

# فصل 1: مقدمه

قمرناز تدین

- ▶ Database system concepts, A. Silberschats, H.F. Korth, S. Sudarshan, 6th Edition, McGraw Hill, 2011.
- ▶ An introduction to Database systems, C. J. Date, 8nth Edition, Addison Wesley, 2004

## Database Management System (DBMS)

► DBMS شامل اطلاعاتی درباره یک سازمان خاص است

- مجموعه ای از داده هایی که با هم مرتبط هستند
- مجموعه ای از برنامه ها برای دسترسی به داده ها
- محیطی که استفاده از آن کارآمد و آسان باشد.

► کاربردهای پایگاه داده:

- بانکداری: تراکنشها
  - خطوط هوایی: رزرواسیون، زمان بندی
  - دانشگاهها: ثبت نام، کلاسها
  - فروش: مشتریها، محصولات، خرید
  - فروش آنلاین: پیگیری سفارش، پیشنهادهای خاص
  - تولید: تولید، بازرگانی، سفارش، زنجیره تدارکات
  - منابع انسانی: رکوردهای کارمندان، حقوق، مالیات
- پایگاههای داده ممکن است بسیار بزرگ باشند.

► پایگاههای داده در همه جنبه های زندگی ما وجود دارند

# مثال: پایگاه داده دانشگاه

▶ مثالهایی از برنامه های کاربردی:

- اضافه کردن دانشجو، استاد و دروس جدید
- ثبت نام دانشجویان در یک درس، ایجاد لیست کلاس
- رتبه بندی دانشجویان، تعیین میانگین نمرات، چاپ کارنامه

▶ در ابتدا، کاربردهای پایگاه داده مستقیماً بر روی سیستمهای فایل نوشته می شدند.

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

<i>dept_name</i>	<i>building</i>	<i>budget</i>
Comp. Sci.	Taylor	100000
Biology	Watson	90000
Elec. Eng.	Taylor	85000
Music	Packard	80000
Finance	Painter	120000
History	Painter	50000
Physics	Watson	70000

(b) The *department* table

## نقاط ضعف سیستمهای فایل برای ذخیره اطلاعات

- ▶ افزونگی و ناسازگاری داده ها
  - فرمتهای متفاوت داده ها، تکرار اطلاعات در فایلهاى مختلف
- ▶ مشکل در دستیابی به داده ها
  - لازم است برنامه جدیدی برای انجام هر وظیفه جدیدی نوشته شود.
- ▶ جداسازی داده ها- فایل ها و فرمت های متفاوت
- ▶ مشکلات جامعیت
  - محدودیت های جامعیت (مثلا این که  $\text{account balance} > 0$  به جای این که بتوان به صورت صریح تعریف کرد، باید در برنامه کد شود.
  - افزودن محدودیت های جدید یا تغییر محدودیتهای موجود مشکل است.

## نقاط ضعف سیستمهای فایل برای ذخیره اطلاعات (ادامه)

### ▶ اتمیسیته بروزرسانی ها

- خرابی ها ممکن است باعث شوند که پایگاه داده در وضعیت ناسازگار قرار بگیرد و فقط بخشی از بروزرسانیها انجام شود.
- مثال: انتقال پول از یک حساب به حساب دیگر باید یا به طور کامل انجام شود یا اصلا انجام نشود.

### ▶ دستیابی موازی توسط کاربران مختلف

- دستیابی موازی برای افزایش کارایی ضروری است.
- دستیابی های موازی کنترل نشده ممکن است باعث عدم سازگاری شود.
- مثال: دو نفر ممکن است بالانس یک حساب را مشاهده کنند (مثلا 100) و مقدار آن را با کسر پول بروزرسانی کنند (مثلا هریک 50)

### ▶ مشکلات امنیت

- کنترل این که دستیابی کاربر فقط به بخشی از داده ها میسر شود مشکل است.

**سیستمهای پایگاه داده راه حل هایی برای همه مشکلات فوق پیشنهاد میدهند.**

## سطوح مجردسازی

▶ **سطح فیزیکی:** توصیف میکند که چگونه یک رکورد (مثلا مشتری) ذخیره میشود.

▶ **سطح منطقی:** داده ذخیره شده در پایگاه داده و روابط بین داده ها را توصیف میکند.

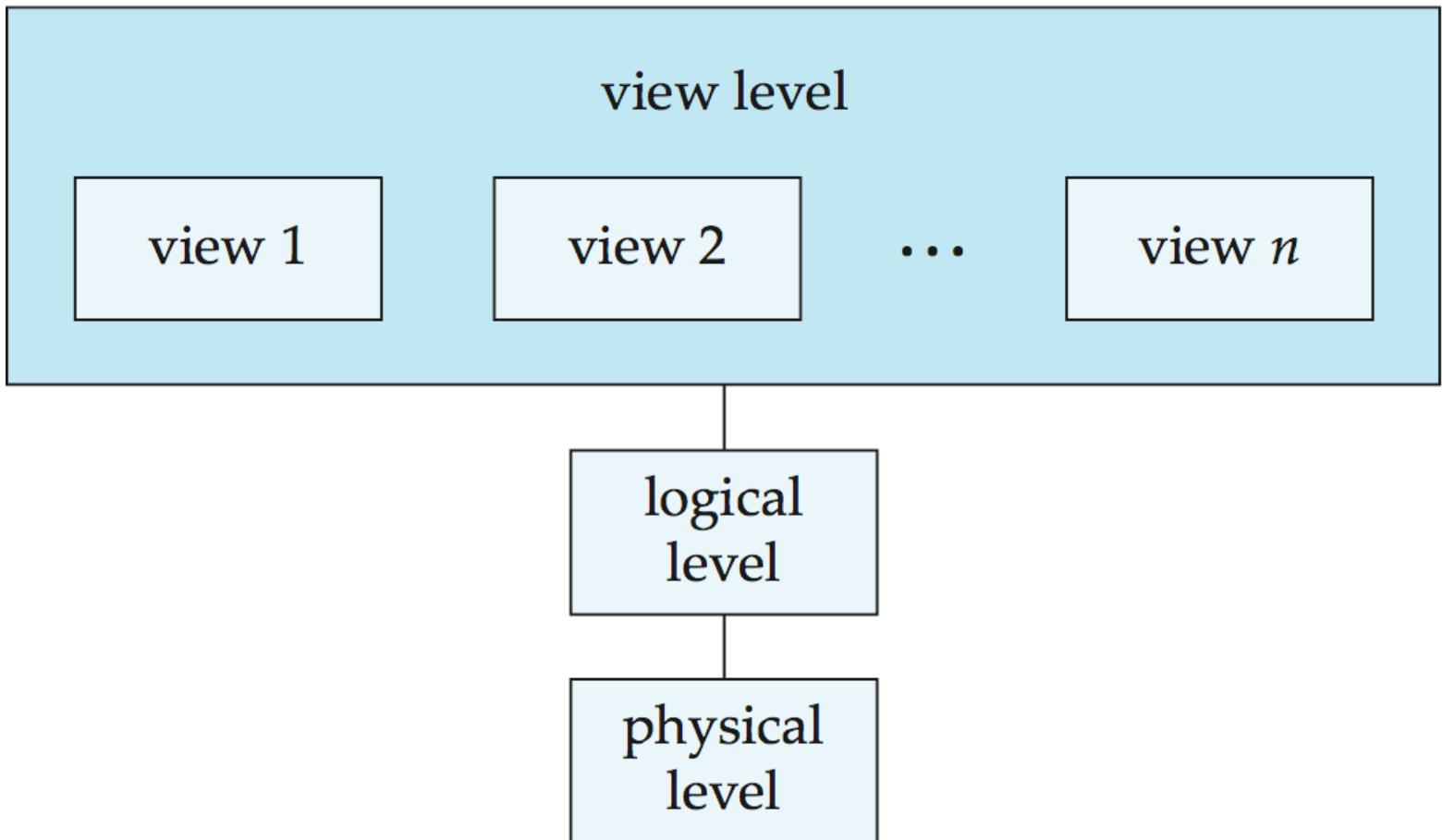
```
type instructor = record
    ID : string;
    name : string;
    dept_name : string;
    salary : integer;
end;
```

▶ **سطح دیدگاه:** برنامه های کاربردی جزئیات انواع داده را مخفی میکنند. دیدگاهها از نظر امنیت نیز میتوانند اطلاعات را مخفی کنند. (مثلا حقوق کارمندان)



# View of Data

معماری سیستم پایگاه داده



# نمونه ها و شما Instances and Schema

▶ مشابه انواع داده و متغیرها در زبانهای برنامه سازی:

▶ **شما** - ساختار منطقی پایگاه داده

- مثال: پایگاه داده ای شامل اطلاعاتی درباره مجموعه مشتریان و حسابها و رابطه بین آنها
- مشابه اطلاعات مربوط به نوع متغیرهای یک برنامه
- **شمای فیزیکی**: طراحی پایگاه داده در سطح فیزیکی
- **شمای منطقی**: طراحی پایگاه داده در سطح منطقی

▶ **نمونه** - محتویات واقعی پایگاه داده در زمان معین

- مشابه مقدار متغیر

▶ **استقلال داده فیزیکی** - قابلیت تغییر شمای فیزیکی بدون تغییر در شمای منطقی

- کاربردها وابسته به شمای منطقی هستند.
- در کل، رابطهای بین سطوح و اجزاء مختلف باید به خوبی تعریف شوند تا تغییرات یک بخش تأثیر قابل توجهی در سایر بخش ها نداشته باشد.

# مدلهای داده

► مجموعه ابزارهایی برای توصیف

- داده

- روابط داده ها

- معناهای داده ها

- محدودیت های داده ها

► مدل رابطه ای

► مدل داده موجودیت - رابطه (معمولا برای طراحی پایگاه داده

استفاده می شود)

► مدلهای داده شیء گرا

► مدلهای داده نیمه ساخت یافته (XML)

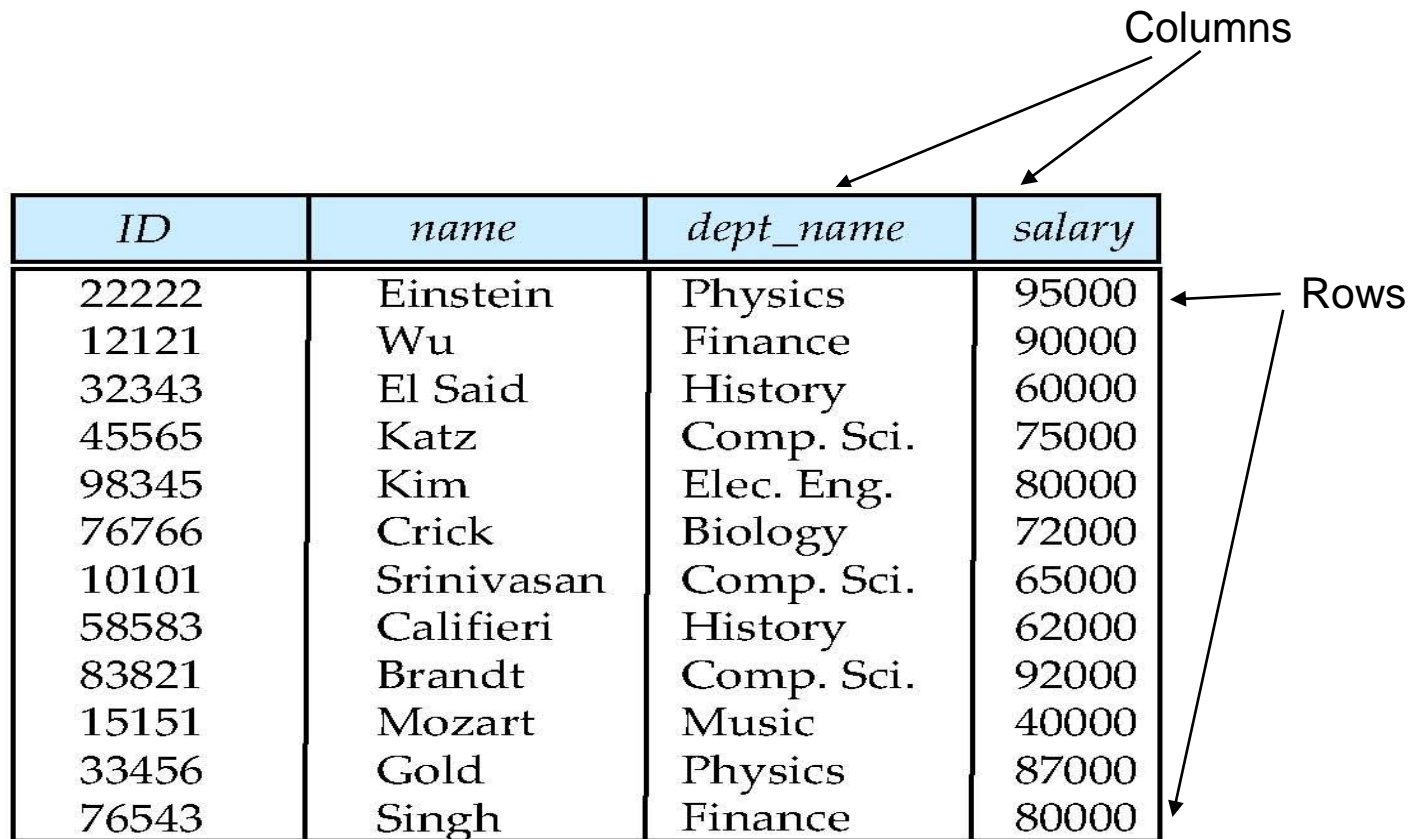
► سایر مدلهای قدیمی

- مدلهای شبکه

- مدلهای سلسله مراتبی

# Relational Model

# مدل رابطه ای



The diagram shows a table with four columns and eleven rows. Two arrows labeled 'Columns' point to the top of the first and second columns. Two arrows labeled 'Rows' point to the first and last rows of the table.

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

# A Sample Relational Database

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

<i>dept_name</i>	<i>building</i>	<i>budget</i>
Comp. Sci.	Taylor	100000
Biology	Watson	90000
Elec. Eng.	Taylor	85000
Music	Packard	80000
Finance	Painter	120000
History	Painter	50000
Physics	Watson	70000

(b) The *department* table

## زبان تغییر داده ها

# Data Manipulation Language (DML)

- ▶ زبانی برای دسترسی و تغییر داده هایی که به صورت مدل داده مناسبی سازماندهی شده است.
  - DML به عنوان زبان پرس و جو نیز شناخته می شود.
- ▶ دو دسته از زبانها
  - **روالی (procedural)** - کاربر تعیین میکند که چه داده ای لازم است و چگونه این داده ها را میتوان به دست آورد.
  - **توصیفی (غیر روالی) - Declarative (nonprocedural)** کاربر مشخص می کند که کدام داده لازم است بدون اینکه مشخص کند چگونه این داده ها به دست می آیند.
- ▶ متداول ترین زبان پرس و جو SQL است.

# Data Definition Language (DDL)

► روش تعریف شمای پایگاه داده ها

Example: **create table** *instructor* (  
                  *ID*                  **char**(5),  
                  *name*              **varchar**(20),  
                  *dept\_name* **varchar**(20),  
                  *salary*          **numeric**(8,2))

► کامپایلر DDL مجموعه ای از الگوهای جدول را تولید میکند که در یک فرهنگ داده (*data dictionary*) ذخیره میشود.

► فرهنگ داده شامل فراداده (metadata) است. (به معنی داده درباره داده)

◦ شمای پایگاه داده

◦ محدودیت جامعیت

• کلید اصلی (Primary key ID) که به طور منحصر به فرد هر استاد را تعیین میکند.)

• جامعیت ارجاع (Referential integrity) (محدودیت ارجاع در SQL)

• مقدار *dept\_name* در هر رکورد استاد باید حتما در رابطه بخش وجود داشته باشد.

◦ مجوزدهی Authorization

# SQL

► **SQL**: متداول ترین زبان غیر روالی است.

◦ مثال: یافتن نام استادی با شماره 22222

```
select name  
from instructor  
where instructor.ID = '22222'
```

◦ مثال: یافتن شماره و ساختمان مربوط به اساتید گروه فیزیک

```
select instructor.ID, department.building  
from instructor, department  
where instructor.dept_name =  
department.dept_name and  
department.dept_name = 'Physics'
```

► برنامه های کاربردی با یکی از روشهای زیر به پایگاه داده دسترسی پیدا میکنند.

◦ توسعه های در زبانها که در آنها SQL اضافه شده است.

◦ رابط های برنامه های کاربردی که امکان ارسال پرس و جو به پایگاه داده را فراهم میکنند. (ODBC/JDBC)



# طراحی پایگاه داده Database Design

فرایند طراحی ساختار عمومی پایگاه داده:

► طراحی منطقی- تصمیم درباره شمای پایگاه داده. در طراحی پایگاه داده لازم است که یک مجموعه مناسبی از شمای رابطه ای پیدا شود

◦ تصمیم های تجاری و کاری: چه خصیصه هایی باید در پایگاه داده ذخیره شوند؟

◦ تصمیم های علمی و کامپیوتری: چه شمای رابطه ای داریم و چگونه باید خصیصه ها در شمای مختلف رابطه ای توزیع شوند؟

► طراحی فیزیکی: تصمیمگیری درباره نمایش فیزیکی پایگاه داده ها

# Database Design?

► آیا طرح زیر مشکلی دارد؟

<i>ID</i>	<i>name</i>	<i>salary</i>	<i>dept_name</i>	<i>building</i>	<i>budget</i>
22222	Einstein	95000	Physics	Watson	70000
12121	Wu	90000	Finance	Painter	120000
32343	El Said	60000	History	Painter	50000
45565	Katz	75000	Comp. Sci.	Taylor	100000
98345	Kim	80000	Elec. Eng.	Taylor	85000
76766	Crick	72000	Biology	Watson	90000
10101	Srinivasan	65000	Comp. Sci.	Taylor	100000
58583	Califieri	62000	History	Painter	50000
83821	Brandt	92000	Comp. Sci	Taylor	100000
15151	Mozart	40000	Music	Packard	80000
33456	Gold	87000	Physics	Watson	70000
76543	Singh	80000	Finance	Painter	120000

# رهیافتهای طراحی

## ► تئوری نرمالسازی

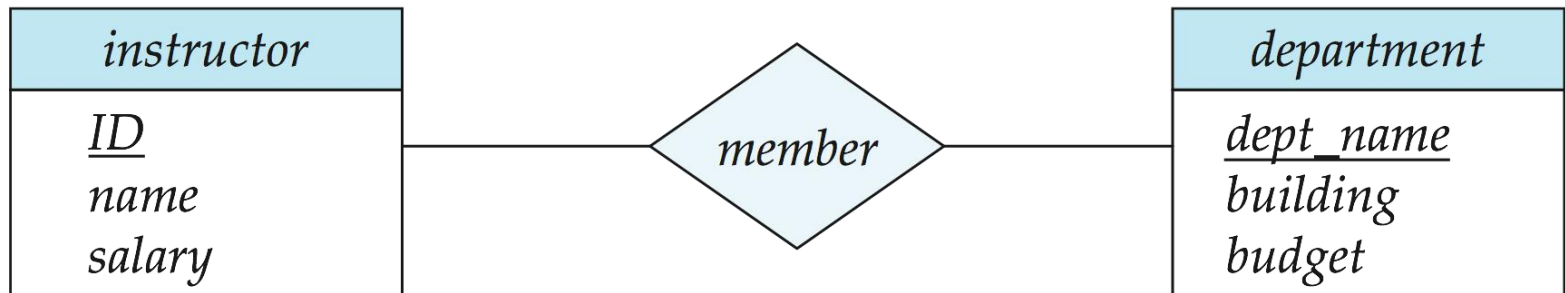
- تعیین نکات نادرست در طراحی و بررسی آنها

## ► مدل موجودیت - رابطه

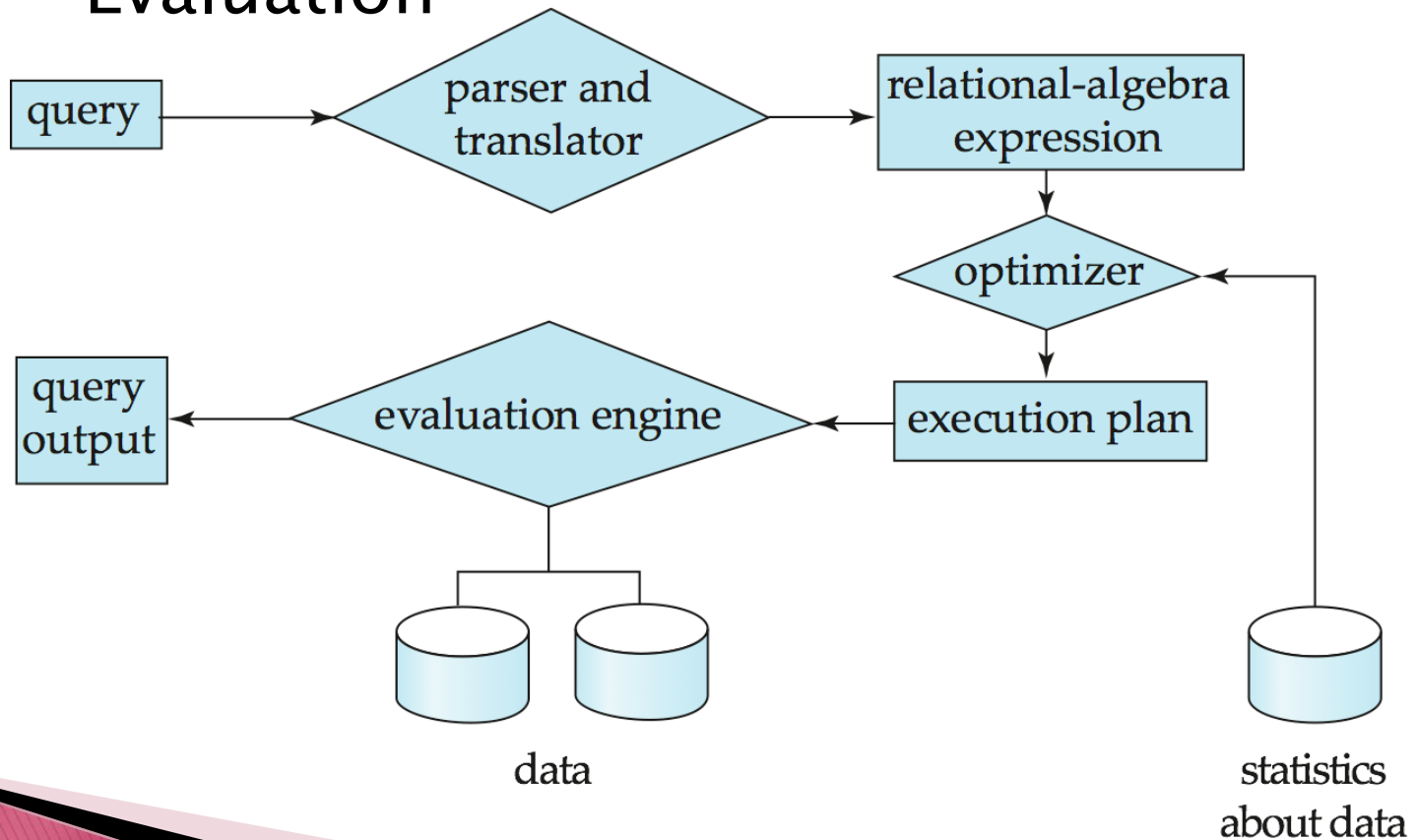
- سازمان به صورت مجموعه ای از موجودیتها و رابطه ها مدل می شود.
- موجودیت: یک شیء در سازمان که از سایر اشیاء قابل تمایز است.
- به صورت مجموعه ای از ویژگیها تعریف می شود.
- رابطه: ارتباط بین موجودیتهای مختلف
- به صورت نمودار موجودیت - رابطه نشان داده می شود.

مدل موجودیت - رابطه

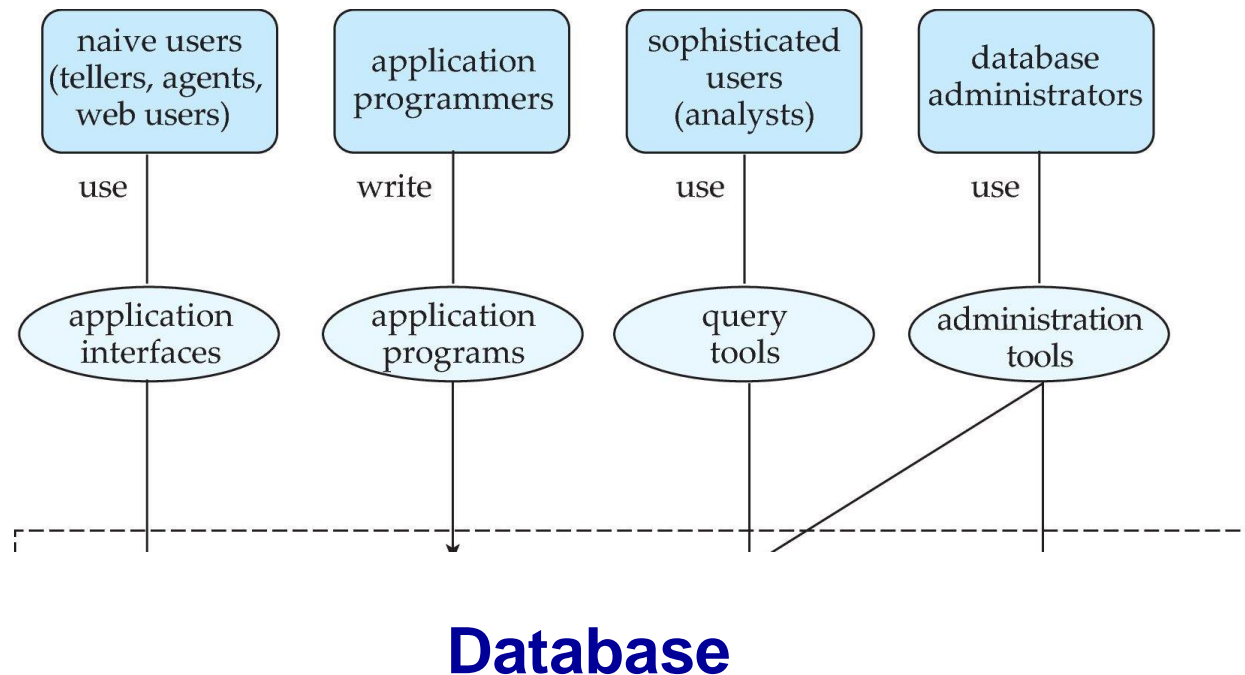
# The Entity-Relationship Model



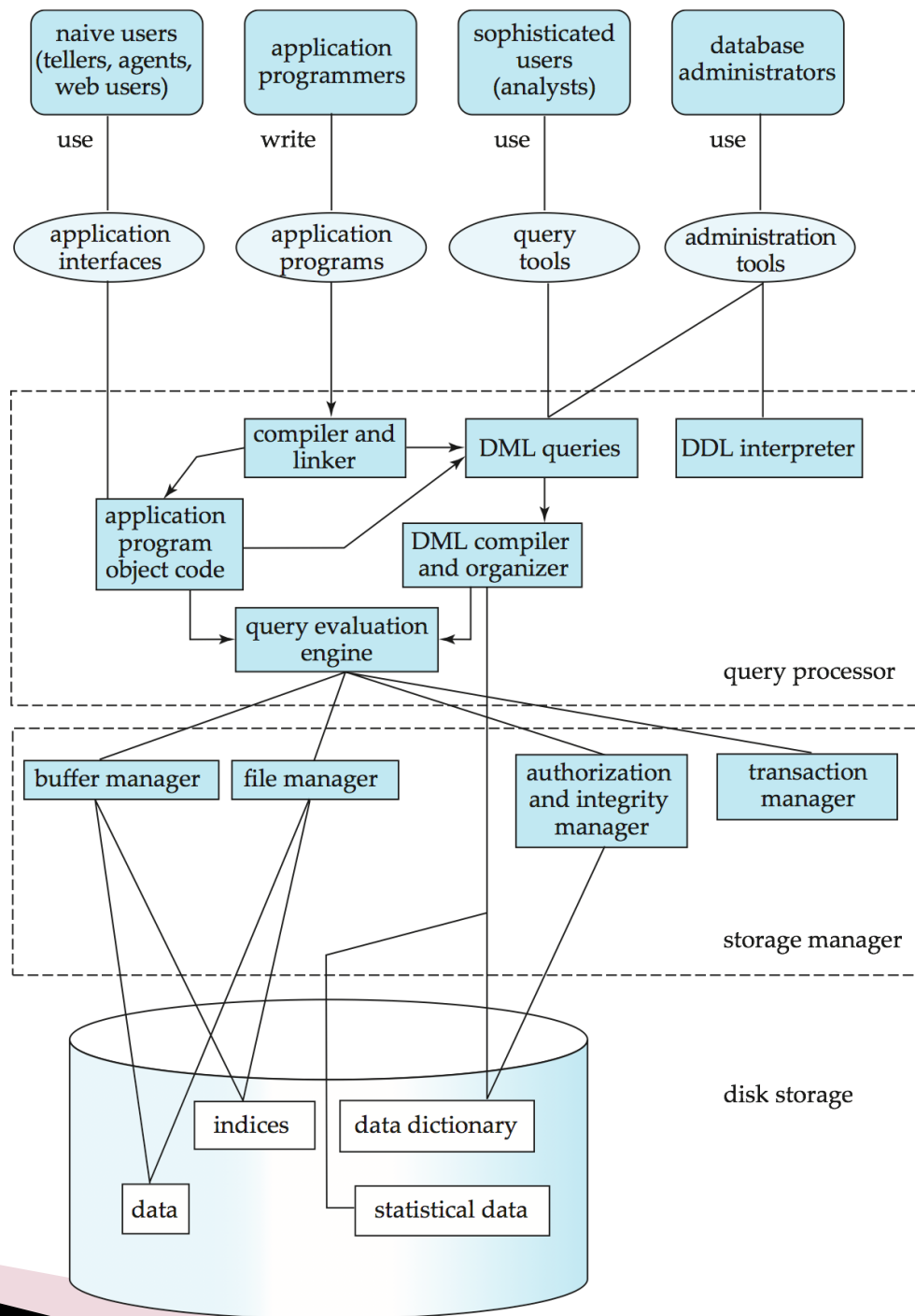
1. Parsing and translation
2. Optimization
3. Evaluation



# Database Users and Administrators



# Database System Internals



# تاریخچه سیستمهای پایگاه داده

## ► دهه 50 و اواخر دهه 60:

- پردازش داده با استفاده از نوار مغناطیسی برای ذخیره سازی اطلاعات
  - در نوار فقط دستیابی ترتیبی به اطلاعات امکان پذیر است.
- کارت پانچ برای ورود اطلاعات.

## ► اواخر دهه 60 و دهه 70:

- استفاده از دیسک سخت امکان دستیابی مستقیم به داده ها را فراهم کرد.
- مدل‌های داده شبکه ای و سلسله مراتبی استفاده شدند.
- مدل داده رابطه ای توسط Ted Codd معرفی شد.
  - جایزه ACM Turing را برای این کار برنده شد.
  - تحقیقات IBM برای System R آغاز شد.
  - UC Berkeley تحقیقات Ingres را آغاز کرد.
- پردازش تراکنش با کارایی بالا



# تاریخچه (ادامه)

## ► دهه 80:

- تحقیقات در مدل رابطه ای منجر به ایجاد سیستمهای تجاری شد.
- SQL به صورت یک استاندارد صنعتی درآمد.
- سیستمهای پایگاه داده موازی و توزیع شده.
- سیستمهای پایگاه داده شیء گرا.

## ► دهه 90:

- کاربردهای داده کاوی و تصمیم یار بزرگ
- انباره های داده بزرگ چند ترابایتی
- تجارت وب

## ► اوایل 2000:

- XML and XQuery standards
- تصدی پایگاه داده ها به صورت خودکار

## ► اواخر 2000:

- سیستمهای ذخیره سازی عظیم
- Google BigTable, Yahoo PNuts, Amazon, ..