

**Short essay: Data vs. Information - Select a database in use today (real or imagined) and identify the elements of “data” stored therein and describe how the database organizes the “data” into “information”. Give contrasting examples of “data” and “information” that illustrate the meaninglessness of “data” without context and organization. Talk about the value the “information” provides once the component data is given context.**

While abroad last semester, I was fortunate enough to intern at The Up Group, a digital executive search firm in London. The Up Group is Europe’s leading digital search firm, and hopes to branch out as a leading digital search firm worldwide in the near future. That being said, the company has loads and loads of data it collects every day to turn into information for them to expand and deliver the best results to their clients. Their database, Invenias, holds all of their clients, candidates, and hundreds of thousands of C-level employees’ information from around the world. As an intern, I would research CEO’s and their companies and input the data in by creating a “profile” for the CEO and said company. I would input names, addresses, job descriptions, past experience, and many more details for each profile. Everything I entered was *data* – it had no meaning on its own other than being cold, hard facts. In turn, the employees of The Up Group would go on Invenias to find out information about a certain company, sector, person, etc. They could type in a company name, and Invenias would sort out who the C-level employees were, lower level employees, the average job salary at different job levels throughout the company, and much, much more. Invenias turned the facts I was inputting into the system to mean something – it turned it into *information*. All my data was analyzed, trends were formed, and conclusions were drawn to answer any questions employees had when they were conducting searches.

Without Invenias, all the data I collected would have just been a collection of facts. The CEOs’ experiences, job salaries, job levels, and the rest of the data I found would not prove anything nor help the employees of The Up Group when they were conducting searches. For example, a C-level salary at Company 1 located in the U.S. does not help find the experience and skills needed to find a VP candidate at Company 2 located in the UK. Combined and analyzed in the database, data came out as information that proved trends or certain conclusions that The Up Group could not otherwise find. Especially since there is so much data that is collected and inputted into Invenias every day, the employees of The Up Group could not analyze it on their own in an efficient manner or within an efficient period of time. Because the database does all the work for them, they are able to study information quickly and thoroughly, which is key for producing optimum results for client searches.

**Short Essay: Data Models - Briefly describe the hierarchical and network pre- relational data models. Explain their shortcomings in relation to the relational model. Considering this, what do you think of XML as a model for data storage?**

IBM's Hierarchical Model has it all in the name – they believed that data was a hierarchy. The data is constructed like a tree, with one piece of data leading to the next connected to one another by links. Like a flat file system, one piece of data represented a field, and fields were built together to represent a record. It is almost as if there are different branches of data that lead back to one, overarching collection of data. While the model seems organized at first, there are a couple flaws to it. First, the model requires that you know the structure of the data beforehand. While you are able to write programs to get information out of the data, if you completely change or restructure the model, you need to reprogram the data over again to get information out like you used to. Additionally, the Hierarchical Model duplicated data, making it inconsistent and a waste of space. As for the Network Model, there is not too much of a difference. It is still set up as a tree-like structure, with fields and records that represented data. What set apart the Network Model was the “networks” within it. Networks got rid of duplications, and were not as much a waste of space as the Hierarchical Model.

Compared to the Relational Model, the Hierarchical and Network models are inconsistent and flawed. The Relational Model focused around the concept of looking at data as if it were a table of rows and columns. In this sense, columns would act as fields, and rows would act as records. Columns and rows make up a table, which is the database. There are no duplicates, and data is organized in a way that can be filtered by attributes. There are also four different cardinalities in which the data can be separated: one-to-one, one-to-many, many-to-one, and many-to-many. One piece of data can be linked to not only another single piece of data, but to many other pieces of data, too. The Relational Model is perfect for getting complicated information that could not otherwise be generated by another model. Considering all this, I believe that XML is a very flexible way to store data. Because of its easy-to-read format, it is easily accessible for different languages and across the Internet.

