Raymond Mattingly

Professor Alan Labouseur

Database Management

September 9th 2016

Data vs. Information

A good example that differentiates between data and information is in stock market analysis. I developed software that takes in data from the Yahoo Finance API and analyzes it to find undervalued companies. I assume that the data in the YH API exists in a database. The metrics that my program draws are last sale price, 52 week high, 52 week low, 1 year target price, PEG ratio, dividend yield, 200 day moving average, EPS, and the EPS estimate for the next quarter. Without context and proper labeling, the return of this data would be as useless as any string of random numbers, but with proper labeling and organization one can take the data and properly extract useful information. Such useful information could be which companies are likely to see growth in the future, which are currently undervalued, which are currently overvalued, etc..

Take Google for example. One can extract the following data from Google's stock quote: 771.46, 789.87, 589.38, 921.08, 1.24. Without context that extraction of data would be useless. With the context that those numbers, respectively, represent last price, 52 week high, 52 week low, target price, and PEG ratio one can extract the information that Google is confident in its ability to grow, has seen good growth this year, but could be overvalued based on their price per earnings and growth rate ratio. In other words, pieces of data are just numbers until given proper context which can lead to information, the analysis of said data and context.

Raymond Mattingly

Professor Alan Labouseur

Database Management

September 9th 2016

Data Models

The hierarchical model organizes data in a tree like structure. This allows for one-to-many relations in a parent-child relationship. The hierarchical model was eventually replaced for its shortcoming in that there could be significant data repetition among child data segments.

The network model came in handy over the hierarchical model when data modeling called for more than one parent for child data segment. The network data model essentially works as a set does with many pairs of data segments within it, which can then be turned into information using principles like union and intersection. A shortcoming of this type of data modelling is that systems can become extremely complex and difficult to edit efficiently.

XML would not necessarily be my first choice for a data storage model as it's hierarchical and sequence oriented, so you would need to deal with having things in a particular order and the potential for repetitions. With a relational data model you don't have to deal with either of those issues.

Raymond Mattingly

Professor Alan Labouseur

Database Management

September 9th 2016