

The Singleton design pattern WAS going to be used for the Database object. As the Database stores vectors of all processed records, there should only ever be one instance. Having one instance ensures all data is stored in the same location, allowing for better management and control of data. The Singleton pattern would have allowed us to ensure only one database instance ever exists and that we are not reading or writing data to somewhere else.

The above highlights what was coded for the instance method for Database.

However, due to time constraints and the uncertainty surrounding changing references to a fundamental object near the deadline of a project caused us to ultimately scrap implementing the singleton design pattern. While it could have been helpful in managing data, development was too far along for such changes to be made with ease and without extensive debugging. In the end, we chose to stick with the existing database constructor in the interest of time and delivering a product that was tried and true throughout development.