COSC 3380
Design of Database Systems

Introduction

January 24, 2024

Typical DBMS Functionality

- ***Define*** a particular database in terms of its data types, structures, and constraints on the data to be stored
- ***Meta-data***
  - Database definition or descriptive information
  - Stored by the DBMS in the form of a database catalog or dictionary
- ***Construct*** or ***Load*** the initial database contents on a secondary storage medium
- ***Manipulating*** the database:
  - Modification: Insertions, deletions and updates to its content
  - Retrieval: Querying, generating reports
  - Accessing the database through Web applications
- ***Processing*** and ***Sharing*** by a set of concurrent users and application programs – yet, keeping all data valid and consistent

Additional DBMS Functionality

- DBMS may also provide –
  - **Protection** or **Security measures** to prevent unauthorized access
  - **Active processing** to take internal actions on data
  - **Presentation** and **Visualization** of data
  - **Maintenance** of the database and associated programs over the lifetime of the database application

Application activities against a Database

- Applications interact with a database by generating
  - **Queries**: to access different parts of data and formulate the result of a request
  - **Transactions**: may read some data; 'update' certain values or generate new data, and store that in the database
- Applications must not allow unauthorized users to access data
- Applications must keep up with changing user requirements against the database

Example of a Database

- **Mini-world** for the example:
  - Part of a UNIVERSITY environment
- Some mini-world ***entities***:
  - STUDENTs
  - COURSEs
  - SECTIONs (of COURSEs)
  - (Academic) DEPARTMENTs
  - INSTRUCTORs

Example of a Database

Mini-world:

- Part of a UNIVERSITY environment
- Some mini-world ***entities***:
  - STUDENTs
  - COURSEs
  - SECTIONs (of COURSEs)
  - (Academic) DEPARTMENTs
  - INSTRUCTORs

Mini-world *relationships*:

- SECTIONs *are for specific* COURSEs
- STUDENTs *take* SECTIONs
- COURSEs *have prerequisite* COURSEs
- INSTRUCTORs *teach* SECTIONs
- COURSEs *are offered by* DEPARTMENTs
- STUDENTs *major in* DEPARTMENTs

Snapshot of the UNIVERSITY database

COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	04	King
92	CS1310	Fall	04	Anderson
102	CS3320	Spring	05	Knuth
112	MATH2410	Fall	05	Chang
119	CS1310	Fall	05	Anderson
135	CS3380	Fall	05	Stone

GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

Characteristics of the Database Approach

- **Self-describing nature of a database system**
  - A DBMS **catalog** stores the description of a particular database
    - Data structures, types, and constraints
    - **Meta-data**
  - Allows the DBMS software to work with different database applications

A simplified database catalog

RELATIONS

Relation_name	No_of_columns
STUDENT	4
COURSE	4
SECTION	5
GRADE_REPORT	3
PREREQUISITE	2

COLUMNS

Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
….	….	…..
….	….	…..
….	….	…..
Prerequisite_number	XXXXNNNN	PREREQUISITE

Note: Major_type is defined as an enumerared type with all known majors. XXXXNNNN is used to define a type with four alpha characters followed by four digits

Characteristics of the Database Approach

- **Insulation between programs and data**
  - Program-data independence
  - Allows changing data structures and storage organization without having to change the DBMS access programs

Characteristics of the Database Approach

- **Data Abstraction**
  - A **data model** is used to hide storage details and present the users with a conceptual view of the database.
  - Programs refer to the data model constructs rather than data storage details
- **Support of multiple views of the data**
  - Each user may see a different view of the database, which describes **only** the data of interest to that user.