

729G28: Webprogrammering och databaser

Fö 4: Normalisering

Termer ur relationsteori

Funktionella beroenden

Normalformer (1NF, 2NF, 3NF och BCNF)

Informationsbevarande

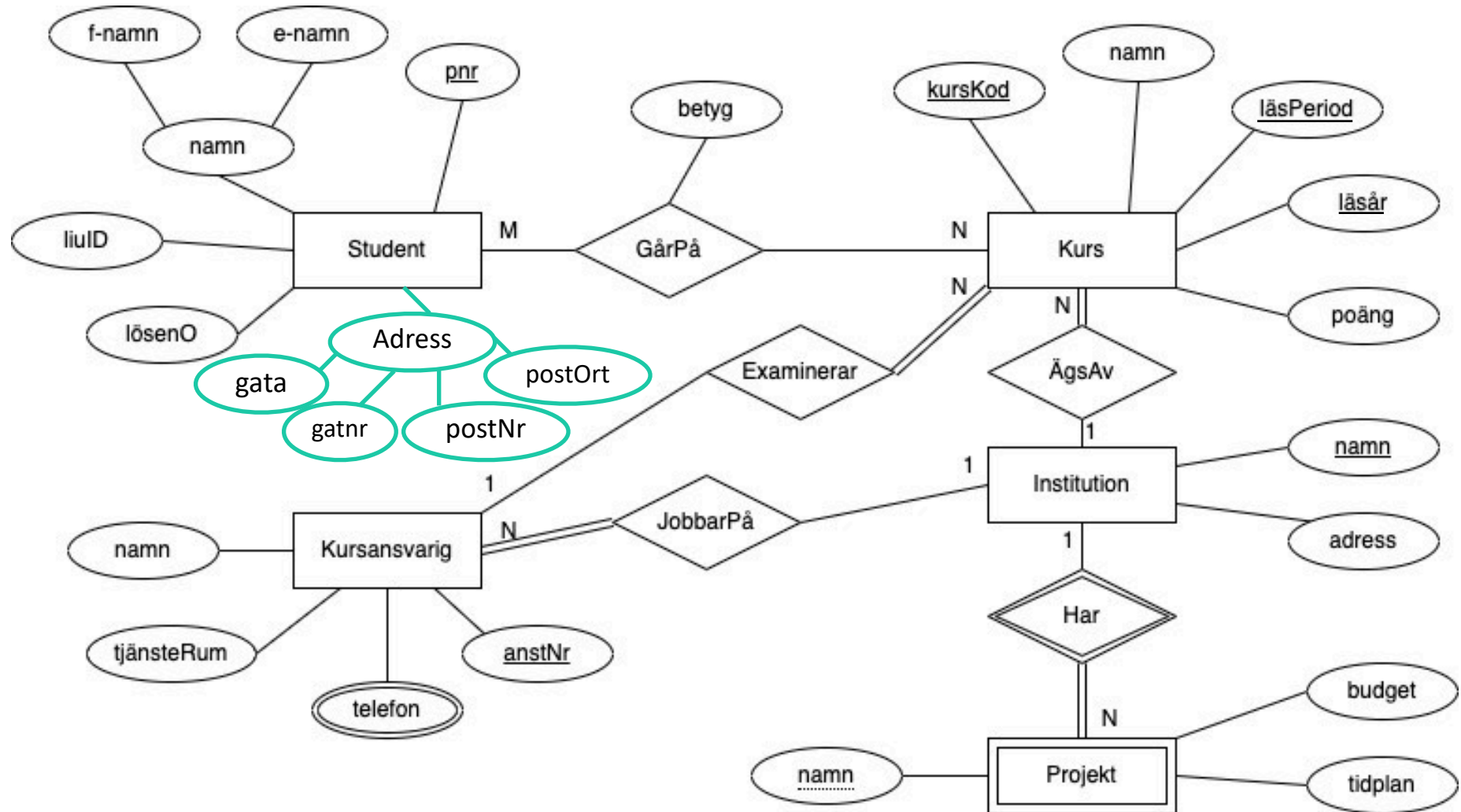
relationsschemauppdelning

HT2023

Varför normalisera?

- Metod att skydda oss från ”dum” design
 - Lagra samma data flera ggr i onödan
 - Inte kunna lagra viss information
 - Otydlig betydelse av en tupel/rad – som kan komma från otydlig betydelse av entitetsinstans

Universitetet: ER-diagram



Universitetet, med adress):

Student

<u>pnr</u>	e-namn	f-namn	liuID	lösen	gata	gatnr	postnr	postort
------------	--------	--------	-------	-------	------	-------	--------	---------

Kurs

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

Anställd

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

Telefon

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

Institution

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

Projekt

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

Registrerad-På

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

Integritetsvillkor från deltagande:
Not Null gäller följande attribut:
Kurs: kursansv, inst
Anställd: inst
Varje kursinstans i Kurs måste
finnas i Registrerad-på.

Problem med Kurs

Kurs

<u>Kurskod</u>	Namn	<u>LäsÅr</u>	<u>period</u>	Poäng	Kursansv	Institution
729G28	Webbpgm o Datab	2015	ht1	6	Erik Prytz	IDA
729G28	Webbpgm o Datab	2022	ht2	6	Eva Ragnemalm	IDA
732G16	Databaser, design	2018	vt2	7,5	Eva Ragnem	IDA
TSBK07	Datorgrafik	2022	ht1	6	Ingemar Ragnemalm	ISY
TGTU90	Vetenskap filosofi	2023	?	4	Maria Eidenstam	TEMA

Normalisering

En metod att finna redundans i en relation

Normaliserings-villkor = Normalformer

Begrepp:

- atomära attribut, supernycklar, kandidatnycklar, primärnycklar och nyckelattribut (primattribut)
- funktionella beroenden, determinant
- informationsbevarande relationsschemauppdelning

1:a Normalform (1NF)

Alla attribut ska vara **atomära** (odelbara)

- Relationsmodellen antar att alla attribut är odelbara.
- 2:a normalform, 3:e normalform, BCNF baseras på **Funktionella beroenden** i relationen.

Funktionellt beroende

X är en delmängd av attributen i en relation.

- Om X entydigt bestämmer värdet på ett attribut Y (i samma relation), kallas det att "Y är **funktionellt beroende** av X" eller att "X bestämmer Y".
- $X \Rightarrow Y$
- X kallas **determinant** i det funktionella beroendet.

Exempel

Ex: Student

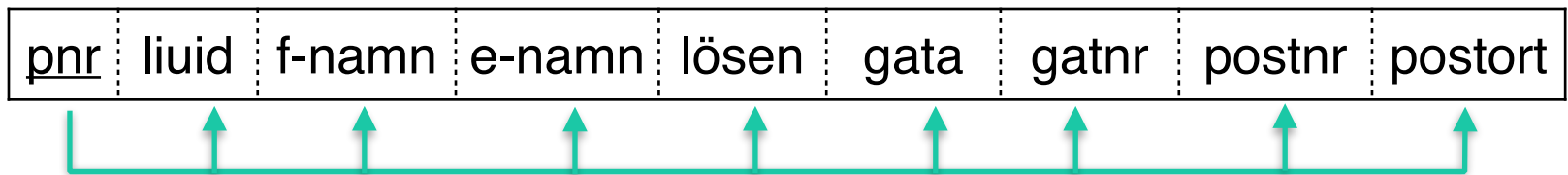
{pnr}⇒f-namn, {pnr}⇒e-namn, {pnr}⇒liuid, {pnr}⇒lösen,
 {pnr}⇒Gata, {pnr}⇒GatNr, {pnr}⇒PostNr, {pnr}⇒PostOrt,
 {pnr}⇒{pnr}

Kan också skrivas:

{pnr}⇒{pnr, e-namn, f-namn, liuid, lösen, Gata, GatNr,
 PostNr, PostOrt}

Och ritas:

Student



Exempel funktionellt beroende

- Determinanten kan vara flera attribut
ex: Kurs
 $\{\text{KursKod}, \text{läsår}, \text{period}\} \Rightarrow \{\text{namn}, \text{poäng}, \text{kursansv}, \text{inst}\}$
- Determinanten behöver inte vara nyckelattribut
ex Student:
 $\{\text{PostNr}\} \Rightarrow \{\text{PostOrt}\}$

Funktionella beroenden, forts

- Funktionella beroenden baseras på relationens semantik. Ett fb ska gälla i alla möjliga instanser av databasen!
- OBS:
 - Att det finns ett fb $X \Rightarrow Y$ betyder **inte** att det finns ett fb $Y \Rightarrow X$
 - ex: $\text{pnr} \Rightarrow \text{Namn}$ betyder inte att $\text{Namn} \Rightarrow \text{pnr}$

Funktionella beroenden kan härledas

- Transitivitet: om $X \Rightarrow Y$ och $Y \Rightarrow Z$ så $X \Rightarrow Z$
 - Kan ibland vara svårt att avgöra vilket som ”kom först” (ex: pnr, liuid)
- Reflexivitet: $X \Rightarrow X$ alltid.
 - Om $Y \subseteq X$ så $X \Rightarrow Y$
- Augmentation: Om $X \Rightarrow Y$ så har vi också $\{XZ\} \Rightarrow YZ$

Fullt funktionellt beroende

Givet: $X \Rightarrow Y$.

- Om inget attribut kan tas bort ur X utan att vi förlorar det funktionella beroendet (dvs X är minimal), kallas det **fullt funktionellt beroende**, ffb.

Fullt funktionellt beroende, exemplet Kurs

Kurs

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

- Nyckeln:
 $\{\text{kurskod, läsår, period}\} \Rightarrow \{\text{namn, poäng, kursansv, inst}\}$
- MEN: $\{\text{kurskod, läsår, period}\} \Rightarrow$ poäng är FB men ej FFB
– för ett program får inte ändra poäng hur som helst!
- OBS att verkligheten styr!

Nycklar och funktionella beroenden

- Supernyckel
 - en attributmängd som har **fb** till alla attribut i relationen!
 - (pga reflexivitet finns alltid en! - Vilken?)
- Kandidatnyckel
 - en minimal supernyckel
- Primärnyckel
 - den kandidatnyckel som väljs ut.

Nyckelattribut

- Attribut som ingår i **någon kandidatnyckel** till relationen.
- Student (pnr, e-namn, f-namn, Gata, Gatnr, Postnr, postort, liuid, lösen)
 - pnr kan identifiera en rad
 - men liuid kan det också, är också **kandidatnyckel!**
 - pnr och liuid är **nyckelattribut** i Student

Nyckelattribut forts.

- Nyckelattribut markeras i relationsschemat med *
 - Student (pnr*, e-namn, f-namn, liuid*,... lösen)
 - Kurs (kurskod*, namn, period*, läsår*, poäng, institution, kursansvarig)

Student

<u>pnr</u> *	liuid*	e-namn	f-namn	...	lösen
--------------	--------	--------	--------	-----	-------

Kurs

<u>kod</u> *	namn	<u>period</u> *	<u>År</u> *	poäng	Institution	Kursansvarig
--------------	------	-----------------	-------------	-------	-------------	--------------

Andra normalformen - 2NF

- Definition: Ett relationsschema R är i 2NF om det är i 1NF och varje icke-nyckelattribut A i R är **FFB** av varje kandidatnyckel i R .
- Eller som fråga: Är alla attribut som inte är nyckelattribut FFB av alla kandidatnycklar?

2NF Exemplet kurs

Kurs (kurskod, namn, läsår, period, poäng, kursansv, inst)

{Kurskod, läsår, period} \Rightarrow kursansv FFB

MEN:

{Kurskod, läsår, period} \Rightarrow namn inte FFB

{Kurskod, läsår, period} \Rightarrow poäng inte FFB

{Kurskod, läsår, period} \Rightarrow inst inte FFB

- Uppfyller ej 2NF

Relationsschemauppdelning

Definition: En uppdelning av en relation R i delarna R_1 och R_2 är **informationsbevarande** om:

- $R_1 * R_2 = R$

där $*$, naturlig sammansättning (join), är en relationsalgebra-operator som:

- sätter ihop varje tupel i R_1 med varje tupel i R_2 och väljer ur den relationen de tupler där attribut som heter (har samma domän) har lika värde.
- är samma sak som sql: $R_1 \text{ join } R_2 \text{ on } a=b$, **om** $R_1.a$ är en främmande nyckel till R_2 och b är nyckel till R_2

Informationsbevarande relationsschemauppdelning

Om vi delar upp en relation R i relationerna R_1 och R_2 så blir uppdelningen **informationsbevarande**

- om R_2 innehåller en determinant och de attribut som bestäms av den
- och R_1 innehåller ”resten av attributen i R ” plus determinanten i det funktionella beroendet.
- (och inget av attributen som bara finns i R_2 ingår i en determinant som bestämmer attribut i R_1).

Informationsbevarande relationschemauppdelning, inte

- Person(Pnr,Namn,Adress)
med funktionella beroenden:
Pnr \Rightarrow Namn, Pnr \Rightarrow Adress,
inte Namn \Rightarrow Adress

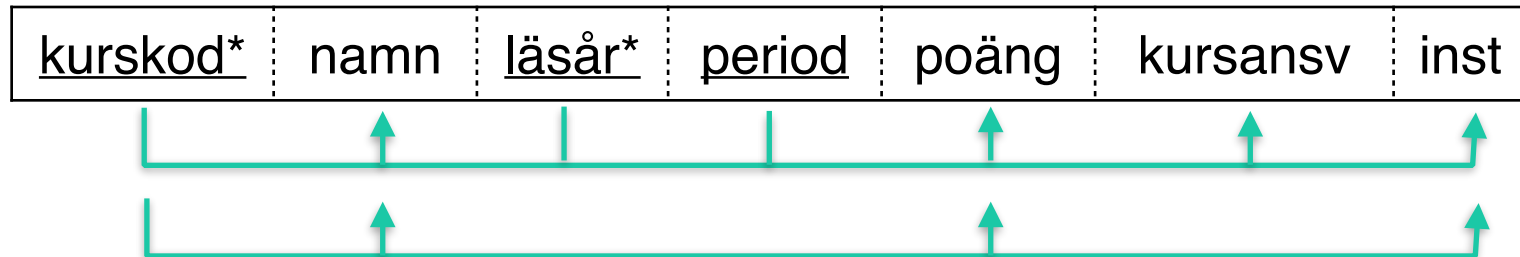
pnr	namn	Adress
7908101234	Anna	Ågatan 3
8112237890	Anna	Rydsv.12

pnr	namn
7908101234	Anna
8112237890	Anna

namn	Adress
Anna	Ågatan 3
Anna	Rydsv.12

Exemplet Kurs

Kurs



Ger:

Kurs1(Kurskod, namn, poäng, inst)

Kurs2(Kurskod, läsår, period, kursansv)

Tredje normalform - 3NF

- Definition: Ett relationsschema R är i 3NF **om** det är i 2NF **och** inget icke-nyckel-attribut är ffb av något annat icke-nyckel-attribut
- Frågan för 3NF:
 - Finns ffb mellan icke-nyckel-attribut?

Exemplet Student

Student (pnr*, e-namn, f-namn, gata, postnr, postort, liuid*, lösen)

Funktionella beroenden:

$\{pnr\} \Rightarrow \{namn, gata, postnr, postort, liuid, lösen\}$

$\{liuid\} \Rightarrow \{pnr, namn, gata, postnr, postort, lösen\}$

$\{postnr\} \Rightarrow postort$

- Uppfyller INTE 3NF

Boyce-Codd normalform - BCNF

- Ett relationsschema R är i BCNF
om det är i 3NF
och inget attribut är ffb av något icke-nyckel-attribut
- (Eller, i princip ekvivalent: om det är i 1NF och varje determinant är en kandidatnyckel)
- Frågan för BCNF: är något attribut beroende av en determinant som innehåller icke-nyckel-attribut?
(eller: är varje determinant en kandidatnyckel?)

3NF/BCNF

- Inte alltid möjligt att transformera ett schema till BCNF och behålla alla beroendena.
- 3NF har de flesta av BCNF's fördelar.
- Det är inte självklart att man måste uppfylla BCNF.
- OBS: 3NF tillåter någon sorts redundans som BCNF inte gör (funktionella beroenden mellan nyckelattribut).

Frågor?