AWS Case Study: Z2

About Z2

Z2's mission is to develop games that are fun, immersive, and deliver daily delight to players around the world. The company creates multiplayer, free-to-play mobile games with rich graphics and storylines designed to appeal to global audiences. Z2 games run on Amazon, Android and Apple systems and allow players to compete and collaborate in real time. Games include a community layer that features in-game email, text and voice chat, player icons, achievements, badges and a social graph integrated with Facebook and Twitter. Founded in 2009, Z2 is based in Seattle, Washington and has a staff of more than 130 employees.

The Challenge

Z2 estimates that more than 55 million people worldwide have played its games on smartphone and tablet devices and generated roughly 3 billion gameplay sessions. Games often include a robust cast of humorous characters. To attract and keep players, Z2 needs speed and performance to support them as they interact with each other worldwide in real-time. Additionally, Z2 needs to be able to mine data collected during gameplay to improve customer support and drive feature development.

Why Amazon Web Services

Z2 has used Amazon Web Services (AWS) since the company was still in the development stage. ""If I want to launch a game in a regular datacenter, I have to buy new servers for a new game. There are many complexities in building out a physical environment, including power, network connectivity and disk space," says Director Live Operations, Markus Schweig. "Working in the cloud is faster and more agile. We can launch instances quickly and experiment with them — all at a better price and performance using AWS."

Using AWS for an immersive game experience

As shown in Figure 1 below, MetalStorm: Aces is a real-time, multiplayer air combat game featuring a fleet of modern jets, 3D graphics, exotic environments and enemy ace pilots.
Imagine that a player in Des Moines, Iowa starts a session of MetalStorm on a mobile device. The client-side application runs on the player's device and supplies the artwork and code to run a game. The device connects to Z2's infrastructure platform on AWS where game servers run the game logic. Load balancing distributes traffic to the application servers communicating with multiple services and databases.

Figure 1. MetalStorm: Aces, a Z2 Multi-Player Game

Once the device connects to the game servers, the player asks for a match. The game servers identify a match (based on the player's statistics) from Singapore. "Because AWS is available in so many countries, players have the ability to play real time latency dependent games with anyone in the world," Schweig comments. "We can place packet relay servers as close to players as possible. It allows for low latency compared to using centralized server hosting. We're able to provide location update information between different devices at a pretty rapid speed."

As the game begins, players can see each other's planes and launch attacks. Z2's engineers wrote custom code on Linux running on Amazon Elastic Compute Cloud (Amazon EC2) instances. The code relays information so players know the locations of the planes and the actions of each player. Additional code captures scores and other information and writes it to a NoSQL leaderboard database. The database captures games statistics and records when someone wins a game. After a game is over, players can retrieve the information and review statistics for the game and their overall game standings.

Z2 runs approximately 120 Amazon EC2 instances of all types across eight AWS Regions. "We experimented heavily to determine what code should run on which instance types," says Schweig. "We've optimized instances to launch faster by using our own customized operating system images and we developed a DevOps automation framework to increase our agility."

Z2 moved to bigger instances on game servers, allowing the company to carry more players on an instance. "Swapping out different instance types or leveraging an instance and its characteristics helps us reduce and optimize AWS costs over time," Schweig comments. For its game servers, the company moved from standard large instances to high compute medium instances and eventually gained its best benefits from high compute extra large instances. For its core databases, the company uses instances with SSDs and high network throughput. Z2 strives to exceed an 80 percent ratio of Amazon EC2 Reserved Instances, which yields greater than 50 percent cost savings over running on demand.

The game company uses Amazon CloudFront in 43 edge locations to deliver content updates over the air to players. Amazon Simple Storage Service (Amazon S3) acts as an origin store for pushing content to CloudFront. It also stores snapshots (backup copies) of data volumes. Schweig explains, "In the event of a disaster, we can recover data from Amazon S3 and continue out of different Regions."

Z2 also uses Amazon Relational Database Service (Amazon RDS), Amazon Route 53, AWS Identity and Access Management (IAM), and Elastic Load Balancing in its architecture. Figure 2 demonstrates the Z2 game platform on AWS.

Figure 2. Z2 Architecture on AWS

Using Amazon Redshift for big data analysis

Z2 has its own custom software and meters to monitor operational data in real time and gauge the game environment. "I need to know that databases and code are functioning correctly and that customers are having a good time," says Schweig.

To capture data for business analytics, the company uses Amazon Redshift as its data warehouse to store hundreds of TB of data. Z2 uses an extract, transform and load (ETL) pipeline and Amazon Simple Queue Service (Amazon SQS) to process event data from Amazon EC2 instances and store the data in Amazon Redshift.

Data is loaded from Amazon Redshift to a front-end application. Z2's product teams analyze the data to identify areas for improvement, to compare one week to another, or to review the performance of a new feature. Figure 3 demonstrates the analytics pipeline on AWS.

Figure 3. Z2 Analytics Pipeline on AWS

The Benefits

Running on the AWS Cloud lets Z2 focus on making sure that its code is robust and that the architecture meets business needs without worrying about the number of servers that it needs. Z2 can scale quickly when new features, promotions on the Apple store, and other time-based events cause a rapid influx of players. "We don't need to know what is going to happen tomorrow because we can quickly scale as needed," says Schweig. "Thanks to AWS, we can accommodate spikes in traffic."

By using Amazon Redshift, Z2's engineers can capture information and provide timely information to the business. "We could do this before, but using Amazon Redshift in combination with a newly designed pipeline allowed us to reduce our Analytics related AWS hosting cost significantly," says Schweig.

"Our games draw players in and keep their attention because we're able to successfully blend a rich, compelling experience with innovative gameplay, mechanics, features, and a superior level of customer support," says Schweig. "We're using AWS to grow our roadmap. We look forward to additional innovations that raise the bar for innovative gaming experiences we can deliver to our players."

Next Step

To learn more about how AWS can help you run your game applications, visit our Game Hosting details page: http://aws.amazon.com/game-hosting.