Problem Set 3

15-440/15-640 Distributed Systems Spring 2020

Assigned: Thursday March 26, 2020
Due: Thursday April 2, 2020 at 11.59pm

Submission procedure:

- Create a .pdf of your answers and upload to Gradescope. All enrolled students should have Gradescope accounts. Ask on Piazza well in advance of the deadline to be added to Gradescope if you don’t have access; it is unacceptable to ask to be added immediately before the deadline.

- Here are some tips to make your submission easy to read and to grade. Remember, the easier you make this, the less likely we are to make grading errors. Following these guidelines will help us to focus on the technical content of your answers rather than trying to understand what you have written.
  - Use Latex rather than handwriting your answers. A Latex template can be found on the course web page.
  - Put the answer to each question on a separate page.
  - Carefully tag your pdf pages to each question on gradescope. You can use the SHIFT key to select multiple pages and associate them with a single question.
  - As an absolute last resort, if you must hand write your answers, make sure that you create the .pdf using a real scanner. Images taken with a camera (e.g., your smartphone or tablet) are not acceptable. Illegible submissions will receive a zero grade.

- Remember that there are no grace days for problem sets. Submit on time.

- Assume SI notation. In other words:
  - “KB” → $10^3$ bytes, “MB” → $10^6$ bytes, “GB” → $10^9$ bytes
  - “Kbps” → $10^3$ bits per second, “Mbps” → $10^6$ bits per second, “Gbps” → $10^9$ bits per second
  - but a byte is still 8 bits (not 10 bits!)
Question 1 (25 points)

You work for Zooom University, the hottest online university in 2020. Zooom allows client applications to connect to several frontend servers, each of which is connected to the database (DB) server. Your application is written such that the frontend servers can scale out, but not the database. Zooom relies heavily on AWWS cloud service to rent servers or purchase new servers. AWWS’s pricing model is explained in the table below. Assume that a purchased server has a service life of 1000 days, servers can be rented at 1 hour intervals, and the frontend server and database server must run on separate machines.

<table>
<thead>
<tr>
<th>Size</th>
<th># cores</th>
<th>Memory</th>
<th>Speed (GFLOPs)</th>
<th>Cost to buy</th>
<th>Cost to rent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>4</td>
<td>8 GB</td>
<td>500</td>
<td>$2000</td>
<td>$0.20/hour</td>
</tr>
<tr>
<td>Medium</td>
<td>4</td>
<td>32 GB</td>
<td>500</td>
<td>$4500</td>
<td>$0.45/hour</td>
</tr>
<tr>
<td>Large</td>
<td>8</td>
<td>24 GB</td>
<td>750</td>
<td>$4500</td>
<td>$0.45/hour</td>
</tr>
<tr>
<td>XLarge</td>
<td>16</td>
<td>32 GB</td>
<td>1000</td>
<td>$9000</td>
<td>$0.90/hour</td>
</tr>
<tr>
<td>Ultra</td>
<td>128</td>
<td>1024 GB</td>
<td>8000</td>
<td>$100000</td>
<td>$10.00/hour</td>
</tr>
</tbody>
</table>

Zooom has benchmarked their application. Their DB server can handle 400 transactions per second (tps) per core. The following table is the result of their benchmarking for the frontend service. Use this data to predict the resources needed for higher workloads.

<table>
<thead>
<tr>
<th>Number of concurrent students on a server</th>
<th>Memory used</th>
<th>Compute needed</th>
<th>DB transactions per second (tps) needed</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.5 GB</td>
<td>25 GFLOPS</td>
<td>10 tps</td>
</tr>
<tr>
<td>4</td>
<td>4 GB</td>
<td>100 GFLOPS</td>
<td>40 tps</td>
</tr>
<tr>
<td>8</td>
<td>6 GB</td>
<td>200 GFLOPS</td>
<td>80 tps</td>
</tr>
</tbody>
</table>

A. For the initial deployment, you consider using four small frontend servers and a medium server for the DB server. Calculate the number of students that can be supported assuming the following resources as the bottleneck: (i) the DB machine, (ii) Compute on frontend servers, and (iii) memory of frontend servers. What is the bottleneck resource in this system? How many concurrent students can be supported? Show your work.
B. The system needs to handle 200 concurrent students. What is the cheapest way of doing this? You may assume this capacity is needed indefinitely. Show your reasoning and clearly state how many of each type of server you will need to rent/purchase for each tier. For example, you may use tables similar to those shown below in showing your reasoning.

**Frontend Server(s):**

<table>
<thead>
<tr>
<th></th>
<th>Ultra</th>
<th>XLarge</th>
<th>Large</th>
<th>Medium</th>
<th>Small</th>
</tr>
</thead>
<tbody>
<tr>
<td># Purchase</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rent (# machines x hours/day)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Total cost per hour:

**Database Server(s):**

<table>
<thead>
<tr>
<th></th>
<th>Ultra</th>
<th>XLarge</th>
<th>Large</th>
<th>Medium</th>
<th>Small</th>
</tr>
</thead>
<tbody>
<tr>
<td># Purchase</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rent (# machines x hours/day)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Total cost per hour:

C. On further inspection, it turns out that use of the application shows a distinct diurnal pattern:
What is the lowest cost way of deploying servers to handle this load on an ongoing basis? Again, as in part B, show your reasoning and clearly state how many of each type of server you will need to rent/purchase for each tier.

Frontend Server(s):

<table>
<thead>
<tr>
<th># Purchase</th>
<th>Ultra</th>
<th>XLarge</th>
<th>Large</th>
<th>Medium</th>
<th>Small</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rent (# machines x hours/day)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Total cost per hour:

Database Server(s):

<table>
<thead>
<tr>
<th># Purchase</th>
<th>Ultra</th>
<th>XLarge</th>
<th>Large</th>
<th>Medium</th>
<th>Small</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rent (# machines x hours/day)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Total cost per hour:
D. Under what circumstances (if any) would the expensive Ultra server be needed? If this was the case, how could you avoid using such an expensive server? Answer in 2 - 3 sentences.

Question 2 (12 points)
You are a research scientist at Zoom. Observing a dramatic increase in online education using Zoom software, you feel the need to develop some machine learning (ML) model that summarizes students' real-time reaction, so that instructors can adjust their lecturing pace accordingly.

A. You need a powerful GPU to train the ML model fast enough. There are two options:
   Option 1: Buy a GPU, which costs $10,000 and lasts 3 years. There is an extra $1 operational cost per day of use.
   Option 2: Use a cloud service, AWWS, and rent an instance with a powerful GPU, which costs $2/hr.
   a. Assume that training is run 12 hrs per day, and there are 30 days in a month. How many days should the project run so that Option 1 and Option 2 cost the same? Please show your work.
   b. Give a reason why Option 1 can be preferred over Option 2, and a reason why Option 2 can be preferred over Option 1 (one sentence for each reason).

B. You got some users' permissions and have their data stored, so that you can use the data to improve your model. As the dataset becomes larger and larger, you need more storage. There are two options:
   Option 1: Buy some SSDs, each with 1TB (=1000GB) storage and cost $500.
   Option 2: Use AWWS cloud storage, which costs 0.1 per GB per month.
   a. You have 2.2TB data and you need to store it for 6 months, which option is cheaper? Please show your work.
   b. You choose option 1 and store the data for 6 months. After 6 months, a global crisis bursts out and you have to extend the time of your work. You realize that you need to store your data for another 6 months. At the same time, all your SSDs start failing due to a manufacturing defect. You can either choose to repair them or completely turn to AWWS for the rest of the time. Repair of SSDs takes one month and you will have to
store the data to AWWS during time of repair. What criteria on the repair cost will make you want to choose to repair the SSDs? Please show your work.

**Question 3 (24 points)**

Consider the 3-tier Web system shown below, which is similar to what was discussed in class. You are using this for a powerful disease spread information website, where people in Pittsburgh can record who they've been in contact with to track the spread of a disease. Given that this disease propagates very quickly, people always want to get the latest information. The DFS stores static information such as past news announcements addressing this disease. The MySQL database stores dynamic information about Pittsburgh’s contact history. The App Server overlays a graph of contact history on a map to display to clients.

A. In each tier, servers may be configured with different amounts of disk (storage) and RAM (processing). Which tier would you want to have the most disk? RAM? Justify your answer in one or two sentences. Clarify any assumptions you make for this. (Note: multiple answers may be acceptable.)

B. On the Monday after spring break, your website started to operate. As soon as it starts to function, there are 250 user requests arriving at your system per second regarding regional contact history. For simplicity, assume that all requests arrive equally spaced in time. You have 120 tier-1 servers (i.e., static content servers) and 120 tier-2 servers (app servers). 80% of the requests are for browsing the information, and take 0.15s for an app server to process. 20% of the requests are recording new contact histories. 90% of the recording requests can be processed by an app server in 1.5 seconds, but their distribution is long-tailed: 9% of these requests take 5.0 seconds to process, and the remaining 1% take 10 seconds to process. You may ignore the time for processing by tier-1 servers (static content servers). You may also ignore network delays and assume
that no requests are dropped. Assume load is equally distributed among the application servers.

Please show your work for all computations.

a. Under these assumptions, what will be the queue length in front of each tier-2 server (i.e., an app server) 5 minutes after the sale starts?

b. How many more app servers do you need to have a queue length of zero at this rate? You may assume that there is no queue at the start.

c. When the queue length is zero, what is the median, mean, and p95 of an arbitrary user request?

C. You try to find ways to improve the response time for recording requests and found that it was actually the lack of database computation power causing the long-tailed distribution of recording request service time. Aiming to solve this problem, your friend suggests dynamically scaling out the database across multiple machines. You recalled that this might be a bad idea from your 15440 class. What is the most important problem with dynamically scaling out the DB server? Tailor your answer to the database specifically rather than general issues with scaling out.

Question 4 (24 points)

Recall the protocol for live migration of a VM presented in slide 20 of the 2020-03-05-scaling2-rashmi slide deck. Two important metrics in this protocol are:

Total migration time: Time between the start of migration and when the VM is started on the target server

Downtime: Time between when the VM is frozen on the source server and started on the target server. During this time, a service running in the VM cannot process any requests.

A. A VM running a particular service needs to be migrated. Assume the following:
   - Migration starts with 1 GB of read-write data and no read-only data
   - The VM generates new dirty data at a rate of 1 GBps (gigaBYTES/sec)
   - Network bandwidth between the source server and target server is 16 Gbps (gigaBITS/sec)
   - Network delay is negligible and VCPU state is negligible in size; no components other than network transfer of read-write and dirty data contribute to total migration time
Suppose the service can tolerate 150 ms of downtime; higher downtime is not acceptable. What is the total migration time for this transfer? How much downtime is experienced by the service? You must show your work using a table like that shown below.

<table>
<thead>
<tr>
<th>Time since start of migration (ms)</th>
<th>Amount of data to transfer (GB)</th>
<th>Time it will take to transfer data (ms)</th>
<th>Amount of state dirtied during round (GB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0ms</td>
<td>1GB</td>
<td>500ms</td>
<td>0.5 GB</td>
</tr>
<tr>
<td>500ms</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

B. Explain whether each of the following changes would generally cause total migration time to increase or decrease when migrating a VM that generates new dirty data at a constant rate (but at a rate less than the network bandwidth). Explain your answers.

a. Increasing network bandwidth

b. Increasing the maximum tolerated downtime for a migration

C. A service running in a VM currently generates new dirty data at a rate of 1 GBps. The service needs to be scaled to handle 5x the load that it currently supports, which would result in a 5x increase in the rate at which the underlying VM generates new dirty data. You are considering the following options for scaling this service:

Vertical scaling: Run the service in a VM on a server that can handle 5x the load. This single VM will now generate new dirty data at a rate of 5 GBps.

Horizontal scaling: Run the service on 5 VMs, each on their own server that can handle the original load. Each VM continues to generate new dirty data at a rate of 1 GBps.

Assume that the network bandwidth available to a single server in either setup is 50 Gbps (i.e., the single vertically-scaled server has the same network bandwidth as a single server in the horizontally-scaled scenario). Further assume that the tolerable downtime for a single VM is the same in either scenario.

a. Which form of scaling is more likely to result in fewer total VM migrations required for the service? Explain your answer. Hint: it may help to remember what can
trigger a VM migration.

b. Which form of scaling is more likely to result in lower total migration time for a single VM in the service? Explain your answer.

**Question 5 (15 points)**

For each of the following scenarios, describe which encapsulation method you would use: VM, container, or UNIX processes. In a situation where you would use a VM, make sure to specify whether to use Type-1 or Type-2 virtualization. Justify your answer in one or two sentences.

a. Steve is a student at Carnegie Mellon who has a lot of unused machines. After learning about virtualization in his distributed systems class, he sees the potential to capitalize on the demand for computing resources. So, in order to make some money on the side, he decides to start a service offering his compute resources for rent. His clients need to run computationally intensive programs that are specific to Windows and Ubuntu operating systems.

b. Alex just joined Hooli as a remote consultant software engineer and is informed that he needs to develop a feature for the messaging system that runs smoothly on both macOS and Windows 10. He realizes that he needs to test his code on both the operating systems. Alas, he only has a 1 machine with Windows 10 and does not have the capability to access any other machines.

c. Amy is working on a cloud-based distributed bitcoin miner where each node handles a set of miners. She wants to use checkpointing to recover the full-system-state of her distributed system. In case of a failure, she wants to resume all of the nodes using the recovered system state from the latest checkpoint.

d. Amy realizes that her bitcoin miner is too slow and so, this time around she no longer wants to fully recover any nodes that fail in her system. Instead, her design will now just spawn a new node once her central coordinator recognizes that a node has failed. **NOTE:** She has all the nodes running on the exact same OS and wants to solely focus on improving the restart time after a failure.

e. Instead of working on a cloud-based system, Amy decides that she will simply use her personal 28-core computer to spawn multiple miners where each individual miner is single-threaded. She will be the only one who has access to this system. Her goal is to maximize the performance of her bitcoin mining system.