

ORDIX news

Das IT-Magazin der ORDIX AG



Database to go

6 | Oracle Database Mobile Server II g

18 | Testmanagement in IT-Projekten: Gut geprüft ist halb abgenommen

37 | Neuheiten von JBoss7: Faster, smarter, smaller

44 | Der neue Bootmanager: Besser booten mit GRUB2?

48 | Container unter Linux: LXC - Virtualisierung auf OS-Ebene



DOAG 2013 - die Oracle Community trifft sich

19. bis 21. Nov. 2013 in Nürnberg

Das ORDIX Team ist mit 4 Vorträgen
und einem Schulungstag dabei !

Anmeldungen unter:

<http://www.doag.org/konferenz/doag/2013/>

Unsere Referenten



Datenbank-Experte,
Trainer und Autor

Martin Hoermann
**snapshot too old -
Eine unerwartete Reise**

20.11.13, Raum 1, 16.00 Uhr



Datenbank-Experte, Trainer,
Autor und Leiter der SIG MySQL

Matthias Jung
**RMAN² - Einführung eines
Oracle-Backup-Konzeptes**

20.11.13, Raum 2, 10.00 Uhr



Datenbank-Experte,
Trainer, Autor und
Oracle-Experte

Klaus Reimers
Der Oracle Optimizer

21.11.13, Raum 19, 11.00 Uhr



Datenbank-Experte,
Trainer und Autor

Veit Jäger
Kerberos und das Oracle

21.11.13, Raum 16, 16.00 Uhr

DOAG Schulungstag 22.11.2013

Der Schulungstag bietet allen Einsteigern, die bislang keine Gelegenheit hatten sich mit dem Recovery Manager (RMAN) auseinanderzusetzen, einen pragmatischen Einstieg in die Welt des Oracle Backup und Recovery.

Referent: Klaus Reimers, ORDIX AG

Wir freuen uns auf Sie bei der
DOAG Konferenz 2013

Unseren Messestand finden Sie auf der 3. Etage, Stand 322.

ORDIX AG

Machtspiele, Machtwechsel und die Macht des Geldes

Paderborn, Oktober 2013

Zum Glück habe ich noch einen Tag gewartet. Mein Editorial war schon komplett fertig und dann macht Larry nicht nur mir sondern auch den Neuseeländern einen gewaltigen Strich durch die Rechnung. Larry Ellison meine ich natürlich, nicht unseren Larry Ratlos.

Und das, obwohl die Amerikaner zunächst ratlos schienen, so wie ihnen die Neuseeländer im wahrsten Sinne des Wortes den Wind aus den Segeln nahmen. Letztendlich triumphiert jedoch trotzdem ein Neuseeländer: Sir Russel Coutts der Teamchef der Amerikaner ist Neuseeländer.

Für mich unglaublich aber wahr und genauso unsinnig ist jedoch der Geldeinsatz: Die Neuseeländer setzten umgerechnet 60 Millionen Euro für die 19 Regatten ein, die Amerikaner (oder genauer der Amerikaner Ellison) 150 Millionen Euro (oder 200 Millionen Dollar). Da fragt sich der durch seltsame Lizenzmethoden und -spiele geplagte Oracle-Anwender schon, wieviel Quadratzentimeter des Flugzeugflügelsegels wohl ihm gehören. Ähnlich wie im Fußball zeigt sich aber leider, dass diejenigen, die am meisten unsinnig Geld verprassen, hinterher doch triumphieren. Schade, das ist kein echter Sport mehr.

Ursprünglich sollte das hier geschriebene vor der Wahl veröffentlicht werden und deshalb auch nichts über die Großkopferten in Berlin enthalten. Jetzt muss aber doch einiges raus. Da habe ich noch vor kurzem geschrieben, dass mancher nicht an Rücktritt denkt, obwohl er besser sollte (Rösler Editorial news 01/2013), jetzt ist er nahezu weggepustet, zusammen mit vielen anderen beispielsweise von den Grünen.

Rücktritt ist also nachwievor IN. Nur nicht bei denen mit Dreck am Stecken, wie unserem Verteidigungsminister, der zwar im Sommer wegtauchen wollte, aber sehr zu seinem Leidwesen tauchten immer wieder neue Geldverschwendungen in seinem Ministerium auf. Diese Summe würde sogar für weit mehr als zwei deutsche Teams beim America's Cup à la Larry reichen.

Und natürlich erkennt man schon am Montag nach der Wahl, wie vollmundig und langlebig Wahlversprechen sind, wenn der (Noch- und Wieder-?) Finanzminister laut über Steuererhöhungen nachdenkt. Gleich kommt die Entschuldigung: Steuererhöhungen ja, aber nur weil die Sozis das wollen. Die hatten bis dahin aber weder gesagt, dass sie über eine Koalition verhandeln wollen und schon gleich gar nicht zu welchen Bedingungen. Peinlich!

Na dann warten wir mal die Machtspielchen ab, die ja auch in Bezug auf die PKW-Maut innerhalb der Union passieren. Eigentlich müssten wir regierungslos bleiben oder noch einige Politiker (Merkel, Seehofer...) von der Bildfläche verschwinden, wenn jeder auf seine Worte und Versprechungen von vor der Wahl beharrt.

Wir läuten jetzt den Herbst ein, natürlich wieder mit viel Themen rund um Oracle (auch wenn Larry ihr Geld gerade im Pazifik versenkt hat): Wir schreiben über den mobile Server, genauso wie über Cloud Control (vielleicht hat Larry wegen seiner Cloud Control gewonnen?) und über neue Features des 12c Servers. Unser passendes Seminar dazu läuft auch demnächst vom Stapel.

Natürlich haben wir auch wieder andere Datenbanken an Bord: Die Reihe zur DB2-Version 10.1 geht mit Teil 4 an den Start. Bei dem Thema NoSQL steht nun schon Teil 5 an, dieses Mal geht es um Neo4j.

Im Umfeld Betriebssysteme lesen Sie dieses Mal was zu Storage, genauer NetApp und Speicher, Speicher, Speicher, sowie zu Virtualisierungsmethoden unter Linux. Linux Container mal zur Abwechslung, die Solaris Container haben wir ja schon oft genug vorgestellt.

Auch der Boss ist neu und zwar der JBoss 7. Da ist vieles um nicht zu sagen alles anders, sagt unser Autor. Ab in die Grube? Nein GRUB ist eher das Gegenteil nämlich der Bootmanager unter Linux und viel besser (als was eigentlich?).

Last but not least haben unsere Projektmanager wieder einiges zusammengetragen, um Ihnen zu helfen, Ihre IT-Projekte erfolgreich zu gestalten - ob mit SCRUM, einem erfolgreichem Testmanagement oder zertifizierten Projektleitern überlasse ich Ihnen.

Machen Sie es gut, der Jahresendspurt 2013 hat begonnen und ich wünsche Ihnen dabei alles Gute. Egal wer die Berliner Machtspielchen gewinnt, in jedem Fall wechselt die Macht und ein Versprechen nach dem anderen wird platzen. Wie stark und wie viele wird sich in den nächsten Wochen herausstellen.

Ihr



Wolfgang Kögler





Oracle Database Mobile Server II:
Database to go

Datenbanken

- 6 **Oracle Database Mobile Server 11g:
Database to go**
In den letzten Jahren sind viele mobile Datenbanken auf den Markt gekommen. Was bietet die mobile Servervariante aus dem Hause Oracle? Wir geben Ihnen die Antworten.

- 15..... **Oracle Cloud Control 12c (Teil VI):
emcli unter Cloud Control - das unterschätzte Tool**
Die Fähigkeiten und Einsatzmöglichkeiten des Enterprise Manager Command Line Interface (emcli) werden häufig unterschätzt. Neben den Einsatzmöglichkeiten gibt Ihnen der Autor auch einen Einblick in die Konfiguration dieses nützlichen Werkzeuges.

- 21..... **NoSQL vs. SQL - Hype oder echte Alternative? (Teil V):
Neo4j - die Graphendatenbank**
Im letzten Teil dieser Reihe stellen wir die Graphendatenbank Neo4j vor, welche sehr häufig Anwendung in sozialen Netzwerken findet.

- 34..... **Neuerungen in der Oracle Database 12c:
Oracle 12c - Eine für alle, alle in einer**
Seit Juni ist das neue Major Release von Oracle auf dem Markt. Wir geben Ihnen einen ersten Überblick über die Neuerungen.

- 41..... **Die neue Version 10.1 von IBM DB2 (Teil IV):
Funktionale Erweiterung im Bereich Monitoring**
Für eine optimale Leistung und den störungsfreien Betrieb einer Datenbank ist ein Monitoring unerlässlich. Mit der Version 10 gibt IBM dem Administrator weitere Monitoring-Instrumente an die Hand.



Testmanagement in IT-Projekten:
Gut geprüft ist halb abgenommen

Betriebssysteme und Monitoring

- 28..... **NetApp und Data ONTAP 8.x:
Drei Dinge braucht das Rechenzentrum:
Speicher, Speicher, Speicher ...**
Die Anforderungen an das Speichermanagement eines Rechenzentrums werden immer höher. Welche Möglichkeiten bietet das Betriebssystem ONTAP, um diesen Anforderungen gerecht zu werden? Diese und mehr Antworten geben wir Ihnen.

- 48..... **Container unter Linux:
LXC – Virtualisierung auf OS-Ebene**
Die technischen Grundlagen, welche Einsatzmöglichkeiten und Verbesserungspotentiale die Linux Container bieten, zeigen wir Ihnen in diesem Artikel.

Java/JEE

- 37..... **Neuheiten von JBoss 7:
Faster, smarter, smaller**
Red Hat hat mit der neuen Version des JBoss nicht nur eine Weiterentwicklung vorgenommen, sondern ihn quasi komplett neu implementiert. Unser Java-Experte stellt Ihnen die wesentlichen Neuerungen vor.

Open Source

- 44..... **Der neue Bootmanager:
Besser booten mit GRUB2?**
Seit knapp einem Jahr ist der GRUB2 Bootloader verfügbar und inzwischen ist er in den meisten Linux-Distributionen zum Standard geworden. Wir nehmen dieses Werkzeug für Sie genauer unter die Lupe.



Der neue Bootmanager:
Besser booten mit GRUB2?



Container unter Linux:
LXC - Virtualisierung auf OS-Ebene

Projektmanagement

- 10..... Scrum - aber richtig! (Teil II):
Kunde und Product Owner**
Eine wesentliche Herausforderung in Projekten ist das Zusammenwirken von Kunde und dem Product Owner. Wir zeigen auf, wie man für beide Seiten zu einem guten Ergebnis gelangt.
- 18..... Testmanagement in IT-Projekten:
Gut geprüft ist halb abgenommen**
Die Phasen der Testdurchführung und den Aspekten, die der Tester dabei beachten sollte, stehen im Fokus. Neben der Durchführung der Testfälle werden auch die einzelnen Testarten und das Reporting erläutert.
- 30..... Projektmanagement-Methoden:
Wie nützlich sind Zertifizierungen?**
In IT-Projekten treten regelmäßig dieselben Projektmanagement-Fehler auf. Können Zertifizierungen helfen solche Fehler zu vermeiden? Wir geben Ihnen die Antwort und stellen dazu drei etablierte PM-Zertifizierungen vor.

Aktuell | Standards

- 25..... Larry Ratlos**
26..... Seminarübersicht:
Oktober bis Dezember 2013

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 | Westernmuer 12 - 16 | 33098 Paderborn
Tel.: 05251 1063-0 | Fax: 0180 1673490
Gestaltung/Layout: Jens Pothmann
- Auflage:** 7.200 Exemplare
Druck: Druckerei Bösmann, Detmold
- Bildnachweis:** © stockvault.net | Toy car | homero chapa
© sxc.hu | White clouds in blue sky | takatuka
© sxc.hu | container | g-point
© flickr.com | como Kaffeebecher 2 | como coffee and more
© pixabay.com | Test Testing Optical Magnifier Search
© lanrentuku.com
© freepsdfiles.net | PSD Shopping Bag Set
- Autoren:** Peter Berend, Bernhard Burkhard, Marius Dörlöchter, Christian Fertsch, Kathrin Hammerschmidt, Wolfgang Kögler, Philipp Loer, Sven Müller, Dominic Oberländer, Andreas Schlicht, Thomas Schwarz, Michael Thieme, Alexander Zeller
- 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.
- 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.



Oracle Database Mobile Server 11g

Database to go

Das Internet ist heute dank UMTS, LTE und EDGE nahezu überall verfügbar. Der Zugriff auf wichtige Unternehmensdaten ist dadurch jederzeit und überall möglich. Dennoch etablieren sich seit einigen Jahren mobile Datenbanken. Im Rahmen dieses Artikels stellen wir vor, wie der Oracle Database Mobile Server den Datenabgleich mit den Datenbanken der Client-Geräte durchführt.

Geschichte des Oracle Database Mobile Server

Die erste Version des Oracle Database Mobile Server erschien 2002 unter dem damaligen Namen Oracle Lite 9i. Seit März 2011 ist die aktuellste Version unter dem neuen Namen Oracle Database Mobile Server 11g verfügbar.

Hauptgrund für die Namensänderung dürfte sein, dass mit dieser neuen Version die Oracle-Lite-Datenbank des Client durch die Oracle Berkeley DB ersetzt wurde.

Architekturmodell

Die Architektur des Gesamtsystems besteht aus drei Teilkomponenten (siehe Abbildung 1):

- Oracle-Master-Datenbank
- Mobile Server
- Mobile Client

Während bei anderen Herstellern mobiler Datenbanksysteme (IBM, Sybase) durchaus Konkurrenzprodukte als Master-Datenbank verwendet werden dürfen, muss für die Master-Datenbank des Oracle Database Mobile Server zwingend eine Oracle-Datenbank der Standard- oder Enterprise-Edition verwendet werden.

Der Oracle Database Mobile Server organisiert den Datenabgleich zwischen der Client- und Master-Datenbank. Dieser Vorgang wird im weiteren Verlauf näher beschrieben.

Mobile Client

Das Gegenstück zum Oracle Database Mobile Server bildet der Mobile Client, welcher wiederum aus drei Komponenten besteht:

- Client-Datenbank
- Device Management Agent
- Sync Engine

Als Datenbank auf der Client-Seite kann zwischen SQLite und der Oracle Berkeley DB gewählt werden. SQLite ist ein eingebettetes, relationales Datenbanksystem, das den SQL-92-Standard weitestgehend unterstützt. Die Oracle Berkeley DB ist ebenfalls ein eingebettetes Datenbanksystem, sie ist jedoch im Kern ein Key-Value Store. Mit dem Erscheinen der Version 11g wurde zusätzlich eine SQLite-kompatible SQL-API integriert.

Der Device Management Agent erfüllt zwei Aufgaben: Seine primäre Aufgabe ist es, auf Anforderung der mobilen Applikation oder des Oracle Database Mobile Server, den Datenabgleich zu starten. Als zweite Aufgabe kann alternativ ein Datenabgleich automatisiert ausgelöst werden, wenn der Device Management Agent den Mobile Server nach einer Verbindungsunterbrechung wieder erreichen kann.

Die Durchführung des Datenabgleichs obliegt der Sync Engine. Diese führt alle den Client betreffenden Transaktionen der Master-Datenbank auf der Datenbank des Client durch und übermittelt dem Mobile Server die auf dem Client während der letzten Offline-Phase durchgeführten Transaktionen.

Publikationen

Um eine Datensynchronisation zwischen der Client- und der Master-Datenbank durchzuführen, muss zunächst festgelegt werden, welche Daten zwischen den beiden Datenbanken abgeglichen werden sollen. Dies erfolgt über die Anlage von Publikationen. Eine Publikation besteht dabei aus Publikationselementen, Sequenzen und Skripten.

In einem ersten Schritt müssen die Publikationselemente definiert werden. Ein Publikationselement ist im Wesentlichen mit einer View vergleichbar. Der Unterschied ist, dass eine View ein Datenbankobjekt ist. Ein Publikationselement hingegen ist lediglich eine in einer Tabelle im Datenbankschema des Mobile Server gespeicherte Abfrage. Darüber hinaus können bei der Definition eines Publikationselements Bind-Variablen verwendet werden. Durch ihre Verwendung kann die Menge der zwischen einem Client und dem Server zu synchronisierenden Daten parametrisiert werden. Dadurch könnte beispielsweise

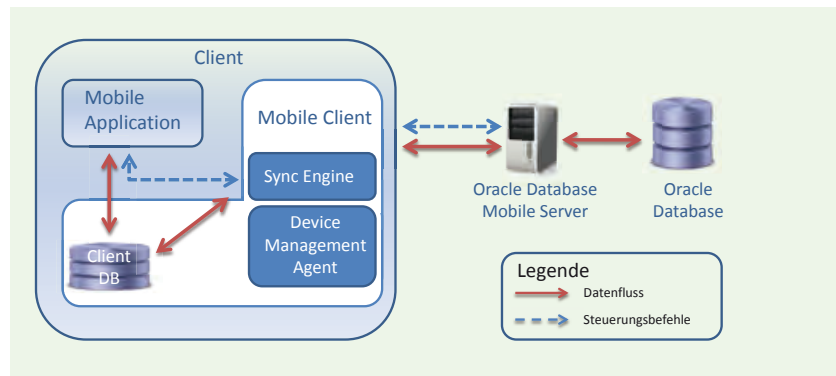


Abb. 1: Architekturmodell des Oracle Database Mobile Server.

ein Vertriebsmitarbeiter nur die Kundendaten für seinen individuellen Vertriebsbereich erhalten. Dies reduziert sowohl den auf dem Client benötigten Speicherplatz, als auch die Menge der bei einem Replikationsvorgang zu übertragenden Daten.

Bei der Anlage einer Sequenz, die auch parallel von den Clients während einer Offline-Phase genutzt werden kann, wird in der Master-Datenbank eine Sequenz mit einer hohen Schrittweite (Inkrement) erstellt. Jedem Client wird nun der Bereich zwischen zwei Schritten für die Erstellung einer eigenen Sequenz zugewiesen. So ist sichergestellt, dass trotz paralleler Verwendung niemals ein Wert doppelt vergeben werden kann. Zusätzlich kann ein Threshold definiert werden: Wird z.B. eine Sequenz in der Master-Datenbank mit einem Inkrement von 100 und einem Threshold von 25 erzeugt, wird im Client eine Sequenz mit 100 Werten angelegt. Sind nur noch 25 oder weniger Werte aus dieser Sequenz in einem Client verfügbar, so erhält der Client bei dem nächsten Datenabgleich einen neuen Sequenzabschnitt aus der Master-Datenbank.

Zusätzlich zu den Publikationselementen können auch Skripte erstellt werden. Diese in Java geschriebenen Skripte ermöglichen es, Änderungen an dem Tabellenmodell und den Daten des Client durch Java Stored Procedures durchzuführen.

Datensatzversionierung

Der Abgleich der Daten zwischen Client- und Master-Datenbank wird über Trigger und Logging-Tabellen sichergestellt.

Für jede Tabelle der Master-Datenbank (im Folgenden als Originaltabelle bezeichnet), die offline durch den Client geändert oder gelesen werden soll, werden ein DML-Trigger und eine Logging-Tabelle erstellt. Die Logging-Tabelle enthält zu jedem Datensatz in der Originaltabelle einen Versionszähler, der bei jeder

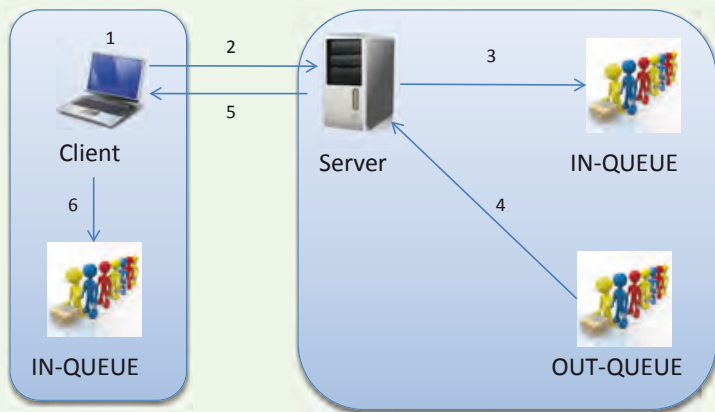


Abb. 2: Datenabgleich zwischen Client und Server.

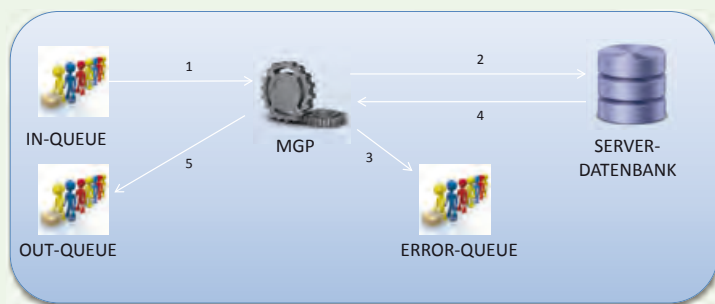


Abb. 3: Datenverarbeitung durch den Message Generator and Processor (MGP).

Änderung des betreffenden Datensatzes inkrementiert wird. Sowohl Trigger als auch Logging-Tabelle befinden sich im selben Datenbankschema wie die Original-Tabelle. Das Gleiche gilt für die in der Client-Datenbank vorgehaltenen Tabellen. Hier existiert ebenfalls eine Logging-Tabelle und ein entsprechender DML-Trigger.

Darüber hinaus gibt es in der Master-Datenbank für jede Originaltabelle eine Mapping-Tabelle. In dieser Tabelle wird vermerkt, welche Version eines Datensatzes der Originaltabelle der Client bei dem letzten Datenabgleich erhalten hat.

Auf diese Weise kann der Mobile Server ermitteln, welche Datensätze bei einem erneuten Datenabgleich an einen bestimmten Client zu übertragen sind. Des Weiteren wird bei dieser Art der Versionierung eines Datensatzes festgestellt, ob der Client die Veränderung eines Datensatzes auf einem veralteten Datenbestand durchgeführt hat.

Datenabgleich

Der Datenabgleich zwischen Client und Server besteht aus zwei Phasen. In der ersten Phase (siehe Abbildung 2) wird zunächst auf beiden Seiten ermittelt, welche Datensätze sich seit dem letzten Abgleich geändert haben. In den Versionierungstabellen wird lediglich festgehalten, welcher Datensatz sich geändert hat und von welcher Art das DML-Statement war (**Insert**, **Update** und **Delete**).

Daher muss nicht nur bei einem **Insert**, sondern auch bei einem **Update**-Statement der gesamte Datensatz übertragen werden. Bei einem **Delete**-Statement wird lediglich ein Löschkennzeichen übermittelt.

Die übertragenen Datensätze werden auf der Seite des jeweiligen Empfängers zunächst in einer Tabelle (In-Queue) gesammelt und nicht direkt in die jeweilige Zieltabelle übertragen. Dies ermöglicht es, einen geänderten Datensatz trotz einer eventuellen Sperre in die jeweiligen Zieltabelle zu übertragen und so den Datenabgleich zu beschleunigen.

Nach Abschluss der Datenübertragung (Phase 1 des Datenabgleichs) wird durch den Message Generator and Processor (MGP), einer Komponente des Mobile Server, die Änderung der vom Client geänderten Datensätze in der Master-Datenbank vorgenommen (Phase 2, siehe Abbildung 3). Dafür generiert der MGP für jeden einzelnen Datensatz aus der In-Queue ein DML-Statement.

Schlägt die Ausführung des Statements, z.B. durch eine eventuelle Constraint-Verletzung in der Master-Datenbank fehl oder stellt der MGP fest, dass der Client die Datenänderung auf einem veralteten

Glossar

Bind-Variablen

Die Verwendung von Bind-Variablen ermöglicht die Wiederverwendung von Ausführungsplänen für SQL-Statements, bei denen sich nur die Werte der Variablen unterscheiden.

Inkrement

Ein Inkrement ist eine elementare Operation, bei der eine schrittweise Erhöhung einer Größe/Variablen erfolgt.

Java Stored Procedures

Sie ermöglichen es, in Prozeduren statt der üblichen Sprache PL/SQL Java zu verwenden.

Key-Value Store

Ein Key-Value Store ist eine Variante einer NoSQL-Datenbank. Hauptmerkmal eines Key-Value Store ist es, dass einige Spalten eines Datensatzes den Schlüssel (Key) bilden und andere den Wert (Value). Soll ein Datensatz abgefragt werden, so kann nur nach den Schlüsselspalten gesucht werden.

Threshold

(engl. Grenzwert) - Überschreiten die Werte einer Client-Sequenz den Grenzwert, wird dem Client eine neue Sequenz zugewiesen.

Datenbestand durchgeführt hat, so wird der betreffende Datensatz in die Error-Queue verschoben.

Die Abarbeitung der Error-Queue kann entweder automatisiert durch PL/SQL-Prozeduren oder manuell über die Administrationsoberfläche des Mobile Server erfolgen. Auf der Client-Seite wird diese Aufgabe des MGP durch die Sync Engine übernommen.

Fazit

Der Oracle Database Mobile Server ermöglicht es, Daten auch in infrastrukturell unterversorgten Gebieten zur Verfügung zu stellen. Problematisch ist jedoch, dass die Daten in den Client-Datenbanken nicht vor einem unbefugten Zugriff geschützt werden. Auch der Aufwand für die Einrichtung des Datenabgleichs und die Fehlerbehandlung ist nicht unerheblich.

Durch die verbreitete Verfügbarkeit von UMTS und LTE dürfte der Einsatz des Oracle Database Mobile Server nur unter sehr begrenzten Umständen einem mobilen Zugriff auf eine zentrale Datenbank vorzuziehen sein.

Links

- ▶ [1] Oracle Database Mobile Server 11g:
http://docs.oracle.com/cd/E22663_01/index.htm
- ▶ [2] Oracle Berkeley DB 12c:
http://docs.oracle.com/cd/E17076_03/html/index.html
- ▶ [3] SQLite:
<http://www.sqlite.org/docs.html>

Bildnachweis

© flickr.com | como Kaffeebecher 2 | como coffee and more



*Philipp Loer
(info@ordix.de)*

Sie mögen komplexe Aufgaben?

Sie suchen eine neue Herausforderung?

Werden Sie Teil unseres TEAMS als
Oracle Consultant (m/w)

Dies erwartet Sie:

- über 100 Kollegen an fünf ORDIX Standorten
- flache Hierarchien
- breites Aufgabenfeld
- kontinuierliche Weiterbildung
- attraktive Vergütung und Firmen-PKW

Scrum - aber richtig! (Teil II)

Kunde und Product Owner

Nachdem im vorangegangenen Teil dieser Reihe [1] die inneren Faktoren für einen optimalen Einsatz von Scrum behandelt wurden, widmet sich dieser Artikel nun dem Zusammenwirken zwischen Kunde und Product Owner. Es geht vor allem um die Schnittstelle zwischen der Anforderungs- und der Entwicklungsseite, also die Frage, inwieweit die Kundenerwartungen erfüllt werden, die maßgeblich für den Erfolg eines Projektes verantwortlich sind. Um hier ein beiderseitig zufriedenstellendes Ergebnis erzielen zu können, sollten die Anforderungen zum Schnüren eines passenden Paketes über den gesamten Projektverlauf hinweg erfasst und angepasst werden, wobei die Qualität, sowie die Kontrolle des Fortschritts parallel zu berücksichtigen sind.

Die erste Herausforderung: Passende Anforderungen sammeln

Zu Projektbeginn setzen sich der Kunde und der Product Owner zusammen, um gemeinsam die Anforderungen für das Produkt zu erarbeiten. Aufgrund der Komplexität heutiger Softwarevorhaben, sind Anforderungen zum Start eines Projektes jedoch nicht in vollem Umfang bekannt, da sie teilweise erst im Verlauf des Projektes entstehen. Ferner ist es üblich, dass sich bereits erfasste Anforderungen im Rahmen der empirischen Prozesssteuerung im Laufe der Zeit verändern.

Zu Beginn ist es jedoch wichtig, ein Gefühl für die Größe des Projektes zu bekommen, damit eine grobe Vorstellung über den Nutzen und die Projektkosten entstehen kann. Dieses Set an Anforderungen dient als Ausgangspunkt für einen sich stetig anpassenden Entwicklungsprozess. Anforderungen sind also keine starren Entitäten, die nur darauf warten, aufgelesen und archiviert zu werden. Die Herausforderung ist es, möglichst flächendeckend alle Anforderungen zu sammeln, die sich zum gegebenen Zeitpunkt ermitteln lassen und sie anschließend zu pflegen.

Lösung: Kommunikation mit dem Kunden

Kommunikation ist hierbei einer der wichtigsten Erfolgsfaktoren. Der Product Owner muss den Kunden in seiner Domäne abholen und für einen ausgeglichenen Kommunikationsfluss zwischen ihm und dem Team

sorgen. Um festzustellen, welche Anforderungen existieren, müssen alle Beteiligten zunächst die gleiche Sprache sprechen. Gerade der Dialog bewirkt dabei, dass die Anforderungen eindeutig und transparent werden. Die Ziele des Kunden lassen sich so besser hinterfragen, um die Notwendigkeit der Anforderung zu klären und neue Erkenntnisse auf beiden Seiten zu gewinnen. Der einseitige Kommunikationsweg ist nach Möglichkeit zu vermeiden, da sonst das Risiko von Missverständnissen zu hoch ist. Ob eine Anforderung wirklich „im Sinne des Erfinders“ ist oder sich möglicherweise noch mal verändert bzw. komplett erübrigt, lässt sich am schnellsten im Dialog klären.

Im Anforderungsworkshop setzen sich Kunde und Product Owner deshalb mit dem Team zusammen, um zunächst ein gemeinsames Ziel in Form einer Produktvision auf Basis von Zielgruppe, Kundenbedürfnissen, Produkthanforderungen und USP zu formulieren. Die Produktvision dient später allen Projektbeteiligten als Orientierungshilfe und gibt die Richtung vor, in die sich das Projekt bewegen soll. In ihrer Formulierung ist sie leicht verständlich und prägnant gehalten. Im Anschluss werden möglichst viele Anforderungen mit einem hohen Anwendernutzen ermittelt. Dazu lassen sich verschiedene Benutzerrollen definieren, welche die Anwender des Systems repräsentieren. Diese zusätzlichen Perspektiven erleichtern neben dem Ermitteln von Anforderungen

auch das Verständnis für die jeweils gewünschte Funktionalität.

Es gibt noch weitere Methoden zur Anforderungsermittlung, wie Interviews, die Beobachtung des Anwenders am System oder Fragebögen. Jede der Methoden kann zu nützlichen Informationen führen, den Vorrang sollten jedoch immer Vorgehensweisen mit der Möglichkeit zur beidseitigen Kommunikation haben.

Der Kunde muss sich stets bewusst sein, dass es einer dauerhaften Zusammenarbeit mit dem Product Owner bedarf, um auf die sich ändernden Gegebenheiten positiv einzuwirken. Es ist Aufgabe des Product Owner, dies dem Kunden bewusst zu machen. Er stellt sicher, dass hinsichtlich des Geschäftswerts über die gesamte Projektdauer hinweg an der Zielerreichung gearbeitet wird.

Die zweite Herausforderung:

Das Paket – Lieferzeitpunkt und Umfang bestimmen

Das Ziel ist es, mit Hilfe von Scrum eine früh auslieferbare Software in hoher Qualität zu entwickeln und dabei einen möglichst hohen Grad an Kundenzufriedenheit zu erreichen. Dem sind allerdings natürliche Grenzen gesetzt und der Product Owner hat die Aufgabe, die Erwartungen in einem realistischen Rahmen zu halten. Da Umfang, Termin und Ressourcen die Qualität eines Produktes bestimmen, müssen Kunde und Product Owner die gesammelten Anforderungen betrachten und die Bedingungen dazu festlegen.

Lösung: Ein Release planen

Ein iterativ gesteuerter Prozess erlaubt zwar die Festlegung eines bestimmten Datums, der bis dahin erzielte inhaltliche Umfang lässt sich aufgrund der Agilität allerdings nicht genau vorhersagen. Fest steht, dass das Produkt den bis dahin größtmöglichen Geschäftswert erzielen soll. Dazu müssen jedoch zunächst die Anforderungen mit dem Kunden priorisiert werden. Eine gute Methode für die Priorisierung von Anforderungen bietet das MoSCoW-Prinzip (Stapleton, 2003). MoSCoW ist eine Abkürzung für Anforderungen, die sich nach folgendem Prinzip ordnen lassen:

- Must: unverzichtbar
- Should: wichtig, aber nicht unverzichtbar
- Could: nicht besonders wichtig, notfalls verzichtbar
- Won't: gerne, aber zu einem späteren Zeitpunkt

Der Kunde und der Product Owner besprechen bei der Priorisierung der Anforderungen deren jeweiligen Nutzen. Das Team bewertet die Anforderungen außerdem aus der technischen Perspektive hinsichtlich Risiko

und Machbarkeit. Am Ende bestimmt jedoch der Kunde die festzulegende Reihenfolge.

Kunde und Product Owner ermitteln zudem eine passende Sprintlänge. Kürzere Zyklen erlauben es zwar, häufiger Kurskorrekturen vorzunehmen, erzeugen aber insgesamt auch einen höheren Planungsaufwand als längere Zyklen. Insgesamt darf die Sprintdauer nur so lang gewählt werden, wie sie für alle Projektbeteiligten überschaubar bleibt. Außerdem sollte sie über den gesamten Projektzeitraum hinweg konstant bleiben. Ein regelmäßiger Rhythmus verringert den Overhead und erhöht die Wahrscheinlichkeit der zeitgerechten Fertigstellung eines Inkrements.

Um die Velocity, also den Gesamtwert aller pro Sprint umsetzbaren User Stories, bestimmen zu können, sollte man möglichst auf historische Werte zurückgreifen. Alternativ kann ein erster Sprint durchlaufen und dessen Ergebnis herangezogen werden oder die Velocity wird eingangs direkt geschätzt, bis später erste Daten vorliegen. Um die Velocity sofort schätzen zu können, ermitteln Sie die im ersten Teil dieser Reihe [1] vorgestellten Nettoarbeitstage. Über die Teamgröße, lässt sich errechnen, wie viele ideale Arbeitstage in einem Sprint geleistet werden können. Skaliert auf das ebenfalls geschätzte Gesamtanforderungspaket für das Projekt, die sogenannte Projekt-Velocity, lässt sich schließlich die Anzahl der benötigten Sprints und damit das Release planen.

Die dritte Herausforderung:

Anforderungen managen

Eine der Leitlinien des Agilen Manifests, auf dem Scrum aufbaut, besagt, dass funktionsfähigen Produkten ein höherer Stellenwert beigemessen wird als einer ausgedehnten Dokumentation. Trotzdem ist auch die schriftliche Fixierung von Anforderungen erforderlich. Durch sie werden alle Erinnerungstützen für weiterführende Dialoge, Zufriedenheitsbedingungen, Akzeptanzkriterien etc. festgehalten. Die Herausforderung besteht darin, neben den Prioritäten, die passende Granularität und einen adäquaten Umgang für nachträglich auftretende und sich ändernde Anforderungen zu finden.

Lösung: Der Product Owner und sein Product Backlog

Für das Festhalten aller Anforderungen eines Projektes gibt es in Scrum das Artefakt des Product Backlog. Der Verwalter des Product Backlog ist der Product Owner.

Durch den Verzicht auf eine lange Vorbereitungsphase werden viele Funktionen zunächst sehr grobgranular oder auch episch beschrieben. Mit fortschreitendem Projekt werden diese Epics nach und nach verfeinert.

- **I** ndependent User Stories sollten möglichst voneinander unabhängig geschrieben sein.
- **N**egotiable Sie sollten diskutier- und anpassbar gehalten sein.
- **V**aluable Sie sollten werthaltig für den Kunden oder Anwender sein.
- **E**stimatable Sie sollten abschätzbar durch das Team sein.
- **S**mall Sie sollten klein genug sein, um innerhalb eines Sprints umgesetzt werden zu können.
- **T**estable Sie sollten testbare Akzeptanzkriterien beinhalten.

Abb. 1: INVEST - Eigenschaften einer guten User Story (nach Wake).

- **D**etailed Appropriately Anstehende Einträge müssen angemessen detailliert spezifiziert sein, um im anstehenden Sprint vom Team umgesetzt werden zu können.
- **E**stimated Einträge im Product Backlog müssen für die Planbarkeit geschätzt sein. Je feiner der Eintrag, desto genauer die Schätzung.
- **E**mergent Ein Product Backlog wächst mit der Zeit. Nachträgliche User Stories finden ihren Weg hinein, andere fallen raus oder bekommen eine andere Priorität.
- **P**rioritized Um den Geschäftswert des Produktes zu maximieren, werden die Product Backlog Items ihrer Wichtigkeit nach in absteigender Reihenfolge eingetragen.

Abb. 2: DEEP - Eigenschaften eines guten Product Backlog (nach Pichler).

Der Übergang vom Groben zum Feinen ist darin verwurzelt, dass sich bestimmte Anforderungen zum Zeitpunkt ihrer Erfassung nicht genauer spezifizieren oder aufgrund einer geringen Priorität zunächst absichtlich nicht weiter verfeinern lassen. Ebenso sollte man nicht dem Irrglauben verfallen, dass an vorab detailliert spezifizierten Anforderungen kein weiterer Diskussionsbedarf bestünde, da der Anschein von Formalität sie als abgeschlossen wirken lässt.

Der Product Owner erstellt das Product Backlog, indem er die weitreichenden Funktionen als Epics und die detaillierteren Product Backlog Items als User Stories erfasst und sie optional in Themes gruppiert. Eine gute User Story sollte dabei die Eigenschaften des INVEST-Ansatzes berücksichtigen (siehe Abbildung 1). Die Anforderungen werden in absteigender Reihenfolge nach ihrer Priorität vom Product Owner ins Product Backlog eingetragen und schrittweise verfeinert. Da die Items mit der höchsten Priorität so früh wie möglich umzusetzen sind, sollten sie auch als Erste in eine passende Größe mit ausreichendem Detailierungsgrad gebracht werden, so dass sie sich innerhalb eines Sprints umsetzen lassen.

Wenn Funktionen erst sichtbar werden, während das Produkt bereits Gestalt annimmt, hat dies keine negativen Auswirkungen auf die laufende Entwicklung. Durch die iterativ inkrementelle Vorgehensweise, werden nachträglich auftretende Anforderungen wie üb-

lich priorisiert, spezifiziert und vom Product Owner ins Product Backlog eingetragen und für anstehende Sprints eingeplant. Als Konsequenz kann möglicherweise eine weniger wichtige User Story nicht mehr in der verbleibenden Zeit fertiggestellt werden. Der feste Timebox-Ansatz in Scrum sorgt dafür, dass es zu keinen Zeitverzögerungen kommt. Er ist aber auch wie ein Gefäß, in das nur eine bestimmte Menge hineinpasst - die werthaltigste Menge.

Dadurch wird auch ersichtlich, warum der Kunde und der Product Owner regelmäßig miteinander kommunizieren sollten. Ein Product Backlog lebt und verändert sich. Eine regelmäßige Pflege, das sogenannte Backlog Grooming, ist daher ratsam, um das Product Backlog an die sich ändernden Gegebenheiten anzupassen. Zur Bewertung eines guten Product Backlog empfiehlt sich eine Orientierung an dem DEEP-Ansatz nach Pichler (siehe Abbildung 2).

Die vierte Herausforderung: Qualität sicherstellen

Qualität liegt bei Scrum in der Verantwortung des gesamten Scrum Teams und ein Stück weit auch beim Kunden. Diese Haltung gilt es bei allen Projektvorhaben mit Hilfe von Scrum zu festigen. Anders als in klassischen Projekten, findet die Stabilisierungsphase zur Qualitätssicherung nicht erst im Anschluss an die Entwicklungsphase statt. Eine Trennung von Analyse, Design, Programmierung und Stabilisierung ist nicht vorgesehen. Eine hohe Qualität wird von Beginn an verfolgt und sichergestellt.

Lösung: Akzeptanztestgetriebene Entwicklung

Da das Ziel die vollumfängliche Erreichung der Produktvision des Kunden ist, sollte der Kunde dem Product Owner dabei behilflich sein, die Akzeptanzkriterien der User Stories festzulegen. Dies kann in Form von Zufriedenheitsbedingungen erfolgen, in denen beschrieben wird, was die jeweiligen Funktionalitäten leisten sollen. Diese werden anschließend in einzelne Akzeptanzkriterien/-tests vom Product Owner bzw. von Mitgliedern des Teams für jede anstehende User Story genauer spezifiziert. Die Akzeptanzkriterien für die jeweils umzusetzende User Story sind für das Team der Maßstab zur Erreichung der gewünschten Qualität. Nur durch sie weiß das Team, wann eine User Story korrekt umgesetzt wurde und als abgeschlossen betrachtet werden kann.

Dieses Vorgehen ermöglicht es, Tests als Teil des Prozesses einzubinden. Durch eine akzeptanztestgetriebene Entwicklung (ATDD) werden Fehler schneller erkannt und die Möglichkeit für ein zeitnahes Feedback geschaffen. Die erstellten Tests zeigen dem Team sofort, ob die Akzeptanzbedingungen während der Ent-

wicklung erfolgreich umgesetzt werden konnten. Dies reduziert nachträgliche Korrekturarbeiten und sorgt für eine dauerhafte Qualitätsorientierung.

Im Laufe der Entwicklung können sich immer neue Anforderungen ergeben oder bestehende Anforderungen verändern. Wenn beispielsweise im Rahmen einer umzusetzenden User Story ein technisches Hindernis auftaucht und dieses Hindernis nur mit einem erheblichen Mehraufwand beseitigt werden kann, liegt es in der Verantwortung des Product Owner, dies mit dem Kunden abzustimmen. Möglicherweise erhält dadurch eine zuvor als sehr wichtig eingestufte Funktion eine niedrigere Priorität oder aber der Kunde entscheidet sich, weiterhin daran festzuhalten. Wichtig ist, dass der Kunde zeitnah vom Product Owner über eintretende Ereignisse in Kenntnis gesetzt und gemeinsam das weitere Vorgehen besprochen wird. Für die Güte und Qualität eines Produktes ist das Erfüllen der Zufriedenheitsbedingungen des Kunden ein wichtiger Faktor.

Die fünfte Herausforderung:
Den Fortschritt kontrollieren

Die Anforderungen wurden gesammelt, ein Release-Plan erstellt und die Entwicklung hat begonnen. Dabei arbeitet das Team innerhalb der Sprints interdisziplinär und selbstorganisiert an der Zielerreichung. Die zuvor festgelegt Velocity unterliegt dabei meist einer gewissen Unschärfe. Daher ist es notwendig, sie nachträglich anzupassen, um eine Entscheidungsgrundlage für Korrekturmaßnahmen zu haben und damit das Team seinen eigenen Fortschritt jederzeit zuverlässig ablesen kann.

Lösung: Team Velocity messen

Um eine verlässliche Velocity des Teams zu erhalten, sollten üblicherweise zunächst zwei bis drei Sprints abgeschlossen sein, da die Geschwindigkeit besonders am Anfang noch recht volatil sein kann. Hierzu ist es hilfreich, die Soll-Velocity mit der Ist-Velocity nach jedem Sprint zu vergleichen (siehe Abbildung 3). Oftmals lässt sich eine Stabilisierung über die fortschreitende Entwicklungsdauer hinweg erkennen. Daher ist es nicht ratsam den Release-Plan sehr früh zu korrigieren. Später hilft die Velocity sehr gut dabei zu erkennen, ob sich der gewünschte Umfang zeitgerecht liefern lässt oder ob korrigierend eingegriffen werden muss.

Ein nützliches Hilfsmittel zur Kontrolle des Fortschritts ist das Projekt-Burndown-Chart. Es zeigt dem Scrum-Team an, wie viel Arbeit gemessen am Gesamumfang des Projektes nach jedem Sprint noch verbleibt, indem die im zurückliegenden Sprint geleistete Arbeit davon abgezogen wird. Das Projekt-Burndown-Chart ist jedoch kein Indikator für die Team-Velocity. Wenn der Kunde den Gesamumfang während der Entwicklung

erhöht, wird der geleistete Burndown auf dem Chart entsprechend geschmälert.

Das Sprint-Burndown-Chart (siehe Abbildung 4) ermöglicht dem Team den Fortschritt schon während des Sprints abzulesen und zu kontrollieren. Es visualisiert die noch verbleibende Arbeit des im Sprint-Planning zuvor festgelegten Sprint-Backlog. Das Team pflegt es alleinverantwortlich, indem die jeweils fertiggestellten Aufgaben vom verbleibenden Sprintumfang abgezogen werden.

Die Erfassung des Fortschritts durch Burndown Charts dient in erster Linie dem Blick nach vorne, weniger dem Erfassen von bereits aufgewendeter Arbeit und

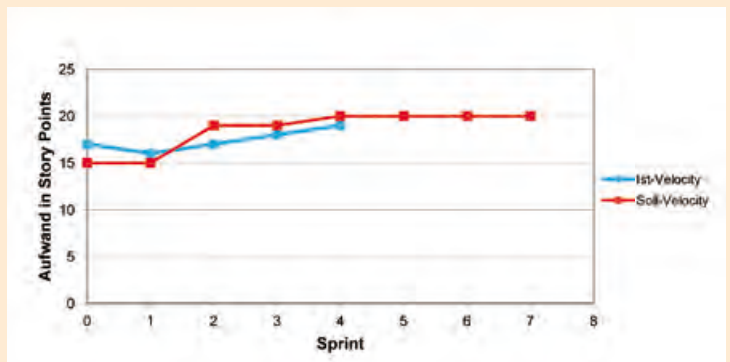


Abb. 3: Vergleich der Soll-Ist-Velocity.

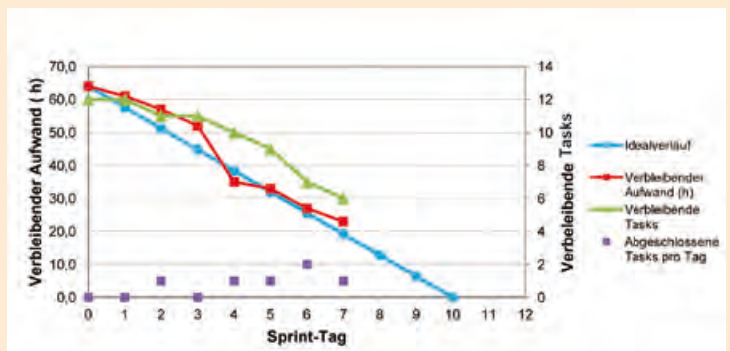


Abb. 4: Sprint-Burndown-Chart.

Quellen

- Roman Pichler (2012): Agiles Produktmanagement mit Scrum: So entwickeln Sie Produkte, die begeistern, Addison-Wesley Verlag, 2012
- Ralf Wirdemann (2011): Scrum mit User Stories, Carl Hanser Verlag GmbH & Co. KG, 2. Auflage (2011)
- Mike Cohn (2010): Agile Softwareentwicklung: Mit Scrum zum Erfolg!, Addison-Wesley Verlag (2010)

Glossar

ATDD

Acceptance Test Driven Development - Akzeptanztestgetriebene Entwicklung

Artefakt

Ein Artefakt ist das Ergebnis aus einem Arbeitsprozess.

Backlog Grooming

Das Backlog Grooming beschreibt die regelmäßige Pflege des Product Backlog durch Anpassen, Hinzufügen und Entfernen von Produkteigenschaften.

Empirismus

Empirismus ist die Theorie, Entscheidungen auf Basis gewonnener Erfahrungen treffen zu können („Inspect & Adapt“).

Epic

Ein Epic ist eine grobgranulare Anforderung, deren beschriebene Funktionalität nicht in eine einzelne User Story passt und inhaltlich mehrere Sprints umfassen kann.

Product Backlog

Das Product Backlog ist eine priorisierte Liste, die alles enthält, was im zu entwickelnden Produkt enthalten sein sollte.

Sprint

Ein Sprint umfasst eine festgelegte Dauer zur Erstellung eines potentiell auslieferbaren Inkrements.

Theme

Inhaltlich zusammenhängende User Stories werden einem bestimmten Thema zugeordnet.

Timebox-Ansatz

Dieser Ansatz ist eine Technik der Projektplanung, bei der ein fester Zeitrahmen vorgegeben wird, wie beispielsweise die Dauer der Sprints.

USP

Die Unique Selling Proposition ist das Alleinstellungsmerkmal, welches ein Angebot vom gesamten Wettbewerb abhebt.

User Story

Eine User Story ist eine bewusst kurz gehaltene, schriftlich fixierte Anforderung in narrativer Form, inkl. Akzeptanzkriterien für eine erfolgreiche Abnahme.

Velocity

Die Velocity beschreibt die Menge an Arbeit aus dem Product Backlog, die ein Team innerhalb eines Sprints bewältigen kann.

Links

- ▶ [1] ORDIX news Artikel 02/2013
„Scrum - aber richtig! (Teil I) - Optimale Abläufe und innere Faktoren“:
http://www.ordix.de/images/ordix/onews_archiv/2_2013/ORDIX_news_2_2013_opf_files/WebSearch/page0047.html
- ▶ [2] Agiles Manifest:
www.agilemanifesto.org
- ▶ [3] Official Scrum Guide:
<http://www.scrum.org/Scrum-Guides>
- ▶ [4] Mike Cohn:
<http://www.mountaingoatsoftware.com/blog/>
- ▶ [5] Roman Pichler, DEEP:
<http://www.romanpichler.com/blog/product-backlog/making-the-product-backlog-deep/>
- ▶ [6] Bill Wake, INVEST:
<http://xp123.com/articles/invest-in-good-stories-and-smart-tasks/>
- ▶ [7] Coley Consulting, MoSCoW:
<http://www.coleyconsulting.co.uk/moscow.htm>

gar nicht der Kontrolle einzelner Mitarbeiter. Es geht vielmehr um die Einhaltung der Werte Transparency, Inspect & Adapt.

Der Kunde sollte dabei die Diagramme verstehen können und die Velocity des Teams kennen. Es liegt auch in seiner Verantwortung den Gesamtumfang an Anforderungen im Blick zu behalten, so dass die Projektziele weiterhin erreichbar bleiben. Die Aufgabe des Product Owner liegt darin, den Kunden ggf. darauf aufmerksam zu machen und gemeinsam mit ihm die mögliche Zielerreichung des Projektes zu wahren.

Die nivellierte Velocity eines eingespielten Teams spiegelt die nachhaltige Entwicklungsgeschwindigkeit wider, mit der die Arbeit im Sprint geleistet werden kann. Das Ziel sollte es sein, dieses Tempo zu halten. Ein eher ungeeignetes Instrument zur Kurskorrektur und Zielerreichung sind Überstunden über einen längeren Zeitraum hinweg. Auf dieses Mittel wird häufig zurückgegriffen, wenn sich in einem Projekt abzeichnet, dass der verbleibende Umfang bis zum festgelegten Termin nicht mehr fertiggestellt werden kann. Wenn sich der Zeitraum, in dem Überstunden geleistet werden, aber über mehr als zwei Wochen erstreckt, sinken die Motivation und die durchschnittliche Entwicklungsgeschwindigkeit mit fortschreitender Dauer immer weiter bis schließlich unter das Normalniveau ab. Es liegt daher in der Verantwortung des Product Owner, das Team stattdessen durch eine ansprechende Vision des Produktes zu motivieren und falls nötig, mit dem Kunden den Anforderungsumfang anzupassen bzw. entsprechend neu zu priorisieren.

Fazit

Das Ziel der schrittweisen Verbesserung ist in allen Bereichen der Projektentwicklung mit Scrum gut sichtbar. Im Backlog festgehaltene Anforderungen unterliegen genauso einem Wandel, wie die Praktiken des Teams oder das Anpassen von organisatorischen Prozessen durch den Scrum Master.

Eine der Hauptaufgaben des Kunden und des Product Owner besteht darin, das Team auf diesem Weg zu begleiten und dabei die Maximierung des Geschäftswerts zu erreichen. Wenn die wenigen aber klaren Regeln von Scrum befolgt werden und der Product Owner den Kunden stets informiert und involviert, lässt sich eine Zielerreichung des Projektes gemäß den Vorstellungen des Kunden am ehesten bewerkstelligen.



Sven Müller
(info@coniatos.de)

Enterprise Manager Oracle Cloud Control 12c (Teil VI)

emcli unter Cloud Control - das unterschätzte Tool

Die Einsatzmöglichkeiten des Enterprise Manager Command Line Interface, kurz emcli, werden häufig unterschätzt. Gerade wenn es darum geht, eine große Anzahl von Zielsystemen zu verwalten, kann man mit dem emcli viel Zeit sparen. Im Folgenden werden die Einsatzmöglichkeiten des Werkzeuges kurz erläutert und in Beispielen die Konfiguration und die Verwendung des emcli dargestellt.

emcli - Was ist das eigentlich?

Das Enterprise Manager Command Line Interface ist ein Administrations-Client für den Oracle Enterprise Manager. Mit dem emcli können fast alle administrativen Tätigkeiten für den Enterprise Manager (EM) auf der Kommandozeile ausgeführt werden. Die Management-Agents der einzelnen Targets können nicht direkt mit emcli administriert werden. Das Werkzeug kann ausschließlich für die Administration des EM-Server verwendet werden. Dabei ist es ein Bestandteil der EM-Server-Installation, wodurch es nicht auf dem EM-Server installiert werden muss. Möchte man emcli von einem beliebigen Client aus verwenden, so muss es auf diesem gesondert installiert werden [6].

Einrichten des emcli-Client

Vor der Benutzung des emcli-Client muss dieser konfiguriert werden. Die Konfiguration wird wie folgt durchgeführt:

```
emcli setup -url="https://[MY_OMS.de]:7799/em" -username=[EMCLI_ADMIN] -dir=/home/oracle/emcli
```

Über den Parameter `url` werden die Adresse und der Port der EM-Console übergeben. Als Port muss HTTPS (SSL) oder ein HTTP-Port angegeben werden (7799 ist der Standard-HTTPS-Port). Mit `username` wird der EM-Benutzername übergeben, über den emcli sich mit dem EM verbindet. Das Passwort des Benutzers wird interaktiv in einem neuen Fenster abgefragt. Der Parameter `dir` gibt den Pfad an, in welchem die emcli-Konfiguration gespeichert werden soll.

Die Parameter `url` und `username` müssen bei der Konfiguration angegeben werden. Der Parameter

`dir` ist optional, wird er nicht angegeben werden die Konfigurationsdateien in das Home-Verzeichnis des Benutzers geschrieben, welcher den emcli-Setup-Befehl ausführt.

Dem Setup-Kommando können noch eine Reihe weitere Parameter übergeben werden. Eine Liste dieser Parameter und eine Beschreibung ihrer jeweiligen Funktion findet man über die Hilfefunktion von emcli oder in der Produktdokumentation [7].

Nach der Konfiguration von emcli kann eine Verbindung zum EM-Server hergestellt werden:

```
emcli login -username=[EMCLI_ADMIN]
```

```
[oracle@oracle ~]$ emcli help list_named_credentials
emcli list_named_credentials
    -cred_name=[cred_name]
    -cred_owner=[cred_owner]

Description:
List the named credentials.
And EM Administrator can list the credentials that he owns
or has explicit access on the credential.

Options:
    -cred_name
        Credential name to filter the list of credentials displayed.
    -cred_owner
        Credential owner to filter the list of credentials displayed.

Examples :
emcli list_named_credentials
    -cred_name="NC"
This command will list the named credentials that have credential
name like NC
```

Abb. 1: Beispiel für die Hilfefunktion für das Verb `list_named_credentials`.

```
[oracle@oracle ~]$ emcli add_target_property -target_
type="oracle_database" -property="Abteilung"
```

```
Property "Abteilung" added successfully
```

```
[oracle@oracle ~]$
```

Name	Value
Abteilung	
Comment	

Abb. 2: Erstellen einer User-Defined-Target-Property und das Ergebnis in der EM-Oberfläche.

Die Hilfefunktion

Die Hilfefunktion von emcli wird mit dem Befehl `emcli help` aufgerufen. Dieser listet alle zur Verfügung stehenden Befehle auf. Um eine detailliertere Hilfe mit Beispielen zu den einzelnen Befehlen zu erhalten, kann `emcli help <emcli-Verb>` aufgerufen werden (siehe Abbildung 1).

Die Funktionalitäten

Mit emcli kann man fast alle administrativen Tätigkeiten für den EM durchführen, ohne auf die Oberfläche des EM zuzugreifen. Im Folgenden nennen wir einige Beispiele für Tätigkeiten, welche mit emcli durchgeführt werden können:

- Anlegen eines neuen EM-Benutzers (normaler Benutzer oder Administrator)
- Jobs anlegen und weiterleiten
- Setzen von Named Credentials oder Preferred Credentials
- Starten und Stoppen von Agenten
- Setzen und Löschen von Blackouts
- Anlegen und Löschen von Targets
- Anlegen und Löschen von Gruppen
- Zuweisung von Targets zu Gruppen
- Anlegen und Löschen von Target-Properties
- Installation/Deinstallation von Plugins

Eine Liste aller Funktionalitäten von emcli ist in der emcli-Dokumentation von Oracle enthalten [7].

Weiterhin gibt es einige neue Funktionen, welche vorerst nur in emcli implementiert wurden: Ein Beispiel hierfür ist das Anlegen von User-Defined-Target-Properties, mit deren Hilfe man die von Oracle vorgegebenen Target-Properties erweitern kann. Dadurch ist es möglich, zusätzliche Informationen den einzelnen Targets als Eigenschaften zuweisen. In Abbildung 2 wurde eine User-Defined-Target-Property mit dem Namen „Abteilung“ angelegt. Hier könnte man für jedes Target die zuständige Fachabteilung hinterlegen und die Information direkt in der Oberfläche des EM auslesen.

Ebenso ist es möglich, auf dieser Basis dynamische Gruppen zu erzeugen. Leider ist dies aktuell nur aus den vordefinierten Properties möglich. Derzeit können User-Defined-Target-Properties noch nicht für die Anlage einer dynamischen Gruppe verwendet werden, diese Funktion soll aber in einer der nächsten EM-Versionen verfügbar sein.

Die Einsatzmöglichkeiten

Mit Hilfe des emcli können administrative Tätigkeiten automatisiert durchgeführt werden. So können die emcli-Aufrufe zum Beispiel in ein Bereitstellungsskript für die Agent-Installation auf den Targets integriert werden. emcli bietet auch die Möglichkeit neue Targets dem EM hinzuzufügen, wodurch der Installationsprozess weitestgehend automatisiert werden kann.

Ebenso kann das Hinzufügen der Targets zu bestehenden Gruppen realisiert werden. Wodurch das zum Teil recht mühselige Einsortieren der Targets in die einzelnen Gruppen entfällt.

Weiterhin bietet das emcli auch die Möglichkeit, Passwörter von Datenbankbenutzern flächendeckend zu ändern. Zum einen können hier die Passwörter innerhalb des EM geändert werden, aber auch die Passwörter auf der Datenbank selbst. Das folgende Beispiel zeigt die Syntax für das Ändern von Passwörtern der Datenbankbenutzer in der Datenbank und im EM-Server:

```
emcli update_db_password
      -target_name=myDB
      -user_name=Admin1
      -change_at_target=yes
```

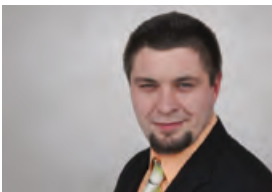
Für den Bereich Reporting bietet emcli ebenfalls einige Möglichkeiten. So können zum Beispiel einzelne Reports für mehrere Targets erzeugt werden, ohne dass diese aufwendig in der Oberfläche einzeln erstellt werden müssen.

Fazit

emcli bietet eine Fülle an Möglichkeiten, die administrativen Tätigkeiten im EM-Umfeld auf Kommandozeilenebene durchzuführen. Gerade bei Änderungen an vielen Targets kann man die Abläufe über dieses Werkzeug optimieren. Hier ist es meist effizienter, die Änderungen mit dem emcli durchzuführen, als sie für jedes einzelne Target in der Oberfläche vorzunehmen.

Bei der Installation von neuen Targets kann man einige Tätigkeiten mit emcli durchführen und sie so in den Installationsprozess integrieren. Auch hier kann man Zeit einsparen, da die Tätigkeiten durch ein Skript für alle neuen Targets ausgeführt werden. Weiterhin wird das Fehlerpotenzial bei der Übernahme neuer Targets in den EM minimiert.

Das Potential von emcli wird häufig unterschätzt. Gerade bei administrativen Tätigkeiten, welche für eine große Anzahl an Targets durchgeführt werden müssen, lohnt es sich, einen Blick auf die Möglichkeiten dieses Werkzeuges zu werfen.



Michael Thieme
(info@ordix.de)

Glossar

OMS

Der Oracle Management Server ist eine Softwarekomponente, welche die Schicht zwischen den einzelnen Management-Agenten der Targets und der Enterprise Management-Konsole darstellt.

EM

Der Enterprise Manager ist eine Software, welche im Kern Möglichkeiten für die Verwaltung, Überwachung und Administration von Datenbanken und anderen Softwarepaketen von Oracle bietet. Er beinhaltet für diese Aufgaben eine grafische und eine textbasierte Oberfläche.

emcli

Das Enterprise Manager Commandline Interface ist eine textbasierte Oberfläche für die Verwaltung und die Überwachung von Softwarekomponenten über den EM.

Target

Als Target bezeichnet man die einzelnen Komponenten eines vom EM überwachten Zielsystems. Dies können u.a. Datenbanken, Listener, Host und auch Management-Agents selbst sein.

Named Credentials

Sie speichern Benutzerkennungen und die dazugehörigen Passwörter, hierbei erhalten diese Kombinationen eindeutige Bezeichnungen. Im Unterschied zu Preferred Credentials können Named Credentials von mehreren EM-Benutzern verwendet werden.

Preferred Credentials

Preferred Credentials bieten einem EM-Benutzer die Möglichkeit, Zugangsdaten für einzelne Targets (z.B. eine Datenbank) zu speichern. Die Kombination aus Target-Benutzerkennung und Passwort wird für den EM-Benutzer separat gespeichert und einem Target zugewiesen.

Links

- ▶ [1] ORDIX news Artikel 02/2012
„Enterprise Manager Oracle Cloud Control 12 c (Teil I) - Über den Wolken...“:
http://www.ordix.de/images/ordix/onevs_archiv/2_2012/oracle_cloud_control.html
- ▶ [2] ORDIX news Artikel 03/2012
„Enterprise Manager Oracle Cloud Control 12 c (Teil II) - Die Wolke beginnt zu schweben“:
http://www.ordix.de/images/ordix/onevs_archiv/3_2012/pages/page0044.html
- ▶ [3] ORDIX news Artikel 04/2012
„Enterprise Manager Oracle Cloud Control 12 c (Teil III) - Das Gitternetz wird zur Wolke“:
http://www.ordix.de/images/ordix/onevs_archiv/4_2012/ORDIX_news_4_2012_opf_files/WebSearch/page0010.html
- ▶ [4] ORDIX news Artikel 01/2013
„Enterprise Manager Oracle Cloud Control 12 c (Teil IV) - Eine Wolke in der Cloud“:
http://www.ordix.de/images/ordix/onevs_archiv/1_2013/ORDIX_news_1_2013_opf_files/WebSearch/page0045.html
- ▶ [5] ORDIX news Artikel 02/2013
„Enterprise Manager Oracle Cloud Control 12 c (Teil V) - Chargeback in der Praxis“:
http://www.ordix.de/images/ordix/onevs_archiv/2_2013/ORDIX_news_2_2013_opf_files/WebSearch/page0015.html
- ▶ [6] Oracle Enterprise Manager User Manual:
http://docs.oracle.com/cd/E24628_01/em.121/e17786.pdf
- ▶ [7] Oracle Enterprise Manager Lifecycle Management Administrator's Guide:
http://docs.oracle.com/cd/E24628_01/em.121/e27046/emcli.htm



Testmanagement in IT-Projekten

Gut geprüft ist halb abgenommen

In der letzten Ausgabe der ORDIX news haben wir festgestellt, dass das Testmanagement in IT-Projekten nicht erst nach Abschluss der Entwicklung, sondern bereits mit dem Start eines Projektes beginnen sollte [1]. Die vorbereitenden Schritte vor der eigentlichen Testdurchführung wurden bereits beleuchtet. In diesem Artikel möchten wir uns nun der eigentlichen Durchführung der Tests widmen.

Testmanagement in den Projektphasen

Wie bereits im ersten Artikel [1] beschrieben, kann das Testmanagement in verschiedene Phasen eingeteilt werden, die den Entwicklungsphasen eines Projektes zugeordnet werden können (siehe Abbildung 1). Nachdem wir uns bereits mit der Erstellung des Testkonzeptes und der Testfälle sowie der Testplanung beschäftigt haben, soll es nun um die sich daran anschließenden Phasen gehen. Es handelt sich dabei um den Unit-Test, der während der Entwicklungsphase durchgeführt wird, den Systemtest und um den Abnahmetest.

Unit-Test

Der Unit-Test (auch Modultest oder Komponententest genannt) wird während der Entwicklung vom Entwickler auf der Entwicklungsumgebung durchgeführt. Durch den Unit-Test soll sichergestellt werden, dass

die Komponente, die getestet wird, ohne technische Fehler, wie z.B. Kompilierfehler ausgeführt werden kann. Außerdem wird an dieser Stelle geprüft, ob die Komponente die korrekten fachlichen Ergebnisse bzw. Teilergebnisse liefert. Bei performance-kritischen Komponenten sollte an dieser Stelle bereits, zusätzlich zum fachlichen bzw. technischen Test, ein Last- und Performance-Test durchgeführt werden. Während des Unit-Tests müssen die durchgeführten Überprüfungen und die jeweiligen Ergebnisse dokumentiert werden.

Systemtest

Der Systemtest schließt sich an die Entwicklungsphase an. Das primäre Ziel ist es, dass die Qualität der Software ausreichend ist, um diese für die Abnahme bereitstellen zu können. Dabei wird u.a. die Zusam-

menarbeit der voneinander abhängigen Komponenten getestet. Die Schnittstellen der beteiligten Komponenten werden untersucht, so dass korrekte Ergebnisse über komplette Abläufe hinweg nachgewiesen werden können.

Es werden alle in der Spezifikation gestellten Anforderungen zu 100 % getestet. Alle Tests werden in einer Systemumgebung getestet, die dem produktiven Einsatzszenario möglichst nahe kommt. Die Software wird hier als komplettes Lieferpaket installiert. Dies ist wichtig, damit später sichergestellt werden kann, dass immer der Softwarestand getestet wird, der später auch zur Abnahme bereitgestellt und in der Produktion installiert wird.

Durchführung von Testfällen

Um die Softwarequalität zu ermitteln, werden die während der DV-Konzepterstellung verfassten Testfälle durchgeführt. Dabei muss sichergestellt werden, dass alle abnahmerelevanten Testfälle während der Systemtestphase durchgeführt werden. Zudem können weitere Testfälle und auch Freihandtests erfolgen. Die Durchführung der Testfälle erfolgt nach dem in Abbildung 2 dargestellten Schema.

Vor der eigentlichen Durchführung muss der Testfall vorbereitet werden, d.h. es müssen erforderliche Testdaten erzeugt oder aber eine bestimmte Ausgangssituation hergestellt werden. Im Anschluss werden die im Testfall beschriebenen Testschritte durchgeführt. Für jeden Testschritt wird das Ergebnis dokumentiert.

Tritt bei einem Testschritt nicht das erwartete Ergebnis ein, sollte zunächst geprüft werden, ob im Testfall ein Fehler vorliegt. Ist dies der Fall, so muss der Testfall überarbeitet werden. Liegt ein Softwarefehler vor, so wird dieser in einem Werkzeug zur Fehlerverfolgung dokumentiert. Der Fehler wird der Entwicklung zugewiesen, die ihn entsprechend behebt. Wichtig ist dabei, dass Fehlerbehebungen immer nur mit einer neuen Softwarelieferung in die Testumgebung eingespielt werden. Nur so kann sichergestellt werden, dass alle Fehler in der Software, die später zur Abnahme bereitgestellt wird, auch behoben werden. Wurde eine neue Lieferung auf der Testumgebung installiert, so wird ein Nachttest aller behobenen Fehler durchgeführt. In der Regel wird dazu der Testfall erneut durchgeführt.

Regressionstests

Um zu gewährleisten, dass die Software weiterhin gemäß der fachlichen Anforderungen arbeitet, werden in der Testphase zusätzliche Regressionstests durchgeführt. Neben den Tests des funktionalen Zuwachses und der behobenen Fehler werden auch wesentliche Funktionalitäten, die vorher schon geliefert und

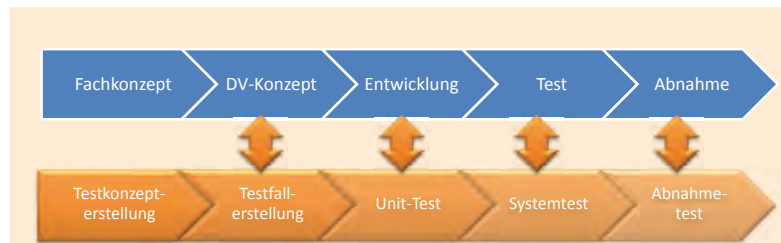


Abb. 1: Testmanagement in den Projektphasen.

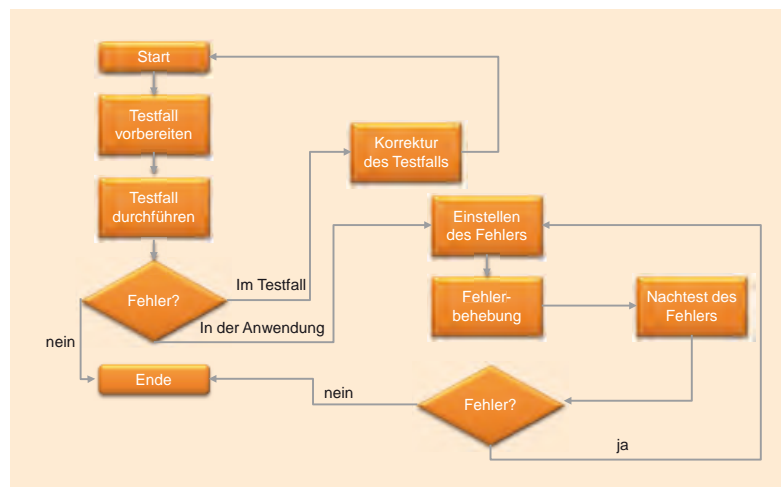


Abb. 2: Durchführung der Testfälle.

erfolgreich getestet wurden, nachgetestet. Mit diesem Regressionstest wird gewährleistet, dass die „alten“ Funktionalitäten weiterhin wie spezifiziert laufen.

Last- und Performance-Test

Bei gravierenden Änderungen an einer bestehenden Software bzw. bei einer neu erstellten Software empfiehlt es sich immer, ihr Verhalten auch unter Last zu überprüfen. Dabei müssen die vorgesehenen Systemkonfigurationen auf der bereitgestellten Hardware der zu erwartenden Last gewachsen sein. Eventuell vorhandene Schwachstellen müssen gezielt identifiziert werden.

Diese Art von Tests sollten für alle performancekritischen Komponenten (häufiges I/O, komplizierte DB-Abfragen usw.) durchgeführt werden.

Ein Last- und Performance-Test wird in der Regel parallel zum Systemtest durchgeführt. Dabei sollte allerdings darauf geachtet werden, dass er erst ca. zwei Wochen nach dem Systemtest gestartet wird. Dadurch wird sichergestellt, dass die Software lauffähig ist und gravierende Fehler bereits behoben sind.

Glossar

DV-Konzept

Beim DV-Konzept handelt es sich um ein technisches Konzept, in dem beschrieben wird, wie die Anforderungen aus dem Fachkonzept technisch umgesetzt werden sollen.

Fachkonzept

Das Fachkonzept beschreibt die funktionalen Anforderungen in einem IT-Projekt.

Fehlerverfolgungs-Werkzeug

Werkzeug zum Erfassen und Dokumentieren von Programmfehlern - es bietet die Möglichkeit den Status aller dokumentierten Fehler zu jedem Zeitpunkt zu ermitteln und auszuwerten. Meistens wird auch die Möglichkeit geboten, Reports zu erzeugen. Bekannte Werkzeuge sind z.B. Bugzilla, Jira und HP ALM.

Freihandtest

Beim Freihandtest werden keine speziellen Testfälle abgearbeitet. Vielmehr überprüft der Tester die Anwendung, indem er einzelne Funktionen intuitiv bearbeitet.

Links

- ▶ ORDIX news Artikel 02/2013
„Verbesserung der Softwarequalität - Testmanagement in IT-Projekten“:
http://www.ordix.de/images/ordix/onews_archiv/2_2013/ORDIX_news_2_2013_opf_files/WebSearch/page0037.html

Bildnachweis

© pixabay.com | Test Testing Optical Magnifier Search

Zum Abschluss des Last- und Performance-Tests ist immer ein Lasttestbericht zu erstellen, der die Ergebnisse des Tests und ggf. durchzuführende Maßnahmen dokumentiert.

Reporting

Täglich wird eine Übersicht aller Fehler des Systemtests erstellt und der Projektleitung zur Verfügung gestellt. Diese Fehlerstatistik kann in einem eigenständigen Excel-Dokument oder auch in Fehlerverfolgungs-Werkzeugen gepflegt werden. Die Fehler werden nach ihrer Priorität und innerhalb dieser nach dem Fehlerstatus aufgeteilt.

Ein Fehler gilt solange als „offen“, bis er vom Entwickler behoben und in einer neuen Softwareversion geliefert worden ist. Solange die Fehlerkorrektur geliefert, aber noch nicht nachgetestet worden ist, befindet sich der Fehler im Status „im Nachtest“. War der Nachtest erfolgreich, erhält der Fehler den Status „geschlossen“.

Konnte die Fehlerkorrektur nicht erfolgreich nachgetestet werden, so wird der Fehler wieder an den Entwickler gegeben und mit dem Status „offen“ in der Fehlerstatistik geführt.

Am Ende des Systemtests wird vom Testmanager ein Testendebericht erstellt. In diesem gibt er die Software für die Abnahme frei. Zudem werden noch offene Fehler dokumentiert und festgelegt, wann diese behoben werden.

Abnahmetest

Im Anschluss an die dargestellten Tests erfolgt schließlich der Abnahmetest, der als Ziel die Erteilung der Abnahme hat.

Der Abnahmetest wird i.d.R. vom Kunden durchgeführt. Dazu wird die durch den Systemtest überprüfte Software auf einer gesonderten Umgebung installiert. Anschließend werden alle zuvor definierten abnahme-relevanten Testfälle durchgeführt. Zusätzlich können noch Regressionstests und Freihandtests durchgeführt werden. Das Vorgehen bei der Testfalldurchführung entspricht dem des Systemtests. Auch das Reporting sollte in der gleichen Form wie beim Systemtest erfolgen. Allerdings geht hier die Statistik nicht nur an die Projektleitung sondern zusätzlich an den Kunden.

Am Ende des Abnahmetests erstellt der Kunde im Idealfall die Abnahmeerklärung. Durch die Erteilung der Abnahme erklärt er, dass alle Funktionalitäten der Anwendung der Spezifikation entsprechen und somit der Vertrag zwischen Auftragnehmer und Auftraggeber erfüllt wurde. Erst nach Erteilung der Abnahme durch den Kunden, kann der Auftragnehmer die (Schluss-)Rechnung stellen.

Fazit

Um den Anforderungen in der Softwareentwicklung gerecht zu werden, ist ein ausführliches Testmanagement in allen Phasen eines Projektes unerlässlich. Durch die stetig wachsende Komplexität ist ein strukturiertes Testmanagement sowie ein qualifiziertes Testen zwingend notwendig.



Kathrin Hammerschmidt
(info@ordix.de)

NoSQL vs. SQL - Hype oder echte Alternative? (Teil V)

Neo4j - die Graphendatenbank

Mit Neo4j stellen wir Ihnen in diesem fünften und letzten Teil unserer Artikelreihe zum Thema „NoSQL vs. SQL“ eine der meistgenutzten Graphendatenbanken (GDB) vor. Hinter vielen Anwendungen, die wir tagtäglich verwenden, verbirgt sich eine Graphenstruktur. Vor allem in den sozialen Netzwerken sind die Anbieter auf effiziente und zuverlässige Datenbanklösungen angewiesen, da die Anzahl der Benutzer, der Funktionsumfang der Anwendungen und somit auch die Komplexität der zu speichernden Daten stetig steigt. Neo4j ist eine Datenbank, die genau diesen Anforderungen gerecht werden kann.

Finde den kürzesten Weg

In der Informatik werden sehr häufig Graphen herangezogen. So beispielsweise um das sogenannte „kürzeste-Wege-Problem“ zu lösen. In dem Bereich der Netzwerktechnik stößt man dabei unweigerlich auf das Spanning-Tree-Protokoll (STP), welches dafür sorgt, dass kein Datenpaket dupliziert wird und damit doppelt an seinem Ziel ankommt. Die Abbildung 1 zeigt ein vereinfachtes visualisiertes Beispiel eines Routengraphen, der für das „kürzeste-Wege-Problem“ verwendet werden kann. Graphendatenbanken haben sich demnach auf vernetzte Informationen und deren effiziente indexfreie Traversierung spezialisiert [1].

Von Graphen und Brücken

Dabei haben sie, wie alle graphenbasierten Systeme, die Graphentheorie von Leonhard Euler aus dem 18. Jahrhundert zur Grundlage, die sich im Rahmen des bekannten Königsberger Brückenproblems entwickelt hat [2].

Graphenstrukturen wurden im Datenbankbereich überwiegend in relationalen Datenbanken (RDB) abgebildet, bis diese die gestiegenen Anforderungen nicht mehr zur vollsten Zufriedenheit erfüllen konnten. Denn das Ressourcenintensivste und Komplizierteste, was in einer Datenbank gespeichert werden kann, sind Beziehungen zwischen verschiedenen Elementen. Erhöhte Datenmengen, Abfragen und Zugriffe sind hier am auffälligsten. Mit sehr großem Erfolg agieren an dieser Stelle speziell auf die Anforderungen der Beziehungsdarstellung verschiedener Elemente zugeschnittene Graphendatenbanken (GDB).

Kanten, Knoten und Gewichte

Graphen besitzen in der Informatik eine interessante Eigenschaft. Sie bilden nicht nur die Beziehungen einfach und anschaulich ab, sondern bieten die Möglichkeit weitere Informationen (sogenannte Labels) zu den einzelnen Objekten zu speichern. Wenn einer Kante ein numerischer Wert oder ein sogenanntes Gewicht zuordnet wird, spricht man von einer gewichteten Kante.

Im Beispiel der Routenfindung in einem Computernetzwerk wird eine Kante z.B. durch Leitungskosten oder durch die zur Verfügung stehende Netzwerkgeschwindigkeit (Bandbreite) gewichtet. Zudem können die einzelnen Knoten durch Standortinformationen der Router oder Switches als Labels im Knoten modelliert werden (siehe Abbildung 2).

Unterschiedliche Beziehungen

Ein Graph heißt (Kanten-)gewichtet, wenn er an seinen Kanten semantische Inhalte in Form sogenannter Gewichte besitzt. Kanten können in einem Graphen gerichtet oder ungerichtet sein. Dabei gehen gerichtete Kanten eine einseitige Beziehung und ungerichtete Kanten eine beidseitige Beziehung mit dem Knoten ein. Die Beziehung zwischen Freunden kann i.d.R. als ungerichtete Kante dargestellt werden, wobei die Beziehung zwischen Vater und Kind als gerichtete Kante dargestellt werden muss.

Zwei Knoten können allerdings auch von mehr als einer Kante verbunden werden. Dabei spricht man dann von Mehrfachkanten bzw. Multigraphen [1, Seite 172].

Warum nicht relational?

Relationale Datenbanken (RDB) können nur mit einem sehr großen Aufwand (Zeit und Ressourcen) rekursiv vernetzte Informationen abbilden. Am Beispiel eines sozialen Netzwerkes kann dieser Sachverhalt anschaulich dargestellt werden. Soziale Netzwerke bilden z.B. die Beziehungen zwischen Personen ab („ist befreundet mit“). Diese Beziehungen haben verschiedene Stati. Bei Graphen spricht man auch von Eigenschaften (siehe Abbildung 2). Um die Eigenschaft der Beziehung zwischen zwei Personen herauszufinden, würde eine RDB für diese Statusabfrage eine JOIN-Anweisung benötigen, was auf der obersten Ebene - z.B. „Wer sind die Eltern von Klaus?“ - unproblematisch ist. Mit zunehmender Tiefe - z.B. „Wer ist der Ur-Ur-Enkel von Peter?“ - werden diese Anweisungen aufwendig und nur schwer mit einer RDB in einem angemessenem Zeitraum zu

lösen sein. Dies ist darauf zurückzuführen, dass der Aufwand für diese Anweisungen mit der Anzahl der Datensätze, der Komplexität der Beziehungen und der damit verbundenen Tiefe an Beziehungen exponentiell steigt.

Semi-strukturierte Datenverarbeitung

Die Verarbeitung von semi-strukturierten-Daten [4], wie sie in sozialen Netzwerken, Blogs, etc. zu finden sind, ist als weitere Anforderung an eine GDB anzusehen. Hier haben die Vertreter von NoSQL-Datenbanken den Vorteil, dass sie eine schwache bzw. lediglich eine lose Bindung an ein Datenbankschema haben. Relationale Datenbanken haben hingegen aufgrund des relationalen Datenbankmodells von J. F. Codd [5], kaum die Möglichkeit auf Schemaänderungen zu reagieren. Bei solchen Änderungen müsste in einer RDB über eine sogenannte „Denormalisierung“ oder über Sharding nachgedacht werden. Dieses aber nachträglich in einer Datenbank zu implementieren ist in den meisten Fällen sehr kostspielig und aufwendig.

Modellierung mit Neo4j

Neo4j beruht auf dem Property-Graph-Datenmodell (PGM) und ermöglicht dadurch die Modellierung von umfangreichen Graphen, die folgende Charakteristika aufweisen:

- Beschriftungen (engl. label)
- gerichtete Kanten (engl. directed)
- Eigenschaft (engl. attribute)
- mehrere Kanten (engl. multi-graph)

Obwohl das PGM einen wissenschaftlichen Fundus vermissen lässt, hat es sich bei allen auf dem Markt befindlichen Graphendatenbanken durchgesetzt. Dies lässt sich darauf zurückführen, dass sich das PGM im Gegensatz zu anderen Graphenmodellen durch einfache Modifikationen in andere Modelle umwandeln lässt und somit eine enorme Flexibilität aufweist [3].

Unterstützte Abfrage-/Programmiersprachen

Derzeit gibt es für Graphendatenbanken keine standardisierte Abfragesprache. Dies könnte zum einen an dem jungen Alter der Graphendatenbanken liegen und zum anderen an der Komplexität der Abfragen. Bei Abfragen von RDF-Daten gibt es zumindest einen de facto Standard mit der SPARQL Protocol And RDF Query Language (SPARQL) [7]. SPARQL ist die dominierende Abfragesprache für RDF-Daten. Sie wurde vom World Wide Web Consortium (W3C) am 15.01.2008 mit dem Status „Empfehlung“ freigegeben und wird von Neo4j durch die SAIL-Schnittstelle [8] unterstützt.

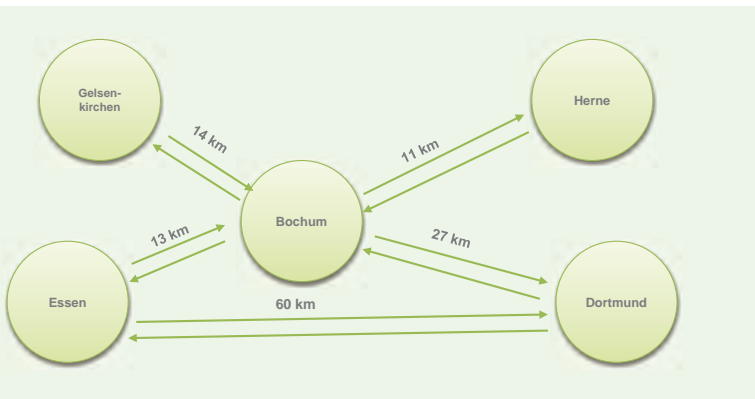


Abb. 1: Beispiel eines Routengraphen.

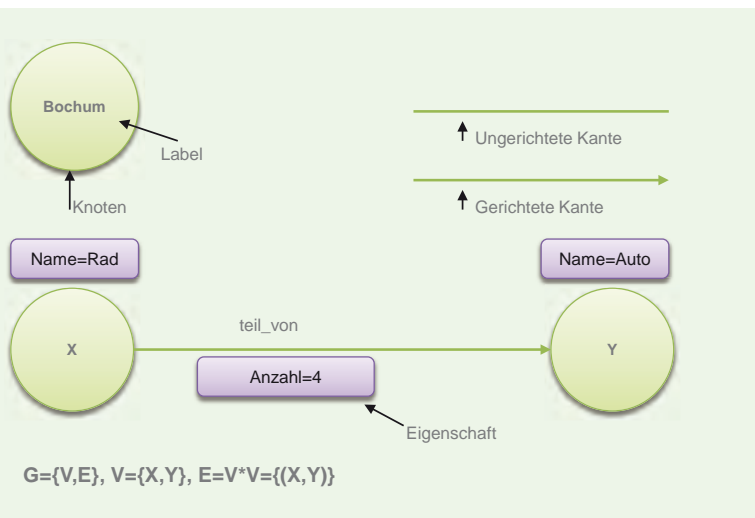


Abb. 2: Kanten, Knoten und Gewichtung in einem Graphen.

Cypher und Gremlin

Für Abfragen, die andere Muster aufweisen als diejenigen, die bei RDF-Daten vorzufinden sind (z.B. Pfade von unbekannter Länge), hat Neo4j die Cypher Query Language (CQL) implementiert. CQL hat sich an SQL orientiert, wie man z.B. an den Schlüsselwörtern **WHERE** und **ORDER BY** erkennen kann. Sie ist eine deklarative Sprache und konzentriert sich auf das Ergebnis einer Suche und nicht auf deren Formulierung. Die CQL besteht, ähnlich wie SQL, aus mehreren Teilen:

- **START** (gibt den Startknoten der Suche anhand der ID des Knotens an)
- **MATCH** (gibt das Suchmuster an, auf das die Suche abzielt)
- **WHERE** (Filterkriterien, ähnlich derer von SQL)
- **RETURN** (gibt das Ergebnis der Suche zurück)

Neben der implementierten graphentraversierten Abfragesprache CQL unterstützt Neo4j das an XPath angelehnte Tinkerpop Gremlin 174 (im folgenden Gremlin genannt). Gremlin ist ein externes Framework, welches im Wesentlichen direkt die Neo4j-Core-API nutzt und daher perfekt zu Neo4j passt.

Wie schnell kann es schon sein?

Wenn man die kürzeste-Wege-Funktion (Wer-kennt-wen-Problem, siehe Quellcode-Auszüge 1-4) zugrunde legt, liegt die normale Lesegeschwindigkeit bei knapp 2.000 Kantenschritten pro Millisekunde. Dabei soll Neo4j bereits bei kleinen Graphen von lediglich 1.000 Knoten rund 1.000-mal schneller sein als MySQL. Dieser Wert soll exponentiell mit der Größe des Graphen steigen. Die Ursache dafür sieht Peter Neubauer, ein Mitbegründer der Neo Technology, in der konstanten Abfragegeschwindigkeit, die unabhängig von der Größe/Tiefe des Graphen ist. Einen anderen Vergleichsansatz haben Vicknair, et.al. angestrebt. Sie haben drei verschiedene Anfragen abgesetzt und konnten dabei den Faktor 10 zu Gunsten von Neo4j feststellen [8].

Transaktionen in Neo4j

Der Kernel von Neo4j ist ein extrem schneller Graphenmotor, der alle Charakteristika aufweist, die man von einem RDBMS erwartet - 2-phase-commit transactions, XA compliance, ACID, etc.

Im Gegensatz zu den Leseoperationen müssen alle Schreiboperationen in Datenbanktransaktionen ausgeführt werden. Die Transaktionen sind dabei durch die Anzahl der möglichen Threads beschränkt und können als sogenannte „flat nested transactions“ verschachtelt werden. Jede Schreiboperation setzt eine Sperre auf den zu schreibenden Bereich, wodurch die Datenintegrität gewährleistet wird. Dabei ist es

```
public class Neo4j_Abb13
{
    public enum RelTypes implements RelationshipType
    {
        ist_mitglied_von,
        studiert_an,
        ist_befreundet_mit,
        liebt,
        beinhaltet,
        FRIEDA_NODE
    }
    ...
}
```

Quellcode 1: Neo4j - Festlegung von Beziehungen.

```
...
private static void createNodespace()
{
    Transaction tx = graphDb.beginTx();
    try{
        // Create the nodes who are necessary
        Node frieda = graphDb.createNode();
        frieda.setProperty( "name", "Frieda" );
        frieda.setProperty( "type", "person" );

        // connect Frieda to the reference node
        Node referenceNode = graphDb.getReferenceNode();
        referenceNode.createRelationshipTo( frieda,
            RelTypes.FRIEDA_NODE );

        Node paul = graphDb.createNode();
        paul.setProperty( "name", "Paul" );
        paul.setProperty( "type", "Person" );
        Relationship FPrel = frieda.createRelationshipTo(paul,
            RelTypes.ist_befreundet_mit);
        FPrel = paul.createRelationshipTo(frieda, RelTypes.
        liebt);
    }
    ...
}
```

Quellcode 2: Neo4j - NodeSpace-Festlegung.

```
...
public void printFriedaFriends() throws Exception
{
    Node friedaNode = getFriedaNode();
    System.out.println( friedaNode.getProperty( "name" ) +
        "'s friends:" );
    Traverser friendsTraverser = getFriends( friedaNode );
    int numberOfFriends = 0;
    for ( Node friendNode : friendsTraverser )
    {
        System.out.println( "At depth "
            + friendsTraverser.currentPosition()
            .depth() + " => " + friendNode.getProperty( "name" );
        numberOfFriends++;
    }
    assertEquals( 4, numberOfFriends );
    System.out.println("Number of Friends: "+numberOfFriends);
}
...
}
```

Quellcode 3: Neo4j - Freunde finden.

```
Ausgabe: Frieda's friends:
At depth 1 => Paul
At depth 2 => Hans
```

Quellcode 4: Neo4j - Ausgabe der Freunde.

enorm wichtig, dass jede einzelne Transaktion durch ein `commit` oder `rollback` abgeschlossen wird. Ansonsten kann die Sperre nicht entfernt werden und es kommt bei konkurrierendem Zugriff zu einem Deadlock. Diese Deadlocks erkennt Neo4j und gibt eine Fehlermeldung aus. Die Transaktion wird dabei auto-

matisch auf „Rollback“ gesetzt und der Benutzer kann die Sperre durch Beenden der Transaktion lösen (es wird keine Änderung durchgeführt).

Transaktionsmodell

Das Transaktionsmodell bei Neo4j sieht wie folgt aus:

1. Beginn der Transaktion
2. Durchführen der Operation (`Insert`, `Update`, `Delete`)
3. Markieren der Transaktion (`commit` oder `rollback`)
4. Beenden der Transaktion

Jede Transaktion wird im Hauptspeicher geladen. Das bedeutet, dass Transaktionen die zu groß für den Hauptspeicher sind in kleinere Transaktionen aufgeteilt werden müssen.

Begrenzungen von Neo4j

Abgesehen von den Einschränkungen, die sich aus der Graphenstruktur ergeben und sich somit auf die Anwendungsfälle wie bspw. kürzeste Wege, maximaler Fluss, PageRank, etc. beziehen, ist auch diese Form der Datenbank durch ihre Speicherverwaltung eingeschränkt. Dazu zählen insbesondere die Adressräume der Knoten, Beziehungen und deren Eigenschaften, die eine Instanz der Datenbank verarbeiten kann. Weiter ist die Limitierung der Adressräume von der jeweiligen Implementierung abhängig. Neo4j ist beispielsweise in der Größe auf ca. 34 Milliarden Knoten in einer Datenbankinstanz limitiert [6].

Fazit

Neo4j findet häufig dort Anwendung, wo Systeme soziale Netzwerke, geografische Daten oder z.B. Empfehlungen abbilden müssen. Dies belegen die seit 2003 produktiven Einsätze der Graphendatenbank. Eine Vorhersage des Datenaufkommens sollte im Vorfeld grob abgeschätzt werden, um ein geeignetes technologisches Umfeld als Grundgerüst nehmen zu können.

Neo Technology bietet mit dem „Hardware Sizing Calculator“ ein Werkzeug an, mit dem man die benötigte Hardware ermitteln kann [9]. Mit dem High Availability Clustering bietet Neo4j zudem einen Master-Failover-Modus an, welcher in weiteren Versionen um dynamisches Sharding erweitert werden soll und damit die Skalierung der Datenbank vereinfacht.



Dominic Oberländer
(info@ordix.de)

Glossar

Denormalisierung

Unter einer Denormalisierung versteht man die bewusste Rücknahme einer Normalisierung zum Zweck der Verbesserung des Laufzeitverhaltens einer Datenbankanwendung.

GDB

Graph Database - Graphendatenbanken speichern ihre Daten in einer Netzstruktur, in der die einzelnen Informationen durch Knoten repräsentiert werden und die spezielle Eigenschaften besitzen.

PGM

Ein Property-Graph-Modell unterscheidet sich von einem einfachen Modell dahingehend, dass die Knoten und Kanten aus Objekten mit darin eingebetteten Eigenschaften (Properties) bestehen.

RDF

Ein Resource Description Framework ist ein Datenmodell mit einer definierten formalen Semantik, das auf gerichteten Graphen basiert. Daten in RDF sind Aussagen über Ressourcen.

RDBMS

Ein Relation Database Management System ist ein Datenbank-Managementsystem von relationalen Datenbanken, das Inhalte in Beziehungen zueinander stellt.

Sharding

Unter Sharding versteht man die horizontale Fragmentierung aller Datensätze einer Relation auf mehrere Tabellen.

STP

Das Spanning Tree Protocol ist ein Redundanzverfahren zur Schleifenunterdrückung und damit zur Pfadoptimierung in Netzwerken.

Links

- ▶ [1] Edlich, S. / Friedland, A. / Hampel, J. / Brauer, B.: NoSQL: Einstieg in die Welt nichtrelationaler Web 2.0 Datenbanken, München (2010), Hanser Verlag
- ▶ [2] Die Anfänge der Graphentheorie: <http://www.mathe.tu-freiberg.de/~hebis/cafegraphentheorie.html>
- ▶ [3] Rodriguez, M. A. / Neubauer, P.: Constructions from Dots and Lines. In: CoRR abs/1006.2361 (2010)
- ▶ [4] Florescu, D.: Managing Semi-Structured Data. In: ACM Queue (2005), Nr. 3, S. 20–24
- ▶ [5] Codd, E. F.: The relational model for database management, 2. Aufl., München (1990), Addison-Wesley Verlag
- ▶ [6] Neo4j Team: The Neo4j Manual v1.5, Version: 2011: <http://docs.neo4j.org/pdf/neo4j-manual-stable.pdf>
- ▶ [7] Webseite zu SPARQL: <http://www.w3.org/TR/rdf-sparql-query/>
- ▶ [8] Vicknair, C. / Macias, M. / Zhao, Z. / Nan, X. / Chen, Y. / Wilkins, D.: A comparison of a graph database and a relational database: a data provenance perspective. In: Proceedings of the 48th Annual Southeast Regional Conference, New York, NY, USA: ACM, 2010 (ACM SE '10), S. 42:1–42:6
- ▶ [9] Hardware Sizing Calculator: <http://neotechnology.com/calculator/trial.html>



Rätsel

Larry und sein Geburtstagskalender

Larry hat die Geburtstage aller seiner Kollegen gesammelt. Jetzt möchte er ein Programm schreiben, welches ihm alle Geburtstagskinder, die heute oder in den nächsten 6 Tagen Geburtstag haben, auflistet.

Larry und die Schaltjahrproblematik

Larry hat noch keine Ahnung, wie er dieses Programm umsetzen soll. Die Programmiersprache ist ihm egal. Wichtig ist ihm aber, dass die Geburtstagskinder vom 29.02. natürlich nicht vergessen werden und über den Jahreswechsel soll das Programm auch funktionieren.

Er programmiert in den unterschiedlichen Sprachen, aber er kommt zu keinem Ergebnis. Können Sie Larry auf die Sprünge helfen?

Diesen kleinen Datenbestand nimmt Larry zum Testen.

```
'01.01.1970', 'Martin'
'29.02.1980', 'Kevin'
'31.12.1966', 'Bärbel'
'01.07.1952', 'Theo'
```

Können Sie Larry helfen?

Larry freut sich auf Ihren Lösungsvorschlag! Senden Sie bis zum **10. November 2013** Ihre Antwort an kniffel@ordix.de.

Das Rätsel aus der letzten Ausgabe wurde gelöst - Larry's Zeichenkette konnte gespeichert werden

Larry bedankt sich bei allen Einsendern. Viele hatten den richtigen Lösungsansatz, übersahen aber einen Bug, der je nach Datenbankversion auftreten kann.

Larry bedankt sich mit einer kleinen Aufmerksamkeit für die richtige Lösung bei: Gabriele Herold

Lösung Larry Rätsel 2/2013:

Beim `SELECT ... INTO` wechselt der Kontext von PL/SQL nach SQL, dort stehen für den `VARCHAR2` aber nur 4000 Zeichen zur Verfügung. Daher die Fehlermeldung.

Die folgende Variante funktioniert, liefert aber aufgrund eines Bugs unter Umständen weniger als 32.000 Zeichen zurück:

```
-- Check out bug #5896994 VALUE TRUNCATED WHEN READING >=
8K CHUNKS FROM CLOB USING DBMS_LOB.SUBSTR

DECLARE
    v_config VARCHAR2(32000);
    v_clob    xml_config.config%TYPE;
BEGIN
    SELECT config
    INTO    v_clob
    FROM    xml_config
    WHERE   id = 1;
    v_config := dbms_lob.substr( v_clob, 32000, 1 );
END;
/
```

Die folgende Lösung funktioniert dann fehlerfrei:

```
DECLARE
    v_config VARCHAR2(32000);
    v_clob    xml_config.config%TYPE;
    v_length  NUMBER := 32000;
BEGIN
    SELECT config
    INTO    v_clob
    FROM    xml_config
    WHERE   id = 1;
    dbms_lob.read( v_clob, v_length , 1, v_config );
END;
/
```

Bild: © istockphoto.com | Computer Geek: Two Thumbs Up | sdominick

Datenbanken

Oktober - Dezember 2013

DB-DB-03	Datawarehouse Grundlagen	3 Tage	1.290,00 €	28.10.2013
DB-ORA-01	Oracle SQL	5 Tage	1.890,00 €	21.10.2013
DB-ORA-01A	Oracle SQL für Experten	3 Tage	1.290,00 €	28.10.2013
DB-ORA-02	Oracle Datenbankprogrammierung mit PL/SQL Grundlagen	5 Tage	1.890,00 €	14.10.2013 18.11.2013
DB-ORA-34	Oracle Datenbankprogrammierung mit PL/SQL Aufbau	5 Tage	1.890,00 €	09.12.2013
DB-ORA-42	Oracle PL/SQL Tuning	3 Tage	1.890,00 €	07.10.2013 02.12.2013
DB-ORA-38	Objektorientierung in Oracle	5 Tage	1.990,00 €	auf Anfrage
DB-ORA-03	Oracle Datenbankadministration Grundlagen	5 Tage	1.990,00 €	14.10.2013 11.11.2013 09.12.2013
DB-ORA-04	Oracle Datenbankadministration Aufbau	5 Tage	1.990,00 €	21.10.2013
DB-ORA-32	Oracle Backup und Recovery mit RMAN	5 Tage	1.990,00 €	07.10.2013 02.12.2013
DB-ORA-07	Oracle Tuning und Monitoring	5 Tage	1.990,00 €	02.12.2013
DB-ORA-41	Oracle AWR und ASH Analyse und Interpretation	3 Tage	1.290,00 €	21.10.2013
DB-ORA-11	Oracle Troubleshooting Workshop	5 Tage	1.990,00 €	auf Anfrage
DB-ORA-08	Oracle 11gR2 RAC und Grid Infrastructure	5 Tage	2.050,00 €	25.11.2013
DB-ORA-15	Oracle 11g Neuheiten	5 Tage	1.990,00 €	07.10.2013
DB-ORA-33A	Oracle Security	4 Tage	1.590,00 €	11.11.2013
DB-ORA-31	Oracle Data Guard	4 Tage	1.590,00 €	04.11.2013
DB-ORA-35	Oracle Grid Control	3 Tage	1.290,00 €	14.10.2013
DB-ORA-40	Oracle Capacity Planning	3 Tage	1.290,00 €	28.10.2013
DB-ORA-46	Oracle APEX Anwendungsentwicklung Grundlagen	3 Tage	1.290,00 €	04.11.2013
DB-ORA-47	Oracle APEX Anwendungsentwicklung für Fortgeschrittene	3 Tage	1.290,00 €	07.10.2013 09.12.2013
DB-ORA-48	Oracle Golden Gate	3 Tage	1.290,00 €	04.10.2013 09.12.2013
DB-ORA-49	Oracle 12c Neuheiten	5 Tage	1.990,00 €	25.11.2013
DB-INF-01	IBM Informix SQL	5 Tage	1.790,00 €	04.10.2013
DB-INF-02	IBM Informix Administration	5 Tage	1.990,00 €	04.11.2013
DB-INF-04	IBM Informix Backup und Recovery	3 Tage	1.290,00 €	18.11.2013
DB-INF-03	IBM Informix Tuning und Monitoring	5 Tage	1.990,00 €	11.11.2013
DB-DB2-01	IBM DB2 für Linux/Unix/Windows SQL Grundlagen	5 Tage	1.890,00 €	21.10.2013
DB-DB2-02	IBM DB2 für Linux/Unix/Windows Administration	5 Tage	1.990,00 €	04.11.2013
DB-DB2-05	IBM DB2 für Linux/Unix/Windows Monitoring und Tuning	3 Tage	1.290,00 €	auf Anfrage
DB-MY-01	MySQL Administration	3 Tage	1.190,00 €	02.12.2013
MS-SQL-01	Querying SQL Server 2012	5 Tage	1.890,00 €	04.11.2013
MS-SQL-02	Administering Microsoft SQL Server 2012	5 Tage	1.890,00 €	18.11.2013
MS-SQL-03	Schreiben von Abfragen mit MS SQL Server 2008 T-SQL	3 Tage	1.190,00 €	14.10.2013
MS-SQL-04	Verwalten einer MS SQL Server 2008 Datenbank	5 Tage	1.890,00 €	25.11.2013

Entwicklung

Oktober - Dezember 2013

P-PHP-01	PHP Programmierung Grundlagen	5 Tage	1.690,00 €	25.11.2013
P-PERL-01	Perl Programmierung Grundlagen	5 Tage	1.690,00 €	18.11.2013
P-PERL-02	Perl Programmierung Aufbau	5 Tage	1.690,00 €	02.12.2013
P-UNIX-01	Shell, Awk und Sed	5 Tage	1.690,00 €	auf Anfrage
P-UNIX-01A	Awk Intensiv-Workshop	3 Tage	1.190,00 €	28.10.2013
P-XML-01	Einführung in XML	3 Tage	1.190,00 €	11.11.2013
P-XML-03	Oracle und XML	3 Tage	1.190,00 €	auf Anfrage

In Kürze erscheint unser neues **Seminarprogramm 2014** mit über 100 Seminaren und über 340 Terminen im nächsten Jahr. Reservieren Sie sich schon jetzt die Termine!

Sie erhalten noch kein Seminarprogramm? Dann kontaktieren Sie uns über www.ordix.de

Informationen und Anmeldung

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

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

**Online-Anmeldung,
 aktuelle Seminarinhalte
 und Termine unter:**
<http://training.ordix.de>



Unser Seminarstandort ist Wiesbaden. Die Preise gelten pro Seminar pro Teilnehmer in Euro zzgl. ges. MwSt., Inhouse-Preise auf Anfrage.

Betriebssysteme und Monitoring

Oktober - Dezember 2013

BS-01	Unix/Linux Grundlagen für Einsteiger	5 Tage	1.690,00 €	11.11.2013
BS-25	Unix Aufbau-seminar für Datenbank- und Applikationsbetreuer	5 Tage	1.790,00 €	09.12.2013
BS-02	Linux Systemadministration	5 Tage	1.690,00 €	18.11.2013
BS-09	Linux Hochverfügbarkeits-Cluster	3 Tage	1.290,00 €	04.11.2013
BS-19	Linux Cluster mit Pacemaker	3 Tage	1.190,00 €	25.11.2013
BS-03	Solaris 10 Systemadministration Teil I	5 Tage	1.990,00 €	25.11.2013
BS-04	Solaris 10 Systemadministration Teil II	5 Tage	1.990,00 €	07.10.2013 09.12.2013
BS-06	Solaris 10 für erfahrene Unix/Linux-Umsteiger	5 Tage	1.990,00 €	02.12.2013
BS-24	Solaris 11 Administration Neuheiten	5 Tage	1.990,00 €	28.10.2013
BS-18	Solaris Virtualisierung mit ZFS und Container (Zonen)	5 Tage	1.990,00 €	14.10.2013
BS-23	Solaris Virtualisierung mit LDOM	3 Tage	1.290,00 €	11.11.2013
AIX-01	IBM AIX Systemadministration Grundlagen	5 Tage	1.990,00 €	18.11.2013
AIX-02	IBM AIX Installation, Backup und Recovery mit NIM	3 Tage	1.290,00 €	auf Anfrage
AIX-03	Analyse v. komplexen AIX Performance Problemen	5 Tage	2.300,00 €	07.10.2013
SM-NAG-01	Systemüberwachung mit Nagios Grundlagen	3 Tage	1.190,00 €	21.10.2013
SM-NAG-02	Systemüberwachung mit Nagios Aufbau	2 Tage	890,00 €	24.10.2013

Projekt-/IT-Management

Oktober - Dezember 2013

PM-01	IT-Projektmanagement - Methoden und Techniken	5 Tage	1.990,00 €	14.10.2013
PM-06	System. Projektmanagement - Projektteams souverän führen	4 Tage	1.850,00 €	18.11.2013
PM-08	Agiles Projektmanagement mit SCRUM	2 Tage	1.100,00 €	02.12.2013
PM-08-Z	SCRUM Praxis und Zertifizierung	1 Tag	250,00 €	04.12.2013
PM-09	Capability Maturity Model Integration (CMMI)	2 Tage	1.100,00 €	auf Anfrage
PM-10	Kennzahlen der IT	2 Tage	1.100,00 €	21.10.2013
PM-07	Krisenmanagement in Projekten	2 Tage	1.100,00 €	24.10.2013
PM-05	IT-Projektcontrolling	3 Tage	1.290,00 €	28.10.2013
PM-11	Konfliktmanagement	2 Tage	1.100,00 €	auf Anfrage
MGM-02	IT-Architekturen	3 Tage	1.650,00 €	04.11.2013
MGM-01	E-Business	2 Tage	1.100,00 €	auf Anfrage
MGM-07	IT-Strategien effizient entwickeln	2 Tage	1.100,00 €	28.10.2013
MGM-03	IT-Management	5 Tage	1.990,00 €	09.12.2013
MGM-05	IT-Risikomanagement	3 Tage	1.650,00 €	21.10.2013
MGM-04	IT-Prozessmanagement	3 Tage	1.650,00 €	07.10.2013

Java-JEE

Oktober - Dezember 2013

E-UML-01	Anforderungsanalyse mit UML	3 Tage	1.190,00 €	07.10.2013
E-SWA-01	Softwarearchitekturen	5 Tage	1.890,00 €	02.12.2013
OO-01	Einführung in die Objektorientierte Programmierung	3 Tage	1.190,00 €	28.10.2013 09.12.2013
P-JAVA-01	Java Programmierung Grundlagen	5 Tage	1.690,00 €	14.10.2013
P-JAVA-03	Java Programmierung Aufbau	5 Tage	1.690,00 €	25.11.2013
P-JAVA-02	Java GUI Entwicklung mit Swing	5 Tage	1.690,00 €	auf Anfrage
P-JEE-01	JEE für Entscheider	1 Tag	590,00 €	25.11.2013
P-JEE-02	Einführung in JEE	3 Tage	1.290,00 €	auf Anfrage
P-JEE-03A	JSP und Servlet Programmierung	5 Tage	1.590,00 €	11.11.2013
P-JEE-04	EJB Programmierung	5 Tage	1.590,00 €	auf Anfrage
P-JEE-05	Web-Anwendungen mit JavaServer Faces (JSF)	5 Tage	1.590,00 €	04.11.2013
P-JEE-06	Entwickeln mit dem Spring-Framework	3 Tage	1.190,00 €	auf Anfrage
INT-05	Java Web Services	3 Tage	1.190,00 €	18.11.2013
J-HIB-01	Hibernate und die Java Persistence API	5 Tage	1.690,00 €	auf Anfrage
P-JEE-08	Java Performance Tuning	3 Tage	1.290,00 €	09.12.2013

Web- und Applikations-Server

Oktober - Dezember 2013

INT-04	Apache Web-Server Installation und Administration	3 Tage	1.190,00 €	21.10.2013
INT-07	Tomcat Konfiguration und Administration	3 Tage	1.190,00 €	25.11.2013
INT-08	WebSphere Application Server Installation und Administration	3 Tage	1.390,00 €	04.11.2013
INT-11	Administration und Konfiguration für JBoss 5	3 Tage	1.190,00 €	18.11.2013
INT-11_7	Administration und Konfiguration für JBoss 7	3 Tage	1.190,00 €	25.11.2013

NetApp und Data ONTAP 8.x

Drei Dinge braucht das Rechenzentrum: Speicher, Speicher, Speicher

Nicht nur die NSA (National Security Agency) kämpft mit einem rasant wachsenden Datenvolumen innerhalb ihrer IT-Infrastrukturen. Seien es große Datenbanken oder kleinere Benutzerdateisysteme - die Anforderungen an das Speichermanagement werden immer komplexer und anspruchsvoller. Nur viel und schneller Plattenplatz alleine reicht nicht mehr aus. Themen wie Virtualisierung, Verfügbarkeit und unterbrechungsfreie Skalierbarkeit nehmen heute im Speicherumfeld eine wichtige Rolle ein.

Daher wollen wir uns in dieser und zukünftigen Ausgaben der ORDIX news anschauen, was NetApp den Anwendern mit seinen Systemen und dem dazugehörigen Betriebssystem Data ONTAP bietet.

Aus zwei mach eins

Das aktuelle Betriebssystem Data ONTAP 8 ist das Nachfolgeprodukt von zwei Varianten, die bisher getrennt voneinander entwickelt wurden. Nun werden in einem Produkt die Funktionalitäten von ONTAP 7.x und ONTAP GX unter einer Code-Basis vereint.

Wie bei der 7er Version bietet der 7-Mode dieselbe Unterstützung der Benutzeroberflächen und der Management-Werkzeuge. Durch diesen Modus ist eine Migration bestehender NetApp-Versionen unterbrechungsfrei möglich.

Der Cluster-Mode beinhaltet zudem die Hochverfügbarkeitsfunktionalitäten der GX-Variante.

Es darf aber auch noch etwas mehr sein

Data ONTAP 8 aber lediglich als eine Zusammenfassung von zwei separaten Varianten anzusehen, würde ihm nicht gerecht werden. Daher wollen wir im Folgenden einen Überblick über die wichtigsten Funktionserweiterungen geben.

Mit dem Flash Cache wurde eine intelligente Cache-Speicherung eingeführt. Dadurch können unterschiedliche Medien (Flash-Cache-Karten, Laufwerke) kombiniert werden. Trotz des Einsatzes von kostengünstigem Speicher muss kein Performance-Verlust hingenommen werden.

Die Datenkomprimierung kann individuell für jedes einzelne Volume aktiviert werden (Storage Efficiency). Damit werden Dateien und LUNs während des Schreibvorgangs auf dem Laufwerk komprimiert. Durch eine Deduplizierung, also das Eliminieren redundanter Datenblöcke, lässt sich die Einsparung an Speicherplatz noch weiter optimieren.

Mit der Funktion DataMotion können Volumes unterbrechungsfrei zwischen Aggregaten auf dem selben Controller verschoben werden. Dies bringt eine Reihe von Vorteilen mit sich, die wir in einem der kommenden Artikel näher erläutern werden.

Die Unified-Storage-Architektur bietet sich für die Konsolidierung heterogener IT-Umgebungen an.



Abb. 1: NetApp FilerView.



Abb. 2: Netapp OnCommand System Manager.

SAN- und NAS-Umgebungen lassen sich jetzt auf einen zweistelligen Petabyte (PT)-Bereich skalieren.

Durch den Einsatz von 64-Bit-Aggregaten wird die bisherige 16-TB-Grenze überwunden und es sind nun Volumes mit einer Größe bis zu 100 TB möglich.

Im Rahmen der Administration hat sich NetApp von FilerView verabschiedet (siehe Abbildung 1). Als Nachfolger kommt jetzt der OnCommand System Manager zum Einsatz (siehe Abbildung 2).

Fazit

Der vorliegende Artikel gibt zunächst einen Überblick über die wichtigsten Funktionserweiterungen, die Data ONTAP 8 mit sich bringt. In nachfolgenden Artikeln werden wir die einzelnen Funktionalitäten sowie die beiden Modi intensiver beleuchten.



Peter Berend
(info@ordix.de)

Glossar

Aggregate

Aggregate sind mehrere zusammengefasste Platten im RAID-Verbund.

FilerView

FilerView ist eine browser-basierte, primäre Oberfläche für administrative Aufgaben auf allen NetApp Storage-Systemen – von der Installation bis zur vollständigen Konfiguration.

iSCSI

Das internet Small Computer System Interface ermöglicht die Nutzung des SCSI-Protokolls über TCP.

LUN

Die Logical Unit Number ist ein virtueller Datenträger, der über iSCSI eingebunden wird.

OnCommand System Manager

Der OnCommand System Manager ist eine browser-basierte Oberfläche für administrative Aufgaben auf dem Clustered Data ONTAP Betriebssystem.

RAID

Redundant Array of Independent Disks - Ein RAID-System dient zur Organisation mehrerer physischer Festplatten eines Computers zu einem logischen Laufwerk. Dadurch wird eine höhere Datenverfügbarkeit beim Ausfall einzelner Festplatten erreicht.

Quelle

- [1] NetApp Homepage:
<http://www.netapp.com>

Wie nützlich sind Zertifizierungen?

Unternehmen setzen zunehmend auf projektorientierte Organisationsformen zur Bewältigung und schnellen Umsetzung von organisatorischen Veränderungen. Dazu zählen insbesondere bereichsübergreifende Produkt- und IT-Entwicklungen. Die Verantwortlichen erwarten, um die gestellten Ziele zu erreichen, für ihre Projekte eine fundierte und realistische Planung, ein präventives Risikomanagement und ein Changemanagement auf Grundlage einer professionellen Projektmanagement-Kompetenz.

Dennoch treten in IT-Projekten regelmäßig dieselben Projektmanagement-Fehler auf. Das Projekt erreicht nicht die beabsichtigten Ziele im geplanten Umfang oder scheitert möglicherweise komplett. Dieser Artikel stellt dar, wie Projektmanagement-Methoden und deren Zertifizierungen den Erfolg von Projekten unterstützen können.

Die Anforderungen steigen

Die häufigsten Probleme in IT-Projekten ergeben sich durch die nachfolgenden Aspekte:

- Eine Methodik ist für die Durchführung von Projekten nicht durchgehend etabliert.
- Der Umfang eines Projektes ist nicht eindeutig definiert und von den Stakeholdern bestätigt worden.
- Die Projektorganisation und Verantwortlichkeiten sind nicht klar definiert.
- Das Projekt verfügt nicht über die erforderlichen fachlichen Qualifikationen oder die notwendige Ressourcen.
- Der Terminplan des Projektes ist unrealistisch, erlaubte Toleranzen sind nicht bestimmt.
- Das Budget des Projektes ist nicht realistisch angesetzt und erlaubte Toleranzen sind nicht bestimmt.
- Die Abhängigkeiten zu konkurrierenden Projekten/ Linienaktivitäten sind nicht ausreichend berücksichtigt.
- Ein Changemanagement fehlt oder ist unzureichend.
- Problem-, Risiko- und Qualitätsmanagement fehlen oder sind unzureichend
- Der Status quo des Projektes ist nicht transparent.
- Notwendige Entscheidungsbedarfe werden nicht klar gestellt oder zu spät adressiert.

Diese Probleme bewirken einen steigenden Bedarf an erfahrenen und gut ausgebildeten Projektmanagern. Die Erwartung liegt dabei auf der Planung und Realisierung der wichtigsten operativen und strategischen IT-Vorhaben. Dies führt wiederum zu steigenden Anforderungen an die Projektmanager.

Verstärkt nachgefragt werden dabei Projektmanager, die neben einer langjährigen Erfahrung bei der Realisierung von IT-Projekten auch die etablierten Vorgehensmodelle nach „Best Practice“ kennen. Zusätzlich sollen sie möglichst über ein geprüftes und zertifiziertes Projektmanagement-Know-how verfügen, um eine erfolgreiche Planung und Realisierung der Vorhaben gewährleisten zu können.

Bei der Auswahl von kompetenten und erfahrenen Projektmanagern müssen sich die Unternehmen daher fragen:

- Welche Zertifizierungen existieren im Projektmanagement?
- Welchen Nutzen/Chancen bieten die Zertifizierungen?
- Welche Grenzen haben die Zertifizierungen?
- Welche Projektmanagement-Methode passt zu unserem Unternehmen bzw. zu den verfolgten Zielen?
- Welche Zertifikate passen zu den im Unternehmen

- eingesetzten Standards?
- Wie wichtig ist die Kenntnis von nationalen und internationalen Methodenstandards?
- Wie wichtig ist ein sofort und praktisch einsetzbares Vorgehensmodell?
- Welche Erfahrungen sind für welches Unternehmensfeld als Voraussetzung wichtig?

Die wichtigsten PM-Zertifizierungen

Bis dato haben sich drei Institutionen für Projektmanagement-Zertifizierungen auf dem deutschen Markt etabliert (siehe Abbildung 1):

1. Project Management Institute (PMI): PMBOK Guide
2. Internationale Project Management Association (IPMA): 4 Level Certification System
3. Office of Government Commerce (OGC): PRINCE 2

Alle Anbieter ermöglichen dabei eine umfassende Projektmanagement-Qualifizierung und -Zertifizierung. Sie haben ein spezifisches Zertifizierungssystem und adressieren ähnliche Zielgruppen. Allerdings unterscheidet sie das jeweilige Prüfungsvorgehen.

Zertifizierungsstufen

Im Folgenden werden die einzelnen Projektmanagement-Zertifizierungen erläutert. Eine zusammenfassende Darstellung zeigt auch die Abbildung 2.

PMI

Das PMI bietet ein Zertifizierungsverfahren auf Basis des PMBOK Guide, welcher Vorgaben zum methodischen Vorgehen und terminologische Vorgaben umfasst. Der PMBOK Guide ist in der USA eine anerkannte „De-facto-Norm“. Dabei sind 42 Prozesse definiert, die in einer Matrix verschiedene Prozessgruppen und Wissensgebieten zugeordnet sind. Für die Kommunikation und die Zusammenarbeit im Projekt wird eine einheitliche Grundlage definiert (global akzeptierter Standard für Formulierungen und der Umgang mit Projektmanagement-Themen).

Das PMI ist im Ausland und bei international agierenden Firmen weit verbreitet. Dabei ist es stark prozessorientiert, um die Transparenz von Projekten im internationalen Kontext gewährleisten zu können.

IPMA

Die IPMA legt ein vierstufiges Zertifizierungssystem zugrunde. Die Gewährleistung des Qualitätsniveaus dieses Systems liegt in Deutschland in der Verantwortung der Zertifizierungsstelle der Deutschen Gesellschaft für Projektmanagement e.V. (GPM). Die vier Ebenen

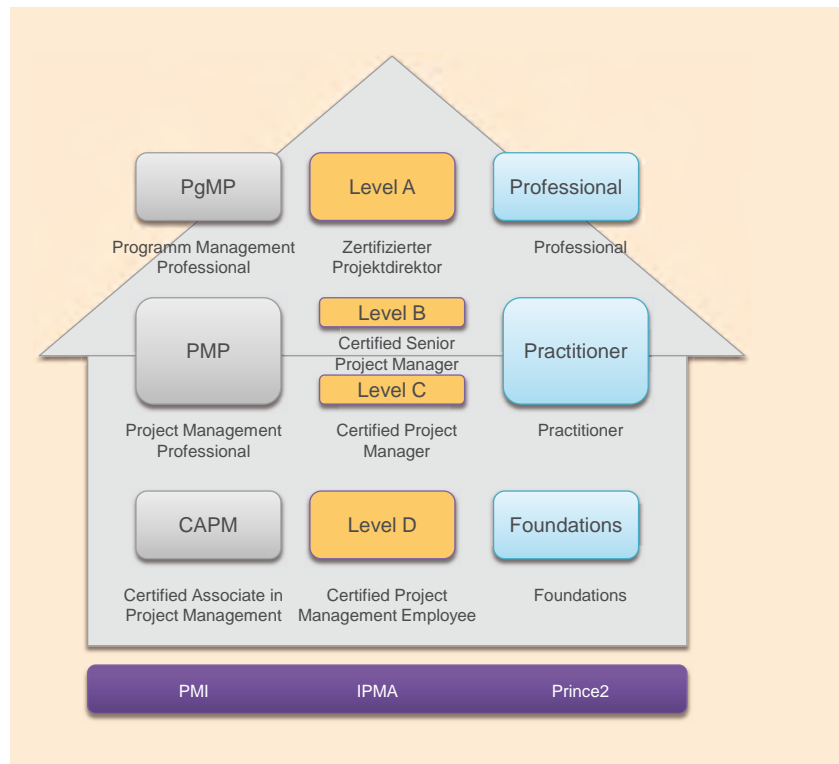


Abb.1: Etablierte Projektmanagement-Zertifizierungen.

Zertifizierungsstufe	Zertifizierung PMI	Zertifizierung IPMA	Zertifizierung PRINCE2
Projektmanager	PgMP	Level A	Professional
Senior Projektmanager	PMP	Level B	Practitioner
Projektmanager	PMP	Level C	Practitioner
Projektmitarbeiter	CAPM	Level D	Foundations

Abb. 2: Einstufungen der Projektmanagement-Zertifizierungen.

definieren Einsatzgebiete bzw. Funktionsbereiche in der Projektmanagement-Praxis. Auf allen Ebenen wird im Rahmen der Zertifizierung das Projektmanagement-Wissen überprüft und bewertet.

Ab der Zertifizierungsebene C werden zusätzlich die Anwendungserfahrungen und die sozialen Fähigkeiten im persönlichen Verhalten überprüft.

Dabei wird für jeden Zertifikatslevel ein zweiköpfiges Sachverständigen-Team eingesetzt, welches das Wissen, die Erfahrung und die Persönlichkeit des Aspiranten auf den Prüfstand stellt. Das für die Zertifizierung erforderliche Wissen und die Einordnung auf den jeweiligen Zertifikatslevel ist in der ICB 3.0 (IPMA Kompetenzrichtlinie) beschrieben.

Inhalte	PMI	IPMA	PRINCE2
Zertifizierungsgrundlage	Gesammeltes Wissen mit dem Fokus auf Prozesse und Methoden im Projektmanagement	Gesammeltes Wissen mit dem Fokus auf kompetenzbasiertem Projektmanagement	Projektmanagement mit dem Blick auf Prozesse und Ergebnisse
Bekanntheitsgrad	Weltweit bekannt und in den USA im Grunde Pflicht	Besonders in Europa sehr verbreitet und wächst	In Großbritannien entstanden und im Grunde Pflicht, wächst stetig in Europa
Prüfungen	Schriftliche Multiple-Choice-Tests	Schriftliche Tests, Workshops mit Assessment	Schriftliche Multiple-Choice-Tests, Workshops mit Assessment
Re-Zertifizierung	Für PMP nach 3 Jahren	Für alle Level nach 5 Jahren	Für Practitioner und Professional nach 3-5 Jahren

Abb. 3: Gültigkeit der Zertifizierungen.

PRINCE2

Projects in Controlled Environments (PRINCE2) ist eine prozessorientierte und frei skalierbare Projektmanagement-Methode, welche die Ergebnisse und den Nutzen eines Projektes in den Vordergrund stellt. Sie bildet einen strukturierten Rahmen um das jeweilige Projekt und gibt den Mitgliedern des Projektmanagement-Teams anhand des Prozessmodells konkrete Handlungsempfehlungen und Ergebnistypen für jede Projektphase.

Die Entstehung und Weiterentwicklung der Methode erfolgt nach dem Best-Practice-Gedanken (PRINCE2-standards: derzeit PRINCE2:P3M3, 2009). Seit 2012 ist weiterführend, im Rahmen eines 2,5-tägigen Assessment, der Erwerb einer PRINCE2 Professional-Zertifizierung möglich.

Gültigkeitsdauer der Zertifizierungen

Bezogen auf die Gültigkeitsdauer verfolgen IPMA, PMI und PRICE2 vergleichbare Ansätze.

Nach Ende der Gültigkeitsdauer muss der Zertifikatsinhaber die Leistungsnachweise für eine Verlängerung der Zertifizierung erneut prüfen lassen (siehe Abbildung 3).

Welchen Nutzen hat eine Projektmanagement-Zertifizierung?

Die positive Wirkung von anerkannten Zertifizierungen besteht in zweierlei Hinsicht. Erstens kann sie ein Qualitätsmerkmal für eine Auswahl von Projektmanagern durch Auftraggeber und Kunden sein. Zum anderen kann sie eine Option innerhalb eines Unternehmens darstellen, da sie eine Einschätzung von vorhandenen

Qualifikationen ermöglicht und die Weiterentwicklung von internen Potenzialen aufzeigen kann.

Durch eine Zertifizierung der Projektmanager wird es zudem ermöglicht, im Unternehmen eine gemeinsame Herangehensweise und gemeinsame Sprachstandards in den Projekten zu schaffen (z.B. analog zu „Best Practice“ in ITIL). Mögliche Konflikte können frühzeitig identifiziert und strukturiert bearbeitet werden. Die Kommunikation mit Projektbeteiligten, Zulieferern und schließlich dem Management kann systematischer und effektiver erfolgen.

Eine Zertifizierung dient somit der Zielerreichung von Projekten, beginnend bei der Planung, über die Realisierung und ein effizientes Changemanagement bis hin zu einem transparenten Reporting gegenüber dem Management. Die fortlaufende und transparente Bewertung des Fortschrittes und des zu erwartenden Nutzens (Wirtschaftlichkeit und Sinnhaftigkeit) von Projekten ist dabei von wesentlicher Bedeutung.

Der explizite Nutzen der drei vorgestellten Projektmanagement-Zertifizierungen liegt in der objektiven und neutralen Bestätigung der Projektmanagement-Kompetenzen.

- PMI befasst sich in erster Linie mit der Standardisierung von Prozessen, bewährten Methoden und Begrifflichkeiten.
- IPMA hat seine Stärken in der Erfassung qualifizierter Erfahrungen und Soft Skills.
- PRINCE 2 bietet eine erprobte und direkt anwendbare Methode für ein ergebnisorientiertes Projektmanagement.

Fazit

Die Projektmanagement-Zertifizierungen bieten den Unternehmen die Möglichkeit, eine höhere Standardisierung und Transparenz bei der Realisierung ihrer wichtigen IT-Vorhaben zu etablieren. Zudem fördern sie damit die Weiterentwicklung der Kompetenzen im Unternehmen.

Sie unterstützen den Projektmanager und die Projektbeteiligten nachhaltig bei der Erreichung der angestrebten Ziele. Die Praxis zeigt, dass je höher die Qualität von Projektmanagement-Standards gekoppelt mit der persönlichen Erfahrung des Projektmanagers ist, desto erfolgreicher ist die Realisierung von Projekten. Projektmanagement-Zertifizierungen helfen somit die gesetzten Unternehmensziele besser zu erreichen.



Bernhard Burkhard
(info@coniatos.de)

Glossar

Best Practice

(deutsch: Erfolgsmethode) Best Practice bezeichnet bewährte, optimale bzw. vorbildliche Methoden, Praktiken oder Vorgehensweisen.

Stakeholder

Eine Person oder Gruppe, die ein berechtigtes Interesse am Verlauf oder Ergebnis eines Prozesses oder Projektes hat.

ITIL

Die IT Infrastructure Library ist eine Sammlung von Best Practices in einer Reihe von Publikationen zur Umsetzung eines IT-Service-Managements (ITSM) und gilt inzwischen international als De-facto-Standard

Links

- ▶ [1] Webseite vom Project Management Institute (PMI):
<http://www.pmi.org/>
- ▶ [2] Webseite der Deutschen Gesellschaft für Projektmanagement e. V. (GPM):
<http://www.gpm-ipma.de>
- ▶ [3] Webseite von PRINCE2:
<http://www.prince-officialsite.com/>

Seminarempfehlung: IT-Projektmanagement - Methoden und Techniken

▶ **Informationen/Online-Anmeldung:** <http://training.ordix.de>

In diesem Seminar erhalten Sie einen umfassenden und systematischen Überblick über alle Begriffe, Methoden und Arbeitstechniken zum Management von IT-Projekten. Sie lernen, IT-Projekte zu planen und zu steuern sowie verfügbare Arbeitshilfen projektspezifisch anzupassen.

Seminarinhalte

- Grundlagen
- Von der Idee zum Projekt
- Projektinitialisierung
- Risikomanagement
- Strukturierung und Planung
- Projektdurchführung
- Test, Abnahme und Abschluss
- Qualitätsmanagement
- Arbeits- und Hilfsmittel
- Fallbeispiele, Übungen

Termine

14.10. - 18.10.2013 in Wiesbaden

Seminar-ID: PM-01

Dauer: 5 Tage

Preis pro Teilnehmer:
1.990,00 € (zzgl. MwSt.)

Frühbucherpreis:
1.791,00 € (zzgl. MwSt.)



Buchen Sie gleich hier!

Neuerungen in der Oracle Database 12c

Oracle 12c - Eine für alle, alle in einer

Heimlich, still und leise ist Oracle mit dem neuen Major Release ihrer Datenbank an den Start gegangen. Seit dem 25. Juni ist Oracle 12c für Linux, Oracle Solaris und seit kurzem auch als Windows-Version verfügbar. Frei nach dem Motto „Tous pour un, un pour tous“ (deutsch: „alle für einen, einer für alle!“) oder besser „eine für alle, alle in einer“, hat Oracle in der neuen Datenbankgeneration einen einschneidenden Paradigmenwechsel vorgenommen: Multitenancy - Mandantenfähigkeit.

Ein erster Überblick

In dieser neuen Version sind über 500 neue Funktionen verborgen, wobei in allen wichtigen Teilbereichen der Oracle-Datenbank Veränderungen und Neuerungen zu entdecken sind. Grob betrachtet handelt es sich um Erweiterungen in den Bereichen:

- Multitenant Architecture
- SQL und PL/SQL
- Partitioning, Compression, Archiving und Data Warehousing
- Installation, Migration, Patching
- RMAN & Data Guard
- Tuning (Manageability und Performance)
- Oracle RAC and Grid Infrastructure
- Security
- Besonderheiten bei der Verwendung von Oracle unter Microsoft Windows

Im Folgenden werden wir die wichtigsten neue Funktionalitäten anreißen.

Multitenant Architecture

Die neue kostenpflichtige Option zur Mandantenfähigkeit wird von Oracle als die zentrale Funktion genannt. Hier wird an einem Grundprinzip von Oracle gerüttelt. Bisher galt immer, dass eine Instanz genau mit einer Datenbank kommunizieren konnte. Nun ist die Mandantenfähigkeit eingeführt worden (siehe Abbildung 1).

Eine Instanz kann somit nun mehrere Datenbanken versorgen. Die zentralen Komponenten sind nur noch einmal in einer Container-Datenbank (CDB) vorhanden (Undo Tablespace, Redo Log Files etc.), die eigentlichen Datenbanken sind jetzt austauschbar – sogenannte Pluggable-Databases (PDB) und enthalten die individuellen (Nutzer-)Daten.

Der eine oder andere Leser wird sich jetzt die Augen reiben. Denn, wer andere relationale Datenbankmanagementsysteme kennt, für den ist das ein alter Hut. In DB2, SQL Server und Informix ist diese Funktionalität schon immer Stand der Technik.

Table Restore

Wer schon einmal eine einzelne Tabelle aus einem Backup benötigt hat, der weiß, wie aufwändig das bisher war. Es gab verschiedene Wege, um hier zum Ziel zu gelangen:

- Flashback Database mit oder ohne Data Guard
- Tablespace Point-in-Time Recovery mit anschließendem Export
- Database Point-in-Time Recovery mit anschließendem Export

Oracle hat mit der neuen Version 12c nun das Table Restore eingeführt. Jetzt kann man mittels RMAN einzelne Tabellen aus einem Backup zurückholen und diese anschließend bei Bedarf unter einem anderen Namen in der Datenbank speichern.

Wir sind sicher, dass viele Administratoren händelnd auf diese Funktion gewartet haben.

Limitierung der PGA

Auch im Bereich Performance sind so manche Neuerungen zu finden. In Versionen kleiner 12c war es z.B. nicht möglich, einem Wildwuchs der PGA Einhalt zu gebieten.

Mit dem neuen Parameter `PGA_AGGREGATE_LIMIT` kann nun verhindert werden, dass die Werte der PGA Memory dynamisch anwachsen und gegebenenfalls ein Swapping auslösen.

Der Datentyp VARCHAR2

Die Grenze der Feldgrößen für die alphanumerischen Datentypen `VARCHAR2` und `NVARCHAR2` sind endlich denen von PL/SQL angeglichen worden. Die Beschränkung liegt jetzt auch bei 32.767 Bytes.

Allerdings muss für die Nutzung der neue Initialisierungsparameter `MAX_STRING_SIZE` auf den Wert `EXTENDED` geändert werden. Der Vorgang ist anschließend irreversibel.

Patch-Informationen in der Datenbank

Wäre es nicht schön, Informationen zu eingespielten Patches direkt aus der Datenbank abfragen zu können? Mit dem neuen Release ist dies jetzt möglich. Die 12c liefert hierfür eine PL/SQL-API zu `OPatch` für Abfragen und Skripte:

```
SQL> select dbms_qopatch.get_opatch_list
from dual;
GET_OPATCH_LIST
---
<patches/>
```

Unified Auditing

Das Auditing ist nach ca. 20 Jahren komplett überarbeitet worden und firmiert jetzt unter der Bezeichnung Unified Auditing. Der DBA kann zwar noch zwischen der alten und der neuen Variante wählen, aber selbstverständlich wird die Nutzung der neuen Variante empfohlen. Kennzeichen hierzu sind:

- Policies steuern nun das Auditing und nicht mehr Initialisierungsparameter (Unified Audit Trail).
- Die Audit-Daten gehören jetzt dem Schema `AUDSYS` und liegen in einer Tabelle im Tablespace `SYSAUX`.
- Die Audit-Einstellungen werden über die Rollen `AUDIT_ADMIN` und `AUDIT_VIEWER` geändert.

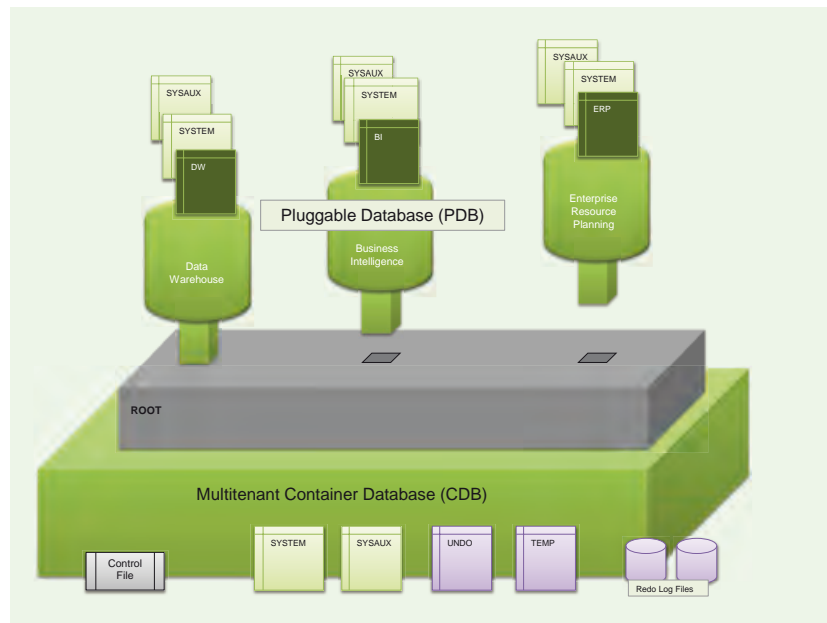


Abb. 1: „Plug into the Cloud - Die neue Multitenant Architektur“ (Slogan von Oracle für die neue Version 12c).

Netzwerkverschlüsselung

Oracle Database 12c bietet nun auch eine Unterstützung für die Verschlüsselung des Netzwerkverkehrs zu und von einer Oracle-Datenbank ohne die lizenzpflichtige Funktion „Advanced Security Option“ (ASO) an und ist somit kostenfrei.

Online-Operationen

Die neue Oracle Version 12c garantiert eine höhere Unterbrechungsfreiheit im Betriebsablauf durch mehr Funktionalitäten bei Online-Operationen.

Dies ist z.B. die neue Anweisung `MOVE DATAFILE`. Durch diese kann eine Umbenennung bzw. Verschiebung der Datendateien erfolgen, ohne dass im Vorfeld der Status auf `OFFLINE` geändert werden muss. Auch die Anweisung `ONLINE PARTITION MOVE`, das Löschen eines Index, einer Spalte oder eines Constraint sind jetzt ohne das Setzen von blockierenden Sperren möglich.

Replikation

Mit dem neuen Release müssen wir uns aber leider auch von einigen Funktionen verabschieden. Die Advanced Replication und Streams werden nicht mehr weiterentwickelt und sind als „deprecated“ (deutsch: veraltet) gekennzeichnet. Dazu findet man im Upgrade Guide der Oracle-Dokumentation den Hinweis, für zukünftige Replikationsverfahren doch das Produkt Oracle GoldenGate einzusetzen.

Glossar

ASO

Die Advanced Security Option bietet im Rahmen der Netzwerkverschlüsselung erweiterte Sicherungsmöglichkeiten.

CDB

Die Container Database ist die zentrale Datenbank in der Multitenant-Architektur.

Multitenancy

Mandantenfähigkeit

PDB

Die Pluggable Database ist die (Nutzer-)Datenbank in der Multitenant-Architektur.

PGA

Eine Program Global Area ist ein Speicherbereich (RAM), der Daten, Steuer- und Kontrollinformationen eines Serverprozesses enthält.

RAC

Der Real Application Cluster ist die Hochverfügbarkeitslösung für Oracle.

RMAN

Der Recovery Manager ist das Werkzeug zur Datenbanksicherung von Oracle.

Swapping

Unter Swapping wird die Umlagerung von Daten einer höheren in eine tiefere Ebene innerhalb einer Speicherhierarchie verstanden.

Leider ist der Einsatz von GoldenGate nicht kostenfrei, sondern muss zusätzlich lizenziert werden.

Fazit

Das neue Release stellt einen tiefen Einschnitt in die bisher geltenden Architekturgesetze der Oracle-Datenbank dar. Den Löwenanteil trägt hierbei sicherlich die Mehrmandantenfähigkeit, die sich auf sehr viele andere Teilkomponenten auswirkt und eine Neuorganisation der administrativen Prozesse erforderlich macht.

Ausblick

Nachdem dieser Artikel zunächst einen ersten Überblick über die zentralen Funktionalitäten der neuen Version von Oracle gegeben hat, werden wir uns in kommenden Artikeln der ORDIX news näher mit den einzelnen Erweiterungen von Oracle Database 12c beschäftigen und dabei detaillierter auf den jeweiligen Funktionsumfang eingehen.

Links

- ▶ [1] Oracle Database 12c New Features:
http://docs.oracle.com/cd/E16655_01/server.121/e17906/chapter1.htm#NEWFTCH1



Andreas Schlicht
(info@ordix.de)

Seminarempfehlung: Oracle 12c Neuheiten



- ▶ Informationen/Online-Anmeldung: <http://training.ordix.de>

Dieses neue Seminar vermittelt Ihnen die neuen Funktionen von Oracle 12c. Schwerpunkte sind dabei SQL, PL/SQL, Administration, Multitenant-Architektur, Migration, Security, Performance Tuning, Backup & Recovery und Hochverfügbarkeit. Zahlreiche Übungen und Beispiele helfen Ihnen, die neuen Konzepte zu beherrschen. Außerdem zeigen wir die Stärken und Schwächen der Version 12c auf.

Seminarinhalte

- SQL- und PL/SQL-Erweiterungen
- Partitioning, Compression, Archiving und Data Warehousing
- Installation, Migration, Patching
- Multitenant-Architektur
- Neuerungen von RMAN & Data Guard
- Tuning Erweiterungen (Manageability und Performance)
- Neuerungen von Oracle RAC and Grid Infrastructure
- Security-Neuerungen
- Besonderheiten bei Verwendung unter Microsoft Windows
- Vertiefung der Theorie durch praktische Übungen und Beispiele

Termine

25.11. - 29.11.2013 in Wiesbaden

Seminar-ID: DB-ORA-49

Dauer: 5 Tage

Preis pro Teilnehmer:
1.990,00 € (zzgl. MwSt.)

Frühbucherpreis:
1.791,00 € (zzgl. MwSt.)



Buchen Sie gleich hier!

Neuheiten von JBoss 7

Faster, smarter, smaller

Waren die letzten Versionen des JBoss Application Server aus dem Hause RedHat jeweils Weiterentwicklungen der vorherigen Version, so stellt JBoss 7 quasi eine komplette Neuimplementierung dar. Dadurch hat sich nicht nur die Administration des JBoss völlig geändert, auch bei der Migration von Anwendungen, die bisher auf älteren JBoss-Versionen liefen, muss erheblich mehr beachtet werden. Viele Kunden, die JBoss einsetzen, haben sich schon mit der neuen Version beschäftigt und bei einigen Kunden hat JBoss 7 bereits den Weg in die Produktivumgebungen gefunden. Es wird also höchste Zeit, dass wir einige der wichtigsten Neuerungen kurz vorstellen.

Aller Anfang ist schwer

Der Start der neuen JBoss-Version verlief zunächst ein wenig holprig: fast zwei Jahre ist es mittlerweile her, dass ein JBoss dessen Versionsnummer mit einer 7 beginnt, erschienen ist. Die erste voll JEE6-zertifizierte Version (7.1) erschien jedoch erst ein dreiviertel Jahr später und die kommerzielle Variante des Server in Form der Enterprise Application Platform (EAP) 6 wurde nach mehreren Verschiebungen erst im Juni 2012 veröffentlicht.

Seit der ersten veröffentlichten 7er Version (7.0.1) bis zur heute aktuellen EAP 6.1 (basierend auf der JBoss-Version 7.2) hat sich viel getan: Anfängliche Fehler und Schwachpunkte wurden ausgebessert und fehlende Funktionalitäten ergänzt. Hierbei wurden jedoch auch Schnittstellen und APIs geändert, so dass Beispiele und Tutorials, die für ältere 7er Versionen erstellt wurden mit aktuelleren Versionen häufig nicht mehr funktionieren.

Mittlerweile ist der JBoss jedoch auf einem stabilen Stand und bringt alles mit, um auch komplexe Enterprise-Anwendungen performant betreiben zu können.

Aus viele wird eine

Eine der wesentlichsten Änderung aus Sicht des Administrators stellt sicherlich die Reduzierung der zuvor zahlreichen Konfigurationsdateien auf nur noch eine einzige (Standalone-Modus) bzw. zwei Dateien (Domain Mode) dar. In älteren Versionen haben

viele Module bzw. Services ihre eigene Konfigurationsdatei mitgebracht. So gab es z.B. eine eigene Konfigurationsdatei für das Logging, eine Datei für das Clustering, eine für das Monitoring, usw.

In der zentralen Konfigurationsdatei können nun alle Module/Services konfiguriert werden, ähnlich wie man es z.B. vom Apache WebServer oder vom Tomcat Servlet-Container her kennt. Dabei gibt es für jedes Modul einen eigenen Konfigurationsabschnitt, der durch eine eigene XML-Schema-Definition beschrieben wird. Die Syntax einiger Konfigurationen ähnelt daher häufig immer noch denen der vorherigen Version.

Neue Architektur: JModules

Die in den vorherigen Versionen eingesetzte Architektur, welche auf dem Microkernel bzw. JMX basierte, wurde mit der neuen Version nun endgültig abgelöst. An ihre Stelle ist das neue JModules getreten, das von RedHat auch als eigenes Projekt angeboten wird, um eigene Anwendungen auf dieser Basis zu erstellen. Die eigentlichen Services bzw. Funktionalitäten des Server sind dabei in einzelnen Modulen untergebracht, die unabhängig voneinander sind und auch einen eigenen Classloader besitzen. Das geänderte Classloader-Konzept ist ebenfalls einer der Punkte, die bei der Migration älterer Applikationen auf die neue Server-Struktur berücksichtigt werden müssen.

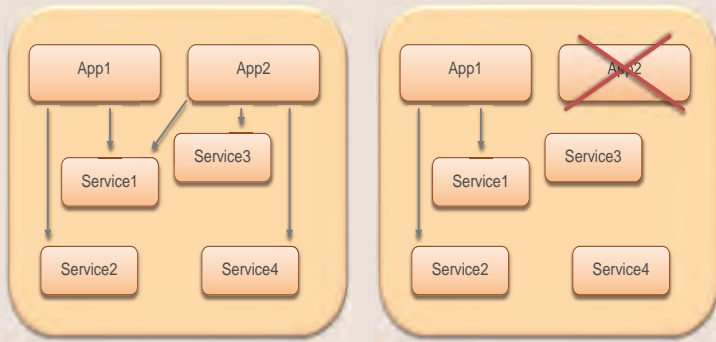


Abb. 1: Servicestart „On Demand“.

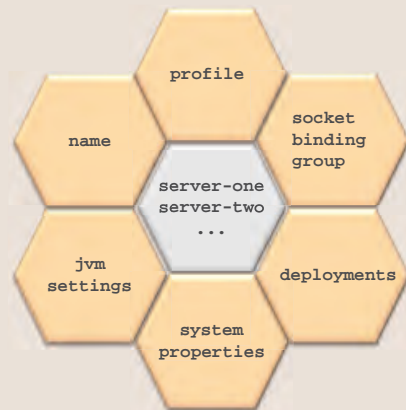


Abb. 2: Eigenschaften einer Server-Gruppe.

Haben Module oder Deployments untereinander Abhängigkeiten, so müssen diese explizit angegeben werden, damit sie vom Classloading-Mechanismus berücksichtigt werden können. Ein Deployment hat immer Abhängigkeiten zu Services des Application Server, diese müssen i.d.R. aber nicht explizit angegeben werden. Findet der Server innerhalb der Klassen eines Deployment-File beispielsweise Hibernate-Annotations, wird implizit eine Abhängigkeit zu den Hibernate-Services des Containers erstellt.

Ein spürbarer Effekt dieser neuen Architektur fällt sofort beim Start des Server auf: Der neue JBoss startet erheblich schneller als vorherige Versionen. Dies liegt daran, dass beim Start nur die Services, die auch benötigt werden, mit gestartet werden. Weitere Services werden „On-Demand“ gestartet (siehe Abbildung 1).

Wird beispielsweise ein JBoss mit einer Cluster-Konfiguration gestartet, werden die Cluster-Services nicht automatisch bereitgestellt. Sie werden erst dann geladen, wenn tatsächlich eine Applikation, die im Cluster laufen soll, deployed wird. Dies erkennt der Server in diesem Fall an den Deployment-Diskrepanzen.

Der Domain Mode

Eine gänzlich neue Funktionalität stellt der Domain Mode dar. Oft wird der Betrieb mehrerer JBoss-Instanzen im Domain Mode mit dem Cluster-Betrieb in Verbindung gebracht. Der Domain Mode ist jedoch lediglich dazu gedacht, eine oder mehrere Gruppen von Servern mit derselben Konfiguration zu betreiben und vor allem um diese Gruppen anschließend zentral konfigurieren zu können.

Anstelle der Pflege mehrerer `standalone.xml`-Konfigurationsdateien bietet der Domain Mode die Möglichkeit, die Konfiguration für mehrere Server-Instanzen zentral in der Konfigurationsdatei `domain.xml` vorzunehmen. Hier können verschiedene Server-Gruppen definiert werden, für die jeweils ein Konfigurationsprofil und einige andere gemeinsame Eigenschaften festgelegt werden (siehe Abbildung 2).

Auf den einzelnen physikalischen oder virtuellen Servern können dann beliebig viele JBoss-Instanzen gestartet werden, die genau einer Server-Gruppe zugewiesen werden. Dies erfolgt in der Konfigurationsdatei `host.xml`, welche in jedem System auf dem JBoss-Instanzen laufen existiert (siehe Abbildung 3). Der in diesem Beispiel angegebene Parameter `port-offset` legen fest, dass für die betreffenden Server der Wert von 100 bzw. 200 auf die zentral in der `domain.xml` definierten Port-Definitionen aufaddiert wird. Weiterhin sind in der `host.xml` rechner-spezifische Konfigurationsparameter, wie z.B. die Interface-Einstellung sowie der Verweis auf den zuständigen Domain Controller untergebracht.

Im Domain Mode werden neben den eigentlichen Server-Prozessen mindestens zwei weitere Prozesse gestartet: Der erste zusätzliche Prozess stellt den Domain Controller dar, der die zentrale Konfiguration verwaltet und Änderungen über die Host Controller an die einzelnen Instanzen verteilt. Weiterhin existiert für jeden physikalischen/virtuellen Server ein Host Controller, der die eigentlichen JBoss-Instanzen startet und stoppt (siehe Abbildung 4).

Die Domain-Funktionalität ist wie eingangs erwähnt, völlig losgelöst vom Cluster-Betrieb. In der Praxis bietet es sich jedoch an, für in einem Cluster be-

triebene JBoss-Instanzen den Domain Mode zu verwenden. Dies ist ratsam, da die Instanzen eines Cluster in aller Regel identisch konfiguriert und auf den einzelnen Instanzen meist auch dieselben Applikationen deployed werden. Es ist aber auch möglich und für einige Szenarien durchaus sinnvoll, einen Cluster aus mehreren Standalone-Instanzen zu erstellen.

Die Admin-CLI

Werden viele andere Application Server vorwiegend über eine Konsole administriert, sind Änderungen im JBoss hingegen entweder mit Anpassungen in den XML-Dateien verbunden oder können über die Kommandozeile vorgenommen werden. In älteren Versionen stand dem Administrator hierfür das Werkzeug Twiddle zur Verfügung, das bereits seit der Version 3.x mit ausgeliefert wurde.

Die Verwendung der neuen Admin-CLI-Schnittstelle bietet mehrere Vorteile: Zum einen ist durch die Verwendung der CLI sichergestellt, dass die in die Konfigurationsdatei zurückgeschriebenen Änderungen einen gültigen XML-Code erzeugen. Syntax- bzw. Parsing-Fehler, die bei manuellen Änderungen an einer XML-Datei immer mal vorkommen können, sind bei Verwendung der CLI ausgeschlossen.

Zum anderen wird bei jeder Änderung, die mittels CLI (oder über die Admin-Konsole) durchgeführt wird, eine Version der vorherigen Konfiguration in eine History geschrieben, so dass ältere Stände bei Bedarf stets wieder hergestellt werden können. Als Ablageort dient das Verzeichnis:

```
<JBOSS_HOME>/standalone/configuration/standalone_xml_history
```

bzw.

```
<JBOSS_HOME>/domain/configuration/domain_xml_history
```

Die Versionsstände werden hier im Unterverzeichnis **current** abgelegt und der Dateiname um eine laufende Nummer erweitert. Beim Neustart des Server wird das Verzeichnis umbenannt und mit einem Datums- und Zeitstempel versehen.

Seit JBoss 7 existiert neben dem normalen Web/HTTP-Interface, über das die Applikationen erreichbar sind, ein separates Management-Interface, das über einen anderen Port - und soweit vorhanden - auch über ein anderes physikalisches Interface angesprochen werden kann. Die CLI kommuniziert hierbei standardmäßig über den Port 9999 mit dem Server.

```
<host>
  ...
  <servers>
    <server name="server-blau" group="server-gruppe-eins">
    </server>
    <server name="server-rot" group="server-gruppe-eins"
auto-start="true">
      <socket-bindings port-offset="100"/>
    </server>
    <server name="server-gelb" group="server-gruppe-zwei"
auto-start="true">
      <socket-bindings port-offset="200"/>
    </server>
  </servers>
</host>
```

Abb. 3: Zuordnung der JBoss-Instanz eines Rechners zu einer Server-Gruppe.

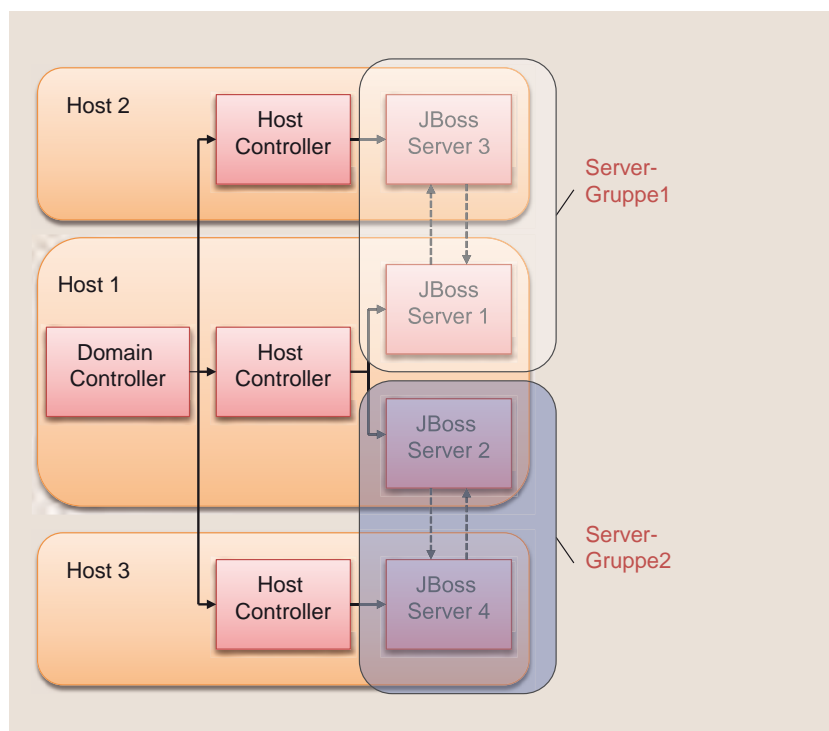


Abb. 4: Zusammenspiel von Domain Controller, Host Controllern und Server-Instanzen.

Die Admin-CLI kann entweder im interaktiven oder im nicht-interaktiven Modus verwendet werden. In den interaktiven Modus gelangt man einfach durch den Start der entsprechenden Skript-Datei **jboss-cli.sh** bzw. **jboss-cli.bat** im **bin**-Verzeichnis der Installation. Durch den Aufruf gelangt man in eine Art Shell, in der man mit Befehlen wie **ls**, **cd** und **pwd** durch die JBoss-Konfiguration ähnlich wie durch ein File System

Glossar

EAP

Die Enterprise Application Platform ist die kommerzielle Variante des JBoss Application Server.

CLI

Das Command-Line-Interface ist das neue Kommandozeilen-Werkzeug für die Administration des JBoss.

JON

Das JBoss Operations Network ist eine Management- und Monitoring-Software aus dem Hause RedHat (OpenSource Version: RHQ).

Links

- ▶ [1] Download des Application Server:
<http://www.jboss.org/jbossas/downloads>
- ▶ [2] Seminarempfehlung: Administration und Konfiguration für JBoss 7
<http://training.ordix.de/siteengine/action/load/kategorie/Web%20und%20Application%20Server/nr/1411/index.html>

navigieren kann. Insgesamt stehen dem Anwender über 30 Kommandos zur Verfügung. Zur Anpassung von Konfigurationseinstellungen sind vor allem die folgenden Befehle relevant:

- **:add**
Dieser Befehl fügt eine neue Ressource oder ein neues Attribut hinzu.
- **:remove**
Dieser Befehl entfernt eine bestehende Ressource oder ein bestehendes Attribut.
- **:write-attribute**
Mit diesem Befehl ändert man den Wert eines bestehenden Attributs einer Ressource.
- **:read-attribute**
Mit Hilfe dieses Befehls liest man den aktuellen Wert eines Attributs aus.

Der nicht-interaktive Modus bietet sich vor allem für die Ausführung von vorgefertigten Skripten an, um wiederkehrende Aufgaben schnell und effektiv durchführen zu können. Hierfür kann der Startdatei entweder direkt ein Befehl oder der Pfad zu einer Datei als Argument mit übergeben werden:

```
jboss-cli.sh --connect --command=<Kommando>
jboss-cli.sh --connect --file=<PFAD_ZUR_
DATEI/DATEINAME>
```

Eine Datei kann beliebig viele Befehle enthalten, die nacheinander ausgeführt werden. Um mehrere Operationen in einer Art Transaktion laufen zu lassen bzw. um sicherzustellen, dass entweder alle oder eben

keine Änderung umgesetzt wird, bietet die Admin-CLI einen sogenannten Batch-Modus an. Über den Befehl **batch** wird der Modus gestartet. Alle anschließend eingegebenen Kommandos werden nicht unmittelbar ausgeführt, sondern für den Batch-Lauf vorgesehen. Mit dem Befehl **run-batch** können anschließend alle Kommandos zusammen ausgeführt werden. Kommt es bei einem der Befehle zu einem Fehler, wird der Lauf abgebrochen und alle innerhalb des Laufes vorgenommenen Änderungen rückgängig gemacht.

Fazit

In einigen Bereichen bietet die aktuelle Version einen geringeren Leistungsumfang als die vorherigen Versionen. So bietet die Admin-Konsole (noch) sehr eingeschränkte Möglichkeiten, die i.d.R. nicht ausreichen, um die täglichen Administrationsaufgaben zu erledigen. Auch die Bordmittel, die JBoss von Haus aus für das Monitoring mitbringt, sind nicht so umfangreich wie noch in der Version 5 - hier setzt RedHat offenbar verstärkt auf Zusatzprodukte wie etwa JON.

Eine verbesserte Performance und neue Funktionen wie der Domain Mode, eine vereinfachte Cluster-Konfiguration und die sehr leistungsstarke Admin-CLI machen die neue JBoss-Version jedoch zu einem soliden und leistungsstarken JEE Application Server, der sich vor seiner Konkurrenz aus dem Hause IBM oder Oracle nicht verstecken muss.



Alexander Zeller
(info@ordix.de)

Die neue Version 10.1 von IBM DB2 (Teil IV)

Funktionale Erweiterung im Bereich Monitoring

Um bei einer Datenbank sowohl eine optimale Leistung zu erzielen, als auch einen störungsfreien Betrieb sicherzustellen, ist ein Monitoring unerlässlich. Mit den verschiedenen Entwicklungsstufen der einzelnen DB2-Versionen hat auch der Datenbankadministrator ständig neue und verfeinerte Monitoring-Elemente an die Hand bekommen, um bestimmte Messwerte zu erhalten. Ab der Version 10 wurden nun weitere Monitoring-Funktionen hinzugefügt, die wir in diesem Artikel näher betrachten wollen.

Nutzungslisten

Schon in der Vorgängerversion 9.7 wurde die Spalte **Lastused** in den Katalogtabellen eingeführt. In dieser wird für Tabellen, Indizes und Packages ein Nutzungszeitstempel für das jeweilige Objekt eingetragen. Das Datum des letzten Verweises in der Spalte **Lastused** kann somit zum Ermitteln von Zugriffen auf Indizes und andere Objekte dienen. Prüft man z.B. in regelmäßigen Zeiträumen in der Katalog-View **SYSCAT.INDEXES** diese Spalte auf alte Einträge von Zeitstempeln, oder ob gar noch der Default-Wert 0001-01-01 eingetragen ist, so können zum Beispiel ungenutzte Indizes identifiziert werden. Nicht verwendete Indizes belegen nur unnötig Platz und führen zu einem erhöhten Aufwand was **Reorg**, **Runstats**, **Inserts** und **Updates** betrifft.

Mit der Einführung des neuen Datenbankobjektes Nutzungslisten (**USAGE LIST**) wurde die Möglichkeit für das Monitoring der Objektnutzung erheblich erweitert. Mit diesen Listen können DML-Anweisungsabschnitte aufgezeichnet werden, die auf eine bestimmte Tabelle oder einen bestimmten Index referenzieren. Während der Ausführung werden anschließend Statistikdaten erfasst, um zu erkennen ob der Optimizer für einen Datenzugriff bestimmte Indizes verwendet.

Ein Beispiel für die Funktionsweise von Nutzungslisten

An dieser Stelle wollen wir nun mit Hilfe eines Beispiels die Funktionsweise für die Nutzung eines Index auf einer Tabelle darstellen.

Zunächst wird eine Tabelle angelegt und ein Index erstellt (siehe Abbildung 1, Schritt 1). Anschließend wird die Tabelle mit Daten gefüllt.

Im Schritt 2 werden die Nutzungslisten für die Tabellen und Indizes angelegt. Wobei für das Beispiel nur die Nutzungsliste für den Index benötigt wird, vollständigshalber wird an dieser Stelle aber auch das Anlegen einer Nutzungsliste für Tabellen dargestellt.

Zusätzlich kann beim Erstellen der Nutzungslisten noch definiert werden, wie viele Zeilen mitgeloggt werden und wie das Verhalten beim Erreichen der vorgegebenen Größe ausfällt. Es wird definiert, ob die Liste wieder von vorn beschrieben oder ob das Monitoring beendet wird. Dies kann direkt bei der Definition oder im Nachhinein durch ein **ALTER**-Statement bestimmt werden (siehe Schritt 3)

Ebenso kann festgelegt werden, ob sie erst aktiviert werden müssen oder beim Datenbankstart direkt aktiv sind. Mit dem in Schritt 4 des Beispiels dargestellten Befehl werden die Listen aktiviert.

In der Katalog-View **SYSCAT.USAGELISTS** können anschließend die erstellten Listen abgefragt werden. Die Auswertung der Nutzungslisten erfolgt über die Monitor-Funktion (siehe Schritt 5).

Der konkrete Status für eine bestimmte Nutzungsliste kann mit dem Befehl in Schritt 6a abgefragt werden. Alternativ lässt sich der Status für alle Listen, die definiert und aktiv sind, abfragen (siehe Schritt 6b).

Schritt 1

```
CREATE TABLE TEST.BENUTZER
(ID INT, VORNAME CHAR(50), NACHNAME CHAR(50), DATUM DATE)
```

```
CREATE INDEX TEST.IDX1 ON TEST.BENUTZER (ID)
RUNSTATS ON TABLE TEST.BENUTZER AND INDEXES ALL
```

Schritt 2

```
CREATE USAGE LIST TEST.BEN_I1 FOR INDEX TEST.IDX1
CREATE USAGE LIST TEST.BEN_T FOR TABLE TEST.BENUTZER
```

Schritt 3

```
ALTER USAGE LIST TEST.BEN_I1 LIST SIZE 1000
ALTER USAGE LIST TEST.BEN_I1 WHEN FULL (WRAP oder DEACTIVATE)
```

Schritt 4

```
SET USAGE LIST TEST.BEN_I1 STATE ACTIVE (INACTIVE)
```

Schritt 5

```
SELECT * FROM TABLE (MON_GET_INDEX_USAGE_LIST('TEST','BEN_I1',-1))
```

Schritt 6a

```
SELECT * FROM TABLE (MON_GET_USAGE_LIST_STATUS('TEST','BEN_I1',-1))
```

Schritt 6b

```
SELECT * FROM TABLE (MON_GET_USAGE_LIST_STATUS(NULL,NULL,-1))
```

Schritt 7

```
SELECT
  SUBSTR(rtrim(USAGELISTSHEMA)||'.'||rtrim(USAGELISTNAME),1,20)
as    USAGELIST,
  SUBSTR(rtrim(INDSCHEMA)||'.'||rtrim(INDNAME),1,20) as INDEX,
  NUM_REFERENCES,
  LAST_UPDATED,
  EXECUTABLE_ID
from
  TABLE (MON_GET_INDEX_USAGE_LIST(NULL,NULL,-1))
```

Schritt 8

```
SELECT
  SUBSTR(rtrim(USAGELISTSHEMA)||'.'||rtrim(USAGELISTNAME),1,12)
as    USAGELIST,
  SUBSTR(rtrim(INDSCHEMA)||'.'||rtrim(INDNAME),1,12) as INDEX,
  SUBSTR(t.STMT_TEXT,1,43) as STMT,
  t.NUM_EXECUTIONS, i.LAST_UPDATED
from
  TABLE (MON_GET_INDEX_USAGE_LIST(NULL,NULL,-1)) as i
  inner join
  TABLE (MON_GET_PKG_CACHE_STMT(NULL,NULL,NULL,-1)) as t
on i.EXECUTABLE_ID = t.EXECUTABLE_ID
ORDER BY t.NUM_EXECUTIONS DESC;
```

Abb. 1: Beispiel für die Funktionsweise von Nutzungslisten.

Um nun identifizieren zu können, wie oft auf einen Index zugegriffen wird, kann die Nutzungsliste mit Hilfe des Statement aus Schritt 7 abgefragt werden.

Die Spalte **EXECUTABLE_ID** entspricht auch einem eindeutigen Eintrag im Package Cache. Somit besteht nun die Möglichkeit, das dazugehörige SQL-Statement aufzulösen, indem man einfach die Ergebnisse aus der Index-Nutzungsliste mit der Monitor-Funktion **MON_GET_PKG_CACHE_STMT** verbindet und die Join-Bedingung die **EXECUTABLE_ID** ist (siehe Schritt 8).

Das Ergebnis sieht nun wie folgt aus: Durch das Setzen des Wertes **NULL** in der Funktion **MON_GET_INDEX_USAGE_LIST** erhält man eine Information über die gesamte Indexnutzung bei verschiedenen Abfragen gegen eine bestimmte Tabelle. Es kann nun ermittelt werden, ob der Optimizer jeden zur Verfügung stehenden Index auch bei den Statements verwendet (siehe Abbildung 2).

An dieser Stelle sind zwei Aspekte zu beobachten: Zum einen, dass der Package Cache groß genug ist, um alle Statements, die man erfassen möchte, aufnehmen zu können und zum anderen, dass keine Operation vom Typ „dynamic flush package cache“ ausgeführt wird, die dann die SQL-Statements aus dem Package Cache entfernen würde.

Event-Monitor für die Änderungshistorie

Mit dem Event-Monitor für die Änderungshistorie können Parameteränderungen an den Registry-Variablen, an den Datenbank- und Instanz-Konfigurationsparametern mitgeschnitten werden. Es werden auch DDL-Statements erfasst. Je nach Einstellung können hier alle Statements oder nur bestimmte Bereiche wie Indexzugriffe, Zugriffe auf Funktionen oder Packages etc. aufgezeichnet werden. Bei den Utilities können **LOAD**, **BACKUP**, **REORG**, **RUNSTATS** etc. mitgeschnitten werden. Sinn und Zweck des Monitors ist es hauptsächlich, dass anhand der Daten ermittelt werden kann, ob irgendwelche Datenbankprobleme oder Leistungseinbußen von Parameteränderungen oder nicht durchgeführten Wartungsprozessen der Datenbank herrühren.

Ein Beispiel des Monitors, der alle Änderungen an den Konfigurationsparametern und Registry-Variablen mitloggt ist in der Abbildung 3 zu sehen.

Für die Auswertung werden die folgenden Tabellen angelegt:

- **CHG_SUMMARY_HISTORY**
- **DB_DBM_HISTORY**
- **REGVAR_HISTORY**

Eine weitere Möglichkeit ist das Erstellen eines Event-Monitors, welcher alle Dienstprogramme wie das Ausführen von **LOAD**, **MOVE TABLE**, **ONLINE BACKUP**, **ONLINE RESTORE**, **ONLINE ROLLFORWARD**, **REDISTRIBUTE**, **REORG** und **RUNSTATS** mitschreibt:

```
CREATE EVENT MONITOR UTIL
FOR CHANGE HISTORY WHERE EVENT IN (UTILALL)
WRITE TO TABLE
AUTOSTART
```

Beim Anlegen des Monitors werden die folgenden Tabellen angelegt, welche für die Auswertung herangezogen werden können. Die Tabellen wurden hier nicht explizit definiert, der Event-Monitor legt die Tabellen fest, die benötigt werden:

- **CHANGESUMMARY_UTIL**
- **CONTROL_UTIL**
- **EVMONSTART_UTIL**
- **UTILLOCATION_UTIL**
- **UTILPHASE_UTIL**
- **UTILSTART_UTIL**
- **UTILSTOP_UTIL**

Fazit

Was die Nutzungslisten betrifft hat man von IBM ein Werkzeug an die Hand bekommen, welches in Verbindung mit der Funktion des Package-Cache-Monitors dazu dient, eine schnelle Aussage zu tätigen, ob der Optimizer bei verschiedenen Statements auf einen bestimmten Index zugreift oder nicht.

Beim Event-Monitor für die Änderungshistorie hat man den Vorteil, dass verschiedene historische Daten an einer gemeinsamen Schnittstelle zusammengefasst werden. Somit können die verschiedenen Parameteränderungen über einen bestimmten Zeitraum nachvollzogen werden. Hierdurch können Schlüsse über die Verhaltensänderung der Datenbank gewonnen werden.



Thomas Schwarz
(info@ordix.de)

USAGELIST	INDEX	STMT	NUM_EXECUTIONS
TEST.BEN_I1	TEST.I0X1	select id from test.benutzer where id = 4	6
TEST.BEN_I1	TEST.I0X1	select count(*) from test.benutzer	2
TEST.BEN_I1	TEST.I0X1	select id from test.benutzer where id = 3	2
TEST.BEN_I1	TEST.I0X1	select id from test.benutzer where id = 9	2
TEST.BEN_I1	TEST.I0X1	delete from test.benutzer	1
TEST.BEN_I1	TEST.I0X1	insert into test.benutzer values (1,'MARL',	1

Abb. 2: Informationen zur Indexnutzung.

```
CREATE EVENT MONITOR KONFIG
FOR CHANGE HISTORY WHERE EVENT IN (CFGALL)
WRITE TO TABLE
  CHANGESUMMARY (TABLE CHG_SUMMARY_HISTORY),
  DBDBMCFG (TABLE DB_DBM_HISTORY),
  REGVAR (TABLE REGVAR_HISTORY)
AUTOSTART
```

Abb. 3: Event-Monitor.

Glossar

DDL

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

REORG

Mit dem Dienst REORG können DB2-Tabellen reorganisiert und administriert werden.

RUNSTATS

Der Dienst RUNSTATS dient zur Ermittlung von statistischen Daten über die Inhalte von DB2-Tabellen und deren Indizes.

Package Cache

In diesem Speicher legt die Datenbank kompilierte SQL-Statements ab, um bei einer erneuten Ausführung auf diese zugreifen zu können.

Links

- ▶ [1] ORDIX news Artikel 03/2012
„Die neue Version 10.1 von IBM DB2 (Teil I) - Codename „Galileo“ - alles neu?“:
http://www.ordix.de/images/ordix/onews_archiv/pdf/news_1203.pdf
- ▶ [2] ORDIX news Artikel 04/2012
„Die neue Version 10.1 von IBM DB2 (Teil II) - Multi-Temperature Storage“:
http://www.ordix.de/images/ordix/onews_archiv/pdf/news_1204.pdf
- ▶ [3] ORDIX news Artikel 01/2013
„Die neue Version 10.1 von IBM DB2 (Teil III) - Zeitreise mit DB2“:
http://www.ordix.de/images/ordix/onews_archiv/pdf/news_1301.pdf
- ▶ [4] IBM DB2 Version 10.1 Information Center:
<http://pic.dhe.ibm.com/infocenter/db2luw/v10r1/index.jsp?lang=de>
- ▶ [5] IBM Developer Works:
<http://www.ibm.com/developerworks/data/library/techarticle/dm-1211indexanalysis/index.html?ca=drs->



Der neue Bootmanager

Besser booten mit GRUB2?

Mittlerweile ist der GRUB2 Bootloader [1] seit über einem Jahr verfügbar und hat den GRUB, der jetzt nur noch als GRUB Legacy bezeichnet wird, abgelöst. Schon jetzt hat er sich in den meisten Linux-Distributionen zum Standard etabliert. Daher wollen wir ihn in diesem Artikel genauer unter die Lupe nehmen.

Funktionsweise von GRUB

Normalerweise ist der Bootloader das erste Softwareprogramm, welches beim Booten auf einem Computer ausgeführt wird. Beim GRUB Legacy Bootloader wird die sogenannte Stage 1 in den Master Boot Record (MBR) geschrieben, welcher sich in den ersten 512 Bytes einer Festplatte befindet. Da GRUB mehr Platz in Anspruch nimmt, gibt es einen Verweis auf den ersten Sektor von Stage 2, in dem sich der Programmcode und eine Blockliste zum Lesen der restlichen Sektoren von Stage 2 befinden.

Stage 2 enthält die Dateisystemtreiber, Programmcode für das Auswahlmenü und die GRUB-Kommandozeile sowie die Laderoutine für die Betriebssystemkerne (Kernel). Nach dem Laden von Stage 2 wird die Konfigurationsdatei `/boot/grub/menu.lst` eingelesen und verarbeitet.

Für den Nachfolger GRUB2 wurde ein vollständiges Redesign durchgeführt und auf Rückwärtskompatibilität zu GRUB Legacy verzichtet. Die Stage 2 wurde in einen Kernel (`kernel.img`) und viele ladbare Module (`*.mod`) aufgeteilt. Der Kernel enthält nur essentiellen Code mit Dekompression, ELF-Loader für Module, Festplattenzugriff und eine Rettungs-Shell. Bei der Installation werden die Module für das Dateisystem, das die restlichen Komponenten enthält, an den Kernel angehängt und als Datei `core.img` abgelegt. Hierbei kommt eines der Kompressionsverfahren LZMA bzw. LZO zum Einsatz.

Nach dem Laden wird der Code entpackt und die Konfigurationsdatei `/boot/grub2/grub.cfg` geladen. Bei Bedarf werden Module für weitere Dateisysteme, das Boot-Menü, Boot-Routinen für verschiedene Betriebssysteme und die GRUB-Shell vom Datei-

system nachgeladen. Neben einer Shell-ähnlichen Skriptsprache bietet GRUB2 auch eine Unterstützung für die Skriptsprache Lua.

Funktionen von GRUB2

- Die Konfiguration wurde auf mehrere Dateien (Shell-Skripte) aufgeteilt, mit deren Hilfe die Konfigurationsdatei `grub.cfg` erstellt wird.
- Die Syntax der Konfiguration wurde geändert, sodass jetzt auch ein komplexes Skripting möglich ist.
- Rescue-Modus
- Möglichkeit eigene angepasste Menüs zu erstellen
- Eine Unterstützung für neue Dateisysteme wie z.B. ext4, ZFS und Btrfs wurde hinzugefügt.
- GRUB2 unterstützt das direkte Lesen von Dateien, die sich auf LVM- oder RAID-Geräten befinden.
- GRUB2 ist modular aufgebaut. Weitere Funktionen können bei Bedarf nachgeladen werden.
- CDs können direkt von der Festplatte ausgeführt werden.
- Eingebaute Kommandozeile ähnlich der Bash

Konfigurationsdateien von GRUB2

Während bei der Version GRUB Legacy die Datei `/boot/grub/menu.lst` direkt editiert wird, wird bei GRUB2 die Konfiguration, mittels Shell-Skripten und anschließendem Aufruf des folgenden Befehls durchgeführt:

```
grub2-mkconfig -o /boot/grub2/grub.cfg.
```

Die relevanten Konfigurationsdateien sind hierbei:

- `/boot/grub2/grub.cfg`
Die eigentliche Konfigurationsdatei von GRUB2. Die Erstellung der Datei erfolgt nicht direkt, sondern über Shell-Skripte und den Befehl `grub2-mkconfig`.
- `/etc/grub.d`
Verzeichnis für die Shell-Skripte zur Konfiguration von GRUB2. Über Zahlen in den Dateinamen, z.B. `30_os-prober` kann die Reihenfolge der Skripte bei der Ausführung von `grub2-mkconfig` bestimmt werden.
- `/etc/default/grub`
Durch das Setzen von Umgebungsvariablen in dieser Datei kann ein Großteil der Konfiguration eingestellt werden.

Shell-Skripte im Verzeichnis /etc/grub.d

Die innerhalb des Verzeichnisses `/etc/grub.d` enthaltenen Shell-Skripte werden während der Ausführung des Kommandos `grub2-mkconfig` interpretiert und daraus die Datei `/boot/grub2/grub.cfg` erzeugt. Die Reihenfolge der Einträge in `grub.cfg` ist hierbei abhängig von der Reihenfolge, in der die Skripte innerhalb von `/etc/grub.d` ausgeführt werden.

Über Zahlen am Anfang der Dateinamen kann diese Reihenfolge beeinflusst werden: `10_linux` wird demnach nach `00_header`, aber vor `30_os-prober` ausgeführt. Wichtig ist an dieser Stelle noch, dass alle Dateien ausführbar sind. Einige vordefinierte Dateien werden mit GRUB2 mitgeliefert:

- `00_header`
Datei für globale Umgebungsvariablen, z.B. für Dateilokationen, Graphikeinstellungen und Themes für GRUB2. Dieses Skript verarbeitet die Einstellungen von `/etc/default/grub`.
- `10_linux`
Erkennt vorhandene Linux-Kernel und erstellt automatisch die Boot-Einträge. Nur der Neueste wird am Bootmanager direkt angezeigt, alle weiteren sind durch ein Untermenü zugänglich.
- `30_os-prober`
Nutzt das Kommando `os-prober`, um Linux und weitere Betriebssysteme auf den Festplatten zu erkennen und entsprechende Menüeinträge vorzunehmen. Aktuell werden Linux, Windows, Hurd und Mac OS X erkannt.
- `40_custom`
Vorlage, um eigene Menüeinträge für das Boot-Menü zu erstellen. Erstellte Menüs werden automatisch der Datei `grub.cfg` hinzugefügt.
- `90_persistent`
Dieses Skript stellt sicher, dass direkt in `grub.cfg` eingetragene Änderungen zwischen den speziell markierten Zeilen `BEGIN` und `END` persistent bei einer Neugenerierung von `grub.cfg` erhalten bleiben.

Konfiguration von Boot-Einträgen

Eigentlich ist es nicht notwendig bei GRUB2 Veränderungen an der Konfiguration vorzunehmen, da über die mitgelieferten Shell-Skripte alle Betriebssysteme erkannt und für jedes automatisch ein Boot-Eintrag erzeugt wird. Für eigene Veränderungen ist ansonsten die Datei `40_custom` zuständig.

```
menuentry "Desktop - openSUSE 11.2" {
  insmod ext2
  search --set=root --label GRUB --hint hd0,msdos1
  linux /boot/vmlinuz-2.6.31.8-0.1-desktop root=/dev/sda2
  initrd /boot/initrd-2.6.31.8-0.1-desktop
}
```

Abb. 1: GRUB2 Boot-Eintrag für ein Linux-Betriebssystem.

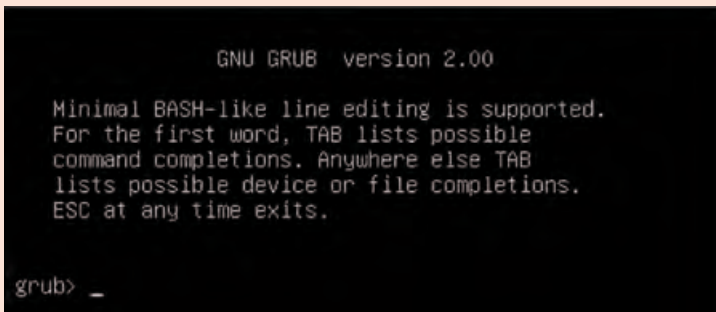


Abb. 2: GRUB2-Shell im Kommandomodus.

```
set superusers="root"
password_pbkdf2 root rub.pbkdf2.sha512.10000.FA0A-
F66111010524EA316D110...
password operator geheim

menuentry "Desktop - openSUSE 11.2 " --users operator {
```

Abb. 3: Absicherung eines Menüeintrages.

```
menuentry "SystemRescueCd" {
  loopback loop /systemrescuecd.iso
  linux (loop)/isolinux/rescuecd isoloop=/systemrescuecd.iso
  setkmap=de docache dostartx
  initrd (loop)/isolinux/initram.igz
}
```

Abb. 4: Menüeintrag für ein boot-bares ISO-Image.

Die Abbildung 1 zeigt einen Boot-Menüeintrag der in dieser Datei eingetragen wurde. Wichtig: Nach Änderungen an einem der Shell-Skripte ist das folgende Kommando aufzurufen:

```
grub2-mkconfig -o /boot/grub2/grub.cfg
```

Wie in Abbildung 1 zu erkennen ist, haben sich die Befehle der Konfigurationsdatei grundlegend geändert. Mit dem Befehl `insmod` können vom Bootmanager Module bei Bedarf geladen werden. Aktuell gibt es bereits ca. 200 Module für verschiedene Eventualitäten im Verzeichnis `/boot/grub2/i386-pc`.

Neu ist auch die Bezeichnung der Partitionen auf den Festplatten. Stand (`hd0, 0`) bei GRUB Legacy noch für die erste (0) Partition auf der ersten (`hd0`) Festplatte, beginnt die Zählweise bei GRUB2 bei den Partitionen bei 1 und einem zusätzlichen `msdos`. Andere mögliche Angaben wären auch `netbsd1` oder `gpt1`. Auch der Befehl zum Laden des Kernels wurde von `kernel` zu `linux` geändert.

GRUB2-Shell

Wie schon in der Vorgängerversion bietet auch GRUB2 eine Shell an, bei der schon im Boot-Menü auf Dateisysteme zugegriffen oder Einfluss auf defekte Boot-Einträge genommen werden kann. Weiterhin sind natürlich die Funktionen der TAB-Vervollständigung oder der PXE-Boot erhalten geblieben.

Insgesamt gibt es alles wie bei GRUB Legacy, nur mit viel mehr Möglichkeiten und Funktionen. Die Abbildung 2 zeigt die GRUB2-Shell, in der interaktiv Befehle eingegeben werden können.

Da die GRUB-Shell immer nur solange nützlich war, wie der MBR nicht überschrieben wurde, mussten Administratoren früher auf Rescue-Systeme von Drittanbietern zurückgreifen, um GRUB wieder im MBR zu installieren. Bei GRUB2 wird ein Befehl mitgeliefert, mit dem sehr einfach ein ISO-Image mit einer GRUB-Shell erstellt werden kann. Mit Hilfe des Kommandos `grub2-mkrescue -o grub.iso` kann ein ISO-Image, eine Diskette oder ein boot-barer USB-Stick erstellt werden.

Konsolensicherheit

Haben Benutzer einen physikalischen Zugriff auf die Konsole, so besteht bei GRUB mit seiner Shell die Gefahr, dass diese Personen auch Zugriff auf Dateien des Betriebssystems erhalten. Unter GRUB Legacy konnte dies mittels eines Kennwortes unterbunden werden. Unter GRUB2 wurde der Schutz um die Angabe eines Benutzernamens erweitert. So kann jetzt feingranuliert bestimmt werden, welcher Benutzer nach Angabe von Benutzername und Kennwort Zugriff auf bestimmte Menüs und Funktionen haben darf.

Auch bei GRUB2 gibt es die Falle, dass zunächst nur die englische Tastatur zur Verfügung steht. Das Kennwort für den Benutzer kann als Hash-Wert oder im Klartext eingetragen werden. Mit dem Befehl `grub2-mkpasswd-pbkdf2` kann der Hash-Wert eines eingegebenen Kennworts erstellt werden. Die Abbildung 3 zeigt ein Beispiel, wie ein Eintrag abgesichert werden kann.

Der Menüeintrag ist nun nur noch von einem „super-user“ oder dem Benutzer „operator“ nach Eingabe des Kennworts zugänglich.

Direkt ISO-Images booten

GRUB2 ermöglicht es, direkt ISO-Images, die auf einer Festplatte abgelegt sind zu booten. Hierzu wurde die Funktion `loopback` in GRUB2 implementiert. Um das Image einer Live-Distribution zu booten, müssen allerdings die Boot-Parameter der jeweiligen Distribution bekannt sein. Durch das Einhängen des ISO-Image und Auslesen der entsprechenden Parameter sind diese Informationen aber leicht zu besorgen. Leider hat sich hierbei herausgestellt, dass die Funktion `loopback` nicht bei allen Linux-Distributionen gleich gut funktioniert. Teilweise müssen Betriebssystemkomponenten gepatched werden. Die Abbildung 4 zeigt einen Menüeintrag für ein Rescue-System.

Fazit

GRUB2 stellt eine moderne Weiterentwicklung des GRUB Legacy Bootmanager dar. Von Haus aus wird es so ausgeliefert, dass eigentlich nie etwas angepasst werden muss. Trotzdem bietet es Möglichkeiten Veränderungen vorzunehmen. Egal ob passwortgeschützter Zugang, Netzwerk-Boot, Zugriff auf LVM und Software-RAID, über die entsprechenden Module sind alle Funktionen nutzbar. Zudem ist GRUB2 auch optisch vorzeigbar, damit durch sogenannte Themes jede grafische Spielerei umgesetzt werden kann.

Zu Recht setzt sich GRUB2 als Standard durch und löst den in die Jahre gekommenen GRUB Legacy ab.



Christian Fertsch
(info@ordix.de)

Glossar

Btrfs

Ein B-tree File System ist ein Copy-On-Write-Dateisystem, das von der Oracle Corporation als freie Software unter der GNU License (GPL) für das Betriebssystem Linux entwickelt wurde. Es weist viele Gemeinsamkeiten zu dem Dateisystem ZFS auf.

ELF-Loader

Das Executable and Linkable Format ist ein allgemeiner Standard für ausführbare Binaries, Objekt-Code, Shared Libraries und Core Dumps.

GRUB

Der Grand Unified Bootloader ist ein freies Bootloader-Programm, das oft zum Starten von unixoiden Betriebssystemen eingesetzt wird.

GRUB-2-Shell

Diese Shell ist eine Laufzeitumgebung von GRUB 2, mit der interaktiv Befehle ausgeführt werden können.

Lua

Lua ist eine erweiterbare Skriptsprache zum Einbinden in Programmen, um diese leichter weiterentwickeln und warten zu können.

LZMA/LZO

Der Lempel-Ziv-Markow (LZMA) und der Lempel-Ziv-Oberhumer (LZO) sind freie Datenkompressionsalgorithmen für verlustfreie Allzweck-Datenkompressionen. Beide Algorithmen erlauben eine extrem hohe Geschwindigkeit beim Entpacken von Archiven.

MBR

Der Master Boot Record ist der erste Sektor eines in Partitionen aufteilbaren Speichermediums, wie beispielsweise einer Festplatte.

ZFS

Das Zeta File System ist ein von Sun Microsystems entwickeltes transaktionales Dateisystem, welches zahlreiche Erweiterungen für die Verwendung im Server- und Rechenzentrumsbereich enthält.

Links

- ▶ [1] GRUB2-Webseite:
<http://www.gnu.org/software/grub/>

Bildnachweis

- ▶ © lanrentuku.com



Container unter Linux

LXC – Virtualisierung auf OS-Ebene

Was Solaris kann, können wir schon lange. So oder so ähnlich könnte eine Diskussion zwischen einem Linux- und einem Solaris-Anhänger zum Thema „Container“ beginnen. Zweifelsohne kopieren beide Seiten regelmäßig Innovationen voneinander, doch wer hat in diesem Bereich die Nase vorn? Solaris Container sind mittlerweile ein alter Hut und haben bei vielen Unternehmen längst den Weg ins Rechenzentrum gefunden. Doch was hat das Linux-Lager in diesem Bereich anzubieten? Der vorliegende Artikel beantwortet die Fragen zu den technischen Grundlagen der Linux Container (LXC), wie sie eingesetzt werden können und an welchen Stellen noch Probleme bestehen.

Was sind Container?

Container sind eine Virtualisierungstechnologie, die auf Applikationsebene operiert. Im Gegensatz zu auf Hypervisor basierenden Technologien wird aber keine zusätzliche Softwareschicht benötigt. Das Host-System teilt sich einen Kernel mit den Containern. Dies hat den Vorteil, dass die Container annähernd ohne Performance-Verlust betrieben werden können.

Ein Container ist (einfach gesagt) eine Gruppe von Prozessen, die über eine eingeschränkte Sicht auf die Betriebssystemressourcen verfügt. Innerhalb eines Containers lassen sich einzelne Applikationen oder komplette Systeme betreiben. So können z.B. ver-

schiedene Applikationen auf einem System existieren, ohne dass sich diese gegenseitig beeinflussen.

Da die Container den gleichen Kernel nutzen wie das Host-System, sind andere Betriebssysteme nicht lauffähig. Dies gilt allerdings nicht für Linux-Distributionen, die auf dem gleichen Kernel basieren. Container können sehr schnell eingerichtet werden und eignen sich somit hervorragend, um z.B. in einer sauberen Umgebung Tests vorzunehmen. Für Linux existieren neben LXC noch weitere Lösungen (OpenVZ und Virtuozzo) um Container einzusetzen [2].

Control Groups

Hinter LXC verbirgt sich nicht nur eine Abkürzung, sondern auch eine Reihe von Werkzeugen, die Control Groups (cgroups) [3] und Namespaces nutzen, um Container zu erstellen. Bei LXC werden sie genutzt, um die Ressourcen für einzelne Container zu beschränken. Seit längerem sind sie ein fester Bestandteil des Linux-Kernels und stehen somit bei allen gängigen, aktuellen Distributionen zur Verfügung.

Mit Hilfe von Control Groups können Prozesse zu Gruppen zusammengefasst und mit Subsystemen in Verbindung gebracht werden. Vom Kernel werden unterschiedliche Subsysteme zur Verfügung gestellt, welche verschiedene Aufgaben erfüllen. Diese lassen sich grob in folgende Kategorien zusammenfassen: Ressourcenlimitierung, Priorisierung, Protokollierung der Ressourcennutzung und Kontrolle.

Namespaces

Der entscheidende Schritt auf dem Weg zu Containern wird jedoch von den Namespaces, die eine Funktion des Linux-Kernels sind, ermöglicht. Natürlich ist es nützlich einem einzelnen Container zum Beispiel eine höhere Gewichtung bei der Nutzung der CPU zuzuweisen. Essentiell für den Einsatz von Containern ist aber, dass sie sich untereinander und das Host-System nicht beeinflussen und sich auch nicht verselbständigen können.

Genau an dieser Stelle kommen die Namespaces ins Spiel. Sie ermöglichen es, Prozessen eine eingeschränkte bzw. angepasste Sicht auf Betriebssystemressourcen zur Verfügung zu stellen und so die Prozesse zu isolieren. Die im Folgenden näher erläuterten Namensräume sind jeweils für eine eigene Ressource zuständig.

Der Namespace, der als erstes im Kernel implementiert wurde, ist der Mount Namespace. Er stellt einer Gruppe von Prozessen eine eingeschränkte Sicht auf die File-System-Hierarchie zur Verfügung. Die System-Calls `mount()` und `umount()` beeinflussen nur den Namespace des Prozesses, von dem sie aufgerufen wurden. So können mit Hilfe dieses Namespace chroot-ähnliche Umgebungen geschaffen werden, die jedoch sicherer und flexibler sind. Darüber hinaus kann eine Master-Slave-Beziehung zwischen einzelnen Mount Namespaces hergestellt werden. Dies ermöglicht es z.B., dass ein File System, welches in dem einen Namensraum eingehängt wurde, automatisch auch in den Slave Namespaces verfügbar ist.

Neben der unterschiedlichen Sicht auf einzelne Mountpoints ist es für die Container wichtig, auch einen eigenen PID-Namensraum zu besitzen. Durch die hierarchische Anordnung der PID Namespaces ist es dem Parent Namespace möglich, die Prozesse

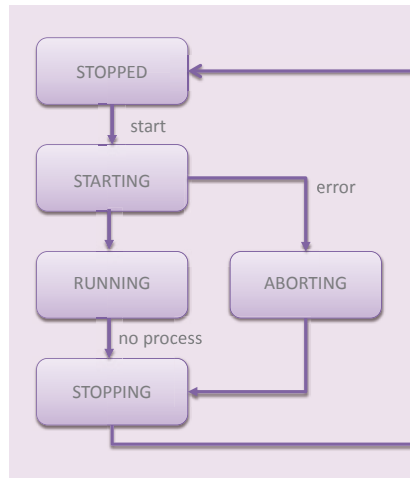


Abb. 1: Lebenszyklus eines Containers.

Parameter/Kategorie	Beschreibung
<code>lxc.arch</code>	Es kann die Architektur des Containers gewählt werden. So kann ein 32-Bit-Container auf einem 64-Bit-Host betrieben werden.
<code>lxc.utsname</code>	Dieser Parameter entspricht dem UTS Namespace und ermöglicht es, einen Host-Namen für den Container zu setzen.
<code>lxc.network</code>	Hinter dieser Kategorie verbergen sich die Parameter, die die Netzwerkfunktionen des Containers beeinflussen.
<code>lxc.console</code>	Hiermit wird die Angabe einer Datei ermöglicht, in der die Ausgabe der Konsole des Containers geschrieben wird.
<code>lxc.mount</code>	In dieser Kategorie können Mounts angegeben werden, die nur für den Container sichtbar sind. Dies können auch bind-Mounts von Ordnern des Host-Systems sein.
<code>lxc.rootfs</code>	Hier kann das Root-File System des Containers angegeben werden. Fehlt dieser Parameter teilt sich der Container das Root-File-System mit dem Host.
<code>lxc.cgroup</code>	Diese Kategorie ist die Schnittstelle zu den einzelnen cgroup-Subsystemen, die über das Kommando <code>lxc.cgroup.[subsystem]</code> angesprochen werden. Sie ermöglichen es z.B. den Arbeitsspeicher, den der Container nutzen darf, zu begrenzen.
<code>lxc.cap.drop</code>	Wenn der Container als <code>root</code> läuft, stehen die hier angegebenen Ressourcen dem Container nicht zur Verfügung, um z.B. zu verhindern, dass in dem Container Module geladen werden.

Abb. 2: Konfigurationsparameter und -kategorien.

seiner Childs zu sehen und zu beeinflussen. Dies führt dazu, dass ein Prozess in einem Child Namespace mehrere PID besitzt - eine in seinem Namensraum und eine in dem Parent Namespace. So kann es mehrere Container geben, die jeweils einen `init`-Prozess mit der ID 1 aufweisen.

Für gewisse Aufgaben, wenn z.B. komplette Systeme virtualisiert werden sollen, ist es erforderlich, dass der Container über einen eigenen Host-Namen verfügt. Welchen Node- und Domain-Namen der System-Call

Kommando	Beschreibung
<code>lxc-ls</code>	Der Befehl zeigt alle Container auf dem System.
<code>lxc-list</code>	Wie der Befehl <code>lxc-ls</code> , zeigt aber zusätzlich noch den Status an.
<code>lxc-info</code>	Gibt detaillierte Informationen zu einem Container aus.
<code>lxc-ps</code>	Zeigt die Zugehörigkeit der Prozesse zu den Containern.
<code>lxc-cgroup</code>	Ermöglicht Änderungen an <code>cgroup</code> -Subsystemen im laufenden Betrieb.
<code>lxc-monitor</code>	Wenn der Befehl gestartet wird, zeigt er die Statusänderungen des Containers an bis er beendet wird.

Abb. 3: Kommandos für den Betrieb von Containern.

Glossar

chroot

chroot steht für „change root“ und ist eine Funktion auf Unix-Systemen, um das Root-Verzeichnis zu ändern.

Parent Namespace

Ein Parent Namespace ist in einer Namespace-Hierarchie jeder Namespace, in dem weitere Namespaces erstellt wurden. Er hat die Möglichkeit die Ressourcen seiner Child Namespaces zu beeinflussen.

Child Namespace

Ein Child Namespace ist eine Umgebung in der mittels Namespaces unter Linux Ressourcen isoliert wurden. Ein Child Namespace ist durch seinen Parent beeinflussbar, er kann selber weitere Namespaces erstellen und ist für diese dann der Parent.

Loopback Interface

Das Loopback Interface ist eine virtuelle Schnittstelle, die vom Betriebssystem zur Verfügung gestellt wird. Sie kann z.B. genutzt werden, um mit einem Client auf ein Server-Programm zuzugreifen, das auf der selben Maschine läuft.

System V

System V ist der Name einer Version des Unix-Betriebssystems und unterscheidet sich unter anderem in der Vererbung zu anderen Versionen. So erhalten neu erzeugte Objekte die eingestellten Default-Eigenschaften des Nutzers.

IPC-Objekte

IPC steht für Interprozesskommunikation und stellt verschiedene Methoden bereit mit deren Hilfe Prozesse miteinander kommunizieren können. Ein IPC-Objekt ist eine konkrete Instanz einer IPC-Methode, also z.B. eine Message Queue oder ein SHM-Segment.

POSIX Message Queues

Message Queues sind eine Technik die es erlaubt, dass Prozesse asynchron miteinander kommunizieren können. Eine spezielle Form der Implementierung sind die POSIX Message Queues, die für verschiedene Betriebssysteme standardisiert sind.

AppArmor

Application Armor (deutsch: Anwendungs(programm)-Rüstung) ist eine freie Sicherheitssoftware für Linux, mit der Programmen einzeln bestimmte Rechte zugeteilt oder entzogen werden können.

SE Linux

Security-Enhanced Linux (deutsch: sicherheitsverbessertes Linux) ist eine Erweiterung des Linux-Kernels, die den ersten Versuch darstellt, das FLASK-Konzept des US-amerikanischen Geheimdienstes NSA umzusetzen.

`uname ()` zurückliefert wird durch den UTS Namespace gesteuert.

Über den NET Namespace können den einzelnen Containern eigene Netzwerkressourcen (Netzwerkarten, IP-Adressen, Sockets, usw.) zugewiesen werden. Nach dem Erstellen eines neuen Network Namespace verfügt dieser standardmäßig nur über ein eigenes Loopback Interface. Ihm kann aber auch der dedizierte Zugriff auf physikalische Interfaces ermöglicht oder virtuelle Interfaces zugewiesen werden. Über virtuelle Brücken können diese Interfaces verbunden und eine Verbindung nach außen hergestellt werden. Mit Hilfe des IPC Namespace können System V IPC-Objekte und POSIX Message Queues isoliert in Containern verwendet werden. Jeder Namensraum hat dabei eigene System V IPC-IDs und ein eigenes Dateisystem für die POSIX Message Queues.

Für bestimmte Anwendungszwecke ist es notwendig, dass man im Container über `root`-Rechte verfügen kann. Diese Rechte dürfen aber auf keinem Fall über den Container hinausgehen. Für diesen Zweck wurde der USER Namespace entwickelt. Er ist jedoch erst seit der Kernel-Version 3.8 vollständig implementiert. Dieser Namespace bildet die Grundlage, dass Prozesse innerhalb und außerhalb eines Containers unter verschiedenen User- und Group-IDs laufen können. Somit kann ein Prozess innerhalb des Containers mit der UID 0 - und daher mit allen Rechten - und außerhalb des Containers mit einer unprivilegierten ID laufen.

LXC und Sicherheit

Bevor im Folgenden die Handhabung von LXC vorgestellt wird, sollen die „noch“ vorhandenen Probleme und Einschränkungen aufgezeigt werden. LXC wird zwar von den führenden Distributionen unterstützt, aber meist noch als experimentell gekennzeichnet. Dies lässt sich vor allem auf Probleme bezüglich der Sicherheit zurückführen. Da der USER Namespace, wie schon erwähnt, erst ab der Version 3.8 im Kernel implementiert ist, kann er in den meisten Distributionen nicht genutzt werden. Diese verwenden, auch in den angekündigten Enterprise-Versionen, noch eine ältere Kernel-Version. Ohne den USER Namespace können Linux-Container jedoch nicht als sicher bezeichnet werden, da der User `root` in dem Container über mehrere Möglichkeiten verfügt, um das Host-System und damit auch andere Container zu beeinflussen.

Durch den Einsatz von AppArmor oder SELinux kann die Sicherheit jedoch auch ohne den USER-Namensraum gewährleistet werden. Dies erhöht aber die Komplexität, da die Konfiguration der beiden Lösungen nicht trivial ist. Einige Distributionen versuchen die Komplexität für den Administrator zu verringern, indem sie - wie z.B. Ubuntu - vorgefertigte AppArmor-Profile für LXC bereitstellen.

Der Weg zum eigenen Container

Vor dem Erstellen eines Containers sollte überprüft werden, ob der Kernel die Anforderungen erfüllt, um alle Funktionen (außer den USER Namespace) nutzen zu können. Dazu sollte der Kernel mindestens in der Version 2.6.29 vorliegen. Der Befehl `lxc-checkconfig` prüft, ob alle benötigten Namespaces und cgroup-Subsysteme verfügbar sind. Er folgt der gleichen Syntax wie die anderen LXC-Befehle, die alle mit `lxc-*` beginnen.

Die erste Entscheidung, die getroffen werden muss, ist, ob man einen Applikations- oder System-Container erstellen möchte. Für einen Applikations-Container, muss weiterhin die Frage beantwortet werden, welche Ressourcen isoliert, also welche Namespaces genutzt, werden sollen. Diese Frage stellt sich bei einem System-Container nicht, da in diesem Fall alle Ressourcen isoliert werden müssen.

Sobald ein Container erstellt ist (`lxc-create`), besteht er im Wesentlichen aus einer Konfigurationsdatei, die bei Ubuntu im Ordner `/var/lib/lxc/<Container>/` abgelegt wird. Ohne Anpassungen an dieser Datei werden für den Container die Ressourcen PIDs, IPC und Mountpoints isoliert. In Abbildung 1 ist der Lebenszyklus eines Containers dargestellt. Diese zeigt ebenso die Stati, die ein Container einnehmen kann.

Applikations- und System-Container

Eine Applikation kann mit dem Befehl `lxc-execute` in einem Container gestartet werden. Wenn dieser vorher noch nicht erstellt wurde, wird er automatisch generiert. Der erste Prozess in einem Applikations-Container ist `lxc-init` und er hat die PID 1. Anschließend startet er das Programm (PID 2), das als Argument beim `lxc-execute` mitgegeben wurde. Dadurch wird es ermöglicht, einen Daemon in einem Container zu starten. Ein System-Container wird über den Befehl `lxc-start` gestartet. Der Unterschied zu `lxc-execute` liegt darin, dass nicht `lxc-init` als erster Prozess gestartet wird, sondern `/sbin/init` falls nichts anderes angegeben wurde.

Mittels `lxc-stop` kann ein Container wieder beendet werden. Dies führt dazu, dass ohne Rücksicht alle Prozesse innerhalb des Containers terminiert werden.

Konfiguration und Betrieb von Containern

Einen guten Überblick über die möglichen Konfigurationsparameter und einige Beispiele finden sich in der Ubuntu Manpage [4]. In der Abbildung 2 sind die wichtigsten Parameter bzw. Kategorien aufgeführt. Diese vermitteln einen Einblick in den Funktionsumfang von LXC.

LXC ermöglicht es bei der Erstellung von Containern Templates zu verwenden. Ein Template ist ein Shell-Skript, in dem die Konfiguration des Containers und

Links

- ▶ [1] ORDIX news Artikel 01/2005 „Solaris 10 New Features“:
http://www.ordix.de/images/ordix/onevs_archiv/1_2005/solaris.html
- ▶ [2] ORDIX news Artikel 01/2008 „Alternative Virtualisierungslösungen: Alles virtuell mit OpenVZ und Virtuozzo“:
http://www.ordix.de/images/ordix/onevs_archiv/1_2008/virtualisierungslösungen.html
- ▶ [3] ORDIX news Artikel 03/2010 „Resource Management mit Linux: Control Groups - Linux legt Prozesse in Ketten“:
http://www.ordix.de/images/ordix/onevs_archiv/3_2010/control_groups.html
- ▶ [4] Ubuntu Homepage:
manpages.ubuntu.com/manpages/karmic/man5/lxc.conf.5.html
- ▶ [5] LWN.net Artikel „Namespaces in operation, part 1: namespaces overview“:
<http://lwn.net/Articles/531114/>

Bildnachweis

- ▶ © stockvault.net | Toy car | homero chapa
- ▶ © sxc.hu | White clouds in blue sky | takatuka
- ▶ © sxc.hu | container | g-point

beliebige Aufgaben, wie z.B. die Installation einer Distribution, enthalten sein können. Dies erleichtert die Erstellung komplexer Container und es lassen sich schnell mehrere gleichartige Container einrichten. Mit dem LXC-Paket werden einige Templates ausgeliefert, die als Beispiel dienen können.

Für den Administrator stehen verschiedene Kommandos zur Verfügung, die den Betrieb der Container erleichtern. Diese sind in der Abbildung 3 aufgelistet. Sie helfen Informationen zu den vorhandenen Containern anzuzeigen und die Ressourcenzuordnung zu beeinflussen.

Fazit

Die Control Groups und Namespaces bilden eine solide Grundlage für die Container-Virtualisierung unter Linux. Zudem sind sie ein fester Bestandteil des Kernels. Das ist auch der größte Vorteil, den LXC gegenüber OpenVZ und Virtuozzo bietet und wird wohl dafür sorgen, dass LXC in Zukunft einen größeren Marktanteil einnehmen wird als die beiden anderen Lösungen.

Es fehlen jedoch noch wichtige Funktionen - wie z.B. die Live-Migration. Dies und die angesprochenen Einschränkungen bei der Sicherheit könnten bei einer heutigen Entscheidung für eine Container-Lösung gegen LXC sprechen. So hat auch der Solaris-Admin aus der Einleitung beim Thema Container noch die Nase vorn - aber die Betonung liegt auf „noch“.



*Marius Dörlöchter
(info@ordix.de)*

Sie suchen ein passendes SEMINAR für die Administration Ihres Application Server?



Administration
und Konfiguration
für JBoss 5/ JBoss 7

Seminardauer: 3 Tage
Seminarpreis: 1.190 €
(zzgl. MwSt.)

Seminar ID: INT-11
INT-11_7

WebSphere
Application Server
Installation und
Administration

Seminardauer: 3 Tage
Seminarpreis: 1.390 €
(zzgl. MwSt.)

Seminar ID: INT-08

Oracle Weblogic
Administration
Grundlagen

Seminardauer: 3 Tage
Seminarpreis: 1.390 €
(zzgl. MwSt.)

Seminar ID: DB-ORA-50

Tomcat
Konfiguration
und Administration

Seminardauer: 3 Tage
Seminarpreis: 1.190 €
(zzgl. MwSt.)

Seminar ID: INT-07

Apache
HTTP Server
Administration

Seminardauer: 3 Tage
Seminarpreis: 1.190 €
(zzgl. MwSt.)

Seminar ID: INT-04



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

ORDIX AG