At the top of the concept map, I drew the data sources. Data can come from anywhere. Some examples are Facebook posts, an email, pictures, text messages, etc. Then, through online transaction processing, they are fed to either a relational database or a NoSql database. Relational databases are fed structured tabular, structured data while NoSql typically collects unstructured data. Some examples of unstructured data are text files, emails, voice recordings or videos. Relational databases, compared to NoSql databases, handle smaller amounts of data; however, this data is consistently updated according to the daily operations of the application.

Once this data is collected, any information needed for historical purposes are placed in the Data Warehouse. SQL developers create and modify both the relational database and the data warehouse. Once the data is stored, members of the business intelligence teams can collect data to create visualizations and models to drive company decisions. These members are usually data scientists or data analysts. Last, the Database Administrator/Architect and Database Developer are not connected to any part of the chart. That’s because they are typically involved in most of the process. A database architect manages a database while it is being designed while database administrator helps manage it once it’s developed. Thus, they can be involved in any area of the chart. Database developers are typically the members creating the database hands on. They follow the vision of the Architect. Thus, they can be important in any step.

Third, I believe it’s important to understand the 3 V’s and which type of database, relational or NoSQL can handle them. First, volume, describes the amount of data while velocity and variety describe the speed and structure of it, respectively. Large datasets require a lot of memory that many relational databases can’t handle efficiently. More data may require the data to be split across different nodes of a database server. With the typical tabular format, it would be computationally expensive to gather all the information when requested. However, with NoSQL, all related information is aggregated together; therefore, data that was once in the same database can be safely partitioned across different nodes. When discussing variety, the relational database fails. They work well when the data is in a tabular format; however, they fail when faced with semi-structured or unstructured data such as a video recording. Last, NoSQL can handle higher velocity data compared to relational databases. Because the data is normalized and typically requires joins to find all information of an instance, relational databases are slower at taking in data. On the other hand, NoSQL databases aggregate all the related data together. Therefore, when writing data, it doesn’t have to search across many tables; it can place all data in the same area. No wonder NoSQL is becoming more popular.