Data vs. Information: Data can be recognized as a set of numbers, but without any context behind them. Data becomes information, when it is given meaning or context such as Walmart's order fulfilment system. When an order is placed by a customer within the system, it will give information such as: Customer name, shipping address, billing address, quantity purchased, dollar amount purchased etc. If I were to purchase an order, all my information that I provide on their website will be sent to their database, where all my "information" will be provided to whoever is filling and shipping my order. Meaningless data in this scenario could be seen as my quantity purchased of say two items or the number "2" in their database. Without my complete order information behind the quantity, it is merely just a number of products in which are waiting to be shipped. These "2" items are in Walmart's inventory system, until I provide all of my relevant information regarding proof of purchase, shipping address and so on, to then push my order to a member of their company who will fill my order and ship it to my location. All of these characteristics above are meaningless in this scenario until they are all pieced together to create one whole action or order in this case. Names, numbers and addresses can all be seen as "data" until they are given context behind them, which in this case would be the order that I have placed and the information I have given.

Data Models: The hierarchical model transposes data into a tree-like arrangement, in which the data is kept together through the use of records. These records of data are connected through links via a top-down structure to display said commonalities between each other and their values. The relational model involves the use of records and sets, which are then placed into tables within the database. The relational model uses a many-to-many relationship. The data stored within the columns and rows of the table each have their own unique identifier, but when compiled all together through the many-to-many relationship, this data becomes information that is tied together (ex: customer order). I personally do not believe that XML would be a beneficial use for data storage as its use is not meant to be a database, but as a means of pulling data from other various sources.