

729G28 Webprogrammering och databaser

Fö 3: Relationsmodellen

definitioner

ER-modell -> relationsmodell

nycklar, olika varianter

Relationsmodellen

- Introducerades av Edward Codd 1970
- Mycket vanlig
- Stödjer kraftfulla och ändå enkla deklarativa språk
- Matematisk grund i relationsalgebra

Relation - en typ av funktion

- t.ex: $y=x^2$

y	x
1	1
4	2
9	3
1	-1

Relation - en generalisering av funktion

- godtyckligt antal kolumner (domäner)
- varje rad representerar värden som hör ihop
- Relationsalgebra - räkna med relationer

Relationer

Relationen har **attribut** och är en mängd av **tupler**

Kolumn (Attribut) Attributvärde

Student

Rad (Tupel)

Pnr	Namn	epost
801212	Anna B	annbe
850101	Oskar A	oskan

Relationer – som tabeller

- Tabellen består av en **mängd** rader!
 - Rader får ej ha dubletter
 - Raderna betraktas som osorterade
- attributvärdena är **atomära** (odelbara).
- Notera att ER-modellens attribut hade en värdemängd (domän) på samma sätt som relationsmodellens attribut (kolumner).

Relationsschema

relationsnamn (attributnamn1, ... attributnamnN)

- exempel:

Student (namn, gata, husNr, postNr, liuID, program)

Alternativa notationer:

Student

namn	gata	husNr	postNr	liuID	program
------	------	-------	--------	-------	---------

Student

namn

gata

husNr

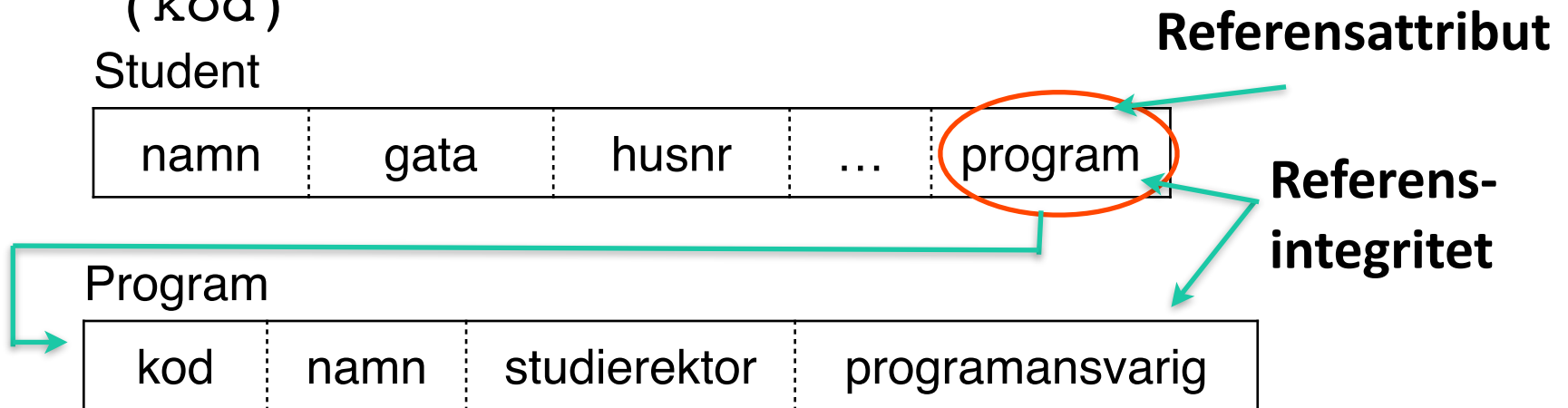
postNr

liuID

program

Flera relationer

- En databas består av flera relationer med kopplingar mellan sig.
- Ex: studenter och program:
- Foreign key program references Program (kod)



Referensintegritet

- Delete from Program where kod=f7kko
- Delete from Program where kod=f7ksta

ERROR

OK!

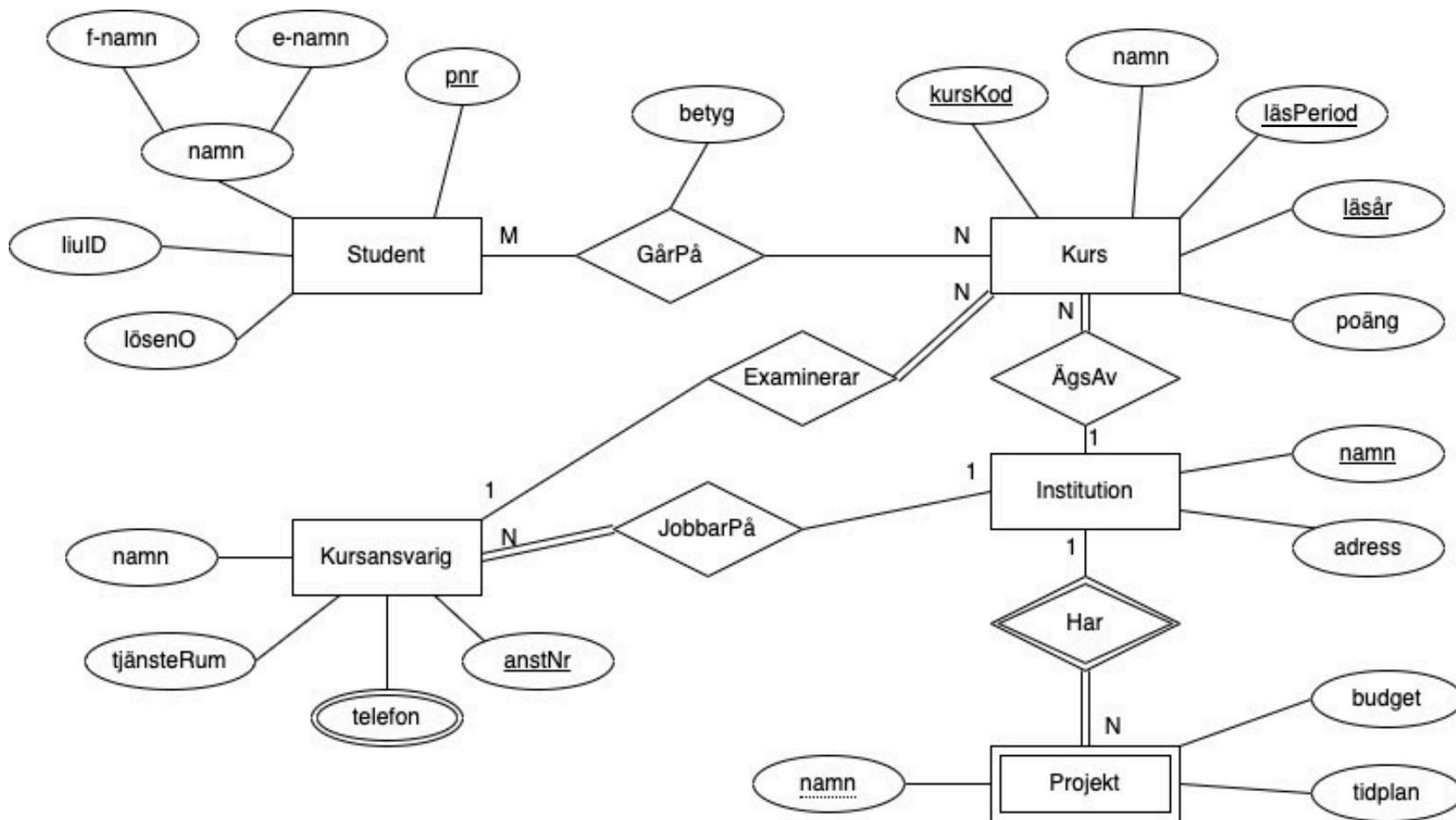
Student

Namn	gata	...	liuID	program
Pelle	Bygat 2	...	pelan123	f7kko
Anna	Rydsv 3	...	annax123	6DDD

Program

kod	namn	Studierektor	Programansv
f7kko	Kognitionsv	Jalal Maleki	Erik Prytz
f7ksta	Statistik o da	Linda H	Kalle

Universitetet: ER-diagram



Från ER till Relationsmodell (Kokbok+)

1. För varje entitetstyp: definiera motsvarande relationsschema. Varje tupel i relationen kommer att representera en entitetsinstans
 - Vanliga attribut blir attribut (kolumner) (1)
 - Sammansatta attribut representeras av delarna (8)
 - Flervärda attribut blir en egen relation med nyckeln till entitetstypen och attributet enkelt (9)
 - Svag entitetstyp får som extra attribut nyckeln till den ägande entitetstypen (7)

Från ER till Relationsmodell

2. För varje sambandstyp:

- Binära samband

1:N (N:1) - lägg in 1-entitetens nyckel i relationen för N-entiteten (2)

1:1 - lägg in nyckeln till den ena entiteten i den andras relation (3)

N:M - blir en egen relation bestående av nycklarna till båda entiteterna (4)

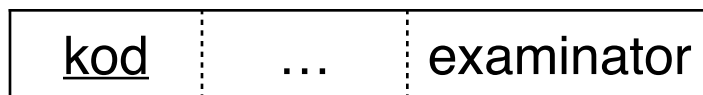
- Flervägssamband blir egen relation (pss som N:M) (5)
- Ägandesamband ignoreras
- Attribut på samband läggs in i respektive relation (6)

Obs: Sambandstyper - referensattribut

Entitetstypers nycklar som läggs in i andra relationer än den som representerar den egna entitetstypen blir referensattribut!

- t.ex. Examinerar: examinator refererar Kursansvarig
- Gäller även svag entitetstyp (ägande e-typen)

Kurs



Kursansvarig



Från ER till Relationsmodell

3. Markera nycklar

- ER-modellens nyckelattribut?
- Nya relationer?
- Alternativ?

Relationsmodellens nycklar, definition

Om något eller några av attributen i en relation kan användas för att identifiera hela tupeln (raden) så är det/de attributen en **nyckel**

En **nyckel** är alltså en delmängd av attributen i en relation.

Typer: Supernyckel, kandidatnyckel, primärnyckel

Supernyckel, formellt:

Om k är en delmängd av attributen i en relation R sådan att k kan användas för att identifiera raderna i R så är k en **supernyckel** för R .

Kandidatnyckel = minimal supernyckel

En supernyckel k är minimal om vi inte kan ta bort något attribut ur k så att den nya k , k' , fortfarande är en supernyckel. k är då en kandidatnyckel.

Generellt finns det flera kandidatnycklar för en relation.

De attribut som ingår i någon kandidatnyckel kallas **nyckel-attribut**.

Primärnyckel

- Den kandidatnyckel som **väljs** av databasdesignern som huvudnyckel för en relationstabell R kallas primärnyckel eller **nyckeln** till R.
- Används i andra relationer för att referera till en viss rad i R.
- Primärnyckeln markeras i relationsschemat med understrykning.

Från ER till Relationsmodellen : 3.Nycklar

Relationer baserade på:

- vanlig entitetstyp
- svag entitetstyp
- flervärt attribut

Får som primärnyckel:

- nyckeln ur ER-diagrammet.
- nyckeln till den ägande entitetstypen **plus** den partiella nyckeln.
- nyckeln till entitetstypen **plus** attributet självt.

Från ER till Relationsmodellen : nycklar, forts

Relationer baserade
på sambandstyp:

- Binär M:N
- flervägssambandstyp

Får som primärnyckel:

- båda entitetstypernas
nycklar
- nycklarna till de
sammanbundna
entitetstyper som kan
ingå i flera
sambandsinstanser

Extra steg till kokboken

4. Kontrollera redundans, förenkla vid behov:
Normalisera (jfr föreläsning om detta).
5. Specificera integritetsvillkoren, t.ex:
 - Totalt deltagande: Not Null
 - Referensattribut = främmande nycklar
Markeras med pilar från referensattribut till refererad relation
 - semantisk integritet

Universitetet, resultat

Student

<u>pnr</u>	e-namn	f-namn	lösen	liuID
------------	--------	--------	-------	-------

Kurs

namn	<u>kurskod</u>	<u>läsår</u>	<u>period</u>	poäng	examinator	inst
------	----------------	--------------	---------------	-------	------------	------

Anställd

namn	<u>anstnr</u>	rum	inst
------	---------------	-----	------

Telefon

<u>anstnr</u>	<u>telnr</u>
---------------	--------------

StudGårKurs

<u>student</u>	<u>kurskod</u>	<u>läsår</u>	<u>period</u>	betyg
----------------	----------------	--------------	---------------	-------

Institution

<u>namn</u>	adress
-------------	--------

Projekt

<u>inst</u>	<u>namn</u>	tidsplan	budget
-------------	-------------	----------	--------

Integritetsvillkor (deltagande):

Kurs: kursansv, inst Not Null

Anställd: inst Not Null

Kokbok+

1. Entitetstyper -> relationer (kom ihåg alla attribut)
2. Sambandstyper in i relationer eller blir egna relationer
3. Nycklar
4. Redundanskontroll, normalisering
5. Integritetsvillkor

Fysisk datamodell: SQL

- Skapa tabellerna (`create table`)
 - Attributens domäner (`integer`, `longint`, `char`, `varchar`, `boolean`, `date`, ...)
 - Integritetsvillkor:
 - Primärnyckel: `primary key`
 - totalt deltagande: `not null`
 - Främmande nycklar: `foreign key (attribut) references måltabell (primärnyckel)`
 - Index, filorganisation... beror av användning och databashanteraren.
-

Frågor?

Nästa tillfälle: Lektion om ER- och
relationsmodeller

www.liu.se