

Data vs. Information:

A database is a collection of information that is acquired over a long period of time. The Experian database is a credit risk score database that provides credit risk information and services to businesses wanting to inquire more about their customer's risk profiles. This database collects data about customers such as name, address, phone number, Intelliscore, financial stability risk score, SIC code, etc. Some of the data elements could include values like "Taylor Dunn", "14 Main Street", "903-495-0938", "73", "80", and "700". These different values are considered data because they are simplistic and seemingly random without supporting context. A database organizes this data into some sort of structured table, assigning it specific rows and columns that provide context as to what each value represents, and why it is relevant. This extra detail is what converts meaningless data into information of value that allows us to make decisions. For example, the value "73" provided above has no real impact then it being the number "73". It could be anything: 73%, 73 apples, 73 dollars, etc. When provided the context that "73" is the Intelliscore for Taylor Dunn, conclusions about Taylor Dunn's risk profile can be made, which can benefit a company when making decisions about certain customers. Another example could be the number "700". Without information regarding what "700" is, it is not possible to make any accurate assumptions about what that number represents. In the case of Experian, this "700" represents a Standard Industrial Classification code, determining this industry to be agricultural services.

Data that is given context has incredible value to companies. It can be used in a variety of ways, depending on the needs of the business. There can be internal and external data which

supplement each other to create profiles on customers, marketing efforts, understanding trends, determining areas of improvement within a company, and supporting many other vital business decisions. Data without the context to make it information can be very dangerous and misleading. It is imperative that context is provided to support this data and make it the most useful it can be.

Data Models:

Over time, different models have been created to view the structure of a database in an organized and consolidated format. One model, known as the hierarchical model displays a database in a family tree style, reading from top to bottom. This specific model cannot have cycles. It allows for multiple one to many relationships, with one “parent” at the top having many children. The network data model is a more flexible way of representing data, and is typically used to manage large, shared data. There is more connection between different entities, and no one entity is at the top, as in the hierarchical model. The relational data model consists of tables that contain rows and columns. These multiple tables are all related through join statements. A major advantage of the relational data model is that it is easy to query from, since SQL was created around it. The hierarchical data model allows for repetitive information to be stored in different entities, which is a cardinal sin when it comes to database management. Duplicate data is a waste of space, and the redundancy weakens the context. The relational data model uses the separate tables to ensure that data is not repetitive. The network data model allows for many to many relationships which is typically avoided when developing a database. The relational data model avoids this by creating two or more tables in place of many to many relationships.

The use of smaller documents, tagged as XML has also been referred to as a database model. I think that the relational data model is better for sorting larger amounts of data, because XML is typically used with collections of small documents. XML has a hierarchical structure, so if the data has that type of structure already, then XML may be the way to go.

Resources used:

http://databasemanagement.wikia.com/wiki/Category:Hierarchical_Data_Model

<https://www.techwalla.com/articles/advantages-disadvantages-of-network-databases>

<https://gadgetopia.com/post/4293>

<http://dbmsenotes.blogspot.com/2014/03/comparison-of-data-models-data-models.html>

pgAdmin III:

I think I am going to try using pgAdmin 4



