CPSC 304
Introduction to Database Systems

Conceptual Database Design:
The Entity-Relationship Model
(Part 1 of 2)

Textbook Reference:

Database Management Systems, Chapter 2

Fall 2017

Learning Goals

- Explain the purpose of an ER diagram, and list its major components.
- Read and interpret an ER diagram.
- Given a problem description and specification, create an ER diagram. Justify the decisions you make for entities, relationships, keys, key constraints, participation constraints, weak entities, is-a relationships, aggregations, etc.
- Given a problem description, identify alternative representations of the problem concepts and evaluate the choices.
- Compare alternative ER models for the same domain and identify their strengths and weaknesses.

Bloom's Taxonomy (Re-visited)

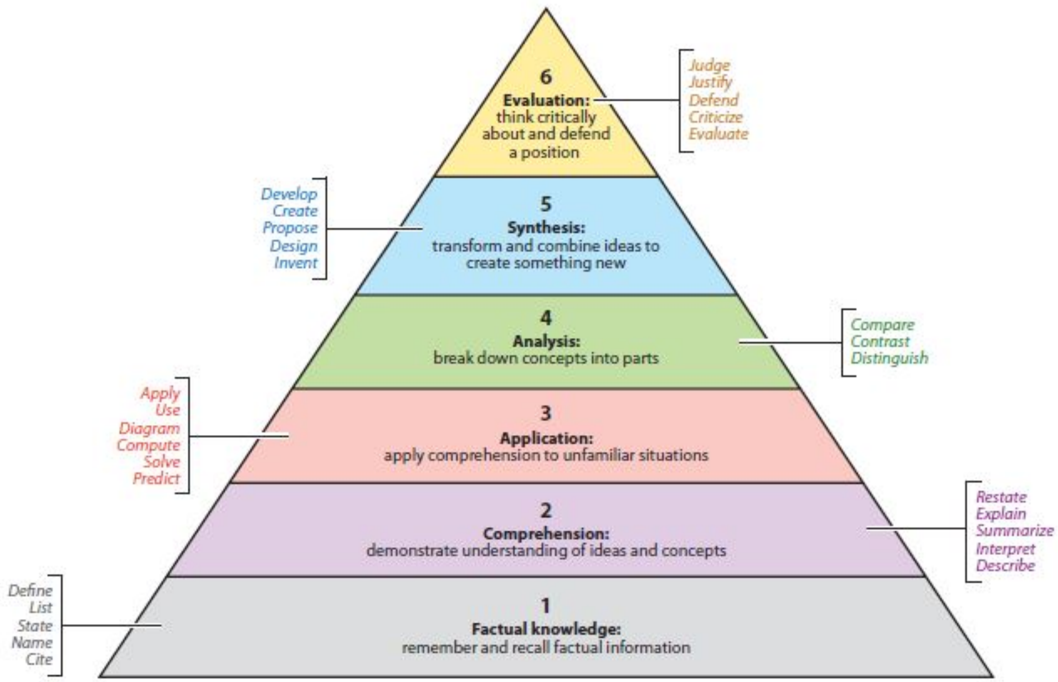


Figure 1

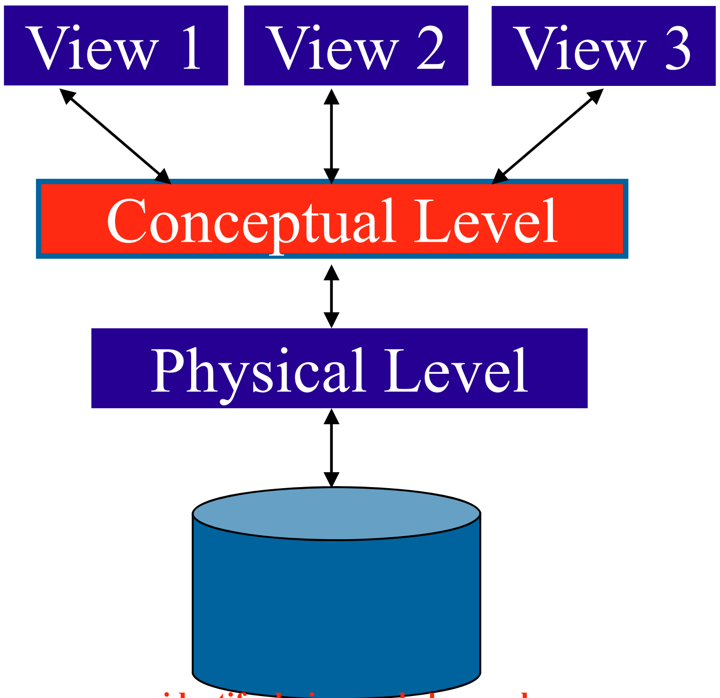
Bloom's levels of understanding. Originally termed Bloom's taxonomy of the cognitive domain, this schema defines six levels of conceptual understanding according to the intellectual operations that students at each level are capable of (Bloom & Krathwohl 1956). The italicized verbs have been added to the original hierarchy; they indicate performance tasks that test achievement of learning goals at each level. Fine distinctions in the hierarchy are difficult, and some educators prefer to classify goals on only three levels: low (1, 2), medium (3, 4), and high (5, 6). (Based on Allen & Tanner 2002.)

Databases: The Continuing Saga …

- Our motivation for using databases includes the ability to:
  - Store large amounts of data in an organized way
  - Handle many transactions, including concurrent ones
  - Allow for efficient querying
  - Manage the data centrally, including backups and recovery
  - Ensure data integrity
  - etc.
- Before we can do all of these, we must *design* the database.

Levels of Abstraction

- A major purpose of a DB and DBMS is to provide an abstract view of the data.
- Three abstraction levels:
  - **Physical level**: how data is actually stored, paying attention to indexing, clustering, partitioning, the I/O cost model, etc.
    - We deal with these in CPSC 404.
  - **Conceptual (or Logical) level**: how data is perceived by the users
  - **External (or View) level**: describes different parts of the database to different users
    - For convenience, security, etc.
    - Compare views of student, registrar, and database administrator—with respect to a University database



-identify design probelms early
-be able to support multiple applications

Good test:
how well does this model support/reflect the business

conceptual design - capture the facts: business requirement about data: entities & relationships
-gain deep understandng of the business
-no single diagram maybe perfect

Schema and Instances

- We create the **schema** – the logical structure of the database (e.g., students take courses)
  - **Conceptual (or logical) schema**: DB design at the logical level
    Logical DB Design: 1. ERD 2. Convert diagra to tables, normalize, etc
  - **Physical schema**: DB design at the physical level
- Later, we'll populate **instances** – the contents of the DB at a particular point in time.
  - e.g., When a database is created, it's initially empty until we load the first set of data (i.e., populate the tables).

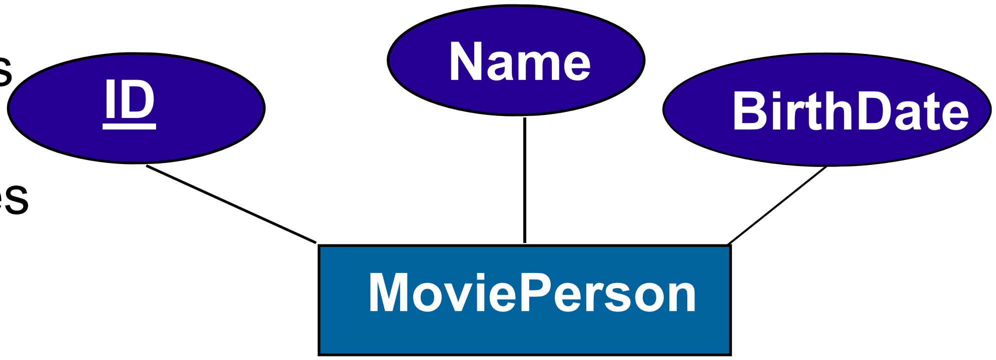
Good design characteristics:
- -simple
- -concise, non-redundant
- -data quality
- -ignore performance for now
- -sufficient scope
- -completeness
- -well organized
- -stable yet flexible
- -reusable

Conceptual Database Design

- What are the *entities* and *relationships* in the *enterprise*?
  - Entities are usually nouns, but avoid irrelevant nouns
    - e.g., Student or Course is good; Person is vague
  - Relationships use verbs, and are statements about 2 or more entities or objects.
    - e.g., a prof teaches a course, a student takes a particular section of a course in a particular term
- What information about these entities and relationships should we store in the database?
- What integrity constraints or other rules hold?
- In relational databases, this whole process generally begins by encoding the information in an **Entity-Relationship (ER) Diagram.**

ER Model Basics: Entities

- ***Entity:*** A real-world object distinguishable from other kinds of objects.
- An entity's features or properties are described using a set of ***attributes*** (e.g., ID, Name).

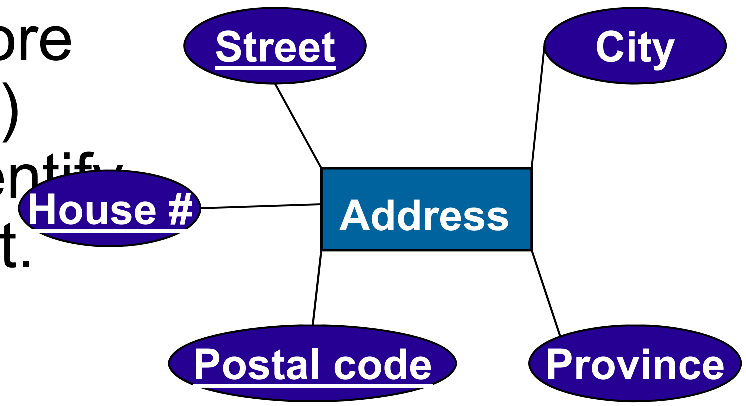


Entity Set: A collection of similar entities (e.g., all the MoviePersons).

- All entities in an entity set have the same set of attributes (at least until we consider ISA hierarchies!)
- Each attribute has a ***domain*** (e.g., float, date, integer, character string).
- Each entity set has a unique ***key*** (e.g., ID).

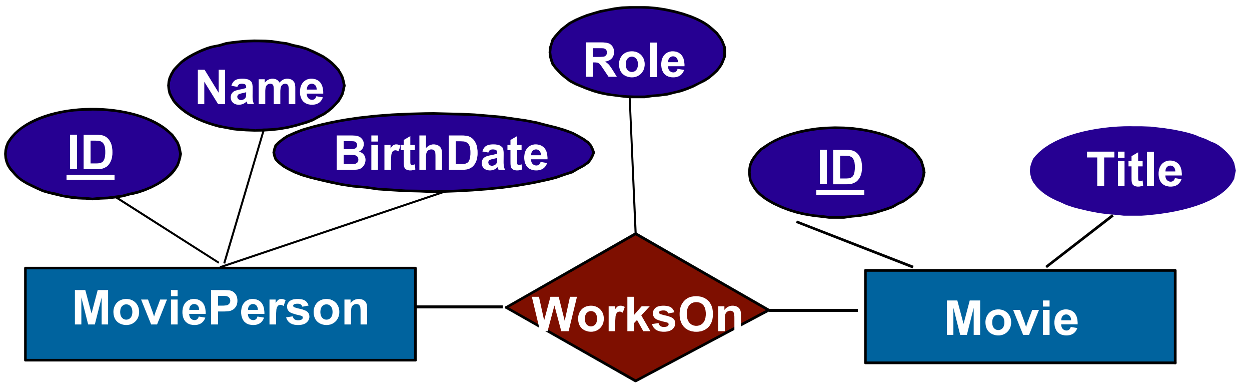
ER Model Basics: Keys

- A key distinguishes specific records of an entity.
- A key is a *minimal* set of one or more attributes (i.e., no excess baggage) which, taken together, uniquely identify a record of an entity in an entity set.
- The *primary key (PK)* is the key chosen as the principal means to identify a given record.
- Primary keys are shown in ER diagrams.
- We'll discuss superkeys when we consider normal forms (more about these, later).



Source: http://dc.wikia.com/wiki/Action_Comics_Vol_1_241

ER Model Basics: Relationships

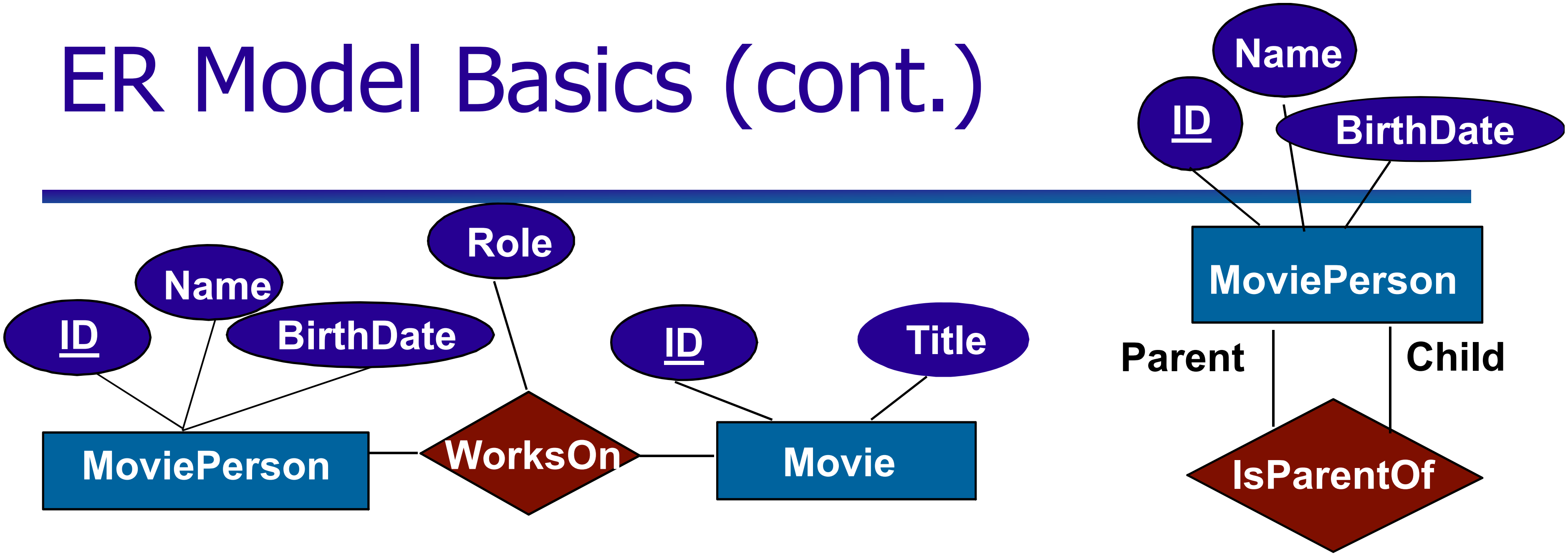


- ***Relationship:*** An association among two or more entities
  - e.g., George Clooney worked on Gravity.

read the diagram in colockwise

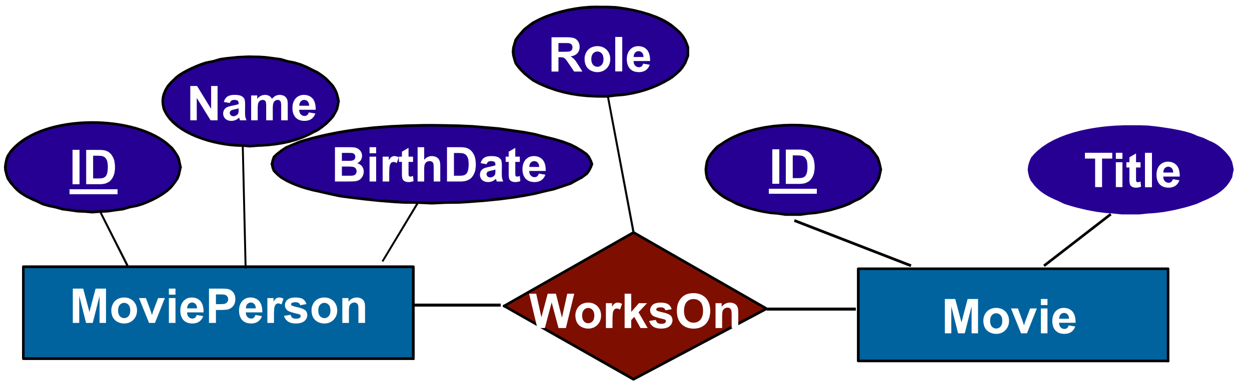
Sources: White House (via Wikipedia, fair use), Wikipedia (fair use)

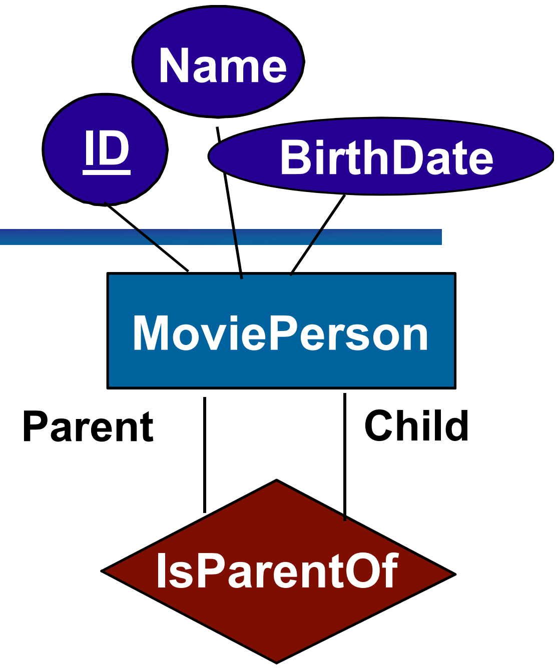
ER Model Basics (cont.)



- ***Relationship Set:*** A collection of similar relationships
  - e.g., The collection of all MoviePersons that have worked on Movies.
- The same entity set could participate in different relationship sets, or in different "roles" in same set (e.g., Kirk Douglas isParentOf Michael Douglas).

ER Model Basics (cont.)





- A relationship set may have ***descriptive attributes*** (like "since").
- An *n*-ary relationship set R relates *n* entity sets $E_1 \ldots E_n$. Each relationship in R involves entities $e_1 \in E_1, \ldots, e_n \in E_n$
  - ***Degree*** or ***arity:*** # of entity sets in the relationship (binary, ternary, etc.) **dimension**

Exercise 1: Radio Station

- A music radio station has various people to make it work including disk jockeys, support personnel, PR people, writers, advertising reps, etc.
- A station might specialize in certain genres of music (e.g., rock, pop, classical, folk).
- A station wants to play popular songs frequently, but there are many choices to be made. Some songs might get forgotten, and not get played; others will be played too often. It will help to have a history of when the songs played on the radio station.
- Callers can make requests for songs.
- Suggest some entities and relationships for a radio station. Think of some attributes for those entities and relationships.

Motivating Case Study, Music: We'll Look at Some of these Examples in Days to Come

- A radio station plays music (e.g., songs).
- An artist is a person.
- Examples of artists are Beyoncé, Adele, Justin Bieber, Sting, Gordon Lightfoot, Paul Simon, Burton Cummings, etc.
- A group (or band) is two or more artists working together. In other words, a group is an aggregation of two or more artists.
- Examples of groups are U2, Beatles, Fleetwood Mac, Bachman-Turner Overdrive, Simon & Garfunkel, etc.
- An artist may or may not be in a group.
- An artist may be in more than one group.

Some of this example is from: Teorey, Toby; Lightstone, Sam; Nadeau, Tom; and Jagadish, H.V. *Database Modeling and Design: Logical Design*, 5th Edition, Morgan Kaufmann, Elsevier, 2011.

Case Study: Music (cont.)

- Composers, lyricists, and musicians are different types of artists.
- A song is associated with one or more composers.
- A song may have any number of lyricists, including zero.
- A song may have any number of renditions (performances or versions).
- A rendition is associated with exactly one song.
- A rendition is associated with musicians and instruments.
- A given musician–instrument combination is associated with any number of renditions.
- A rendition-musician combination is associated with any number of instruments.
- A rendition-instrument combination is associated with any number of musicians.

Case Study: Music (cont.)

- Music is created/placed on, and distributed on, media (which physically hold the recordings).
- Albums, CDs, computer files, etc. are types of music media.
- Music is composed of tracks.
- Tracks are associated with renditions.
- Media is distributed. (This brings the music to you.)
- Music media is associated with groups/artists, publishers, recording studios, and producers.
- Music distribution involves companies (sometimes called labels (e.g., Columbia, Epic, Capitol)).

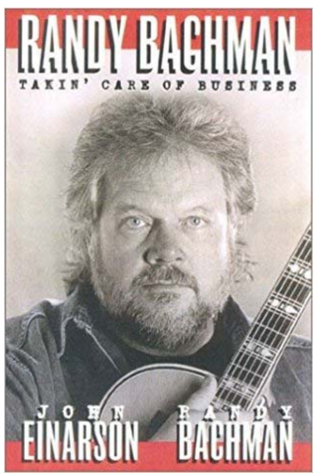
Example: Album, Distribution, Group

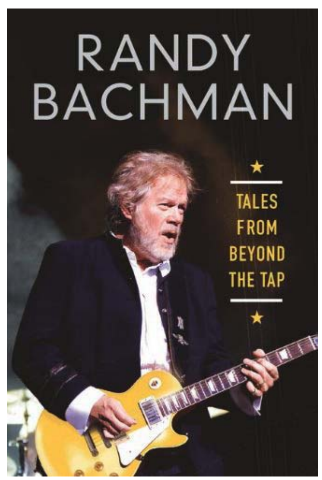
Greatest hits album by Simon & Garfunkel	
Released	June 14, 1972
Recorded	March 1964 – November 1969
Genre	Folk rock
Length	43:25
Label	Columbia
Producer	Paul Simon, Art Garfunkel, Roy Halee

Simon & Garfunkel chronology		
Bridge over Troubled Water (1970)	***Simon and Garfunkel's Greatest Hits*** (1972)	*Collected Works* (1981)

Source: Wikipedia: Fair use, https://en.wikipedia.org/w/index.php?curid=1271607

Stories behind the Songs





Credits: www.randybachman.com, randybachman.rockpapermerch.com, www.amazon.com, Mark Knopfler (of Dire Straits): Ckuhl from nl [GFDL (http://www.gnu.org/copyleft/fdl.html) or CC-BY-SA-3.0 (http://creativecommons.org/licenses/by-sa/3.0/)], via Wikimedia Commons.

Case Study (cont.): Music Ratings, Popularity

- A radio station plays music.
- In order to stay in business and have a sufficient listening audience, the station might utilize the top *n* songs (e.g., most popular, “best”, highest sales, etc.) as reported by Billboard or other rating agencies.
- It is important to the disk jockeys at a radio station that they stay informed about each band’s members, the song’s meaning (if any), the story behind the song (how would you model this?), the band’s performances (e.g., dates, locations, revenue), etc.
- This begs the question: Should the radio station maintain this type of database?

Best-Selling Albums in the US

Over 20 million copies

Year	Artist	Album	Label	Shipments (Sales)
1982	Michael Jackson	*Thriller*	Epic	(27,700,000‡)[3]
1971	Led Zeppelin	*Led Zeppelin IV*	Atlantic	23,000,000
1973	Pink Floyd	*The Dark Side of the Moon*	Harvest/Capitol	(20,502,000‡)[4]

Over 15 million copies

Year	Artist	Album	Label	Shipments (Sales)
1976	Eagles	*Their Greatest Hits (1971–1975)*	Asylum	(18,593,000‡)[5][6]
1980	AC/DC	*Back in Black*	Atlantic	(18,430,000‡)[6][7]
1997	Shania Twain	*Come On Over*	Mercury Nashville	(17,661,500)[6][8]
1977	Fleetwood Mac	*Rumours*	Warner Bros.	(17,258,464‡)[6][9]
1976	Boston	*Boston*	Epic Records	17,000,000
1974	Elton John	*Greatest Hits*	Polydor	17,000,000
1991	Metallica	*Metallica*	Elektra	(16,400,000)[10]
1976	Eagles	*Hotel California*	Asylum	16,000,000
1995	Alanis Morissette	*Jagged Little Pill*	Maverick	(15,350,000)[6][11]
1977	Soundtrack / Bee Gees	*Saturday Night Fever*	RSO	15,000,000

Over 10 million copies

Year	Artist	Album	Label	Shipments (Sales)
1994	Hootie & the Blowfish	*Cracked Rear View*	Atlantic	(14,580,000)[6][12][13]
1987	Guns N' Roses	*Appetite for Destruction*	Geffen	(14,490,000‡)[6][14]
1978	Soundtrack / John Travolta / Olivia Newton-John / Various artists	*Grease*	RSO	(14,000,000‡)[15]
1969	The Beatles	*Abbey Road*	Apple	(14,000,000‡)[16]
1972	Simon & Garfunkel	*Simon and Garfunkel's Greatest Hits*	Columbia	14,000,000
1999	Backstreet Boys	*Millennium*	Jive	(13,890,000)[6][17]
1984	Bob Marley & The Wailers	*Legend*	Island	(13,660,000‡)[18]
1984	Bruce Springsteen	*Born in the U.S.A.*	Columbia	(13,463,000‡)[19]
1992	Soundtrack / Whitney Houston	*The Bodyguard*	Arista	(13,450,000)[6][20]
1979	Pink Floyd	*The Wall*	Columbia/Harvest/EMI	(13,381,000‡)[21]
1987	Soundtrack	*Dirty Dancing*	RCA	(13,232,000‡)[6][22]
1999	Santana	*Supernatural*	Arista	(13,110,000)[6][20]
1984	Prince and the Revolution	*Purple Rain*	Warner Bros.	(13,107,000‡)[23]
2000	The Beatles	*1*	Apple/EMI/Capitol	(12,800,000)[24]
2000	NSYNC	*No Strings Attached*	Jive	(12,680,000)[6][20]

Source: https://en.wikipedia.org/wiki/List_of_best-selling_albums_in_the_United_States

Exercise 2: Registrar's Database

- Design a registrar's database to store information about students, courses, the courses students have taken, and the grades students have gotten in these courses. Some relevant details are: Courses have a number, a department, and a title. For example, "CPSC 304: Introduction to Relational Databases" has department = CPSC, number = 304, and title = "Introduction to Relational Databases".
- Numbers are assigned by departments, and different departments may use the same number.
- Students are represented by their (unique) student ID and their name.
- "Enrolment" consists of a course, a student who took that course, and the grade the student got in the course.
- You should draw one or more ER diagrams that represent this database structure correctly.

Which of the Following Might You Find in a Correct ER Diagram?

A. Entity set Student with attributes ID (not underlined) and Name (underlined).

B. Entity set Student with attributes ID and Name (both underlined).

C. Entity set Course with attributes Department and Number (both underlined) and Title (not underlined).

D. Entity set Course with attribute Department (underlined) and attributes Number and Title (both not underlined).

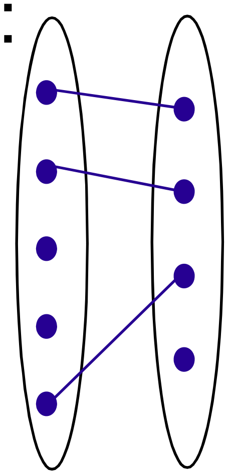
E. Two of the above

Cardinalities

- The **cardinality ratio** for a relationship set specifies the number of relationships in the set that an entity can participate in.

Let R be a relationship set between sets A and B. R can have 1 of 4 cardinalities:

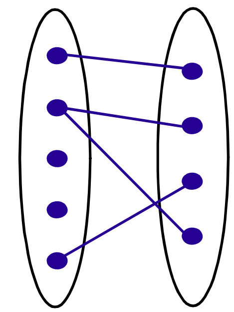
1. ***One-to-One***
   - An entity in A is associated with at most one entity in B, and vice versa.
   - e.g., A: driver, B: driver's license



1-to-1

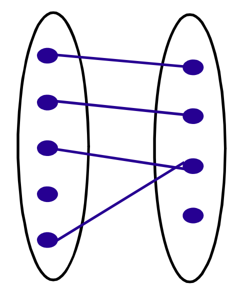
Cardinalities (cont.)

2. ***One-to-Many***
   - An entity in A is associated with any number of entities (possibly zero) in B.
   - An entity in B is associated with at most one entity in A.
   - e.g., A: biological-mother, B: children



1-to-Many

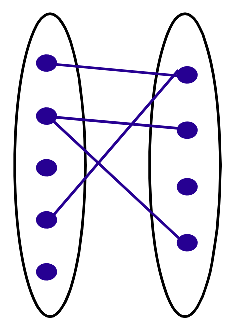
3. ***Many-to-One*** *from A to B:*
   - e.g., Switch A and B, above.



Many-to-1

Cardinalities (cont.)

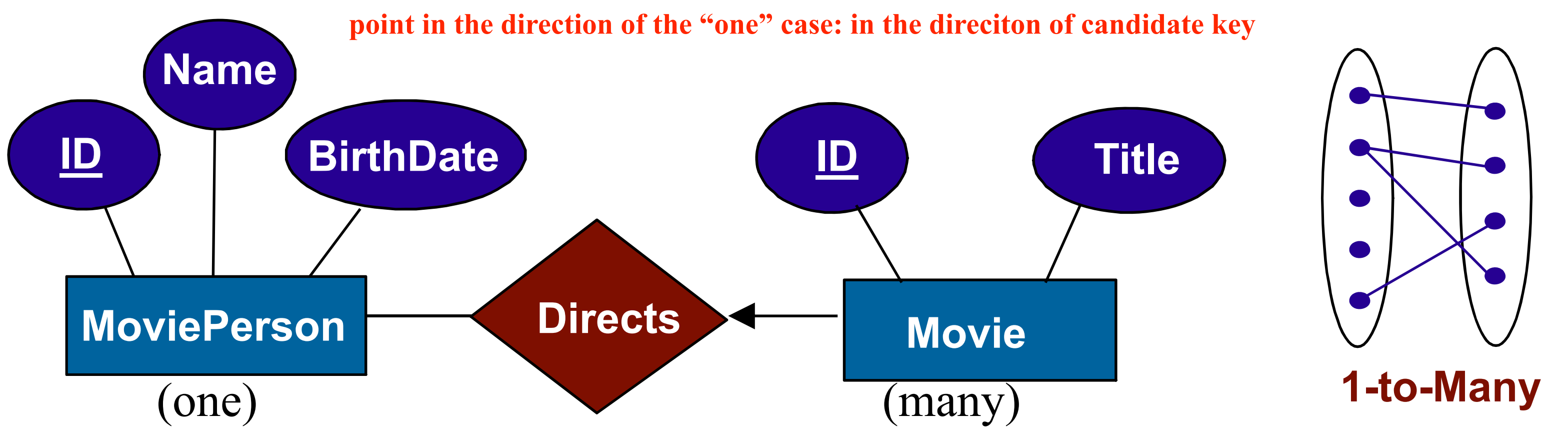
4. ***Many-to-Many***:
   - *An entity in A is associated with any number of entities in B, and vice versa.*
   - *e.g., A: students, B: courses*



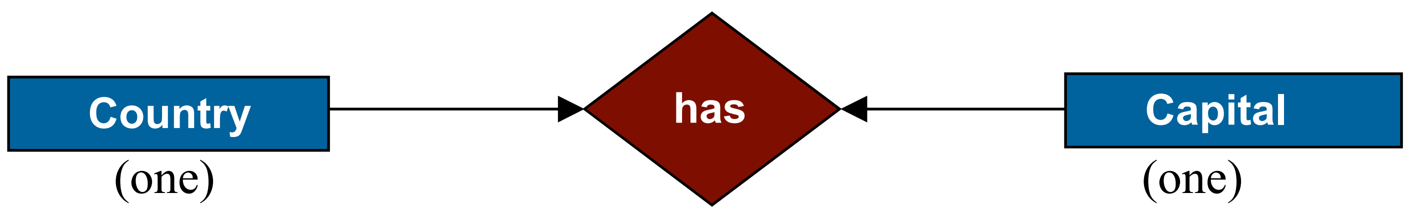
Many-to-Many

Key Constraints

- The restriction imposed by 1-to-1 and 1-to-many ratios are examples of ***key constraints***.
- A key constraint is shown with an arrow in the ER diagram.
- This is especially important for insertions.

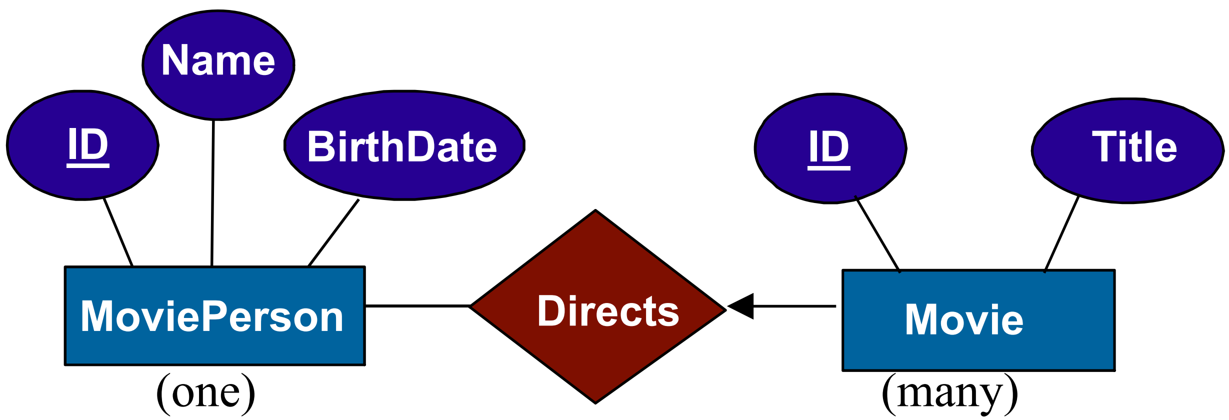


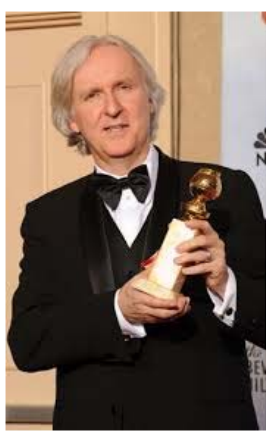
or

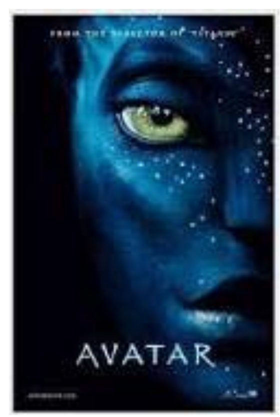


A Brief Digression on Notation

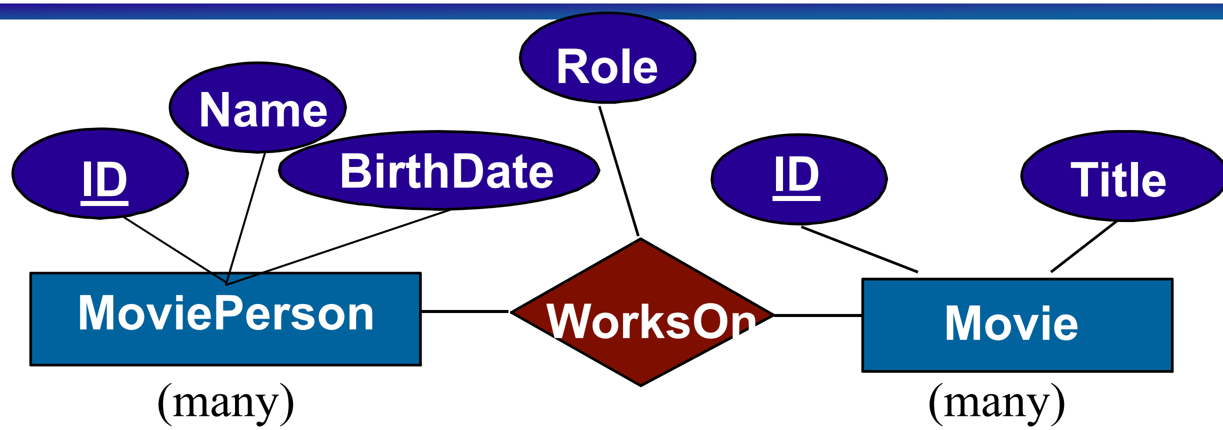
- The arrow's direction? Point the arrow towards the entity that there is only one of.



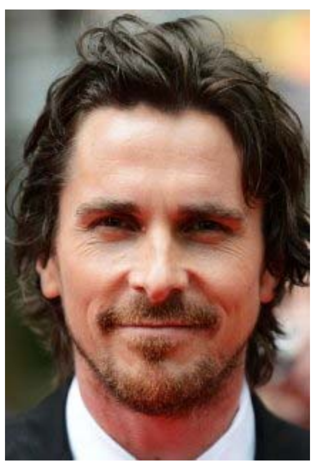




How Can We Uniquely Identify a Relationship?



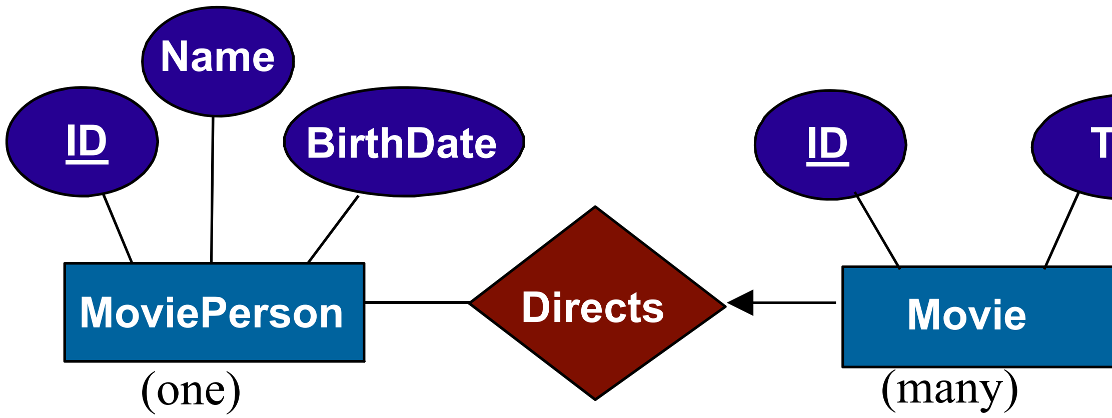
- How can we identify the role of a specific MoviePerson in a specific Movie?



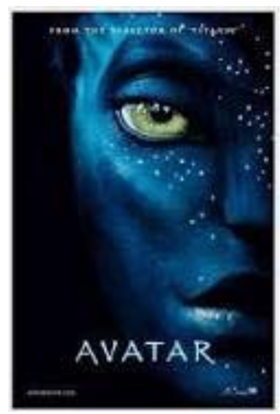
Christian Bale
as
Bruce Wayne

Can the same person have multiple roles?

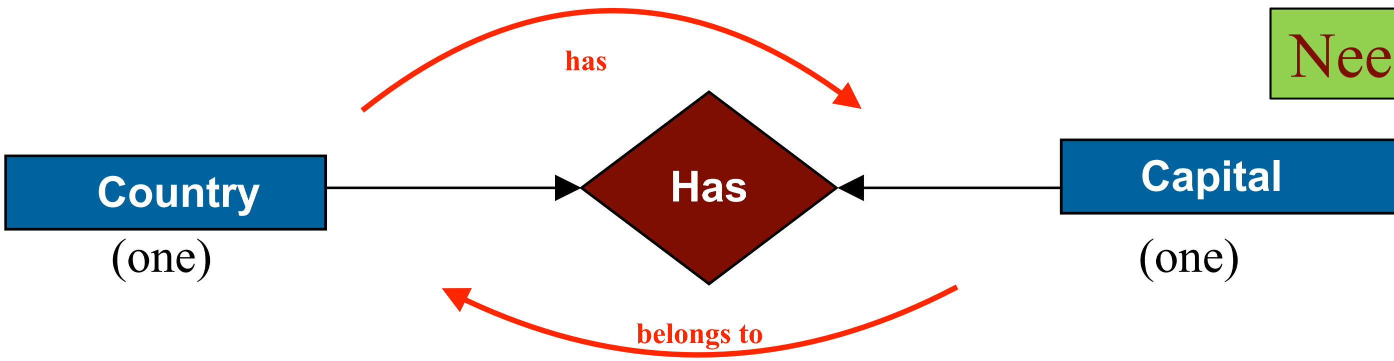
How Can We Uniquely Identify a Relationship? Contrast with …



Needs no “Role”



How Can We Uniquely Identify a Relationship?



Need either one

Ottawa

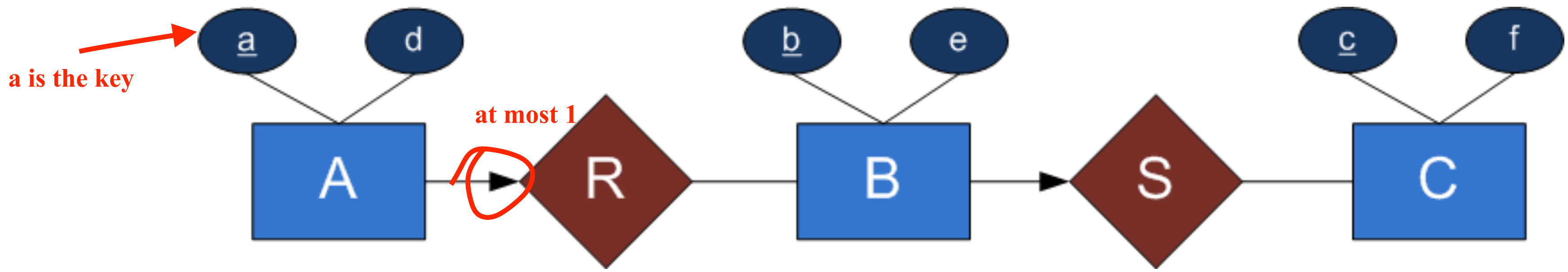
Summary: Primary Keys of Relationship Sets

- Let R be a relationship set between sets A and B. What is R's primary key?

TYPE OF R	PRIMARY KEY OF R
One-to-one	Use primary key of A <u>or</u> primary key of B either suffices without ambiguity
One-to-many from A to B	Use primary key of B to uniquely identify
Many-to-many	Use primary key of A <u>and</u> primary key of B

- Note: R may have its own key in addition to the key it inherits from the entities.

rel-name = {(e1, f1), (e2,f2)} means that a relationship between e1 and f1 exists in the relationship set for rel-name, as does a relationship between e2 and f2.



Suppose that a1 and a2 are the only entities of A, b1 and b2 are the only entities of B, and c1 and c2 are the only entities of C.

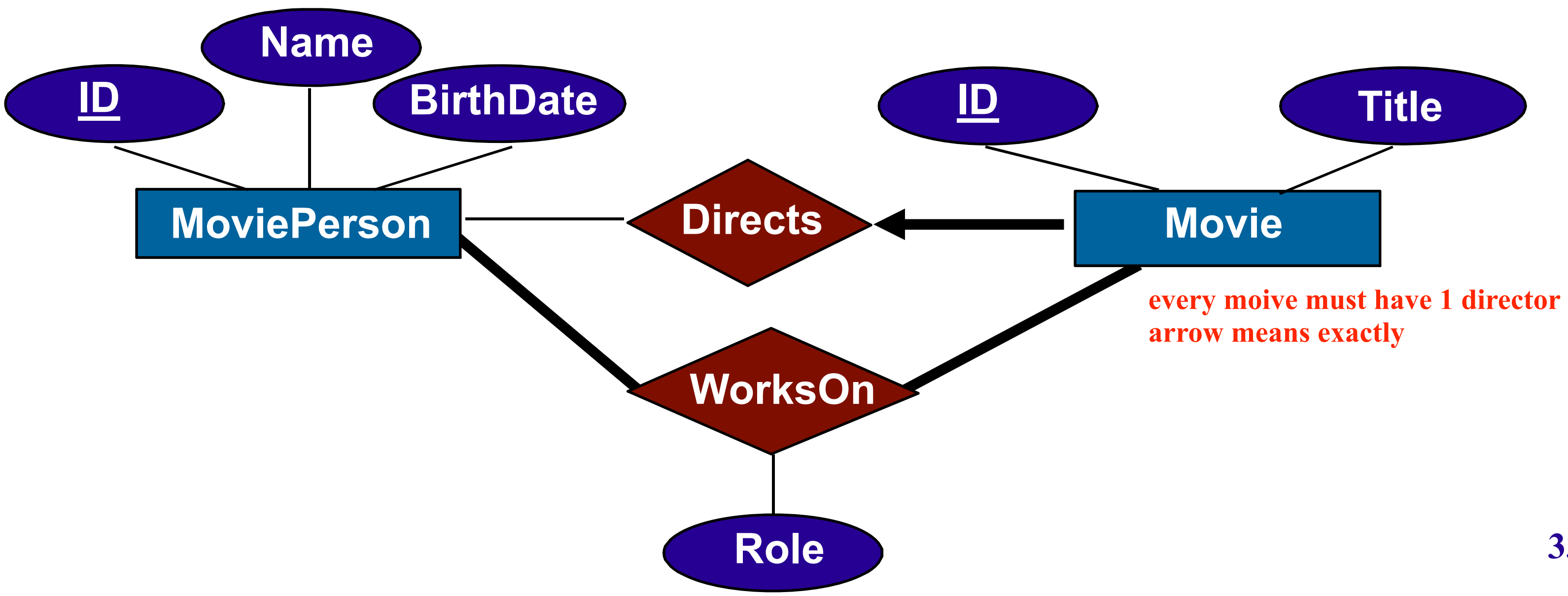
Which of the following relationship sets for R and S are possible according to the diagram, where T = {(e1,f1)} means that a relationship between e1 and f1 exists in relationship set T?

A. R = {}; S = {(b2, c1), (b2, c2)} b2 can only have at most 1
B. R = {}; S = {(b1, c2), (b2, c2)}
C. R = {(a2, b2)}; S = {(b2, c1), (b2, c2)}
D. R = {(a1, b2), (a2, b1), (a2, b2)}; S = {}
E. None of the above

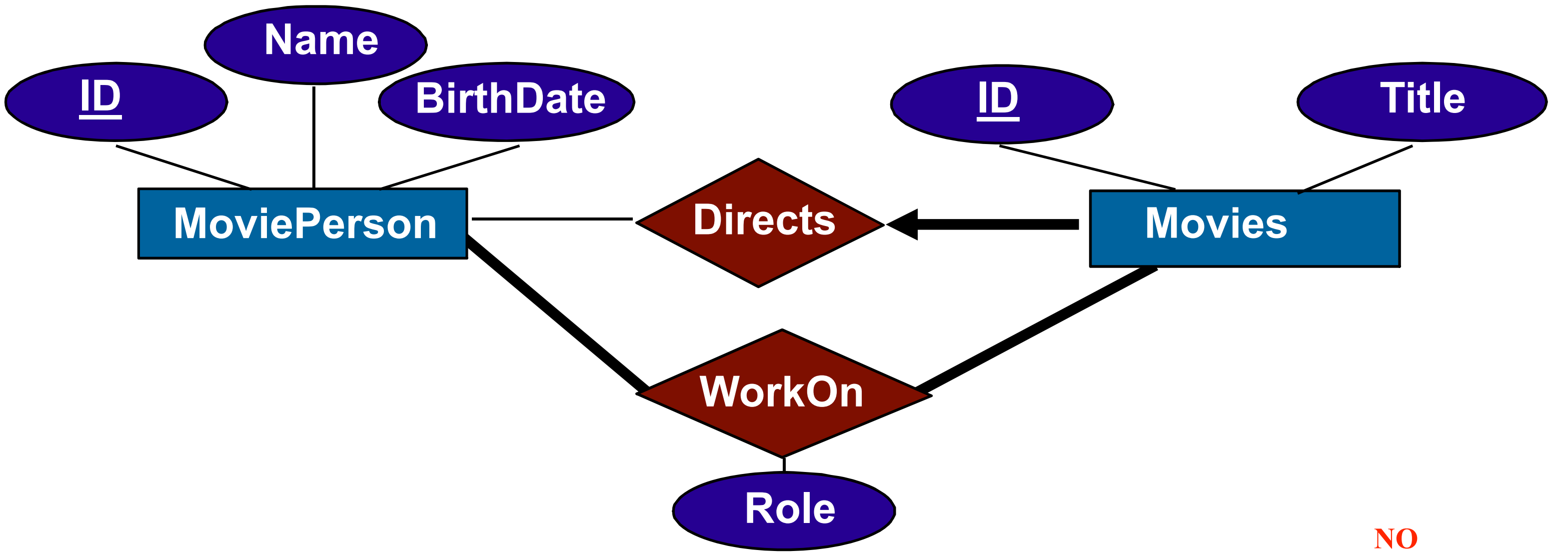
thick

Participation Constraints

- Participation: Indicates if the other entity must participate in the relationship.
- An entity's participation can be ***total*** or ***partial***.
- Requiring total participation is a ***participation constraint*** and it is shown with a thick line.
  - Important for deletions
  - e.g., participation of Movie in Directs is total (thick line) which means:
    - Every movie must appear in some relationship in the Directs set.

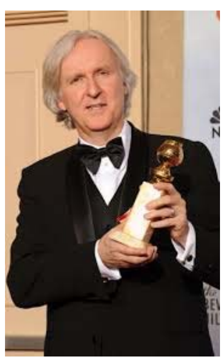


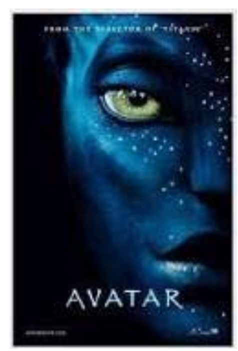
Why is the Participation Constraint Important?



NO

Would I be able to delete James Cameron without deleting Avatar?
Would I be able to delete Avatar without deleting James Cameron?



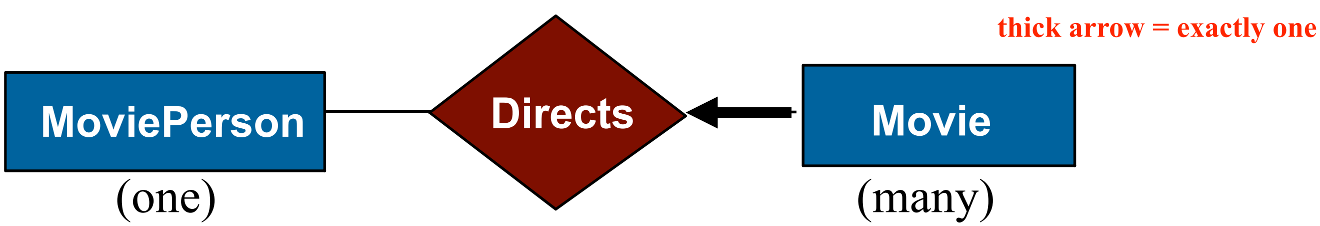


Yes, unless Avatar was his only job

Line Types Summarized

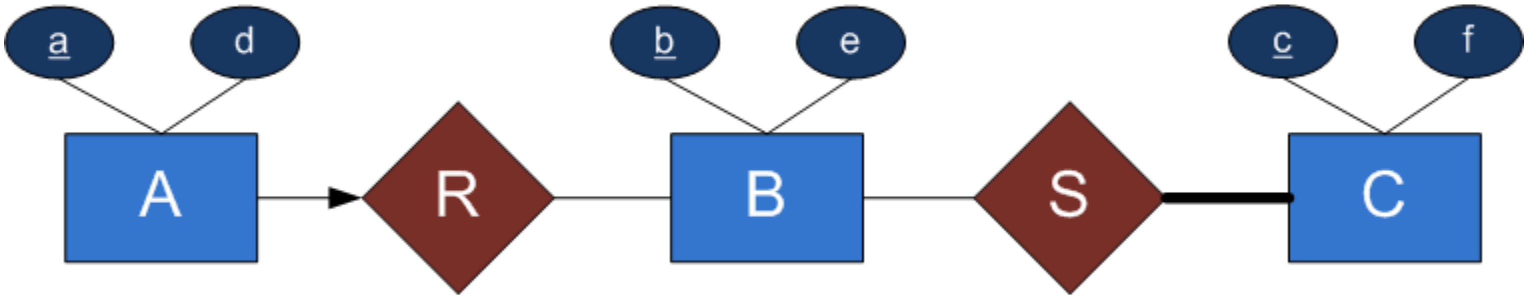
- Plain lines mean many-to-many:

- Arrows mean the other side has a cardinality of one: thin arrow = at most one



- A thick line requires total participation and can be added to any line—arrow or not.

rel-name = {(e1, f1), (e2,f2)} means that a relationship between e1 and f1 exists in the relationship set for rel-name, as does a relationship between e2 and f2.



Suppose that a1 and a2 are the only entities of A, b1 and b2 are the only entities of B, and c1 and c2 are the only entities of C.

Which of the following relationship sets for R and S are possible according to the diagram, where T = {(e1,f1)} means a relationship between e1 and f1 exists in relationship set T?

A. R = {}; S = {}
B. R = {(a1,b1)}, S = {(b2,c2)} no c1
C. R = {(a1,b1), (a1,b2)}; S = {(b1,c1), (b2,c2)}
D. R = {(a1, b2)}; S = {(b1,c2), (b2, c1), (b1,c1)}
E. None of the above