

# **Database Management Systems**

**COP 5725**

**Dr. Markus Schneider**

Department of Computer and Information Science and Engineering  
CSE Building, Room E450  
University of Florida

# Welcome

- ❑ Database systems are software systems for managing large volumes of data
- ❑ Industry sector valued at between US\$35-US\$50 billion annually
- ❑ **Relational database systems** (still) dominate the database sector
- ❑ Application examples
  - ❖ Sales: Customer, product, purchase information
  - ❖ Accounting: Payments, receipts, account balances, assets
  - ❖ Human resources: Employee data, salaries, payroll taxes, benefits
  - ❖ Banking and finance: Customer information, accounts, loans, banking transactions, credit card purchases, generation of monthly statements, stock and bond purchases
  - ❖ Universities: Student information, course registrations, grades
  - ❖ Airlines: Reservations, schedule information
  - ❖ Telecommunications: Keeping records of calls made, monthly phone bills
  - ❖ Web-based applications on the Internet

# Course Contents and Objectives (I)

- ❑ Understand the differences between file systems and database systems
- ❑ Understand the features and advantages of database systems
- ❑ Perform conceptual database design based on the Entity-Relationship (E-R) model
- ❑ Understand the relational data model
- ❑ Be able to transform an Entity-Relationship diagram into a relational database schema
- ❑ Comprehend the Relational Algebra and be able to formulate queries as algebraic expressions
- ❑ Learn the SQL query language (both relational and object-oriented components) and be able to formulate SQL queries of high complexity

## Course Contents and Objectives (II)

- ❑ Comprehend the foundations of **relational database design theory** including **functional dependencies** and **normalization** as well as appreciate their relevance for database practice
- ❑ Learn possible technologies for **database application programming**
- ❑ Understand the main concepts and benefits of **object-relational database systems** and their difference to purely relational database systems
- ❑ Obtain general knowledge of **advanced database systems**
- ❑ Work and get experience with a commercial database system (e.g., Oracle)
- ❑ Design and implement a sophisticated **web-based database application** project that includes database design, application user interface design, and their web-based connection
- ❑ Learn to work in a project group, argue, discuss, compromise, write technical documents, and solve arising social conflict situations at a professional level

# General Information (I)

- ❑ Credits: 3
- ❑ Prerequisites
  - ❖ General knowledge of data structures and algorithms
  - ❖ Web programming skills, e.g., PHP (if you do not have them, you must obtain them in self-study)
- ❑ Instructor's email address: *mschneid@cise.ufl.edu*
- ❑ Use prefix [COP 5725] in subject line of emails
- ❑ COP 5725 course material is available in Canvas (<http://elearning.ufl.edu/>)
- ❑ Syllabus available in Canvas

## General Information (II)

- ❑ Messages to the class are communicated by the class mailing list
  
- ❑ Apply for a CISE Oracle account as fast as possible
  - ❖ Needed for the group project and the homework assignments
  - ❖ CISE account required
  - ❖ If you are not a CISE student, get a guest account
  - ❖ [CISE Help Pages](#) have all the needed information
  
- ❑ In which situation should you ask whom for advice?
  - ❖ Lecture questions and project questions: me and the TA
  - ❖ Homework assignments and exams: the TA and me
  - ❖ Regrading requests: grader(s), in case of lacking agreement: me
  - ❖ Oracle contents questions: TA
  - ❖ Oracle system questions: `consult@cise.ufl.edu` (system administration)

# Textbooks

## Required textbooks

- ❑ R. Elmasri & S. B. Navathe. *Fundamentals of Database Systems*. Addison Wesley, 7th edition, 2016.
- ❑ T. Connolly & C. Begg. *Database Systems - A Practical Approach to Design, Implementation, and Management*. Addison Wesley, 6th edition, 2014

## Recommended textbooks

- ❑ A. Silberschatz, H. F. Korth & S. Sudarshan. *Database System Concepts*. McGraw Hill, 6th edition, 2010.
- ❑ H. Garcia-Molina, J. D. Ullman & J. Widom. *Database Systems: The Complete Book*. Prentice Hall, 2nd edition, 2008.

# Course Assessment (I)

## Exams

- ☐ 3 exams
- ☐ Exam 1 after one third of the semester, Exam 2 after two thirds of the semester, Exam 3 at the end of the semester
- ☐ All three exams with closed notes and closed books
- ☐ Failure to take an exam will lead to 0 points in that exam
- ☐ Each student who wants to take an exam must be in the classroom **before** the exam begins; exam sheets are not handed out to delayed students
- ☐ Contents of the exams is not accumulative but limited to the previous third of the course
- ☐ Knowledge questions and applied exercises
- ☐ Sometimes creative questions will be asked
- ☐ Mathematical concepts and notations are part of the exams



# Course Assessment (II)

## Homework assignments

- ☐ 5 homework assignments
- ☐ Important to learn and deepen the student's knowledge and as a preparation for exams
- ☐ All assignments must be solved by each student **individually**.
- ☐ Working in groups on homework assignments is **not** allowed.
- ☐ Assignments consist of
  - ❖ written parts
  - ❖ practical parts performed with CISE Oracle (original snapshots required)
- ☐ Homework submission in Canvas as a PDF file
- ☐ Failure to submit an assignment before the announced deadline will lead to 0 points for that assignment
- ☐ No late policy

## Course Assessment (III)

### Group software project

- ❑ 3 project deliverables and 1 group software demonstration at the end of the semester
- ❑ 3 to 4 students per group cooperate on a semester-long database project
- ❑ Each group member will get the same group grades
- ❑ Goal: Design and implementation of a **web-based database application program** to give the student some real “hands-on” experience with a full-fledged database management system
- ❑ We will use the (CISE) Oracle database system, otherwise large flexibility
- ❑ Required: web-based programming language (e.g., PHP, Javascript), not taught in class
- ❑ Group software demonstrations will last 30 minutes and be conducted online with a web-based presentation tool that each group selects on its own

## Course Assessment (IV)

### Grading

- ❑ During the semester a student can only earn *points* and *not* grades
- ❑ 100 points are the maximum that can be achieved in any exam, homework assignment, and project deliverable. These points are relative and unweighted.
- ❑ The table below shows the *weighted percentage* of and the *maximum absolute point number* for each exam, each project deliverable, the project software, and each homework assignment.

Exam		Project		Homework	
Exam 1	15	Project Deliverable 1	4	Homework 1	6
Exam 2	15	Project Deliverable 2	4	Homework 2	6
Exam 3	15	Project Deliverable 3	1	Homework 3	6
		Project Software	16	Homework 4	6
				Homework 5	6

- ❑ Maximum number of absolute points in this course: 100

# Course Assessment (V)

## Grading (*continued*)

- ❑ Formula of the student's overall performance computation can be found in the syllabus
- ❑ Each student will obtain information about the overall course performance and corresponding grade based on the exams, homework assignments, and project deliverables that have been graded at any time during the semester
- ❑ Each student will be enabled to compare his/her performance to the other students in class
- ❑ Each student will obtain an overview of the overall class performance based on the graded items so far at any time during the semester
- ❑ Regrading policy
  - ❖ Regrading request allowed within 5 workdays after the return of a graded item
  - ❖ Regrading of an item not possible outside the regrading period

## Course Assessment (VI)

### Grading (*continued*)

- ❑ Mapping of a student performance to a letter grade

Student's Performance in % or absolute points	Grade
>94-100	A
>88-94	A–
>82-88	B+
>76-82	B
>70-76	B–
>64-70	C+
>58-64	C
>52-58	C–
>46-52	D+
>40-46	D
>34-40	D–
0-34	E

## Other Important Topics

### ❑ Academic Honesty

- ❖ Student Honor Code see syllabus
- ❖ In our class
  - Homework assignments have to be performed *individually* and *not in groups*
  - Project deliverables have to be performed *in groups* and *not individually*

### ❑ Students with disabilities and/or special needs

- ❖ Register with the Disability Resource Center (DRC) of the Dean of Students Office (DSO) at the beginning of the semester
- ❖ DRC will provide documentation to the student
- ❖ Documentation must be shown to the instructor

### ❑ Online evaluation of the quality of instruction at the end of the semester at <https://evaluations.ufl.edu>.

# How to Succeed in this Course

- ☐ Attend all lectures
- ☐ Skim the assigned sections of the textbooks *before* attending a lecture
- ☐ Read and study the assigned sections of the textbooks *after* a lecture together with the lecture slides. The textbook sections give you another wording and perspective of the lecture contents.
- ☐ Start early on your homework assignments and software project
- ☐ If you are having difficulties, you owe it to yourself to get help. Talk then to the instructor and the teaching assistant.
- ☐ If you work hard and master the material presented in this class, you will learn some powerful, fundamental concepts of database management systems as well as application development skills, which are very marketable in today's high-tech industry.
- ☐ The instructor will try his best to make the course as interesting and stimulating as possible and an enriching experience for you.

## What I expect from you

- ☐ Read the syllabus
- ☐ Attend the class regularly
- ☐ Study and learn the material presented in the class *and* in the reference book
- ☐ Learn also explanations and definitions given in mathematical terms
- ☐ Perform well in the exams
- ☐ Do the homework (alone)
- ☐ Do the software project (in a team). Be a team player.
- ☐ Don't ask for extra treatment, extra credit, and exceptions. I won't grant them.
- ☐ Don't cheat

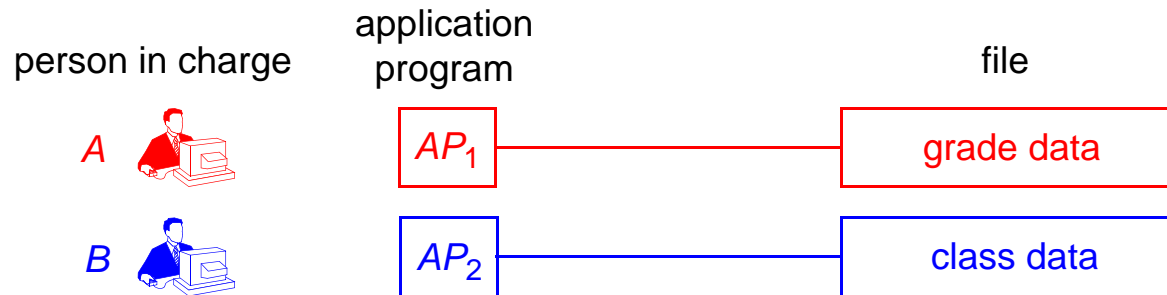


# 1. Introduction

## 1.1 Traditional Data Management Using File Systems

A time when there were no database systems ...

❑ scenario „university administration“



❑ assumptions

- files contain the last name, the first name, the registration number and the address of a student
- grade data additionally include the results of oral exams, written exams, seminars, etc.
- class data additionally comprise the taken classes of each student

## Problems of early data processing

### ❑ **redundancy**

- repeated occurrence of the same data in different files
- waste of external memory, increased management and processing costs

### ❑ **inconsistency**

- lacking logical concordance of file contents
- especially caused due to changes

### ❑ **data-program dependence**

- data are directly created and accessed by an application program
- changes of the file structure lead to changes of the application program
- extensions of the functionality of an application program lead to new requirements of the file structure and to a restructuring of files

### ❑ **inflexibility**

- analysis of data as well as the realization of new applications is problematic
- data from several files can only be combined with very high costs