

Seit letztem Juni ist der Amazon-Service RDS verfügbar. Amazon bietet mit dem Relational Database Service Oracle- und MySQL-Datenbanken in der Cloud an und regelt gerade die Lizenzierung. Der Artikel zeigt einen ersten Blick mit der Oracle-Brille in die Wolke und unterzieht den Service einem Praxistest.

# Amazon RDS: Wenn der DBA auf seiner eigenen Wolke schwebt

Martin Berger, Trivadis AG

On-Demand-Dienste aller Art gibt es mittlerweile einige; der bekannteste und größte unter den Anbietern ist sicher Amazon AWS. Bereits seit längerer Zeit sind virtuelle Maschinen, Storage etc. bei Amazon auf Stundenbasis in verschiedenen Größen und Leistungsklassen zu beziehen. Auch das Einbinden eigener konfigurierter Maschinen ist möglich. Oracle unterstützt Amazon schon seit längerem und stellt kostenlose, vorkonfigurierte Maschinen inklusive installierter RDBMS-Software (Elastic Compute Cloud) bereit. Aber wie sieht es mit der Lizenz aus? Amazon AWS hat es bis jetzt dem

Kunden überlassen, seine Software rechtmäßig bei Oracle zu lizenzieren. Auch das Patchen, Backup und Monitoring wurde dem Kunden und seiner Maschine überlassen.

Mit dem Relational Database Service (RDS) [1] kann man bei Amazon nun direkt eine reine Datenbank On-Demand beziehen, ohne sich um Betriebssystem, Parameter und Packages zu kümmern. RDS ist Teil des Amazon-Web-Service-Angebots und wird über die gleiche Kreditkarte abgerechnet sowie über dasselbe Web-GUI verwaltet wie etwa der Storage-Dienst „S3“ oder die Elastic Compute Cloud „EC2“ mit

vorgefertigten Maschinen-Images. Mit ein paar wenigen Mausklicks und der Angabe von Parametern zu Name, Größe und Leistung bekommt man in kürzester Zeit eine funktionsfähige Datenbank-Instanz; der Client benötigt nur noch die Verbindungsdaten zur Datenbank. Abgerechnet wird auf Stundenbasis, für die Kalkulation stellt Amazon ein Tool (Simply Monthly Calculator) zur Verfügung [2]. Als Datenbank-Produkt wird nicht nur Oracle angeboten, auch MySQL steht zur Auswahl.

RDS ist ein reiner Datenbank-Service, der in verschiedenen Größen- und Leistungsklassen zur Verfügung steht. Der DBA bekommt keinen Zugriff auf das darunterliegende Betriebssystem. Als Oracle-Edition gibt es Standard Edition One, Standard Edition und die Enterprise Edition. Es stehen zwei verschiedene Lizenzierungsmodelle zur Verfügung. Bei allen drei Oracle-Editionen kann die eigene Lizenz „mitgebracht“ werden (Bring-Your-Own-License), die Übernahme der Lizenzgebühren in den On-Demand-Kosten ist nur mit der Standard Edition One möglich (License Included). Generelle Fragen rund um RDS wie Zugriff, Backup etc. sind in den FAQ [3] kurz und bündig erklärt.

Leistungsmäßig stehen fünf verschiedene Instanzklassen zur Verfügung, alle basieren auf einem 64-Bit-Linux-OS. Von 1,7 GB bis 68 GB Memory ist alles möglich, CPU-mäßig rechnet Amazon mit der Elastic Compute Unit (ECU), es sind 1 bis 26 ECU möglich. Eine ECU entspricht der CPU-Kapazität eines 1,0 bis 1,2 GHz 2007 Opteron

Abbildung 1: Instanz erstellen

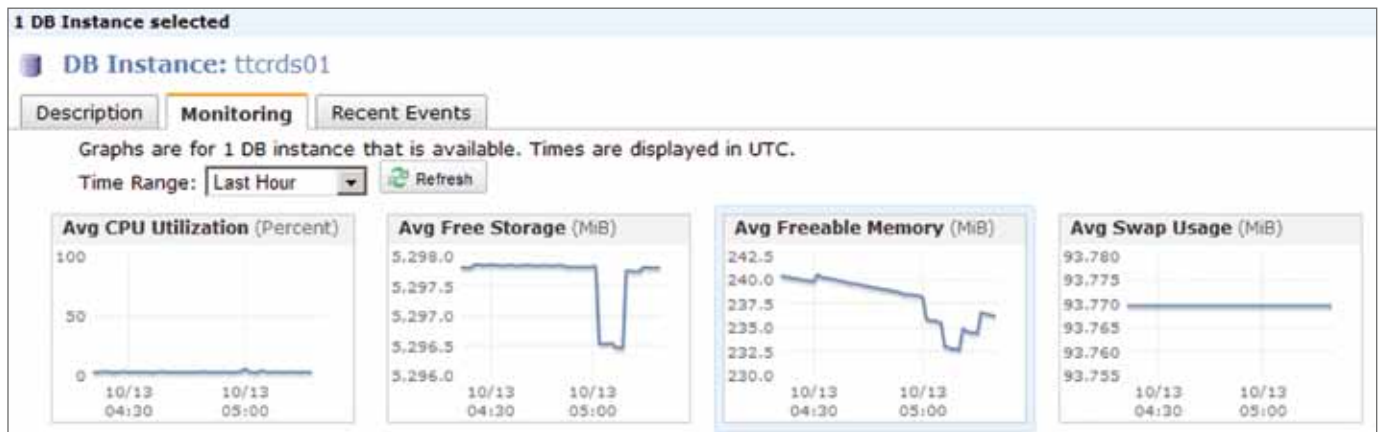


Abbildung 2: Basis-Monitoring

oder Xeon Prozessors. Storage kann von 10 GB bis zu 1 TB bereitgestellt werden. Je nach Klasse und Region, in der die Datenbank erstellt wird, sind die Kosten unterschiedlich; der Speicherplatz für Backups und Snapshots der Datenbank muss in jedem Fall separat bezahlt werden, ist aber mit ca. 0,15 US\$ pro GB und Monat günstig. Als Vereinfachung für den täglichen Betrieb übernimmt Amazon die klassischen zwei DBA-Tätigkeiten:

- **Automatic Host Replacement**  
Amazon RDS ersetzt bei Hardware-Defekt die Instanz automatisch auf einem neuen Server.
- **Point-in-time-Recovery**  
Die Datenbank wird zu einem gewünschten Zeitpunkt zurückgestellt, was jedoch nur in einer neuen Instanz möglich ist.

**Arbeiten mit Amazon RDS**

Eine Datenbank ist mit der Amazon-Web-Oberfläche (AWS-Konsole) oder dem Kommandozeilen-Tool „RDSCLI“ (bei Amazon downloadbar) innerhalb weniger Minuten erstellt. Nach der Angabe von Edition, Leistungsklasse, Menge an Storage, Bezeichnungen und einem Master-Benutzer mit Passwort wird die Instanz vollautomatisch erzeugt (siehe Abbildung 1).

Über eine zuweisbare DB-Security-Gruppe kann der Zugriff auf die Datenbank bis auf Stufe IP/Port geregelt werden, mit dem DB-Parameter „Group“ können die Datenbank-Parameter gesteuert werden. Alle Angaben im Web-GUI können auch per Kommandozeile ausgeführt werden. Listing 1 zeigt ein Beispiel für die Erstellung einer Enter-

```
mbg@srv00112 :~/RDSCLI-1.4.007$
rds-create-db-instance ttrcrds01
--allocated-storage 20 --db-
instance-class db.m1.small
--engine oracle-ee --master-
user-password manager --master-
username awsuser --license-model
bring-your-own-license --db-name
ttrcrds01 --port 1521
DBINSTANCE mynew db.m1.small
oracle-ee 20 awsuser
```

Listing 1

prise Edition mit eigener Lizenz mit dem Kommandozeilen-Tool RDSCLI [5].

Nach dem Erstellen der Datenbank muss beim Client das TNSNAMES.ORA angepasst werden, wie das bei einer lokalen Datenbank auch der Fall ist, und schon ist die Datenbank „ready-to-use“. Als Alternative für den Client ist aber auch Easy Connect möglich. Daten wie Servername, Instanzname etc. können bei der AWS-Konsole per Mausklick oder CLI abgefragt werden.

Beim Anlegen der Datenbank muss ein Master-Benutzer angegeben werden. Er hat in der Datenbank sehr hohe Privilegien (Rolle „DBA“), bekommt aber keine SYSDBA-Rechte. Diese bleiben im Besitz von Amazon.

- **Patches**  
Es kann gewählt werden, ob die Aktualisierung während eines vom Benutzer konfigurierten Wartungsfensters automatisch oder manuell durchgeführt werden soll. Neue Patches werden via E-Mail und im AWS-Forum rechtzeitig angekündigt [4].
- **Backups**  
Die Datenbank wird in einem vom Benutzer definierten Backup-Fenster mittels User Managed Backups automatisch gesichert.

Darüber hinaus bietet Amazon folgende Funktionalitäten:

- **Push-Button-Scaling**  
Dynamisches Ändern von Instanzklassen, Memory, CPU und Storage
- **Datenbank-Snapshots**  
Mit einem Snapshot werden der aktuelle Zustand der Datenbank gesichert und eine aktuelle Kopie erzeugt. So ist es möglich, mit einem Snapshot neue Datenbanken zu erstellen, beispielsweise für Test-Umgebungen.

```
SQL> SELECT comp_name,version,status
2 FROM dba_registry;

COMP_NAME                                VERSION                                STATUS
-----
Oracle Text                               11.2.0.2.0                            VALID
Oracle Database Catalog Views            11.2.0.2.0                            VALID
Oracle Database Packages and Types      11.2.0.2.0                            VALID
```

Listing 2

Für viele SYSDBA-Befehle hat Amazon deshalb Workaround-Prozeduren erstellt. Nachfolgend eine Auswahl von Prozeduren, die Amazon für DBA-Jobs anbietet:

- rdsadmin\_util.flush\_shared\_pool
- rdsadmin\_util.kill(sid, serial#)
- rdsadmin\_util.alter\_default\_tablespace(,usersnew')

Amazon stellt zurzeit leider nur die Komponente Oracle-Text zur Verfügung, andere Komponenten wie Apex, JVM und Label Security sind geplant (Aussage eines AWS-Mitarbeiters im Amazon-Forum), aber aufgrund der Einschränkung mit dem OS nicht einfach umzusetzen (siehe Listing 2). Im Hintergrund werden Datenfiles mit Oracle Managed Files (OMF) verwaltet. Standardmäßig entstehen Tablespace mit „AUTOEXTEND=YES“. Neue Tablespaces werden OMF-like mit der Größe von 100 MB angelegt, und für die Memory-Verwaltung ist AMM (Automatic Memory Management) parametrisiert.

### Monitoring und Tracing

Für das Datenbank-Monitoring stehen Basis-Anzeigen zu CPU-Auslastung, Storage etc. bereit (siehe Abbildung 2). Zusätzlich besteht die Möglichkeit, über den Amazon-Service CloudWatch Metriken zu definieren und die Alarmierung festzulegen. Die Alarmierung ist kostenpflichtig (pro Metrik inklusive Alarmierung 0,50 US\$ pro Monat) und kann über die Konsole grafisch konfiguriert werden. Eine Anzeige der Datenbank-Daten, Benutzer und SQL-Statements durch eine GUI wie Insider für SQL Developer oder Toad ist jederzeit möglich und funktioniert mit dem beim Erstellen angegebenen Benutzer

problemlos, da dieser Lesezugriff auf Views des Users „SYS“ hat.

Der direkte Zugriff auf die Logfiles der Datenbank und den Listener gestaltet sich aufgrund des fehlenden Zugriffs auf den OS-Layer schwieriger. Um Fehlern auf die Spur zu kommen, sind die Oracle-Views „alertlog“ und „listenerlog“ zu verwenden. Diese beinhalten aber seit 11g alle Daten, die in den Logfiles vorhanden sind, und bieten somit einen guten Anhaltspunkt bei der Suche nach Hinweisen, zumal via SQL nach dem Fehlertext gesucht werden kann. Für eine detaillierte Session-Analyse per Tracing gibt es das passende PL/SQL-Package „rdsadmin.manage\_tracefiles“, das die Daten aus den ausgewählten Tracefiles in der Datenbank zur Verfügung stellt (siehe Listing 3).

Dass nicht auf den Server zugegriffen werden kann, ist für moderne Datentransfer-Methoden wie Oracle Data Pump ein Stolperstein. Für Logging und Tracing ist der Zugriff durch Packages und Tabellen in den meisten Fällen auch ausreichend. Aber für einen Datenimport/-export muss das alte, clientbasierte Tool „export“ erhalten. Das ist nicht ganz zeitgemäß, wenn man bedenkt, dass im Hintergrund eine 11.2.0.2-Datenbank arbeitet.

Einer der beliebtesten Kurse bei der Trivadis AG ist der dreitägige Einsteigerkurs für SQL. Die Teilnehmer arbeiten auf der gleichen Datenbank mit verschiedenen Benutzern und legen Tabellen, Views etc. an. Für diesen Kurs wurden eine Amazon-RDS-Datenbank der kleinsten Leistungsklasse erstellt (Small DB Instance – Standard Edition One – License Included), die Kurs- und Demotabellen angelegt und das zentrale TNSNAMES.ORA der eigenen Schulungsumgebung angepasst. Dabei hat sich gezeigt, dass die Datenbank in

der Wolke stabil und zuverlässig läuft. Die vielen Kursaufgaben konnten von den Kursteilnehmern performant erledigt werden; selbst als der Referent parallel zu den Teilnehmerübungen eine Demotabelle mit vier Millionen Einträgen erstellte, war die Performance nicht beeinträchtigt.

### Fazit

Das License-Included-Modell ist vor allem für Schulungs- und Test-Datenbanken ein attraktives Angebot, die Kosten sind klar kalkulierbar. Die verschiedenen Leistungsklassen sind breit gefächert und sollten die meisten Bedürfnisse abdecken können. Der Zugriff auf den Service via Web-Konsole oder CLI funktioniert flüssig und ist ausführlich dokumentiert. Der Zugriff kann mit Firewall-Settings geregelt werden.

RDS läuft stabil und ist einfach zu konfigurieren, der Praxistest zeigte keinerlei Probleme. Bei den Features und Komponenten hat Amazon noch ein paar Hausaufgaben zu machen; keine Applikation kommt heute nur noch mit der Oracle-Text-Komponente aus. Auch dass ein PITR nur in einer neuen Datenbank erstellt werden kann, ist keine Lösung. Noch hat RDS offiziell den Beta-Status, aber bereits jetzt ist die eigene Wolke für den DBA ein durchaus gangbarer Weg.

### Weitere Informationen

- [1] RDS Einstieg: <http://aws.amazon.com/de/rds>
- [2] Preise: <http://aws.amazon.com/de/rds/pricing>
- [3] FAQ: <http://aws.amazon.com/de/rds/faqs>
- [4] AWS Forum: <https://forums.aws.amazon.com>
- [5] RDSCLI: <http://aws.amazon.com/developmenttools/2928>

Martin Berger  
Trivadis AG  
[martin.berger@trivadis.com](mailto:martin.berger@trivadis.com)



```
SQL> select * from tracefile_table;
TEXT
-----
Instance name: TTCRDS01
Redo thread mounted by this instance: 1
Oracle process number: 27
Unix process pid: 30260, image: oracle@ip-10-58-101-97
*** 2011-10-07 14:01:25.850
*** SESSION ID:(38.1815) 2011-10-07 14:01:25.850
```

Listing 3