Database

A database, simply speaking, is a collection of data maintained according to a pre-defined structure. It serves as the backbone of any system and requires a lot of research and analysis before it can be created. Database design falls under the design phase of the systems development life cycle and thus follows directly after the analysis phase. Once the informational needs and flow of the system have been finalized, a database is designed keeping in mind the needs of the system and the way it will support the system at the back end.

Databases more or less follow a general rule of design in most cases. The preliminary stage is of analysis of the existing environment where the system will operate in order to determine:

- Data requirements
- Classification of information
- Relationship between information
- Transactions among data entities
- Rules of identity

Once this is done, database design then steps into the design stage. There are two aspects to database design, the data model and the functional model. The data model is the actual data and how it is managed. The functional model on the other hand decides how this data is supposed to be accessed, processed, queried etc. In order to design the data model, the following steps have to be followed, taken from the Information Technology Services page of the University of Austin (2004):

Identification of data objects and relationships: this part is analyzed in the analysis stage and data entities are given meaning through the flow of information in the system.

Drafting the initial ER diagram with entities and relationships: here, the relationships between various data entities are established and documented using an Entity-Relationship diagram.

Refining the ER diagram: this ER diagram is further refined by removing ambiguities, repetitions, etc.

Add key attributes to the diagram: in continuation of the previous step, the ER diagram is further enhanced along with incorporating key attributes of data entities (class diagrams). These attributes are what will set each entity apart from the other and serve as identifiers. Adding non-key attributes: non-key attributes are extras and may also be repeated in various entities.

Diagramming Generalization Hierarchies: here, the diagram is generalized with hierarchies in terms of security and data relativity are set.

Validating the model through normalization: the database is normalized so as to avoid any discrepancies and logical ambiguities when transactions are conducted between data entities. Many normal forms exist, which one to decide upon depends solely on the requirements of the system.

Adding business and integrity rules to the Model: this is where the database becomes more than just a collection of data. It becomes meaningful and useful at this stage once the business perspective is put in.

Among these steps, normalization of a database is the most important step after the ER diagram has been designed. The planning and analysis stage is one part that will determine how the end system will be (Davidson 2007). This is because keys and relationships between data entities are simplified through normalization which then helps in designing a better functional model. Again, the degree of normalization depends on what is required (Ullman, 2003).

For the case at hand, users usually are confused as to how a system actually works and whether it would serve their needs. Since the proposed model will be a relational database model, all tables within the database will be interlinked one way or the other using a primary-foreign key infrastructure. The best way to convince such an audience is to involve them in the whole data analysis and the systems analysis process. This way, they will have a say in whatever goes into the system and expect what they ask for thus solving the political aspect of handling such situation.

On the technical side however, end users can be shown a demonstration of a 'prototype' database where data entered by the purchasing manager pertaining to raw materials bought becomes available for the operations manager to use on his system. Once the operations manager is done and makes the appropriate entries, the marketing manager can then view how many units of a certain product are available for accepting orders from customers and meeting sales targets. The personnel managers work is interlinked since employment records will be kept and appropriate people will be assigned to different departments or projects within the organization. The operations manager will enter the amount worked by a certain employee

along with the marketing manager entering how many sales targets were met by certain sales personnel. All this information will help the personnel manager in calculating salaries etc.

References:

Davidson, L. (2007). Ten Common Database Design Mistakes. Retrieved, May 30, 2007, from http://www. simple-talk. com/sql/database-administration/ten-common-database-design-mistakes/

The University of Texas at Austin (2004). Introduction to Data Modeling. Retrieved, May 30, 2007, from http://www. utexas. edu/its/windows/database/datamodeling/dm/design. html

Ullman, L. (2003) MySQL Database Design. Retrieved, May 30, 2007, from http://www. peachpit. com/articles/article. asp? p= 30885&rl= 1