



Running Databases in the Cloud

January 2011

Running Databases in the Cloud

While there has been a recent uprising of new data store models, like NoSQL, there remains a functional need for traditional relational databases. Specifically, with the prevalence of SaaS and SOA architectures, and the rise of social networking and mobile applications, the need for a scalable database has never been greater. Unfortunately, the pain points and costs of running relational databases remains the same. Purchasing the hardware and software to run and scale a database and hiring skilled database administrators (DBAs) can be costly. For many companies, the basic administration and management associated with their backend data store is not a core competency and is oftentimes a deterrent from higher-value application development. Common database management challenges include setting up high availability/durability architectures, managing backup and restore operations, responding to increased load on the database server or a rapidly expanding data set. All of these issues are critical to ensuring an optimal customer experience, but not core to the company's mission. Fortunately, cloud computing is transforming the experience of deploying, managing and scaling a traditional relational database.

In late 2009, Amazon Web Services introduced a new option to address this need with Amazon Relational Database Service (Amazon RDS). Amazon RDS is a managed database “in the cloud”, designed for developers and businesses that require the full features of a relational database, but wish to offload time-consuming database administration tasks. Amazon RDS automatically patches the database software and conducts nightly backups, storing the backups for a user-defined retention period and enabling point-in-time recovery. Amazon RDS provides the flexibility of being able to scale the compute resources or storage capacity associated with a relational database instance via a few clicks or a single API call. In addition, Amazon RDS makes it easy to use replication to enhance availability and reliability for production databases and to scale out beyond the capacity of a single database deployment for read-heavy database workloads. Amazon RDS is billed on a per-use basis, you simply pay for the compute hours needed for your database instances, storage, and bandwidth.

Extending cloud benefits to the database

Customers have made clear that one of the great benefits of running in the cloud is the agility it affords. With no commitment and in a matter of minutes, you can launch an Amazon RDS Database (DB) Instance, with the database software already up and running and pre-configured for your use. There is no need to provision the hardware, install the proper software version, configure the storage layer, and so on. All of this infrastructure management is completed on the user's behalf in a matter of minutes. Decide you're done using the database for the day? Take a “snapshot” of your database data and delete the DB Instance, stopping the clock on your hourly usage charges. If when you resume development work you decide you need more RAM or compute power, simply restore that snapshot to a new, larger DB Instance class.

Many of our customers take advantage of this scaling capability in real-time, scaling up their DB Instance class at times of peak load and scaling it back down during lower stress cycles. Think of a streaming video service that has peak usage between 7pm Eastern and 11pm Pacific time. Such a service might consult its database backend to serve up content recommendations to its subscribers, with responses required in milliseconds. In this case, the company can scale up its DB Instances for this seven hour period to meet client demands, but scale back down afterwards to reduce total hourly expenses. Similarly, many RDS customers will use our “Read Replica” feature to create a running “replica” or copy of their master database for periodic business reporting. Once they've run these ad-hoc queries against the Read Replica (keeping the strain off of their master database), they simply shut it back down, incurring no further expense.

With sometimes tight budgets and limited staff, our customers are always looking for opportunities to reduce the administrative burden of their IT operations. We've been told that the automation built into Amazon RDS is a much appreciated time-saver. For example, by default, each DB Instance is backed up nightly with “change logs” backed up in

real-time throughout the proceeding day. In addition, if a critical security or performance patch is released for the underlying database software, Amazon RDS will automatically apply the patch during the customer-defined maintenance window.

In addition to these helpful features, Amazon RDS affords customers peace of mind knowing that their databases are running on Amazon's proven network infrastructure and datacenters, leveraging the same industry-leading practices and procedures that keep the Amazon.com's global web properties running optimally. Performance and operational excellence are our top priority so we've also added features like "Multi-AZ deployments" wherein Amazon RDS transparently maintains a synchronous copy of a DB Instance to which it will automatically fail over in the event of an unplanned outage on the master database. This feature is easily the most highly adopted option for Amazon RDS, with customers who run production applications insisting on this added durability and availability.

Cloud databases in action

Since launch, companies of all sizes and industries across the globe have chosen Amazon RDS to power their applications and workloads. The Washington Post, for example, uses Amazon RDS for backend data storage of various web properties, including web widgets. Sony Computer Entertainment uses Amazon RDS to store real-time and historical player information for online PlayStation™ games. Gumi, a Japanese online social gaming company, relies on Amazon RDS to enable over 10 million users to play its online gaming platform daily. Internal teams within Amazon.com have also adopted the service, like the Customer Experience Analytics group, which uses Amazon RDS to store and query web traffic simulation data of the 98 million active Amazon.com customers.

Another customer example is redBus, an Indian travel agency that specializes in bus travel by selling over 1.8 million bus tickets annually across 2,500 cities in India. Tickets are purchased through the company's website (<http://www.redbus.in/>) or through the Web services of its agents and partners. The company previously ran its operations from a traditional data center by purchasing and renting its systems and infrastructure. The company has since migrated to the Amazon cloud, incorporating many of the AWS products into its solution, including Amazon RDS. According to Chief Technology Officer, Charan Padmaraju, "By scaling up and down dynamically based on the load, we maintain performance as well as minimize cost. With the time savings that the IT and development staffs obtain from the AWS solution, AWS gives us an overall cost benefit of about 30-40%." Padmaraju also emphasized that the ability to "instantly replicate the whole setup on demand for testing by creating and destroying instances on demand for experimentation," is a huge benefit in reducing the time to market and thereby improving profitability.

References and Further Reading

To learn more about the Amazon Relational Database Service, including technical details, pricing options, how to get started and more, visit:

1. Amazon RDS technical details and pricing: <http://aws.amazon.com/rds>
2. Amazon RDS Getting Started Video: <http://s3.amazonaws.com/awsVideos/GettingStartedWithAmazonRDS/Getting%20Started%20With%20Amazon%20RDS.html>
3. DB Instance Sizing Guide <http://aws.amazon.com/articles/Amazon-RDS/2936>
4. Data Import Guide <http://aws.amazon.com/articles/Amazon-RDS/2933>
5. Amazon RDS Tech Tips Series: <https://forums.aws.amazon.com/ann.jspa?annID=813>