Azerbaijan Journal of High Performance Computing, Vol 5, Issue 2, 2022, pp. 183-192
https://doi.org/10.32010/26166127.2022.5.2.183.192

AzJHPC
Azerbaijan Journal of High Performance Computing

*Correspondence:
Rustam Eyniyev,
Azerbaijan State Oil
and Industry University,
rustam.eyniyev@gmail.
com

Optimizing Parallelism in Unity with Job System Tool

Rustam Eyniyev

*Azerbaijan State Oil and Industry University, Baku, Azerbaijan,
rustam.eyniyev@gmail.com*

Abstract

Every day the gaming industry is developing faster and faster. Especially the popularity of games increased during the pandemic, as many people were forced to stay at home for a long time and play games in their free time. With the increasing popularity of games, the requirements for the games themselves also increase. As a result, modern games require large computer resources. However, not all people can afford to buy computers with the most expensive video cards, many expensive CPUs, and many RAMs and other components. In order to make games accessible to users with weaker devices, developers have few options. At the moment, there are two main ways to solve this problem. The first is to host games on the cloud, and the second is to optimize games for weaker devices. To optimize games, you can use various methods, including a package Job System for Unity. The Job System is one of the main components for working with threads in an Entity Component System (ECS). Also, in the future, in games with the Unity engine, parallelization between weak devices and High-Performance Computing (HPC) technologies will be combined and implemented using cloud technologies. In the future, when this happens, it will help developers, gamers, and even companies providing cloud services.

Keyword: Game development, Unity, Multithreading, Entity Component System, Job System, High Performance Computing, Cloud Technologies.

1. Introduction

During the pandemic, interest in games has increased for various reasons.

Some people, for example, relieved stress and anxiety by playing games because, during the pandemic, people's physical and mental health suffered. The mental health of adults and children suffered (Imran, N., Zeshan, M., & Pervaiz, Z., 2020; Ford, T., John, A., & Gunnell, D., 2021; Usher, K., Durkin, J., & Bhullar, N., 2020). For several reasons, children are initially more interested in games than adults. Healthcare workers also suffered mentally during the pandemic (Stuijfzand, S., Deforges, C., et al., 2020).

Other people used games to learn something new. They studied something new to improve their professional skills; for example, someone learned English through games

(Khasawneh, M. A. S., 2021).

Some companies also began using games to improve work efficiency in the office, remote meetings, and other essential aspects (Chanana, N., 2021).

Games have been introduced in some educational institutions for online lessons (Bar-El, D., & E. Ringland, K., 2020, September). The increased popularity of gamification during online lessons helped increase the effectiveness of online lessons (Park, S., & Kim, S., 2021; Nieto-Escamez, F. A., & Roldán-Tapia, M. D., 2021; Sillaots, M., Jesmin, T., Fiadotau, M., & Khulbe, M., 2020, September).

Universities also create games that help students learn research methods. For example, these can be browser-based adventure games that do not require many resources and do not take up a large amount of memory on the user's device (Sillaots, M., Fiadotau, M., et al., 2020, October).

There is also a category of games that are considered serious games. Several serious games have been developed to help healthcare professionals from various fields improve their skills during COVID-19 (Hoopes, S., Pham, T., Lindo, F. M., & Antosh, D. D., 2020; Sipiyaruk, K., Hatzipanagos, S., Reynolds, P. A., & Gallagher, J. E., 2021).

Military serious games were popular even before the pandemic. Before being sent to war, many soldiers are trained for specific military operations and possible situations through virtual reality simulation (Engberg-Pedersen, A., 2017).

There are also serious games that help people exercise while they play these games. These games help fight against various dangerous diseases. Sometimes they have a multiplayer mode, allowing users to compete in various sports games with other players (Senevirathne, S. G., Kodagoda, M., et al., 2011).

During the pandemic outbreak, museums were among the most affected institutions. Many of these museums even had to close (Marty, P. F., & Buchanan, V., 2022). The primary way out of this situation for museums was the creation of digital museums. These digital museums encourage people to keep visiting various ways (via games, virtual exhibitions, VR/AR, etc.). When using games for virtual museum visits, museums allow all users to create avatars for themselves and use a multiplayer system. This was done for people who visited museums to see works of art and communicate with others, and share their impressions about some paintings (Markopoulos, E., Ye, C., Markopoulos, P., & Luimula, M., 2021, July).

Streamers of players before the pandemic also significantly impacted the game industry (Johnson, M. R., & Woodcock, J., 2019). However, during a pandemic, streamers let players who broadcast how they play games on video receive a good increase in audience and financial profit (Scerbakov, A., Pirker, J., & Kappe, F., 2022, January). These factors further increased their impact on the game industry. This happened precisely because not all people could play some game novelties themselves. The main reason is that these game novelties are not supported by their computer, as the games require more resources than they have on their computer.

This is because some developers need to optimize their games fully or cannot optimize even more than they have already done.

At the moment, there are two main ways to run CPU-intensive games on weaker devices so as not to harm the user's computer:

1. Cloud technology
2. Optimization of the game code

The first method has many disadvantages:

1. It requires regular payments for a subscription
2. Cloud systems are still being prepared to take all the games to their servers.

However, despite these disadvantages, the popularity of cloud technologies is growing. The areas of application of cloud technologies are increasing (Ismayilov, E., 2022).

The second method has no definite limits, and you can constantly improve the optimization. However, even the most optimized game on weak devices may fail to run. For example, rendering 50,000 characters, even with perfect code optimization, can be very slow on weaker devices.

In the future, partial execution of processes in a combination of HPC and cloud technologies and partial execution on weak devices with a multi-core processor can help solve these problems. In the future, the implementation of such a concept is possible in the following form: At the same time, resource-intensive tasks (rendering, boss fights, complex mathematical calculations, etc.) will be performed using HPC on the side of cloud servers and quickly downloaded to the player’s computer using high-speed Internet, while ordinary game tasks (using items from the inventory, dialogues between characters, some regularly repeated processes) will be executed on the player's computer immediately. This will reduce high CPU usage on the weak components of gamers' computers and on cloud servers.

Because the capabilities of user devices are growing every day, developers can use data-oriented design and multithreading technologies to improve the performance of their games and optimize resource-intensive mechanisms for weaker computers. All this development became possible only after the introduction of multi-core processors and Flynn's taxonomy (SISD, MISD, SIMD, MIMD) (Borufka, R., 2020).

SIMD (Single instruction, multiple data) and MIMD (multiple instructions, multiple data) are the most commonly used methods for accelerating programs and applications in HPC and machine learning (ML) programs. After the 80s, MIMD replaced SIMD (Rico, A., Joao, J. A., Adeniyi-Jones, C., & Van Hensbergen, E., 2017, May).

SIMD uses data-level parallelism, not task parallelism or concurrency (Sengupta, B., & Das, A., 2017). When using SIMD, data is shared between processing elements and executes the same instruction on them. An example of how SIMD and data parallelism works in practice: you need to display "Hello world" 1000 times. This data can be divided into 4 parts, allowing each processing element to display "Hello World" 250 times.

When using MIMD, there are multiple instructions, multiple data, or pieces of data, and these instructions are applied to them asynchronously and independently on different CPUs. An example of the use of MIMD is modeling and simulation.

Both SIMD and MIMD are necessary when working with HPC because there can be various tasks (Britt, K. A., & Humble, T. S., 2017). You can choose a different approach to solving the problem depending on the tasks.

When parallelizing in HPC, job scheduling plays a significant role. There are many different ways of job scheduling (Srinivasan, S., Kettimuthu, R., Subramani, V., & Sadayappan, P., 2002, August). Job scheduling runs Jobs and generally manages all the processes associated with jobs. Jobs are tasks that are given to the operating system (OS). Usually, they are given as a batch and run in the background when the computer is not performing some urgent and important task. Jobs may consist of sub-processes called steps.

Job scheduling is also important in cloud systems. The traditional, well-established task scheduling algorithms are ineffective for a cloud system. The algorithm needs to meet certain criteria. Therefore, new algorithms based on various principles are proposed for the cloud system (Ghanbari, S., 2019; Rezaei, F., & Ghanbari, S., 2021).

2. Job System Tool in Unity

You can use the package Job System to improve parallel work in Unity. The Job System works with multithread code using jobs, not threads. The job system uses worker threads on multiple cores so that each logical CPU core works with one worker thread. This helps to avoid context switching. Some cores may not be involved in this process, as they are reserved for the OS or other important applications or tasks.

Worker threads execute jobs from the job queue. The Job system controls this entire process. This tool also controls dependencies and checks the correct order of execution of jobs.

The Job System also considers the possible occurrence of a race condition when working with multithreaded code. This package initially detects potential race conditions and protects the user from their occurrence. For example, a copy of that data is used instead of referring to data. This helps to avoid such errors since each job works with data isolated from others.

2.1. Practice Tests with Different Conditions

In order to show that the Job System has a strong effect on game optimization, the tests will use 4 computers with different key characteristics, which will run the same simple application that will generate a number of sprites of the same type that will move horizontally.

The essential characteristics of computers are listed in Table 1.

Table 1: Important characteristics of computers

№	Processor	CPUs	Graphic Card	RAM
1st	AMD Ryzen 7 3750H	8	Nvidia GeForce GTX 1650	16GB
2nd	Intel Xeon W-2135	12	Nvidia GeForce GTX 1060 6GB	128GB
3rd	Intel Core i7-10070F	16	Nvidia GeForce RTX 3070	32GB
4th	12th Gen Intel Core i7-12700F	20	Nvidia GeForce RTX 3070	16GB

The FPS (frames per second) results when using this tool with different amounts of 2D object generation for all 4 computers are shown in Table 2.

Table 2: Test results without Job System

Number of objects	1st	2nd	3rd	4th
100	244.3	131.5	411.4	452.7
250	148.8	51.4	112.6	118.3
500	62.8	14.2	32.9	33.8
1000	19.3	5.8	8.5	8.9
2500	3.3	1.1	1.4	1.5
5000	0.92	0.27	0.34	0.36
10000	0.23	0.07	0.09	0.09
20000	0.06	0.017	0.022	0.022

The table results show that all selected computers, when generating and moving 5000 or more objects, give a result of less than 1 FPS. When generating 20,000 objects, all results become less than 0.1 FPS. In the game, this means that objects hardly move at all.

The results of tests in FPS using the Job System are shown in Table 3.

Table 3: Test results when Job System is used

Number of objects	1st	2nd	3rd	4th
100	263.3	449.9	707.5	882.2
250	235.2	306.5	597.5	650.8
500	180.7	240.2	451.6	470.4
1000	102.5	140.6	240.1	272.3
2500	30.5	37.5	68.7	72.2
5000	9.7	11.1	21.6	22.9
10000	2.6	2.8	5.8	6.1
20000	0.78	0.66	1.5	1.6

The results show that the Job System increased FPS on all tested devices in all cases.

A graphical display of the FPS before and after applying the Job System for each computer is shown in Figure 1, Figure 2, Figure 3 and Figure 4.

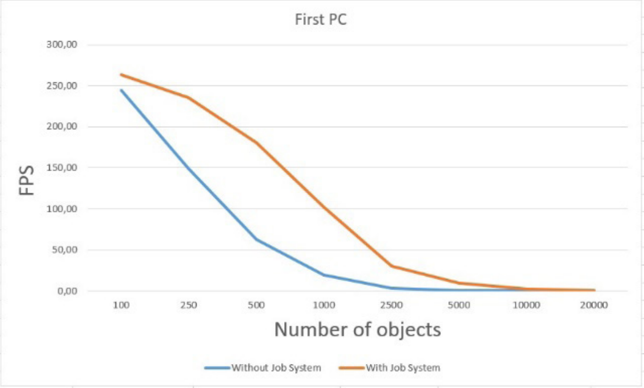

Fig. 1: First PC performance

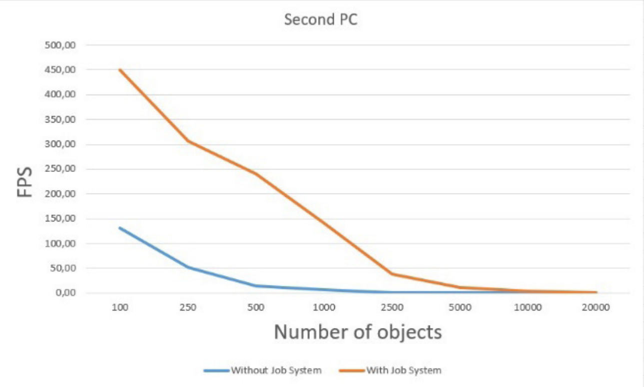

Fig. 2: Second PC performance

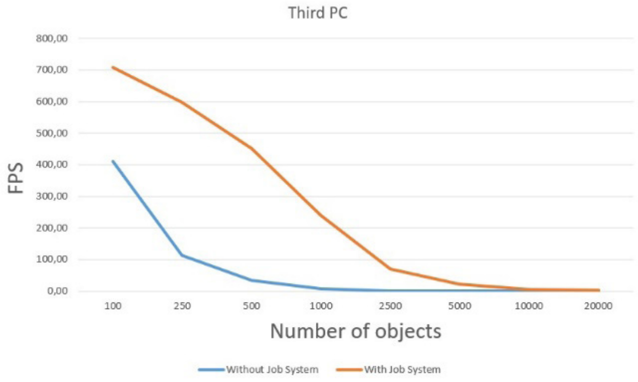

Fig. 3: Third PC performance

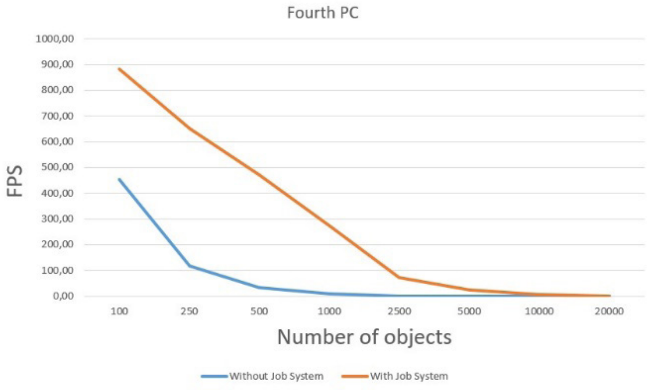

Fig. 4: Fourth PC performance

Table 4 and Figure 5, in the form of a table and a graph, respectively, show how many times the FPS increases when the Job System is used with a different number of objects for different computers.

Table 4: FPS improvement

Number of objects	1st	2nd	3rd	4th
100	1.08	3,42	1,72	1,95
250	1,58	5,96	5,31	5,50
500	2,88	16,92	13,73	13,92
1000	5,31	24,24	28,25	30,60
2500	9,24	34,09	49,07	48,13
5000	10,54	41,11	63,53	63,61
10000	11,30	40,00	64,44	67,78
20000	13,00	38,82	68,18	72,73

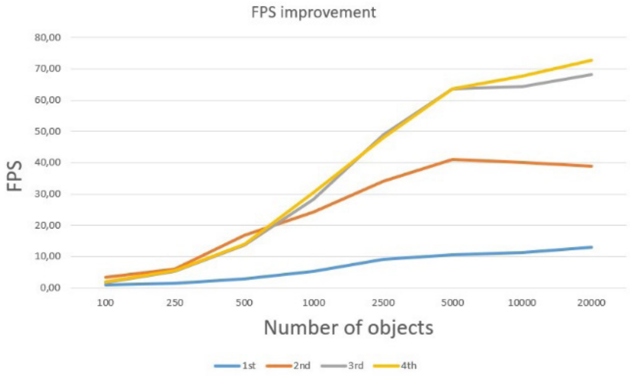

Fig. 5: FPS improvement

Conclusions

As a result of the tests, it turned out that with an increase in the number of CPUs, an improvement in Processors, and a Graphic card, the impact from the Job System also increased. Computers with a large number of processors, good processors for parallelization, and good graphics cards are used in HPC centers. In the future, when Unity implements the ability to run some processes on the user's device, and some third-party devices (HPC devices), the Job System will help parallelize, speed up and optimize processes on both the user's device and the HPC device.

Acknowledgments

Thanks for providing devices for testing results:

1. HPC Advanced Research Center of Azerbaijan State Oil and Industry University, Head of General and Applied Mathematics Department, Professor Araz Aliev, and Ph.D. on Engineering Elviz Ismayilov for providing access to the HPC Center.

2. European Commission and MAGnUS master's program for providing access to gaming computers.

References

Bar-El, D., & E. Ringland, K. (2020, September). Crafting game-based learning: An analysis of lessons for minecraft education edition. In *International Conference on the Foundations of Digital Games* (pp. 1-4).

Borufka, R. (2020). *Performance testing suite for Unity DOTS* [Master's thesis, Charles University - Prague] https://dspace.cuni.cz/bitstream/handle/20.500.11956/116800/120352992.pdf?sequence=1&isAllowed=y

Britt, K. A., & Humble, T. S. (2017). High-performance computing with quantum processing units. *ACM Journal on Emerging Technologies in Computing Systems (JETC), 13*(3), 1-13.

Chanana, N. (2021). Employee engagement practices during COVID-19 lockdown. *Journal of public affairs, 21*(4), e2508.

Engberg-Pedersen, A. (2017). Technologies of Experience: Harun Farocki's Serious Games and Military Aesthetics. *Boundary 2, 44*(4), 155-178.

Ford, T., John, A., & Gunnell, D. (2021). Mental health of children and young people during pandemic. *bmj, 372.*

Ghanbari, S. (2019). Priority-aware job scheduling algorithm in cloud computing: A multi-criteria approach. *Azerbaijan Journal of High Performance Computing, 2*(1), 29-38.

Hoopes, S., Pham, T., Lindo, F. M., & Antosh, D. D. (2020). Home surgical skill training resources for obstetrics and gynecology trainees during a pandemic. *Obstetrics and gynecology.*

Imran, N., Zeshan, M., & Pervaiz, Z. (2020). Mental health considerations for children & adolescents in COVID-19 Pandemic. *Pakistan journal of medical sciences, 36*(COVID19-S4), S67.

Ismayilov, E. (2022). Cloud Security: A Review of Current Issues and Proposed Solutions. *AzJHPC, 5*(1), 52-56.

Johnson, M. R., & Woodcock, J. (2019). The impacts of live streaming and Twitch. tv on the video game industry. *Media, Culture & Society, 41*(5), 670-688.

Khasawneh, M. A. S. (2021). The Effect of Using a Language Games-Based Electronic Program on Acquiring Oral Expression Skills among People with Learning Difficulties in English Language during Covid-19 Pandemic. *MANAZHIM, 3*(2), 136-150.

Markopoulos, E., Ye, C., Markopoulos, P., & Luimula, M. (2021, July). Digital museum transformation strategy against the Covid-19 pandemic crisis. In *International Conference on Applied Human Factors and Ergonomics* (pp. 225-234). Springer, Cham.

Marty, P. F., & Buchanan, V. (2022). Exploring the Contributions and Challenges of Museum Technology Professionals during the COVID-19 Crisis. *Curator: The Museum Journal, 65*(1), 117-133.

Nieto-Escamez, F. A., & Roldán-Tapia, M. D. (2021). Gamification as online teaching strategy during COVID-19: a mini-review. *Frontiers in psychology,* 1644.

Park, S., & Kim, S. (2021). Is sustainable online learning possible with gamification?—The effect of gamified online learning on student learning. *Sustainability, 13*(8), 4267.

Rezaei, F., & Ghanbari, S. (2021) Trust Base Job Scheduling in Cloud Computing. *AzJHPC, 4*(1), 39 – 47.

Rico, A., Joao, J. A., Adeniyi-Jones, C., & Van Hensbergen, E. (2017, May). ARM HPC Ecosystem and the Reemergence of Vectors. In *Proceedings of the Computing Frontiers Conference* (pp. 329-334).

Scerbakov, A., Pirker, J., & Kappe, F. (2022, January). When a Pandemic Enters the Game: The Initial and Prolonged Impact of the COVID-19 Pandemic on Live-Stream Broadcasters on Twitch. In *HICSS* (pp. 1-10).

Senevirathne, S. G., Kodagoda, M., et al. (2011). Application of serious games to sport, health and exercise.

Sengupta, B., & Das, A. (2017). Use of SIMD-based data parallelism to speed up sieving in integer-factoring algorithms. *Applied Mathematics and Computation, 293,* 204-217.

Sillaots, M., Fiadotau, M., et al. (2020, October). Methodyca: A Digital Game to Teach Research Methods. In *Proceedings of the 11th Nordic Conference on Human-Computer Interaction: Shaping Experiences, Shaping Society* (pp. 1-4).

Sillaots, M., Jesmin, T., Fiadotau, M., & Khulbe, M. (2020, September). Gamifying Classroom Presentations: Evaluating the Effects on Engagement Across Demographic Factors. In *European Conference on Games Based Learning* (pp. 537-XIX). Academic Conferences International Limited.

Sipiyaruk, K., Hatzipanagos, S., Reynolds, P. A., & Gallagher, J. E. (2021). Serious games and the COVID-19 pandemic in dental education: an integrative review of the literature. *Computers, 10*(4), 42.

Srinivasan, S., Kettimuthu, R., Subramani, V., & Sadayappan, P. (2002, August). Characterization of backfilling strategies for parallel job scheduling. In *Proceedings. International Conference on Parallel Processing Workshop* (pp. 514-519). IEEE.

Stuijfzand, S., Deforges, C., et al. (2020). Psychological impact of an epidemic/pandemic on the mental health of healthcare professionals: a rapid review. *BMC public health, 20*(1), 1-18.

Usher, K., Durkin, J., & Bhullar, N. (2020). The COVID-19 pandemic and mental health impacts. *International journal of mental health nursing, 29*(3), 315.

Submitted: 05.09.2022
Accepted: 28.10.2022