SPATIAL DATABASE SYSTEMS LECTURE 1

FT OKYERE

DIFFERENCES BETWEEN A DATABASE MANAGEMENT SYSTEM AND A FILE-PROCESSING SYSTEM

1. BOTH SYSTEMS CONTAIN A COLLECTION OF DATA AND A SET OF PROGRAMS WHICH ACCESS THAT DATA. A **DATABASE MANAGEMENT SYSTEM COORDINATES BOTH THE PHYSICAL AND THE LOGICAL ACCESS TO THE DATA**, WHEREAS A FILE-PROCESSING SYSTEM COORDINATES ONLY THE PHYSICAL ACCESS.
2. A DATABASE MANAGEMENT SYSTEM **REDUCES THE AMOUNT OF DATA DUPLICATION** BY ENSURING THAT **A PHYSICAL PIECE OF DATA IS AVAILABLE TO ALL PROGRAMS** AUTHORIZED TO HAVE ACCESS TO IT, WHEREAS DATA WRITTEN BY ONE PROGRAM IN A FILE-PROCESSING SYSTEM MAY NOT BE READABLE BY ANOTHER PROGRAM.
3. A DATABASE MANAGEMENT SYSTEM IS DESIGNED TO ALLOW **FLEXIBLE ACCESS TO DATA** (I.E., QUERIES), WHEREAS A FILE-PROCESSING SYSTEM IS DESIGNED TO ALLOW PREDETERMINED ACCESS TO DATA (I.E., COMPILED PROGRAMS).
4. A DATABASE MANAGEMENT SYSTEM IS **DESIGNED TO COORDINATE MULTIPLE USERS ACCESSING THE SAME DATA AT THE SAME TIME.** A FILE-PROCESSING SYSTEM IS USUALLY DESIGNED TO ALLOW ONE OR MORE PROGRAMS TO ACCESS DIFFERENT DATA FILES AT THE SAME TIME. **IN A FILE-PROCESSING SYSTEM**, A FILE CAN BE ACCESSED BY TWO PROGRAMS CONCURRENTLY ONLY IF BOTH PROGRAMS HAVE **READ-ONLY ACCESS** TO THE FILE

FILE LOCKING

- **FILE LOCKING** IS A MECHANISM THAT RESTRICTS ACCESS TO A COMPUTER FILE BY ALLOWING ONLY ONE USER OR PROCESS ACCESS AT ANY SPECIFIC TIME.
- A **RACE CONDITION** IS AN UNDESIRABLE SITUATION THAT OCCURS WHEN A DEVICE OR SYSTEM ATTEMPTS TO PERFORM TWO OR MORE OPERATIONS AT THE SAME TIME, BUT BECAUSE OF THE NATURE OF THE DEVICE OR SYSTEM, THE OPERATIONS MUST BE DONE IN THE PROPER SEQUENCE TO BE DONE CORRECTLY.

FILE LOCKING

- **PROCESS A READS A CUSTOMER RECORD FROM A FILE** CONTAINING ACCOUNT INFORMATION, INCLUDING THE CUSTOMER'S ACCOUNT BALANCE AND PHONE NUMBER.
- **PROCESS B NOW READS THE SAME RECORD** FROM THE SAME FILE SO IT HAS ITS OWN COPY.
- **PROCESS A CHANGES THE ACCOUNT BALANCE** IN ITS COPY OF THE CUSTOMER RECORD AND WRITES THE RECORD BACK TO THE FILE.
- **PROCESS B, WHICH STILL HAS THE ORIGINAL *STALE* VALUE** FOR
- PROCESS B HAS NOW WRITTEN ITS STALE ACCOUNT-BALANCE VALUE TO THE FILE, CAUSING THE CHANGES MADE BY PROCESS A TO BE LOST.

RECORD LOCKING

- **RECORD LOCKING** IS THE TECHNIQUE OF PREVENTING SIMULTANEOUS ACCESS TO DATA IN A DATABASE, TO PREVENT INCONSISTENT RESULTS.
- THE CLASSIC EXAMPLE IS DEMONSTRATED BY TWO BANK CLERKS ATTEMPTING TO UPDATE THE SAME BANK ACCOUNT FOR TWO DIFFERENT TRANSACTIONS. CLERKS 1 AND 2 BOTH RETRIEVE (I.E., COPY) THE ACCOUNT'S RECORD.
- CLERK 1 APPLIES AND SAVES A TRANSACTION. CLERK 2 APPLIES A DIFFERENT TRANSACTION TO HIS SAVED COPY, AND SAVES THE RESULT, BASED ON THE ORIGINAL RECORD AND HIS CHANGES, OVERWRITING THE TRANSACTION ENTERED BY CLERK 1. THE RECORD NO LONGER REFLECTS THE FIRST TRANSACTION, AS IF IT HAD NEVER TAKEN PLACE.

RECORD LOCKING

- THE CLASSIC EXAMPLE IS DEMONSTRATED BY TWO BANK CLERKS ATTEMPTING TO UPDATE THE SAME BANK ACCOUNT FOR TWO DIFFERENT TRANSACTIONS. CLERKS 1 AND 2 BOTH RETRIEVE (I.E., COPY) THE ACCOUNT'S RECORD.
- CLERK 1 APPLIES AND SAVES A TRANSACTION. CLERK 2 APPLIES A DIFFERENT TRANSACTION TO HIS SAVED COPY, AND SAVES THE RESULT, BASED ON THE ORIGINAL RECORD AND HIS CHANGES, OVERWRITING THE TRANSACTION ENTERED BY CLERK 1. THE RECORD NO LONGER REFLECTS THE FIRST TRANSACTION, AS IF IT HAD NEVER TAKEN PLACE.

RECORD LOCKING

- A SIMPLE WAY TO PREVENT THIS IS TO LOCK THE FILE WHENEVER A RECORD IS BEING MODIFIED BY ANY USER, SO THAT NO OTHER USER CAN SAVE DATA.
- THEY MAY BE ABLE TO VIEW THE RECORD WITHOUT EDITING IT
- THE INTRODUCTION OF GRANULAR (SUBSET) LOCKS CREATES THE POSSIBILITY FOR A SITUATION CALLED DEADLOCK.

SHARED LOCKS

- SHARED LOCKS DIFFER FROM EXCLUSIVE LOCKS IN THAT THE **HOLDER LIST** CAN CONTAIN MULTIPLE ENTRIES.
- SHARED LOCKS ALLOW ALL HOLDERS TO READ THE CONTENTS OF THE RECORD KNOWING THAT THE RECORD CANNOT BE CHANGED UNTIL AFTER THE LOCK HAS BEEN RELEASED BY ALL HOLDERS. **EXCLUSIVE LOCKS** CANNOT BE OBTAINED WHEN A RECORD IS ALREADY LOCKED (EXCLUSIVELY OR SHARED) BY ANOTHER ENTITY.

FUNCTIONS AND FACILITIES OF DATABASE MANAGEMENT SYSTEM

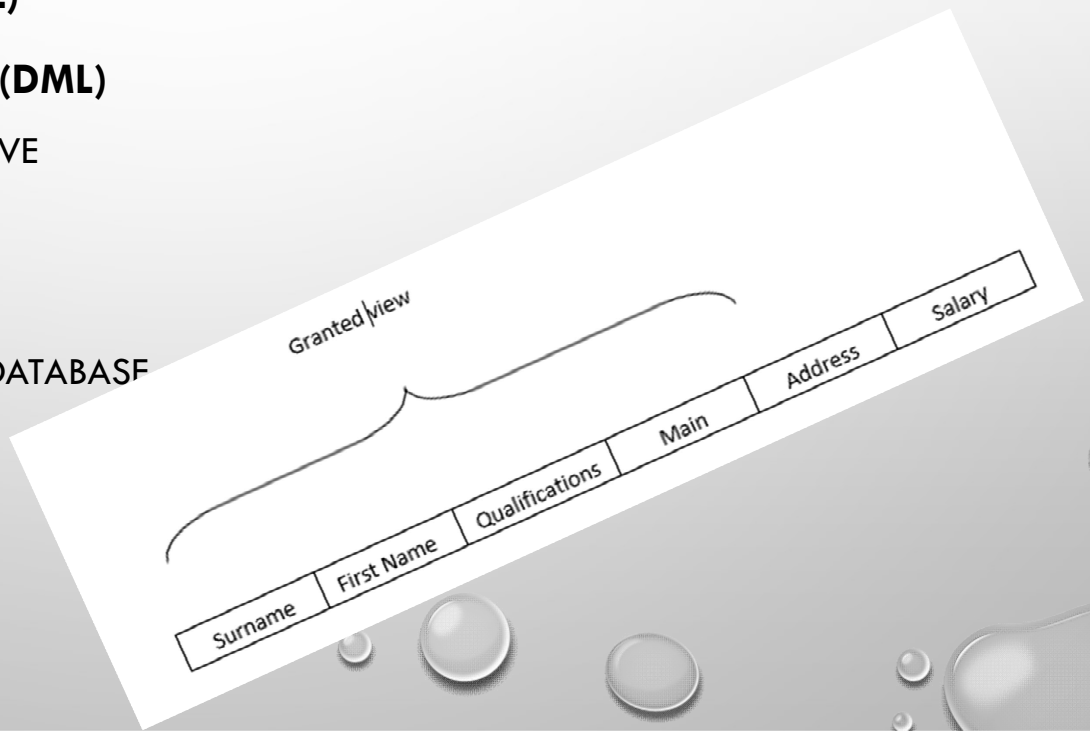
- DATA STORAGE MANAGEMENT
- DATA MANIPULATION MANAGEMENT
- DATA DEFINITION SERVICES
- DATA DICTIONARY/SYSTEM CATALOGUE MANAGEMENT
- DATABASE COMMUNICATION INTERFACES
- AUTHORIZATION / SECURITY MANAGEMENT
- BACKUP AND RECOVERY MANAGEMENT

FUNCTIONS AND FACILITIES OF DATABASE MANAGEMENT SYSTEM

- CONCURRENCY CONTROL SERVICE
- TRANSACTION MANAGEMENT
- DATABASE ACCESS AND APPLICATION PROGRAMMING INTERFACES

FACILITIES OF A DBMS

- **DATA DEFINITION LANGUAGE (DDL)**
- **DATA MANIPULATION LANGUAGE (DML)**
  - ❖ INSERT, UPDATE, DELETE AND RETRIEVE
  - ❖ A QUERY LANGUAGE - SQL
- **VIEW MECHANISM**
  - ❖ A VIEW THAT IS A SUBSET OF THE DATABASE



FACILITIES OF A DBMS

- **MULTIPLE INDEXES**
- AN INDEX IS A MECHANISM FOR REDUCING THE TIME TAKEN TO FIND A SPECIFIC ITEM OF DATA IN A DATABASE.
- AN INDEX IN A DATABASE CAN STORE EACH VALUE OF AN INDEXED DATA ITEM (FIELD), E.G. STUDENT ENROLMENT NUMBER, TOGETHER WITH THE PAGE NUMBER IN THE STORAGE MEDIUM WHERE THE DATA BELONGING TO THIS VALUE IS STORED.
- **FOR EXAMPLE, INFORMATION STORED IN A DATABASE ABOUT A PARTICULAR STUDENT**

DATA MODELS USED IN GIS

- ZERO-DIMENSIONAL OBJECTS OR POINTS.
- POINTS ARE USED FOR REPRESENTING THE LOCATION OF ENTITIES WHOSE **SHAPE IS NOT CONSIDERED AS USEFUL,** OR WHEN THE AREA IS QUITE SMALL WITH RESPECT TO THE EMBEDDING SPACE SIZE. CITIES, CHURCHES, AND CROSSINGS ARE EXAMPLES OF ENTITIES WHOSE SPATIAL EXTENT MIGHT BE REDUCED TO A POINT ON A LARGE-SCALE MAP.

DATA MODELS USED IN GIS

- ZERO-DIMENSIONAL OBJECTS OR POINTS.
- POINTS ARE USED FOR REPRESENTING THE LOCATION OF ENTITIES WHOSE **SHAPE IS NOT CONSIDERED AS USEFUL,** OR WHEN THE AREA IS QUITE SMALL WITH RESPECT TO THE EMBEDDING SPACE SIZE. CITIES, CHURCHES, AND CROSSINGS ARE EXAMPLES OF ENTITIES WHOSE SPATIAL EXTENT MIGHT BE REDUCED TO A POINT ON A LARGE-SCALE MAP.

DATA MODELS USED IN GIS

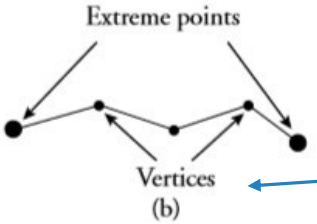
- **ONE-DIMENSIONAL OBJECTS** OR LINEAR OBJECTS.
- THESE OBJECTS ARE COMMONLY USED FOR REPRESENTING NETWORKS (ROADS, HYDROGRAPHY, AND SOON).

DATA MODELS USED IN GIS

- **A POLYLINE IS DEfiNED AS A FINITE SET OFFLINE SEGMENTS OR EDGES, SUCH THAT EACH SEGMENT ENDPOINT (CALLED A VERTEX) IS SHARED BY EXACTLY TWO SEGMENTS, EXCEPT FOR TWO ENDPOINTS (CALLED THE EXTREME POINTS), WHICH BELONG TO ONLY ONE SEGMENT.**
- A POLYLINE IS CLOSED IF THE TWO EXTREME POINTS ARE IDENTICAL.
- A SIMPLE POLYLINE IS SUCH THAT NO PAIRS OF NON-CONSECUTIVE EDGES INTERSECT AT ANY PLACE.
- A POLYLINE IS MONOTONE WITH RESPECT TO A LINE L IF EVERY LINE L ORTHOGONAL TO L MEETS THE POLYLINE AT ONE POINT AT MOST.

Line segment

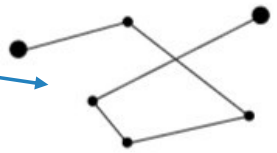
(a)



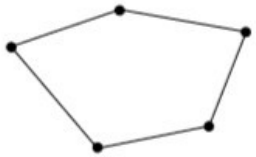
(b)

polyline

non- simple polyline



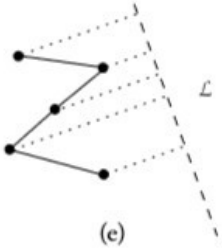
(c)



(d)

simple closed polyline

monotone polyline

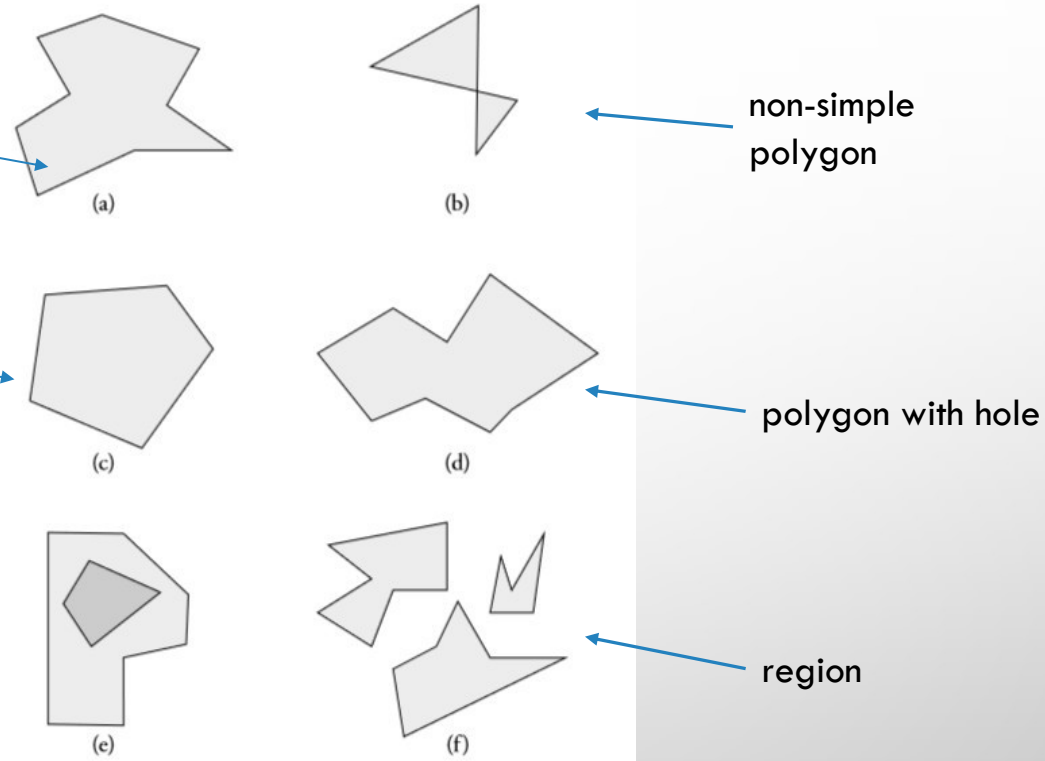


(e)

(f)

and non-monotone polyline

- A POLYGON IS A REGION OF THE PLANE BOUNDED BY A CLOSED POLYLINE, CALLED ITS BOUNDARY.
- A POLYGON IS SIMPLE IF ITS BOUNDARY IS A SIMPLE POLYLINE
- A CONVEX POLYGON P IS SUCH THAT FOR ANY PAIR OF POINTS A AND B IN P THE SEGMENT AB IS FULLY INCLUDED IN P.
- A MONOTONE POLYGON IS A SIMPLE POLYGON SUCH THAT ITS BOUNDARY ΔP CAN BE SPLIT INTO EXACTLY TWO MONOTONE POLYLINES MC1 AND MC2


simple polygon
non-simple polygon
convex polygon
polygon with hole
polygon with hole
region
(a)
(b)
(c)
(d)
(e)
(f)

DATA ARCHITECTURE

- IN INFORMATION TECHNOLOGY, **DATA ARCHITECTURE** IS COMPOSED OF
- MODELS,
- POLICIES,
- RULES OR STANDARDS THAT
- GOVERN WHICH DATA IS COLLECTED, AND HOW IT IS STORED, ARRANGED, INTEGRATED, AND PUT TO USE IN DATA SYSTEMS AND IN ORGANIZATIONS.

DATA ARCHITECTURE

- 3 TRADITIONAL ARCHITECTURAL PROCESSES:

❖CONCEPTUAL - REPRESENTS ALL BUSINESS ENTITIES.

❖LOGICAL - REPRESENTS THE LOGIC OF HOW ENTITIES ARE RELATED.

❖PHYSICAL - THE REALIZATION OF THE DATA MECHANISMS FOR A SPECIFIC TYPE OF FUNCTIONALITY.

DATA ARCHITECTURE

- CONSTRAINTS AND INFLUENCES
- ENTERPRISE REQUIREMENTS
- TECHNOLOGY DRIVERS
- ECONOMICS
- BUSINESS POLICIES
- DATA PROCESSING NEEDS- ACCURATE AND REPRODUCIBLE TRANSACTIONS PERFORMED IN HIGH VOLUMES,

DATA SECURITY, CONSISTENCY AND INTEGRITY

- **DATA SHARING**
- **DATA SECURITY**
- **DISK ENCRYPTION**
- SOFTWARE-BASED SECURITY SOLUTIONS ENCRYPT THE DATA TO PROTECT IT FROM THEFT.

BACKUPS

DATA MASKING

DATA ERASURE

DATA SECURITY, CONSISTENCY AND INTEGRITY

- **DATA INTEGRITY** IS THE MAINTENANCE OF, AND THE ASSURANCE OF THE ACCURACY AND CONSISTENCY OF, DATA OVER ITS ENTIRE LIFE-CYCLE
- PHYSICAL INTEGRITY DEALS WITH CHALLENGES ASSOCIATED WITH CORRECTLY STORING AND FETCHING THE DATA ITSELF.
- **LOGICAL INTEGRITY**
- THIS TYPE OF INTEGRITY IS CONCERNED WITH THE CORRECTNESS OR RATIONALITY OF A PIECE OF DATA, GIVEN A PARTICULAR CONTEXT.

TYPES OF INTEGRITY CONSTRAINTS

- **TYPES OF INTEGRITY CONSTRAINTS**
- ENTITY
- REFERENTIAL
- DOMAIN

TYPES OF INTEGRITY CONSTRAINTS

- *ENTITY INTEGRITY* CONCERNS THE CONCEPT OF A PRIMARY KEY. ENTITY INTEGRITY IS AN INTEGRITY RULE WHICH STATES THAT EVERY TABLE MUST HAVE A PRIMARY KEY AND THAT THE COLUMN OR COLUMNS CHOSEN TO BE THE PRIMARY KEY SHOULD BE UNIQUE AND NOT NULL.
- *REFERENTIAL INTEGRITY* CONCERNS THE CONCEPT OF A FOREIGN KEY. THE REFERENTIAL INTEGRITY RULE STATES THAT ANY FOREIGN-KEY VALUE CAN ONLY BE IN ONE OF TWO STATES. THE USUAL STATE OF AFFAIRS IS THAT THE FOREIGN-KEY VALUE REFERS TO A PRIMARY KEY VALUE OF SOME TABLE IN THE DATABASE. OCCASIONALLY, AND THIS WILL DEPEND ON THE RULES OF THE DATA OWNER, A FOREIGN-KEY VALUE CAN BE NULL. IN THIS CASE WE ARE EXPLICITLY SAYING THAT EITHER THERE IS NO RELATIONSHIP BETWEEN THE OBJECTS REPRESENTED IN THE DATABASE OR THAT THIS RELATIONSHIP IS UNKNOWN.
- *DOMAIN INTEGRITY* SPECIFIES THAT ALL COLUMNS IN A RELATIONAL DATABASE MUST BE DECLARED UPON A DEFINED DOMAIN. THE PRIMARY UNIT OF DATA IN THE RELATIONAL DATA MODEL IS THE DATA ITEM. SUCH DATA ITEMS ARE SAID TO BE NON-DECOMPOSABLE OR ATOMIC. A DOMAIN IS A SET OF VALUES OF THE SAME TYPE. DOMAINS ARE THEREFORE POOLS OF VALUES FROM WHICH ACTUAL VALUES APPEARING IN THE COLUMNS OF A TABLE ARE DRAWN.
- *USER-DEFINED INTEGRITY* REFERS TO A SET OF RULES SPECIFIED BY A USER, WHICH DO NOT BELONG TO THE ENTITY, DOMAIN AND REFERENTIAL INTEGRITY CATEGORIES.

DATA CONSISTENCY AND INDEPENDENCE

- POINT IN TIME CONSISTENCY
- TRANSACTION CONSISTENCY
- CONSISTENCY (DATABASE SYSTEMS) IN THE REALM OF DISTRIBUTED DATABASE SYSTEMS REFERS TO THE PROPERTY OF MANY ACID DATABASES TO ENSURE THAT THE RESULTS OF A DATABASE TRANSACTION ARE VISIBLE TO ALL NODES SIMULTANEOUSLY.
- ACID --- **ATOMICITY CONSISTENCY ISOLATION** AND **DURABILITY**

APPLICATION CONSISTENCY

- IS THE STATE IN WHICH ALL RELATED FILES AND DATABASES ARE SYNCHRONIZED REPRESENTING THE TRUE STATUS OF THE APPLICATION
- LOGICAL DATA INDEPENDENCE: THE ABILITY TO CHANGE THE LOGICAL (CONCEPTUAL) SCHEMA WITHOUT CHANGING THE EXTERNAL SCHEMA (USER VIEW)
- PHYSICAL DATA INDEPENDENCE: THE ABILITY TO CHANGE THE PHYSICAL SCHEMA WITHOUT CHANGING THE LOGICAL SCHEMA IS CALLED PHYSICAL DATA INDEPENDENCE.

DATABASE MODELS

- FLAT
- HIERARCHICAL
- NETWORK
- RELATIONAL
- OBJECT -RELATIONAL