IP-Netzwerk kann per **BLOWFISH** verschlüsselt werden. Die Passwörter in den Konfigurationsdateien können auch verschlüsselt werden, so dass keine unbefugten Datenzugriffe möglich sind. Weiterführende Informationen zu diesem Thema sind in der GoldenGate Dokumentation [1] zu finden.

## Glossar

### CSN

Commit Sequence Number – Oracle GoldenGate-interne Nummer, die für jede festgeschriebene Transaktion genutzt wird. Diese Nummer basiert auf der Oracle-internen SCN.

### DDL

Data Definition Language – Über DDL-Kommandos werden Datenstrukturen gepflegt (z.B. Tabellen anlegen oder löschen).

### Defgen

Eine Datentypdefinition der Tabellenstruktur der Quelldatenbank. Wird immer dann benötigt, wenn die zu replizierenden Tabellendefinitionen der Quell- und Zieldatenbank nicht exakt übereinstimmen.

### DML

Data Modification Language – Über DML-Kommandos werden Daten (Zeilen) in Tabellen eingefügt, gelöscht oder geändert.

### GGSCI

GoldenGate Software Command Interfaces – Kommandozeilen-Werkzeug zur Administration der Oracle-GoldenGate-Prozesse.

### OGG

Oracle GoldenGate – Oracle Produkt für Datenintegration in heterogenen Datenbanklandschaften.

### SCN

System Change Number – Oracle-interne Nummer, die für jeden internen Befehl inkrementiert wird.

## Links

- ▶ [1] Oracle GoldenGate-Dokumentation:  
[http://docs.oracle.com/cd/E22355\\_01/doc.11111/e21513.pdf](http://docs.oracle.com/cd/E22355_01/doc.11111/e21513.pdf)
- ▶ [2] ORDIX news Artikel 2/2011  
„Oracle GoldenGate - Über eine Brücke musst du gehen!“:  
[http://www.ordix.de/ORDIXNews/2\\_2011/oracle\\_goldengate.html](http://www.ordix.de/ORDIXNews/2_2011/oracle_goldengate.html)

eine Tabelle definiert wurde. Siehe hierzu Abbildung 6, bei der von einer Oracle-Oracle-Replikation ausgegangen wird. Für die angesprochene Replikation einer Microsoft SQL-Server-Datenbank zu einer Oracle-Datenbank wird auf jeden Fall eine Defgen-Datei benötigt, da die Datentypen abweichend sind.

Die Defgen-Datei als solches enthält Tabellename, Spaltenname, Datentyp, Datenlänge und Offset, so dass Daten richtig in die dafür vorgesehenen Datentypen konvertiert werden können. Die Datei muss einmalig vorab per Defgen Utility angelegt und auf dem Zielsystem abgelegt werden (unter `./dirdef/`).

Anschließend wird die Datei dann per Parameter **DEFSSFILE** <pfad zur Defgen-Datei> in die Konfiguration eingepflegt. Die Konfigurationsdatei wird entweder über die GGSCI-Kommandozeile oder manuell als \*.prm-Datei mit folgenden Parametern (im Ordner `./dirprm/`) erstellt:

```
defssfile ./dirdef/<NameDerDefgenDatei>.def
USERID <username>@<dbInstance>, PASSWORD
<password>
TABLE <dbSchema>.<tabelle>;
```

Defgen ist eine Utility und muss auf der Kommandozeile (nicht der GGSCI) mit diesem Kommando ausgeführt werden, um die Defgen-Datei selbst zu erstellen:

```
./defgen paramfile
./dirprm/<nameDerDefgenKonfiguration>.prm
```

Da wir nun die Defgen-Datei erstellt haben, werden alle Datentypen richtig konvertiert. Abschließend können, wie unter Initial Load beschrieben, alle Schritte zum Starten der Replikation abgearbeitet werden.

## Fazit

Die in diesem Artikel beschriebenen Praktiken können für jede unterstützte Datenbank - egal ob heterogen oder nicht - abgewandelt werden. Somit ist auch ein Datentransfer von Oracle zu einer Microsoft SQL-Datenbank ohne weiteres möglich. Oracle GoldenGate stellt ein sehr gutes, durchdachtes und performantes Oracle-Middleware-Produkt dar, mit dem viele Probleme gelöst werden können.

Einen großen Mehrwert bietet die Möglichkeit, einen Datentransfer in heterogenen Landschaften herzustellen. Die Konfiguration kann je nach Einsatzbereich allerdings sehr aufwendig sein.



Stefan Mertens  
([info@ordix.de](mailto:info@ordix.de)).



# „Flink und Flexibel“ - Ein Rückblick auf den ersten Treffpunkt 2012

Auch in diesem Jahr lädt die ORDIX AG wieder zu Veranstaltungen der Reihe „Treffpunkt RheinMainIT“ ein. Am 28.02. fand die erste Veranstaltung mit dem Thema „Flink und Flexibel - Agiles Projektmanagement im IT-Betrieb“ statt.

## Agiles Projektmanagement - Betriebsprojekte erfolgreich steuern

Oftmals erweisen sich Softwareentwicklungsprojekte als zu komplex für eine vollständige Planung. Die Anforderungen an ein System lassen sich im Vorfeld nicht klar definieren. Daher werden zunehmend agile Projektmanagement-Methoden erfolgreich zur Steuerung von Entwicklungsprojekten eingesetzt.

Der inkrementelle und iterative Ansatz des agilen Projektmanagements reduziert die Komplexität des Projektes durch eine regelmäßige Erfassung des Projektfortschritts und die Überprüfung der jeweiligen Funktionalitäten. In seinem Vortrag stellte Lars Eisenblatt, Vorstand der coniatos AG, SCRUM als agile Projektmanagement-Methode vor.

Lars Eisenblatt zeigte dabei auf, dass SCRUM nicht nur in der Softwareentwicklung sondern auch zur Steuerung von Projekten im IT-Betrieb erfolgreich eingesetzt werden kann. Besonders interessant war für die Zuhörer die Anpassung der Methode an die besonderen Bedingungen im IT-Betrieb und die Zusammenarbeit mit dem klassischen Projektcontrolling.

Im Anschluss an den Vortrag entwickelte sich eine rege Diskussion über den praktischen Einsatz der Methode sowie der bisherigen Erfahrungen. Der erste Termin der Veranstaltungsreihe wurde von allen Teilnehmern sehr positiv angenommen.

**Nächster Termin: 22.05.2012, 18.00 Uhr**

## Softwarearchitekturen - Zukünftige Herausforderungen für Architekten und Entwickler

Die rasante Entwicklung mobiler Endgeräte haben Anwendungslandschaften und Nutzerverhalten in jüngster Zeit erheblich verändert. Im Zuge des Booms der sozialen Netzwerke ist die Mobilität und eine hohe Verfügbarkeit für den Anwender besonders wichtig geworden.

Der Vortrag am 22.05. im Seminarzentrum Wiesbaden zeigt, wie sich Softwarearchitekturen verändern und wie Softwarearchitekten und Entwickler damit umgehen können. Der Referent erläutert, warum bestimmte Paradigmen auf den Prüfstand gehören und warum das eine oder andere vermeintliche Hype-Thema durchaus auch langfristig seine Berechtigung hat.



Informieren Sie sich jetzt über die weiteren Themen und Termine und sichern Sie sich Ihre Teilnahme an den Veranstaltungen.

Die Informationen finden Sie unter: [www.rheinmainit.de](http://www.rheinmainit.de).

# Aus der PRAXIS für die PRAXIS

## Unser DATAWAREHOUSE Seminar



In unserem Seminar machen wir Sie mit den grundlegenden Konzepten und Komponenten eines Data Warehouse vertraut. Angefangen von der Anbindung von Quellsystemen, über die Befüllung des Warehouse mit ETL-Tools, bis hin zum Erstellen von Datamarts.

Sie lernen die verschiedenen Architekturansätze und Konzepte kennen und erfahren anhand von Live-Demos deren Anwendung.

#### Termine:

26.03. - 30.03.2012	Wiesbaden
07.05. - 11.05.2012	Bielefeld
25.06. - 29.06.2012	Wiesbaden
24.09. - 28.09.2012	Wiesbaden
22.10. - 26.10.2012	Bielefeld



Melden Sie sich gleich an unter: <http://training.ordix.de>

**ORDIX AG**