BTR Business Analysis

Team Members:
Aarushi Singh, Ana Izecksohn, Eduardo Gonzalez, Hetvi Shah, Paul Loh

Faculty Mentor:
Linh Thi Xuan Phan

TA Mentor:
Heena Nagda

Executive Summary

BTR is a data center implementation of the Bounded Time Recovery algorithm, which allows data centers to recover from Byzantine faults within a bounded period of time, providing a stronger fault tolerance guarantee while also saving significant resources. We capitalized on the fact that this algorithm is not widely known or implemented to enter the streaming data center market with a brand new product.

Business Analysis

Value proposition

Our algorithm allows data centers geared towards streaming applications to **save significant resources** while still maintaining **strong fault tolerance guarantees**.

In any distributed system, faulty nodes that crash, stall, or give the wrong output are inevitable. Data centers must have a model for the types of faults they are able to tolerate as well as a bound, f, on the maximum number of faults before the entire system stops working properly.

With our algorithm, clients can tolerate a higher number of faults using a smaller number of replicas. Specifically, a distributed system only needs f + 1 node replicas to handle f faults. Other algorithms such as Practical Byzantine Fault Tolerance, another popular alternative solution to Byzantine fault tolerance, requires at least 3f + 1 replicas to handle f faults.

In addition to requiring fewer replicas, our algorithm also tolerates a stronger fault model than most data centers currently protect against. Because of the high costs associated with Byzantine fault tolerance, most data centers do not actually tolerate Byzantine faults and instead implement a weaker form of fault tolerance. Our algorithm would make this stronger fault tolerance model more affordable for data centers.

The tradeoff for these benefits is that BTR can lead to a short, bounded, period of undefined behavior when faults do occur. However, for applications that are able to tolerate short periods of undefined behavior, the benefits outweigh the costs. While BTR has already been proven to work on cyber-physical systems, we are bringing the algorithm to a data center application.

Customer segment

While there are many potential use cases that can tolerate periods of undefined behavior, like gaming, financial trading, and supply chain management systems, we decided to focus on live video streaming as our main use case. Live video streaming is experiencing extremely high growth, with companies like YouTube Live, Restream, IBM Cloud Video, Facebook Live, Instagram Live, LinkedIn Live, Twitch, Brightcove, Vimeo and DaCast dominating the industry.

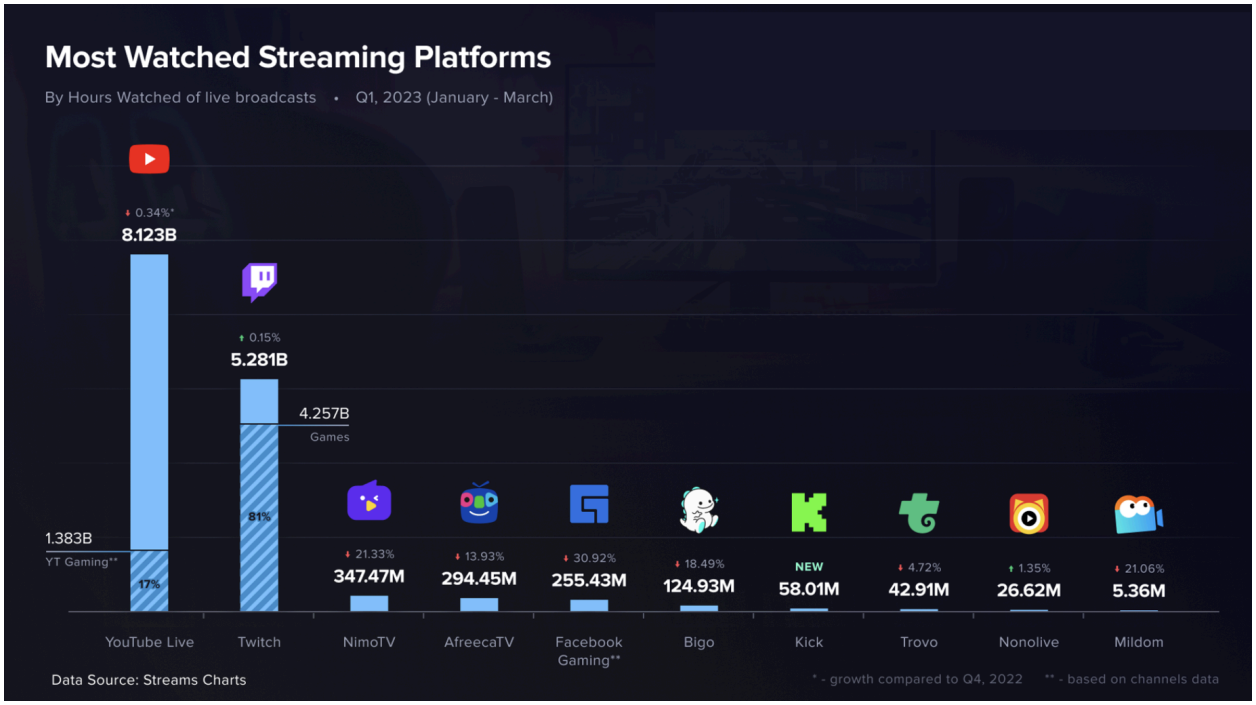
Most of these companies do not actually own their own data centers—rather, they outsource compute work to cloud providers, like AWS, Azure, and Google Cloud Platform. These providers have services specialized for data streaming, like Amazon Kinesis and Azure Stream Analytics. While, to our knowledge, there are no data centers built specifically for streaming applications, as live video streaming continues to grow, so too will demand for specialized infrastructure and services catering to data streaming applications.

That is why our target customers are **cloud providers who want to improve their technology related to streaming services provision**. Once this infrastructure is improved our algorithm will provide significant security and cost savings that the cloud providers can pass onto their streaming customers.

Market Research

The global live streaming pay-per-view market is currently valued at $1.38 billion in 2023, and is expected to reach $2.46 billion in 2027, which implies a CAGR of 15.5%. In addition, the market which includes both on-demand and pay-per-view live streams is expected to have a CAGR of 21.2% until 2027, growing to more than $3 billion by then.

The following chart shows the amount of hours of live streams watched on the respective platforms. In addition, Youtube Gaming streams grew by 13% on Q1 2023, and now accounts for 17% of total hours watched of live videos. Thus, in the first quarter of this year there were more than 14 billion hours of live content watched.



Some of the growth drivers of Live Streaming include the increase in the adoption of e-learning, the growth in the video game market and streams of gaming content and an increase usage of live streams as marketing tools.

For these reasons, we believe that live video streaming is the first use case we should focus on, with our initial target customers being cloud providers looking to provide compute power specialized for live streaming platforms.

Stakeholders

Key sets of stakeholders are the live streaming platforms, the streamers who publish on these platforms, and viewers themselves. The benefits of the BTR algorithm rely on the ability to tolerate short, infrequent periods of undefined behavior, in exchange for a high degree of resource saving. However, we need to make sure that this exchange is beneficial for the platforms and the people using them and that the bounded-time recovery guarantee would still be strong enough for live streaming to occur without much visible interference on the user's end. Thus, we hope to make operations more efficient for streaming applications while still making their users, the streamers and viewers, satisfied with the changes.

In addition, our project is research based; more specifically we are drawing heavily from research done by Professor Phan and we are working closely with her. Thus, Penn and the SEAS department are one of our most important stakeholders. If we were to take this project to market, Penn would certainly be involved as it owns the research our product is based on. Therefore, we would need to discuss profit share or royalties we would have to pay back to Penn and understand what other concerns of theirs we would need to meet.

Competition

According to our research, we seem to be the first group to try implementing any bounded-time recovery algorithm in a data center setting (as opposed to cyber physical systems). Our algorithm is also proprietary and has been developed relatively recently. Both of these factors give us a competitive advantage over other groups trying to implement the same type of algorithm.

We see Amazon Web Services and other cloud providers as our strongest competitors. These companies likely have internal teams focused on developing fault tolerance algorithms specialized for the design of their data centers. Besides needing to be tailored specifically to our customers' data centers, another downside to our solution is the high switching costs associated with moving to a new fault tolerance algorithm.

However, we are confident that the huge benefits of our algorithm in terms of resource-saving far outweigh these negatives. We also believe that our product is not easy for competitors to replicate in-house due to the hard work and skill of our team of SEAS researchers and our first-mover advantage.

It is important to note that an ideal solution would be for us to collaborate with or be acquired by AWS or another cloud provider so we can implement our algorithm on their data center solutions for live streaming. This partnership would be the most ideal because it would allow our solution to be optimized for the specific architecture it runs on, allowing our client to reap the full benefits of the resources efficiency of BTR.

Cost Model

Our main cost drivers are personnel related expenses. For initial steps, we plan to hire 5 engineers to finish implementing the algorithm as well as improving it, customizing it to the clients, and helping them with any installation process. Besides that, costs will include marketing expenses related to customer acquisition and retention and traveling costs for physical implementation. These costs should amount to between $36k to $40k per month, with employee's salaries being 90% of it.

Revenue Model

Our business model works as a subscription service. We offer services to any data center provider that would like to use our algorithms. The services include an installation and customization of the algorithm to their own infrastructure, as well as any maintenance or special request needed throughout the use of our system.

We will charge a fixed fee of $3,000 to be paid upfront for the installation and customization, and a monthly variable fee that will depend on time and usage of the system. The variable fee is $0.15 per thousand minutes, with a possibility of a 5% discount if our program is used on more than 50,000 servers, and a 10% discount if it's used on more than 100,000 servers.

With the previously estimated average monthly cost of $38,000, we need around 253 million minutes of video streaming per month with our system, which is very achievable. Mildom and Nonolive, the platforms in our market research with the fewer hours watched together average 640 million minutes per month. If we landed them as clients, it would generate around $96,000 without discounts, and $86,000 with a 10% discount per month. Our plan is to start with smaller players, make sure our program works smoothly, then move on to bigger platforms.