Databases, Data Warehousing and Information Retrieval

Natalia Criado
Email: natalia.criado@kcl.ac.uk

Contact Details

- Natalia Criado
- Room: 6.27
- Contact Hours: Tuesday 12:00-14:00
- Email: natalia.criado@kcl.ac.uk

Module Overview

Three parts

- Database analysis and design (Entity Relationship Model, Normalization)
- Database implementation using SQL (Structured Query Language)
- Advanced topics: Data warehouses, Information Retrieval and no-SQL databases

Module Assessment

- Written Examination (January) 80%
- 2 Individual Assignments:
  - Database Implementation (10%)
    Hand Out: 3 October 2016
    Hand In: 31 October 2016
  - Database Design and Optimization (10%)
    Hand Out: 7 November 2016
    Hand In: 12 December 2016

Rules and Regulations

- Mobile Phones:
  - Switched off and in your bag/ pocket
- Timekeeping:
  - If you know you will be late, let me know beforehand
- I will treat your session as if we were in a professional environment

Principal Objectives

This module will allow you store data systematically employing modern database technologies, and will equip you with fundamental understanding and skills to independently study advanced data warehousing and information retrieval solutions.

Teaching and Learning Methods

Weekly teaching arrangements:

- Lecture + Tutorial (3 hours per week)
- Practical (2 hours per week)

The week plan is available on KEATS

Suggested Books

- Introduction to Database Systems (8th edition), by C.J. Date, Pearson Publishing, 2003.
- Fundamentals of Database Systems (7th edition), by Ramez Elmasri and Shamkant B. Navathe, Pearson Publishing, 2015.
- Data Warehousing in the Age of Big Data (1st edition), by Krish Krishnan, O'Reilly Media, 2013.
- Introduction to Information Retrieval (1st edition), by Christopher D. Manning, Prabhakar Raghavan and Hinrich Schtze, Cambridge University Press, 2008.

Session Objectives

In this session, you will learn:

- The difference between data and information
- What a database is, the various types of databases, and why they are valuable assets for data science
- How modern databases evolved from file systems
- The main components of the database system
- The main functions of a database management system (DBMS)

Why Databases?

- Databases solve many of the problems encountered in data management
- Used in almost all modern settings involving data management:
  - Business
  - Research
  - Administration
- Important to understand how databases work and interact with other applications

Exercise

- Find examples of situations in which you interact with databases on a daily basis

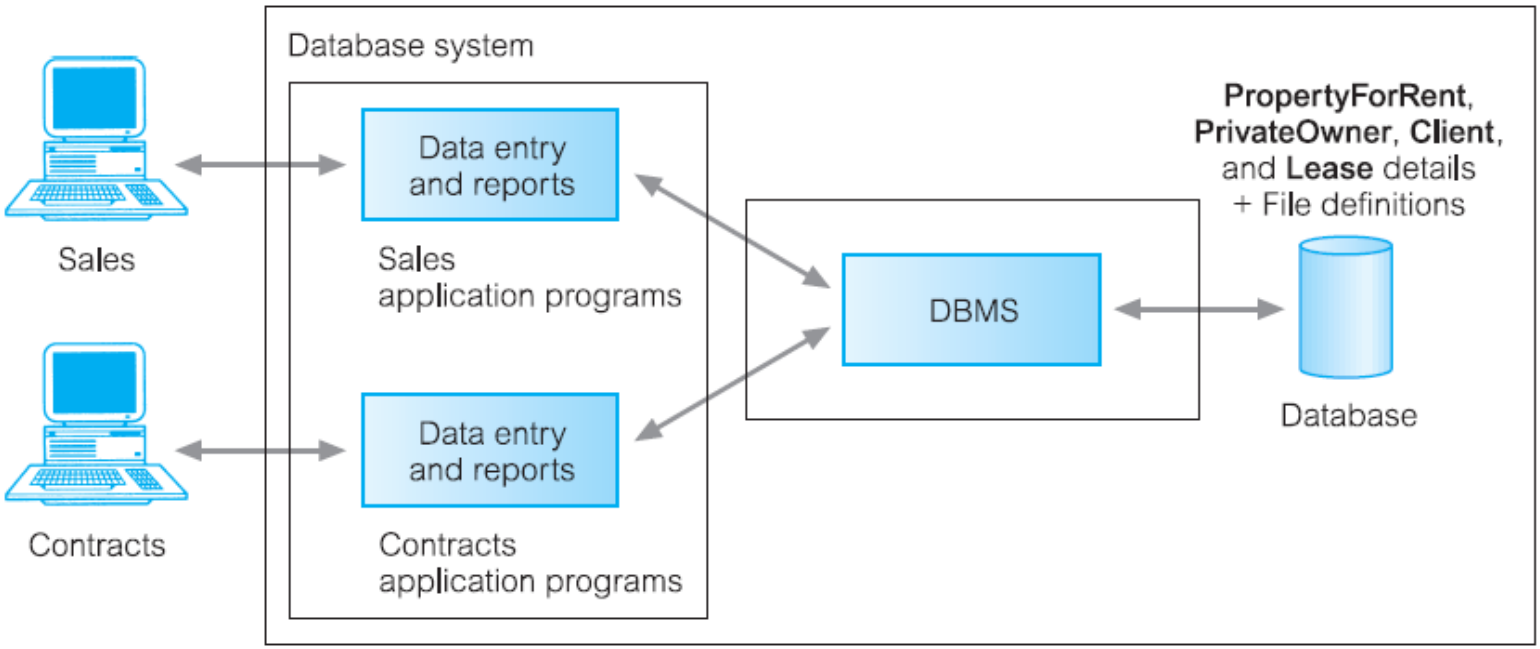
Data vs. Information

- Data are raw facts
- Information is the result of processing raw data to reveal meaning
  - Data: building blocks of information
- Information requires context to reveal meaning
- Data are the foundation of information, which is the bedrock of knowledge

Introducing the Database

- Database: shared, integrated computer structure that stores a collection of:
  - End-user data: raw facts of interest to end user
  - Metadata: data about data
    - Provides description of data characteristics and relationships in data
    - Complements and expands value of data
- Database management system (DBMS): collection of programs
  - Manages structure and controls access to data

Database Environment



File Systems

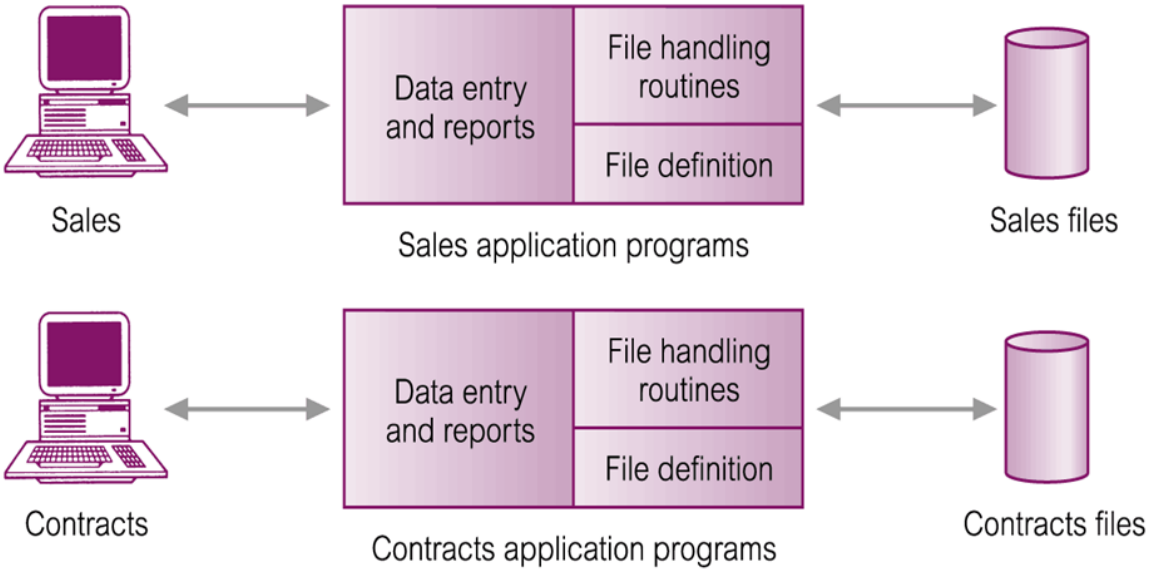


Figure 1.5 File-based processing.

Sales Files

PropertyForRent (propertyNo, street, city, postcode, type, rooms, rent, ownerNo)
PrivateOwner (ownerNo, fName, lName, address, telNo)
Client (clientNo, fName, lName, address, telNo, prefType, maxRent)

Contracts Files

Lease (leaseNo, propertyNo, clientNo, rent, paymentMethod, deposit, paid, rentStart, rentFinish, duration)
PropertyForRent (propertyNo, street, city, postcode, rent)
Client (clientNo, fName, lName, address, telNo)

Each program maintains its own set of data

Limitations of File Systems

- Separation and isolation of data
  - Users of one program may be unaware of potentially useful data held by other programs
- Duplication of data
- Data dependence (File structure is defined in the program code)
- Incompatible file formats
- Fixed Queries/Proliferation of application programs
- ...

Exercise

PROJ_NUM	PROJ_NAME	EMP_NUM	EMP_NAME	JOB_CODE	JOB_CHG_HOUR	PROJ_HOURS	EMP_PHONE
1	Hurricane	101	John D. Newson	EE	$85.00	13.3	653-234-3245
1	Hurricane	105	David F. Schwann	CT	$60.00	16.2	653-234-1123
1	Hurricane	110	Anne R. Ramoras	CT	$60.00	14.3	615-233-5568
2	Coast	101	John D. Newson	EE	$85.00	19.8	653-234-3254
2	Coast	108	June H. Sattlemeir	EE	$85.00	17.5	905-554-7812
3	Satellite	110	Anne R. Ramoras	CT	$62.00	11.6	615-233-5568
3	Satellite	105	David F. Schwann	CT	$26.00	23.4	653-234-1123
3	Satelite	123	Mary D. Chen	EE	$85.00	19.1	615-233-5432
3	Satellite	112	Allecia R. Smith	BE	$85.00	20.7	615-678-6879

What data redundancies do you detect? How could those redundancies lead to anomalies?

Database Systems

Database system consists of logically related data stored in a single logical data repository

- May be physically distributed among multiple storage facilities
- Eliminates most of file systems problems

Role of the DBMS

DBMS is the intermediary between the applications and the database. It enables:

- Defining (describing the structure)
- Constructing (populating by data)
- Manipulating (querying, updating)
- Preserving consistency
- Protecting from misuse (security, authentication)
- Recovering from failure
- Concurrent usage of a database

Types of Databases

- Number of users
- Database location(s)
- Expected type and extent of use

Number of Users

- Single-user database supports only one user at a time
  - Desktop database: single-user; runs on PC
- Multiuser database supports multiple users at the same time
  - Workgroup and enterprise databases

Location

- Centralized database: data located at a single site
- Distributed database: data distributed across several different sites

Usage

- Operational database: supports a companys day-to-day operations
  - Transactional or production database
- Data warehouse: stores data used for tactical or strategic decisions

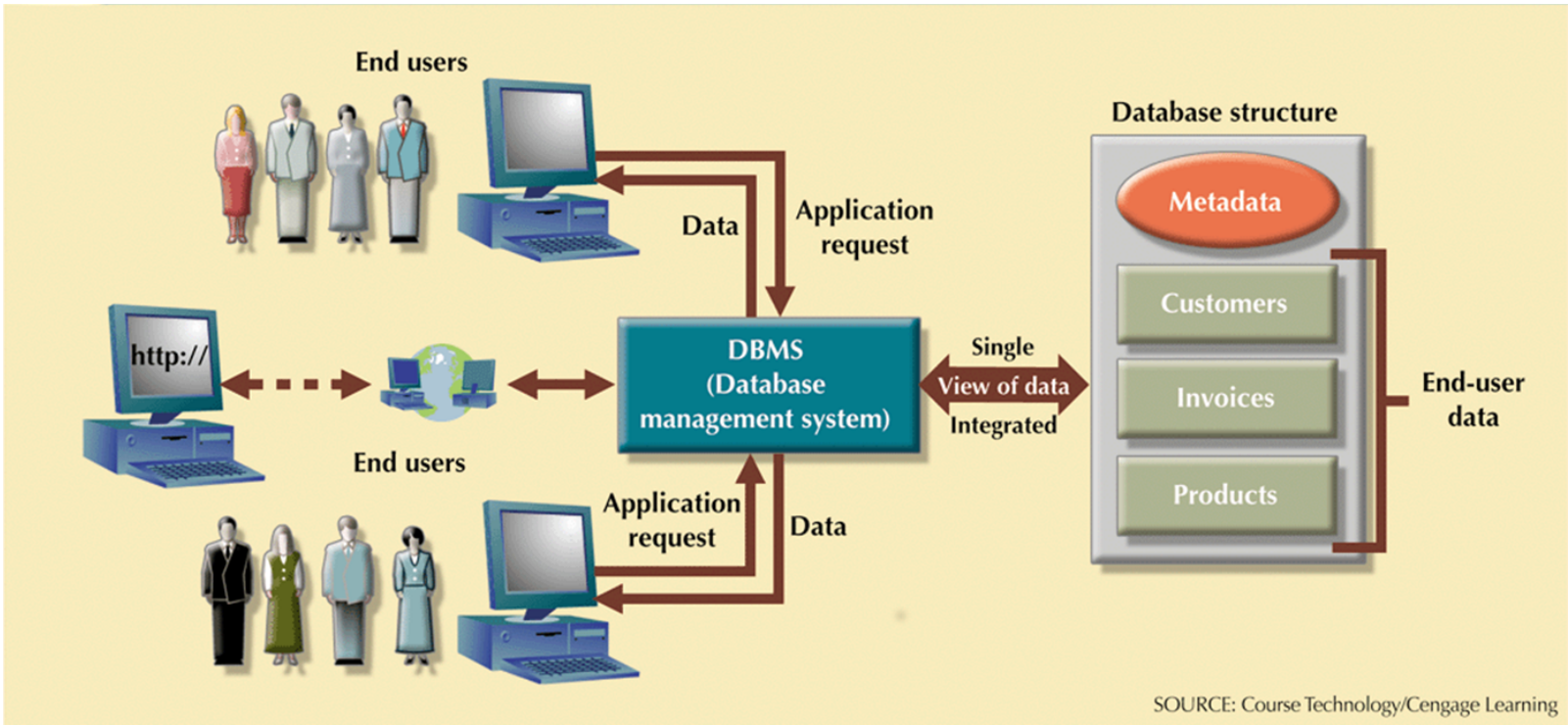
Types of Databases

PRODUCT	NUMBER OF USERS			DATA LOCATION		DATA USAGE		XML
	SINGLE USER	MULTIUSER						
		WORKGROUP	ENTERPRISE	CENTRALIZED	DISTRIBUTED	OPERATIONAL	ANALYTICAL	
MS Access	X	X		X		X		
MS SQL Server	X[3]	X	X	X	X	X	X	X
IBM DB2	X[3]	X	X	X	X	X	X	X
MySQL	X	X	X	X	X	X	X	X
Oracle RDBMS	X[3]	X	X	X	X	X	X	X

Database Languages

- **Data Definition Language (DDL)** used to specify the database structure
- **Data Manipulation Language (DML)** used to both read and update the database:
  - The part of a DML that involves data retrieval is called a query language

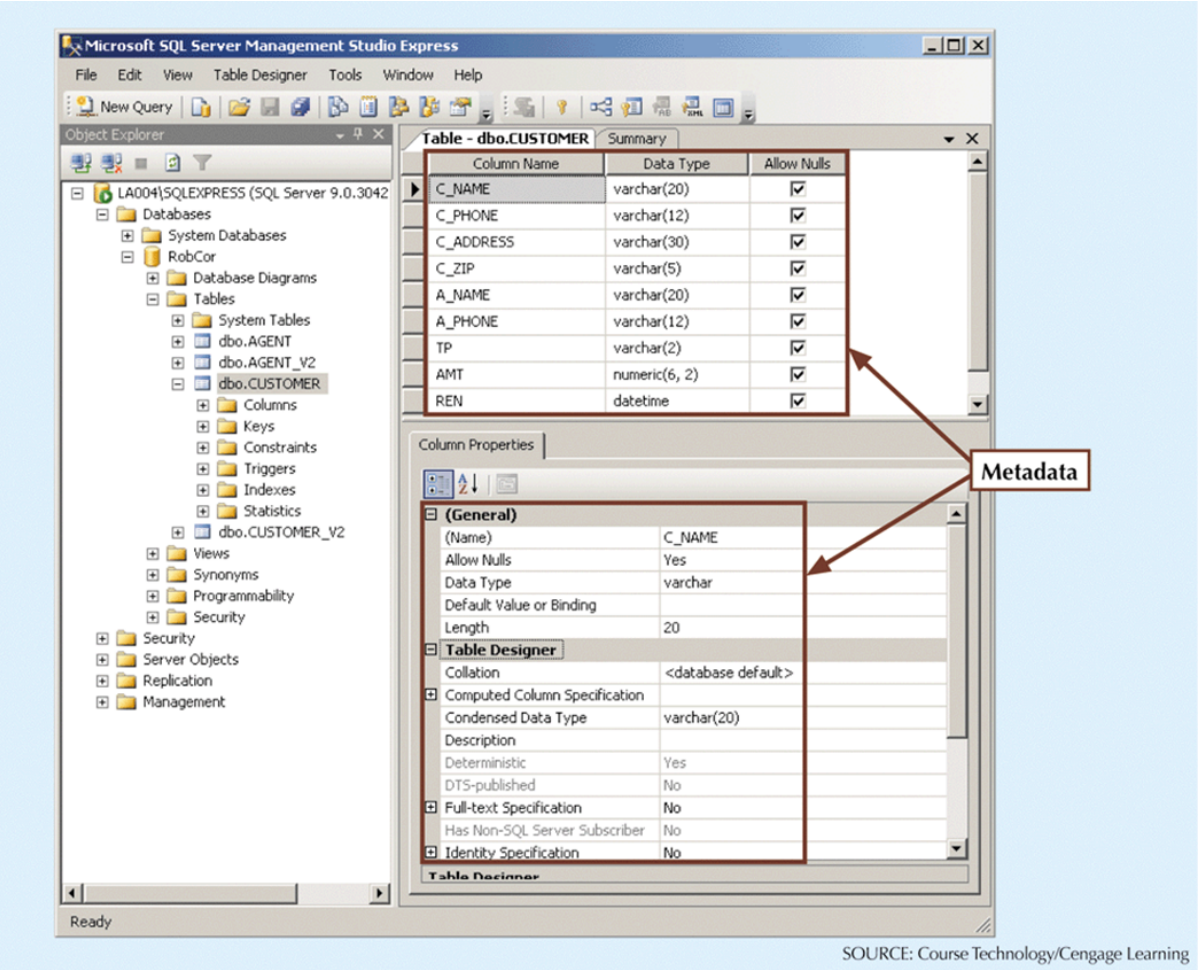
Database management system (DBMS)



DBMS Functions I

- Most functions are transparent to end users
  - Can only be achieved through the DBMS
- Data dictionary management
  - DBMS stores definitions of data elements and relationships (metadata) in a data dictionary
  - DBMS looks up required data component structures and relationships
  - Changes automatically recorded in the dictionary
  - DBMS provides data abstraction and removes structural and data dependency

DBMS Functions I

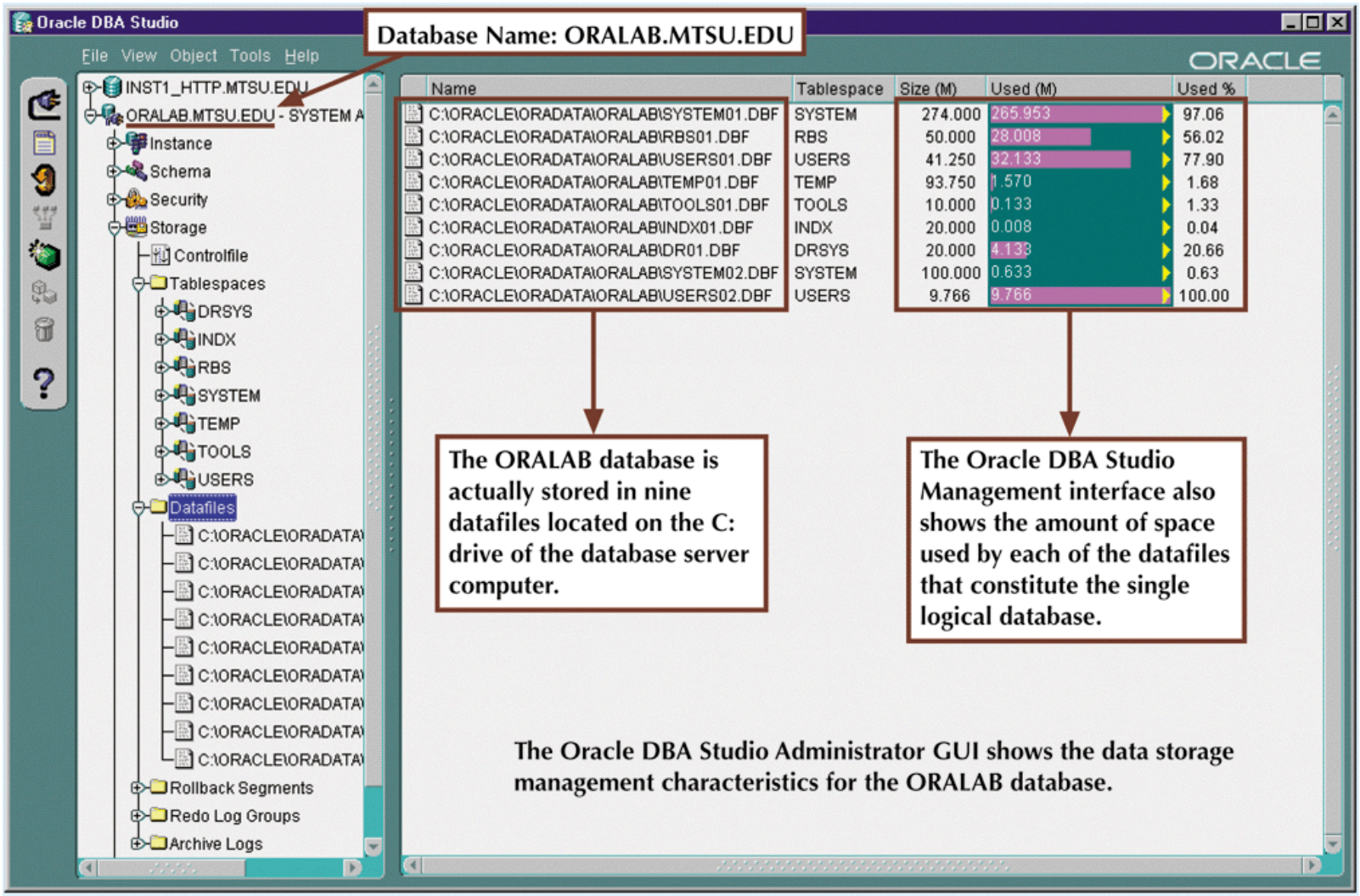


SOURCE: Course Technology/Cengage Learning

DBMS Functions II

- Data storage management
  - DBMS creates and manages complex structures required for data storage
  - Also stores related data entry forms, screen definitions, report definitions, etc.
  - Performance tuning: activities that make the database perform more efficiently
  - DBMS stores the database in multiple physical data files

DBMS Functions II



DBMS Functions III

- Data transformation and presentation
  - DBMS transforms data entered to conform to required data structures
  - DBMS transforms physically retrieved data to conform to users logical expectations
- Security management
  - DBMS creates a security system that enforces user security and data privacy
  - Security rules determine which users can access the database, which items can be accessed, etc.

DBMS Functions IV

- Multiuser access control
  - DBMS uses sophisticated algorithms to ensure concurrent access does not affect integrity
- Backup and recovery management
  - DBMS provides backup and data recovery to ensure data safety and integrity
  - Recovery management deals with recovery of database after a failure
    - Critical to preserving databases integrity

DBMS Functions V

- Data integrity management
  - DBMS promotes and enforces integrity rules
    - Minimizes redundancy
    - Maximizes consistency
  - Data relationships stored in data dictionary used to enforce data integrity
  - Integrity is especially important in transaction-oriented database systems

DBMS Functions VI

- Database access languages and application programming interfaces
  - DBMS provides access through a query language
  - Query language is a nonprocedural language
  - Structured Query Language (SQL) is the de facto query language
    - Standard supported by majority of DBMS vendors

DBMS Functions VII

- Database communication interfaces
  - Current DBMSs accept end-user requests via multiple different network environments
  - Communications accomplished in several ways:
    - End users generate answers to queries by filling in screen forms through Web browser
    - DBMS automatically publishes predefined reports on a Web site
    - DBMS connects to third-party systems to distribute information via e-mail

Exercise

How do you convince a group of friends who run a small business using a file-based approach data management that they should manage their business data using database technology?

Advantages of database systems

- Improved data sharing
- Improved data security
- Better data integration
- Minimized data inconsistency
- Improved data access
- Improved decision making
- Increased end-user productivity

Disadvantages of database systems

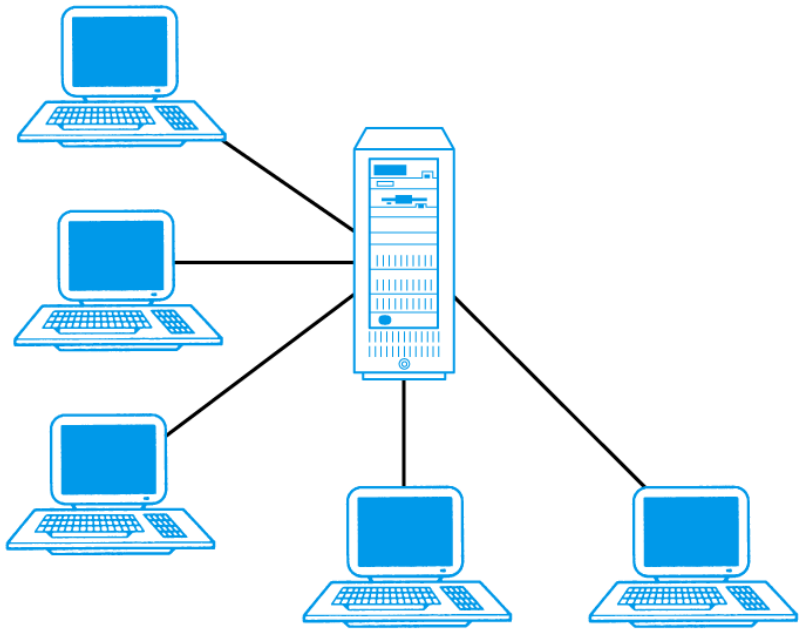
- Increased costs
- Complexity
- Vendor dependence
- Frequent upgrade/replacement cycles

Multi-User DBMS Architectures

- Tele-processing
- File Server
- Two-Tier Client-Server
- Three-Tier Client-Server

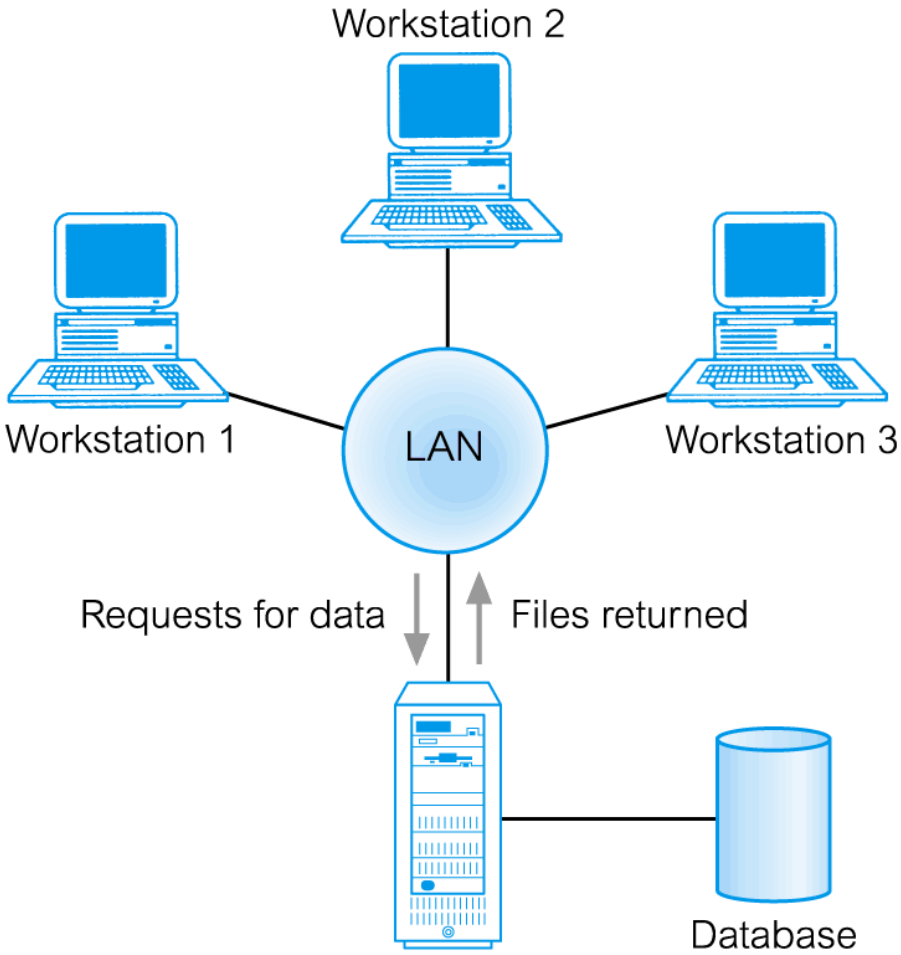
Tele-processing

Single mainframe with a number of terminals attached



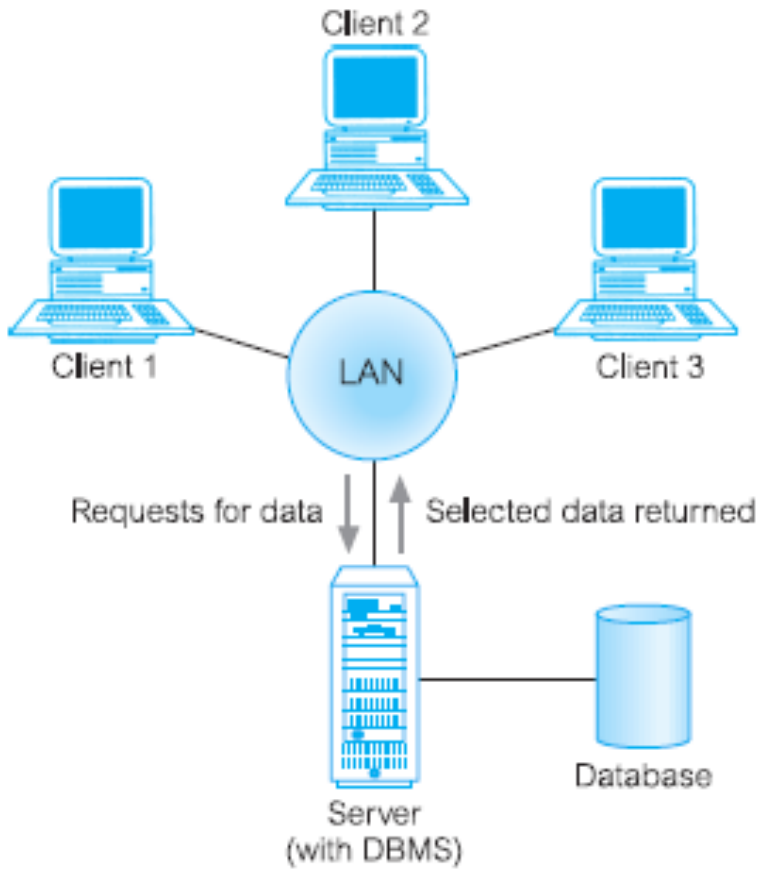
File Server

- Database resides on file-server.
- DBMS and applications run on each workstation



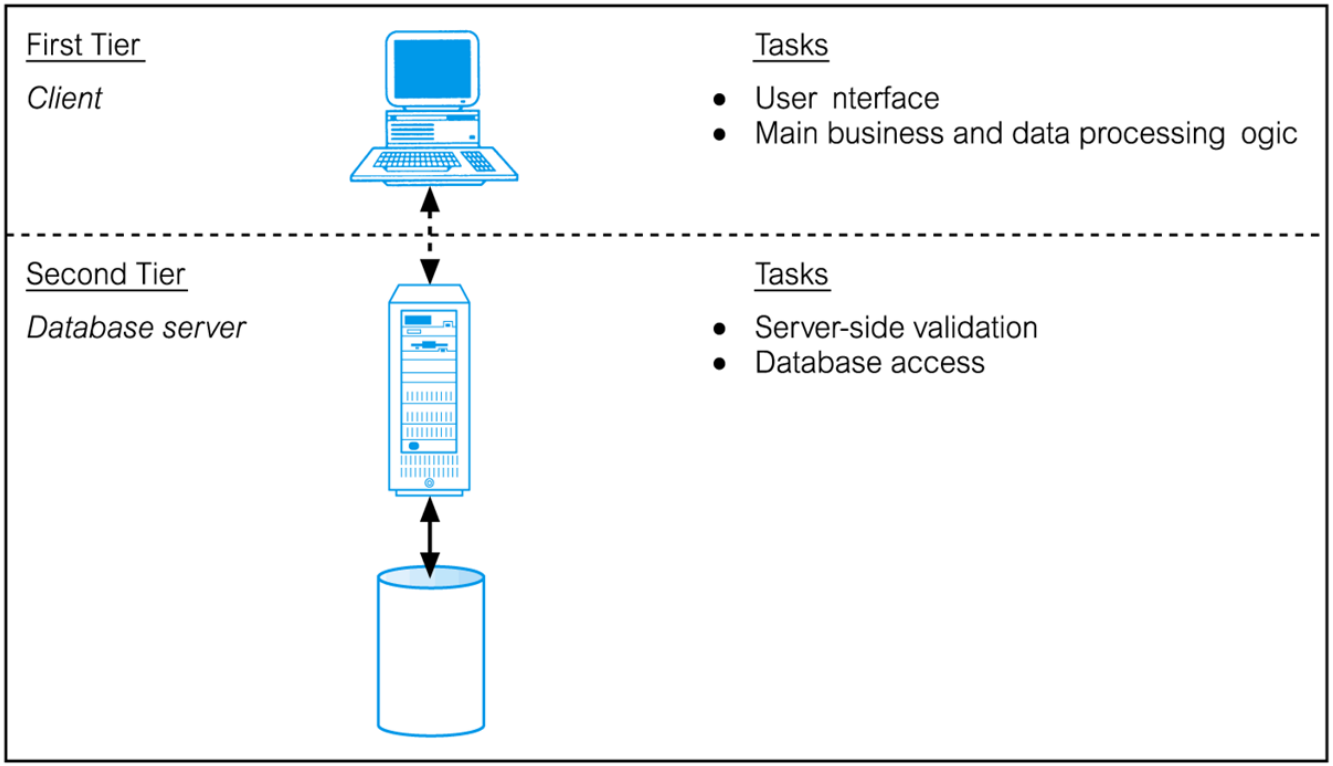
Client-Server Architecture

- Client-server refers to the way in which software components interact to form a system.
- A client process requires some resource, and a server provides the resource



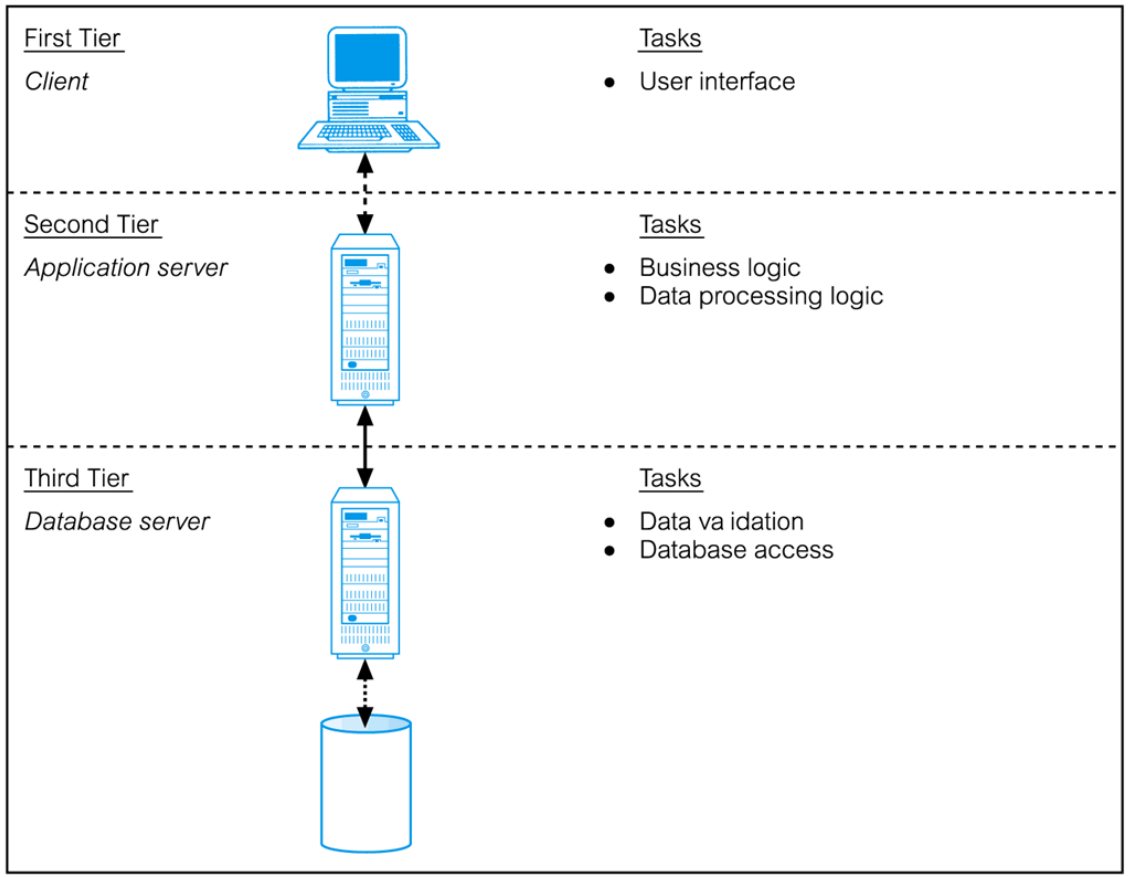
Two-Tier Client-Server

- Client (tier 1) manages user interface and runs applications
- Server (tier 2) holds database and DBMS



Three-Tier Client-Server

- Thin Client (tier 1) manages user interface
- Application Server (tier 2) runs applications
- Database Server (tier 3) holds database and DBMS



Data Models

- Data model is an abstraction
- Data models:
  - Relatively simple representations of complex real-world data structures
  - Often graphical

Evolution of Data Models

GENERATION	TIME	DATA MODEL	EXAMPLES	COMMENTS
First	1960s–1970s	File system	VMS/VSAM	Used mainly on IBM mainframe systems Managed records, not relationships
Second	1970s	Hierarchical and network	IMS, ADABAS, IDS-II	Early database systems Navigational access
Third	Mid-1970s	Relational	DB2 Oracle MS SQL Server MySQL	Conceptual simplicity Entity relationship (ER) modeling and support for relational data modeling
Fourth	Mid-1980s	Object-oriented Object/ relational (O/R)	Versant Objectivity/DB DB2 UDB Oracle 11g	Object/relational supports object data types Star Schema support for data warehousing Web databases become common
Fifth	Mid-1990s	XML Hybrid DBMS	dbXML Tamino DB2 UDB Oracle 11g MS SQL Server	Unstructured data support O/R model supports XML documents Hybrid DBMS adds object front end to relational databases Support large databases (terabyte size)
Emerging Models: NoSQL	Late 2000s to present	Key-value store Column store	SimpleDB (Amazon) BigTable (Google) Cassandra (Apache)	Distributed, highly scalable High performance, fault tolerant Very large storage (petabytes) Suited for sparse data Proprietary API

University Database

STUDENT

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS

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	07	King
92	CS1310	Fall	07	Anderson
102	CS3320	Spring	08	Knuth
112	MATH2410	Fall	08	Chang
119	CS1310	Fall	08	Anderson
135	CS3380	Fall	08	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

Questions

1. What is this database about?
2. What data records are stored in each table?
3. Which data elements are stored for each record in each table?
4. What are the data types of each data element?
5. Are there any relationships between the records on different tables?

MySQL

- MySQL is a DBMS
- MySQL is relational (more about this next week)
- MySQL software is Open Source (You can install it on your devices)
- The MySQL Database Server is very fast, reliable, scalable, and easy to use

Conclusion

In this session we have covered:

- File Systems
- Database
  - Definition
  - Types
  - Languages
  - DBMS
- MySQL

Practical Session

The practical session this week is about getting stared with Unix

Please don't miss it!!!
you need it for the rest of the semester

Suggested Readings

- Chapters 1 and 2 of Fundamentals of Database Systems. Elmasri & Navathe.
- Chapters 1 and 2 of Database systems: a practical approach to design, implementation, and management. Connolly, Thomas M; Begg, Carolyn