COLORADO TECHNICAL UNIVERSITY
FOUNDED 1965

FUNDAMENTALS OF
DATABASE SYSTEMS

CS660 Week 1: Core Database Concepts

Dr. John Conklin

UNIT 1 OVERVIEW

1. **Database Systems**
   Core concepts, architecture, and DBMS functionality

2. **Object-Oriented & Object-Relational Models**
   Advanced database paradigms and their applications

3. **The Relational Database Model**
   Tables, keys, relationships, and normalization

4. **Aligning Database Design With Business**
   Critical success factors and strategic system design

DATABASE SYSTEMS

What is a Database?

A database is a shared, integrated, computer-based repository of data that is organized in a manner that facilitates the efficient retrieval, manipulation, and storage of information for multiple users and applications.

Key Characteristics

- Shared Resource

Multiple users can access simultaneously

- Integrated Data

Related data stored together

- Data Independence

Applications independent of storage

Core Functions

- Data Storage & Retrieval

Efficient access to large datasets

- Data Security

Control access and protect integrity

- Concurrent Access Control

Manage multiple simultaneous users

DATABASE MANAGEMENT SYSTEMS

DBMS Components

User Interface Layer

- Applications
- Query Tools
- Report Generators

Logic Layer

- Query Processor
- Transaction Manager
- Security Module

Storage Layer

- File Manager
- Buffer Manager
- Physical Storage

DATABASE VS FILE SYSTEMS

Traditional File Systems

- Data Redundancy
  Same data stored multiple times

- Data Inconsistency
  Updates not synchronized

- Limited Data Sharing
  Difficult concurrent access

- Data Dependency
  Changes require app modifications

Database Systems

- Data Integrity
  Centralized control ensures accuracy

- Reduced Redundancy
  Data stored once, referenced many

- Concurrent Access
  Multiple users simultaneously

- Data Independence
  Apps unaffected by storage changes

OBJECT-ORIENTED
DATABASE MODEL

An object-oriented database (OODB) stores data as objects—encapsulated entities that contain both data (attributes) and the methods (functions) that operate on that data. This model supports inheritance, polymorphism, and complex data types.

Core OO Concepts

- Objects & Classes

Data + behavior bundled together

- Encapsulation

Hide internal implementation details

- Inheritance

Derive new classes from existing ones

Advantages

- Complex Data Support

Multimedia, spatial, engineering data

- Closer to Real-World

Models entities more naturally

- Better Performance

For specific complex applications

OBJECT-RELATIONAL
DATABASE MODEL

Object-relational databases (ORDB) extend traditional relational databases with object-oriented features. They combine SQL's querying power with support for complex data types, user-defined types, inheritance, and methods—bridging the gap between relational and object-oriented paradigms.

Hybrid Capabilities

User-Defined Types	Create complex data types beyond basic SQL types
Table Inheritance	Tables can inherit structure from parent tables
Array & Nested Types	Store collections and complex structures in columns
Methods on Data	Define functions that operate on custom types

THE RELATIONAL DATABASE MODEL

Foundation of Modern Databases

The relational model organizes data into tables (relations) composed of rows (tuples) and columns (attributes). Introduced by E.F. Codd in 1970, it remains the dominant database model due to its mathematical foundation, simplicity, and powerful query capabilities through SQL.

T

Tables (Relations)

Two-dimensional structure storing data

R

Rows (Tuples)

Individual records or entries

C

Columns (Attributes)

Data fields with specific types

K

Keys

Unique identifiers and relationship links

KEYS AND RELATIONSHIPS

Types of Keys

Primary Key	Uniquely identifies each row
Foreign Key	References primary key in another table
Candidate Key	Potential primary key options
Composite Key	Multiple columns forming unique identifier

Relationship Types

One-to-One (1:1)
Each record in Table A relates to one record in Table B
Example: Person ↔ Passport

One-to-Many (1:N)
One record in Table A relates to many in Table B
Example: Customer ↔ Orders

Many-to-Many (M:N)
Multiple records relate to multiple records
Example: Students ↔ Courses (via enrollment)

DATABASE NORMALIZATION

Organizing data to reduce redundancy and improve integrity

1NF

First Normal Form

Eliminate repeating groups; atomic values only

2NF

Second Normal Form

Meet 1NF; remove partial dependencies on composite keys

3NF

Third Normal Form

Meet 2NF; remove transitive dependencies

BCNF

Boyce-Codd Normal Form

Stricter version of 3NF; every determinant is a candidate key

ALIGNING DATABASE DESIGN WITH COMPANY GOALS

Strategic Alignment

Database design must directly support organizational objectives, operational efficiency, and competitive advantage. Poor alignment leads to systems that hinder rather than enable business processes.

Key Alignment Factors

Business Requirements

Understand what the organization needs to achieve

Performance Needs

Balance normalization with query performance

Data Governance

Establish ownership, quality standards, and policies

Integration Requirements

Connect with existing systems and future platforms

Scalability Planning

Design for growth in data volume and user base

Security & Compliance

Meet regulatory and industry standards

CRITICAL SUCCESS FACTORS

Elements that determine database system success

Stakeholder Involvement

Engage users and decision-makers throughout design process

Proper Modeling

Accurate ER diagrams and normalized schema design

Clear Requirements

Document business rules and data needs comprehensively

Performance Testing

Validate system meets speed and scalability requirements

Skilled Team

Experienced database designers and administrators

Change Management

Plan for data migration and user training

SYSTEM DESIGN PROCESS

1	**Requirements Analysis**	Gather business needs and data requirements
2	**Conceptual Design**	Create ER diagrams and define entities
3	**Logical Design**	Convert to relational schema with normalization
4	**Physical Design**	Implement storage, indexes, and optimization
5	**Implementation**	Build database and populate with data
6	**Testing & Refinement**	Validate performance and iterate

KEY TAKEAWAYS

- Database systems provide structured, efficient data management
- Object-oriented and object-relational models extend traditional databases
- Relational model remains dominant with tables, keys, and normalization
- Database design must align with organizational goals and strategy
- Success depends on requirements analysis, skilled design, and stakeholder buy-in

Master these fundamentals to build robust database solutions! 🎯

INDIVIDUAL PROJECT

Individual Project

The retail store has experienced significant growth in the recent months. The customers have expressed an interest in being able to purchase the retail store's products electronically. Currently, the retail store has a simple Web site, but it is not clear that the system can support dynamic interaction on the Web site or online purchases. The main database system that is used in the physical store has not been working correctly, however, and this is an issue. The retail store wants to be able to eventually integrate an online database system with the current order entry system that is used to support sales made in the local store. The retail store also needs the database system to enforce the rules that have been established for their business practices. What can be done to support the goals and objectives for meeting the needs of their customers?

- It is assumed that Microsoft SQL Server Express Edition (with management tools) was chosen as the database platform. It can be downloaded for free at this Web site.
- You have been asked to develop a solution in the form of a Database System Development Plan for your retail store using the following format:

- Database System Development and Implementation Plan
    - Use Microsoft Word
    - Title page
        - Should include information I provided in the announcements section. (**APA 7 Format Example)**
    - Table of contents (TOC)
        - **Use auto-generated TOC**
        - Maximum of 3 levels deep
        - Be sure to update the fields of the TOC so it is up-to-date before submitting your project.
    - Section headings (create each heading on a new page with TBD as content except for sections listed under new content below)

The project deliverables are as follows:

- A description of the general business environment for the case study organization (give me the name of the store, its location, how many employees work there, and the market you serve)
- A database system’s goals and objectives statement
- A narrative that discusses how the proposed database system addresses the presenting business problem(s)
- A narrative that demonstrates how the proposed system aligns with the mission statement and strategic goals of the organization
- The student's analysis as to how this part of the project fulfills the mission and/or one or more goals of the case study organization
- All sources should be cited in-text, and references should be made using **APA 7 format**.
- Name the document "**yourname_CS660_IP1**"

Contact Information

Email:	Jconklin@coloradotech.edu
Phone:	602.796.5972
Website:	http://drjconklin.com
Office Hours:	Wednesdays: 6:00 PM – 7:00 PM (CST)
	Saturdays: 11:00 AM – 12:00 PM (CST)
Live Chats:	Wednesdays: 6:00 PM – 7:00 PM (CST)