Brian Engel

Memory and Storage Management

With the Gaming Room’s application, Draw It or Lose It, we must manage 200 high-definition images that are approximately 8 MB a piece. Obviously, we could download all the images to the client before starting the game, but that would be a huge waste of time, storage, and memory. It would be much more efficient to send the client the image as needed. For the first image you would have to create an object and load it in the heap before the round started. Then the round would play and once it was done you would want to delete the image’s object so it wouldn’t stay in the memory. Then you are free to load the second image object into the heap. If you don’t take the time to delete the image objects once they are no longer needed, the memory will fill up quickly with a large or long game, slowing down performance more than it needs to.

The way that I would deal with storage is to use cloud storage for all of the images, a normal server or cloud server for the actual program, and then the client side would just have to be able to run the web browser and have minimal storage. This way, using cloud storage, you can expand the application to use as many images as needed, and the cost will scale on the amount of images stored. Then you can download those images one at a time or in a small group (such as the amount you would need for a single game) to the server, and then send them one at a time to the client from the server. This way the server and client aren’t required to store large amounts of images at any one time. The reason cloud storage can’t be used on the client side is to prevent cross site scripting. Any time a user is able to input something in a web browser, the input needs to be “cleaned” to make sure it doesn’t include any malicious script in the text.

The difference between memory and storage in terms of this application is mainly that memory will determine how fast the application will run with short term data, and storage is how the application handles long term data. Memory has more to do with the heap and the objects that the application creates and are volatile, which clears if you turn your computer off such as RAM. Storage is long term, such as the image files, the application files themselves, and any other files that it needs to run, like the operating system. These files are non-volatile and stay in place even when the power is off, such as ROM, HDD’s, and SSD’s.