Scholar Publishing Group
International Journal of Big Data Intelligent Technology
https://doi.org/10.38007/IJBDIT.2026.070106
ISSN 2790-0932 Vol. 7, Issue 1: 45-51

Data Analysis and Performance Evaluation in Game Product Iteration

Xia Hua

SMU Guildhall, Southern Methodist University, Texas, 75205, USA

Keywords: Game iteration, Data analysis, Performance Evaluation, User Experience

Abstract: In today's increasingly competitive gaming industry, continuous product iteration has become an effective method to maintain user stickiness and extend the lifecycle. However, excessive product iteration frequency often leads to problems such as confused goals, unstable performance, and delayed response, seriously hindering the gradual improvement of game quality. This article mainly focuses on the utilization of data analysis and performance evaluation in game product iteration. This article systematically elaborates on the role and impact of user behavior, performance indicators, and collaborative methods in product optimization and improvement processes. It deeply analyzes the main problems such as lack of data fusion, incomplete performance measurement system, and unscientific collaborative strategies. Measures such as establishing a unified database system, improving performance monitoring network, and building comprehensive and diverse collaborative strategies are proposed to provide practical guidance and achieve good suggestions and scientific theories for high-quality game product iteration, and to move towards the transformation goal of the entire industry from experience driven to big data-driven.

1. Introduction

With technological advancements and changing user demands, game products are being updated at a high frequency. Improving product quality, continuously optimizing experience, and ensuring stable operation have become the key to the success of product market and the improvement of user satisfaction. Traditional methods that rely on personal experience and post feedback are difficult to address system prediction and performance assurance issues. Incorporating data research and performance evaluation into the update process has become a key focus of current game product design. Data analysis can reflect user behavior and provide a basis for decision-making; Performance evaluation ensures smooth system operation and seamless user experience. The combination of the two is expected to promote more efficient and intelligent design. This article discusses the actual application status, existing problems, and improvement plans of the product, aiming to promote the transformation of the game product update mechanism from popular experience to a scientific and professional direction.

2. Overview of game product iteration

Copyright: © 2026 by the authors. This is an Open Access article distributed under the Creative Commons Attribution License (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (https://creativecommons.org/licenses/by/4.0/).

For games, "game iteration" refers to the process of adjusting or updating the components of the system (game functions, levels, etc.) within a certain period of time to adapt to changes in players, thereby enhancing user stickiness and revenue generating ability. Compared to conventional software updates, this approach involves a wider range of aspects, such as level design, rule adjustment, numerical balance, and optimization of art resources. With the development of mobile and online games, "iteration" has become an important management link in the lifecycle. Performance evaluation and data analysis are important basis for iterative optimization, including player behavior, market feedback, platform compatibility, and other aspects. Although iteration can stimulate creative passion and improve product quality, it inevitably leads to system instability, interrupted user experience, and other issues. Balancing the creative space of creators while ensuring product quality has become a challenge for developers, providing a basis and theoretical premise for further research and practice.

3. Application of Data Analysis and Performance Evaluation in Game Product Iteration

3.1 User behavior data analysis supports version optimization

In the optimization of game versions, user behavior data has become the main basis. In the process of version updates and upgrades, it is constantly necessary to understand where users touch the most frequently, their favorite function operations, dwell time, behavior routes, and other information to form a behavior path diagram, in order to determine whether the designed functions are effective, smooth, and whether there are process gaps. Relying on the results of behavior data can also help understand the specific situation of the entire process.

Taking the completion status of multi-stage tasks as an example, if the number of completed users in each stage is expressed in sequence as $x_1,x_2,x_3,...,x_n$,Then a continuous conversion ratio model can be constructed, with a conversion efficiency of:

$$\theta = (\frac{x_2}{x_1}) \times (\frac{x_3}{x_2}) \times \cdots \times (\frac{x_n}{x_{n-1}}) \tag{1}$$

This product model can be used to evaluate the overall conversion efficiency in the complete behavioral chain θ,When a certain ratio is much lower than other stages, it usually means that there is a user churn problem in that step. By comparing different versions θ The trend of change,It can be determined whether the functional design can be improved or if the content can be rewritten to promote positive behavior transfer and provide quantitative basis for version updates.

3.2 Performance indicator evaluation ensures stable operation of the system

In the iterative process of game products, the measurement of whether their performance evaluation is stable and fluctuating is often a key indicator that affects the quality of player experience. For example, situations such as lag, slow loading, memory overflow, and crashes during gameplay can often lead to a deep gaming experience for players. To achieve this, it is necessary to establish a complete set of performance evaluation standards suitable for the client, server, and network levels, and to combine them with the entire product development and lifecycle supervision process.

Common performance indicators include average frame rate (FPS), memory usage, and response time. Frame rate dispersion is the most reflective of the stability of graphics performance, and is one of the performance indicators of graphics. It can be represented by the standard deviation model as follows:

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(f_i - \mu)^2} \quad (2)$$

Among them,f_i Indicate the i The actual frame rate of the frame,μ For the average frame rate during the testing period,n For frame rate.Lower σ The value indicates that the frame rate fluctuates less and the image performance is more stable. Based on the above comparison and collection, for the newly added content or features in the iterative version, the performance load of the new content or features on the performance can be accurately calculated in the performance evaluation of the iterative version, providing quantitative data analysis support for performance optimization and release decisions, and reducing performance risks and hidden dangers after the system version is launched.

3.3 Collaboration between data and performance enhances the overall experience

The ultimate experience of a game product is not determined by the planning content or strategic gameplay innovation, and the user experience is not solely determined by data analysis or smooth operation. It is also necessary to integrate these two points to achieve better satisfaction for users. Therefore, during the version iteration process, it is recommended to use a common application system of game product behavior data analysis and performance evaluation to conduct comprehensive quality assessment of the product.

By considering a weighted rating model, the overall user experience level is comprehensively calculated, mainly based on the combination of user behavior activity and system stability. The model is represented as follows:

$$S = \alpha \cdot U + \beta \cdot P \quad (3)$$

Among them, S represents the comprehensive experience score, U is the user behavior score (such as retention rate, active time),P is the performance score (such as frame rate stability, loading speed),α give β For weight coefficients, satisfy $\alpha + \beta = 1$。

By adjusting weights, changes in different game periods or product strategies can be adjusted. In the practical process, the overall user experience of different versions can be compared, problems can be revealed, and improvement suggestions can be given. This can guide the focus on comprehensive evaluation of data analysis and performance evaluation, which is conducive to promoting product optimization from a single link to system improvement, and improving the accuracy and efficiency of version iteration.

4. Problems in Data Analysis and Performance Evaluation in Game Product Iteration

4.1 There is an information gap in data collection and utilization

As one of the core supports for the continuous iteration of game products, data collection and use have a direct impact on the accuracy of subsequent analysis and the rationality of optimization paths for game products. However, in actual development and operation, structural problems in the data chain often arise, seriously restricting the realization of the value of data. The diversity of source data increases the complexity of management. The information from client records, server monitoring, user experience, and business platforms is generally distributed across different systems and groups, lacking a unified digital bridge and data system. In the process of data analysis, there may be obstacles such as inconsistent fields, different update speeds, and duplicate dimensions, which increase the cost of cleaning and matching and reduce the efficiency of publishing analysis

results.

The accuracy of action recording is low, and it is easy to miss important action steps. When the project is updated quickly, the newly added parts are not given enough time to set detection points, and the user's core process actions and page access trajectories cannot be fully reproduced; Lack of overall thinking about the system has resulted in a large amount of hollow or repetitive content being piled up in the action diary, limiting the interpretability of the research.

Although the collected data analysis is complete, the results of using data analysis did not meet expectations. In some special projects, these data analyses are only used for reporting and routine management monitoring, failing to guide product optimization and functional improvement. Due to the fact that the analysis results are not adopted in the product decision-making process, this leads to the separation of analysis and product development work, resulting in the loss of meaning for analysis and verification.

Information fragmentation is mainly caused by inadequate communication among departments. When the data team and product developers or researchers have different ideas about their intentions, indicator settings, model principles, etc., it can result in the data analysis models they build not being used for practical needs, and can also lead to misunderstandings and usage situations. The information fragmentation that occurs also affects the scientific decision-making foundation of game products during the iteration process.

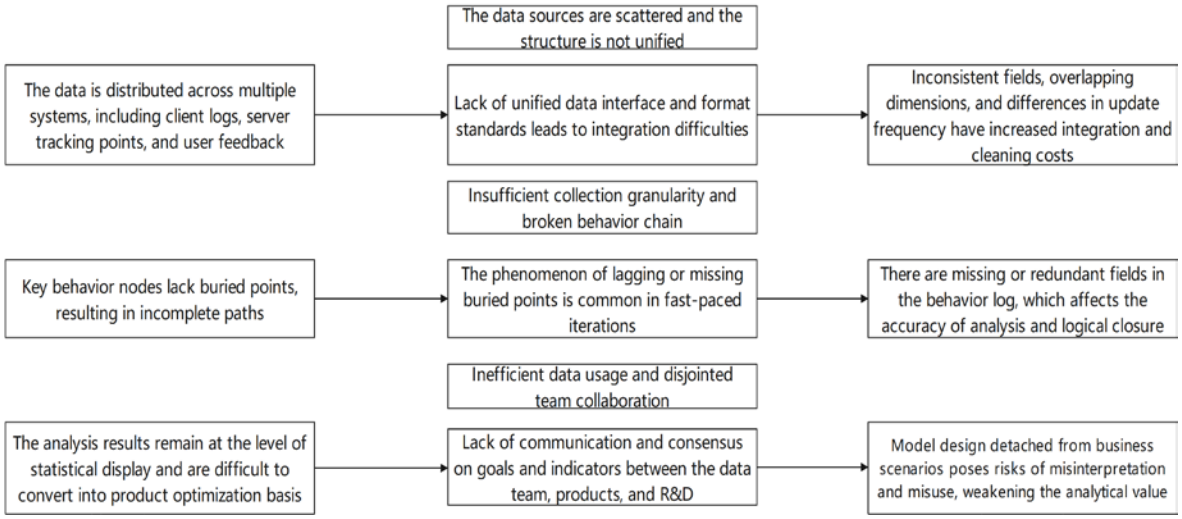

Figure 1: There is an information gap problem in data collection and utilization

4.2 The performance monitoring system is incomplete and feedback lags behind

With the high iteration frequency of game products, performance evaluation often becomes an important factor affecting satisfaction. However, the existing performance evaluation system still has certain shortcomings in many projects, resulting in inadequate problem response and hindered version stability maintenance. Most teams lack a comprehensive performance testing system during the development phase, relying solely on user complaints after release to identify problem areas, which cannot effectively detect potential performance risks.

Another obvious issue is that performance evaluation is not comprehensive, and some key scenarios (such as simultaneous large-scale battles, multiplayer interactions, and complex physical calculations) have not been well considered, resulting in detection results that are not universally applicable. In addition, due to the lack of statistical and performance evaluation standards, as well as alarm systems, it is impossible to automatically detect and report problems when they first become apparent. Investigation can only be carried out after users have started to lose interest and

ratings have decreased. Due to the loose coupling between performance evaluation and software release timing, optimization lags behind software release, resulting in a prominent "slow response" problem that cannot meet the user experience requirements guaranteed in frequent iteration environments.

4.3 Lack of collaborative linkage between data and performance evaluation

In continuously iterating products, data analysis and performance evaluation are often conducted independently by different teams, lacking effective collaboration. Although this division of labor has clarity in responsibility allocation, it can easily lead to incomplete information and optimization breakpoints. The lack of a correlation model between data analysis and performance data, as well as a synchronization mechanism, makes it difficult to explain and trace user experience issues from a system performance perspective. When relying on user feedback to optimize functional frameworks, product departments often overlook the impact of changes on performance; When the technical department is improving performance, it is difficult to reasonably determine the optimization direction and priority due to the lack of user behavior data. At the same time, the lack of a unified visualization platform hinders various departments' understanding of version quality, leading to a decrease in the quality and efficiency of iterative upgrades. The lack of collaborative mechanisms directly weakens the support of data analysis for product decision-making and limits the practical application of “precise tuning” in the iterative process.

5. Data analysis and performance evaluation optimization strategies in game product iteration

5.1 Unified data platform improves collection and analysis efficiency

A unified information system platform is a prerequisite for improving efficiency and rapid response. By obtaining device collected information, server logs, user interaction data, and third-party source information, a central management system is established to achieve automatic information entry, hierarchical storage, and standardized conversion. The platform needs to have the ability to handle structured and unstructured data, unify field naming and labeling systems, and ensure consistent and compatible use of information. Automated ETL processes significantly improve data flow efficiency and support near real-time updates and synchronization of static data. BI visualization tools integrate key indicators into a unified dashboard, promoting collaboration among product, data, and development teams. The platform should also maintain open interfaces and flexible extensions to facilitate access to model analysis and prediction engines, providing stable data support and dynamic adjustment capabilities for version optimization.

5.2 Improve monitoring system risk warning and feedback response

By forming a full chain performance monitoring network at key nodes such as research and development, testing, and deployment, we ensure that performance data is controllable. By using automated performance evaluation tools to perform both stress testing and load testing during the compilation process, vulnerabilities in performance during the release phase have been prevented. According to statistical data, by introducing continuous integration testing, the crash rate can be reduced to about 38% of the original, and the fluctuation of loading time can be reduced to about 22% of the original.

Standard evaluation indicators should be used for system evaluation, benchmark models should be established, historical versions of data should be compared, and dynamic warning values should

be set. If the frame rate increases or decreases by more than or equal to 15%, the memory usage increases by more than or equal to 20%, and the average delay difference exceeds 30%, the system will automatically issue warning messages to inform the research and maintenance teams of abnormal situations and provide feedback in a timely manner.

During the implementation phase, real-time sampling technology will be applied on the client side, and performance monitors will be integrated with the server side to obtain usage information for various models, network environments, geographic locations, and user groups. Multi dimensional data analysis will be used to comprehensively capture performance degradation points in specific usage environments, supporting effective version release and multi platform debugging. Establish relevant models for performance evaluation and user operation events, obtain the risk of user churn caused by performance evaluation, and focus on the points that cause changes in user churn rate exceeding 10%. The comprehensive detection and intelligent alarm mechanism greatly improve the game system's ability to face complex environments, as well as the consistency and stability of released versions.

Table 1 Percentage change of key indicators after optimization of performance monitoring system

Indicator category	Performance before optimization	Optimized performance	Improvement range(%)
Average crash rate	4.5%	2.8%	↓ 38%
Load time fluctuation amplitude	±1.8second	±1.4second	↓ 22%
Frame rate fluctuation threshold setting	Not set	Setting deviation≥15% early warning	—
Abnormal warning line for memory usage	No unified standard	An increase of over 20% triggers a warning	—
Delay over limit alarm triggering ratio	Passive Response	Exceeding the average value by 30% will trigger a warning	—
Rework rate for issues after version release	21.0%	12.2%	↓ 42%
Tracking area for abnormal fluctuations in rate	unmarked	decline≥10%Enter the tracking list	—

5.3 Establish a collaborative mechanism to integrate data and performance

To optimize the game version, it is necessary to establish an efficient collaboration mode between the product team, data team, and technical team, and deeply integrate user behavior data and performance evaluation data. By establishing a unified evaluation standard, a complete experience research system is formed by linking user behavior indicators such as action frequency and participation with performance indicators such as frame rate and loading time. Adopting a dual level evaluation system, combined with user behavior deviation and performance changes, can effectively identify key factors that affect user satisfaction.

Establish a summary data dashboard to achieve synchronized sharing and dynamic updates of data across departments, facilitating efficient collaboration among different roles based on unified information. In the improvement process, a multi-party collaborative review mechanism is introduced to synchronize the analysis results, performance feedback, and design adjustment opinions in real time, further improving the response speed and coordination of version updates. By

building a data-driven collaborative feedback loop, the improvement activities will shift from scattered stages to systematic evolution, providing stable support for continuous product iteration.

6. Conclusion

The speed of product iteration has accelerated, and user experience and system stability have become important evaluation indicators for new versions. Data analysis and performance evaluation have played an important role in this, enhancing the scientificity and accuracy of decision-making and increasing support for optimization. The focus of this article is on user behavior data analysis, system performance evaluation, and the correlation between the two. It points out that the problems lie in data analysis, performance evaluation, team collaboration, and other issues. It also proposes some improvement measures, such as a unified data platform, building a sound performance evaluation system, and establishing collaborative relationships. It points out that comprehensive evaluation of multiple types of data can effectively promote the effectiveness of new versions and improve user experience. In the future, more intelligent analysis methods and automated diagnostic tools can be used to promote the development of game products, from "data-driven iteration" to "intelligent iteration".

References

[1] *Clayton R. Efficient instruction scheduling using real-time load delay tracking. Computing reviews, 2023, 64(12):302-302.*

[2] *Blevins J R. Leveraging Uniformization and Sparsity for Computation of Continuous Time Dynamic Discrete Choice Games. 2024.*

[3] *Hyyti E, Righter R. Optimal Size-Aware Dispatching Rules via Value Iteration and Some Numerical Investigations. 2024.*

[4] *Nakhleh K, Eksin C, Ekin S. Simulation-Based Optimistic Policy Iteration For Multi-Agent MDPs with Kullback-Leibler Control Cost. 2024.*

[5] *Gargiani M, Sieber R, Balta E, et al. Inexact Policy Iteration Methods for Large-Scale Markov Decision Processes. 2024.*

[6] *Lu, Z. (2025). Design and Practice of AI Intelligent Mentor System for DevOps Education. European Journal of Education Science, 1(3), 25-31.*

[7] *Yu, X. (2025). Application Analysis of User Behavior Segmentation in Enhancing Customer Lifetime Value. Journal of Humanities, Arts and Social Science, 9(10).*

[8] *Zheng, H. (2025). Research on Lifecycle Configuration and Reclamation Strategies for Edge Nodes Based on Microservice Architectures. Advances in Computer and Communication, 6(5).*

[9] *Dingyuan Liu. The Relationship between Household Consumption Pattern Changes under Disasters and the Recovery of Business Ecosystems. Academic Journal of Business & Management (2025), Vol. 7, Issue 12: 151-156.*

[10] *Zhen Zhong. Big Data Engineering and Intelligent Analysis Framework for Compliance Investigation. Academic Journal of Computing & Information Science (2025), Vol. 8, Issue 11: 107-115*

[11] *Li, J. (2025). The Impact of Distributed Data Query Optimization on Large-Scale Data Processing.*

[12] *Lu, Z. (2025). AI-Driven Cross-Cloud Operations Language Standardisation and Knowledge Sharing System. European Journal of AI, Computing & Informatics, 1(4), 43-50.*