CS4.301 Data & Applications

Ponnurangam Kumaraguru (“PK”)
#ProfGiri @ IIIT Hyderabad

 pk.profgiri  /in/ponguru  @ponguru  Ponnurangam.kumaraguru

Protocol

Who am I?

- ~~Assistant~~ ~~Associate~~ Professor of Computer Science
- Ph.D. from School of Computer Science, Carnegie Mellon University (CMU)
- Research interests
  - Computational Social Science
  - Social (Societal) Computing
  - Privacy & Security in Social Media
- Courses I teach
  - Data & Applications (1), 4+
  - Online Privacy (1)
  - Privacy and Security in Online Social Media (8), 4+
  - Designing Human Centered Systems (5), 4+
  - Research methods / Advanced research methods (2), 4+
  - Foundations of Computer Security (5), 4+
  - Big Data & Policing (1), 4+

Who you are?

CND

ECD

EHD

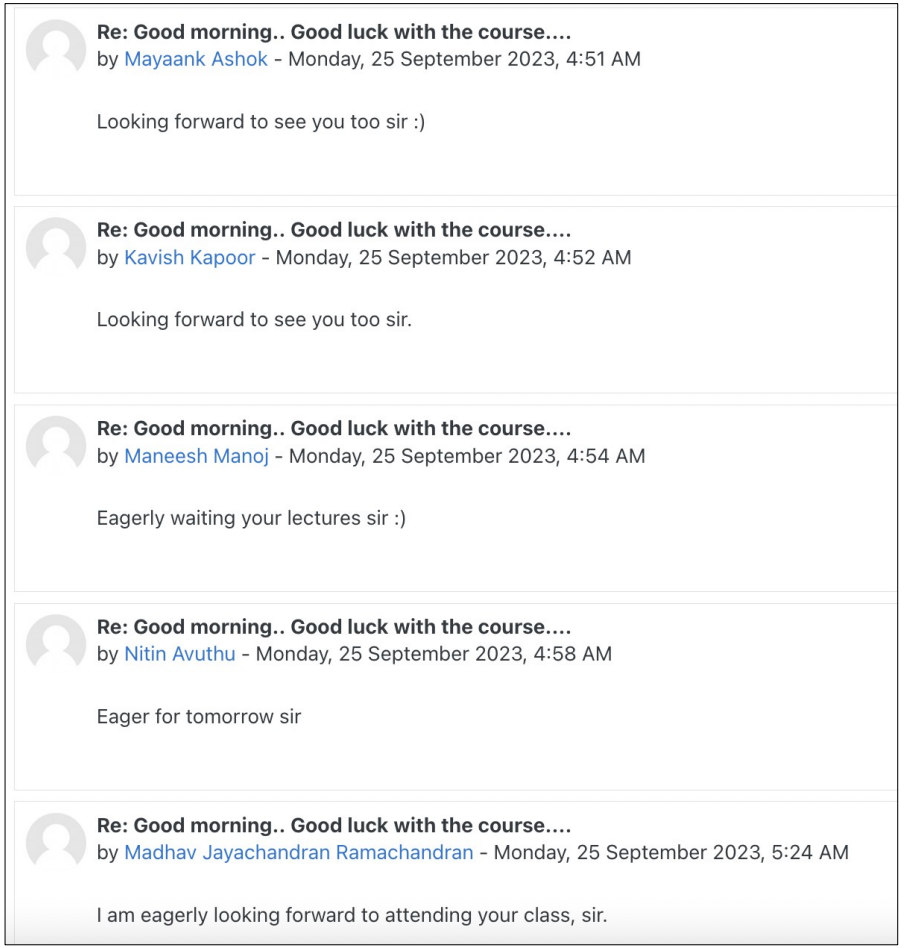
ECE

CLD

CSE

???

Impressed!!!



Re: Good morning.. Good luck with the course....
by Mayaank Ashok - Monday, 25 September 2023, 4:51 AM

Looking forward to see you too sir :)

Re: Good morning.. Good luck with the course....
by Kavish Kapoor - Monday, 25 September 2023, 4:52 AM

Looking forward to see you too sir.

Re: Good morning.. Good luck with the course....
by Maneesh Manoj - Monday, 25 September 2023, 4:54 AM

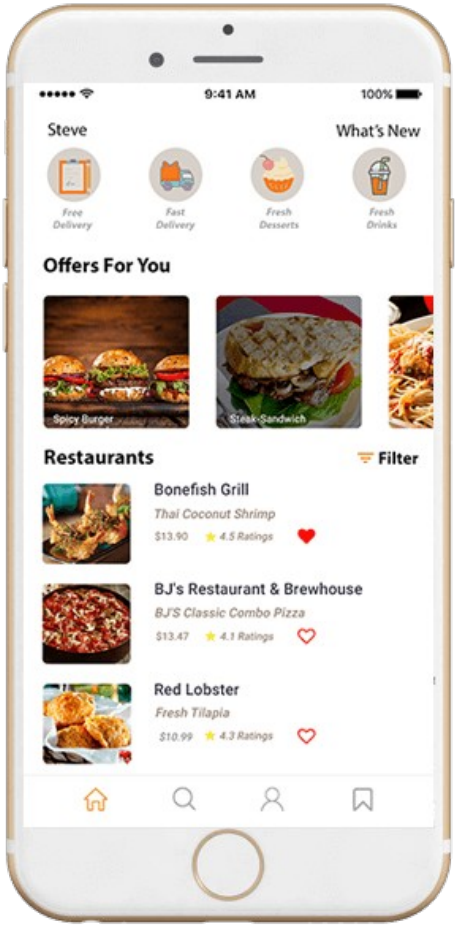
Eagerly waiting your lectures sir :)

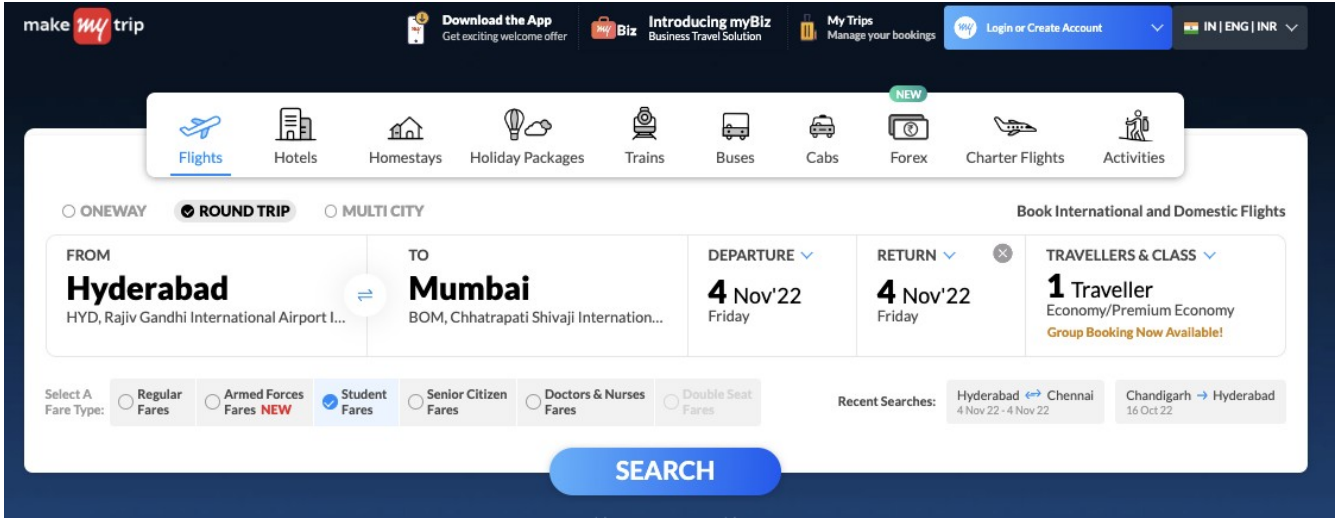
Re: Good morning.. Good luck with the course....
by Nitin Avuthu - Monday, 25 September 2023, 4:58 AM

Eager for tomorrow sir

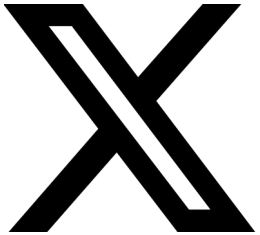
Re: Good morning.. Good luck with the course....
by Madhav Jayachandran Ramachandran - Monday, 25 September 2023, 5:24 AM

I am eagerly looking forward to attending your class, sir.





What kind of data are we generating here?



What kind of data are we generating here?

Grading, Relative

Type of Evaluation	Weightage (in %)
Class Quizzes (3)	15
Assignments (4)	20
Mid sem exam (Quiz-2 as scheduled in almanac)	15
Project	30
End Sem Exam	20

TAs

10 TAs

Students will be rotated among Tas for all evaluations

Akshit Sinha
Ankithvarun J
Ashmit Chamoli
Ayush Agrawal
Balamukumar Velayutham
Kavathekar Ishan Kishorkumar
Raghav Donakanti
Rohan Chowdary V Modepalle
Ronak Dhingra
Siddharth Ashwinkumar Mavani

Memory challenge for me ☺

Plagiarism

What is it?

Copying HWs

Any content taken from another source without citation

Whatever policy from IIITH

Moodle

We will use Moodle for all content sharing – slides, HWs, announcements, clarifications, etc.

Service Level Agreement

- Any question / clarification ask, if not urgent, will be answered in 24 hrs
- If anything urgent, feel free to attach the time in which you want the answer, we will try to respond
- TAs are your 1st point of contact only on escalation, you will being it up to me

Topics that we will cover

Relational Database Systems

SQL

Database design process

Data Models, Normalization

Soups

Cream of Tomato	165
Veg Clear Soup	165
Veg Hot & Sour Soup	165
Veg Corn Soup	165
Veg Silver Soup	165
Veg Cantonese	165
Veg Manchow	165

Starters - Chinese

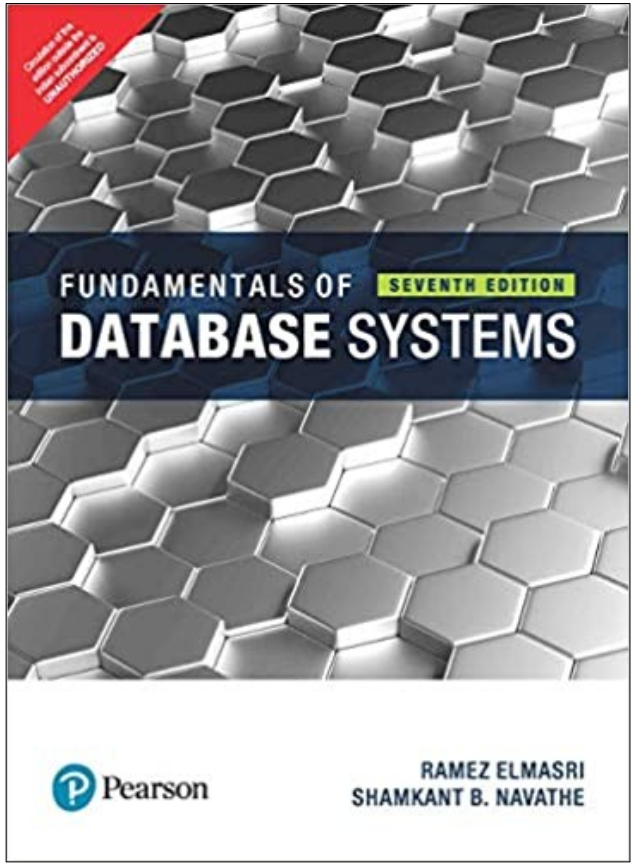
Crispy Vegetable	300
Veg. Gold Coin	300
Veg. Manchurian	335
Veg. Spring Roll	335
Gobi Manchurian	335
Chutneys Spl. Spring Roll	335
Chilly Mushroom	335
Mushroom Manchurian	335
Diced Paneer Red Pepper	335
Baby Corn Manchurian	335
Hong Kong Mushroom	335
Crispy Babycorn	335
Crispy Corn	335
Chilly Paneer	335
Paneer/Gobi/Aloo 65	335
Paneer Majestic	335
Chilly Mushroom	335

https://www.dineout.co.in/hyderabad/chutneys-madhapur-west-hyderabad-11747/menu

Lecture # from Alamanc	Class	Tutorial	Date	Day	Time	Topic	HW	Quiz	Project
15	1		25-Sep	M	0830 - 1000	Intro & data models			
		1	27-Sep	W	0830 - 1000	Data models & project			
			30-Sep		23:59hrs		HW1 publish		
		2	04-Oct	W	0830 - 1000	Data models & project			
16	2		05-Oct	Th	0830 - 1000	Intro & data models			
			08-Oct		23:59hrs		HW1 submission		
17	3		09-Oct	M	0830 - 1000	DB design & ER models	HW2 publish	Quiz 1	
		3	11-Oct	W	0830 - 1000	ER models			
			12-Oct	W	23:59hrs				Data requirements
18	4		12-Oct	Th	0830 - 1000	DB design & ER models			
19	5		16-Oct	Th	0830 - 1000	Relational DB			
		4	18-Oct	W	0830 - 1000	Relational DB	HW3 publish		
20	6		19 - 21 Oct			Institute Quiz			
			22-Oct	S	23:59hrs		HW2 submission		
			24-Oct	M	23:59hrs				ER Model
21	7		26-Oct	Th	0830 - 1000	Relational DB			
22	8		30-Oct	M	0830 - 1000	Normalization			
		5	01-Nov	W	0830 - 1000	Normalization			

23	9		02-Nov	Th	0830 - 1000	Normalization		Quiz 2	
			03-Nov	Th	23:59hrs		HW3 submission		
			05-Nov	S	23:59hrs		HW4 publish		Relational Database design
24	10		06-Nov	M	0830 - 1000	Normalization			
		6	08-Nov	W	1130 - 1300	SQL			
25	11		09-Nov	Th	0830 - 1000	SQL			
26	12		13-Nov	M	0830 - 1000	SQL			
			14-Nov	S	23:59hrs		HW4 submission		
		7	15-Nov	W	0830 - 1000	SQL			
27	13		16-Nov	M	0830 - 1000	SQL		Quiz 3	Application
28	14		20-Nov	Th	0830 - 1000	Revision			
			20-Nov			All marks check			
			21-Nov			Final project demo			
			23 - 30 Nov			End Sem			
			1 - 5 Dec			End Sem paper check			
			06-Dec			Grades to be submitted			

Book we will follow



Any questions / clarifications?

What do you want to know by end of Sem?

How many already use MySQL? Oracle?

Basic Definitions

Database:

A collection of related data.

Data:

Known facts that can be recorded and have an implicit meaning.

Mini-world:

Some part of the real world about which data is stored in a database. For example, student grades and transcripts at a university.

Database Management System (DBMS):

A software package/ system to facilitate the creation and maintenance of a computerized database.

Database System:

The DBMS software together with the data itself. Sometimes, the applications are also included.

What is a Database?

Data: factual (undoubted) information that can be recorded and have implicit meaning

A database is a collection of related data

What is a Database?

A database has the following implicit properties:

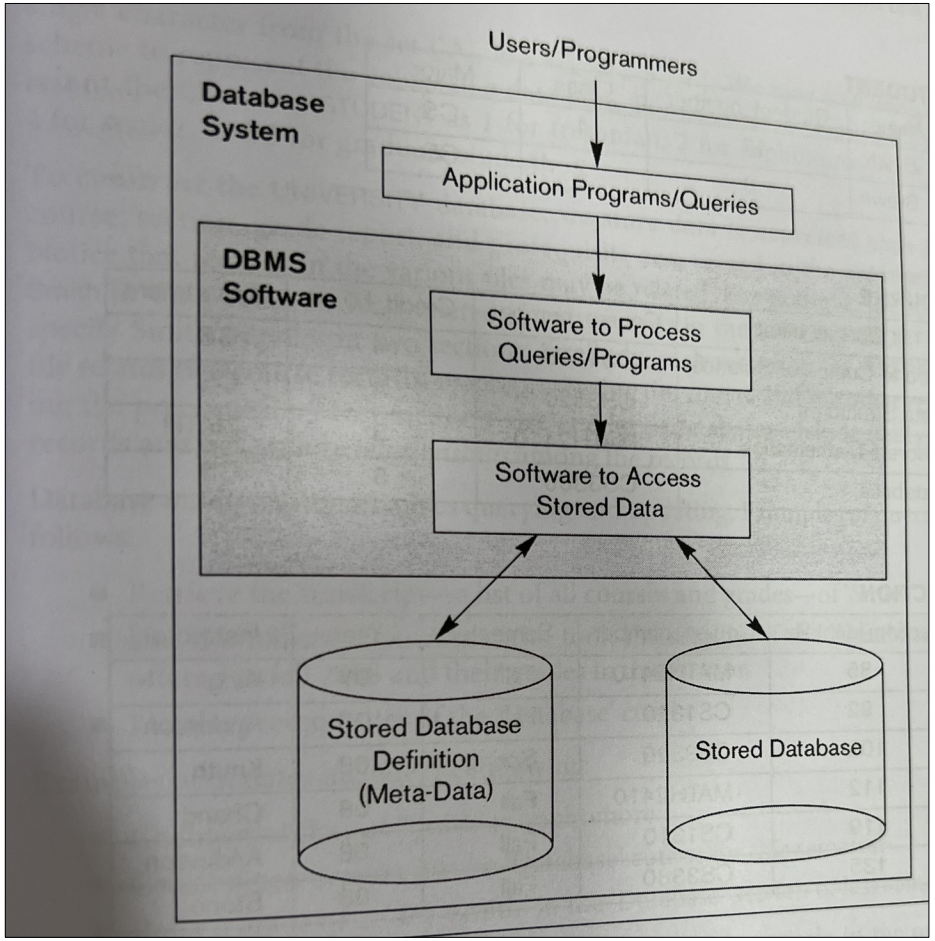
A database represents some aspect of the real world (mini-world or Universe of Discourse (UoD))

A database is a logically coherent (associated, related) collection of data with some inherent meaning

A database is designed, built and populated with data for a specific purpose

It has an intended group of users and some preconceived (already thought of) applications in which these users are interested

Simplified Database



Book

Example of Database: University

Data:

STUDENTs

COURSEs

SECTIONs (of COURSEs)

(academic) DEPARTMENTs

INSTRUCTORs

Relation:

SECTIONs are of specific COURSEs

STUDENTs take SECTIONs

COURSEs have prerequisite COURSEs

INSTRUCTORs teach SECTIONs

COURSEs are offered by DEPARTMENTs

STUDENTs major in DEPARTMENTs

Database

STUDENT			
Name	Student Number	Class	Major
Smith	17	1	COSC
Brown	8	2	COSC

GRADE REPORT		
Student Number	Section-Identifier	Grade
17	85	A
18	102	B+

PREREQUISITE	
Course Number	Prerequisite Number
COSC 3380	COSC 3320
COSC 3320	COSC 1310

COURSE			
Course Name	Course Number	Credit Hours	Department
Intro to CS	COSC 1310	4	COSC
Data Structures	COSC 3320	4	COSC
Discrete Mathematics	MATH 2410	3	MATH
Data Base	COSC 3380	3	COSC

SECTION				
Section-Identifier	Course Number	Semester	Year	Instructor
85	MATH 2410	Fall	91	King
92	COSC 1310	Fall	91	Anderson
102	COSC 3320	Spring	92	Knuth
135	COSC 3380	Fall	92	Stone

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 enumerated type with all known majors.
XXXXNNNN is used to define a type with four alphabetic characters followed by fou

Book

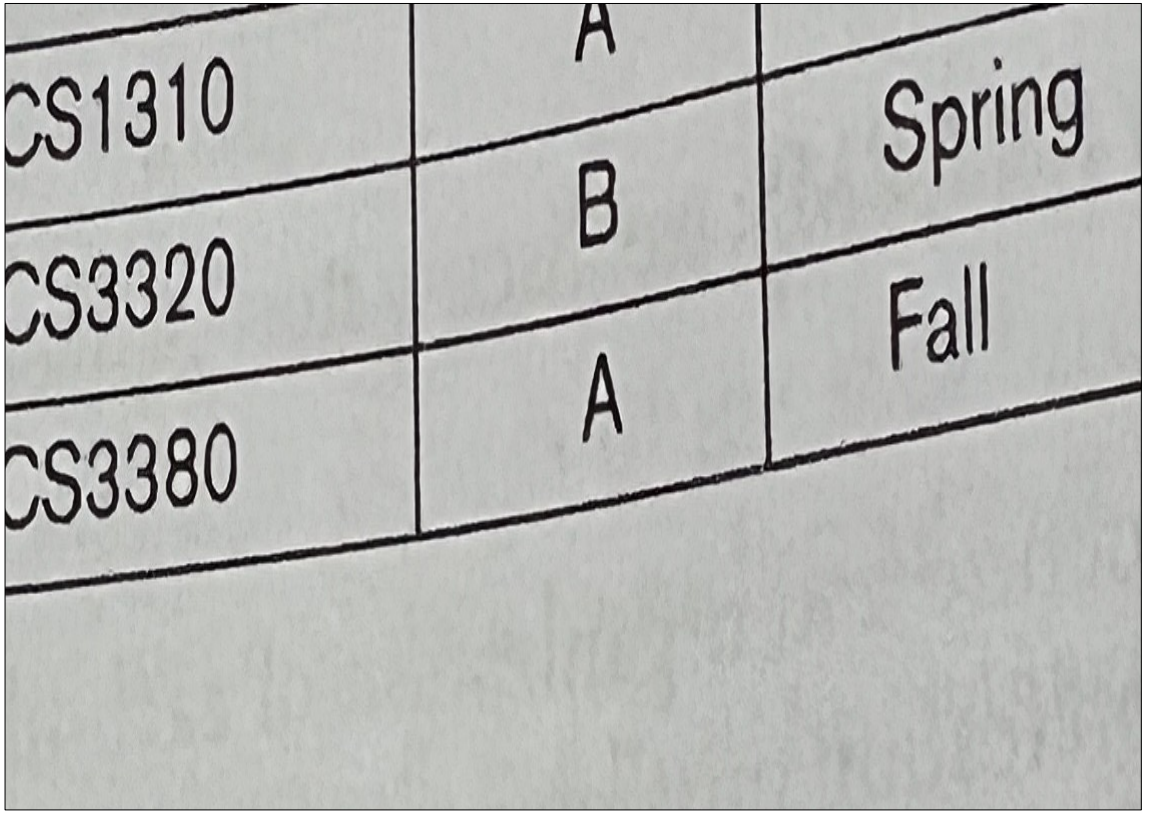
Database catalogue

Views

Many users to DB

Each users may require a different view

View may be a subset or virtual data derived



Online Transaction Processing (OLTP)

Multiuser DB

Concurrency control

Flight ticket booking, seats available

Transaction

Executing program or process that includes one or more database accesses, reading or updating of database records

Properties [ACID]

- Atomicity: either all are executed or none are executed [A/c A → A/c B]
- Consistency: any data written to a DB must be valid according to the defined rules [telephone number]
- Isolation: each transaction appears to execute in isolation, even though 100s may be executing at the same time [updating the seat preference]
- Durability: guarantees that once a transaction has been committed, it will remain committed even in the case of a system failure

Actors on the Scene: Day-to-Day use of DB

Database administrators

authorizing access to DB, coordinating & monitoring its use, accountable for security breaches & response time

Database designers

responsible for identifying the data to be stored in the DB, interact with potential group of users and develop *views* of the DB

End Users: Casual, naïve / parametric, sophisticated, stand-alone users

Casual: occasional users, typically middle or high-level managers

Naïve / parametric: constantly updating the db using *canned transaction,* done using mobile apps

bank tellers checking balances post withdrawals & deposits

reservation agents checking for availability

social media users post and read items on platforms

Actors on the Scene: Day-to-Day use of DB

End Users: Casual, naïve, sophisticated, stand-alone users

sophisticated: thoroughly familiarize themselves with all facilities of DBMS, implement their own, complex requirements

stand-alone: maintain personal DB using ready-made programs; TALLY

System analysts & application programmers

determine the requirements of end-users, including naïve, develop specifications for canned transactions

ap implement above specifications as programs, they test – debug – maintain these canned transactions

software developers / engineers play these roles sometimes

Workers Behind the Scene: Maintain the DB

DBMS designers & implementers

design and implement the DBMS modules; complex modules like query language processing, interface processing, controlling concurrency, handling data recovery & security

Tool developers

design & implement tools; optional packages that are often purchased separately; facilitate DB modeling & design, system design, and improved performance

Operators & maintenance personnel

responsible for running & maintenance of the hardware & software environment for DB

Advantages of using DBMS approach

Controlling redundancy

redundancy in storing the same data multiple times

e.g. student details in university maintained by acad & finance office separately

duplication of efforts, storage space, inconsistent data [Jan-19-1998 vs Jan-29-1998]

ideally student details in only one place, *data normalization*

keeping all needed data together, *denormalization*

Advantages of using DBMS approach

Restricting unauthorized access

your grades accessible to only some; my salary and personal details only to some [hopefully ☺]

Providing storage structures and search techniques for efficient query processing

efficiently executing queries & updates; creating *indexes* and maintaining it; *buffering* & *caching* modules

Advantages of using DBMS approach

Providing backup & recovery

provide facilities for recovering from hardware & software failures

complex updates, should not crash; if crash what state to recover

Providing multiple user interfaces

apps for mobile users; query language for causal; programming language for application programmers; forms / command codes for parametric; menu & natural language interfaces for standalone

Advantages of using DBMS approach

Representing Complex relationship among data

DBMS must have the capability to represent a variety of complex relationships among the data, to define new relationships as they arise, and to retrieve / update related data easily & efficiently

Enforcing integrity constraints

student name: 30 alphabetic characters; record in one file must be related to records in other files [e.g. every SECTION record must be related to a COURSE record] *referential integrity*

uniqueness on data item values [e.g. every COURSE record must have a unique value for COURSE_NUMBER] *key or uniqueness constraint*

Advantages of using DBMS approach

Permitting inferencing and actions using rules and triggers

- triggers associated with tables; trigger is a rule activated by updates to the table results in performing some addition operations to other tables, sending messages, etc.
- stored procedures are invoked appropriately when some conditions are met

Historical Development of Database Technology

Early Database Applications:

- The Hierarchical and Network Models were introduced in mid 1960s and dominated during the seventies.
- A bulk of the worldwide database processing still occurs using these models, particularly, the hierarchical model using IBM's IMS system.

Relational Model based Systems:

- Relational model was originally introduced in 1970, was heavily researched and experimented within IBM Research and several universities.
- Relational DBMS Products emerged in the early 1980s.

Historical Development of Database Technology (continued)

Object-oriented and emerging applications:

- Object-Oriented Database Management Systems (OODBMSs) were introduced in late 1980s and early 1990s to cater to the need of complex data processing in CAD and other applications.
  - Their use has not taken off much.
- Many relational DBMSs have incorporated object database concepts, leading to a new category called *object-relationa*l DBMSs (ORDBMSs)
- *Extended relational* systems add further capabilities (e.g. for multimedia data, text, XML, and other data types)

Historical Development of Database Technology (continued)

Data on the Web and E-commerce Applications:

- Web contains data in HTML (Hypertext markup language) with links among pages.
- This has given rise to a new set of applications and E-commerce is using new standards like XML (eXtended Markup Language). (see Ch. 13).
- Script programming languages such as PHP and JavaScript allow generation of dynamic Web pages that are partially generated from a database (see Ch. 11).
  - Also allow database updates through Web pages

Extending Database Capabilities (1)

New functionality is being added to DBMSs in the following areas:

- Scientific Applications – Physics, Chemistry, Biology - Genetics
- Earth and Atmospheric Sciences and Astronomy
- XML (eXtensible Markup Language)
- Image Storage and Management
- Audio and Video Data Management
- Data Warehousing and Data Mining – a very major area for future development using new technologies (see Chapters 28-29)
- Spatial Data Management and Location Based Services
- Time Series and Historical Data Management

The above gives rise to *new research and development* in incorporating new data types, complex data structures, new operations and storage and indexing schemes in database systems.

Extending Database Capabilities (2)

Background since the advent of the 21st Century:

First decade of the 21st century has seen tremendous growth in user generated data and automatically collected data from applications and search engines.

Social Media platforms such as Facebook and Twitter are generating millions of transactions a day and businesses are interested to tap into this data to "understand" the users

Cloud Storage and Backup is making unlimited amount of storage available to users and applications

Extending Database Capabilities (3)

Emergence of Big Data Technologies and NOSQL databases

- New data storage, management and analysis technology was necessary to deal with the onslaught of data in petabytes a day (10**15 bytes or 1000 terabytes) in some applications – this started being commonly called as “Big Data”.
- Hadoop (which originated from Yahoo) and Mapreduce Programming approach to distributed data processing (which originated from Google) as well as the Google file system have given rise to Big Data technologies (Chapter 25). Further enhancements are taking place in the form of Spark based technology.
- NOSQL (Not Only SQL- where SQL is the de facto standard language for relational DBMSs) systems have been designed for rapid search and retrieval from documents, processing of huge graphs occurring on social networks, and other forms of unstructured data with flexible models of transaction processing (Chapter 24).

When not to use a DBMS

Main inhibitors (costs) of using a DBMS:

- High initial investment and possible need for additional hardware.
- Overhead for providing generality, security, concurrency control, recovery, and integrity functions.

When a DBMS may be unnecessary:

- If the database and applications are simple, well defined, and not expected to change.
- If access to data by multiple users is not required.

When a DBMS may be infeasible:

- In embedded systems where a general purpose DBMS may not fit in available storage

When not to use a DBMS

When no DBMS may suffice:

- If there are stringent real-time requirements that may not be met because of DBMS overhead (e.g., telephone switching systems)
- If the database system is not able to handle the complexity of data because of modeling limitations (e.g., in complex genome and protein databases)
- If the database users need special operations not supported by the DBMS (e.g., GIS and location based services).

Any questions?

Bibliography / Acknowledgements

Instructor materials from Elmasri & Navathe 7e

pk.profgiri

Ponnurangam.kumaraguru

/in/ponguru

ponguru

Thank you for attending the class!!!

pk.guru@iiit.ac.in