

Relational Model

Rules:

Every entity (strong, weak, super entity, sub entity) becomes a table in the relational model

The attributes and the identifiers of entities become columns and primary keys (PK) of the tables.

Relationships:

One-to-One: Either one of the sides takes the other side's PK as Foreign Key (FK). Not BOTH!!!

One-to-Many: Many side takes the PK of One side as FK.

Many-to-Many: We create a new table for the many-to-many relationship. This new table takes both entities PKs. Combination of these columns become PK (we have a composite PK), and separately FKs.

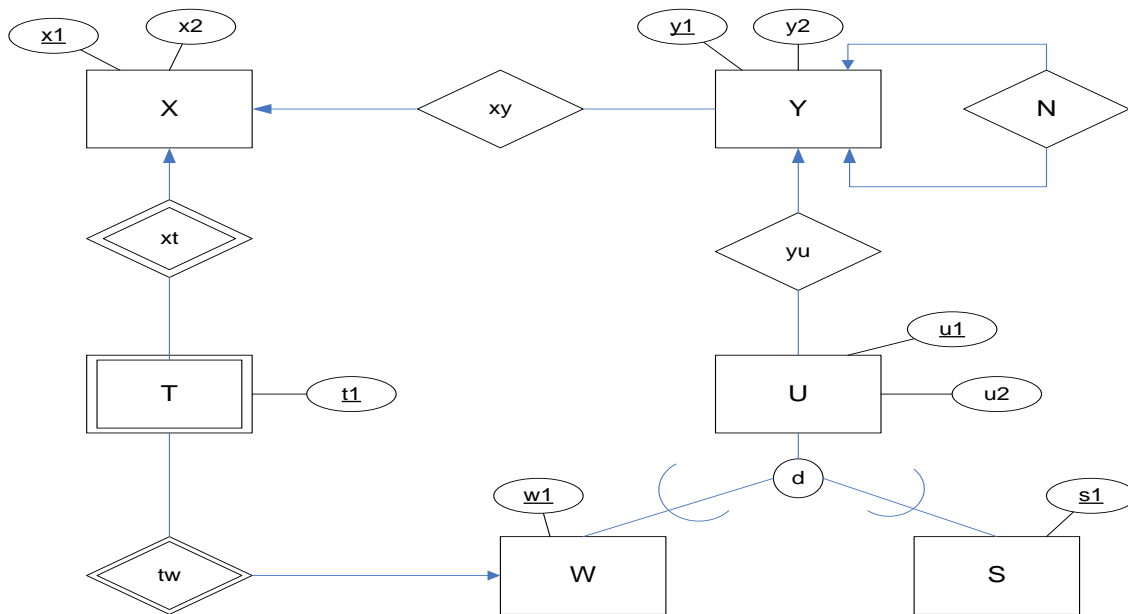
Identifying Relationship: we might either have one-to-many or one-to-one identifying relationship between the weak entity and its strong entities. For both cases: strong entities PK should be added to the table that will be created for weak entity as PART of the PK and as a FK.

Calculated/Derived Attributes, generally do not become part of the Relational Schema.

Composite Attributes: Only the sub-fields of the main attribute will be added into the relation (table).

Multi-valued Attributes (MVA): A new table is created for the MVA in the relational schema. The table will be created without adding the MVA. The table that will be created for MVA will receive the PK of the parent table (appears as part of the PK as well as FK), and the attribute itself will be added into the new table as part of the PK.

Is-a relationship:



We applied Solution 3 for the inheritance relationship

$X(\underline{x1}, x2)$

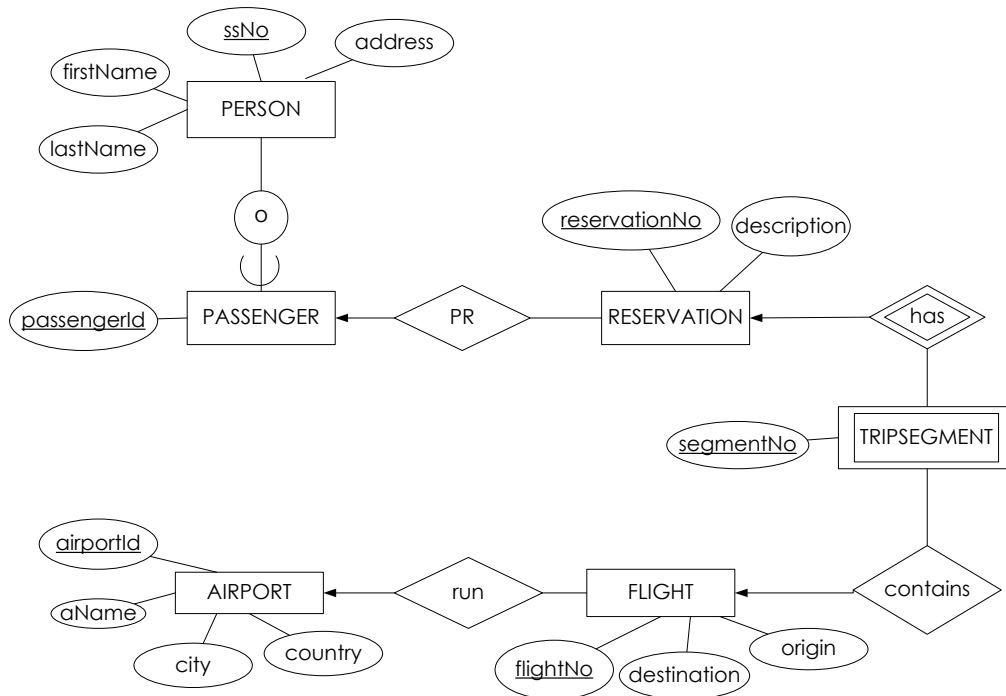
$Y(\underline{y1}, y2, \textcircled{x1}, \textcircled{Ny1})$ $x1$: references $X(x1)$; $Ny1$: references $Y(y1)$

$U(\underline{u1}, u2, \textcircled{y1})$ $y1$: references $Y(y1)$

$W(\underline{w1}, \textcircled{u1})$ $u1$: references $U(u1)$

$S(\underline{s1}, \textcircled{u1})$ $u1$: references $U(u1)$

$T(\underline{t1}, \textcircled{x1}, \textcircled{w1})$ $x1$: references $X(x1)$; $w1$: references $W(w1)$



We apply solution 2 for the is-a relationship. PERSON will be eliminated. Passenger will inherit all attributes of the PERSON entity.

Passenger(passengerId, ssNo, firstName, lastName, address)

Reservation(reservationNo, description, pid) pid: references Passenger(passengerId)

Airport(airportId, aName, city, country)

Flight(flightNo, destination, origin, aportId) aportId: references Airport (airportId)

TripSegment(segmentNo, reservationNo, flightNo)

reservationId: references Reservation(reservationId); flightNo: references Flight(flightNo)