Alibaba Cloud
aliyun.com

The Cloud to Drive New-age Browser Games

Contents

01 Browser Gaming-Overview

Browser gaming comprises computer games played on the Internet using a web browser. In other words, they are games played online. Browser games are typically run using standard web technologies or browser plugins and encompass almost all video game genres, as well as single or multiplayer format. Browser games can be portable and paired with multiple devices, web browsers, and operating systems.

The creation of browser games usually involves standard web technologies as a frontend, and other technologies as a backend. Browser games are played alone with the browser and often require no additional installation or software. Typically, they are written in browser-based languages, including JavaScript, HTML5, and PHP. Also, they may utilize some additional languages (i.e. Flash, Java, and Silverlight) for backend processes and supplementary browser plugins.

Browser games can vary in genre from shooting to racing, and to complex role-playing games with a long-playing life.

1.1 A Lasting Fad-Evolution

Back in the day, gaming was popular amongst young people and involved the shared activity of playing in arcades and competing for high scores. This activity has gradually transitioned into browser games.

Browser games appeal to both ardent and casual players and come in many interesting themes and genres. There are thousands of browser games available now, including new games and reboots of classic games.

For gaming businesses, browser games are the most enduring part of the gaming world because they can be effectively used independently, without an additional client software or monetary expenditure. They have so far managed to capture the attention of a huge number of player communities and kept them engaged over a long period. Browser games are graphically less intensive and consequently have lower system configuration requirements as they are designed to be played on a vast range of systems and devices. Therefore, it is a logical assumption that as the Internet continues to grow, so will browser gaming.

1.2 Advantages

The biggest advantage of browser games is that they are fully and independently playable without any additional software installation. This aspect allows a user to play a browser game anywhere, such as work, school, home, Internet café, etc.

Also, one of the primary reasons for the rising popularity of browser games is the player community. Via the social aspect of the game, competitive spirit and collaboration are encouraged. Joint actions are planned and carried out, players communicate with their friends, and new friends are made as well.

Software-dependent games often require downloadable files in sum of 10GB, 20GB, or even more before you can start playing them. However, cloud backed browser games allow users to instantly start playing on the browser with minimal installation. Browser gaming also offers easy viewing for those who wish to watch, and especially as streaming can be duplicated for different machines at the same time. Many browser games are impossible to pirate, which makes this gaming platform extremely attractive to Digital Rights Management (DRM) publishers as well.

1.3 Challenges

The advancement in browser game technology has been directly proportionate to improvements in browser technology. Browser games that have existed for a long period of time have had to undergo an extreme metamorphosis in terms of design, graphics, and other usability aspects in order to be compatible with browser upgrades.

Browser-based games are typically free and are supported financially through advertisements and subscriptions. This ensures maximum opportunity for player numbers. Players who have a computer with an Internet connection can have access to countless browser games without having to invest in consoles and other specialized equipment.

A standard broadband connection is usually sufficient to handle movie streaming as films exist in a single, unchanging state. What makes games different is that they are always changing dynamics with every user input. Every push of a button is transmitted to the server where the game is being processed (i.e. every time one needs to change the move of the character or switch the camera angle). It is the server's job to process the commands sent by the buttons to determine the moves of the character and stream the results back to the user. Compared to a movie, that's a substantial amount of data transfer made in just a few seconds. This often results in a noticeable delay from the time you press a button until you see it occur on the screen.

02 Browser Cloud Gaming Solutions: Overview

"Gaming on Demand" is an industry buzzword used for cloud gaming, which supports most online gaming whether on PCs, browsers, or handheld devices. The two main types of cloud gaming that exist currently are cloud games based on video streaming and cloud games based on file streaming. The cloud plays a crucial role in providing a seamless experience and direct playability for users of the games across all devices.

The market is getting flooded with a myriad of options for on-demand browser gaming, which is a growing trend in the industry. Browser gaming mostly leverages cloud computing for its backend infrastructure. These games are often executed through private cloud servers and transmitted via the Internet to reach interactive televisions, consoles or desktop PCs. Similarly, user commands are sent to cloud servers that effectively apply the commands to the gaming platform.

Cloud solutions for browser gaming enable enterprises to develop advanced and unique games with a better user experience and response time at low operational and maintenance costs. Transitioning gaming applications to the cloud drives down hardware and upgrade costs thanks to a Pay-As-You-Go or subscription based payment model. The growth of the gaming business and cloud environment also go hand-in-hand. When demand is unpredictable or testing is required for new features, the ability to spin capacity up or down is made easy on the cloud, while the game company only ever pays for what is consumed.

Browser games comprise of two broad categories; **Massively multiplayer online role-playing games (MMORPG)** which has an audience among more serious gamers; and **Casual Browser Games** which include such games as Farmville and Miniclip, which are mainly aimed at people who are looking for options to pass time.

2.1 Browser Cloud Gaming Solution: Main Pitfalls

As the gaming market has expanded, there has been a surge in the number of browser games available. Among the gaming industry, MMORPGs and Casual Browser games have become the most popular subcategories. This has created a need for designers and developers to continue providing the optimal user experience, while constantly innovating at the same time. Designers and developers now face a new challenge to create background architectures that provide gamers the same seamless platform and smooth user experience as experienced on traditional PC games.

1. Game Security

Browser games often include automated gameplay aspects with very little interaction from players. This leaves a space for the use of plugins. Therefore, it becomes imperative to introduce mechanisms that prevent certain players from using plugins and manipulate the in-game benefits that affect the balance of the game's value system. Attacks pose another security risk. Browser games are the most vulnerable to online attacks.

2. Elastic Resource Scalability

Casual social browser games generally run on a social network. Some highly popular games experience a rapid surge of users. MMORPG games also constantly launch new servers to attract new players. In addition, browser games are usually short lived. After a time, players become less active and gradually stop playing. Browser games are often exposed to high and low peak periods, which requires them to leverage server resources that support quick and reliable elastic scaling.

3. Large Offline Interaction Data Read/Write Volume

To enhance user loyalty, Social Network Service (SNS) browser games generally add offline gameplay aspects based on the relationship chain of social networks. This involves reading and writing large amounts of offline interaction data (far exceeding the amount of data to be processed for concurrently online players). Therefore, the games require a data storage solution that supports concurrent reading/writing of mass data.

4. Static Game Content Acceleration

Browser games are non-client games where players must simultaneously play and download. This requires developers to combine download acceleration for some game content with dynamic resource loading policies to improve the gaming experience.

2.2 The Reign of Cloud Gaming Solutions

Considering the adaptability of the cloud to provide tailor-made solutions for all the pain points plaguing the Internet, cloud gaming solutions have given developers freedom to experiment. It allows them to quickly build, deploy, distribute, and monetize their applications. The built-in elasticity and Pay-As-You-Go model of cloud services also make it easy for game providers to scale their service while being cost-effective.

Cloud gaming solutions provide reliable, available and highly scalable infrastructure through powerful and high-memory server clusters. As infrastructure is hosted on the cloud, there is minimal operational overhead for game developers and providers. Developers can then cater to a huge number of gamers to generate large volumes of gaming applications with the help of dedicated gaming clusters that support high concurrency and multi-scenario game deployment. This facilitates the development of new applications as developers have increased bandwidth and flexibility to work with.

Cloud solutions offer a superior geographic distribution network, allowing them to deploy gaming applications from data centers in different regions across the world. At the same time, they ensure data security with layers of firewall protection and eliminate the chance of single-point-of-failure by providing automatic failover.

2.3 MMORPGs Browser Gaming Solution Overview

Massively multiplayer online role-playing games (MMORPGs) constitute a combination of role-playing video games and massively multiplayer online games where a large number of players interact and play together in real-time on a single platform. MMORPGs depend on the concept of emergent gameplay, which is based on interactions of players and groups of players over the game mechanics. Most MMORPGs provide levels in the gameplay with character survival becoming progressively harder. The purpose is to help gamers build up their characters through virtual experience, abilities, and wealth.

Gamers can interact within the game, customize their environment and avatars, make alliances with other characters/ gamers, and experiment with gameplay routes. Also, players that don't want to actively participate in the game can still be a part of the game environment by setting up shops in virtual villages or donning the character of a passerby to add authenticity to the gaming world.

The characters in MMORPG economies use virtual currencies earned by winning battles to buy items that help them progress in the game. There have been instances where the virtual economy has crossed over into a real one. For example, players often exchange real money for virtual items and currency. In some cases, players seek to go a level up by employing other gamers to play their characters while they are logged off.

MMORPG Game Developer Challenges

One of the key characteristics of browser games is that gamers have real-time access at anytime, and anywhere with an Internet connection. The gameplay of browser games is typically divided into traditional MMORPG gaming and casual gaming. Although modern MMORPGs have shown a dramatic evolution and differ extensively from their decedents, most share the same basic characteristics. Common features include level progressions, underlying game environment, social interactions within the game, in-game environment, system architecture, group membership dynamics, and character customization.

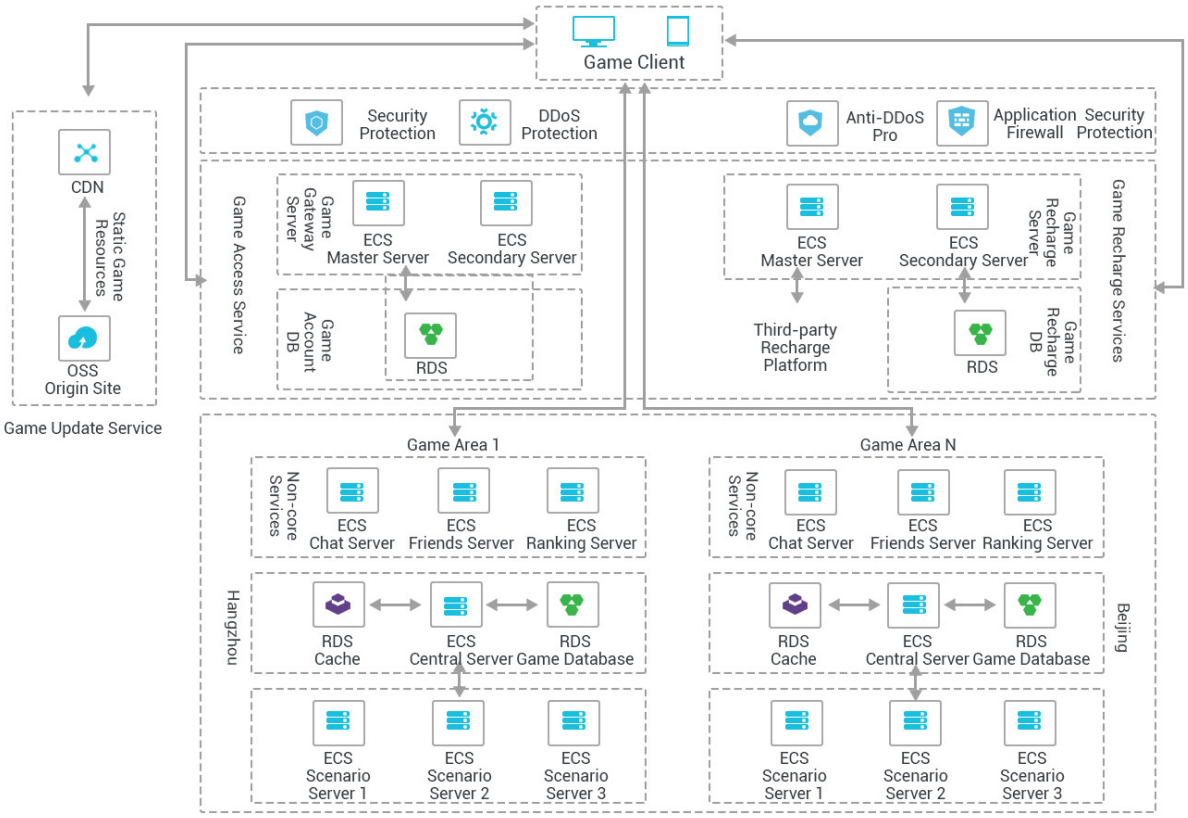
Architecturally, most MMORPGs are deployed using a client–server system architecture. The server software generates a persistent instance of the virtual world that runs uninterruptedly, and players connect to it through client software. The client software authorizes access into the playing world, which may be subject to further "expansions" or "upgrades" that can be purchased during the game.

From a development perspective, an MMORPG server requires thorough expertise with tiered client/server architecture, network protocols, security, and database design. Thus, MMORPGs must include reliable systems to carry out vital tasks. The server needs to handle and verify a huge number of connections, apply changes for it to evolve (bug fixes or added content), and prevent any form of cheating. Also, it is imperative to have a system that records and backs up game data at regular intervals, without causing any hindrance to gameplay.

MMORPGs require an appropriate number of servers and bandwidth, along with dedicated support staff to handle maintenance and upkeep. A lack of sufficient resources can create a lag and may lead to frustration among players and poor game reviews. It is therefore important to carry out suitable provisioning of servers based on increasing or decreasing amounts of traffic. Peer-to-peer set up in MMORPGs may contribute in cheaply and efficiently regulating the server load. However, irregular network bandwidth, render engines hogging CPU capacity, unreliable individual nodes, characteristic lack of security (facilitates cheating), and other issues make for a challenging proposition.
The hosted infrastructure of an effective commercial-grade MMORPG involves deployment of hundreds (or even thousands) of servers. Therefore, development of an affordable infrastructure for an online game needs developers to accommodate huge numbers of players with minimal hardware and network investment. Additionally, developers need to be knowledgeable about the fundamentals of game design, which include the creation of the game world, lore and legends, game mechanics, and the element of gaming intrigue.

Recommended MMORPG Browser Cloud Gaming Solution

The following solution demonstrates the recommended MMORPG cloud gaming solution.



Alibaba Cloud offers comprehensive gaming solutions specially designed for game developers and operators to offer an amazing gaming experience on different devices and platforms, including mobile, browsers and computers. Alibaba Cloud provides a cloud computing platform that addresses the needs and challenges faced in hosting browser games.

1. Static Game Content Acceleration

Developers can use a Content Delivery Network (CDN) and Object Storage Service (OSS) to build an accelerated distribution service for static browser game content such as textures, UIs, audio, sounds and special effects. Alibaba Cloud has hundreds of CDN nodes that perform optimization for image downloads. The seamless integration of OSS and CDN allows developers to achieve automatic detection and delivery of changes to objects stored in OSS. CDN provides API control, allowing users to conveniently preheat resources through the API. OSS can be used to compress, crop, transcode, and process images in batches.

2. Multi-Level Game Security

Alibaba Cloud anti-DDoS defenses are provided through Alibaba Cloud security services' protected IP address and Electric Serial Number (ESN). Alibaba Cloud's WAF product provides security defenses for browser game servers and sites, effectively preventing hackers from using vulnerabilities for infiltration and penetration. Server Guard regularly performs vulnerability scans and analysis on servers. In addition, some anti-plugin policies are added in the game service, such as parameter validation, maximum possible server reproduction of operation results, and verification code mechanism.

3. Static Game Content Acceleration and Security

Alibaba Cloud achieves content acceleration and security in two ways: using CDN and OSS to build an accelerated distribution service for static browser game content such as textures, UIs, audio, sounds, special effects, and optimizing resource loading policies. The following methods are generally effective: reducing resource size, pre-loading, and loading as needed. The OSS service supports image processing for efficient batch compression, cropping, transcoding, and other image operations.

4. Partitioned Server Architecture

The architectural diagram shows that server-based MMORPGs have basically the same architecture as traditional PC-based Massively Multiplayer Online games. The difference is that browser games rely more on CDN for accelerated distribution of large volumes of static resources for players to play and download simultaneously. Console games require players to download the entire game client before playing. In addition, MMORPG browser games can go without centralized storage methods. They use a specialized distributed method. Furthermore, casual and MMORPGs use different game server deployment methods because of their different server interaction requirements. Casual browser games generally deploy all game services for a region in the same machine room or data center. MMORPGs adopt a distributed nearby deployment method like traditional console games.

2.4 Casual Browser Gaming Solution Overview

As an extension to online gaming, a casual browser-based game is played entirely within a web browser instead of a console or another gaming apparatus. Players can access a game from different computers or machines and can involve different players at one given point in time. Unlike MMORPGs, which are aimed at committed or serious gamers, casual browser games are for individuals who wish to indulge in gaming to spend their leisure time. These games often include simple puzzles, but there are instances where these games can be quite detailed and involve an extensive environment.

Anyone can access casual browser games through a computer that is connected to the Internet, although the computer may have certain software prerequisites to play the game. Given its nature, the game runs in a browser, and the user does not need to download it to their desktop or have a special gaming device to play the game. However, the player does need to play the game on the same browser, as that is where it is hosted.

The actual gameplay is normally dependent on player involvement. As described in Csikszentmihalyi's Concept of Flow: "The mental state of operation in which a person performing an activity is fully immersed in a feeling of energized focus, full involvement, and enjoyment in the process of the activity." In other words, if a game challenges the psyche of the player and at the same time allows improvement of skill, the player will be enticed to continue playing. It may also depend on the player's motivation for playing the game. For example, if the player has an opportunity to master a skill or go up a level, the game is more likely to engage them.

Casual Browser Game Developer Challenges

In terms of usability, it is important that the interface doesn't distract from the game itself in order to drive better engagement. Lag is an issue that can break the deal for any browser game because it ruins a player's sense of engagement. Therefore, a casual browser game is engaging only if it's designed to function flawlessly in a browser. Apart from that, other issues may include feedback mechanisms that have to be facilitated using soft notifications and functionality in terms of a player knowing their movements.

With the development of the World Wide Web, web browsers have undergone major advancements, which has led to developers creating more sophisticated browser games. Originally, they created simple, single player games that could be played on a Web browser using HTML and HTML scripting technologies (JavaScript, ASP, PHP, and MySQL).

Improvements in web-based graphic technologies, such as Flash and Java, has enabled the casual browser games to become more complex and endearing. Many games originally released in the 1980s, such as Pac-Man and Frogger, were recreated as contemporary games and now played using Flash plugins on a web page. Most of these traditional casual browser games have limitations when it comes to multiplayer play, as their score format was designed to be shared amongst all players.

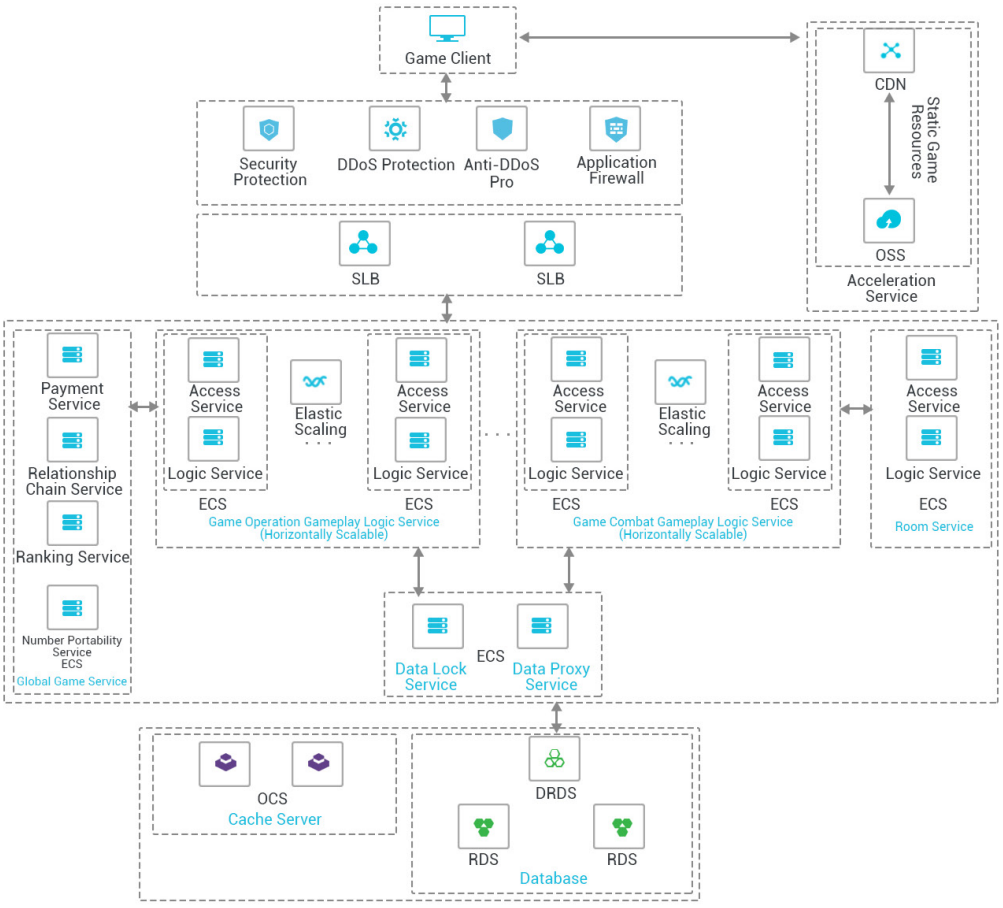
However, this trend has seen a dynamic change in the recent years with the likes of Castle of Heroes and Canaan Online coming into the fray.

Recent casual browser games use web technologies such as Ajax, which aids in making more complicated multiplayer exchanges possible, and WebGL, which helps generate hardware-accelerated 3D graphics without the need for plugins. Java most certainly has become the most widely used programming language for browser games.

Recommended Casual Browser Cloud Gaming Solution

Alibaba Cloud gives game developers the advantage of a maintenance-free, scalable, flexible, and robust cloud platform that helps distribute games efficiently and caters to a large number of concurrent gamers.

The following solution illustrates the recommended casual browser cloud gaming solution.



The server architecture for most traditional browser games is relatively simple. Basically, it uses a two-layer design with a logic layer and database layer. For example, the simplest deployment model would be a game service composed of one ECS plus one RDS instance. This design has some obvious drawbacks (such as single-point-of-failure). As competition in the gaming industry has intensified, each small improvement to the game experience is extremely important. Therefore, Alibaba Cloud's casual online browser game solution offers an advanced server deployment scenario.

1. Accelerated Static Game Content

Alibaba Cloud Content Distribution Network (CDN) and Object Storage Service (OSS) build an enhanced distribution service, which helps with acceleration of a static browser game content such as textures, UIs, audio, sounds and special effects. Alibaba Cloud CDN nodes are available at hundreds of locations covering China and other global regions. Developers can achieve automatic detection and delivery changes to objects stored in OSS, which is facilitated by the seamless integration of Alibaba Cloud OSS and CDN. Also, CDN provides API control, allowing users to conveniently preheat resources through the API. OSS can effectively be used to compress, crop, transcode and process images in batches.

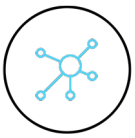
2. Multi-Level Game Security

Alibaba Cloud Security Services provides credible defenses through its Anti-DDoS product with a highly protected IP address and ESN featured platform. The WAF product of the Security Service effectively prevents hackers from using web vulnerabilities for infiltration and penetration providing an additional web security for browser game servers and sites. The Server Guard feature of the Security Services performs vulnerability scans and vulnerability analysis on servers detecting any threats present. Additional anti-plug-in features such as parameter validation maximum possible server reproduction of operation results, and verification code mechanisms are added in the game service.

3. Three-Level Server Architecture

The architectural diagram shows that casual browser game server designs can create a typical three-level architecture, including an access, logic, and data layer. The access and logic layers adopt a non-data stateless design. The Alibaba Cloud Server Load Balancer can perform load balancing for each layer, preventing service unavailability caused by a single-point-of-failure. For casual browser games, the Alibaba Cloud Auto Scaling can perform smooth automatic resizing on core game logic clusters. The server's logical computing layer is designed to be modular, primarily to deliver global and cluster services. The data layer is subdivided into proxy, cache, and persistence layers. As social causal games generally have offline gameplay aspects, multiple online gamers may interact with the same offline gamer, which may in turn lead to data inconsistency. Usually, such games use a data read/write lock to support concurrent data reading/writing.

4. Distributed Databases Supporting High Concurrence



Casual browser games generally adopt a game world mode of operation. This requires a persistence storage layer for globally shared data. In addition, due to the large amount of offline data operations, they require a data read/write lock service to ensure data consistency. Developers can use Alibaba Cloud Distributed Relational Database Service and Relational Database Service offered via ApsaraDB for RDS to build a distributed database that supports high read/write concurrency. For example, database and table partitions can be used for player data, with the player ID used as the table partition key. In addition, the architectural design allows for convenient data resizing at the data persistence layer.

2.5 Alibaba Cloud Advantage for Browser Cloud Gaming Solution

1. Dedicated Regional Server Databases

For a MMORPG server, its data layer is usually equipped with a dedicated database for a single region that enables primary/secondary disaster recovery for database services. Games that frequently have a single server online must set up a data cache layer to lower the database read/write requirements.

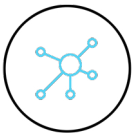
2. Secure and Stable Game Servers

Through protection offered by Alibaba Cloud Anti-DDoS IP addresses, WAF, penetration testing, and other Alibaba Cloud security services, the risk of frequent attacks on browser game servers can be avoided. This provides gamers with a more stable game service, while a series of anti-plugin policies ensure a level playing field.

3. Elastic Resource Scalability

By combining Alibaba Cloud Server Load Balancer and Auto Scaling, developers can easily perform automatic online scaling for game logic servers to respond to sudden changes, such as the increased pressure on servers caused by drastic changes in the number of game users. This can also help users most efficiently use server resources.

4. Distributed Databases Supporting High Concurrent Access



The Alibaba Cloud Relational Database Service and Distributed Relational Database Service like ApsaraDB for RDS and DRDS combine to provide a database service supporting high concurrent read/write and horizontal extensions. In addition, database capacity and specifications can be flexibly adjusted based on different service cycles.

5. Stable Static Game Content Acceleration

The Alibaba Cloud CDN possesses rich node resources and excellent image download performance. Combined with OSS, it creates an integrated system that stores, processes, and distributes static browser game content. This provides convenient image processing and nearby download capability with a high throughput to deliver a stable and fast gaming experience for players. Plus, the solution effectively reduces bandwidth and traffic costs.

03 Browser Gaming: What the Future Holds

Despite the prolonged existence of the web, browser games remain popular. The advent of HTML5 has found players and developers strategically adapting to new technology. Developers are creating quality games in HTML5 that form tough competition for their Flash counterparts. Current search trends bend in favor of browser games over Flash games. The future therefore looks promising for browser gaming.

The verdict is still out on whether browser games will become the gaming technology of choice. However, it is absolutely certain that online casual gaming has already seized and sustained the interest of millions of gamers, and developers in future will be vying to capture this attention with new browser games.

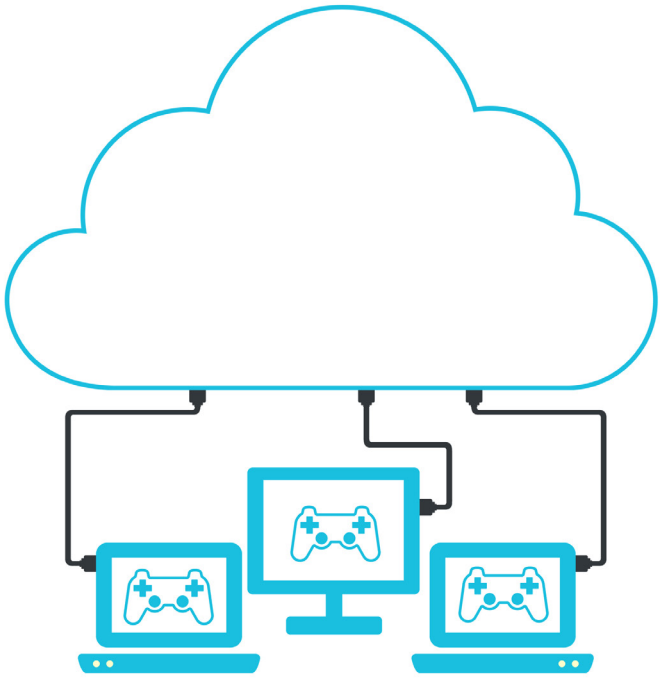
The evolution of browser gaming has led to the creation of many sophisticated and advanced games that have managed to appeal to hardcore and serious gamers. Despite the limitations in bandwidth of web browsers to power games with high-end graphics, experts have touted that modern web browsers have the potential to drive the future of gaming. As a result, many gaming and technology industry bigwigs have tried to convert the modern browser into a high-end gaming console. For example, Mozilla showed the world that games driven by Unreal Engine 3 (the same engine that powers high-end games on many gaming consoles and mobile devices) can effectively be played on its Firefox web browser on an archetypal computer.

The main constraint of browser gaming lies in the browser's inability to deliver high speed gaming compared to a native gaming console. This has been the biggest obstacle in the growth of browser gaming. Mozilla has shown allegiance to Java Script language, a JS code optimizer called ASM.js, and a JS compiler named Emscripten, to provide a high-end browser gaming experience that can be compared to the performance of gaming consoles. However, gaming consoles meant exclusively for gaming are much faster and offer improved performance. There is still a long way to go for browser gaming to cause avid gamers to dump their consoles.

04 Conclusion

Web browsers have been an integral part of gaming. These days, browser developers aim to cater to hardcore gamers who are looking for a comprehensive gaming experience without external support from a console or other equipment. With technology becoming more sophisticated, the gap between browser and console gaming is also becoming narrower. Even though gaming consoles still have an upper hand when it comes to quality gaming experience, the evolution of browser games has been transcendental. The last few decades have overseen the journey of browser gaming from clunky gameplay and patchy graphics to high-end and life-like gaming capabilities.

Smartphones and other mobile devices have helped to attract casual gamers, but staunch gamers still find solace in their gaming consoles. Therefore, to uplift the quality of browser games, developers must make sure that games are free from lag time and run flawlessly on a regular machine without software for external support. Also, with the price benefits that cloud offers, this trend promises to be the next big thing in browser gaming.



Alibaba Cloud
aliyun.com