

INTRODUKTION TILL ER ENTITY-RELATIONSHIP

Mahmud Al Hakim
mahmud@webacademy.se

1

REFERENS TILL DETTA MATERIAL: WWW.DATABASTEKNIK.SE/WEBBKURSEN

En webbkurs om databaser

Av [Thomas Padron-McCarthy](#)

Den här webbkursen om databaser är baserad på föreläsningar som jag har hållit i olika databaskurser. Kursen är klar, men det kan komma att ske ändringar och tillägg. Den motsvarar en bok på 100-150 sidor. Kursen tar inte upp särskilt mycket om hur en databashanterare arbetar internt, utan det handlar om att använda databashanteraren: för att skapa egna databaser, stoppa in data i dem, och söka i dem.

Det här är förstasidan i en webbkurs om databaser som finns fritt tillgänglig på adressen <http://www.databasteknik.se/webbkursen/>. Senaste ändring av den här sidan: 21 januari 2011.

Av [Thomas Padron-McCarthy](#). Copyright, alla rättigheter reserverade, osv. Skicka gärna kommentarer till webbkursen@databasteknik.se.

2

KONCEPTUELLA DATAMODELLER

Om man ska skapa en databas som beskriver en del av verkligheten t.ex. ett företag, brukar man börja med att göra en beskrivning av hur den delen av verkligheten ser ut och fungerar.

Denna beskrivning på hög nivå kan kallas en **konceptuell** eller begreppsmässig beskrivning.

I stället för beskrivning säger man ofta **schema**.

Om man vill skapa en databas måste det konceptuella schemat översättas till ett schema som går att mata in i en databashanterare.

3

ER-DIAGRAM

En vanlig konceptuell datamodell är den så kallade **ER-modellen**.

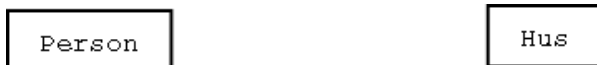
"ER" står för "**Entity-Relationship**", dvs ungefär "saker" och "samband".

ER-diagram är ett enkelt sätt att designa och beskriva saker som finns i databasen samt sambanden mellan dessa saker.

4

GRUNDERNA FÖR HUR MAN RITAR ERD

Rita upp de typer av saker som finns i databasen.
De kallas **entitetstyper**, och ritas som fyrkantiga lådor.



Rita sen upp de samband som finns mellan de olika typerna av saker. De kallas **sambandstyper**, och ritas som diamanter mellan de fyrkantiga lådorna



5

SAMBANDSTYPER

Sambandstyper kan vara av tre olika slag, så kallade kardinalitet (Cardinality).

1. Ett-till-ett-samband (1:1-samband)
2. Ett-till-många-samband (1:N-samband)
3. Många-till-många-samband (N:M-samband)

6

ETT-TILL-ETT-SAMBAND (1:1-SAMBAND)

En sambandstyp där en sak av något slag kan höra ihop med en sak av ett annat slag, och varje sak av det andra slaget kan höra ihop med en sak av det första slaget.

Exempel: En person kan vid ett och samma tillfälle bara köra en bil, och varje bil kan bara köras av en person.



7

ETT-TILL-MÅNGA-SAMBAND (1:N-SAMBAND)

En sambandstyp där en sak av något slag kan höra ihop med flera saker av ett annat slag, men varje sak av det andra slaget kan bara höra ihop med en sak av det första slaget.

Exempel: En person kan äga flera bilar, men varje bil kan bara ägas av en person.

Om man vänder på det och börjar med bilarna blir det i stället ett *många-till-ett-samband* (N:1-samband).



8

MÅNGA-TILL-MÅNGA-SAMBAND (N:M-SAMBAND)

En sambandstyp där en sak av något slag kan höras ihop med flera saker av ett annat slag, och varje sak av det andra slaget kan höras ihop med flera saker av det första slaget.

Exempel: En person kan äga flera hus, och varje hus kan ägas gemensamt av flera personer.



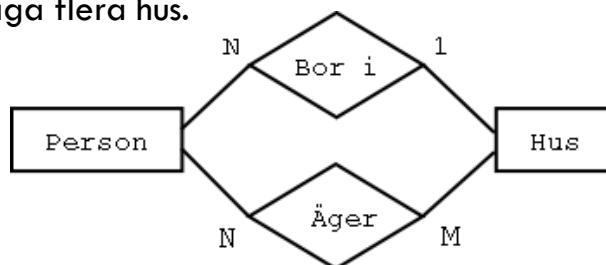
9

FLERA SAMBANDSTYPER MELLAN SAMMA ENTITETSTYPER

Det går bra att ha flera olika sambandstyper som binder ihop samma entitetstyper.

Till exempel kan personer kanske bo i husen, men de kan också äga husen.

Hus kan ägas gemensamt av flera personer, och varje person kan vara med och äga flera hus.



10

FULLSTÄNDIGT DELTAGANDE

Om varje person som är med i databasen måste bo i ett hus, kan man rita ut det villkoret med ett dubbelstreck mellan Person och Bor i.

Det kallas fullständigt deltagande: alla personer som finns med i databasen måste delta i ett boendesamband.

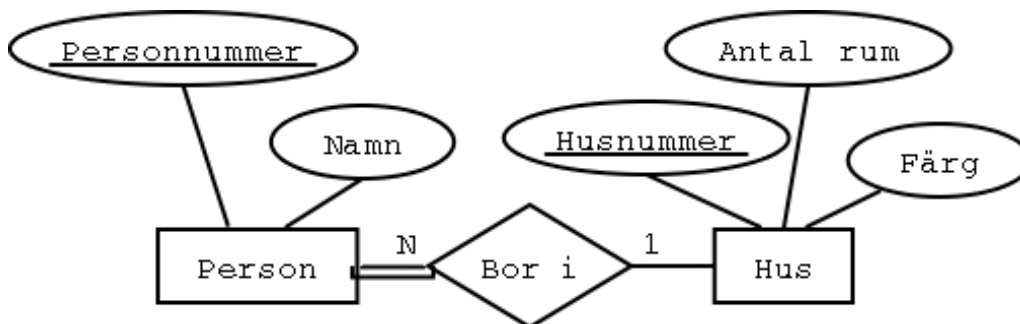


11

ATTRIBUT

Om sakerna eller sambanden har egenskaper, så kallar man dem attribut, och ritas som ovaler.

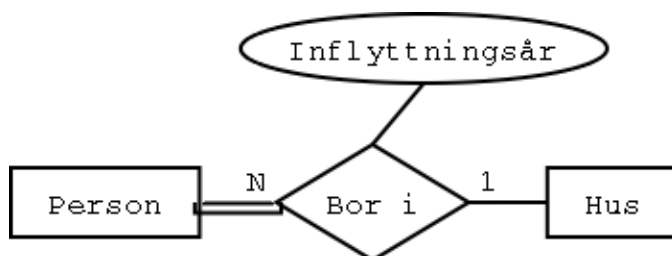
OBS! Personnummer och Husnummer är primära nycklar.



12

SAMBANDSTYPER KAN HA EGENSKAPER

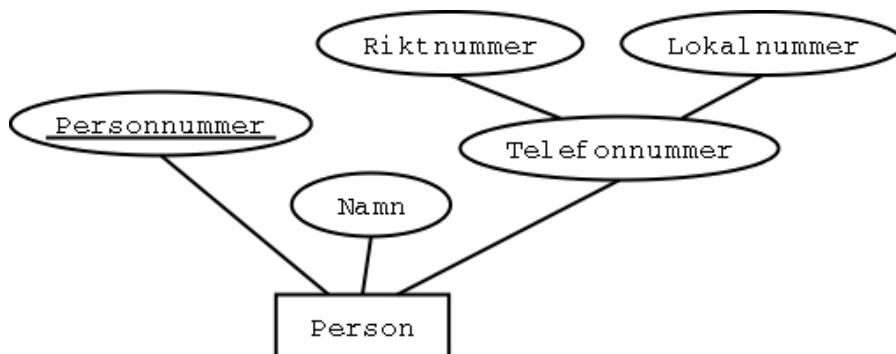
Inflyttningsår anger vilket år en person flyttade in i det hus hon bor i. I det här fallet skulle man också kunna sätta attributet Inflyttningsår på personen, eftersom varje person bor i exakt ett hus, och alltså har exakt ett inflyttningsår, men det är nog naturligare att låta inflyttningsåret höra till Bor i-sambandet.



13

SAMMANSATTA ATTRIBUT

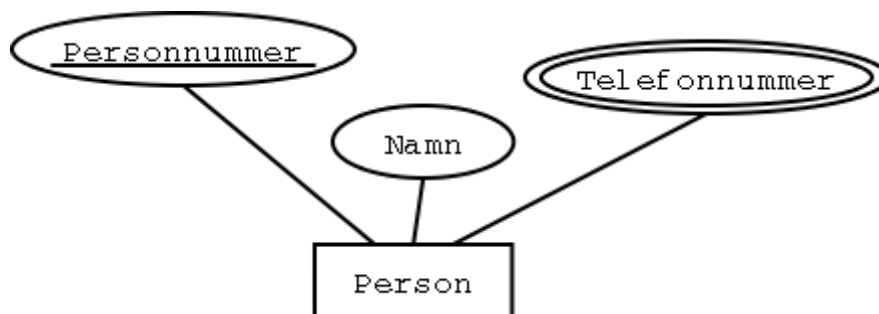
Ibland kan ett attribut vara sammansatt av flera delar, som hör ihop men som man även vill behandla var för sig. Då kan man rita det som ett *sammansatt attribut*.



14

FLERVÄRDA ATTRIBUT

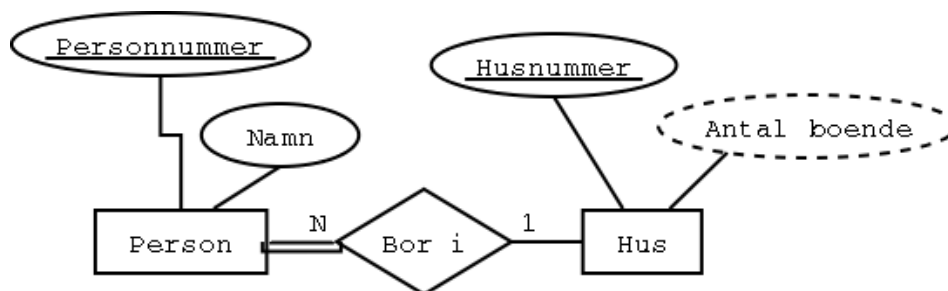
Vill man kunna lagra flera telefonnummer som hör till en person. Då kan man rita det som ett *flervärt attribut* (även kallat multipelt attribut) genom att använda en dubbelellips runt attributnamnet



15

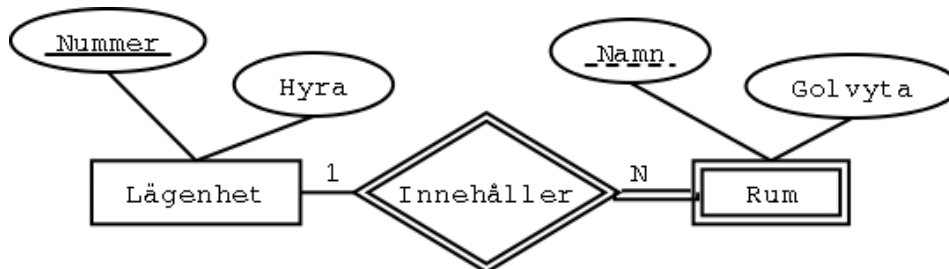
HÄRLEDDA ATTRIBUT

En del attribut vill man kanske inte lagra i databasen, utan man kan räkna ut dem utifrån andra data som redan finns i databasen. Sådana *härledda attribut* markeras med en streckad oval.



16

SVAGA ENTITETSTYPER



Lägenheterna innehåller rum, men rummen har bara namn, som inte är unika för hela databasen.

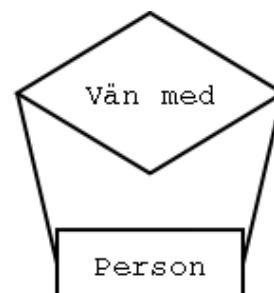
Däremot är de unika inom en viss lägenhet: om man vet ett rumsnamn, och numret på den lägenhet som rummet hör till, så räcker det för att unikt identifiera rummet i hela databasen.

17

SKILJ PÅ SCHEMA OCH DATA

ER-diagrammet är ett schema, och beskriver vilka data som kan lagras. Det är inte en avbildning av datat.

Det här ER-diagrammet säger alltså inte att det finns en person som är vän med sig själv, utan det säger att personer kan vara vänner med andra personer:



18

SKILJ PÅ ER-DIAGRAM OCH TABELLER

Även om man ofta översätter ER-diagram till tabeller, för att kunna lagra sin databas i en relationsdatabashanterare, så har ER-diagrammet i sig inget med tabeller att göra.

Det man ritar ut i ett ER-diagram är alltså inte tabeller och kopplingarna mellan dem, även om en del (men inte alla) av entitetstyperna sen kan översättas till tabeller, och en del (men inte alla) av sambandstyperna kan översättas till kopplingar mellan tabeller.

19

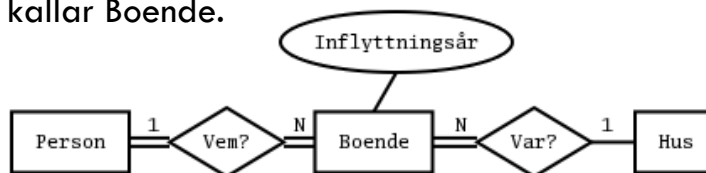
OBJEKTIFIERING AV SAMBANDSTYPER

Ibland kan man välja mellan att använda sig av en sambandstyp eller en entitetstyp.

Att använda en entitetstyp i stället för en sambandstyp betyder att man betraktar kopplingen som en egen sak, och inte bara som en koppling mellan två andra saker.

Det kallas ibland **objektifiering**.

Till exempel kan vi göra om Bor i-sambandet till en egen entitetstyp, som vi kallar Boende.



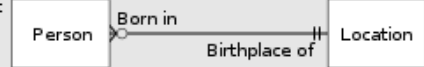
20

ALTERNATIVA NOTATIONER

Chen



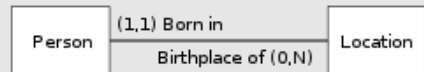
Martin / IE /
Crow's Foot



IDEFIX



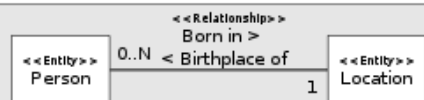
Min-Max / ISO



Bachman



UML



Bildkälla: https://en.wikipedia.org/wiki/Entity%E2%80%93relationship_model

21