

# CSCC43: Introduction to Databases

## Lecture Notes

Joshua Concon

University of Toronto Scarborough – Fall 2017

Pre-reqs are CSCB63 and STAB52. Instructor is Dr. Marzieh Ahmadzadeh. Check RateMyProf. If you find any problems in these notes, feel free to contact me at [conconjoshua@gmail.com](mailto:conconjoshua@gmail.com).

## Contents

<b>1</b>	<b>Thursday, September 7, 2017</b>	<b>2</b>
1.1	Definitions . . . . .	2
1.2	How do we store data? . . . . .	3
1.2.1	Traditional File System . . . . .	4
1.2.2	Database Approach . . . . .	5
1.3	Relational Data Model . . . . .	6
1.3.1	Advantages . . . . .	6
1.3.2	Cost and Risk . . . . .	7
1.4	Database Development Process . . . . .	7

# 1 Thursday, September 7, 2017

## 1.1 Definitions

**Data** Objects or events that could be recorded on a computer media. This includes stuff like an email, an address, a student identification number, etc.

However, there are two types of Data, Unstructured and Structured.

**Structured Data** They are mostly stored in a tabular format, have types including: date, character, numeric, alphanumeric.

**Unstructured Data** Data that must be analyzed to extract information from, does not already show information. Includes multimedia, maps, documents, pictures, voices.

**Database** An organized collection of logically related data that can be of any size or complexity. Example:

- Student ID, name, GPA, courses taken
- Patient name, doctor's ID, ward, date of admission
- Personnel ID, base salary, bonus, hours of work

Another Example:

ID	Course	Mark
100	CSCC43	A
100	CSCC44	B
100	CSCC52	A+
200	CSCC43	A+
200	CSCC62	B

**Database Management System (DBMS)** A software that sits between users and databases, responsibilities include:

- Allowing users to create, update, sort or retrieve data

- Enforcing the integrity of data (consistency, etc.)
- Improving performance when dealing with huge data or many queries
- Assuring the durability of the data
- Allowing data to be shared among different programs
- managing concurrent access by multiple users/processes

**Information** Information is data that has been process in order to increase user knowledge, could be data provided in a way that can be analyzed easily (i.e. has labels, maybe graphs showing relationships, etc.)

**Metadata** "Data about data". Data that describes the properties of end-user data and it's context, it helps the database designer understand what kind of data exists and what it means. Examples include name, type, length, allowable values, ownership (stewardship), source...

Example:

name	type	length	min	max	description	source
Course	Alphanumeric	8			Course ID	Academic Unit
Session	Integer	1	1	9	Session Number	Registrar
Semester	Alphanumeric	10			Semester & year	Registrar
Name	Alphanumeric	30			Student Name	Student IS
ID	Integer	9			Student ID	Student IS
Major	Alphanumeric	4			Student Major	Student IS
GPA	Decimal	3	0	4	Grade point average	Academic Unit

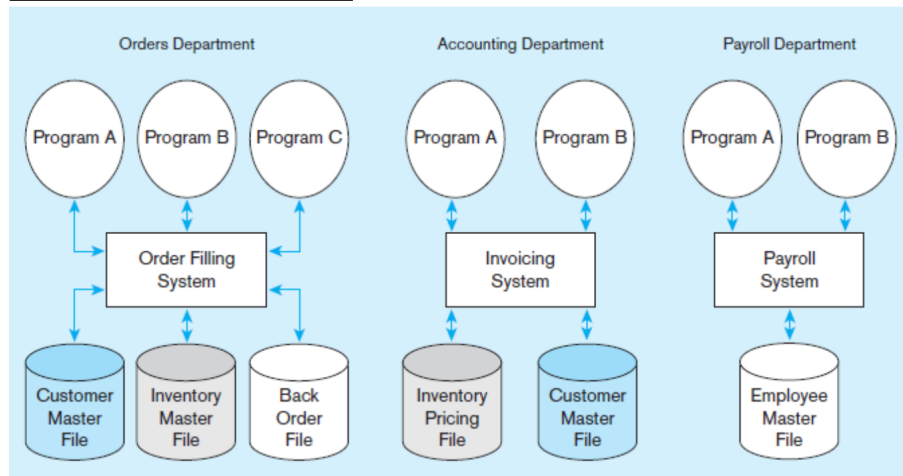
## 1.2 How do we store data?

There are two ways we can store data, either the Traditional file system or the database approach.

### 1.2.1 Traditional File System

The Traditional File System includes files that are also collections of related data, was used prior to the invention of databases, but has its own disadvantages.

Example of a file system:



Disadvantages:

**Program-Data Dependence** If the data is changes, all programs that used the data should be altered to. To use an example from the previous image, if the length of data such as customer name is changed, order filling system programs and invoice system programs are both affected. Using the database approach, this is managed by a Database Management System and no change in order filling or account programs are required, only on the file itself.

**Duplication of Data** As seen from the last image, the order filling system and the invoice system both have a customer master file. This is a waste of space as it is essentially two sets of the same data, and reliable metadata is difficult to establish since it may be inconsistent. Changes to one copy must also be applied to the other, so this may compromises the files' data integrity.

**Limited Data Sharing** Since each application has its own data, data sharing between applications is not easy, a system does not have access to another system's data. With the database approach, data sharing is doable.

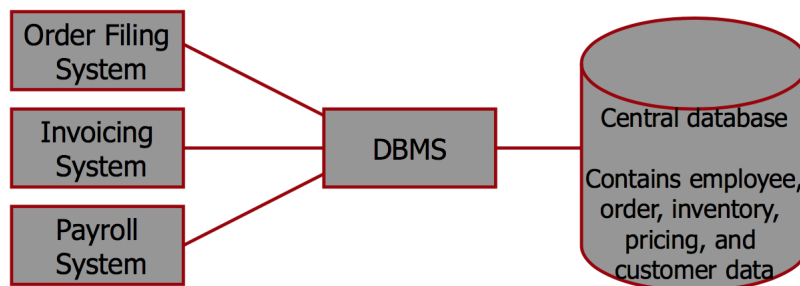
**Length Development Times** Each new application should be started from scratch by designing a new file format and description, however, in the database approach, you only need to design your database once, and can easily access the data you need.

**Excessive Program Maintenance** All preceding factors creates heavy program maintenance, so to avoid these disadvantages, the database approach is preferred. Note that if a database is not properly designed, mentioned disadvantages may not disappear.

### 1.2.2 Database Approach

The database approach creates a repository of shared data, data is managed by a controlling agent (Database Management System), and all data is stored in a standardized convenient form.

Visualization Example:



Every Database uses a data model, and this data model is a notation for describing data that specifies the structure of the data, the constraints on the content of the data, and the operations on the data.

Some examples of data models include a relational data model, a semistructured data model, unstructured data, and an Object-oriented model.

## 1.3 Relational Data Model

The main concept of the relational data model is a relation, which is based off the concept of relations in math. A relation in a database is like a table consisting of rows and columns.

Example:

Customer relation			Inventory relation			
Cust_ID	Cust_name	adr_city	Prod_Id	Prod_desc	quantity	price
100	Jane	Toronto	12	Sofa	20	1500
120	John	Waterloo	14	Chair	100	1250
130	Jade	Vancouver	15	Desk	55	650
143	Jack	Calgary	20	Armchair	100	700

### 1.3.1 Advantages

**Program-data independence** Metadata is stored in a central location called a repository, and data could be changed independent of what program uses it.

**Planned data redundancy** Data is only duplicated when we want it to be duplicated, so we can control redundancy.

**Improved data consistency** Which is implied since data redundancy is eliminated.

**Increased productivity of application development** Since the application does not deal with the data management, since it is taken care of by the Database Management System, so time and cost of development is saved.

**Enforcement of standards** Standards are enforced by the uniform procedures for accessing and updating data.

**Improved data quality** By defining constraints that the data must follow.

**Improved data accessibility and responsiveness** Can be done by using a query system to access data.

**Reduced program maintenance** Because of data-program independence, changing the data does not affect any program.

**Improved data sharing** By creating a user view and since all applications share the same data already, data sharing is doable.

### 1.3.2 Cost and Risk

**New specialized personnel** Companies need to hire new people for database design and database administration services.

**Installation and Management of Cost and Complexity** Implementation of the database and scaling it will be a new concern for companies

**Conversion Costs** Costs for converting legacy system to a modern database

**Need for backup and recovery** New costs and concerns for backing up the database and new services companies must worry about for recovery.

**Organizational Conflict** On data definition, data format and right to update data.

## 1.4 Database Development Process

**Model the rule of the organization** Such as, "each student at UofT can take up to 5 courses in one semester"

**Explore a relation between various entities** Such as, "A course might be taken by many students", "A student must take at least one course", etc.

**Create your relation** Creating the tables with relations between the entities and data associated with each object

**Normalize your relation** We will learn what this is later in the course

**Implementation**   Actual Implementation of the database