Module 3: Critical Thinking

Case Studies of Large Data Files

Nolan Byrnes

CSC507 – Foundations of Operating Systems

Colorado State University – Global Campus

Professor Joseph Issa

December 1st, 2022

Module 3: Critical Thinking

Data files can contain information from a couple rows up to billions of rows of information, and there are many use cases that require data to be efficiently inserted and retrieved in a fraction of a second. This paper describes three use cases that would require efficient data entry and retrieval of data from data files, as well as discusses ways to optimize processing times for data files.

In the stock market, we have market makers which buys and sells shares from investors who are looking to trade. One example is Citadel Securities, which is trading roughly 900 million shares a day (). When trades are coming into Citadel to either buy or sell shares, they must know what the current price of each stock is at the second that the trade is being executed. Furthermore, for a trade to occur, they must know which account a bought stock would be coming from. With so many shares being traded throughout the day, and the frequency on the price changes of the stocks being traded, it is important to have all this information up to date within fractions of a second, to ensure all trades are being executed at the price that the end user is wanting to buy/sell a stock for.

There are flight tracking websites that “provide real-time and historical access to flight information, including aircraft details, routes, position, speed, altitude and heading” (Hayward, 2022, para. 1). These websites such as Flightradar.com can acquire all this information about planes by using Automatic Dependent Surveillance-Broadcast (ADS-B) satellite-based tracking (Hayward, 2022, para. 3-4).

ADS-B data is collected by ADS-B receivers and feeds the data to the tracking websites. Flightdata.com “claims to have a network of over 20,000 such receivers in place”, which are constantly feeding the website flight information. For these flight tracking websites have the latest information, they need to be able to update their data in fractions of a second because of the volume of planes that are flying, the speed at which they are flying, and the frequency of planes entering and leaving the skies.

When shopping in a brick-and-mortar store, it is easy to keep track of inventory that is available to purchase, however, in an online store, anyone could be ordering goods at any given time, and it is important to not have an online retailer sell more than what they have in stock. Online retailers must keep track of their inventory so that they can prevent customers from purchasing items that are still in stock. When a customer searches for goods on a retailers search engine, it must only display items that are available to sell. It would be important to keep a data file containing all the current inventory so that only the goods that are available to be sold are listed on their website. It is also important to process the inventory in fractions of a second, because of how many searches and purchase orders that are coming through at any given time. For example, Amazon receives 18.5 purchase orders per second (Thomas, 2021).

When you break down the steps of updating a data file, there are two actions that are taking place, which is acquiring the information to be written to the data file and writing the data to the file. To increase the efficiency of inserting this information, you could have multiple processes running concurrently that will gather the data that needs to be inserted into the data file and insert the data. To control access on which process could update the data file, semaphores could be used to allow only one process to update the data file at one time.

A semaphore is defined as “an integer value used for signaling among processes.” (Stallings, 2018, pg. 217). By using a semaphore, multiple programs would be able to know whether the data file is currently being updated by another process, or if there is availability to write to the data file.

Multithreading can also be used to improve the efficiency of updating the data file. Stallings explains that multithreading refers "to the ability of an OS to support multiple, concurrent paths of execution within a single process" (Stallings, 2018, pg. 151). In the context of the flight tracking websites, it takes time to receive the ADS-B feed, and it takes time to write to a data file. During the time that the running program is waiting for data to come in, or data to be written to storage, it can be working on another task. The program could send data to be written to storage, and while the program is waiting for the write to occur, it could be gathering data that is coming in. An analogy to multithreading could be a worker at a fast-food restaurant, where when they are waiting for people to make an order, they could switch their “thread” and cook food while they wait, thus improving their efficiency.

There are many use cases that would require data to be processed in fractions of a second, such as keeping up with inventory on an online store, keeping track of availability and prices of stocks, and flight data. By using multithreading, programs can switch their thread while waiting for an input or output action to complete, like waiting for data to be received and writing data to storage. By using concurrency, you can have multiple processes working to gather data, and keep a limit on how many of those processes could be writing to a data file at one time.

By using semaphores, you can limit the amount of processes that could update the file at a given time to one, allowing the processes to continue to gather data until there is availability to write the acquired data to the data file. By using multithreading or concurrency, processing times for these large data files can be optimized to make sure that data sets are updated in the least amount of time.

**REFERENCES**

Farrell, M. (2013, July 8). The computers that run the stock market. CNNMoney. <https://money.cnn.com/2013/07/08/investing/stock-market-citadel/>

Hayward, J. (2022, March 15). How do flight tracking websites work? Simple Flying. <https://simpleflying.com/how-do-flight-tracking-websites-work/>

Stallings, W. (2018). Operating systems: Internals and design principles. Pearson.

Thomas, M. (2021, October 1). How many orders does Amazon get every second, minute & hour? QuerySprout. <https://querysprout.com/how-many-orders-does-amazon-get-every-second-minute-hour/>