

Introduction and Course Information

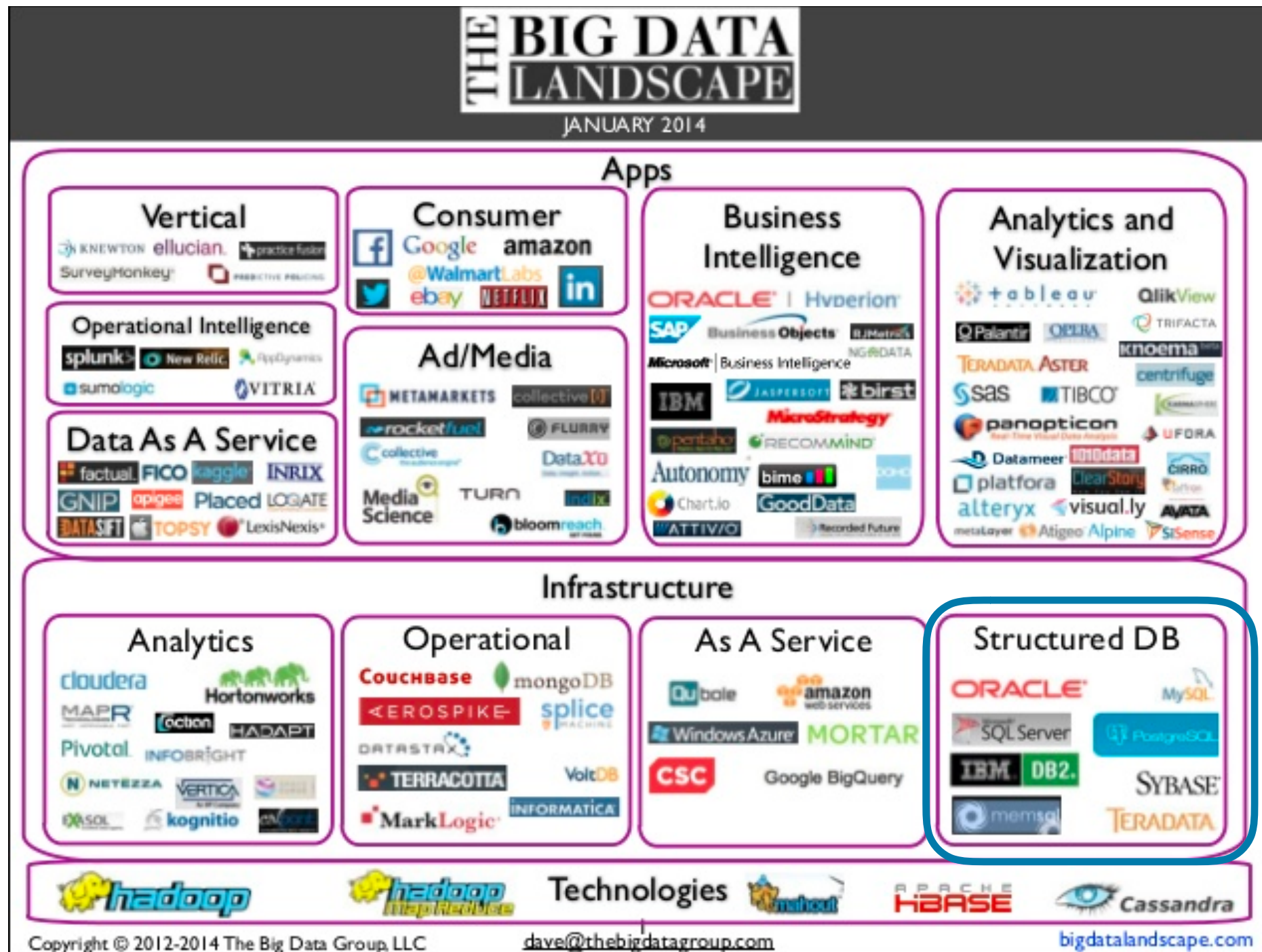
CS 377: Database Systems

Why Should I Care?

- Data can be found anywhere and everywhere
- Data management is necessary for:
 - Scientific discoveries
 - Service industries
 - Decision makers
- Databases are the core technology



Current Landscape



Course Overview

- Fundamental concepts of database and database systems
 - Modeling and design
 - Database programming
- Preview of database implementation
- Selected additional topics



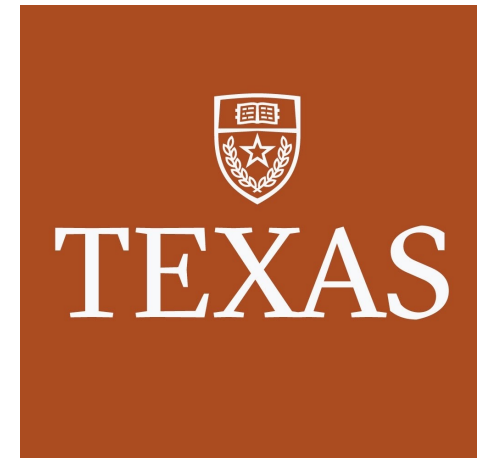
Course Logistics

Teaching Staff

- Instructor: Joyce Ho
 - Email: joyce.c.ho@emory.edu
 - Office Hours: Tues/Thurs 1-4 pm @ MSC W414
- TA: Camilo Valderrama
 - Email: cevacu@gmail.com
 - Office Hours: Mon 2-4 pm @MSC N410

About Me

- Undergraduate / MEng from MIT
- PhD from University of Texas at Austin
- Research interests:
 - Data Mining / Machine Learning
 - Healthcare Informatics
- More information: <http://joyceho.github.io>

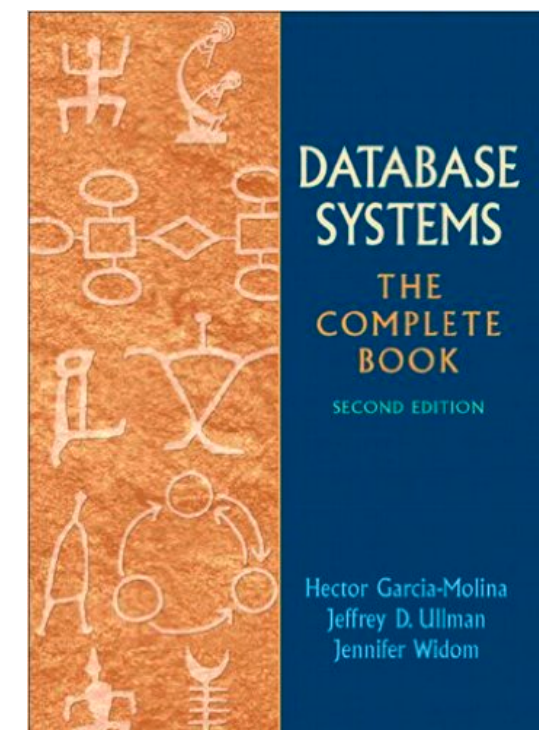
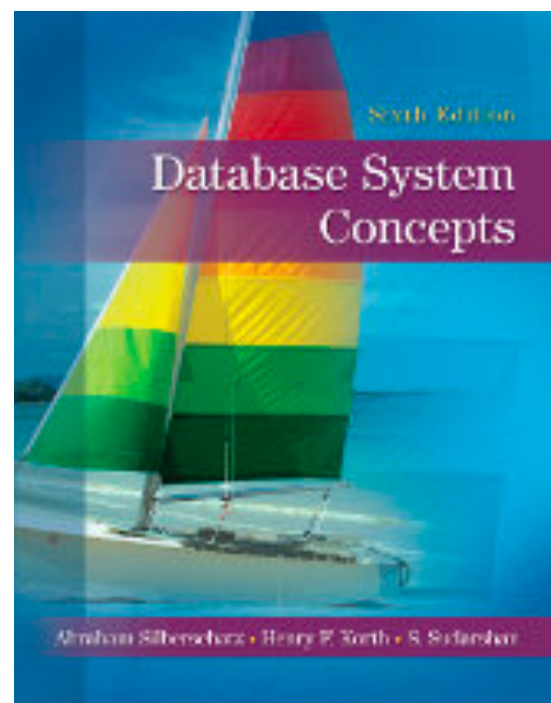
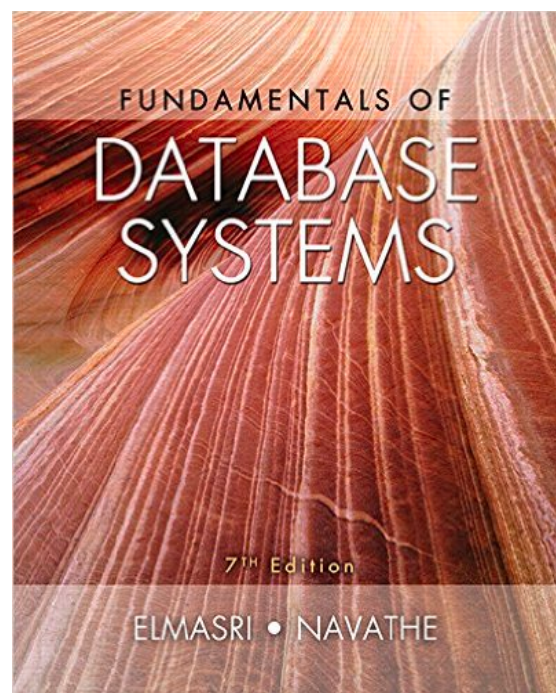


Course Information

- Webpage: <http://joyceho.github.io/cs377-s16/index.html>
 - Lectures
 - Assignments
- Piazza: <http://piazza.com/emory/spring2016/cs377>
 - Announcements
 - Questions + Discussions
 - Note: Please use an emory address or your OPUS name

Course Textbook

- Fundamentals of Database System (Recommended)
Ramez Elmasri and Shamkant Navathe
- Database System Concepts (Supplemental)
Abraham Silberschatz, Henry Korth, and S. Sudarshan
- Database Systems: The Complete Book (Supplemental)
Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom



Grading

Assignments / Projects	50%
Midterm	20%
Final	25%
Participation	5%

Grades may be curved (up) so the class mean falls at least in a B range.

Participation Details

- Commitment to help you learn
- Bi-Weekly “quizzes” to assess class comprehension
 - Format will be 1 question, 10 minutes
 - Not meant to require much (if any) studying
- Bi-Weekly “polls” to tailor class for the students

In theory, if you attend class regularly and pay attention this should be automatic

Policies

- Exams
 - Must be taken at the required time
 - Midterm rescheduling can be possible if request is made at least a week prior to the date
- Late Assignment
 - Will be accepted within 3 days of due date with 10% penalty per day
 - 2 late date assignments per student

Honor Code

- College Honor Code and Departmental Policy
- Acceptable and encouraged to discuss assignments with other students but **ANY WRITEUP AND CODE MUST BE YOUR OWN**
- All program assignments must include the following comment at the top of the file:

```
/*
```

```
THIS CODE IS MY OWN WORK, IT WAS WRITTEN WITHOUT  
CONSULTING CODE WRITTEN BY OTHER STUDENTS.
```

```
_Your_Name_Here_
```

```
*/
```

Databases: High-Level Introduction

What does it mean?

- **Data:** A set of known facts that can be recorded and have an implicit meaning
- **Database:** A collection (files) of related data
- **Database Management System (DBMS):** A software package or system that facilitates the creation and maintenance of a computerized database, allowing data to persist over long periods of time

Examples of DBMS

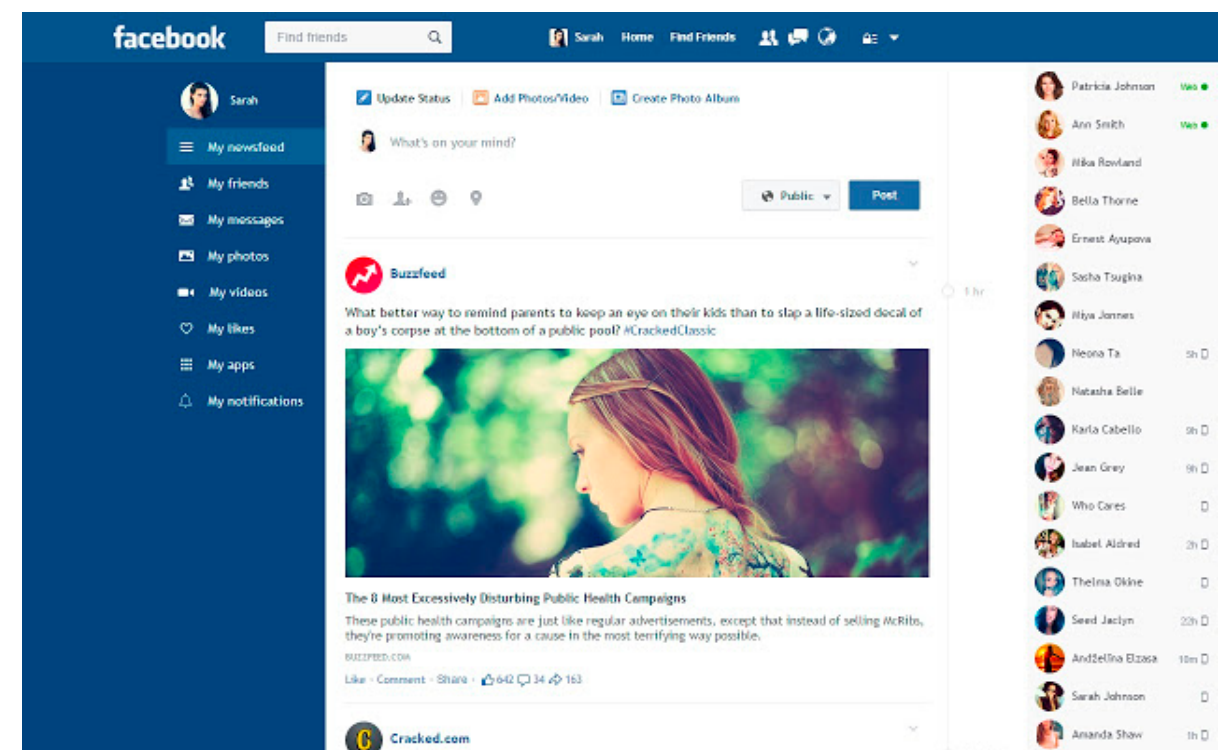
- SQL Server, Microsoft Access (Microsoft)
- DB2 (IBM)
- Oracle
- MySQL, PostgreSQL, SQLite (Open Source)



All are "relational" (or "object-relational") database systems

Example: Facebook

- What data needs to be stored?
 - Status Updates
 - Requests
 - Alerts
 - ...

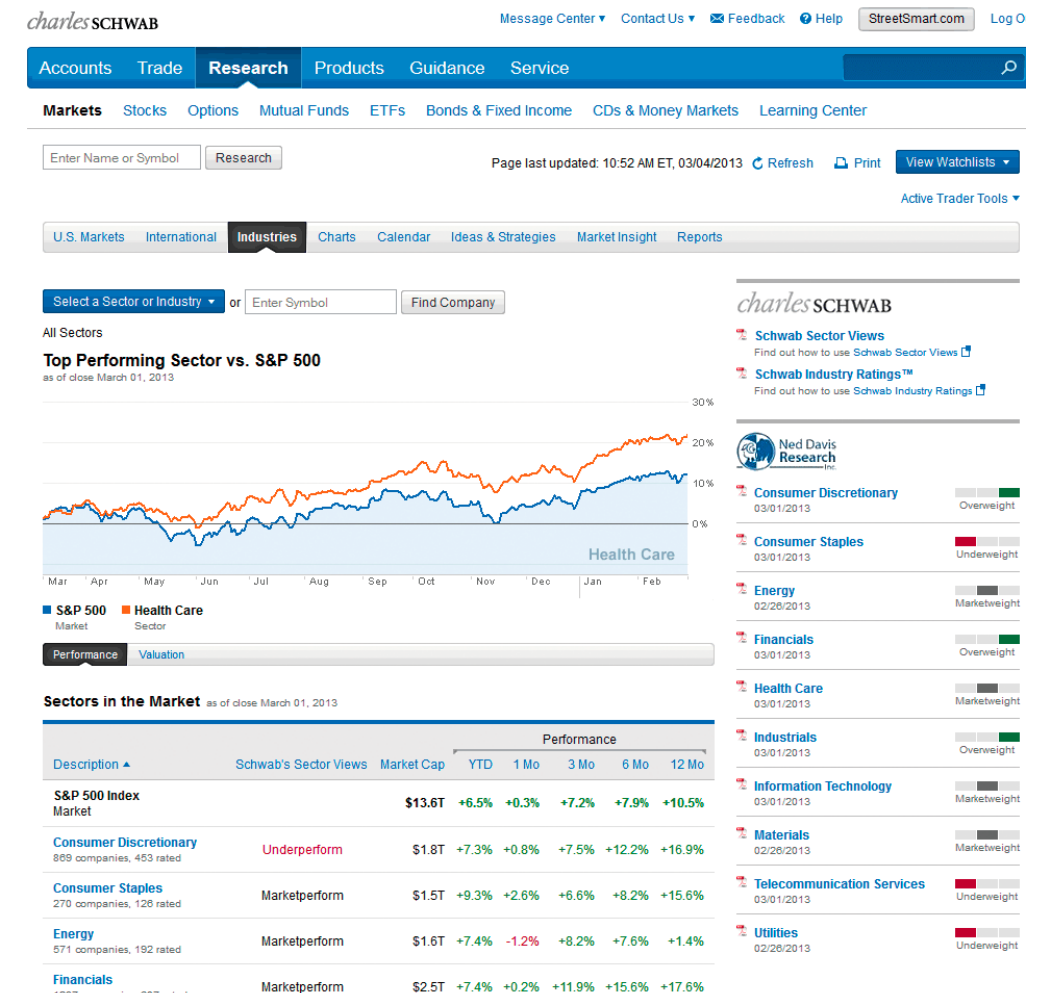


Facebook uses MySQL

- How is data stored used?

Example: Charles Schwab

- What data needs to be stored?
 - Customers
 - Transaction histories
 - Balances
 - ...
- How is data stored used?

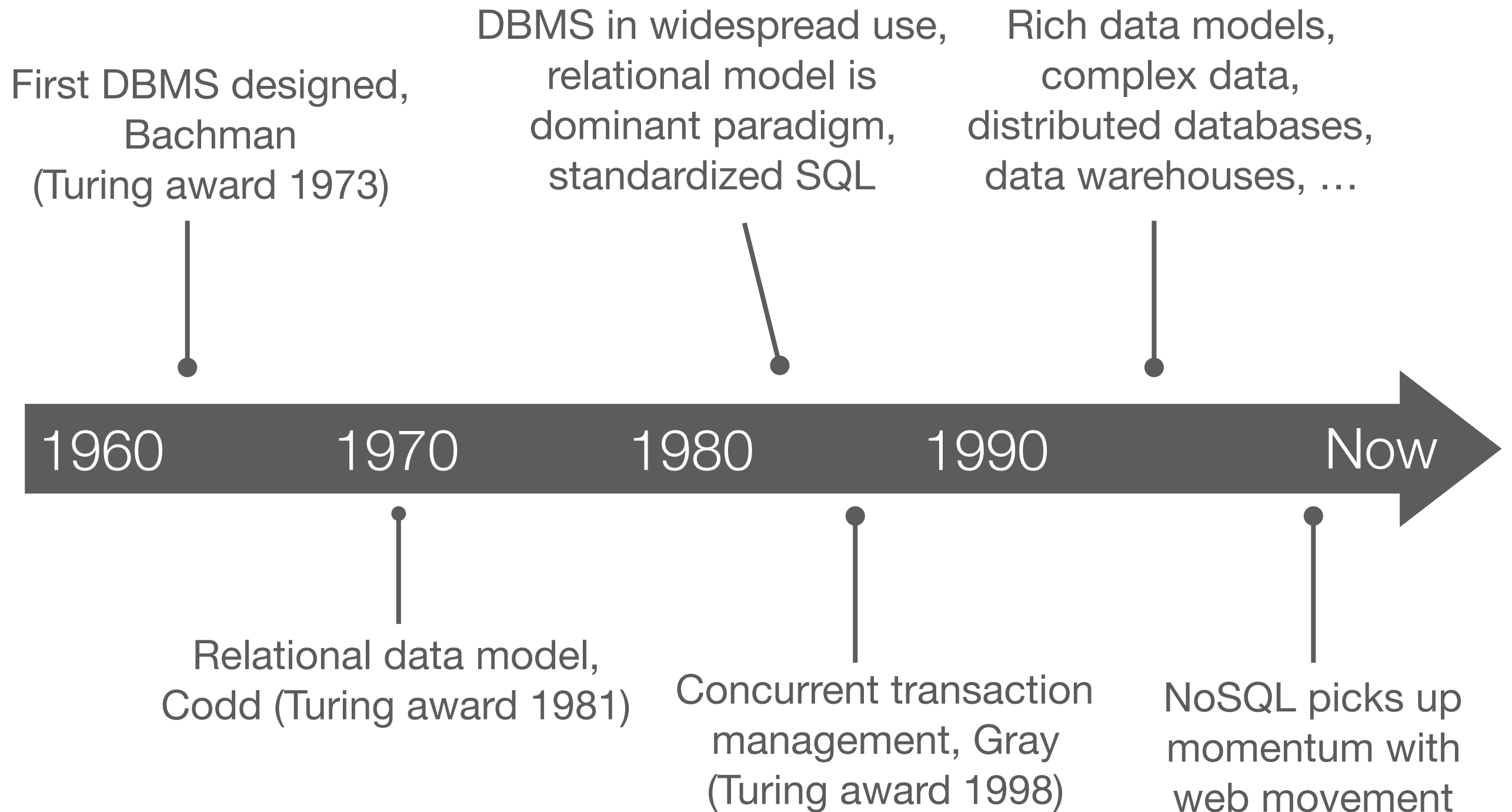


Charles Schwab uses multiple DBMS including Oracle and SQL Server

Standard DBMS Functionality

- Database manipulations: insertions, deletions, and modifications
- Efficient querying
- Concurrent processing and sharing by multiple users
- Consistent and valid data
- Recovery after crashes
- Security and user authorization

Evolution of DBMS



People

- **Database administrator:** sets up software/hardware, authorizes access, monitors its use, and tunes system
- **Database designer:** define and set up schema, sometimes loads the data
- **Database user:** modifies and queries the data
- **Database application developer:** builds applications to query/modify data
- **DBMS implementor:** builds the DBMS system



Course Content

- Basic concepts
 - Data independence, database system components, 3 level database architecture
- Data modeling and design
 - Entity-Relationship Model
 - Relational Data Model
 - Relational Database design theory - normal functions, functional dependencies
- Database programming
 - Relational algebra
 - SQL (Structured Query Language)
 - Oracle JDBC and PHP
- Selected Topics
 - Database internals (indexes, query optimization, transactions)
 - NoSQL

