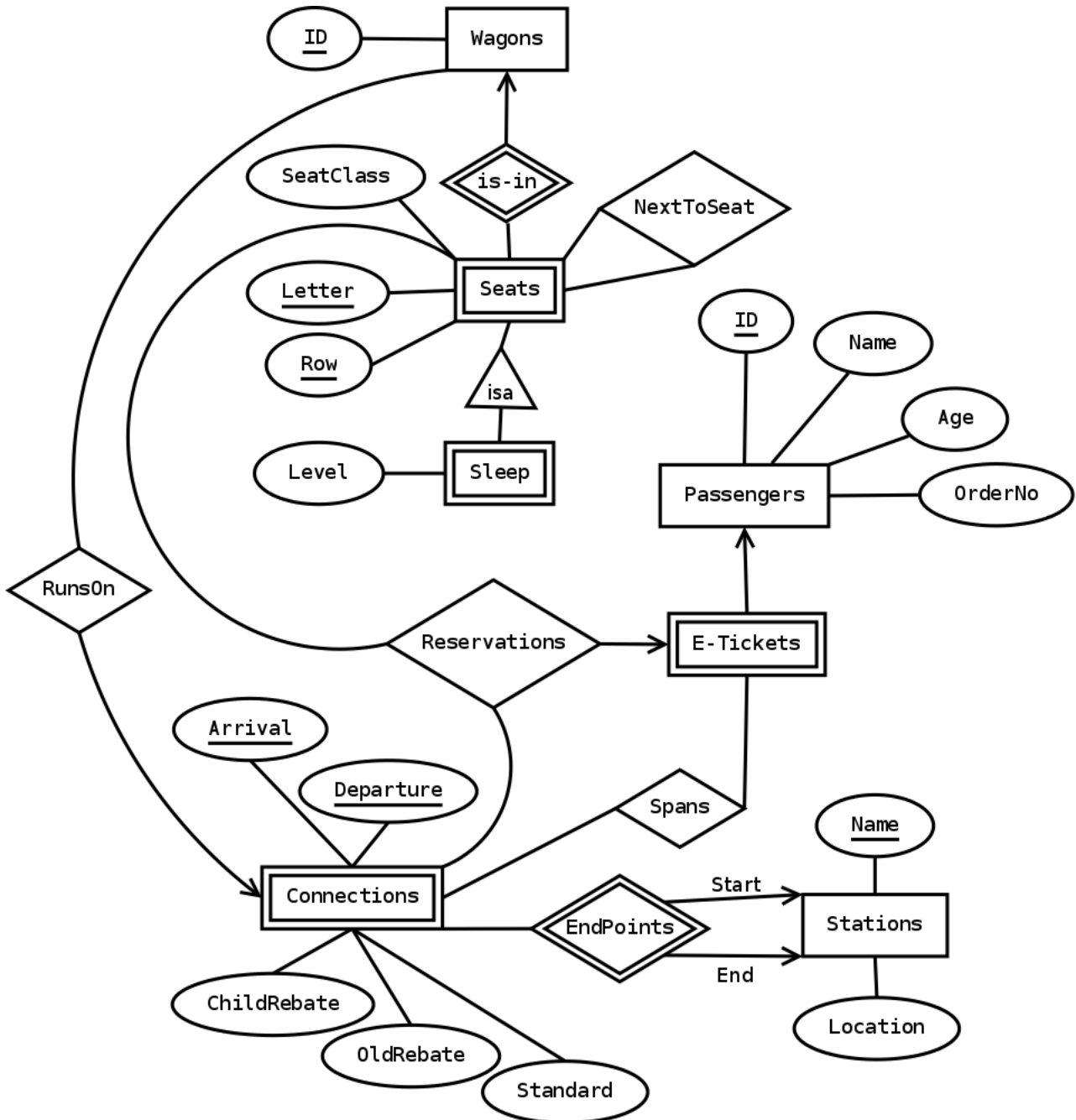


dDB Exercise Week 2

Tilføjelse af constraints

Disse findes i 5 former; Key, Single-value, Referential integrity, Domain og General constraints.

En tilføjelse af disse til E/R diagrammet ser ud som følger:



Opdateret E/R

Som det ses, så er key constraints de understregede attributter i E/R diagrammet. Disse, og så Referential integrity constraints, som er de relationships som forbinder de forskellige entity-sets, de er de eneste constraints jeg mener jeg kan angive direkte i mit E/R diagram.

I E/R diagrammet er weak entity-sets også markerede, de kommer i betragtning når nu vi skal lave skemaerne for relationerne lidt senere.

Andre constraints såsom bl.a. domain constraints er betingelsen om, hvilken type indhold attributterne har. Sådant som, at 'Passengers(age)' attributten, og de forskellige 'Connections(*Rebate)' attributters indhold er, og kan kun repræsenteres ved integers og ikke tekst. Rebate er en procent-sats, som trækkes fra prisen for den valgte SeatClass, over den pågældende Connection.

Single value constraints binder sig til unikke attributter, det er f.eks alle key-attributterne. De må selv sagt kun optræde én enkelt gang. En single value constraint, er også OrderNo-attributten til Passengers entiteten. Der er et unikt ordre nummer pr. billet bestilling, men den er ikke en key, da den ikke unikt identificerer en passager i vores database.

Yderligere så har E/R diagrammet fået disse ændringer. Som nævnt, så er Passengers-entiteten er nu inkluderet, til at indeholde passagerdata på tværs af bestillinger. Seats entity-sets har fået en NextToSeat, så nabosæder kan lokaliseres, en SeatClass som fortæller om det er et 1. klasse eller standard sæde. En ISA relation til Sleep entiteten, fortæller om 1. klasses eller standard sædet er en soveplads, og i hvilket niveau den befinder sig.

Fra E/R til Relationelt design.

Ud fra det producerede E/R diagram er det nu muligt, at danne de skemaer som beskriver databasestrukturen. Disse skemaer formes ud fra de i E/R diagrammet konstruerede entity-sets og relationships. Vi har givet tre fremgangsmåder til at konvertere disse diagrammer. Den første er E/R-style konvertering, så er det den Objekt Orienterede tilgang, og sidst er det NULLs metoden.

Da jeg har et E/R diagram, har jeg valgt at foretage en E/R-style konvertering. Det gør jeg ved at danne relationer ud fra mine entity-sets og mine relationships, i det omfang det behøves. NULLs metoden synes jeg ikke virker overskuelig, da alle mine attributter og entity-sets ville skulle vises i en enkelt relation og OO-tilgangen ville, hvis vi har $n+1$ entity-sets, i værste fald producere 2^n relationer. Da vi skal minimere pladsen og undgå redundans, valgte jeg derfor E/R-style konvertering.

Under hver relation vil jeg liste de enkelte constraints, hvis der er nogle knyttet dertil. Som tidligere nævnt, er alle attributter som er Keys, også Single Value constraints, så dem vil jeg ikke liste.

Skemaer dannet fra E/R

- Wagons(ID)
- Seats(Letter, Row, SeatClass, isaSleep, wagon_id)
 - > SeatClass: er enten '1. klasse' eller 'standard'
 - > isaSleep: Boolean om det er en soveplads
 - > wagon_id: er togvognens unikke ID
- NextToSeat(r_Letter, r_Row, r_wagon_id, l_Letter, l_Row, l_wagon_id)
 - > r_: er sædet til venstre for
 - > l_: er sædet til højre for
- Sleep(Level, SeatLetter, SeatRow, wagon_id)
 - > Level: er '1' eller '0'
- Connections(Departure, Arrival, ChildRebate, OldRebate, Std.Rebate, StartName, EndName, wagon_id)
- Spans(Departure, Arrival, StartName, EndName, E-TicketID)
- RunsOn(wagon_id, Departure, Arrival, StartName, EndName)
- E-Ticket(E-TicketID)
- Passengers(ID, Name, Age, OrdreNo)
 - > Age: er et positivt tal
- Reservations(PassengerID, Departure, Arrival, StartName, EndName, SeatLetter, SeatRow, SeatClass, isaSleep, wagon_id)
 - > isaSleep: Boolean om det er en soveplads