

E-Book

# Increase Player Retention with Real-Time Game Personalization

Part 4 of a 4 Part Gaming Series



# Introduction

We all love personalization. Even if it's just a smiley face the barista draws on the cup at our local coffee shop, it's the personal touches that make something truly special. The same is true for your video game players. Every player out there is looking for a great game, one they can sink themselves into and truly make theirs. It's the holy grail for game developers, one where you have a loyal, actively engaged player base with high retention.

Unfortunately, players are also extremely picky about which games are worth their time, which makes the bar you have to clear to get their loyalty and hit those crucial 7-day and 30-day metrics even higher. That's where personalization comes in.

With personalization, you can customize the game experience to give players that personal touch. It includes a wide range of activities, covering everything from ad serving to gameplay customization to adaptive AI to

character/avatar skins—even community interactions. In short, almost every part of your game is a candidate for personalization.

Creating that personalization is a huge lift requiring a robust, complex backend data structure. It's also necessary in the modern environment. Competing games are working just as hard to add exceptional personalization, and even the smallest hiccup can be enough to send players to another game within those crucial first few weeks.

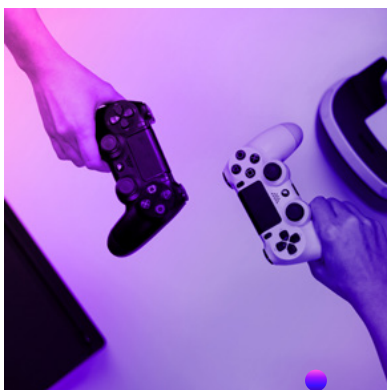
In short, your personalization has to be complex enough to tailor the game perfectly to your players, while the database has to have the performance to make the personalization seamless.

It sounds impossible, but when it's powered by the right database, real-time personalization can be a reality for all of your players—no matter where they log in from.

## What does real time mean?

Research into human response indicates applications have roughly 100 milliseconds (ms)—one third of the time it takes to blink—before users feel like they're waiting for a response. If personalization is going to be seamless, then it needs to happen in real time. That means every request needs to be sent, processed, and responded to in less than 100ms.

# How modern personalization improves the player experience



Personalization comes in thousands of different flavors. While it's not a fully inclusive categorization, for the purpose of data use they can be sorted into player-led personalization and game-led personalization.

## Player-led personalization

Players love having the option to personalize their gaming experience and customize their gameplay. And when they do, they expect games to respond seamlessly, without any lag, loading delays, or dropped responses. Any hiccup or noticeable latency is a potential source of frustration. The system behind these personalizations has to have real-time responsiveness, with each personalization having its own specific use case demands:

- ▶ **Avatar/character personalization** – Whether it's tailoring every aspect of a character's appearance or showing off the latest skin, players love the creation (and bragging rights) of a personalized appearance. But this can also mean rapidly cycling through hundreds, even thousands, of options as players search for just the right look. And for a real-time experience, every last one needs to be pulled from the database and rendered in less than 100ms.
- ▶ **Gameplay customization** – Whether it's switching out the tires on a car, the rifle you use in a FPS, or a puzzle powerup, games thrive on giving players ways to

customize their gameplay. Unlike character personalization, customization has a direct impact on gameplay, such as making cars faster, characters stronger, or puzzles easier. Many customizations even have mods or customizations of their own, allowing players millions of combinations to try out. And you can be sure the players will definitely try them, requiring systems to rapidly load, render graphics, modify stats, and then log the new stats for gameplay.

- ▶ **Interactive leaderboards** – Modern games can easily contain dozens of leaderboards for each player. Racing games might have different leaderboards for each track, fantasy RPGs for arena battles, or puzzle games for each level. Not only are these parsed by variables like leagues, server location, guilds, and more, but many of them have interactive elements that require rapid sorting, search, and so on. Players don't care that the leaderboard can have thousands of entries with multiple data dimensions—they still expect them to load and respond to commands instantly.
- ▶ **Smart matchmaking** – Matchmaking can make or break multiplayer games. When done right, smart matchmaking responds in real time to perfectly match players together, whether it's in battle royales, 1-on-1 matches, or when making teams. Matchmaking is further personalized by giving players choices of opponents, races, modes, and more. And no matter how many times the player hits that "refresh options" button, they still expect the choices to load seamlessly.

These are only a few of the many ways you can personalize your game with player-led interactions. But throughout all of them, there's a single thread: Players don't just demand real-time responsiveness, they expect it. And you have to deliver that responsiveness if you want to keep them engaged and in the game.

## Game-led algorithmic personalization

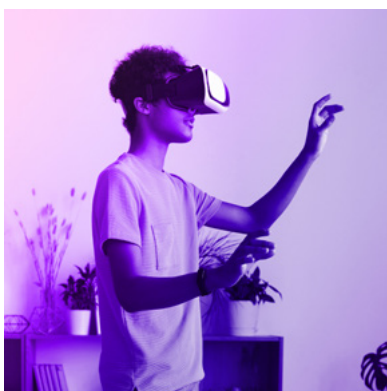
Game-led personalization takes a more proactive approach to improving the player experience. Instead of providing options for players to respond to, game-led personalization analyzes and learns from the player to create a more personalized experience. Rather than seeing the personalization actions, such as when modifying gear and seeing the stats change, players instead see the results of the personalization. Personalizations like these add an extra layer of services, but the end result can have a much more profound effect on the player experience:

- ▶ **Adaptive gameplay** – By using machine learning to create player models, games can personalize their gameplay to fit individual players. This includes anything from adjusting the difficulty of the AI to serving different puzzles types. When done right, players might not even notice the difference. All they notice is that they're engaged and having fun. But doing it right requires the performance and data models to seamlessly analyze player behavior and respond in real time.

- ▶ **Offer serving** – The right offer at the right time not only makes the player more likely to make a purchase, it can also make the player feel like the purchase has more of an impact. For example, a special discount after a player has failed a level multiple times can turn around a negative experience. Losing is frustrating, and odds are good the player might have been about to quit. But when you make the right offer, they'll make the purchase, then feel triumphant when they pass the level. Whether the offer uses real currency or in-game currency, finding that combination of the right offer at the right time requires a flexible, high-performance database.
- ▶ **UI personalization** – Services like searchable/sortable inventories, notifications, or community chats improve the player experience. Each service also builds off personalized player profiles. However, any player engagement gain from these services can be lost if there's too much of a lag or the services are inconsistent. You'll need real-time responsiveness to reap the benefits of these personalizations.

In these and other game-led personalizations, the common thread lies in using backend analysis and data processing to provide real-time personalization services. It's part of the growing standard of **360° player information views**. Rather than break up player data for later batch analysis or piecemeal processing, this approach analyzes a complete view of the player. It's a more data-centric approach that increases personalization and monetization.

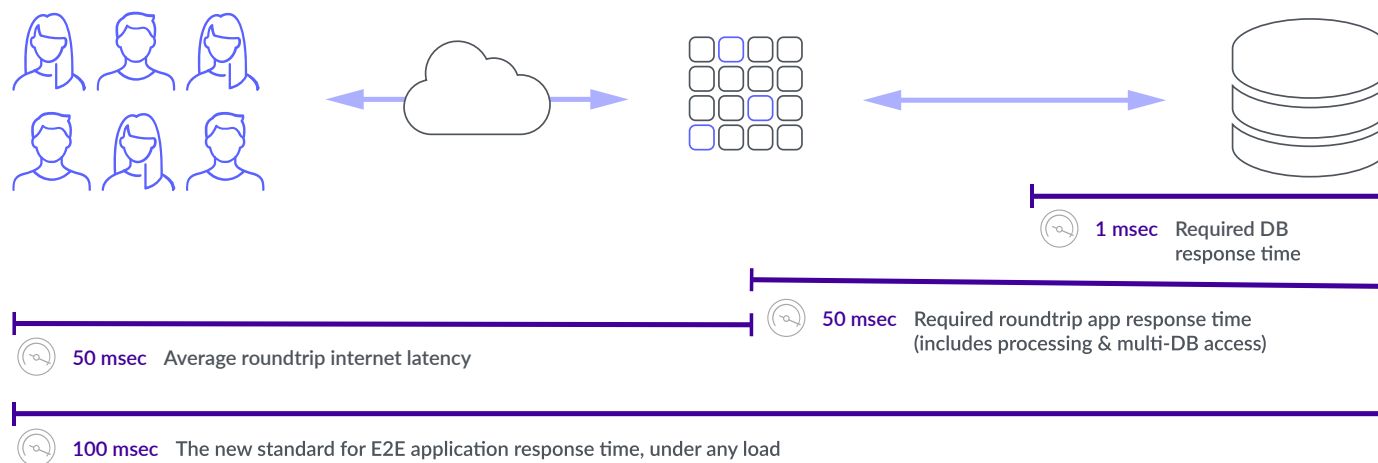
# Technical requirements for a real-time personalization database



Whether it's player-led or game-led, personalization requires processing an incredible amount of data—and the amount of data involved is increasing exponentially. In order to meet the seamless standards required by players, you need both specialized approaches to processing data and a database capable of real-time performance. On top of that, your database needs to be able to reliably serve that personalization anywhere and everywhere your players are. All that translates into specific technical requirements for your database:

- ▶ **High write throughput** – Methods like batch processing create periods of stale data which can hinder your personalization efforts. The latest player data (identity, credentials, inventory items, recent actions, etc.) need to be stored as they happen, which requires a high level of write throughput.
- ▶ **Persistent player stores** – Player data needs to be stored in persistent session stores that can scale with servers so they're always accessible and actionable for personalization—because your players demand the same personalization experience no matter how many people are playing.
- ▶ **Low latency** – A rule of thumb when running applications in production states that every millisecond spent in the database means 100 milliseconds felt at the application level. That means the database needs to be capable of <1ms latency to hit the 100ms goal.
- ▶ **Analytics capabilities** – Every time you have to run data through a separate service, it adds another link in the chain that can fail or introduce latency. For reliability and performance, the database should have integrated analytics capabilities so it can apply smart analytics and surface the insights necessary for personalization.
- ▶ **Flexible deployability** – Personalization needs to happen unilaterally across your game no matter what environment it's deployed in. That means your database also needs to be able to deploy in any environment, whether it's in the cloud, on-premises, or hybrid.
- ▶ **High global availability** – You need to be able to count on your personalization services—and the database behind them. High availability is essential for any personalization database, as well as the ability to provide consistent performance anywhere around the globe.
- ▶ **Scalability** – It doesn't matter if there's thousands of players online or millions. Every player needs to have the same experience, which means your database needs to be able to scale while still maintaining the performance necessary for seamless personalization.

The driving power behind personalization is data. The more data you collect and the faster (and more accurately) you process it, the better your personalization is going to be. And in an age where personalization can be the deciding factor between lasting player loyalty and game deletion, you can't afford to take chances with your personalization database.



## Real-time personalization takes a real-time database

A real-time database is capable of the performance necessary for real-time responsiveness. If you remember from the beginning of the e-book, real time is defined as a round trip of less than 100ms. With 50ms taken up by internet latency, and another 50ms in app response time, that leaves less than 1ms for the database to respond to qualify as a real-time database. Traditional disk-based databases simply aren't capable of latency that low, which is why you'll want to look at in-memory NoSQL databases. Redis Enterprise, for example, is capable of [50 million operations per second at <1ms latency](#).

But speed isn't the only requirement for a real-time database. It needs to be able to maintain that real-time latency at any scale, with high availability across the globe. And it needs to be able to do it while providing the analytics capabilities needed for personalization, such as with Redis modules like [RedisBloom](#). By supporting compact, probabilistic data structures, RedisBloom allows you to create filters that reduce the compute and memory needs to search massive data sets, such as personalization analysis.

When you put all those requirements together, the foundational data platform under your personalization needs to have low latency, 99.999% availability, global scalability, analytics data structures, and the ability to be deployed in any environment.



---

# Conclusion

Personalization is integrated into almost every game out there. The question is, can your personalization be more in-depth and more seamless than in your competitors' games? With the right database, it can. By building real-time personalization into your game, you pave the way for better player experiences, more player retention, higher MAUs, and, ultimately, greater game revenue.

---

Learn more ways a real-time database can level up your game and increase player engagement. Go to [redis.com/gaming](https://redis.com/gaming) or download the *Level Up Your Gametech with a Real-Time Database* white paper.



# About Redis

Data is the lifeline of every business, and Redis helps organizations reimagine how fast they can process, analyze, make predictions, and take action on the data they generate. Redis provides a competitive edge to any business by delivering [open source](#) and [enterprise-grade](#) data platforms to power applications that drive real-time experiences at any scale. Developers rely on Redis to build performance, scalability, reliability, and security into their applications.

Born in the cloud-native era, Redis uniquely enables users to unify data across multi-cloud, hybrid, and global applications to maximize business potential. Learn how Redis can give you this edge at [redis.com](https://redis.com).

