

# 5000 User +, Erfahrungen im Produktivbetrieb einer SaaS-DB bei Amazon

**Bernhard Vogel**  
**esentri consulting GmbH**  
**Ettlingen**

## Schlüsselworte

Amazon Cloud, RDS, SaaS, HAProxy, Open Source

## Einleitung

Der Betrieb von Software as a Service (SaaS) Anwendungen stellt die Infrastruktur-Verantwortlichen vor besondere Herausforderungen. Neben 24/7 Verfügbarkeit gilt es vor allem auch kostengünstige und zuverlässige Lösungen für die Performanceoptimierung, Skalierung und Sicherheit zu finden.

Amazon ist einer der weltweit größten Cloud Anbieter, bei dem sich auch Oracle Datenbanken betreiben lassen. Der Vortrag geht auf die Erfahrungen vom Setup bis hin zum Praxisbetrieb einer Produktivdatenbank für eine SaaS-Lösung ein und widmet sich den Herausforderungen und zu lösenden Problemen. Neben der Lizenzierungsthematik wird auch auf die kritischen Themen Datensicherheit und Backup eingegangen.

## Die richtige Plattform finden

Um für die eigene Anwendung die richtige Hosting Plattform zu finden sind deren Anforderungen genau zu definieren. Welche Konfiguration werden minimal zum Betrieb benötigt, welche wären optimal und welche Kompromisse lassen sich eingehen, um bestimmten Anforderungen gerecht zu werden.

Im konkreten Fall gilt es für eine Konfiguration einer 3-Tier-Lösung, konkret einer modernen Software as a Service Anwendung, die optimale Plattform zu finden. Im Einzelnen sind lassen sich die Anforderungen an die verschiedenen Schichten wie folgt stichpunktartig beschreiben:

1. Frontend mit High-Availability-Proxy: hohe Verfügbarkeit im Netzwerk, Ausfallsicherheit, Failover Konfiguration eher geringere Last, statisches Backup nach Konfigurationsänderung ausreichend
2. Middleware mit Jetty: hohe Verfügbarkeit, Auslegung auf Hohe Last, einfache Skalierbarkeit, nach Konfigurationsänderung ausreichend
3. Datenbank: hohe Verfügbarkeit, Ausfallsicherheit, Auslegung auf hohe Last, einfache Skalierbarkeit, permanentes Backup oder Replication Betrieb

Außerdem soll die Hardwarebasis möglichst flexibel gestaltbar sein, um schnelles Wachstum und Zeiträume mit hoher Last schnell bedienen zu können.

Aus allen möglichen Hostingoptionen die für die eigene Anwendung richtige Infrastruktur zu finden ist damit nicht trivial und im folgenden wird daher der Weg zur Entscheidung für die Amazon Cloud aufgezeigt.

### **Erste Alternative: eigenes Hosting**

Generell hierzu notwendig sind die eigene Hardwarebeschaffung und eine langfristige Planung. Je niedriger die beschaffte Hardware ausgelastet ist, umso ineffektiver ist deren Einsatz.

Die Netzwerkanbindung muss ausreichend dimensioniert sein. Bei einer SaaS Lösung ist eine hohe Kapazität notwendig, die teuer in der Beschaffung ist. Eine Redundanz bei Ausfall ist nur sehr aufwendig planbar und für kleine und mittlere Unternehmen nur bedingt sinnvoll. Ein Vorteil ist jedoch, dass die Datensicherheit hier vollumfänglich selbst kontrolliert werden. Durch Zugriff auf die eigene Hardware kann eine Backupstrategie frei gewählt werden.

### **Zweite Alternative: Root Server Hosting**

Die Anmietung von klassischen Rootservern oder Virtual Private Server ist meist an eine längerfristige Bindung an die Hardware beziehungsweise der vordefinierten Ressourcen gekoppelt. Eine Skalierung nach oben ist typischerweise nicht kurzfristig möglich, nach unten ist häufig mit kostenpflichtigem Vertragswechsel gekoppelt.

Dahingegen ist die Netzwerkanbindung zumindest bei größeren Providern gut und redundant ausgelegt.

Die Verschlüsselung von kritischen Daten ist hier wichtig, da Unbefugte über möglichen Hardwarezugriff vollen Zugang auf die eigenen Daten erlangen können. In Summe sind die Sicherheitsvorkehrungen in vielen Rechenzentren allerdings als besser einzustufen, als diese im eigenen Unternehmen oft gewährleistet werden könnte. Backups können per Netzwerk übertragen werden und sind je nach Datenmenge über die eigene Infrastruktur realisierbar.

### **Cloud Server Hosting hat viele Vorteile**

Es gibt keinen einheitlichen Standards und auch keine gemeinsame Definition was Cloud Computing ist. Daher ist ein Anbietervergleich immer individuell.

Generell ist die flexible Skalierung hier vorteilhaft. Angemietete Ressourcen können einfach und schnell erweitert werden, umso kritischer müssen hier jedoch die Themen Ausfallsicherheit und Datensicherheit betrachtet werden.

Eine Entscheidung pro Amazon wird durch viele verschiedene Faktoren vereinfacht und es können natürlich die generellen Vorteile von Cloud Computing ausgespielt werden. Ein wichtiger Punkt ist, dass Verfügbarkeitszonen überall auf der Welt angeboten werden. Dieser Punkt hat jedoch auch juristische Relevanz, wenn es beispielsweise um personenbezogene Datenverarbeitung geht. Amazon bietet hierfür auch einen Standort innerhalb der EU an.

Die Angebotene API zur Cloud Verwaltung ist relativ umfangreich und intuitiv. Zu einem schnellen Einstieg werden sowohl vorkonfigurierte Instanzen für verschiedene Betriebssysteme als auch für spezielle Datenbankkonfigurationen angeboten, darunter auch Oracle Datenbanken. Die seit einiger Zeit angebotenen RDS Datenbank Instanzen ergänzen die Auswahlmöglichkeit von Amazon Services zusätzlich. Bei diesem Angebot wird neben dem Betrieb auch die Wartung und das Backup der Daten übernommen.

Für ein Cloudbasiertes Datenbank Hosting muss zunächst die grundlegende Entscheidung getroffen werden, ob man eigenes Datenbank Server Image in der Cloud betreiben möchte oder auf die bereits angesprochenen Services aufsetzt. Neben der Hardware-Skalierung behält man damit aber auch alle Nachteile einer selbst gemanagten Umgebung, wie man diese beispielsweise mit einem Rootserver hat. Allerdings behält man die Kontrolle über die Parametrisierung des eigenen Systems und kann auch im Feintuning der Datenbank eingreifen. Je nachdem welche Ressourcen hierfür bei Amazon

gemietet werden, sind hier vor allem lizenzrechtliche Fragen relevant (die DOAG hatte ausführlich über die Lizenzierungsproblematik berichtet). Typische Fragen sind z.B.: darf ich mit meiner vorhandenen Lizenz die doppelte Anzahl virtueller CPUs überhaupt nutzen?

Mit einer RDS (Relation Database Service) Instanz hat man zumindest in der "Standard Edition One" Version die Möglichkeit die Lizenzkosten im Stundensatz der Instanznutzung mit inkludiert zu erhalten. Dabei gibt es folgende unterschiedliche Optionen bei Amazon:

- Standard Edition One - bring-your-own license / license included
- Standard Edition - bring-your-own license
- Enterprise Edition - bring-your-own license

Nachteil aller Lösungen mit Amazon RDS sind der fehlende SYS Zugang und die Festlegung auf vordefinierte Standardkonfigurationen. Eine Individuelle Anpassung ist hier nicht möglich, was natürlich in gewissen Anwendungsfällen einen enormen Nachteil mit sich bringt.

Die Nutzung der RDS Datenbank bietet sich jedoch besonders an, wenn der Umfang der benötigten Ressourcen stark schwankt, man aufgrund gegebener Volatilität sich nicht langfristig binden möchte und eine Konfiguration grundsätzlich ausreicht.

**Launch DB Instance Wizard** Cancel

ENGINE SELECTION **DB INSTANCE DETAILS** ADDITIONAL CONFIGURATION MANAGEMENT OPTIONS REVIEW

To get started, choose a DB Instance engine and class below

**Engine:** oracle-se1

**License Model:** bring-your-own-license

**DB Engine Version:** 11.2.0.2.v3 (default)

**DB Instance Class:** db.m2.xlarge

**Multi-AZ Deployment:** No

**Auto Minor Version Upgrade:**  Yes  No

---

Provide the details for your RDS Database Instance.

**Allocated Storage:\*** 25 GB (Minimum: 10 GB, Maximum 1024 GB)

**DB Instance Identifier:\*** application1 (e.g. mydbinstance)

**Master User Name:\*** companyadmin (e.g. awsuser)

**Master User Password:\*** ..... (e.g. mypassword)

[< Back](#) [Continue >](#)

Abbildung 1: Konfiguration einer Amazon RDS Instanz

Für die Bereitstellung unserer dynamischen, sich weiterentwickelnden und wachsenden SaaS-Plattform überwogen in Summe die Vorteile, vor allem getrieben durch die hohe Skalierbarkeit (Ziel: 5000 User plus!) der Anwendung und den kostengünstigen Betrieb bei Amazon. Nichts desto trotz soll zum Ende auch auf das Thema Daten- und Ausfallsicherheit näher eingegangen werden.

## **Everything fails all the time**

Zunächst einmal ist die Verschlüsselung kritischer Daten als sehr wichtig anzusehen. Hierfür gibt es verschiedene Möglichkeiten. Bei einer Instanz mit Datenbank oder eigenem System kann z.B. der Tablespace verschlüsselt werden. In einer RDS Instanz bleibt hingegen nur die Möglichkeit die Daten selbst zu verschlüsseln - die Kommunikation zwischen Middleware und RDS ist generell leider nicht verschlüsselbar, bei korrekter Konfiguration sollten aber alle Datenpakete sicher den richtigen Empfänger im internen virtuellen Netzwerk erreichen.

Die beste Verschlüsselung ist jedoch nutzlos bei Verlust der Daten, daher hat das Thema Datenverfügbarkeit oberste Priorität. Optimal wäre hier der Einsatz einer Read Replica oder eines Multi-AZ-Deployments, um eine echte Redundanz mit Failoverfunktion nutzen zu können. Für RDS auf Basis MySQL bietet dies Amazon schon an, für Oracle RDS sollen diese Features noch angeboten werden. Aktuell verbleibt daher nur die Option selbst für Redundanz der Daten z.B. auf einer zweiten Instanz, Sorge zu tragen und Backups per Datapump oder alternativen Techniken zu erstellen.

Dass eine gute Backup-Strategie mehr als notwendig ist, zeigte ein Ereignis im Juli 2011 sehr anschaulich, als im Amazon Rechenzentrum die Verfügbarkeitszone EU-West-1a von Amazon einem größeren Ausfall betroffen war. Man sollte sich demnach nicht blind auf den Anbieter verlassen, sondern frühzeitig selbst zusätzliche Konzepte für Backup & Recovery entwickeln.

### **Zusammenfassung**

Amazon bietet mit Oracle RDS eine sehr attraktive und skalierbare Datenbanklösung an. Insbesondere die Möglichkeit zu wachsen, schnell Ressourcen hinzuzubuchen oder bei fehlender Auslastung Kosten zu sparen, sind große Vorteile, die für ein Cloud basiertes Hosting sprechen. Damit am Ende des Tages jedoch auch die Verfügbarkeit und Datensicherheit gewährleistet wird, empfiehlt es sich regelmäßige Backups zu fahren und sich für alle Eventualitäten vorzubereiten. Diese werden im Vortrag konkret an der Live-Instanz vorgestellt.

Kontaktadresse:

Bernhard Vogel  
esentri consulting GmbH  
Pforzheimer Straße 132  
D-76275 Ettlingen

Telefon: +49 (0) 7243-354 90 0  
Fax: +49 (0) 7243-354 90 99  
E-Mail: [bernhard.vogel@esentri.com](mailto:bernhard.vogel@esentri.com)  
Internet: [www.esentri.com](http://www.esentri.com)