**CSC263 Database Systems**

**Homework Assignment #1**

**Steven Moody**

**Introduction**

**Answer the following questions. You should use class textbook as a reference for your answers.**

1. Define/Discuss each of the following terms:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| data | database | database system | database catalog | program-data independence |
| user view | DBA | canned transaction | meta-data |  |
| deductive database system | persistent object | data model | database schema | database state |
| internal schema | conceptual schema | external schema | DDL | DML |
| query language | host language | catalog | client/server architecture | three-tier architecture |

**Data:** Known facts that can be recorded and have an implicit meaning. This can be anything and within any topic.

**Database:** A collection of related data.A good basic example of this is an excel sheet recording information about students within a class.

**Database System:** The DBMS software together with the data itself. This is rather self-explanatory but it is the combination of both a Database with the software used to “define and manipulate” said database.

**Database Catalog:** From my understanding a database catalog is almost like a table of contents”. This will list basic information such as “the structure of each file, the type and storage format of each data item, and certain constraints on the data.”

**User View:** This is the view the end user sees. This is external and is based on whatever group that particular user is interested or has access to. These individuals are usually querying or updating the database itself.

**DBA:** This acronym stands for (Database Administrator), their primary role is to determine who has access to the database as well as monitoring the usage and security of the database.

**Canned Transaction:** This is a “standard type of query” that is used by Naïve end users and this query has been heavily tested and specifically programmed well in advance to the query.

**Meta-Data:** Meta-data is the detailed information about the data within the database. Our book defines Meta-data as “The database definition or descriptive information stored by the DBMS in the form of a catalog.”

**Deductive Database System:** A deductive database System is one in which the database itself is capable of “deducing” certain information based on what is stored within the database via rules given to the DBMS.

**Persistent Object:** These are objects within the DBMS that are able to be retrieved even after a program termination occurs. This is sort of the basis for why Object-oriented DBMS is beneficial.

**Data Model:** A Data Model is an organized display of the elements within a database, showing the structure of said database as well as showing the relationship between them.

**Database Schema:** A Database schema is a detailed description of a database.

**Database State:** A Database State is a display of the current data within a database at any given time.

**Internal Schema:** The Internal Schema is a description of how the databases storage is structured. This includes how the data is stored and whatever paths it may take for retrieval.

**Conceptual Schema:** The conceptual Schema is a description of the database as a whole entity and what data is within the database, without hiding said data rather only hiding the storage structure and access paths.

**External Schema:** An External Schema is multiple descriptions of each of the views various users will see, as well as the contents within those views, each of these views may be different so various amounts of data are shown per view.

**DDL:** DDL stands for Data Definition Language, and this is used for the schema definitions . This language allows for those behind the scenes to help create the structuring of the Database

**DML:** DDL stands for Data Manipulation Language, these are commands that allow users to perform tasks associated with using the database.

**Query Language:** Query Language is the language used to perform queries on a database. This is a subset of DML and is used as a term for the commands used to perform queries in a “standalone manner”.

**Host Language:** Host language, similarly to above is the other side to DML, this is a term used when DML commands are placed within a programming language for use.

**Catalog:** A Catalog contains the descriptions of the data within the database it belongs to. This includes things like data type, size etc.

**Client/Server Architecture:** Client/server architecture is a type of DBMS architecture in which clients access a server in order to access certain data within that server. This is typically used in large settings where groups of people require access to the same data/files.

**Three-Tier Architecture:** A three-Tier Architecture is a type of client/server Architecture that is broken up into three tiers; The Client tier, App/Web server tier, and the Database Server Tier. This is the preferred type of architecture as it allows a higher distribution of work-load as well as added security.

1. What is data independence, and why is it lacking in file systems?

**Data Independence is the concept of being able to change one schema without having to modify the others. This allows one level to be modified without having to change how a program accesses data. This is lacking in file systems because based on how a file system works, data dependence is required.**

1. What is structural independence, and why is it important?

**Structural Independence is achieved when changes can be made to the “structuring and constraints(page 38)” of a database without affecting how or through what programs users interact with the data within the database. This is important as not having structural independence would mean that any time a structural change needed to be made, the programs used to access the database would also have to be modified.**

1. What is a DBMS, and what are its functions?

**DBMS stands for Database Management System, this is a system that allows users to create databases. Its primary function is just that, giving users the ability to create and modify databases. The DBMS is also in charge of the storage device that holds the Database. A DBMS also allows users to share the database.**

1. What is the role of a DBMS, and what are its advantages? What are its disadvantages?

**As stated above, the role of a DBMS is to provide the software platform for users to create and modify as well as share Databases. A few of the reasons listed in our book for the advantages of a DBMS include the ability to provide added security through requiring users to log in to the Database as well as limiting what users are able to access, the ability to enforce rules and constraints across all user groups using the database Controlling redundant information by allowing all information to be stored in one location on one database that can be modified to allow for each user group to have the view and access to data necessary, and lastly it allows for much better storage options over traditional file processing. The disadvantages include high costs of using a DBMS and creating databases as well as the costs incurred to keep it operational, for certain projects there may be requirements that limit the usability of a DBMS, and lastly small databases may not require the use of a DBMS.**

1. What are the main components of a database system?

**Based on figure 1.1 in our book, I believe that the main components of a Database System are the DBMS, the Database, the applications, the storage, and the users.**

1. What are metadata? Give examples of metadata in a relational database. Give examples of metadata in a NoSQL database.

**Metadata is defined as the detailed information of the data within the database. This is usually done in a catalog in relational databases and an example is given on page 11 of our book. They have a database that contains basic school information and use a catalog (a structure within the database giving detailed information about the data) to describe how many columns each Relation\_name has as well as what the names of each column are, the type of data it is, and which Relation\_name the column belongs to. In NOSQL databases, meta data is not required and thus the meta data is stored along with the data it is giving information about. For example, using the same topic of database as earlier in this question, instead of using a catalog, I believe a noSQL database would just have information about the Relation\_names right there in the same location as the data they represent.**

1. What are some basic database functions that a spreadsheet cannot perform?

From a broad standpoint, it would seem spreadsheets lack a couple of functions that databases have. First, a database has relationships amongst the various tables within is. Second, databases I would think are far more secure in terms of the rules and constraints set up within them. A spreadsheet cannot really enforce rules within its sheets. Lastly, a database also contains information about its data in the form of meta data, which unless manually input a spreadsheet cannot do.

1. What are the different types of database end users? Discuss the main activities of each.

**Our book explains that we have 4 types of end users. The first is “Casual End Users”, these are individuals who will only have to access the database from time to time and are primarily using very detailed queries. The second is “Naïve/ Parametric End Users” and these are the folks who will be accessing the database often and will be running similar queries over and over. The third type of end users are the “sophisticated End Users”, this category consists of those who are going to be developing programs to access the database based on their specific job roles. The last type of end users are the “standalone Users” and these are users who will be using programs that already exist to access specific and small amounts of data to build their own databases.**