Database Manager (DBM):

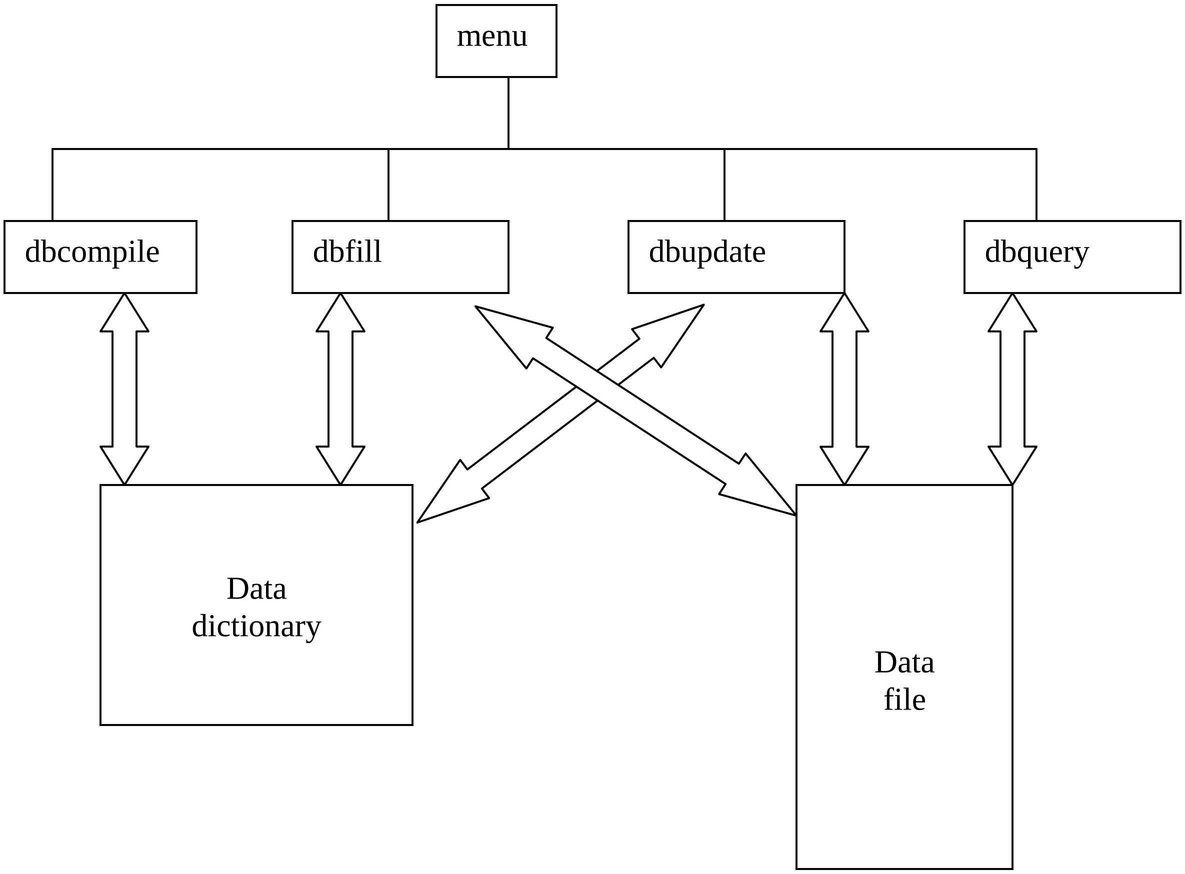
Your Database System will consist of 2 main files :

1. the data dictionary
2. the data files

and 4 main programs :

1. the data dictionary compiler
2. the data base filler
3. the update/delete program
4. the query processor

All these 4 programs will be tied together by a menu program :



The Data Dictionary :

This is a data base about the data base . It contains the specification of the data base record :

- the name of the data file
- the names of the fields within the data base record
- the number of bytes that each of these fields will occupy in the main data file.

An example of the compiled data dictionary:

```
personal.dat
first_name
20
last_name
30
street
20
city
15
..
```

Any access of data in the main file must first go to the DD to find the storage format. It must also get from the dictionary the names of the data fields for display to the terminal screen.

The data base dictionary begins as a source file. This file lays out the requisite information in a way more appropriate for the user. An example of a DD source file :

```
CREATE TABLE personal(
SSN                 NUMBER(10) Primary key,
FIRST_NAME          VARCHAR2(20),
LAST_NAME           VARCHAR2(20),
STREET              VARCHAR2(20),
CITY                VARCHAR2(15),
STATE               VARCHAR2(2),
..
)
```

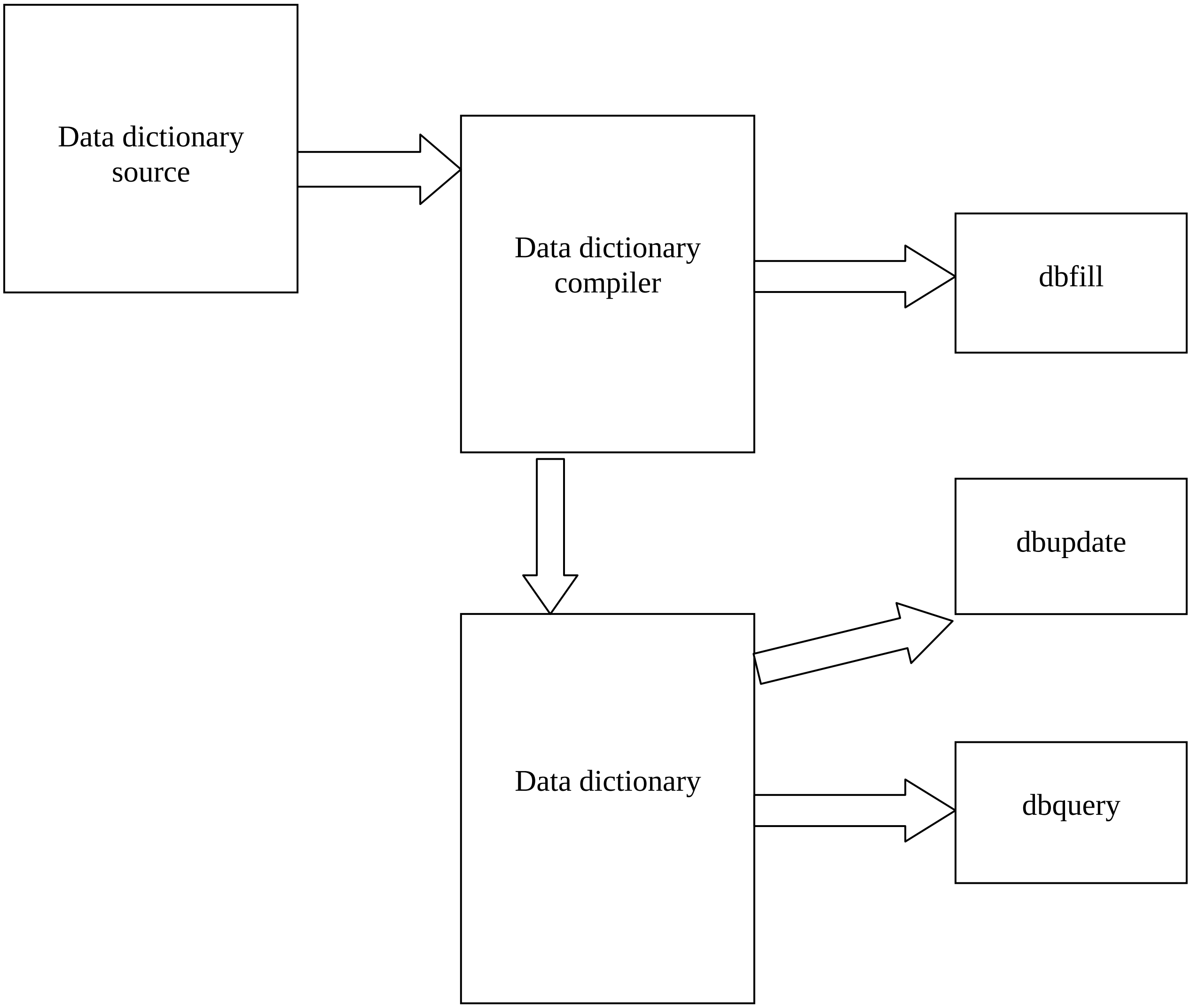
Such a file is an ordinary text file produced by any convenient editor.

It is the job of the DD Compiler program to read though the source file and produce the DD. The compiler is translating the statements in this language into a form that the other programs in the data base system can utilize.

All the integrity constraints of a Database System must be adhered to. (Primary Key, Foreign keys) . The required concepts will be explained in class.

The file where the actual data values are stored has no inherent structure at all. It is , in fact, one long string of characters. All format information lies solely in the DD file.

Creating a DD:



To store and access the data dictionary, you must use the data structure of either hash tables or linked lists .

Functional Decomposition :

Major subsystems of the DBM :

Subsystem	**Description**
dbmenu	This is the user’s entry point into the database system. It will display the available functions, accept the user’s choice, and call the appropriate program. At the end of each subsystem, control will be returned to this program.
dbcompile	The function will accept as input the data dictionary source file, a text file produced outside the data base system. It will produce the compiled data dictionary file as output. The entries in the source file will be transformed into a form usable by the database system. This function will automatically create the data dictionary file (the DD file must have an extension of dic) . If a file by that name already exists, it will be overwritten.
dbfill	The function will allow the user to add new entries to a created database. It will accept as input the name of the data dictionary and the values for each database record from the user; it will put a new record at the end of the data file. In the case of a new database, this function will automatically create the data file.
dbupdate	This function will allow the user to change the values in an individual data base record or to mark the record as deleted. It will take as input a search value used to find the desired record and will produce changes to the data base’s data file.
dbquery	This function will accept from the user a query request and will display the requested information from the data base. Input will be a program written in the data base system’s query language. (Eg: SELECT name FROM EMP WHERE age >20;

The statements that go as input to dbfill and dbquery must have the format of SQL statements.

All error messages must be stored separately in a library called yourFirstName_error_messages and accessed from this library.