Entity & Referential Integrety

Entity integrity constraint: no primary key value can be null

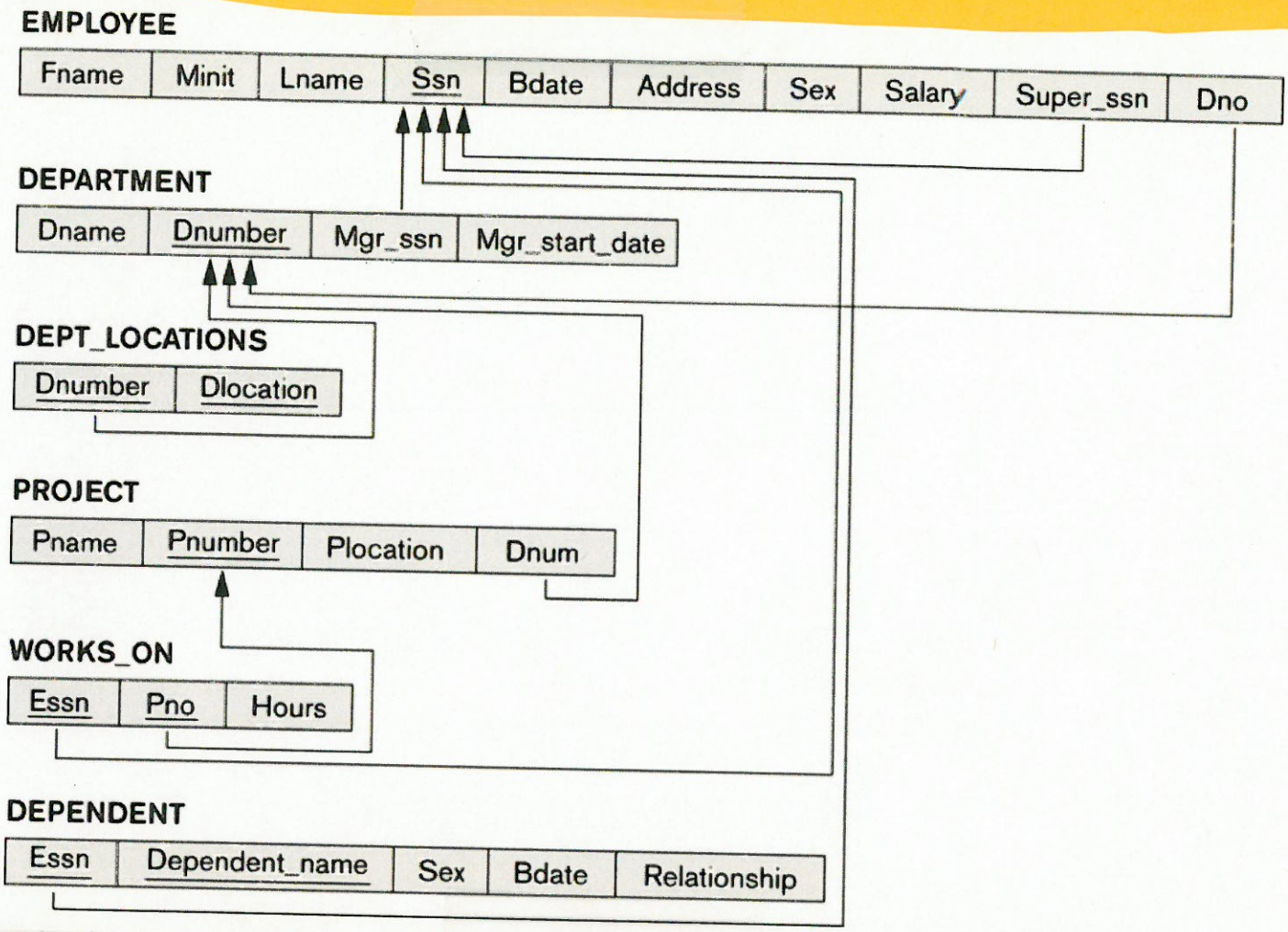
Referential integrity constraints:
specifed between 2 relations and maintains consistency between tuples in relation

Formally:

Foreign key: between relation R_1 and R_2
set of attributes FK in R_1 satisfying:

1. attributes in FK have same domain(s) as primary key attribute of R_2
2. for t_1 current state $r_1(R_1)$ and t_2 in current state of $r_2(R_2)$:
   a. $t_1[FK] = t_2[PK]$
   b. $t_1[FK] = NULL$

Example of Referential Integrity



Other Constraint types

Semantic integrity constraints:
semantics of data (usually at application level)

State constraints: what we have seen so far

Transition constraints: on changes in data
(e.g. salary can only increase)
(usually at application level)

Updates & constraint violations

Op types:

- retrievals (soon)
- modifications

modification ops:

- insert: add tuples to relation
- delete: removes tuples
- update: modifies tuples

Insert operation

- provide list of attributes for new tuple
- Constraints that can be violated
  - domain:
  - key:
  - entity integrity:
  - referntial integrity:
- How handled; operation rejected

Delete operation

- specify condition on tuples to delete

  E.g. Delete the EMPLOYEE tuples w/ birthday before 1950-01-01

Can violate;

- referential integrety

how handled (options);

- restrict: rejects the deletion
- cascade: propagates the deletion
- set default: modifies references to default value (or null)

Update operations

- specify condition of tuple (or tuples) and how to update

  E.g. Update salary of EMPLOYEE w/ Ssn = '999887777' to 28000

- Can violate:
  - domain
  - referential integrity
  - entity integrity
  - key

How to handle:

- Sim to insert or delete

Transactions

- a collection of operations (seen as atomic)
- at end constraints of schema must be satisfied