

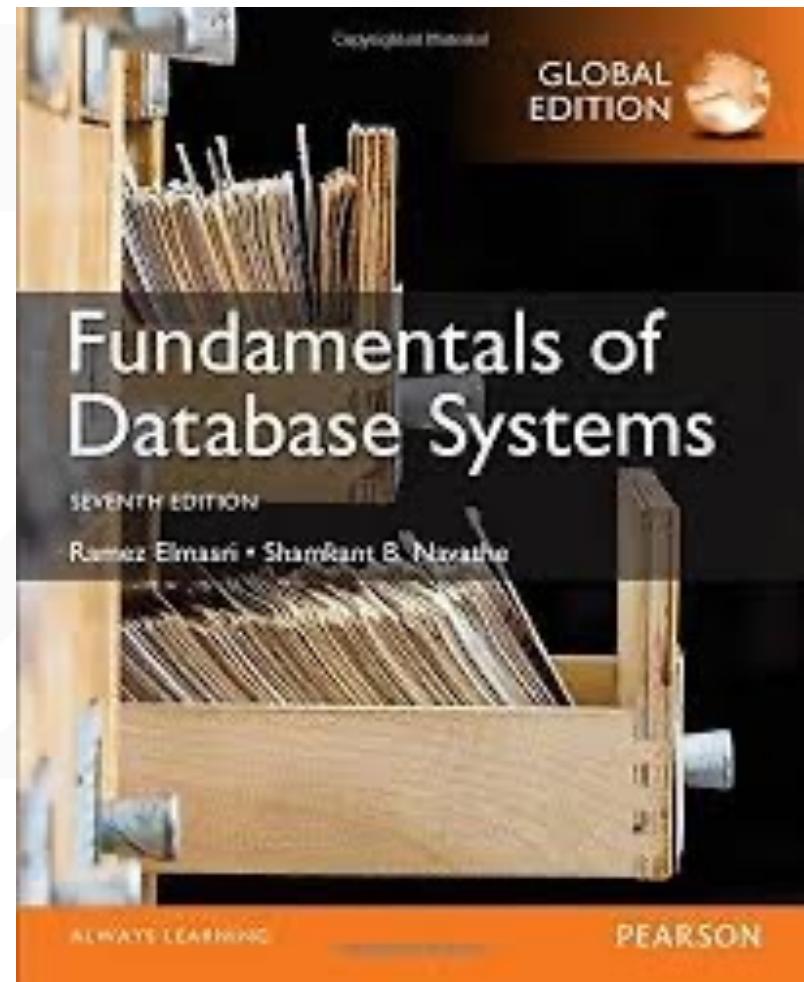
Database Systems

Program in Computer Engineering
School of Engineering

King Mongkut's Institute of Technology Ladkrabang

Text

- Ramez Elmasri and Shamkant B. Navathe.
“Fundamentals of Database Systems”
7th Edition., Pearson, 2017



Chapter 2

Database System Concepts and Architecture

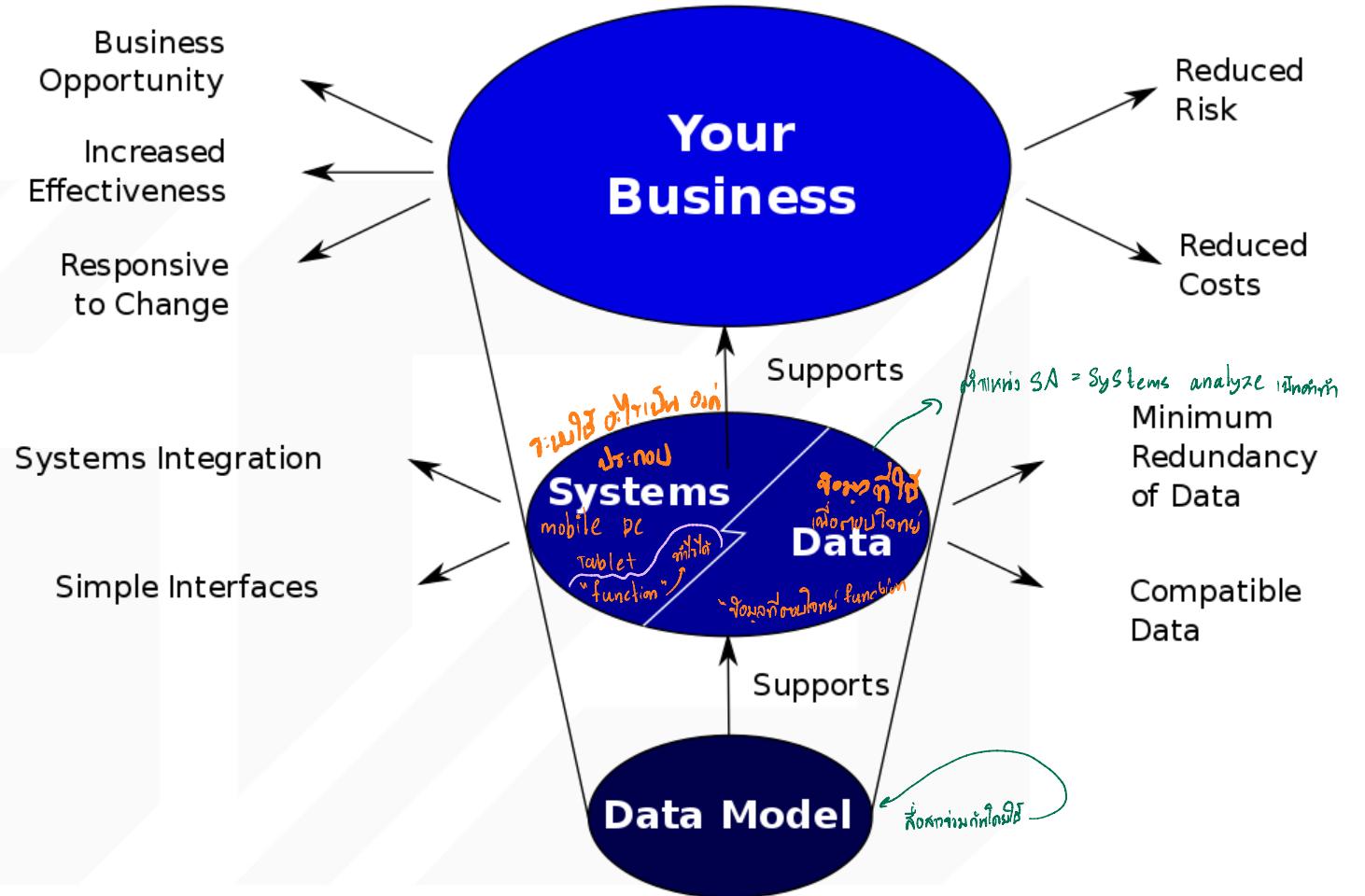
Data Models

- A set of concepts to describe the **structure** of a database, the **operations** for manipulating these structures, and certain **constraints** that the database should obey.

โครงสร้างข้อมูล ลักษณะ ความสัมพันธ์
element ตัวประกอบเชิงลึก
group element → table
relation
โครงสร้างที่ส่งผลต่อการจัดเก็บข้อมูล

คําสั่ง SQL
basic model operation
 $+ - \times \div$
user define operation
โปรแกรมประมวลผล ฟังก์ชัน function คำสั่งภาษา

- Data models provide a **framework** (มอง miniworld) for data to be used within **information systems** by providing specific definition and format



Data Model Structure and Constraints

- Constructs are used to define the database structure
- Constructs typically include
 - elements (and their data types) ឧប្បជ្ជនា រិអំណី (Text, Int)
 - groups of elements (e.g. entity, record, table)
 - relationships among such groups ទាញយកចំណាំសម្រាប់ផ្លូវការ
- Constraints specify some restrictions on valid data; these constraints must be enforced at all times
 - ខ្សោយការណ៍ដែលត្រូវពាក្យ

Data Model Operations

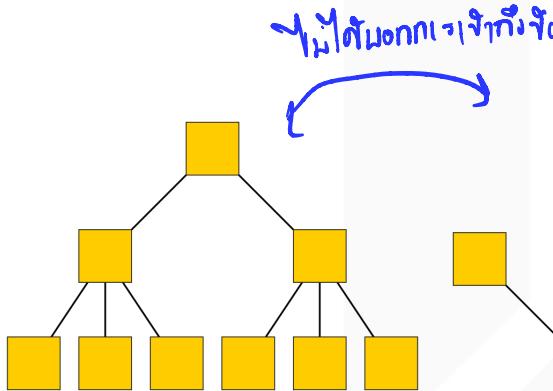
ຕົວຢ່າງ

- These operations are used for specifying database **retrievals** and **updates** by referring to the constructs of the data model. ↳ ສິ້ນທຸກພາຍໃຕ້ insert delete modify
 - Operations** on the data model may include
 - basic model operations** (e.g. generic insert, delete, update) +,-,*,÷
 - user-defined operations** (e.g. compute_student_gpa, update_inventory)
- ↳ ໂດຍມາ process ອັນດີຕະຫຼອນ function ທີ່ນີ້

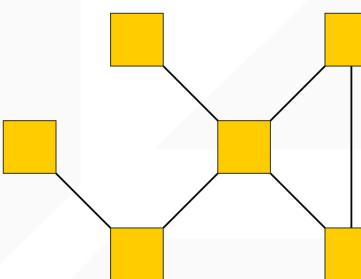
high-level ^{conceptual} និងការប្រើប្រាស់ + ការរាយការណ៍
Representational data models
lower-level ^{physical}

Categories of Data Models

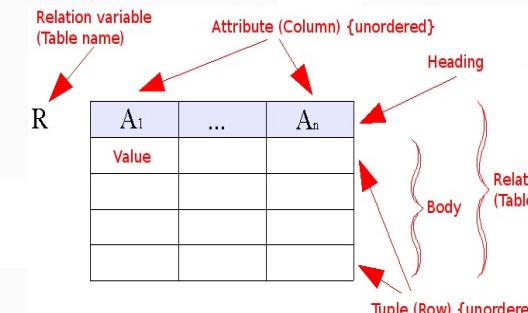
- Conceptual (high-level, semantic) data models:**
- Provide concepts that are close to the way many users perceive data.



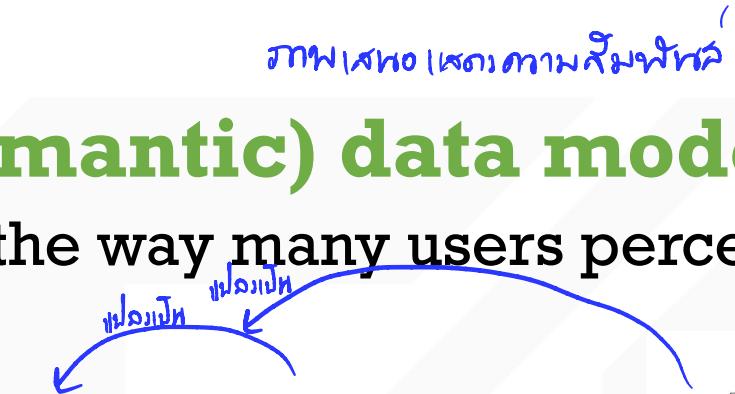
Hierarchical model



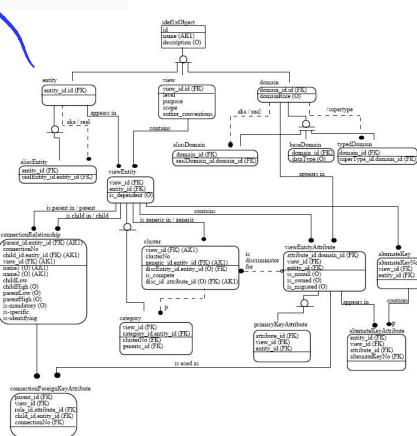
Network model



Relational model (ជីវិត្យាន)
ការរាយការណ៍ ហំរោងអនុញ្ញាត



Concept-oriented model



Object-Role model
កម្មសាន្តរក្សាទុកដៃ

high-level ^{conceptual} \rightarrow អ្នកប្រើប្រាស់ + នគរាយសំខាន់ៗ
Representational data models \rightarrow logical schema + relational data model "អ្នកប្រើប្រាស់ computer storage" } នាំ ទាញយកពេលវេលាដែលត្រូវបាន
lower-level ^{physical}

ក្រឡា

• Implementation (representational) data models:

- A.k.a., **logical schema**
- Provide concepts used by many commercial DBMS implementations (e.g. **relational data models** used in many commercial systems)
- Describe the structure of some domain of information.
- This consists of descriptions of (for example) **tables**, **columns**, **object-oriented classes**, and **XML tags**.
- The logical schema and conceptual schema are sometimes implemented as one and the same.

high-level ^{conceptual} \rightarrow អ្ននាគ់ user និង + លក្ខណភាពប័ណ្ណ

representational data models \rightarrow logical schema + relational data model "នានាគ់ computer storage" } នានា គ្រប់គ្រងដោយការរំពេញនូវការបញ្ចូន

lower-level ^{physical} or physical data models - និងបានរាយការណ៍ទៅលម្អិត សុចរើនូវវិធី physical storage

- កែវឌីជីថប់ partition, CPUs, tablespace

- ការណិតនាំ ad-hoc នៃ DBMS designer & administration manual

លាស់ទៅ

• Physical (low-level, internal) data models:

- Provide concepts that describe details of how data is stored in the computer. \rightarrow កែវឌីជីថប់
- This is concerned with partitions, CPUs, tablespaces, and the like. \rightarrow កែវឌីជីថប់ \rightarrow កិច្ចការ
- These are usually specified in an ad-hoc manner through DBMS design and administration manuals. \rightarrow (កិច្ចការ) \rightarrow និង

Schemas

- In a data model, it is important to distinguish between the **description of the database** and the **database** itself.
 - The description of a database is called the **database schema**.
“ការ (នេរ ដំ) ផ្តល់ព័ត៌មានចិត្ត អងគោរា ដើម្បីប្រព័ន្ធផ្លូវការ ”
→ ឧប្បយ័ត... ទីនេះ
 - A displayed schema is called a **schema diagram**.

Schemas ពាក្យសម្រាប់ (បច្ចុប្បន្នពាក្យសម្រាប់)

STUDENT

| Name | Student_number | Class | Major |
|------|----------------|-------|-------|
|------|----------------|-------|-------|

COURSE

| Course_name | Course_number | Credit_hours | Department |
|-------------|---------------|--------------|------------|
|-------------|---------------|--------------|------------|

PREREQUISITE

| Course_number | Prerequisite_number |
|---------------|---------------------|
|---------------|---------------------|

SECTION

| Section_identifier | Course_number | Semester | Year | Instructor |
|--------------------|---------------|----------|------|------------|
|--------------------|---------------|----------|------|------------|

GRADE_REPORT

| Student_number | Section_identifier | Grade |
|----------------|--------------------|-------|
|----------------|--------------------|-------|

Figure 2.1

Schema diagram for the database in Figure 1.2.

ទំនួរការណ៍ឯងជាអ្នកប្រើប្រាស់

Database State (instance)

(ចំណុចសង្គមនៃពិភព) នឹង return state នៅពេលខ្លួនក្នុងការ គាំទាមសេយដ្ឋាយ

- The actual data stored in a database at a **particular moment** in time.
- This includes the collection of all the data in the database.
- Also called database instance (or **occurrence** or **snapshot**).
state
- The term instance is also applied to individual database components, e.g. **record instance**, **table instance**, **entity instance**

អ្នកប្រើប្រាស់

អ្នកប្រើប្រាស់ពេលខ្លួន

▷ ពេលវេលា នៅ Insert del up
 ▷ នៅពេលវេលាដែលមិនមែនស្ថាព

• Database State:

តួនាទីនៃបច្ចេកទេសនៅពេលវេលា

- Refers to the content of a database at a moment in time.

តួនាទីនៃបច្ចេកទេសនៅពេលវេលាដែលមិនមែនស្ថាព

• Initial Database State:

ជាអតិថតគ្នា → តួនាទីតាមតម្លៃ / តម្លៃ (constant)

- Refers to the database state when it is initially loaded into the system.

→ តួនាទីតាមតម្លៃ ✓ តាមតម្លៃ តម្លៃរហូតដល់

• Valid State:

- A state that satisfies the structure and constraints of the database.

- **Distinction**

โครงสร้างที่คงเปลี่ยนแปลงยาก (**database schema**)

- The **database schema** changes very infrequently.
- The **database state** changes every time the database is updated.
(**database state**)
- **Schema** is also called **intension**.
- **State** is also called **extension**.
(**database extension**)

SCWome

STUDENT

| Name | Student_number | Class | Major |
|------|----------------|-------|-------|
|------|----------------|-------|-------|

COURSE

| Course_name | Course_number | Credit_hours | Department |
|-------------|---------------|--------------|------------|
|-------------|---------------|--------------|------------|

PREREQUISITE

| Course_number | Prerequisite_number |
|---------------|---------------------|
|---------------|---------------------|

SECTION

| Section_identifier | Course_number | Semester | Year | Instructor |
|--------------------|---------------|----------|------|------------|
|--------------------|---------------|----------|------|------------|

GRADE_REPORT

| Student_number | Section_identifier | Grade |
|----------------|--------------------|-------|
|----------------|--------------------|-------|

Figure 2.1
Schema diagram for the database in Figure 1.2.

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 |

intention /schemas →
extension / stat {

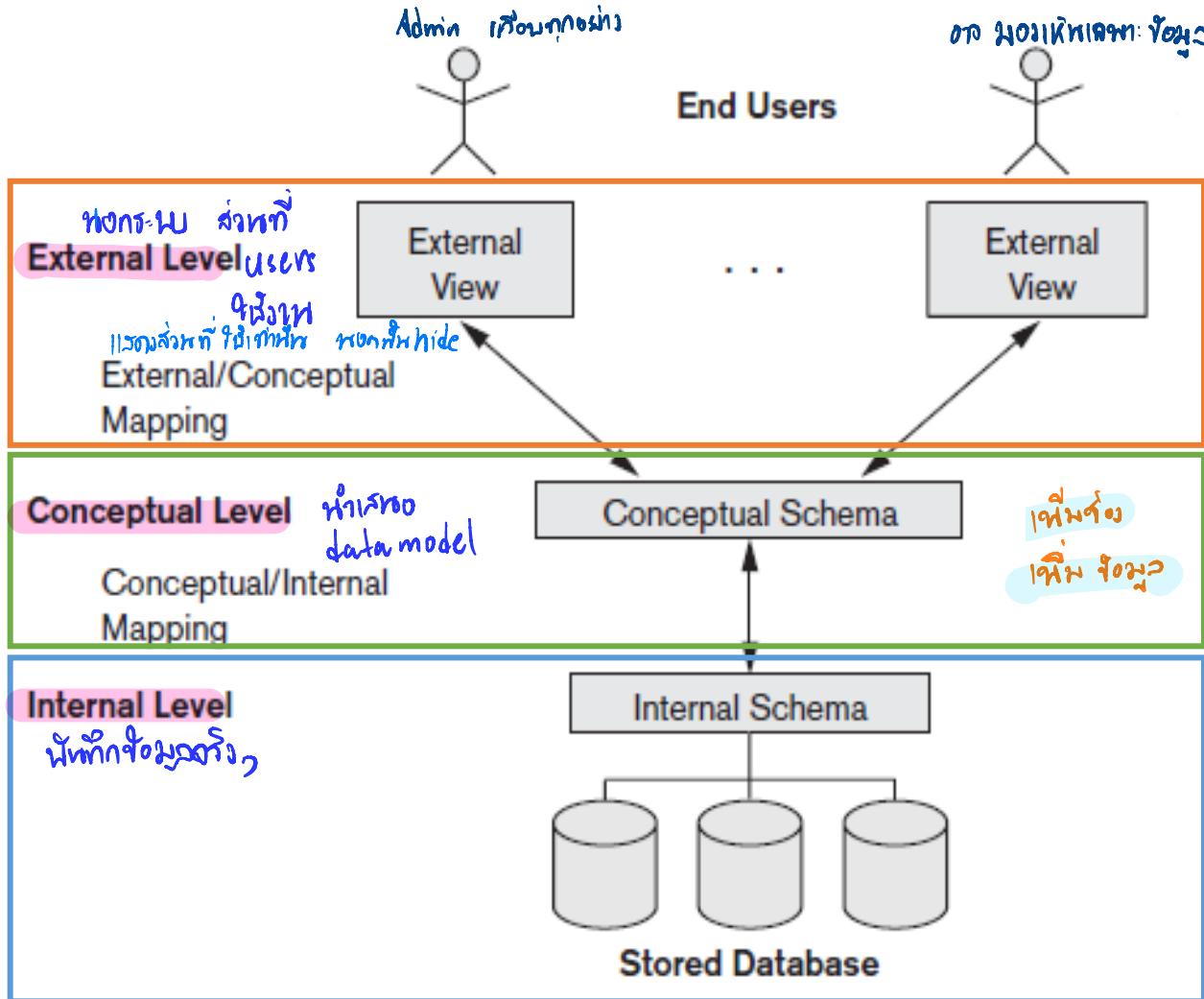
PREREQUISITE

| Course_number | Prerequisite_number |
|---------------|---------------------|
| CS3380 | CS3320 |
| CS3380 | MATH2410 |
| CS3320 | CS1310 |

Figure 1.2
A database that stores student and course information.

សរុបឯកការណ៍ទីនៃទម្រង់ និងអាជីវកម្ម (ជូនអស់)

The Three-Schema Architecture



• External Level

- A number of external schemas or user views.
- Each external schema describes the part of the database that a particular user group is interested in and hides the rest of the database.

• Conceptual Level (Logical Level)

- Describe the structure of the whole database for a community of users.
- Hide the details of physical storage structures and concentrates on describing entities, data types, relationships, user operations, and constraints.

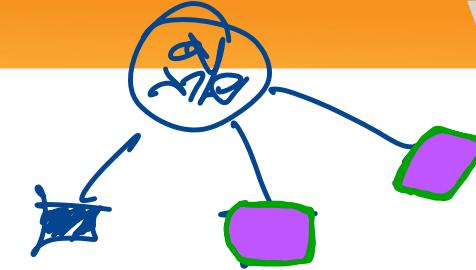
• Internal Level

- Describe the physical storage structure of the database.
- Use a physical model and describe the complete details of data storage and access paths for the database.



Data Independence

Conceptual → External



• Logical Data Independence:

- The capacity to change the conceptual schema without having to change the external schemas and their associated application programs.

គ្រប់គ្រង

• Physical Data Independence:

គ្រប់គ្រង

- The capacity to change the internal schema without having to change the conceptual schema. *Extraction* 
- For example, the internal schema may be changed when certain file structures are reorganized, or new indexes are created to improve database performance

Internal → Conceptual

User-Friendly DBMS Interfaces

- **Menu-based (Web-based)** ເລືອດນາມສັກ
• popular for browsing on the web
- **Forms-based** ນອນຫອຸ່ນ
• designed for naïve users used to filling in entries on a form
- **Graphics-based**
 - Point and Click, Drag and Drop, etc.
 - Specifying a query on a schema diagram
- **Natural language** ອົບ keyword ໂພ້ ກະພາໄນ້
 - requests in written English
- **Combinations of the above**
 - For example, both menus and forms used extensively in Web database interfaces

ກ່ຽວຂ້ອງ
ກອບກັນ
ໄລຍະທຳ

Other DBMS Interfaces

- Natural language
 - free text as a query
- Speech (ภาษาอังกฤษ)
 - Input query and Output response
- Web Browser with keyword search
- Parametric interfaces, *Interface param.*
 - e.g., bank tellers using function keys.
- Interfaces for the DBA:
 - Creating user accounts, granting authorizations
 - Setting system parameters
 - Changing schemas or access paths

Database System Utilities

- To perform certain functions such as:
 - Loading data stored in files into a database. Includes data conversion tools.
 - Backing up the database periodically on tape.
 - Reorganizing database file structures.
 - Performance monitoring utilities.
 - Report generation utilities.
 - Other functions, such as sorting, user monitoring, data compression, etc.

↳cpu
↳drive

Other Tools

Øhrnström Model

- Application Development Environments and CASE (computer-aided software engineering) tools:
- Examples:
 - PowerBuilder (Sybase)
 - JBuilder (Borland)
 - JDeveloper 10G (Oracle)

Typical DBMS Component Modules

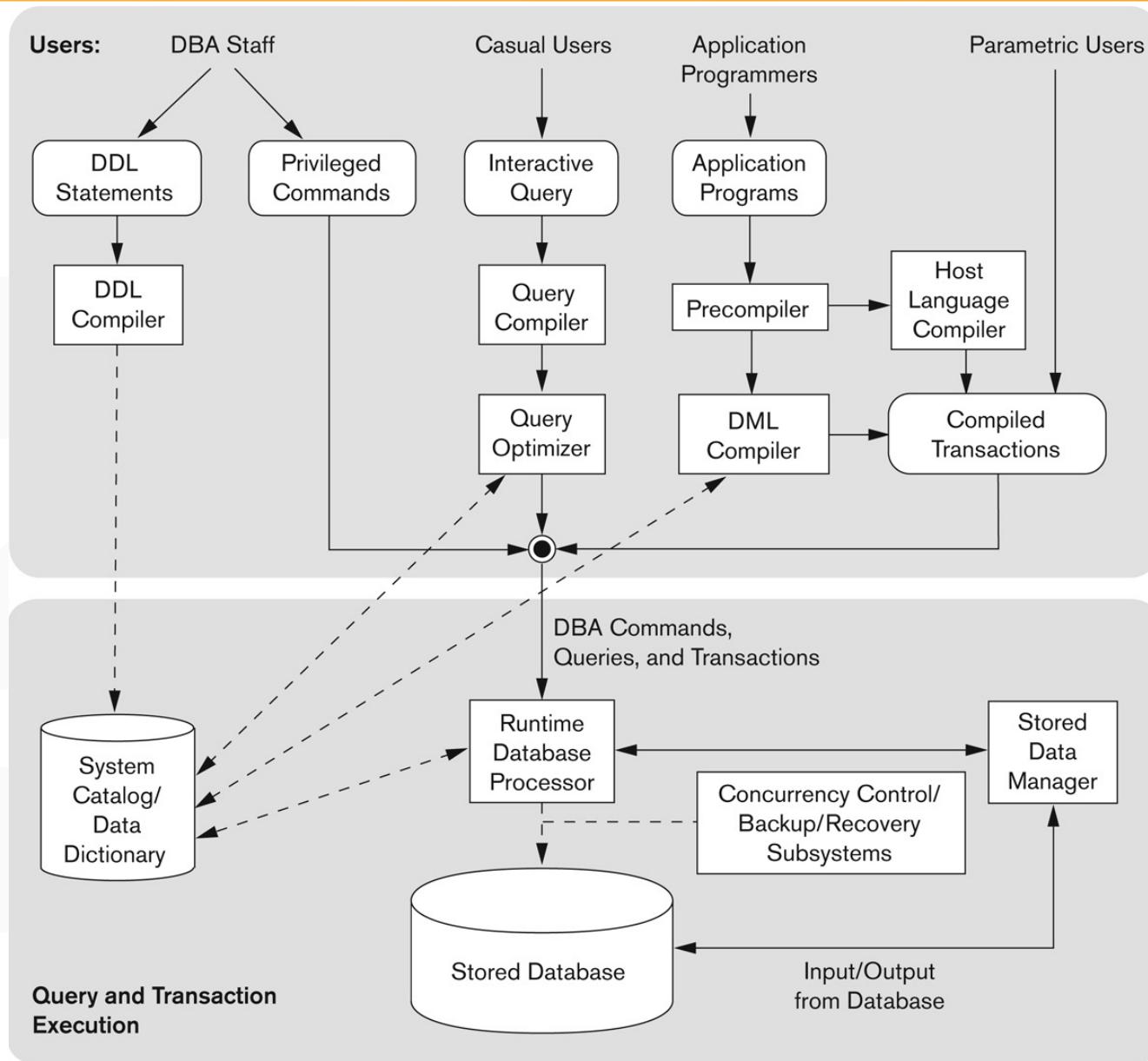


Figure 2.3
Component modules of a DBMS and their interactions.

ក្រុងរបៀបនៃការបង្កើតកម្មវិធី

A Physical Centralized Architecture

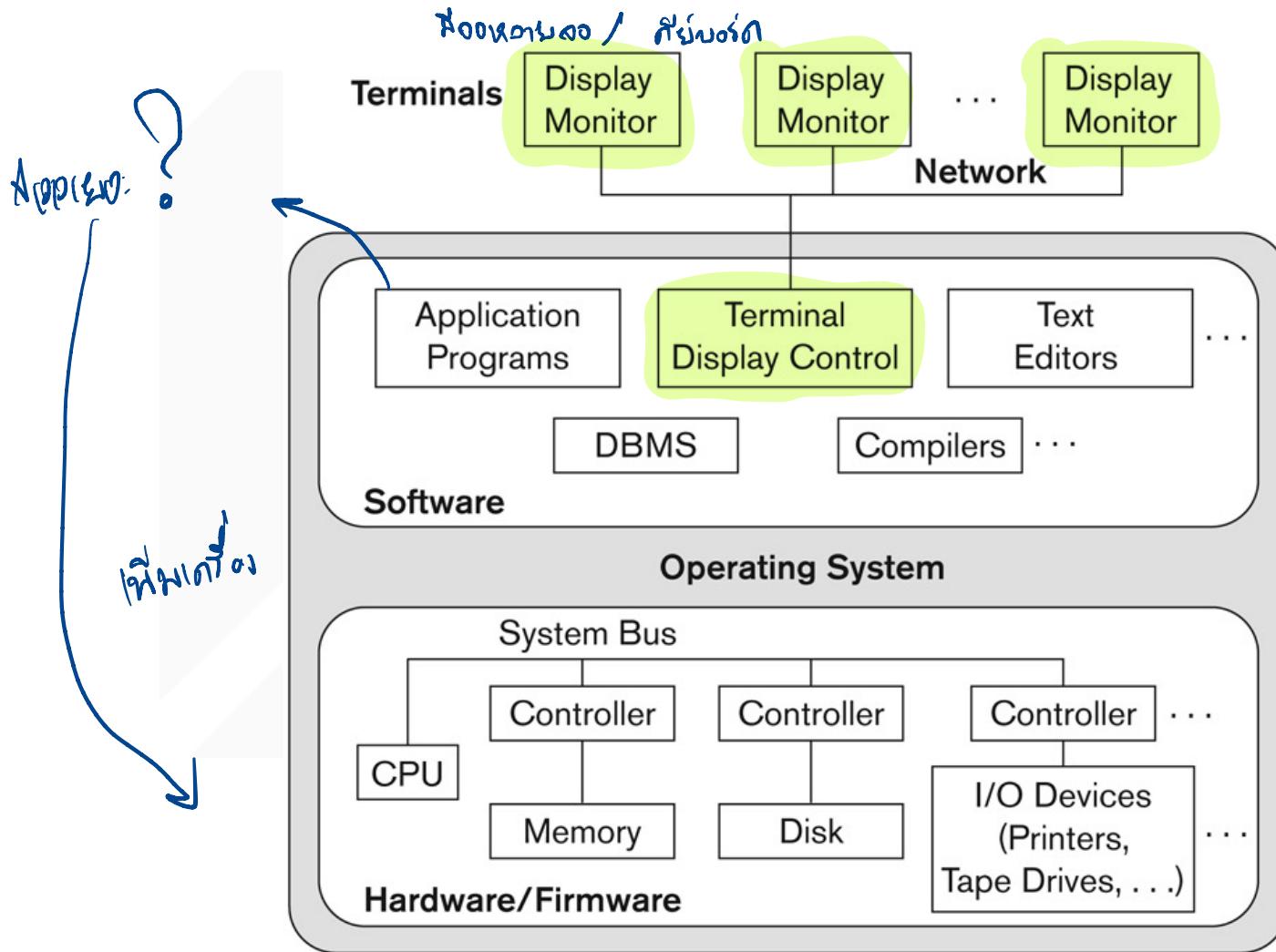


Figure 2.4
A physical centralized architecture.

ການຈົບດັບໄດ້

- **Centralized DBMS:**

- Combines everything into single system including- DBMS software, hardware, application programs, and user interface processing software.
- User can still connect through a remote terminal – however, all processing is done at centralized site

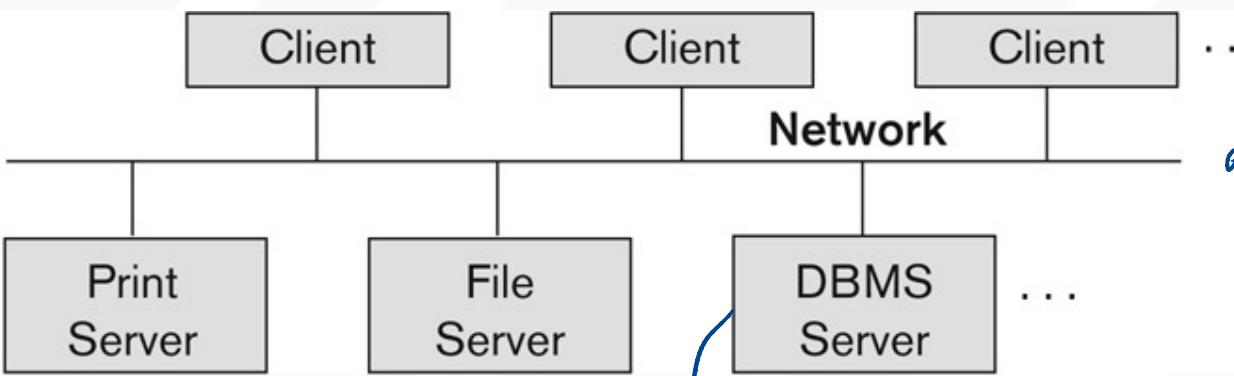
ຂ່າຍກັບ

ຄູກຕົກ

ພິຈານໂນໂມ

Two-tier Client/Server

Figure 2.5
Logical two-tier client/server architecture.



ເກີນໄຟລ໌

provide ຈົນ?

ອະນາກອນນຸ້າໃຈນຳມານຸ້າ

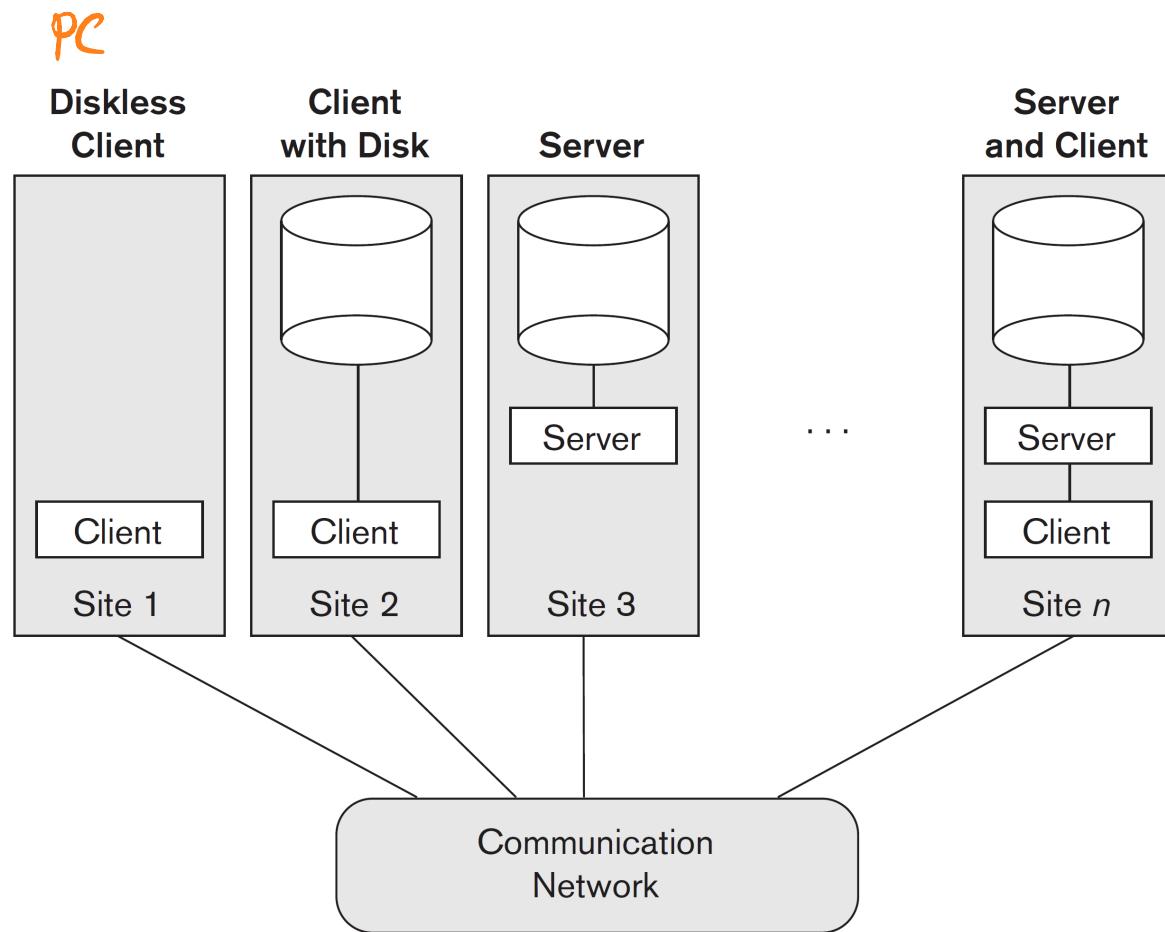
client 1 ຊືວ່າ? update softwv?

ຄູກກັກພາກ network

ກົມ App ອີ່ຫຼິດ
ເປົ້າຫຼິດ

- **Specialized Servers with Specialized functions**
 - Print server
 - File server
 - DBMS server
 - Web server
 - Email server
- **Clients can access the specialized servers as needed**

Figure 2.6
Physical two-tier client/server architecture.



Client

- **Clients**

- Provide appropriate interfaces through a client software module to access and utilize the various server resources.
- Clients may be **diskless machines** or **PCs** or **Workstations with disks** with only the client software installed.
- Connected to the servers via some form of a network.
 - (LAN: local area network, wireless network, etc.)

• DBMS Server

- Provides database query and transaction services to the clients
- Relational DBMS servers are often called **SQL servers, query servers, or transaction servers**
- Applications running on clients utilize an Application Program Interface (API) to access server databases via standard interface
(whom)
 - ODBC: Open Database Connectivity standard
 - JDBC: for Java programming access

Three-tier Client/Server

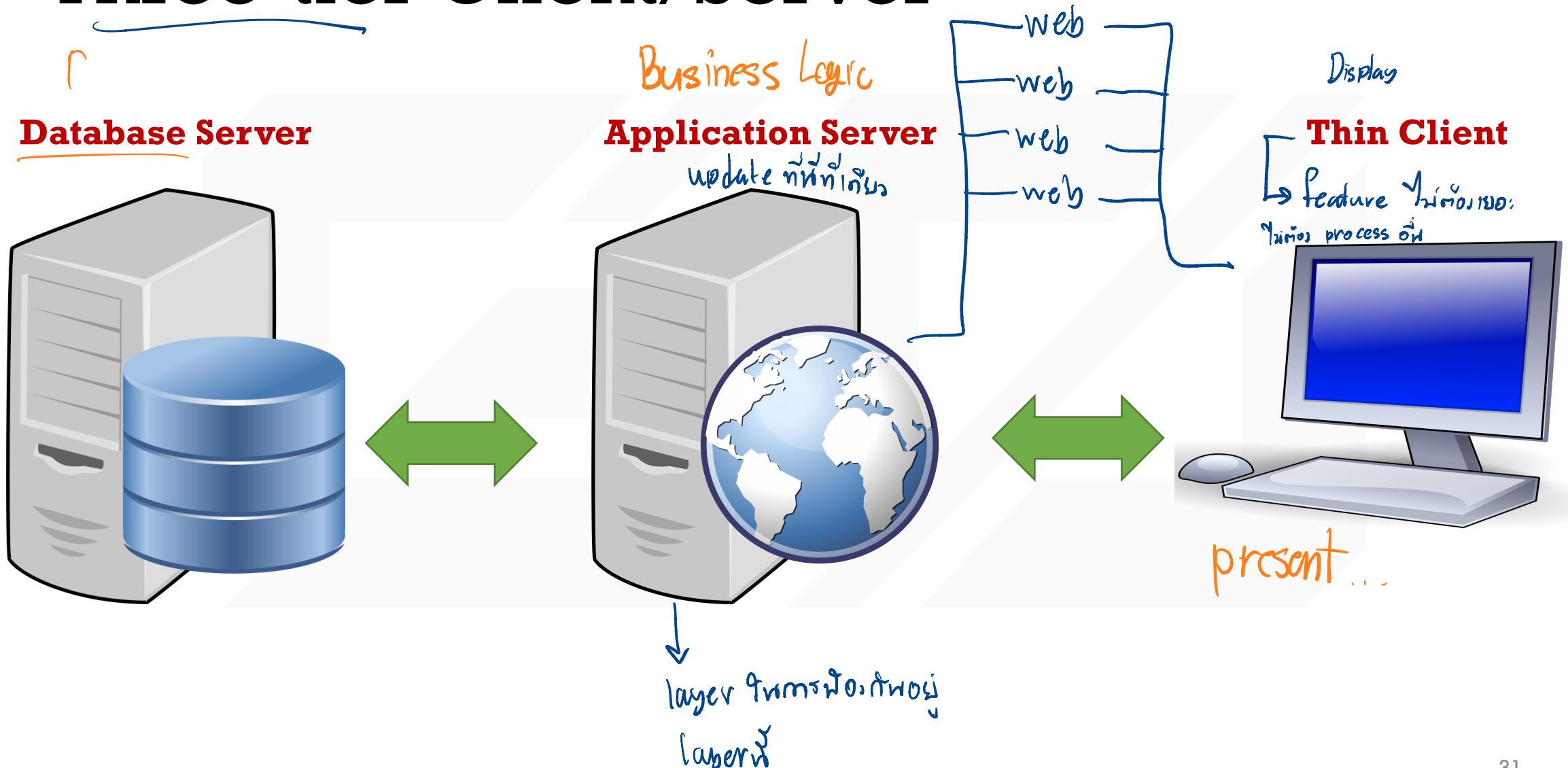
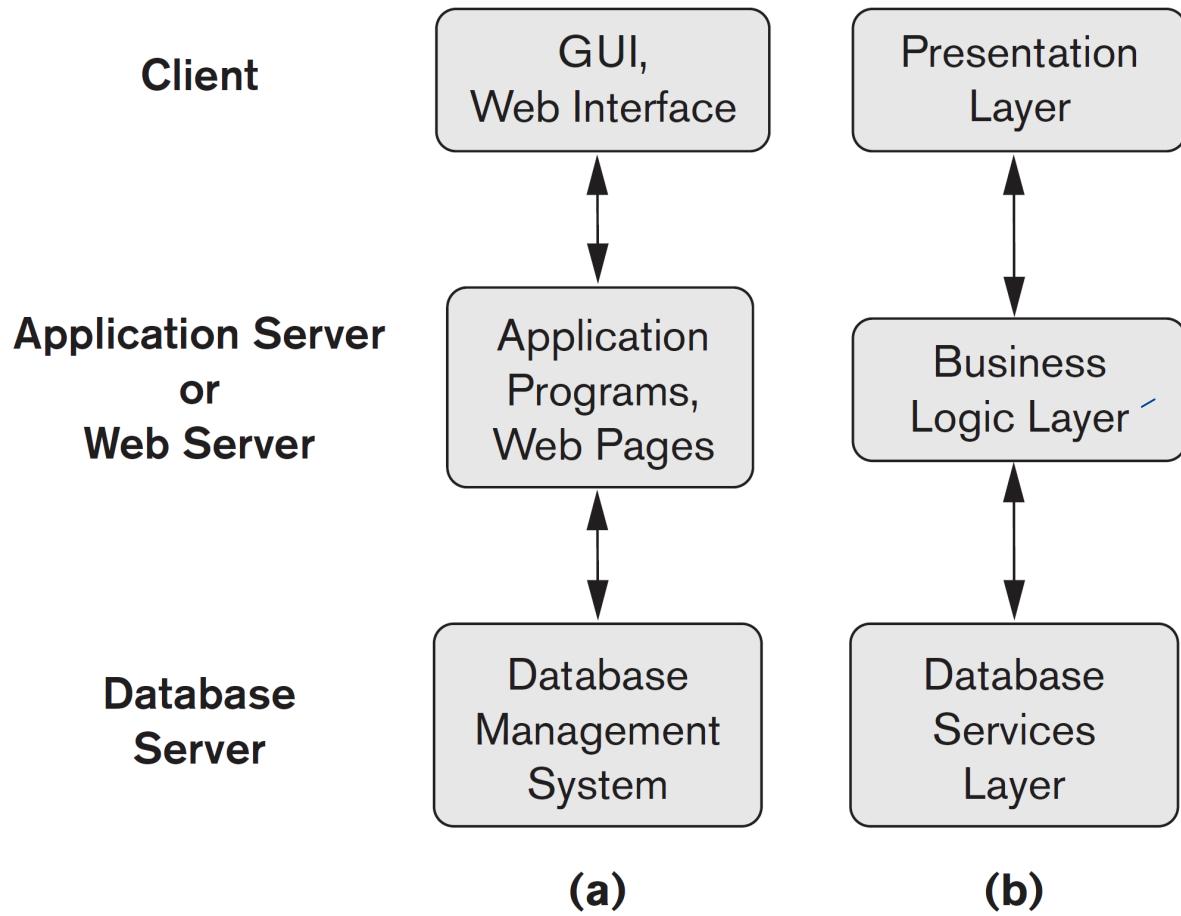


Figure 2.7
Logical three-tier client/server architecture, with a couple of commonly used nomenclatures.



- Common for Web applications
- Intermediate Layer called Application Server or Web Server:
 - Stores the web connectivity software and the business logic part of the application used to access the corresponding data from the database server
ກົດລັບໄປພາຍຫຼຸມ
ການເຊື່ອ
 - Acts like a conduit for sending partially processed data between the database server and the client.
ແກ່ທາງເຊື່ອ for DB ແລະ Client → ຂັ້ນ Processed data ຊົນ
- Three-tier Architecture Can Enhance Security:
 - Database server only accessible via middle tier
DB ນ້າກຳທີ່ຮັມກຳ middle tier ອໍານວຍເຕິ່ງ
 - Clients cannot directly access database server
Client ນັ້ນ ໄດ້ເຂົ້າໃຈ DB ຖ້າ
 - Clients contain user interfaces and Web browsers
Client ນັ້ນ [User Interface
Web Browsers]
 - The client is typically a PC or a mobile device connected to the Web
Client ສັນນິ [PC
mobile } ທີ່ເນັ້ນຕ້ອງວິນ

Classification of DBMSs

- Based on the data model used

- Legacy: ຕຳຫານ
 - Network, Hierarchical.
- Currently Used:
 - Relational, Object-oriented, Object-relational
- Recent Technologies:
 - Key-value storage systems, NOSQL systems: document based, column-based, graph-based and key-value based. Native XML DBMSs.
ລົ້ນກຳເພີດ

- Other classifications

ແບ່ງຕາມການໃຊ້ງານ

- Single-user (typically used with personal computers)
vs. multi-user (most DBMSs).
- Centralized (uses a single computer with one database)
vs. distributed (multiple computers, multiple DBs)

Cost considerations for DBMSs

- **Cost Range:** *พื้นที่ราคา*
 - from free open-source systems to configurations costing millions of dollars
- **Examples of free relational DBMSs:**
 - MySQL, PostgreSQL, others
- Commercial DBMS offer additional specialized modules,
ฟังก์ชันพิเศษ
 - e.g. time-series module, spatial data module, document module, XML module
 - These offer additional specialized functionality when purchased separately
 - Sometimes called cartridges (e.g., in Oracle) or blades
- **Different licensing options:**
 - site license, maximum number of concurrent users (seat license), single user, etc.

