Database Systems

Database Design

H. Turgut Uyar Şule Öğüdücü

2002-2012

License

Topics

Functional Dependency

Definition

- Z: the set of all attributes of the relation R
- $A, B \subseteq Z$
- A functionally determines B: $A \rightarrow B$
  for every A value there can only be one B value
- every functional dependency is an integrity constraint

Sample Relation

Example

R

MOVIEID	TITLE	COU	LANG	ACTORID	NAME	ORD
6	Usual Suspects	UK	EN	308	Gabriel Byrne	2
228	Ed Wood	US	EN	26	Johnny Depp	1
70	Being John Malkovich	US	EN	282	Cameron Diaz	2
1512	Suspiria	IT	IT	745	Udo Kier	9
70	Being John Malkovich	US	EN	503	John Malkovich	14

- assumption: the language of the movie
  is the language of the country where it was made

Functional Dependency Examples

Example

- MOVIEID → COUNTRY
- ACTORID → NAME
- MOVIEID → {TITLE, COUNTRY, LANGUAGE}
- {MOVIEID, ACTORID} → COUNTRY
- {MOVIEID, ACTORID} → MOVIEID
- {MOVIEID, ACTORID} → ORD
- {MOVIEID, ACTORID} → {COUNTRY, ORD}
- COUNTRY → LANGUAGE

Irreducible Set

- S: the set of all FDs of the relation
- $T \subseteq S$, such that
  - T contains as few elements as possible
  - every FD in S can be derived from the FDs in T
- let there be only one attribute on the right hand side of FDs

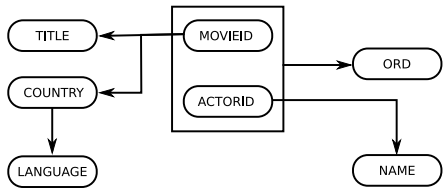
Irreducible Set Example

Example

- MOVIEID → TITLE
- MOVIEID → COUNTRY
- COUNTRY → LANGUAGE
- ACTORID → NAME
- {MOVIEID, ACTORID} → ORD

Dependency Diagram

Example



Normal Forms

- 1NF, 2NF, 3NF, BCNF, 4NF, 5NF
- every form narrows down the scope of the previous form
  - every relation in 2NF is also in 1NF
  - every relation in 3NF is also in 2NF, ...
- 1NF: attribute values are atomic

Normalization

Definition

normalization:
transition from one form to the next, narrower form

- transition between forms must be lossless

Theorem (Heath)

- *Z: the set of all attributes of the relation R*
- *$A, B, C \subseteq Z$*
- *if $A \rightarrow B$, then R can be obtained by joining the relations $\{A, B\}$ and $\{A, C\}$*

Lossless Transition Example

Example

R1

MOVIEID	TITLE	COU	LANG
6	Usual Suspects	UK	EN
228	Ed Wood	US	EN
70	Being John Malkovich	US	EN
1512	Suspiria	IT	IT

R2

MOVIEID	ACTORID	NAME	ORD
6	308	Gabriel Byrne	2
228	26	Johnny Depp	1
70	282	Cameron Diaz	2
1512	745	Udo Kier	9
70	503	John Malkovich	14

- R = R1 JOIN R2

Lossy Transition Example

Example

R1

MOVIEID	TITLE	COU	LANG
6	Usual Suspects	UK	EN
228	Ed Wood	US	EN
70	Being John Malkovich	US	EN
1512	Suspiria	IT	IT

R2

COU	ACTORID	NAME	ORD
UK	308	Gabriel Byrne	2
US	26	Johnny Depp	1
US	282	Cameron Diaz	2
IT	745	Udo Kier	9
US	503	John Malkovich	14

- R $\neq$ R1 JOIN R2
- {MOVIEID, ACTORID} $\rightarrow$ ORD

Anomalies

- *insert*
  - data is known but can not be inserted due to constraints
- *delete*
  - deleting some data causes some other data to be lost
- *update*
  - updating some data requires modifications in multiple tuples

Anomaly Examples

Example

- the country of the movie "Gattaca" is known to be US, but this cannot be inserted if there is no actor in the movie
- deleting that Gabriel Byrne acts in the movie "Usual Suspects" also deletes that the movie was made in the UK
- changing the country of the movie "Being John Malkovich" requires modifications in two tuples

2nd Normal Form

Definition

2NF: every non-key attribute depends on the primary key

transition from 1NF to 2NF

- in a relation R that conforms to 1NF:
  - $R(A, B, C, D)$, primary key: $\{A, B\}$
  - $A \rightarrow D$
- to be 2NF:
  - $R1(A, D)$, primary key: A
  - $R2(A, B, C)$, primary key: $\{A, B\}$
    A is a foreign key referencing $R1$

1NF-2NF Transition Example

Example

- among the non-key attributes, only ORD depends on the primary key
  - A: MOVIEID
  - B: ACTORID
  - C: {NAME, ORD}
  - D: {TITLE, COUNTRY, LANGUAGE}

1NF-2NF Transition Example

Example

- R1(MOVIEID, TITLE, COUNTRY, LANGUAGE)
  primary key: MOVIEID
- R2(MOVIEID, ACTORID, NAME, ORD)
  primary key: {MOVIEID, ACTORID}
  MOVIEID is a foreign key referencing R1

1NF-2NF Transition Example

Example

- R2 still not 2NF: ACTORID → NAME
  - *A*: ACTORID
  - *B*: MOVIEID
  - *C*: ORD
  - *D*: NAME
- R3(ACTORID, NAME)
  primary key: ACTORID
- R4(MOVIEID, ACTORID, ORD)
  primary key: {MOVIEID, ACTORID}
  ACTORID is a foreign key referencing R3

2NF Relation Examples

Example

R1

MOVIEID	TITLE	COU	LANG
6	Usual Suspects	UK	EN
228	Ed Wood	US	EN
70	Being John Malkovich	US	EN
1512	Suspiria	IT	IT

R3

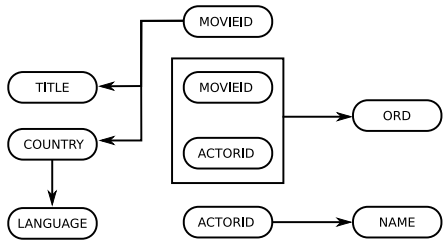
ACTORID	NAME
308	Gabriel Byrne
26	Johnny Depp
282	Cameron Diaz
745	Udo Kier
503	John Malkovich

R4

MOVIEID	ACTORID	ORD
6	308	2
228	26	1
70	282	2
1512	745	9
70	503	14

Dependency Diagram Example

Example



2NF Corrected Anomalies

Example

- if the country of the movie "Gattaca" is known to be US, this can be inserted to R1
- if Gabriel Byrne is deleted from the movie "Usual Suspects", the country of the movie is still kept in R1
- changing the country of the movie "Being John Malkovich" requires updating only one tuple in R1

2NF Remaining Anomalies

Example

- it is known that movies made in Brazil are in Portuguese but this can not be inserted if there is no movie made in Brazil
- deleting the movie "Suspiria" also deletes that the language of movies made in Italy is Italian
- changing the language of the movies made in the US requires two tuples to be updated

3rd Normal Form

Definition

3NF: non-key attributes do not depend on any attributes other than the primary key

transition from 2NF to 3NF

- in a relation R that conforms to 2NF:
  - $R(A, B, C, D)$, primary key: A
  - $C \rightarrow D$
- for it to be 3NF:
  - $R1(C, D)$, primary key: C
  - $R2(A, B, C)$, primary key: A
    C is a foreign key referencing $R1$

2NF-3NF Transition Example

Example

- R1: COUNTRY → LANGUAGE
  - *A*: MOVIEID
  - *B*: TITLE
  - *C*: COUNTRY
  - *D*: LANGUAGE
- R5(COUNTRY, LANGUAGE)
  primary key: COUNTRY
- R6(MOVIEID, TITLE, COUNTRY)
  primary key: MOVIEID
  COUNTRY is a foreign key referencing R5

3NF Relation Examples

Example

R6

MOVIEID	TITLE	COU
6	Usual Suspects	UK
228	Ed Wood	US
70	Being John Malkovich	US
1512	Suspiria	IT

R5

COU	LANG
UK	EN
US	EN
IT	IT

R3

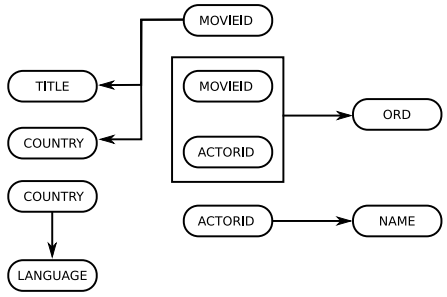
ACTORID	NAME
308	Gabriel Byrne
26	Johnny Depp
282	Cameron Diaz
745	Udo Kier
503	John Malkovich

R4

MOVIEID	ACTORID	ORD
6	308	2
228	26	1
70	282	2
1512	745	9
70	503	14

Dependency Diagram Example

Example



3NF Corrected Anomalies

Example

- if movies made in Brazil are in known to be in Portuguese, this can be inserted into R5
- if the movie "Suspiria" is deleted, R5 still keeps that movies made in Italy are in Italian
- changing the language of the movies made in the US requires modifying only one tuple in R5

Boyce-Codd Normal Form

Definition

BCNF: all functional dependencies must be on candidate keys

- dependencies between attributes that are part of the keys have to be considered

BCNF Example

Example (let movie titles be unique)

- candidate keys:
  - {MOVIEID, ACTORID}
  - {TITLE, ACTORID}
- non-conforming functional dependencies:
  - MOVIEID → TITLE
  - TITLE → MOVIEID

References

Required Reading: Date

- Chapter 11: Functional Dependencies
- Chapter 12: Further Normalization I: 1NF, 2NF, 3NF, BCNF

Entity/Relationship Model

- modeling approach
  - Chen 1976
- components
  - entities
  - properties
  - relationships

Entities

Definition

entity: set of "things" with the same attributes

- elements of the set are *instances* of the entity
- *strong*: can exist by itself
- *weak*: existence depends on another entity

Entity Examples

Example

- entity: movie, person
- person instance: Johnny Depp
- strong entity: person
- weak entity: movie

Properties

Definition

property: data describing entities or relationships

- simple / composite
- key
- single / multiple valued
- empty
- base / derived

Property Examples

Example

- property: title, country, language
- simple: first name, last name
- composite: full name
- base: date of birth
- derived: age

Relationships

Definition

relationship: connections between entities

- *participant*: entities in the relationship
- *degree*: number of participants
- *total* / *partial*: all instances of the entity do / don't participate in the relationship

Relationship Types

- *one-to-one*
- *one-to-many*
- *many-to-many*

Relationship Examples

Example (one-to-one)

- the capital relationship between countries and cities

Example (one-to-many)

- the management relationship between employees and projects

Example (many-to-many)

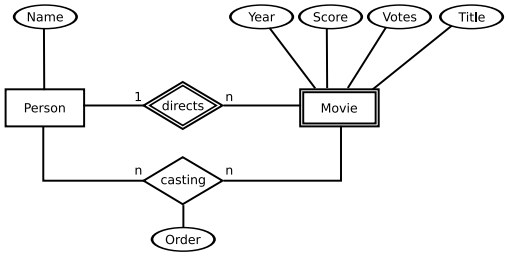
- the enrollment relationship between students and courses

Entity/Relationship Diagrams

- entity: rectangle
  - weak: double lines
- property: ellipsis
  - derived: dashed lines
  - multi-valued: double lines
  - composite: sub-ellipses
- relationship: diamond
  - between weak and strong: double lines
  - total: connection double lines
  - 1 or n depending on the type of the relationship

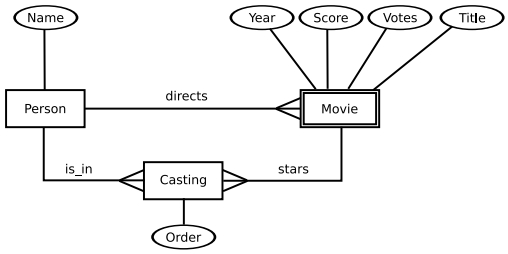
Entity/Relationship Diagram Example

Example



Entity/Relationship Diagram Example

Example



Applying to Design

- every entity a relation
- every property an attribute
- every many-to-many relationship a relation
  - foreign keys to participating entities
- for every one-to-many relationship a foreign from the "many" side to the "one" side

References

Required Reading: Date

- Chapter 14: Semantic Modeling