INF1001: Introduction to Computing

L2a: Database – Entity Relationship (ER)

Aug 2023

Topic outline: Database-ER

Database fundamentals

Definition

DBMS

Relational Model

Terminology

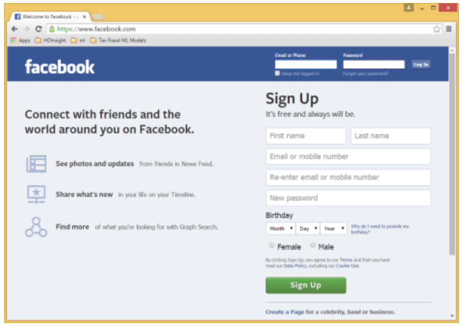
Entity Relationship (ER) Model

Terminology

Notation

Converting ER Model to Relational Model

Data, Data, Data - Everywhere


Welcome to Facebook - L

https://www.facebook.com

Apps | HDInsight | ml | Tax Fraud ML Models

facebook

Email or Phone | Password | Log In

☐ Keep me logged in | Forgot your password?

Connect with friends and the world around you on Facebook.

See photos and updates from friends in News Feed.

Share what's new in your life on your Timeline.

Find more of what you're looking for with Graph Search.

Sign Up

It's free and always will be.

First name | Last name

Email or mobile number

Re-enter email or mobile number

New password

Birthday: Month ▾ Day ▾ Year ▾ Why do I need to provide my birthday?

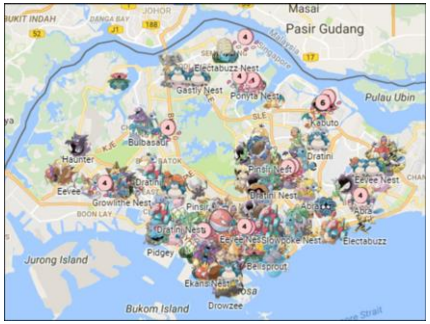
○ Female ○ Male

By clicking Sign Up, you agree to our Terms and that you have read our Data Policy, including our Cookie Use.

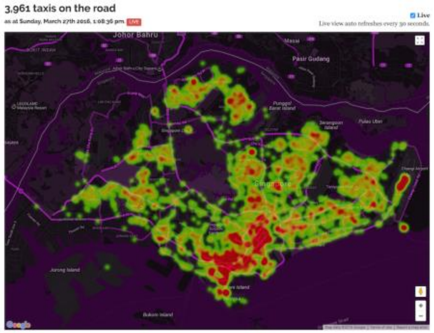
Sign Up

Create a Page for a celebrity, band or business.

Social Media



Games

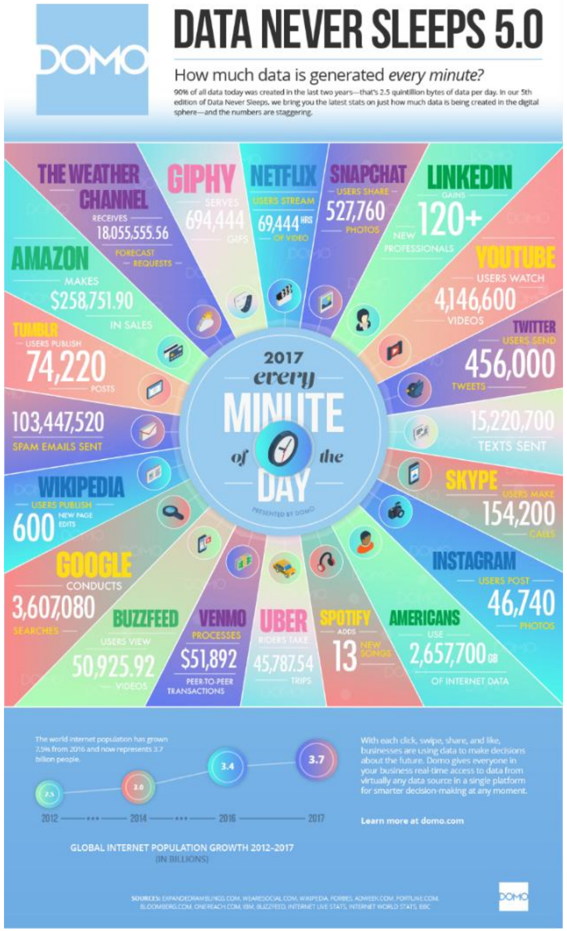

3,961 taxis on the road

as at Sunday, March 27th 2016, 1:08:36 pm LIVE

☑ Live

Live view auto refreshes every 30 seconds.

Transportation


DATA NEVER SLEEPS 5.0

How much data is generated *every minute?*

90% of all data today was created in the last two years—that's 2.5 quintillion bytes of data per day. In our 5th edition of Data Never Sleeps, we bring you the latest stats on just how much data is being created in the digital sphere—and the numbers are staggering.

The world internet population has grown 7.5% from 2016 and now represents 3.7 billion people.

GLOBAL INTERNET POPULATION GROWTH 2012-2017 (IN BILLIONS)

With each click, swipe, share, and like, businesses are using data to make decisions about the future. Domo gives everyone in your business real-time access to data from virtually any data source in a single platform for smarter decision-making at any moment.

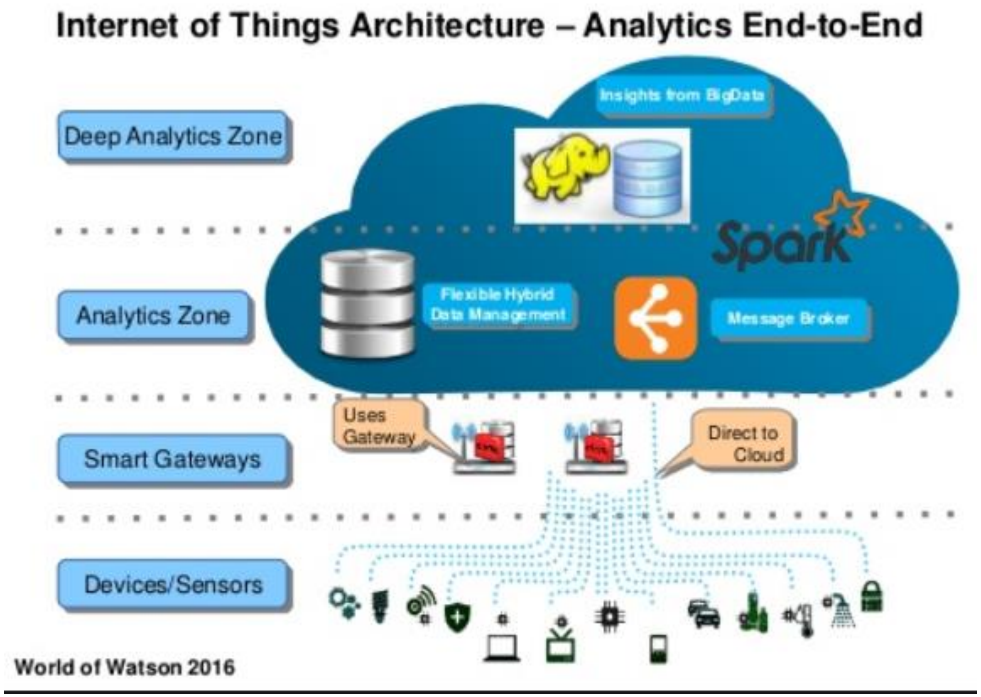
Learn more at domo.com

SOURCES: EXPANDEDRAMBLINGS.COM, WEARESOCIAL.COM, WIKIPEDIA, FORBES, ADWEEK.COM, FORTUNE.COM, BLOOMBERG.COM, ONEREACH.COM, IBM, BUZZFEED, INTERNET LIVE STATS, INTERNET WORLD STATS, BBC

* Data, data everywhere - http://www.economist.com/node/15557443

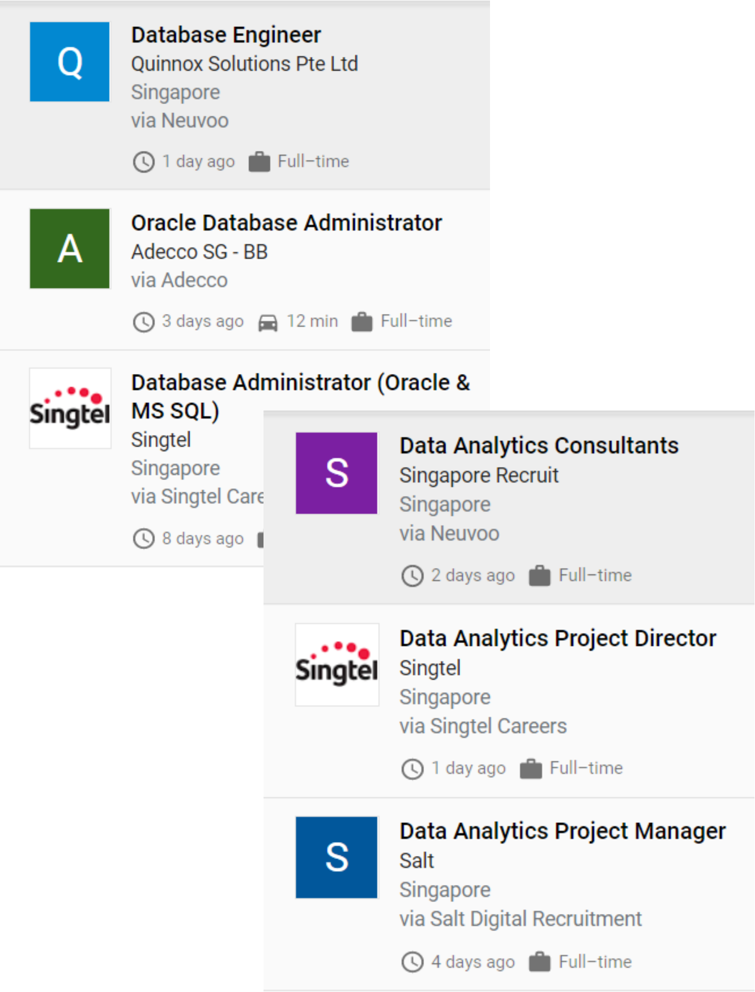
- Scaling the Facebook Data Warehouse - https://code.facebook.com/posts/229861827208629/scaling-the-facebook-data-warehouse-to-300-pb/
- 2.5 EB (Exa bytes or 2^{60}) of data produced every day
- 90% of the data in the world generated over the last 2 years.
- The explosion of data give rise to areas in data science, machine learning and artificial intelligence (AI)

Why learn about database?



World of Watson 2016

Applicable to many domains!



Database Engineer
Quinnox Solutions Pte Ltd
Singapore
via Neuvoo
1 day ago · Full-time

Oracle Database Administrator
Adecco SG - BB
via Adecco
3 days ago · 12 min · Full-time

Database Administrator (Oracle & MS SQL)
Singtel
Singapore
via Singtel Care
8 days ago

Data Analytics Consultants
Singapore Recruit
Singapore
via Neuvoo
2 days ago · Full-time

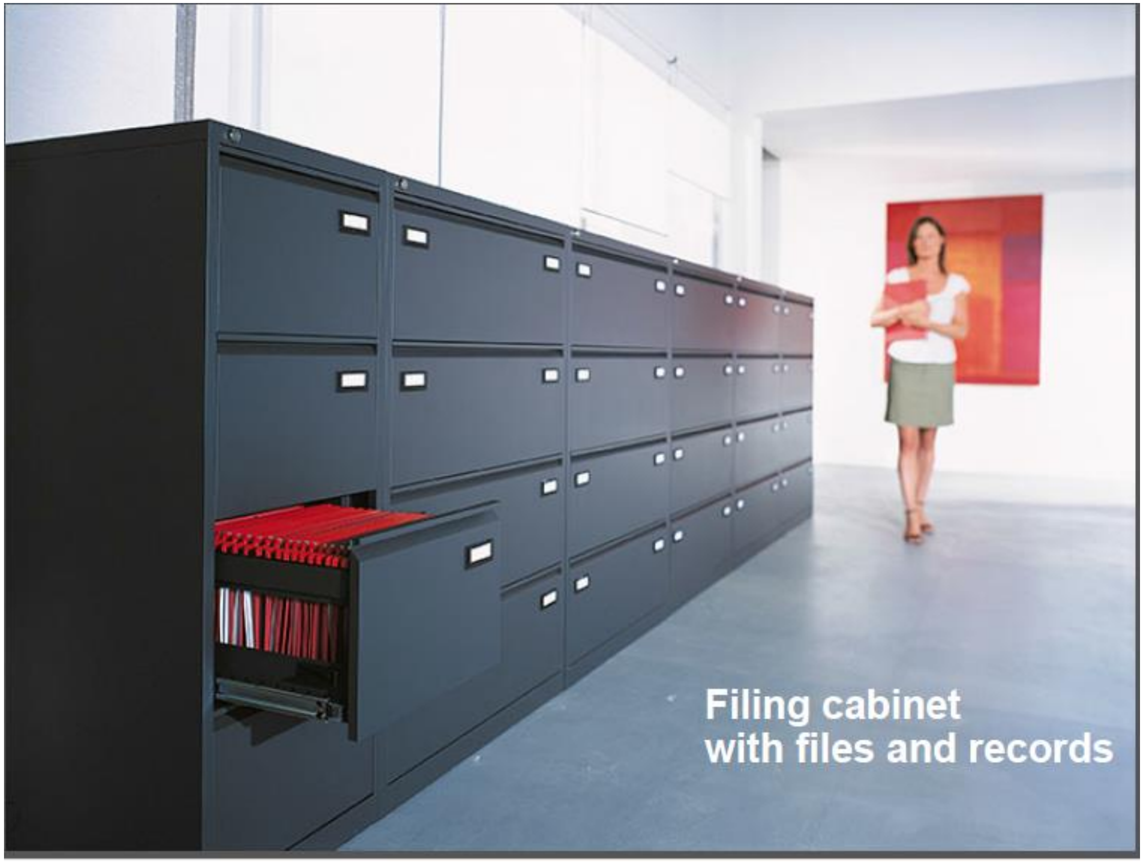
Data Analytics Project Director
Singtel
Singapore
via Singtel Careers
1 day ago · Full-time

Data Analytics Project Manager
Salt
Singapore
via Salt Digital Recruitment
4 days ago · Full-time

What is Database?

- **Organized** collection of **data**
  - Structured in multiple formats
  - Stored in some form of storage medium (which can include hard-disk drives, flash memory, solid-state drives and even DRAM)
  - They could be stored anywhere in the world with cloud technology

How to store? In the past…



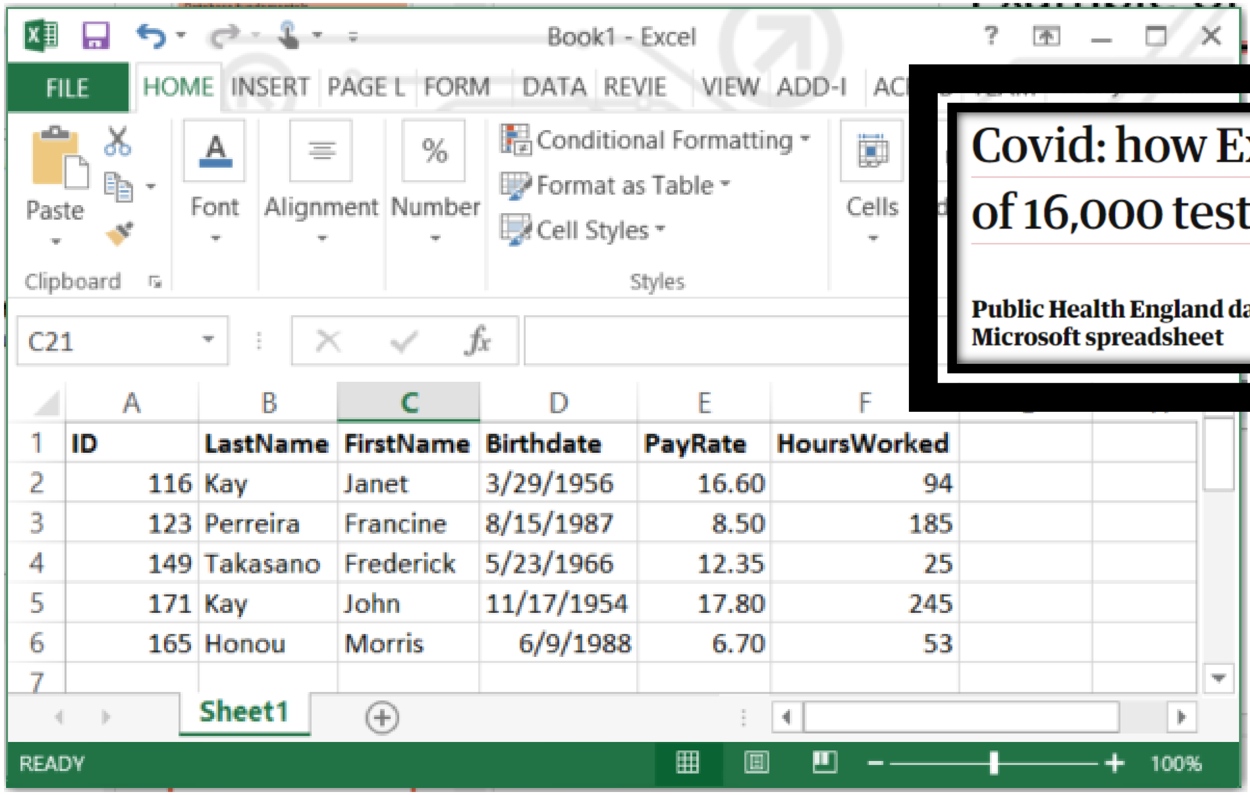
Filing cabinet
with files and records

How to store? In flat text file?

- Example: employee.txt (1 record per line)

- Disadvantage:
  - File corruption risk if computer fails while updating file
  - Custom scripts necessary to manage numerous files
  - Cannot update data **at the same time (concurrency)**
  - No security features
  - Difficulty to search or group/categorise data
  - Hard to track access (data logging of who accessed what data)

How to store? In spreadsheet?



	A	B	C	D	E	F
1	ID	LastName	FirstName	Birthdate	PayRate	HoursWorked
2	116	Kay	Janet	3/29/1956	16.60	94
3	123	Perreira	Francine	8/15/1987	8.50	185
4	149	Takasano	Frederick	5/23/1966	12.35	25
5	171	Kay	John	11/17/1954	17.80	245
6	165	Honou	Morris	6/9/1988	6.70	53
7						

Covid: how Excel may have caused loss of 16,000 test results in England

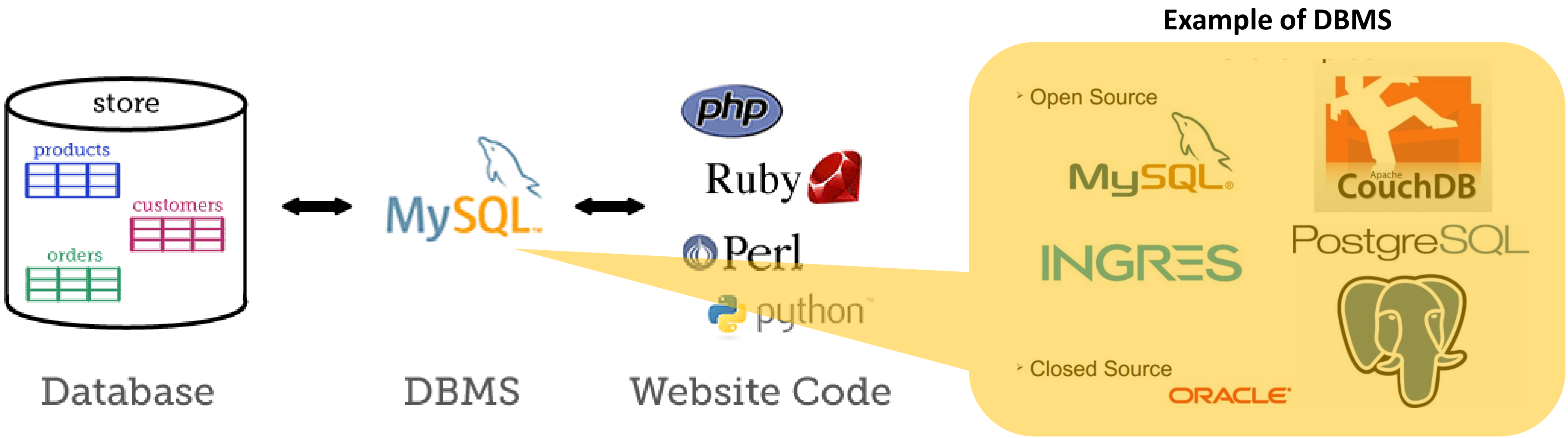
Public Health England data error blamed on limitations of Microsoft spreadsheet

https://www.theguardian.com/politics/2020/oct/05/how-excel-may-have-caused-loss-of-16000-covid-tests-in-england

How to store? DBMS!

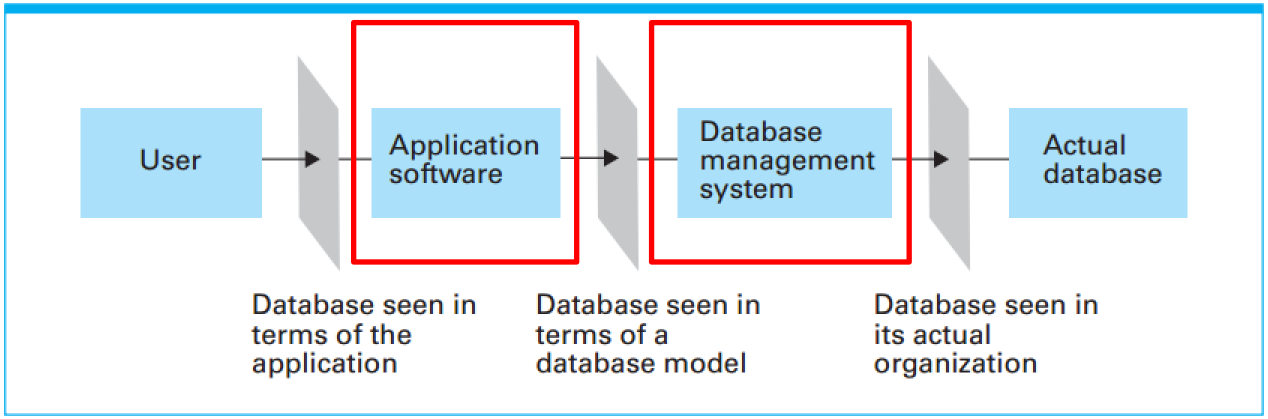
Database Management System (DBMS)

- Complete software system used to define, create, manage, update and query a database



DBMS: Database Management System

- Database application involves two major layers:
  - application layer
  - database management layer

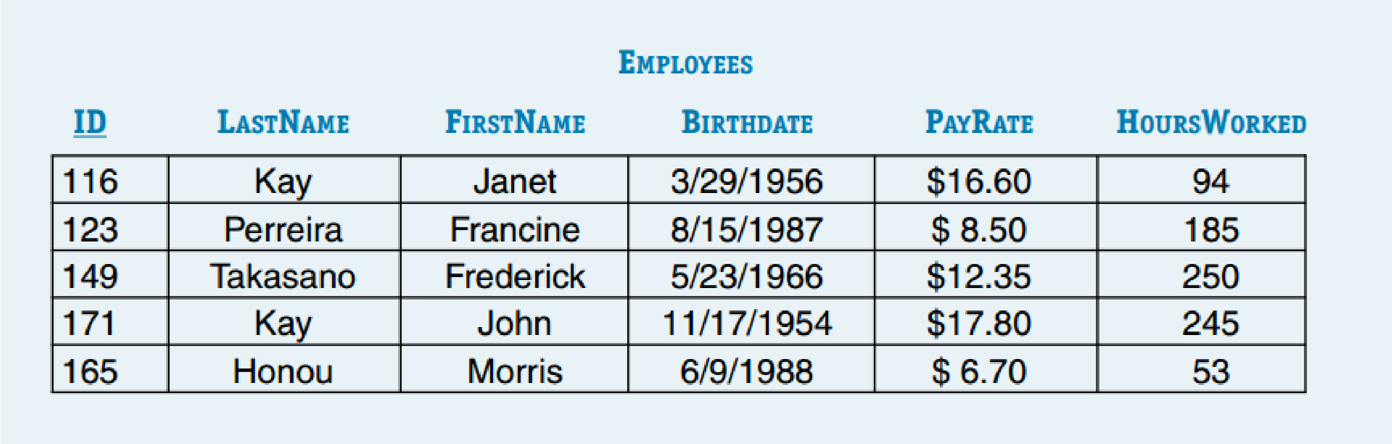

User
Application software
Database management system
Actual database
Database seen in terms of the application
Database seen in terms of a database model
Database seen in its actual organization

How do you describe and store the data in a DBMS?

Data Models

- **High-level description** of the data, which hides **low-level storage details**
- Users define how data is organized in a **DBMS** using the data model
- Examples: Hierarchical model, network model, object-oriented model, object-relational model, **relational model**

Relational data model is most commonly used



EMPLOYEES

ID	LASTNAME	FIRSTNAME	BIRTHDATE	PAYRATE	HOURSWORKED
116	Kay	Janet	3/29/1956	$16.60	94
123	Perreira	Francine	8/15/1987	$ 8.50	185
149	Takasano	Frederick	5/23/1966	$12.35	250
171	Kay	John	11/17/1954	$17.80	245
165	Honou	Morris	6/9/1988	$ 6.70	53

Outline

Database fundamentals

Definition

DBMS

Relational Model

Terminology

ER Model

Terminology

Notation

Converting ER Model to Relational Model

In this module, we will focus on relational data models – Here are some terminologies

Relation	• **Central data description construct** in the model • **Tables** or **set of records**
Relation Instance	• **Snapshot** of a table at a timepoint
Relation Schema	• **Logical structure** of database • **Name** and **type** of each field
Tuple	• **Row** in relation (records)
Attributes	• **Column** in a relation

Relational Model – Example

Student table

Example of **instance** of Student relation

Columns Fields, **Attributes**

Rows

Tuples, Records

studentId	name	nric	Date of Birth	Address
2015001	Nathaniel	S9434567H	10/20/1994	Blk 123 Ang Mo Kio
2005002	Cindy	S9445678J	7/8/1994	Blk 45 Sengkang
2005003	David	S9456789B	3/20/1994	Blk 108 Bukit Batok

Instance of Table Students has 3 rows.

Each row has 5 columns

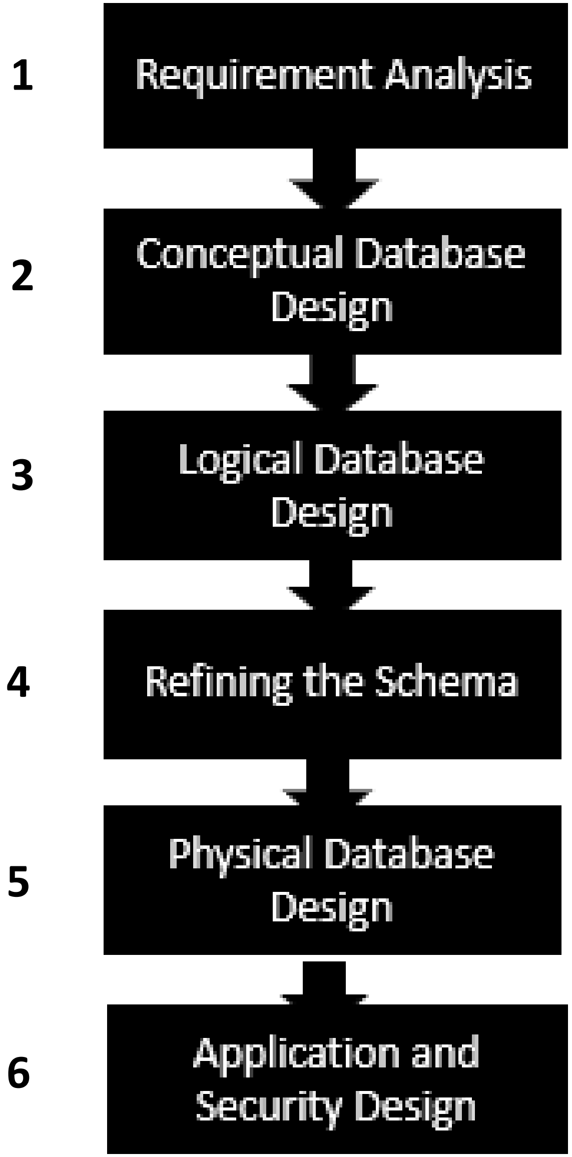
Schema

Students(studentId: string, name: string, nric: string, dateofbirth: date, address: string)

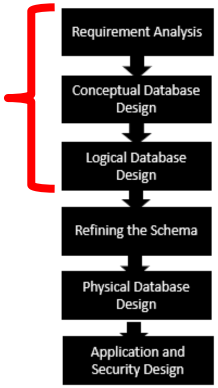
Relational Data Model

- A database is a collection of **one or more relations**
- Each **relation** is a **table with rows and columns**
- Simple data representation
- Enables users to express queries easily

Stages of Database Design - Overview



Stages of Database Design


Requirement Analysis

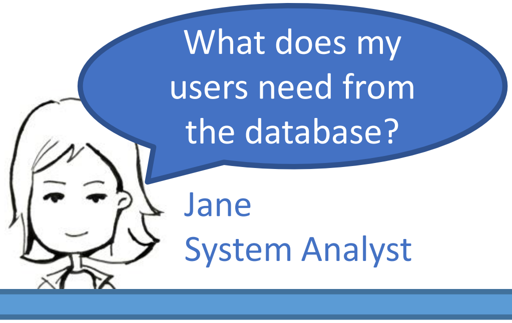
Conceptual Database Design

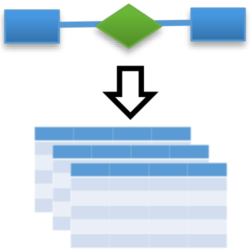
Logical Database Design

Refining the Schema

Physical Database Design

Application and Security Design





1. Requirement Analysis

- Understand **what data** needs to be stored in the database
- What **kind of applications** uses the data?
- What **operation** needs to be performed?

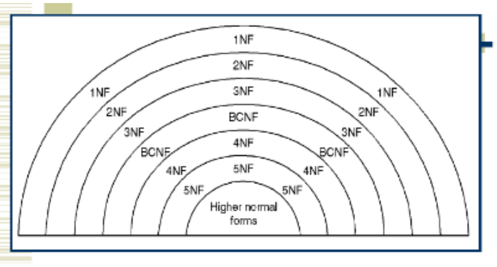
2. Conceptual Database Design

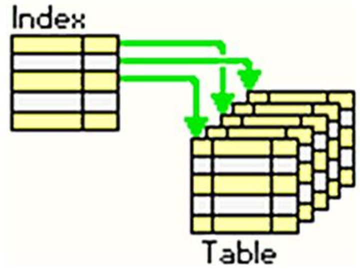
- Uses information from requirement analysis to develop a **high-level description** of the data
- Understand what are the **constraints** that need to be modeled
- Uses the **ER model** for modeling

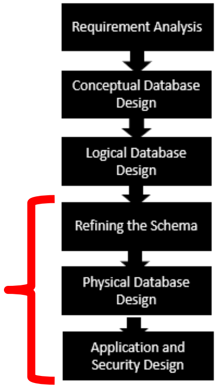
3. Logical Database Design

- Conceptual database design -> **Database Schema**
- Determine the **DBMS** that will be used to implement the database design
- Convert ER to a **relational database** schema

Stages of Database Design







Requirement Analysis

Conceptual Database Design

Logical Database Design

Refining the Schema

Physical Database Design

Application and Security Design

4. Schema Refinement

- Analyze the relations to **identify** potential problems and to **refine** it
- Use design theory for **normalization**

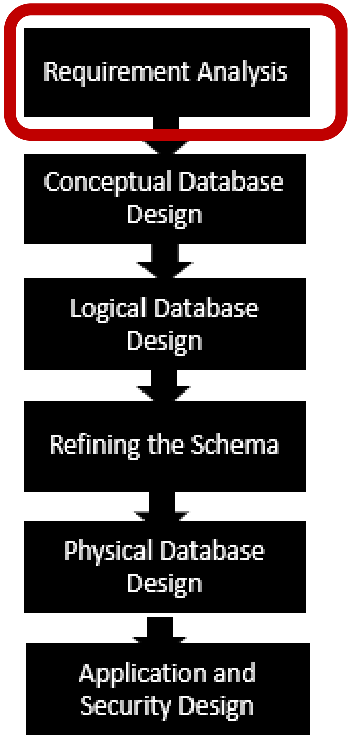
5. Physical Database Design

- Consider typical workloads and **further refine** database design to ensure it meets **performance criteria**
- Build **indexes** on tables and cluster some tables

6. Application & Security Design

- Identify parts of database that must be **accessible** and parts that must not be accessible
- Ensure **access rules** are enforced

Requirements Analysis - Example

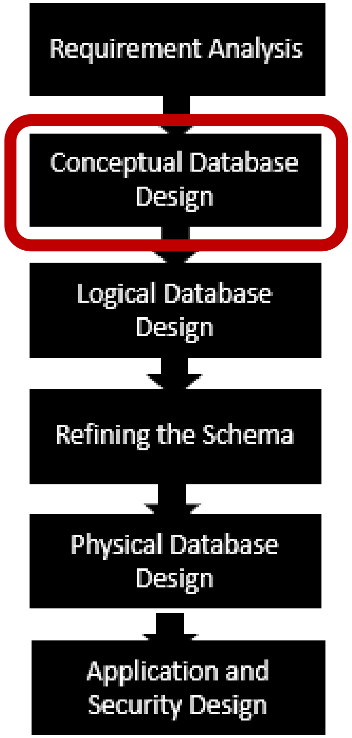


Employee Management System

- We would like to develop a database application to keep information of our Employees.
- We would like to keep track of the employee information such IC, name, and parking lot.
- We would also like to keep track of departments through their unique ID, name and budget. In addition, we would like to keep track of the department that the employee is working in and the time the employee joined.
- A department is managed by one employee. The same employee can manage multiple departments.

Given user requirements, what tools do I have to model the requirements before creating the relevant tables in a DBMS?

ENTITY-RELATIONSHIP (ER) DIAGRAM



Requirement Analysis

Conceptual Database Design

Logical Database Design

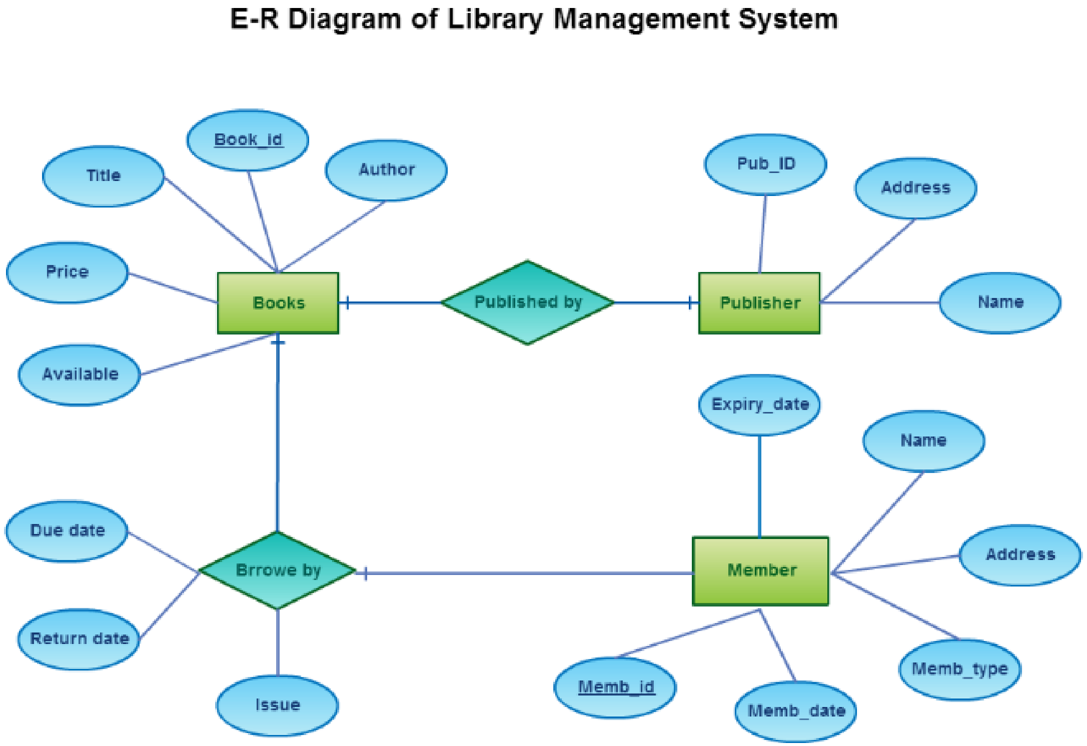
Refining the Schema

Physical Database Design

Application and Security Design

Entity-Relationship (ER) model

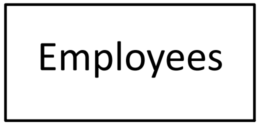
- **Conceptual** database design
- **Abstract, high-level data** model that enables users to translate requirements to the model (before mapping it to a relational model)

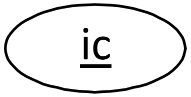


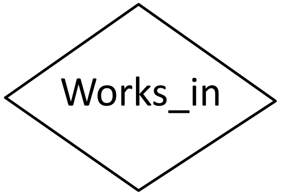
> *"The entity-relationship model adopts the more natural view that the real world consists of entities and relationships."* - **Peter Chen,** Inventor of the Entity Relationship Model

ER Model - Terminology

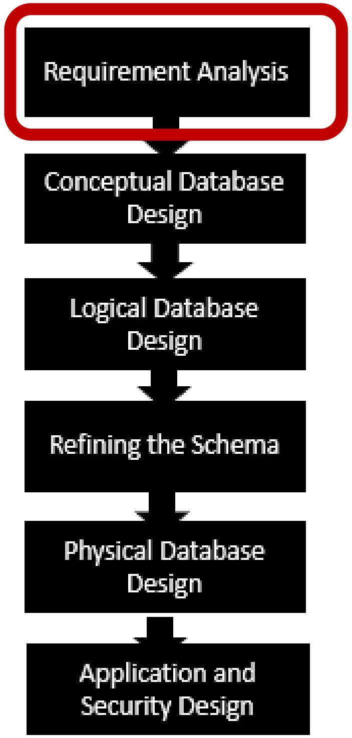
Entity	• Abstract **object** of some sort; a thing (noun)
Entity Set	• Collection of similar entities; class of objects
Attributes	• **Properties** of entities in an entity set
Keys of relations	• Minimal set of attributes whose values **uniquely identify** an entity in the set
Relationship	• **Connection** among two or more entity sets
Relationship Set	• Collection of similar relationships
ER Diagram	• Graph representing the entity sets, attributes and relationships
Instances	• A **snapshot** of a set of relationships







Requirements Analysis - Example



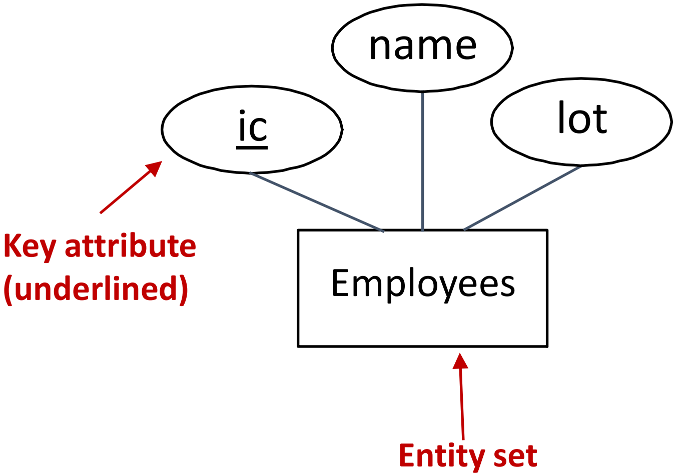
Employee Management System

- We would like to develop a database application to keep information of our Employees.
- We would like to keep track of the employee information such IC, name, and parking lot.
- We would also like to keep track of departments through their unique ID, name and budget. In addition, we would like to keep track of the department that the employee is working in and the time the employee joined.
- A department is managed by one employee. The same employee can manage multiple departments.

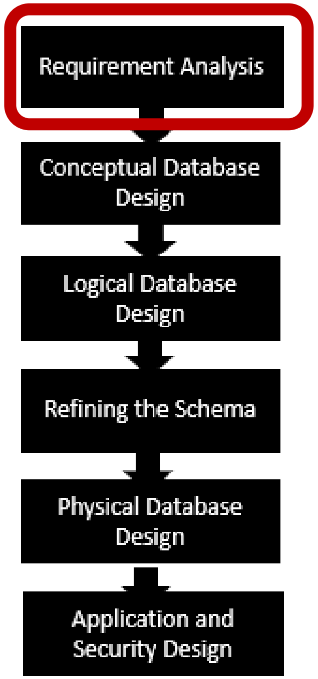
ER Diagram - Notation

- **Entity Set**: rectangle
- **Attribute:** oval, with a line to the rectangle representing the entity set
  - Entity set Employees has three attributes: IC, name and lot
  - Each Employees entity has values for these three attributes
  - A *key* is a **minimal** set of attributes that ***uniquely*** identifies an entity.

We would like to keep track of the employee information such IC, name, and parking lot.



Requirements Analysis - Example



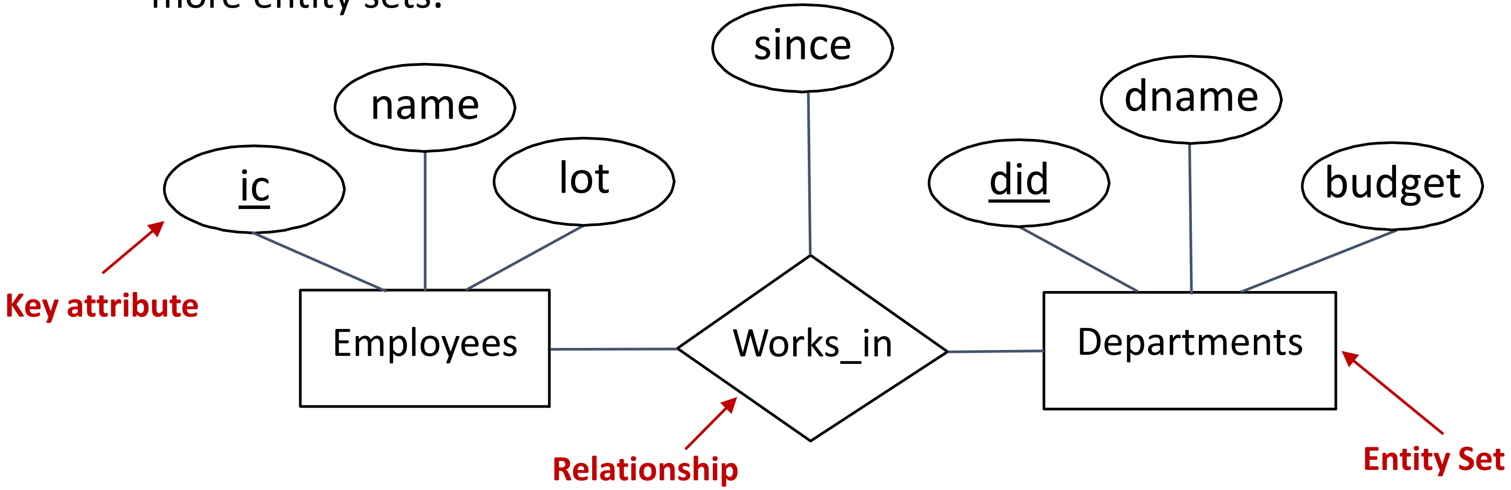
Employee Management System

- We would like to develop a database application to keep information of our Employees.
- We would like to keep track of the employee information such IC, name, and parking lot.
- We would also like to keep track of departments through their unique ID, name and budget. In addition, we would like to keep track of the department that the employee is working in and the time the employee joined.
- A department is managed by one employee. The same employee can manage multiple departments.

ER Diagram - Notation

- **Relationship:**
  - Diamond shape, with lines to each of the entity sets involved.
  - A relationship **connects** two or more entity sets.

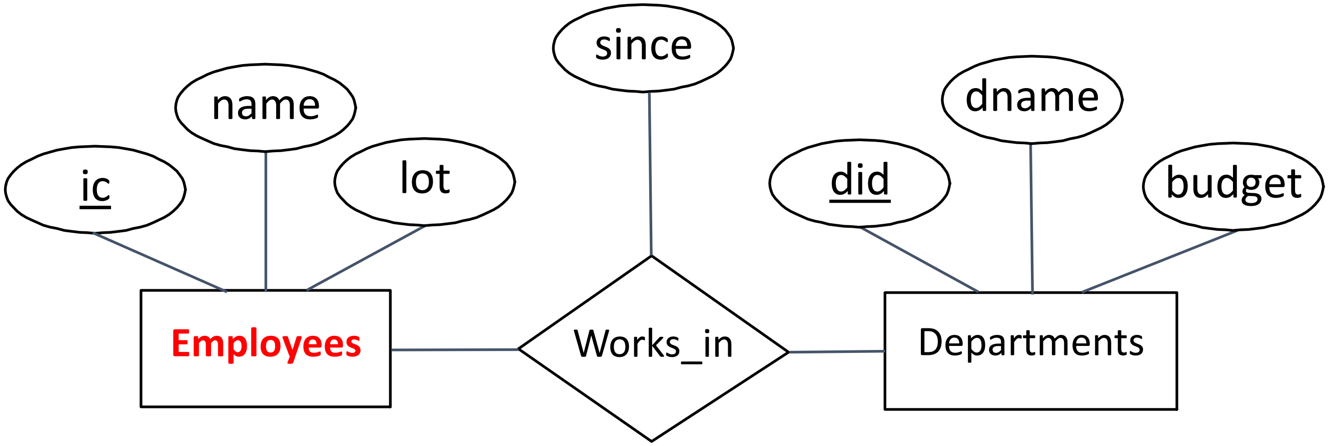
We would also like to keep track of departments through their unique ID, name and budget. In addition, we would like to keep track of the department that the employee is working in and the time the employee joined.

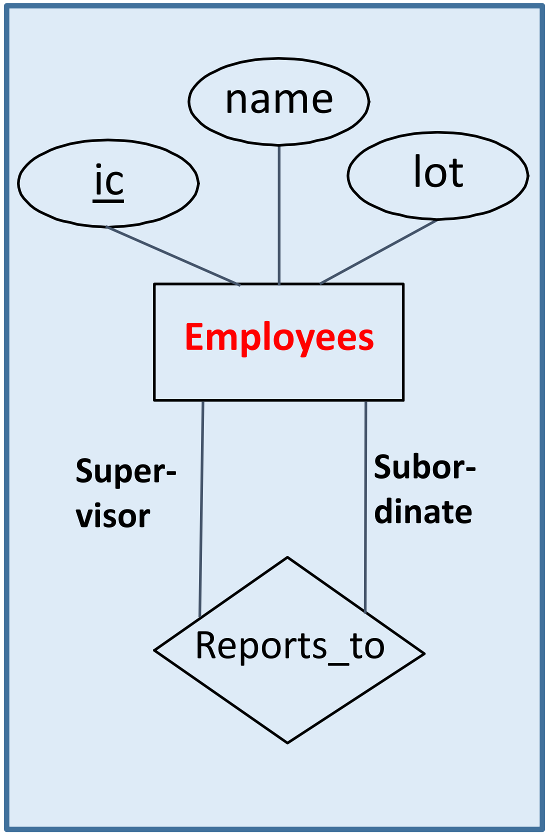


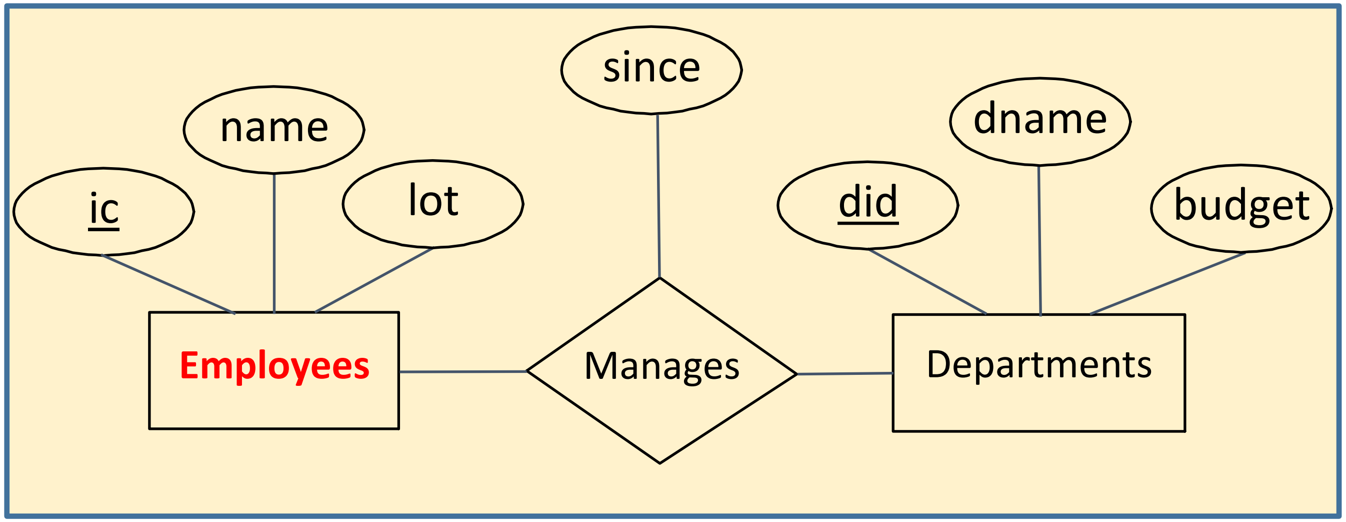
E.g. Jim works in the Pharmacy department since January 1992

ER Model – Notation

- Same entity set could participate in different relationship sets, or in different “roles” in same relationship.

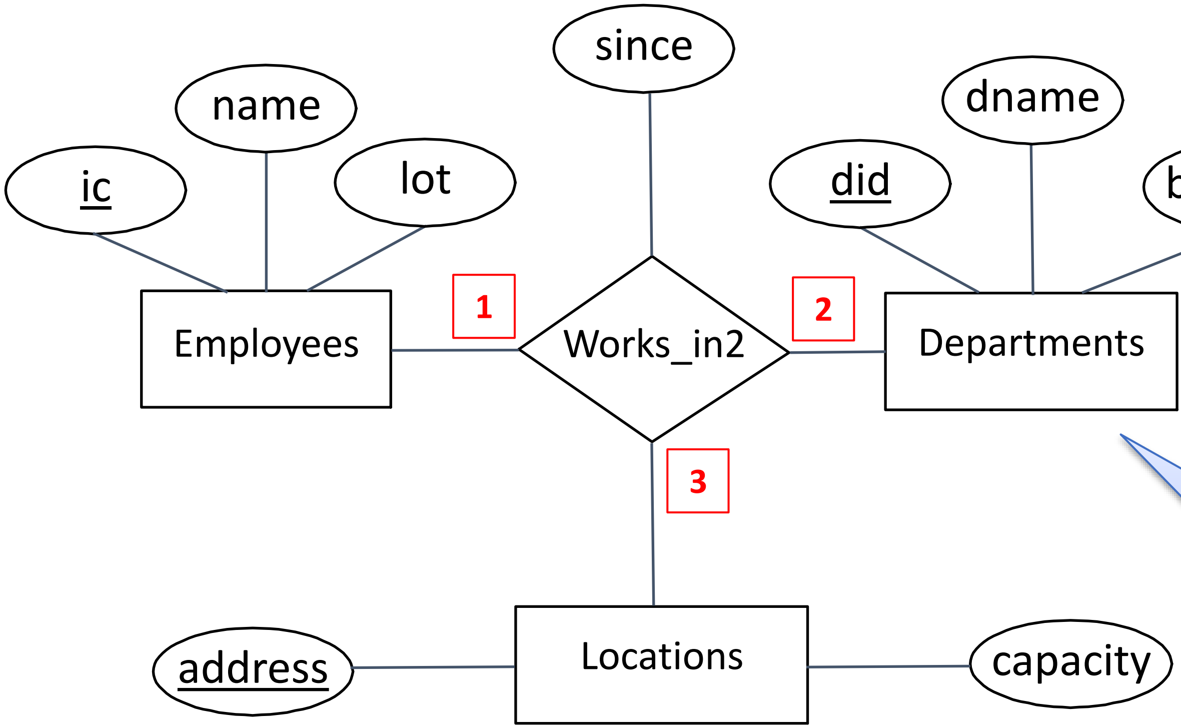






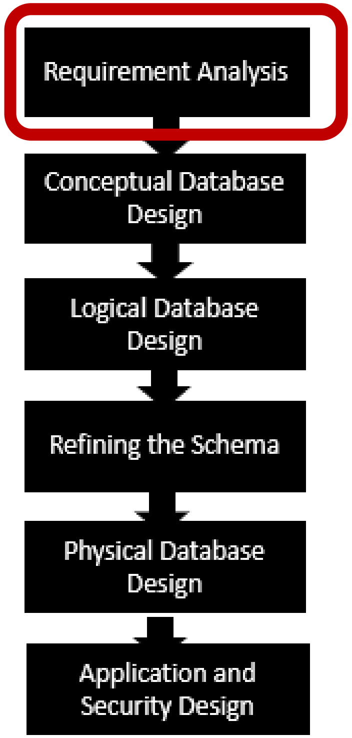
ER Diagram – Multi-way Relationship

- Sometimes, relationships may connect to more than two entity sets.



This is called a ternary relationship (not binary)

Requirements Analysis - Example



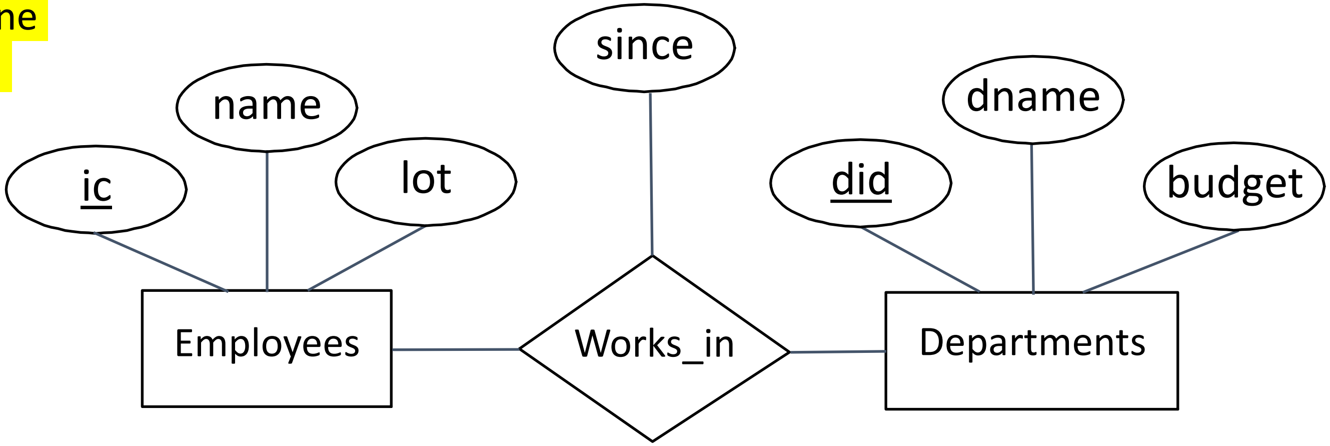
Employee Management System

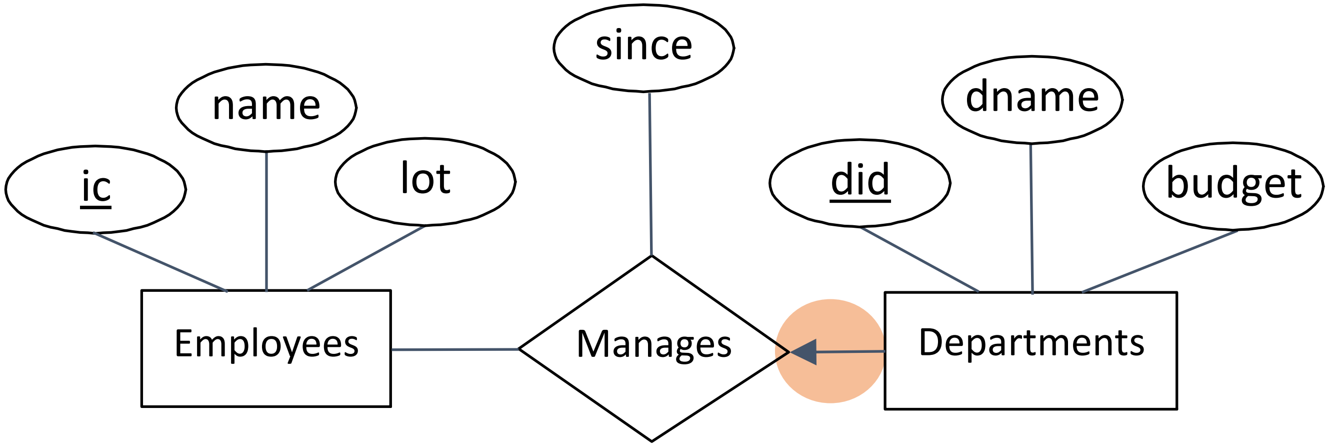
- We would like to develop a database application to keep information of our Employees.
- We would like to keep track of the employee information such IC, name, and parking lot.
- We would also like to keep track of departments through their unique ID, name and budget. In addition, we would like to keep track of the department that the employee is working in and the time the employee joined.
- A department is managed by one employee. The same employee can manage multiple departments.

ER Model – Relationship with Key Constraints

A department is managed by one employee. The same employee can manage multiple departments.

- Consider Works_In: An employee can work in *many* departments; a dept can have *many* employees.
- In contrast, each dept has at most *one* manager, according to the key constraint on Manages. (Relationship with arrowhead)

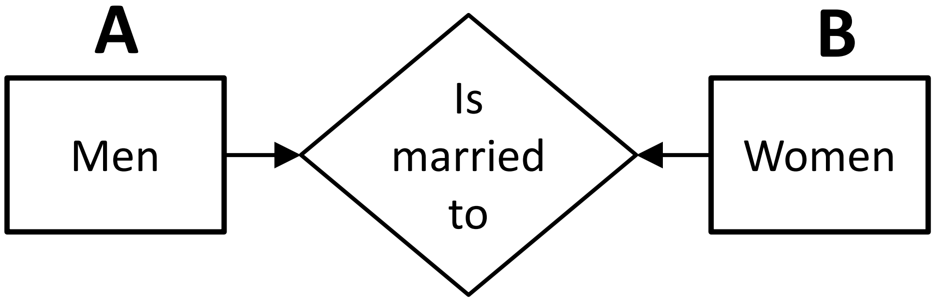




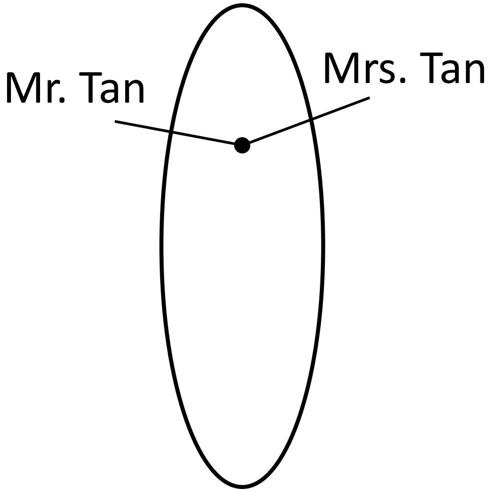
***In this context, we will use arrows pointing only to relationship diagram – to show constraint**

ER Relationship - One to One

- An entity in A is associated with at most one entity in B and an entity in B is associated with at most one entity in A.

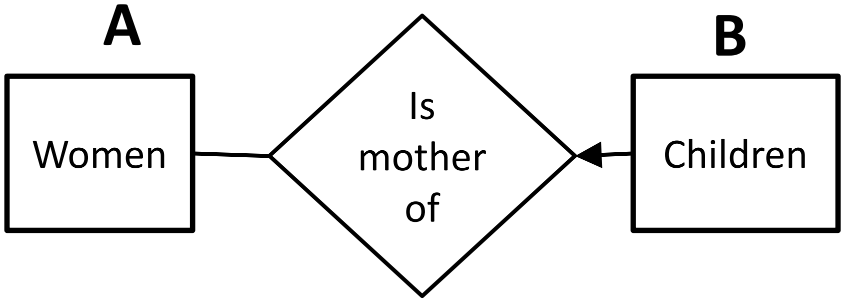


In most countries, a man can only be married to one woman at a single time – monogamous marriage.

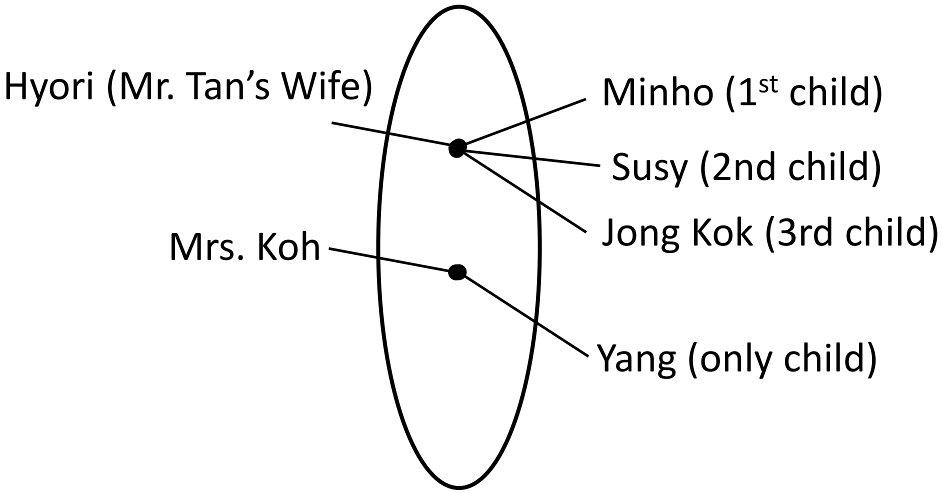


ER Relationship - One to Many

- An entity in A is associated with any number in B. An entity in B is associated with at most one entity in A.

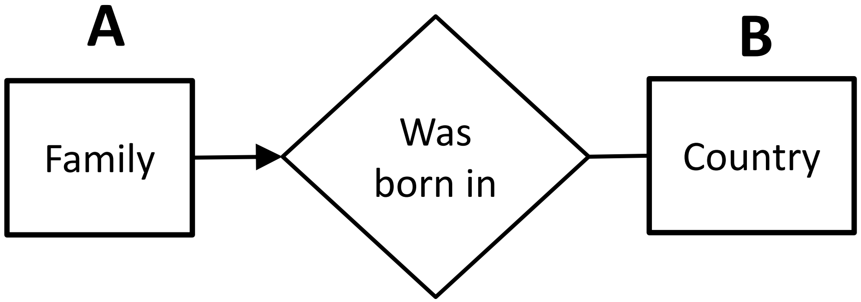


A woman can have more than one child but a child can only have one mother (we are referring to natural mother and not step-mother or god-mother here…)

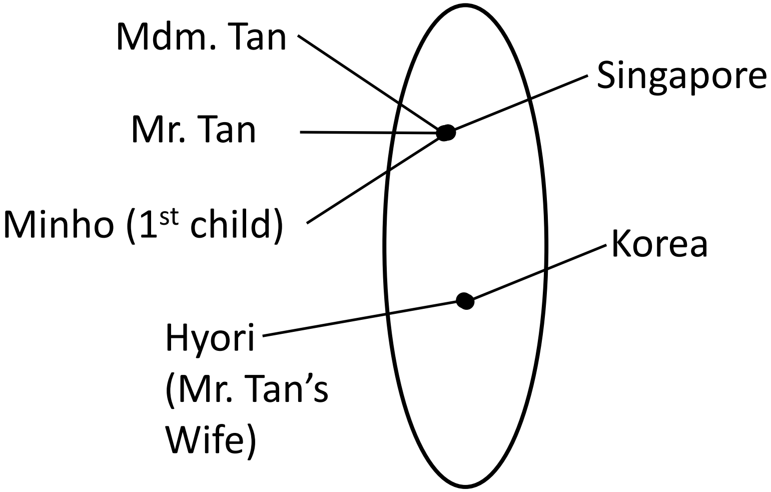


ER Relationship - Many to One

- An entity in A is associated with at most one entity in B.
- An entity in B is associated with any number in A.

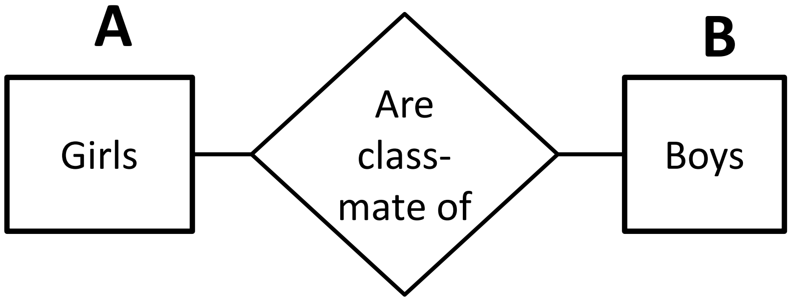


In most countries, dual citizenships are not allowed. Hence, most people (families) belong to only one country. A country can have many people (families).

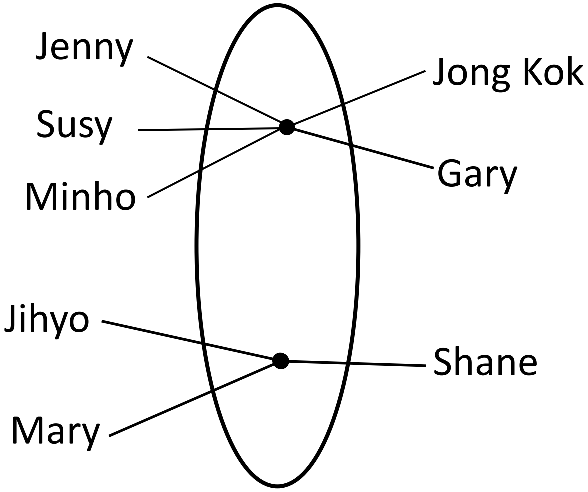


ER Relationship - Many to Many

- Entity in A and B are associated with <u>any numbers</u>

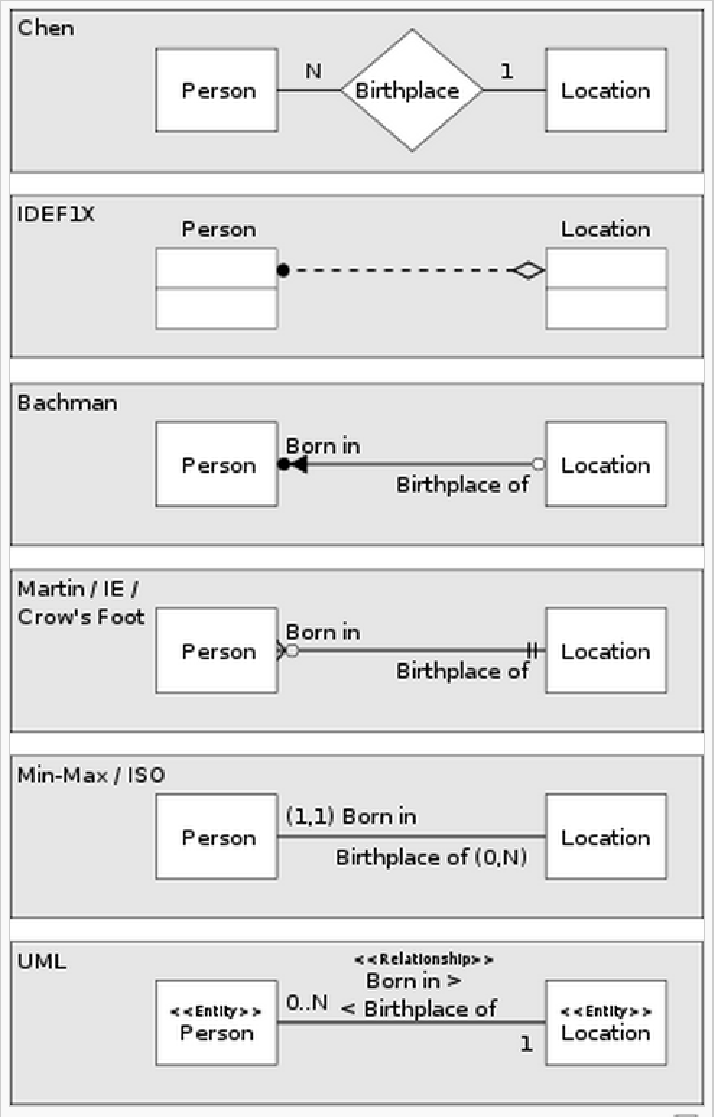


In a large class, typically there are many boys and girls. So, many girls are classmates of many boys and vice-versa



Different Notations

- Various notation to represent the same one to many relationship
- ER Notation: crow's foot, one or many, zero or many
- In this module, we will be using diamond to represent relationships, and arrows for key constraints in relationships



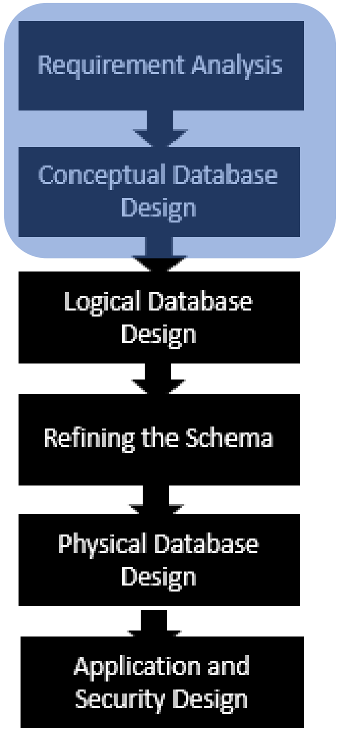
ER Model

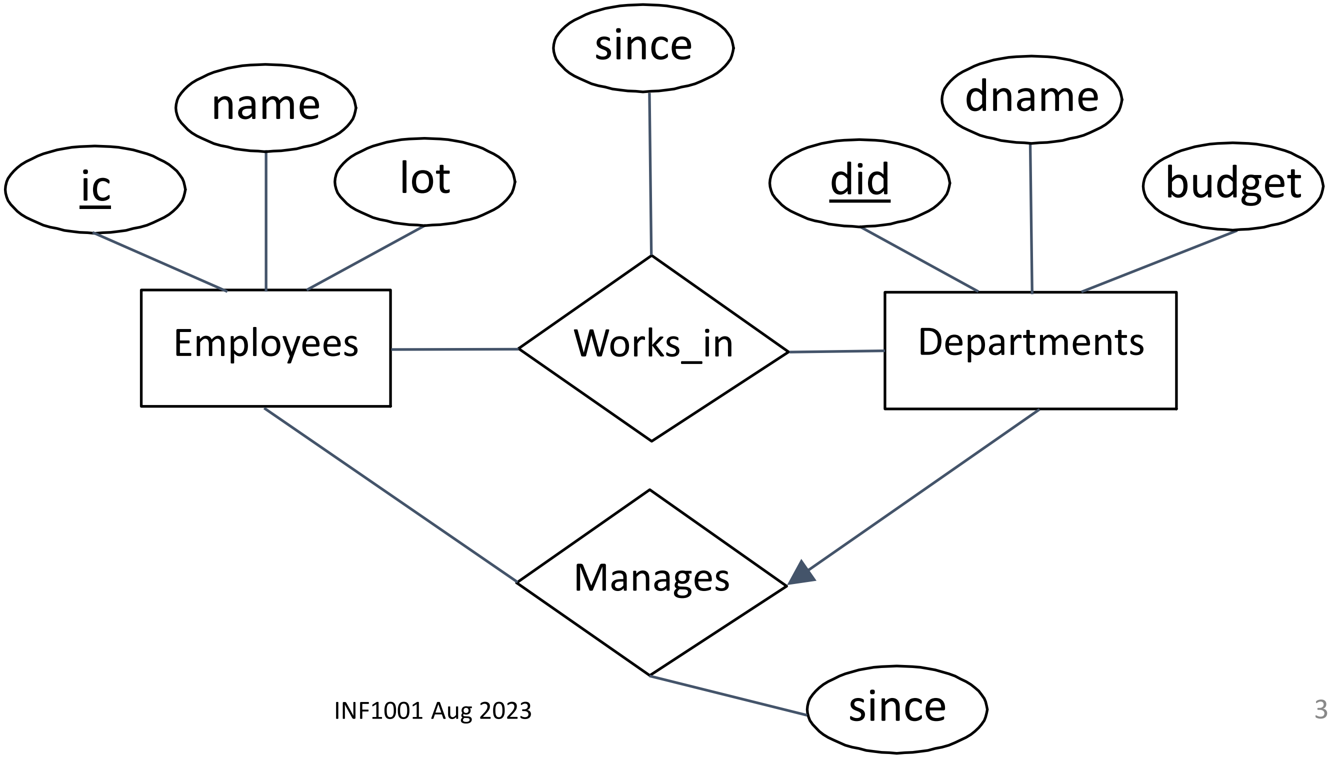
- High-level view of system's data structure:
  - Allow stakeholders to visualize/discuss relationships between entities
  - Provide foundation to design database schema
- Design choices for ER:
  - Modeling a concept as entity, attribute or relationship
  - What type of relationship? Any constraints?
- ER design is *subjective*.
  - Often many ways to model given requirements!
  - Analyzing alternatives can be tricky, especially for a large enterprise
  - Important to ask questions, and keep clarifying requirements with clients
  - Important pre-step before jumping in to develop database schema

From Requirements to Conceptual Database Design (ER)

Employee Management System

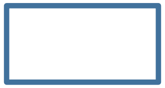
- We would like to develop a database application to keep information of our Employees.
- We would like to keep track of the employee information such IC, name, and parking lot.
- We would also like to keep track of departments through their unique ID, name and budget. In addition, we would like to keep track of the department that the employee is working in and the time the employee joined.
- A department is managed by one employee. The same employee can manage multiple departments.



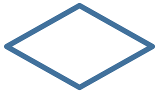


Summary ER Diagram

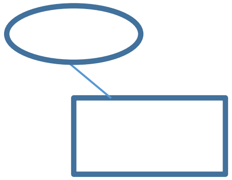
Entity



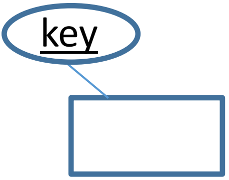
Relationship



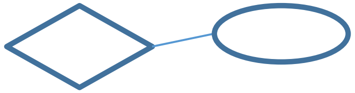
Entity Attributes



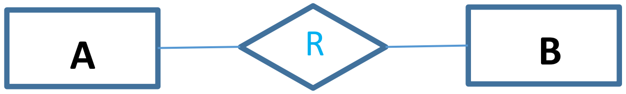
Entity Key attributes



Relationship attributes

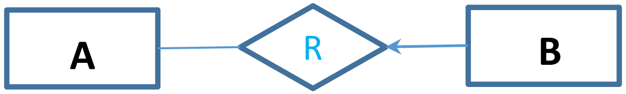


Relationships without constraints



Each entity B appears in many R relationship and each entity A appears in many R relationship
E.g. An Employee works in different department and a department has many employees

Key constraints



Each entity B appears in one R relationship, OR each entity B R relationship one A. E.g. Each employee (B) manages (R relationship) **one** department (A)

Database fundamentals

Definition

DBMS

Relational Model

Terminology

ER Model

Terminology

Notation

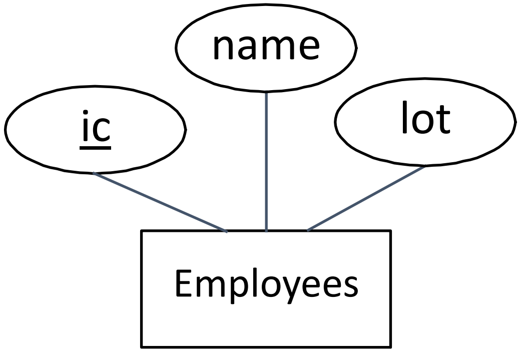
Converting ER Model to Relational Model

From ER Diagrams to Relational Schema

- Key concept:
  - Both ***Entity sets*** and ***Relationships*** become relations (tables in RDBMS)

Entity Sets to Tables

- Entities are translated to relations/table
  - Each tuple is one entity
  - Each tuple is composed of the entity’s attributes, and has the same primary key



ER Model

Relational Schema

Schema

`Employees (ic: varchar, name: string, lot: integer)`

i.e. table name
Relation

Employees

ic	*name*	*lot*
123-22-3666	Attishoo	48
231-31-5368	Smiley	22
131-24-3650	Smethurst	35

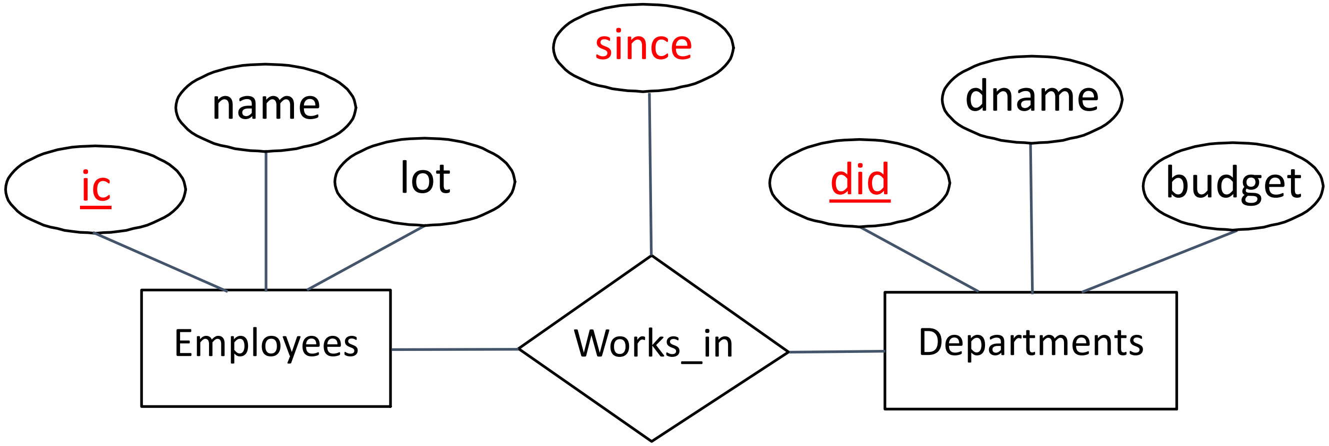
Attributes
i.e. column

Tuple
i.e. row

Instance of Relation Employees

Relationship Sets Without Constraints to Tables - 1

- Relationships without any constraints are translated to relations/table
- In translating a relationship set to a relation, attributes of the relation must include:
  - Primary key of the relationship is the primary key from each of the participating entity set
  - All descriptive attributes of the relationship

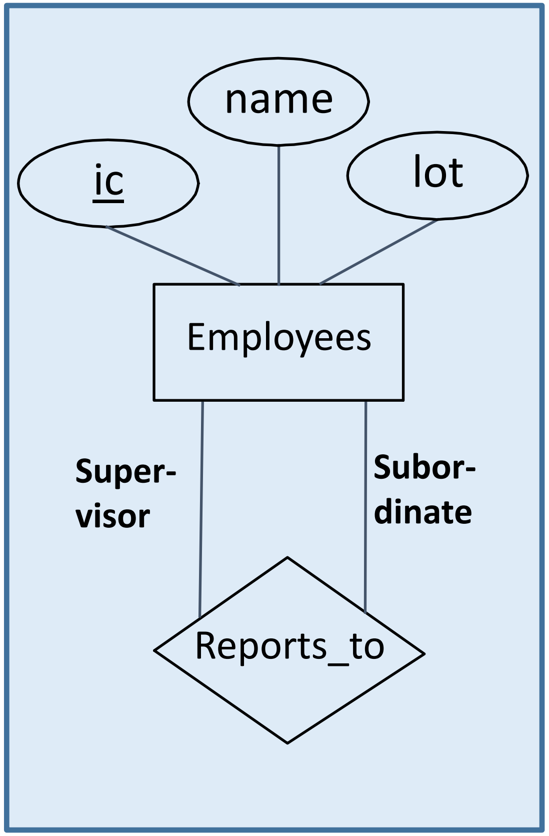


```
Works_in(ic: char(11), did: int, since: date)
```

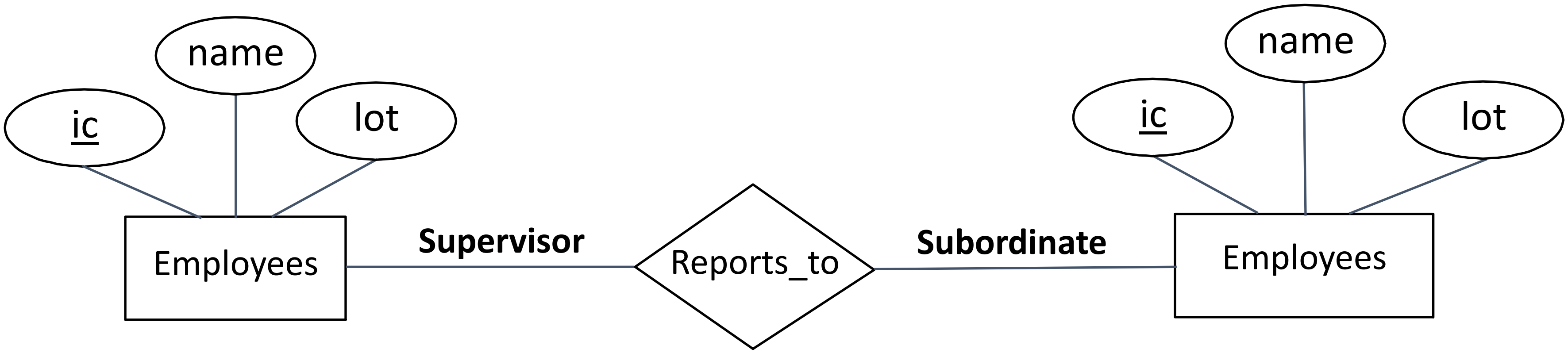
Works_in

ic	did	since
123	1	05/09/14
145	2	02/08/16
199	2	05/9/15

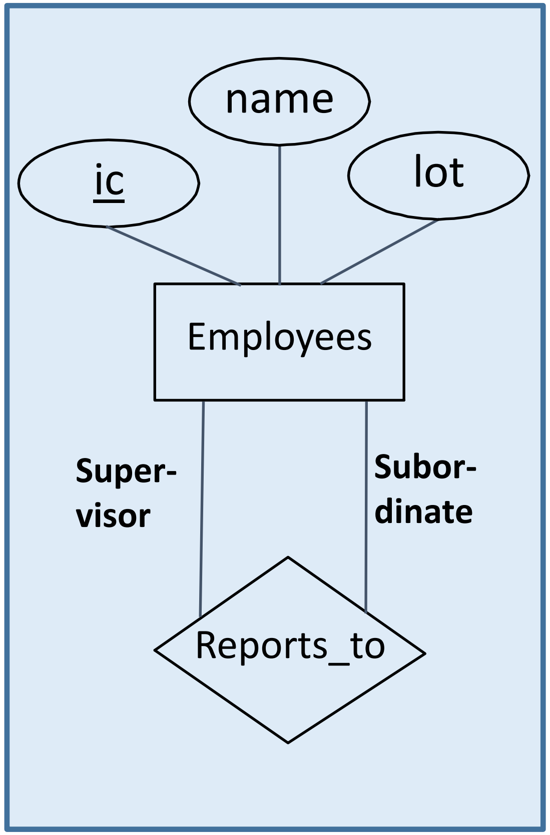
Relationship Sets Without Constraints to Tables - 2



Relationship Sets Without Constraints to Tables - 2



Relationship Sets Without Constraints to Tables - 2



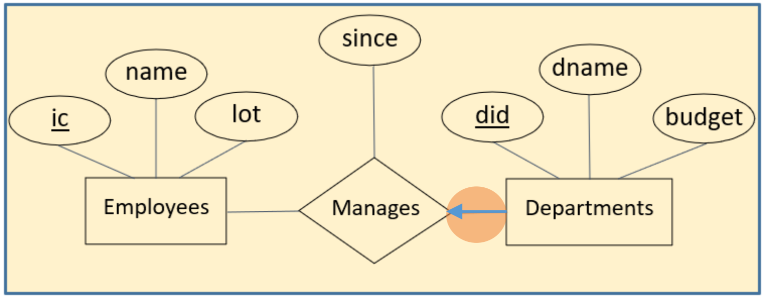
```
Reports_to(supervisor_ic: char(11), subordinate_ic: char(11))
```

Reports_to

Supervisor_ic	Subordinate_ic
123	145
123	199
199	255

Relationship Set with Key Constraints - 1

- Each dept **has at most one** manager, according to the *key constraint* on Manages
- **did** should be the key for Manages



```
Manages (did: int, ic: char(11), since: date)
```

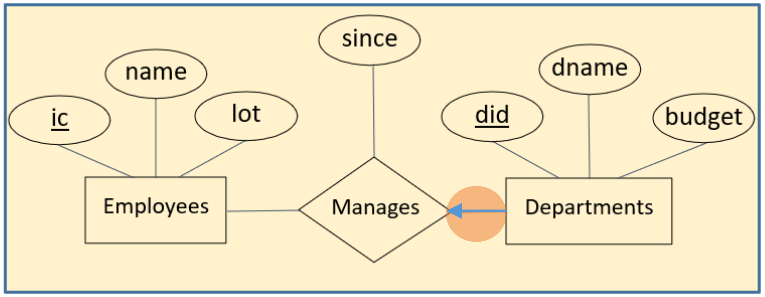
Manages

did	ic	since
1	1	05/09/2013
2	2	04/03/2015
3	2	02/06/2014

Only one entry per did

Relationship Set with Key Constraints - 2

- Relationships with **key constraints** are *merged* with the constrained entity to form one relation/table
- Since each department has a unique manager, we could instead **combine** Manages and Departments.



```
Departments (did: int, dname: char(50), budget: int)
```

```
Manages (did: int, ic: char(11), since: date)
```

```
Dept_mgr (did: int, dname: char(11), budget: real, ic: char(11), since: date)
```

Dept_mgr

did	dname	budget	ic	since
1	Finance	10k	1	05/09/2013
2	HR	5k	2	04/03/2015
3	Transport	20k	2	02/06/2014

Summary –
Rules to Translate ER diagram to Relational Model

1. Entities are translated to relations/table

2. Relationships without any constraints are translated to relations/table.

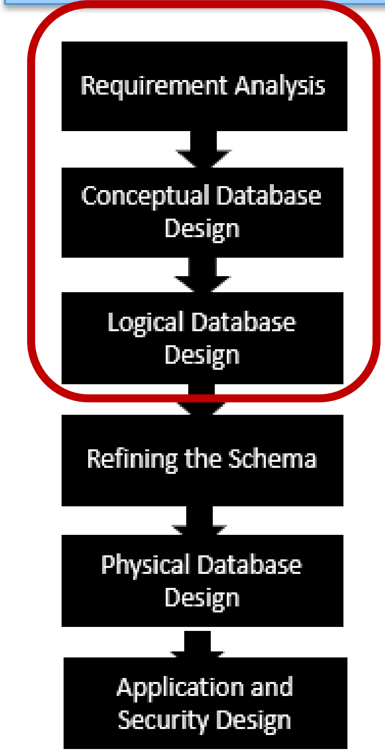
3. Primary key of the relation is a combination of primary keys of all entities involved in the relationship

4. Relationships with **key constraints** are *merged* with the constrained entity to form one relation/table

From Requirements to Conceptual Database Design (ER)

Employee Management System

- We would like to develop a database application to keep information of our Employees.
- We would like to keep track of the employee information such IC, name, and parking lot.
- We would also like to keep track of departments through their unique ID, name and budget. In addition, we would like to keep track of the department that the employee is working in and the time the employee joined.
- A department is managed by one employee. The same employee can manage multiple departments.



Requirement Analysis

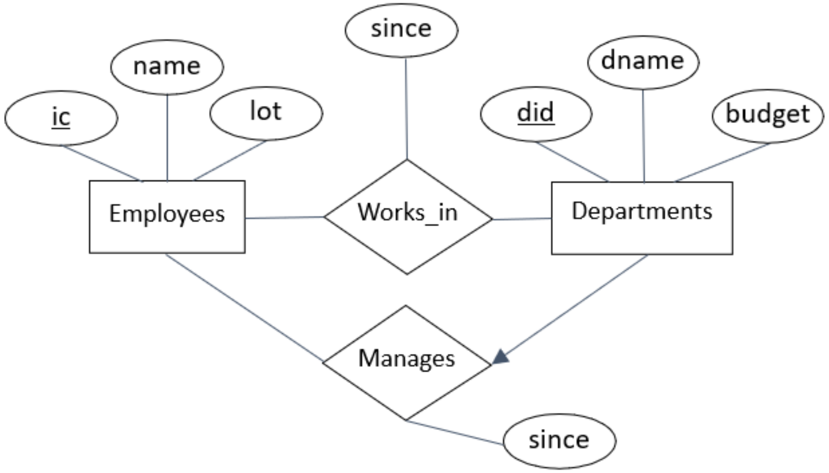
Conceptual Database Design

Logical Database Design

Refining the Schema

Physical Database Design

Application and Security Design



Relational Schema

```
Employees (ic: varchar, name: char, lot: integer)
Works_in(ic: char(11), did: int, since: date)
Dept_mgr (did: int, dname: char(11), budget: real,
          ic: char(11),  since: date)
```

Summary

Database fundamentals

Definition

DBMS

Relational Model

Terminology

ER Model

Terminology

Notation

Converting ER Model to Relational Model

References

- https://www.lucidchart.com/pages/database-diagram/database-models
- *Chapter 7 Software Engineering; Chapter 9 Database Systems*, Computer Science: An Overview, 11th Edition, Addison-Wesley, J. Glenn Brookshear
- *Chapter 4 Domain Modelling*, Systems Analysis Design in a Changing World, 6th Edition, Course Technology, John W. Satzinger