

- Die Abhängigkeit besteht nicht unbedingt zwischen den Paketen, sondern oft zu einer enthaltenen Bibliothek („shared library“). Dabei ist auf die „so“-Version zu achten
- Häufig ist die Analyse einfacher, wenn man zunächst die Pakete herunterlädt und sie dann lokal analysiert (mit „rpm“ beziehungsweise „dpkg“), als wenn man das Repository befragt (mit „yum“ beziehungsweise Apt-Tools).

Mit dieser Analyse ist es möglich, installierte MySQL-Pakete durch solche eines ande-

ren Paketierers zu ersetzen (die natürlich für die gegebene Plattform passend erstellt sein müssen), ohne dabei die Abhängigkeiten des Paket-Managements zu übergehen oder zu brechen. Der kritische Bereich sind die Anforderungen von installierten Anwendungen, die durch Kompatibilitäts-Pakete (meist „libs-compat“) erfüllt werden.

Falls das (mangels Angebot) nicht möglich ist, ist die Installation aus „tarball“ („tar.gz“) die saubere Alternative. Dafür gibt es allerdings keine Upgrade-Unterstützung, sodass der Administrator das selbst periodisch prüfen muss.



Jörg Brühe  
joerg.bruehe@fromdual.com

# MySQL<sup>®</sup> in der Oracle Public Cloud – den Durchblick behalten

Mario Beck und Carsten Thalheimer, ORACLE Deutschland B.V. & Co. KG

Während die MySQL-Datenbank bereits seit einigen Jahren im Amazon-, Microsoft- und Google-Cloud-Angebot ist, hat sich Oracle als Eigentümer von MySQL erstaunlich lange mit einem MySQL Cloud Service Zeit gelassen und den bereits seit Juli 2016 laufenden Testbetrieb während der Oracle OpenWorld freigegeben.

Auch im Jahr 2016 nutzte die Oracle-MySQL-Gruppe wieder die OpenWorld als Plattform, um praktisch alle Produkte der MySQL Enterprise Product Suite auf den neuesten Stand zu bringen. Der Fokus der rund fünfzehn MySQL Product Announcements (siehe „<http://forums.mysql.com/list.php?3>“) lag vor allem auf der neuen Version 8.0, die wie üblich als Development Milestone Release 1 (Beta Quality) als Binary und im Source Code für gängige Plattformen zur Verfügung gestellt wurde. Die Versionsnummern MySQL 6 (Namenskonflikt mit älteren Sun-Microsystems-MySQL-Plänen) und MySQL 7 (reserviert für MySQL NDB Cluster) werden somit übersprungen.

Nicht minder interessant sind die Neuigkeiten in der Oracle Public Cloud (OPC). Das Presse-Release liest sich zunächst recht nüchtern: Es wird von einem sicheren, kosteneffektiven und unternehmerorientierten MySQL Cloud Service (CS) gesprochen, ohne allerdings auf die technischen Details einzugehen.

Die in der Cloud verwendete MySQL-Version basiert auf MySQL Enterprise 5.7.16 und ist somit 100 Prozent kompatibel zu den im Jahr 2015 veröffentlichten MySQL 5.7 Community und kommerziellen Versionen. Somit ist gewährleistet, dass die Daten sowohl in der Cloud als auch On-Premise in eigenen, lokalen Rechenzentren verwendet werden. Eine Daten-Migration

in die Cloud beziehungsweise zurück in das Rechenzentrum ist somit aus Datenbank-Sicht denkbar einfach.

Das im MySQL Cloud Service eingesetzte Release ist die kommerzielle Version von MySQL. Das bedeutet, dass Anwender über die normale MySQL-Funktionalität hinaus noch weitere Features nutzen können. Hervorzuheben sind hier vor allem die Security-Erweiterungen wie transparente Datenverschlüsselung (TDE), Auditing und die integrierte SQL-Firewall.

Das erste Aufsetzen einer MySQL-Instanz in der Oracle-Cloud geht schnell von der Hand. Auf Basis eines vorkonfigurierten und MySQL-optimierten Oracle-Compute-Images wird eine virtuelle Betriebssystem-

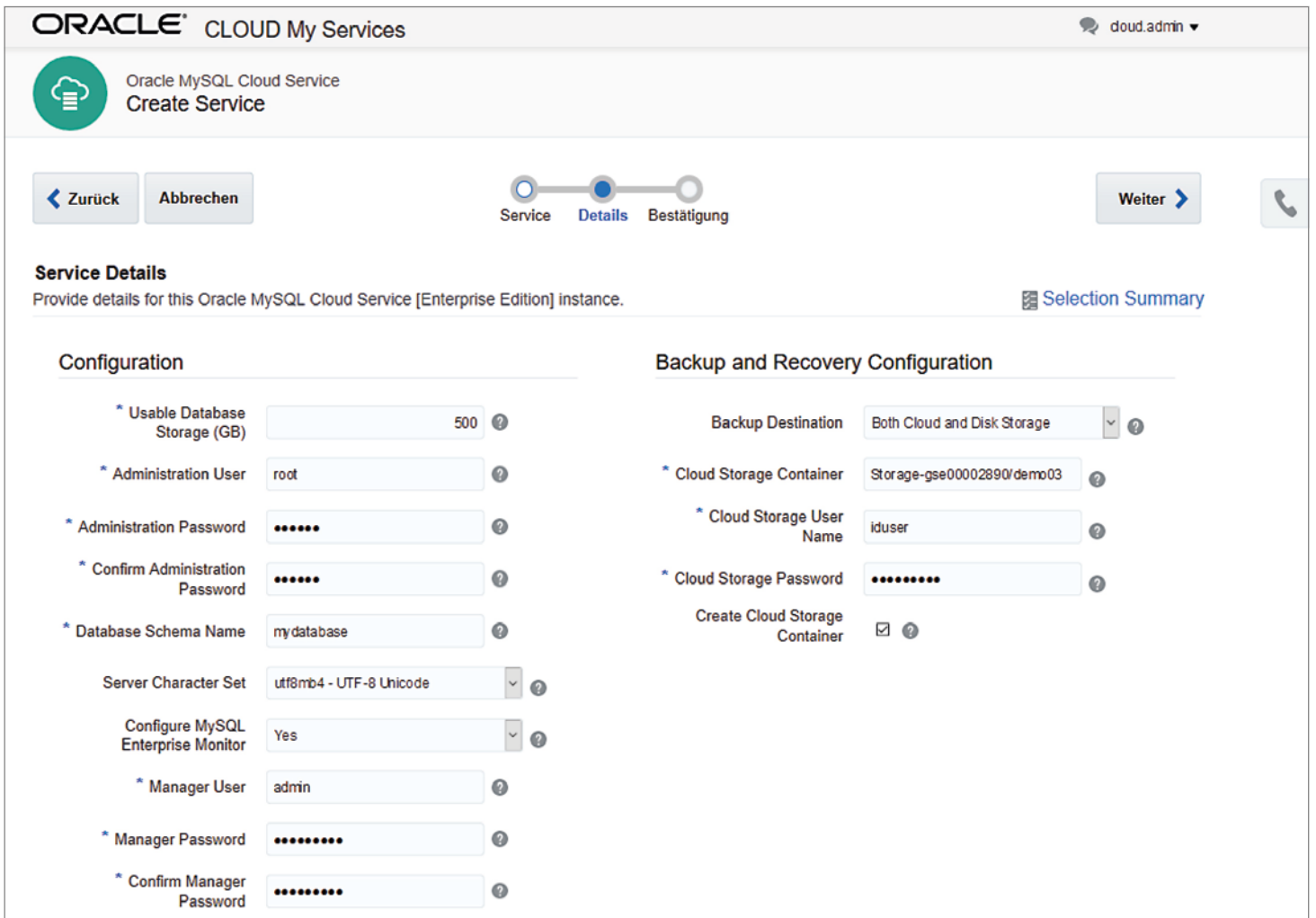


Abbildung 1: Setup einer MySQL-Cloud-Service-Instanz

Instanz mit MySQL ausgerollt. Notwendige Basisdaten wie Datenbankgröße, Zugangsdaten und CPU/Memory/Disk/Userdaten werden initial ausgewählt. Diese sind im Nachgang jederzeit anpassbar.

Der MySQL-Service kann auf Basis von „metered/messbar“ (OCPU pro Stunde) oder als „non-metered“ (OCPU per Monat) bezogen werden. „OCPU“ (Oracle Compute Unit) stellt dabei das Äquivalent zu zwei Hardware-Ausführungs-Threads mit 15 GB Arbeitsspeicher dar. Zusätzlich müssen noch der Compute Block Storage (vergleichbar zu lokaler Festplatte) und der Storage Cloud Service (für Datenbank-Backups) angegeben werden (siehe Abbildung 1).

MySQL CS ist nur zum Teil gemanagt. Wenn ein neuer Datenbank-Anwender oder ein neues Datenbank-Schema in MySQL benötigt wird, muss der Endanwender diese Konfiguration selbst durchführen. Sollte er bei administrativen Arbeiten oder beim MySQL-Betrieb Probleme haben, kann er jederzeit auf den Oracle Premier Support

zurückgreifen. Support ist in allen Abrechnungsmetriken enthalten und identisch zum Support der kommerziellen MySQL-konsultativen Support, der für proaktive Ansätze zur Diagnose von Leistungs- oder Stabilitäts-Problemen, Architekturberatung, aber auch zum Code-Review verwendet werden kann.

Die Verknüpfung des Oracle Premier/konsultativen Supports und der Automatisierung von Maintenance-Aufgaben von MySQL CS macht den MySQL-Betrieb auch ohne tiefgehendes Datenbank-Wissen sehr einfach. Vollautomatisiert sind heute bereits die Datensicherung und Wiederherstellung, tiefgehendes Monitoring sowie stark vereinfachtes Patch-Management.

### Die MySQL-Betriebssystem-Umgebung

MySQL steht derzeit auf Oracle Linux 6.6 mit Unbreakable Enterprise Kernel/ext4

zur Verfügung. MySQL-Tasks werden durch Oracle Linux priorisiert. Als Storage-Backend kommt die Oracle ZFS Appliance zum Einsatz.

Der Zugriff auf Oracle Linux erfolgt auf Basis von SSH für den User „opc“. Die „opc“-Authentifizierung erfolgt auf Basis von SSH-Keys, die während der MySQL-Konfiguration automatisch oder auf Basis von existierenden SSH-Keys durchgeführt werden. Nach dem Einloggen kann man über „sudo“ sehr einfach administrative Befehle ausführen und so auf Wunsch jederzeit vollständigen Einfluss auf MySQL und auf das Betriebssystem nehmen.

Netzwerkseitig steht standardmäßig ein Netzwerk-Interface mit einer öffentlichen Adresse zur Verfügung. Dieses ist durch eine Firewall für jeden Zugriff von außerhalb (außer SSH) gesperrt. Die Konfiguration auf Basis eigener Netzwerke zur Interkommunikation mit anderen Oracle Cloud Services, privaten Netzwer-

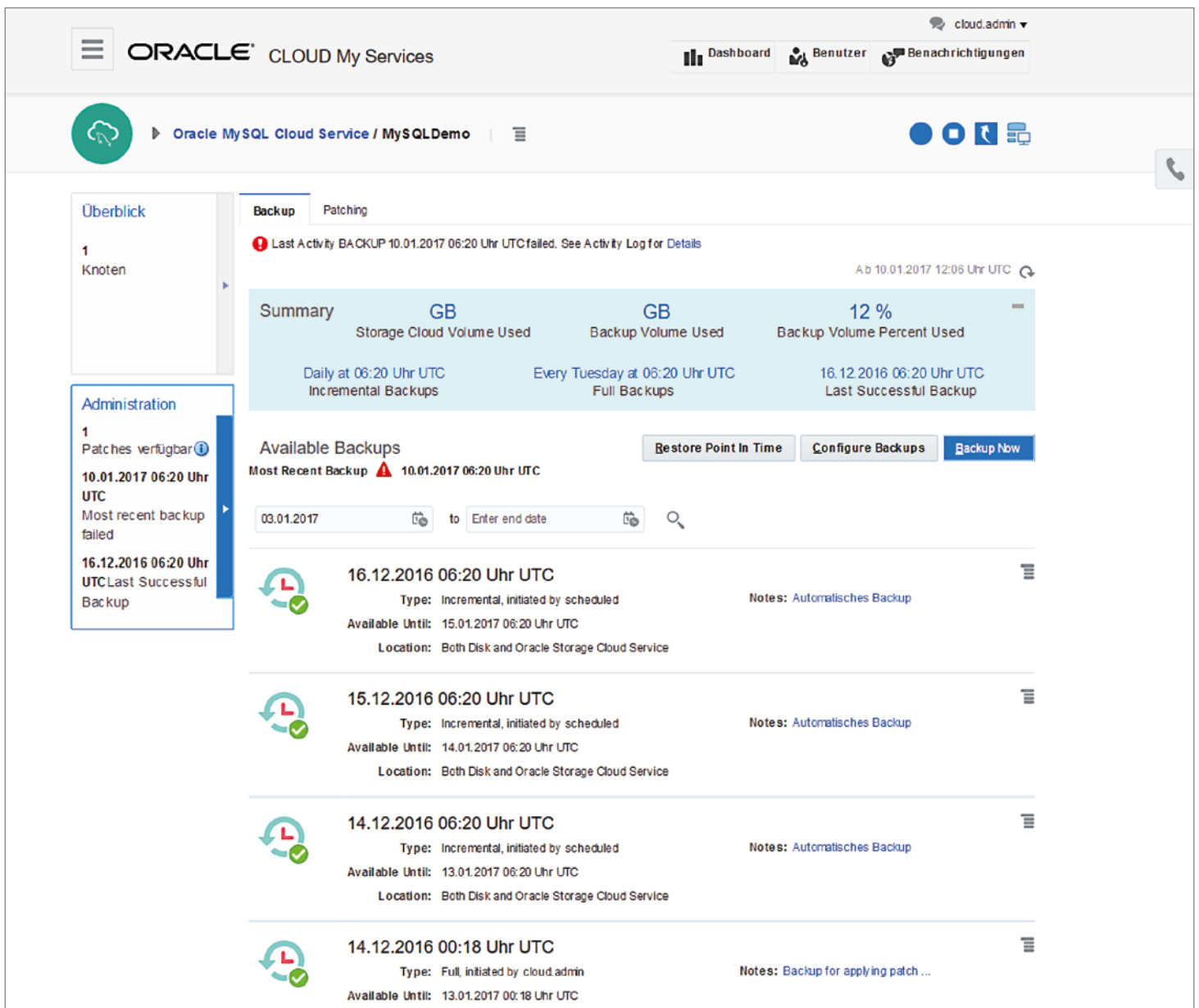


Abbildung 2: Das MySQL-Backup-Dashboard

ken oder VPN ist durch entsprechende Menüpunkte möglich.

## Installationsdetails zum MySQL CS

Der MySQL CS verwendet die gleichen Binaries wie die kommerzielle MySQL-Version der MySQL-Homepage. Allerdings werden die CS derzeit nur etwa einmal im Quartal aktualisiert, während die Originalpakete etwa alle sechs bis acht Wochen erneuert werden. Wer MySQL von verschiedenen Linux-basierten Installationen kennt, muss sich zunächst erstmal ein wenig umgewöhnen: Die Installation

richtet sich nach dem Optimal Flexible Architecture Standard (OFA). Der für Oracle DBAs bekannte Standard spielt bei herkömmlichen MySQL-Installationen oftmals keine Rolle. Daher findet man die zentrale Konfigurationsdatei „my.cnf“ auch nicht, wie in Linux-Installationen üblich, in „/etc“, sondern in „/u01/bin/mysql-5.7.x“. Das MySQL-Basisverzeichnis („basedir“) zeigt auf „/u01/bin/mysql“ und die MySQL-Daten („datadir“) sind in „/u01/data/mysql“.

Der MySQL-Dienst läuft nicht, wie bei Linux üblich, unter dem Userkontext „mysql“, sondern dem Oracle-Public-Cloud-Standard folgend unter „oracle“. Für die Arbeit mit dem MySQL-Client muss daher

zunächst von „opc“ zum Benutzer „oracle“ gewechselt werden.

Die On-Premise-Standard-Konfiguration von MySQL ist typischerweise stark vereinfacht, um möglichst viele Anwendungsfälle und unterschiedlichste Hardware-Ressourcen abzudecken. Im Gegensatz hierzu ist die Konfiguration von MySQL CS hochgradig an die Bedürfnisse von MySQL in der Oracle Public Cloud angepasst.

Die Konfiguration von „innodb-buffer-pool-size“ ist automatisch an den verwendeten Compute Shape (OCPU/Memory) angepasst. Zusätzlich sind zahlreiche weitere Optimierungen in „my.cnf“ eingeflossen: „Number of IO threads“, „innodb-

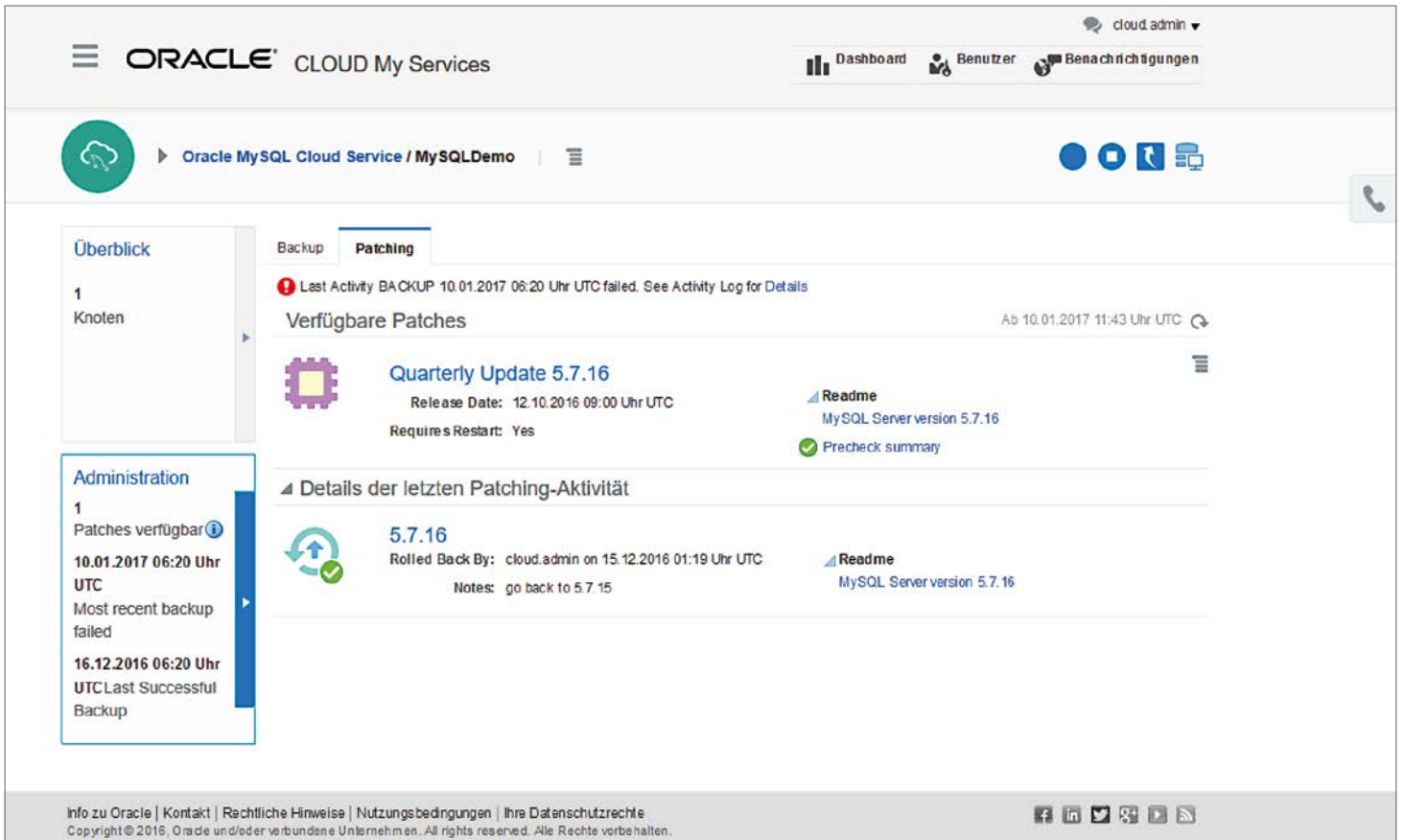


Abbildung 3: Das MySQL-Patch-Dashboard

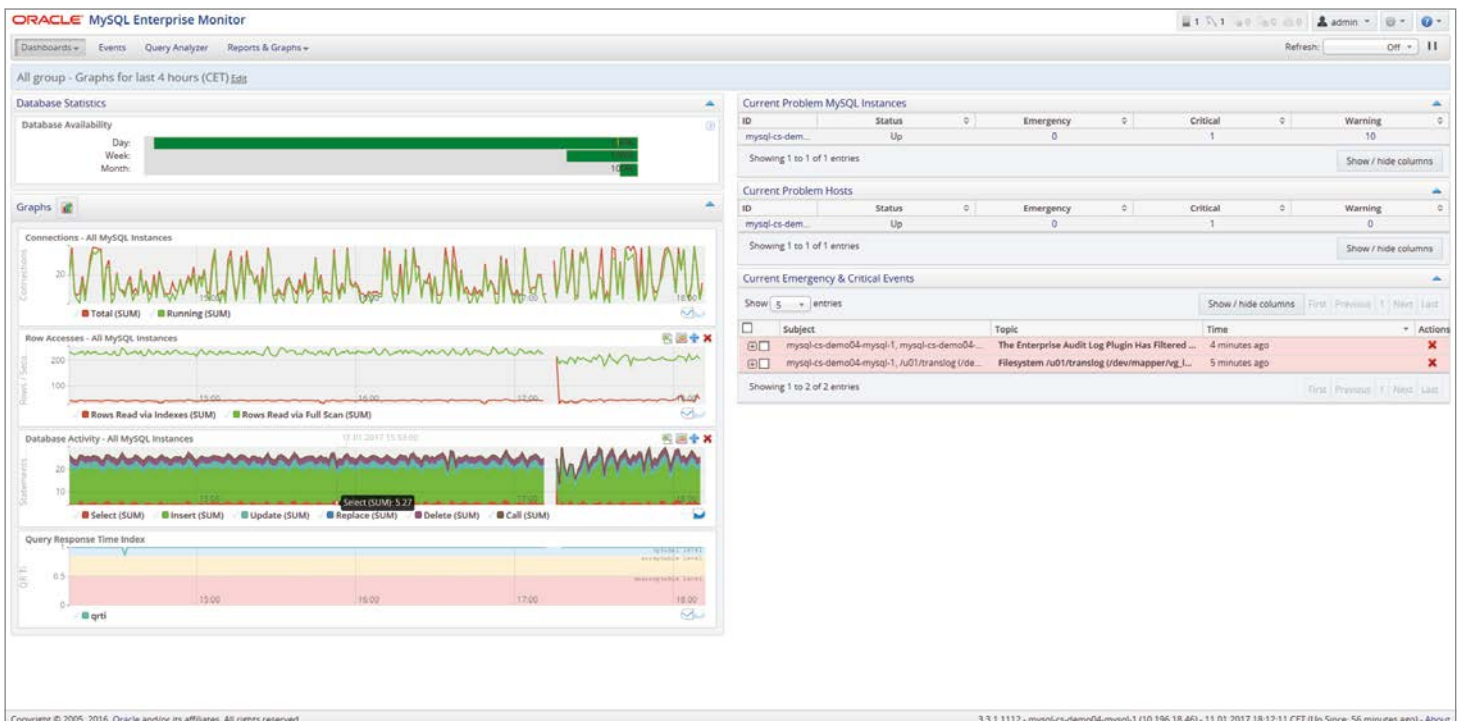


Abbildung 4: Das MySQL-Patch-Dashboard

autoinc-lock-mode=2", „innodb-flush-method=O\_DIRECT\_NO\_FSYNC“, „Thread Pool Enabled“.

Alle Einstellungen können sehr einfach auf On-Premise-Installationen angewendet werden. Die „my.cnf“-Konfigu-

ration von MySQL CS stellt daher auch einen sehr guten Startpunkt für MySQL (auf Basis Linux) dar. Auch die Transak-

tionslogs (Binlogs) werden entgegen der MySQL-On-Premise-Standard-Konfiguration mit aufgezeichnet und stehen für eine eventuelle MySQL-Replikation und/oder für „Point in Time Recovery“-Funktionalitäten zur Verfügung. Die MySQL/Binlog-Instanz ist als MySQL-Master konfiguriert. Eine zweite beziehungsweise beliebig viele MySQL-Instanzen können daher sehr einfach an eine Standard-Instanz angebunden werden, um eine MySQL-Replikationsfarm für Skalierungszwecke oder zwecks Redundanz zu bilden.

## MySQL Backup/Restore und Patching

Jeder MySQL-Service wird automatisch durch MySQL Enterprise Backup täglich gesichert. In dem vordefinierten Setup werden die Backups täglich inkrementell und wöchentlich vollständig ausgeführt. In Verbindung mit den Transaktionslogs lassen sich die Daten sekundengenau wiederherstellen.

Im Rahmen des Storage Cloud Service sind die Backups für dreißig Tage gespei-

chert. Alle Backup-Daten, Konfigurations-Dateien und MySQL-Logs können jederzeit von der Oracle Public Cloud heruntergeladen und für On-Premise-Zwecke verwendet werden. Alle Schritte für das Backup/Restore sind sehr einfach über intuitive Oberflächen des MySQL-Dashboards konfigurierbar. Die Wiederherstellung eines Backups erfolgt ebenfalls über das Dashboard. Besonders erwähnenswert ist hier die automatische Integration von Restore und Binary-Log-Recovery. Im Dashboard wählt der Administrator lediglich den wiederherzustellenden Zeitpunkt. Die Ausführung von Full-Restore, inkrementellem Restore und anschließendem Recovery anhand der Binary-Logs erfolgt voll automatisiert (siehe Abbildung 2).

Ein ähnliches Bild ergibt sich bei den MySQL-Patches: MySQL-CS-Updates werden etwa vierteljährlich im Rahmen der monatlichen OPC-Maintenance ausgerollt. Allerdings wird dabei die neue MySQL-Version nicht installiert, sondern nur in Verbindung mit einem Hinweis „One or more Patches available“ bereitgestellt. Das eigentliche Patching wird vom Administrator angestoßen.

Bevor ein Patch angewendet wird, erfolgt eine tieferegehende Prüfung der Kompatibilität und verfügbaren Ressourcen (Plattenplatz). Außerdem erfolgt ein Full-Backup, um notfalls ein Rollback durchführen zu können. Erst nach diesem Pre-Check erfolgt das Update, das eine kurze Downtime von einigen Minuten nach sich zieht. Das Einspielen des Patches kann über das gleiche Vorgehen widerrufen/zurückgesetzt werden (siehe Abbildung 3).

## MySQL-CS-Monitoring

Es gibt zwei Arten der MySQL-CS-Überwachung. Während sich das Überwachen auf Basis der Service-Konsole auf die Überwachung der verbrauchten Einheiten CPU/IO/Memory bei „metered Service“ richtet, steht mit dem ebenfalls auf Wunsch konfigurierten MySQL Enterprise Monitor ein sehr mächtiges Tool für die Live-Überwachung zur Verfügung. Es bietet einen sofortigen Überblick über den Status der MySQL-Datenbanken und überwacht kontinuierlich alle MySQL-Ser-

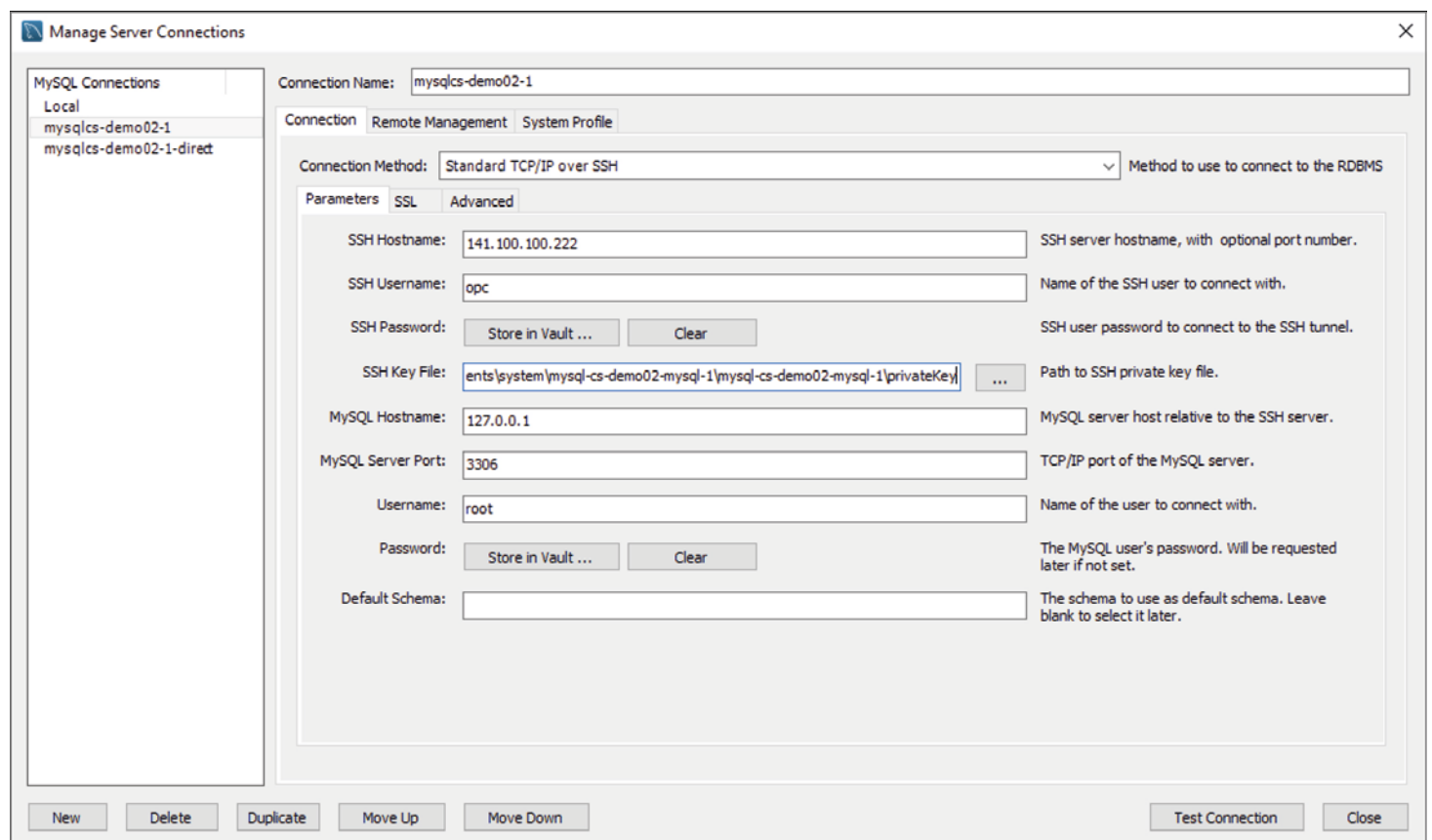


Abbildung 5: Konfiguration des MySQL-CS-Zugriffs in Workbench

ver mit fast 700 MySQL- und Betriebssystem-spezifischen Variablen. Zusätzlich informiert es dank zahlreicher Ratgeber über erkannte Probleme aus mehr als fünfzehn Kategorien (Sicherheit, Datensicherung, Administration, Zugriffsberechtigungen, SQL-Firewall, SQL-Befehle etc.). Über die Funktion der Abfrage-Analyse können SQL-Statements in einem konfigurierbaren Zeitraum eingesehen und ausgewertet werden. Gerade für Performance-kritische Einsätze lassen sich Probleme sehr schnell erkennen und effektiv beheben (siehe Abbildung 4).

## Zugriff auf MySQL

Grundsätzlich kann man über die MySQL-Konnektoren/ODBC, über den MySQL-Kommandozeilen-Client oder die MySQL-Workbench auf die MySQL-Instanz zugreifen. Allerdings sind standardmäßig alle Ports für den Zugriff von außen gesperrt, sodass man für den direkten Zugriff auf MySQL zunächst den MySQL-Port 3306 im OPC-Access-Dashboard zulassen muss. Alternativ besteht auch die Möglichkeit, den beschriebenen SSH-Zugriff als Tunnel zu verwenden. Ein kleiner Fallstrick ist bei der Verwendung des beliebten Putty Clients unter Windows zu beachten: Möchte man Putty zum Zugriff auf MySQL CS verwenden, müssen zunächst die von MySQL CS bereitgestellten SSH-Keys in ein Putty-kompatibles Format („Putty Private Key File.ppk“) mittels „PuTTYgen“ konvertiert werden.

Für administrative Aufgaben empfiehlt sich die MySQL-Workbench. Damit kann man MySQL administrieren, aber auch als SQL-Client oder Entwicklungstool mit ER-Diagrammen verwenden. Die MySQL-Workbench bietet ebenfalls die Möglichkeit, auf Basis von Performance-Reports Leistungsdaten, Index-Verwendung und Ressourcen-Belegung auf Benutzer- und Anwendungsebene ohne SQL oder administrative MySQL-Kenntnisse einzusehen.

Die MySQL-Workbench steht für alle gängigen Betriebssysteme zur Verfügung. Sie kann als kostenfreie Version von „<http://www.mysql.com>“ heruntergeladen werden. Alternativ stellt Oracle eine spezielle Version mit erweiterten Funktionen für den MySQL CS unter „<https://support.oracle.com>“ zur Verfügung.

Die MySQL-Workbench unterstützt ebenfalls SSH-Tunneling. Eine Konvertierung der SSH-Schlüssel wie für Putty ist beim Einsatz der MySQL-Workbench nicht notwendig. Das Einrichten zum Zugriff auf MySQL CS über einen SSH-Tunnel vereinfacht sich daher auf einige grundsätzliche Angaben in Verbindung mit dem privaten SSH-Schlüssel (siehe Abbildung 5).

Besondere Bedeutung bekommt die MySQL-Workbench während Datenbank-Migrationen. Der MySQL-Workbench-Migration-Wizard unterstützt und überprüft die einzelnen Migrationsschritte, übernimmt sehr einfach Daten aus einer beliebigen Quelle (MS Access, Postgres, Sybase) und überträgt diese logisch in ein neues Ziel. Noch einfacher können die Daten mit der MySQL-Workbench-Schema-Transfer-Funktion übertragen werden, die allerdings als Quelle und als Ziel eine MySQL-Datenbank voraussetzt.

## Der MySQL Cloud Service

In typischen Kunden-Szenarien werden nicht nur einzelne IT Services in der Cloud abgebildet. Vielmehr betreiben Kunden oftmals komplette Lösungen in der Cloud.

Daher verlagern sie nicht nur den Datenbank-Betrieb in die Cloud, sondern ganzheitlich auch die mit den Datenbanken verknüpften Applikations- und Infrastruktur-Umgebungen. Die OPC bietet daher bereits seit Längerem nicht nur einen MySQL-Datenbank-Service, sondern auf Wunsch auch anhängende (Applikations-) Software-Pakete (wie LAMP) im Rahmen der Compute Cloud Services mit an. Die OPC beschränkt sich hierbei nicht nur auf Linux, entsprechende Applikationspakete sind auch in Verbindung mit Oracle Solaris oder Microsoft Windows möglich. Auch die Oracle-eigenen Entwicklungstools (Oracle Java Cloud Service und Application Container Cloud) können so sehr einfach mit dem MySQL CS kombiniert und effektiv betrieben werden.

## Fazit

Die Oracle Public Cloud bietet seit Längerem Lösungen für sehr viele Einsatzzwecke und praktisch jede Unterneh-

mensgröße an. Das Portfolio geht von einfachen Einstiegslösungen bis hin zu exklusiv nutzbaren Lösungen auf Basis einer Oracle Exadata. Das ohnehin sehr große OPC-Produktportfolio wird mit MySQL CS erneut um einen wichtigen Baustein erweitert und für Cloud-interessierte Kunden nochmals attraktiver.

Erste Erfahrungen mit MySQL lassen sich über „Free Trials“ unter „<https://cloud.oracle.com>“ sammeln. Oracle stellt hier zeitlich und leistungsmäßig begrenzte Test-Accounts zur Verfügung, um MySQL CS und/oder andere Oracle-Cloud-Angebote einzeln oder im Zusammenspiel ausführlich zu testen.



Carsten Thalheimer  
carsten.thalheimer@oracle.com



Mario Beck  
mario.beck@oracle.com