



## Meeting the scalability and performance challenges of online gaming

### Introduction

Online gaming continues to grow in popularity, whether for huge gaming communities like Riot Games' *League of Legends* or gaming sites like bet365, one of the world's leading online gambling groups. This growth is forcing changes to existing infrastructure in order to keep up with demand and innovation. Traditional relational databases can't meet the requirements for massive scalability, speed, and fault tolerance. Plus, the costs of these traditional systems are breaking the bank. Let's look at some of the challenges you may be facing with your online gaming and betting platforms.

### The Online Gaming Challenge

Over the last decade, the massive adoption of mobile devices and the arrival of ubiquitous connectivity have moved online gaming and betting from a traditional brick and mortar activity to one where people can place bets or play games online anytime, anywhere.

This change has brought huge numbers of new players to online gaming and betting as a form of entertainment. More players, fast development cycles, and the ability to quickly deploy in the cloud have brought new competitors to this fast-growing industry.

Innovation is critical to retain long-term customer loyalty and is changing the way gamers play online. These changes include the move away from single bets on an event to in-game betting on an ever-increasing range of metrics. The advent of regional gaming competitions, like the League of Legends World Championship with an annual grand prize of \$1 million, show just how far gaming has come.

Your IT Infrastructure is being challenged to keep up. Gaming sites have millions of simultaneous players. Betting and gaming applications have to capture and process statistics in real time about many different game variables for each player both before and during the event. In betting, the odds constantly change and players are looking for opportunities to place micro-bets during games, settle those bets, and reuse the proceeds for new opportunities. This means the frequency at which odds must be calculated is extremely high. But those odds must also be reviewed against the marketplace to ensure players bet with you instead of your competitor.

### Why Riak Enterprise for Gaming?

- Simplified Development
- Enhanced Scalability
- High Availability
- Fault Tolerance
- Simplified Operations
- Lower Total Cost of Ownership

Your IT infrastructure is also challenged by the need for agility. Gaming and betting sites must be able to handle both predictable and random spikes in demand. In both cases, maintaining infrastructure that sits idle 90% of the time waiting for a spike is a waste of resources. In addition, a poor user experience will quickly lose customers, and downtime is catastrophic. The database infrastructure is critical to every step of the gaming and betting process and ensuring performance is maintained at scale.

Scaling and performance are only some aspects of infrastructure agility. Your systems must also detect and respond to fraud as well as be adaptable to adhere to complex and detailed regulatory changes.

## Gaming on Riak

Riak is a distributed database designed to address a number of challenges of gaming and betting platforms. Riak delivers performance, scale, and constant uptime at a lower cost than traditional relational databases.

### Managing Huge Volume of Players and Data

Gaming and betting applications track hundreds, even thousands of variables. Calculating odds and managing real-time account and transaction data means that billions of data points and terabytes of data are being acted upon in an online gaming and betting system. Relational Databases are not designed to work at this scale. Riak is storing and retrieving data for some of the world's largest gaming and betting platforms and it scales in a near linear fashion to give you the best possible performance on commodity hardware.

### Agility — Scaling Up and Down

Riak is optimized to use commodity hardware resources efficiently. It delivers results that outperform traditional relational databases that are often running on some of the most expensive hardware systems available. When your peak periods occur, Riak scales easily and quickly with zero downtime, lowering the chances of performance issues for your gamers. Because Riak is designed for commodity hardware, you can expand or reduce the database infrastructure to meet your changing business. Riak's distributed design allows it to easily grow, as needed, both within a single data center or across multiple data centers.

### Maintaining High Availability

Hardware does fail, and when it does your IT Infrastructure needs to be able to cope and your systems must continue to operate while getting the resources back online as soon as possible. Online gaming and betting solutions demand high availability and Riak is designed to meet those requirements. When data is written to Riak, it is automatically replicated throughout the cluster. Even if nodes fail, data is still available for read/write access and the system stays online ensuring your players are not interrupted and your business doesn't stop.

### Processing Speed

Whether calculating odds, providing content to mobile apps, or settling bets, performance is key to you and your customers. A necessity of micro-betting is that in-game variables, and therefore odds, are constantly changing. Riak is designed to operate in this environment and outperforms RDBMS in terms of throughput all while running on commoditized hardware. The way in which data is modeled in Riak, along with its use of simple data types, makes it extremely fast compared to an RDBMS. Riak is the perfect choice for the fast-paced, big-data challenges of online gaming and betting.

### Costs of Malicious Activity

Dealing with malicious betting as quickly as possible is critical to availability and profitability. The speed and volume of bets being placed online makes this harder, particularly as criminals turn to automation to try and beat the system. Riak outperforms RDBMS not just in terms of processing power, but its hardware and licensing costs are considerably lower, thus cutting the operational costs of detecting and fighting fraud. The simplicity and agility of Riak ensures that changes can be quickly made and deployed across the infrastructure to help mitigate evolving threats.

### Regulatory Changes

Implementing regulatory changes, whether specific to a single country or industry-wide, is complex and time consuming. Riak's schema-less design and simplicity allows rapid development and accelerated implementation of regulatory change (versus RDBMS), which decreases go-to-market costs and creates a competitive advantage.

## Key Riak Use Cases



#### PLAYER DATA

Riak provides low-latency, highly available data storage for key player data, including user and profile information, game performance, statistics and rankings, and more. Riak also provides many different tools for querying and indexing this data, such as a full-text search engine and secondary indexing.



#### SESSION STORAGE

Riak is frequently used to store and serve session data with predictable low-latency, which is necessary for game play. Riak imposes no restrictions on the type of content stored (since all objects are stored on disk as binaries), so session data can be encoded in many ways and can evolve without administrative changes to schemas.



#### GLOBAL DATA LOCALITY

While gaming, players require a low-latency experience, regardless of their physical location. Interrupted or slow game play can lead to poor user experience and player abandonment. Riak Enterprise's multi-datacenter capabilities allow game data to be physically close to players and for fast response times regardless of player location.



#### SOCIAL INFORMATION

Riak is built for very fast data storage. Due to its inherent design and Riak's simple key/value data model, Riak is ideal for storing and serving social content such as social graph information, player profiles, player relationships, social authentication accounts, and other types of social gaming data.

These are just a few use case examples where moving from relational databases to Riak makes business sense. Riak can bring reliability at scale, data flexibility, and speed to a your gaming platform.

## Conclusion

The hugely competitive online gaming and betting market is fundamentally based on constantly processing huge amounts of data to fuel every step of the gaming and betting process. Customers have high demands and expect a fast user experience. Players need to be able to make and claim bets instantly since lost time can cause a lost bet or worse odds. Your gamers won't wait when the battle is raging.

The database underpins many aspects of the gaming experience with high availability and performance essential to success. The rapidly evolving industry and massive growth in data volumes mean that moving from relational databases to Riak allows you to meet your gaming challenges. Don't take a chance with your database. You have a winner with Riak.



Given the huge amount of data we process on a daily basis—from customer details to betting odds—it was imperative that we had a platform to support this. We selected Riak, and have not been disappointed with the results.

– Martin Davies, Chief Executive Officer, Technology at bet365



## About Basho

Basho is a distributed systems company dedicated to making software that is highly available, fault-tolerant and easy-to-operate at scale. Basho's distributed database, **Riak** and Basho's cloud storage software, **Riak CS**, are used by fast growing Web businesses and by over 25 percent of the Fortune 50 to power their critical Web, mobile and social applications and their public and private cloud platforms.

Riak and Riak CS are available open source. **Riak Enterprise** and Riak CS Enterprise offer enhanced multi-datacenter replication and 24x7 Basho support. For more information, visit [basho.com](http://basho.com). Basho is headquartered in Cambridge, Massachusetts and has offices in London, San Francisco, Tokyo and Washington DC.

