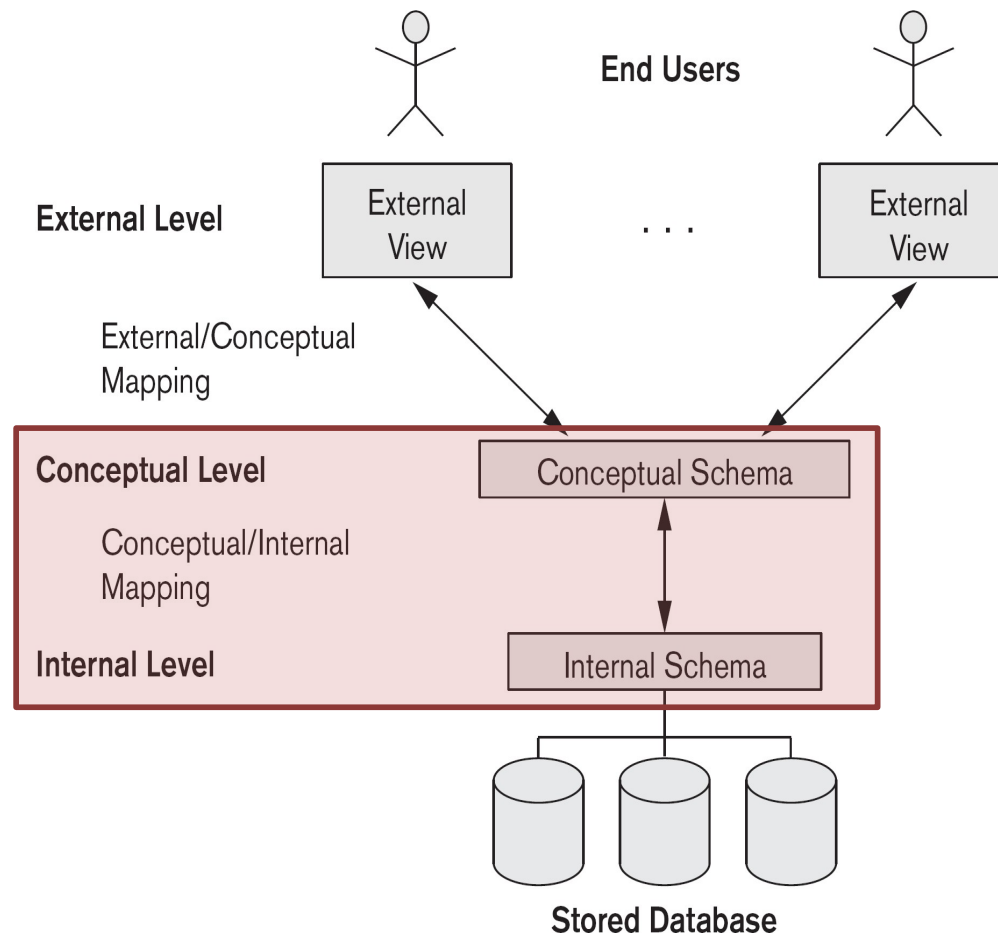




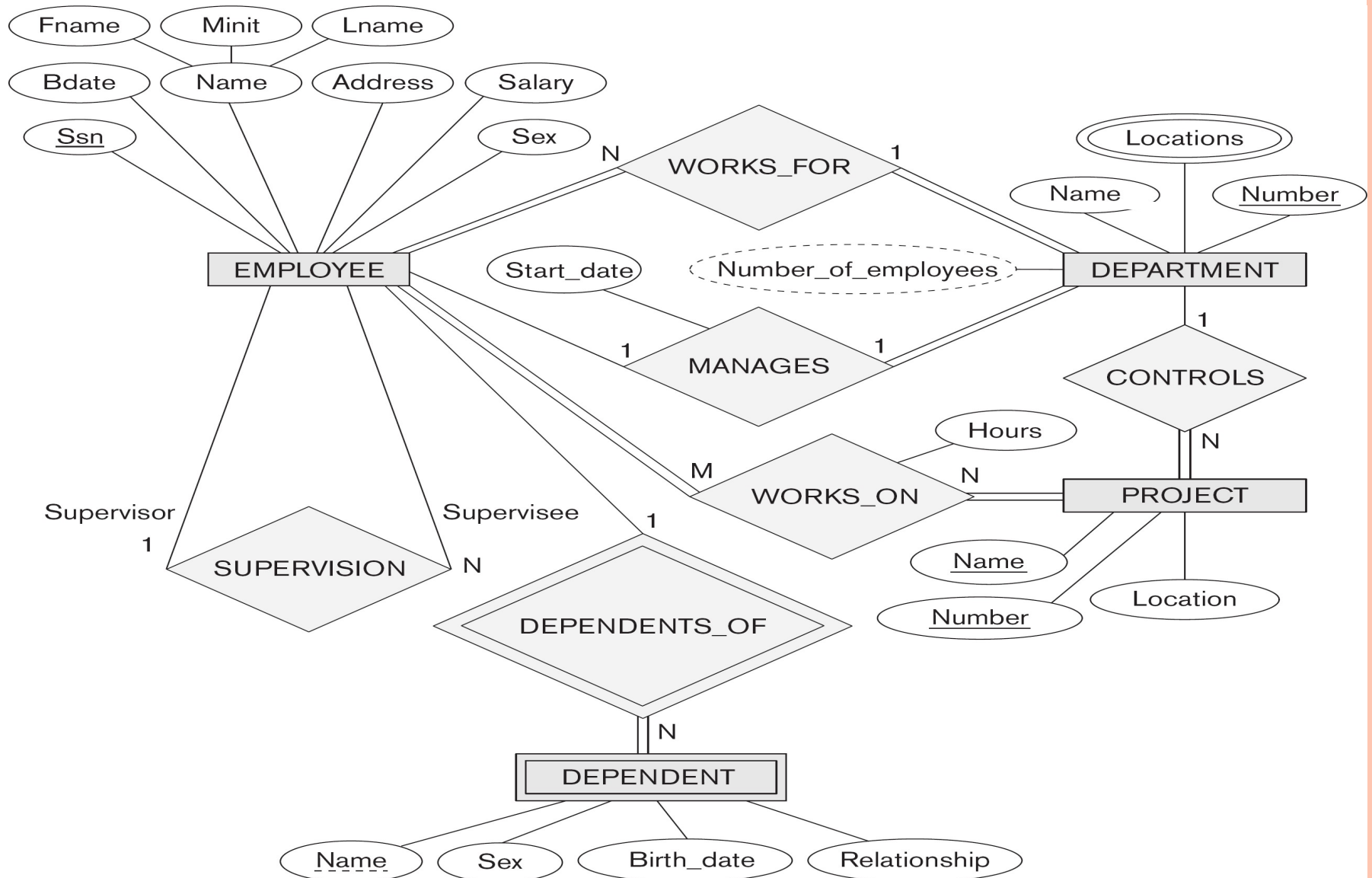
ER-to-Relational Mapping

Very important!!!!!!

DATA MODEL



EXAMPLE OF ER - DIAGRAM

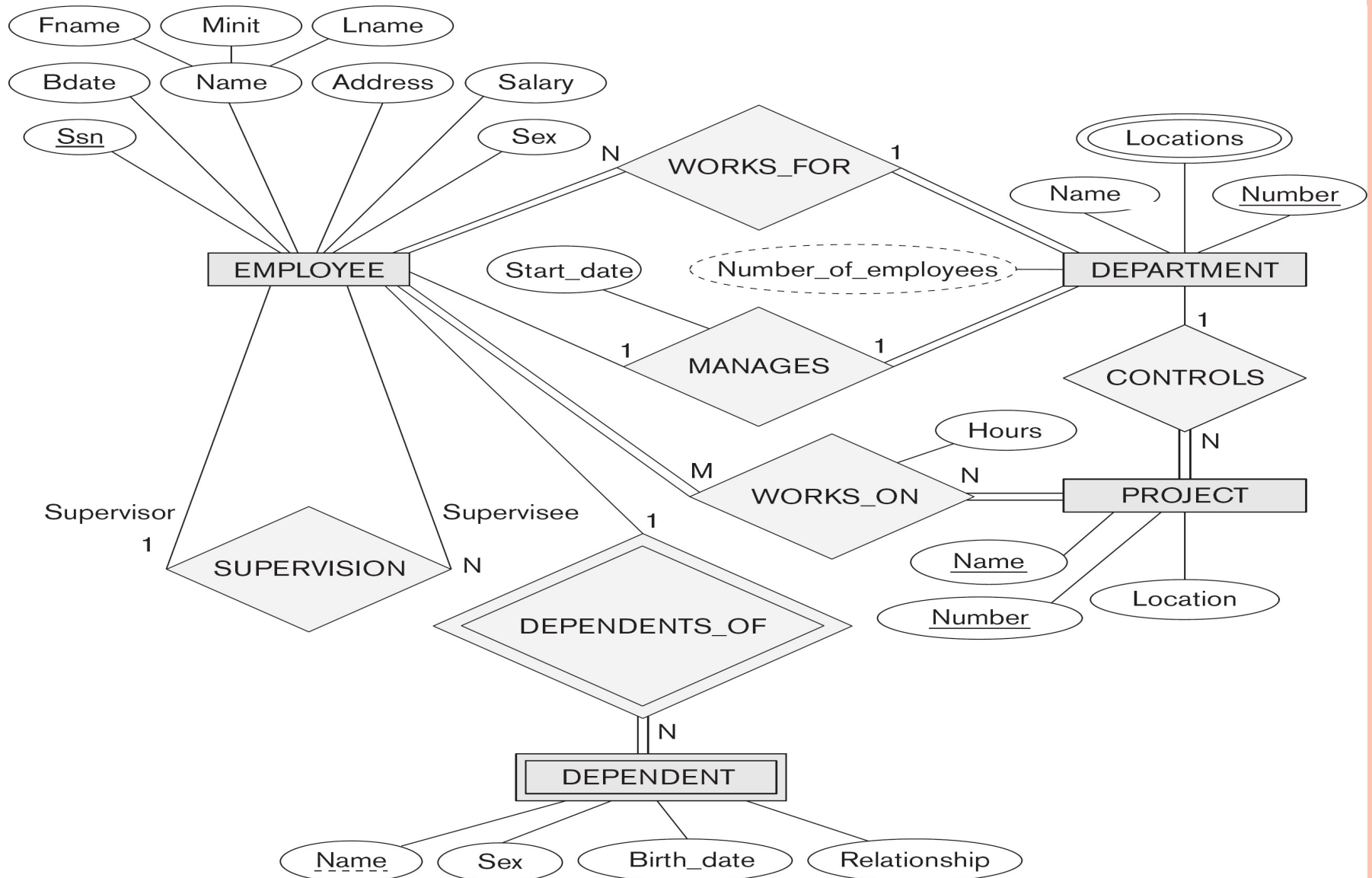


STEP 1: REGULAR ENTITY TYPES

- i. For each regular/strong entity type, create a corresponding relation that includes all the simple attributes (includes simple attributes of composite relations)
- ii. Choose one of the key attributes as primary
 - If composite, the simple attributes together form the primary key
- iii. Any remaining key attributes are kept as secondary unique keys (these will be useful for physical tuning w.r.t. indexing analysis)



EXAMPLE OF ER - DIAGRAM



STEP 1 RESULT

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary
-------	-------	-------	------------	-------	---------	-----	--------

DEPARTMENT

Dname	<u>Dnumber</u>
-------	----------------

PROJECT

Pname	<u>Pnumber</u>	Plocation
-------	----------------	-----------

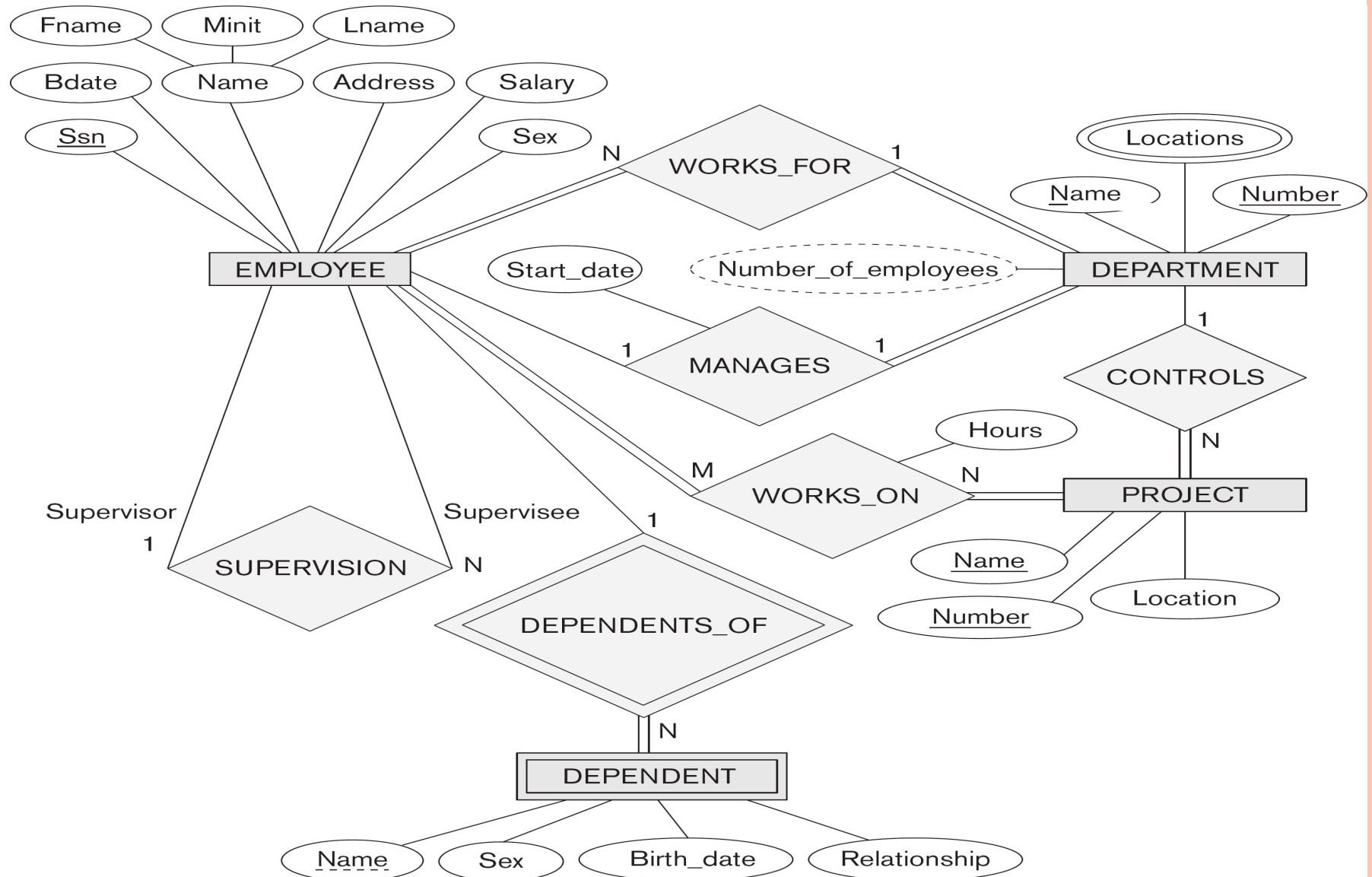


STEP 2: WEAK ENTITY TYPES

- i. For each weak entity type, create a corresponding relation that includes all the simple attributes
- ii. Add as a foreign key all of the primary key attribute(s) in the entity corresponding to the owner entity type
- iii. The primary key is the combination of all the primary key attributes from the owner and the partial key of the weak entity, if any



EXAMPLE OF ER - DIAGRAM



STEP 2 RESULT

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary
-------	-------	-------	------------	-------	---------	-----	--------

DEPARTMENT

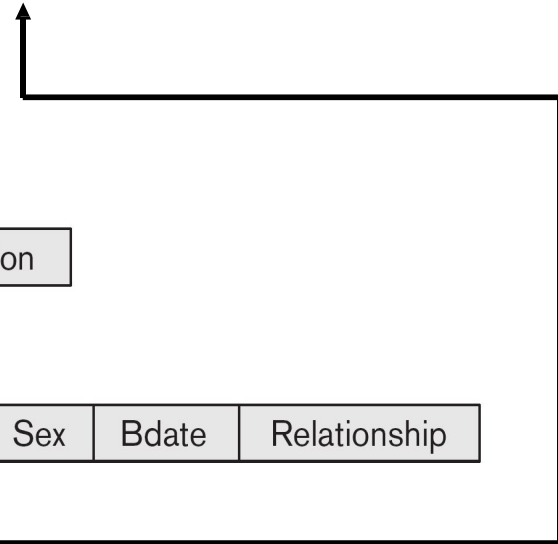
Dname	<u>Dnumber</u>
-------	----------------

PROJECT

Pname	<u>Pnumber</u>	Plocation
-------	----------------	-----------

DEPENDENT

<u>Essn</u>	<u>Dependent_name</u>	Sex	Bdate	Relationship
-------------	-----------------------	-----	-------	--------------



STEP 3: MAPPING BINARY 1-TO-1

□ Three approaches

- **Foreign Key**
 - Usually appropriate
- Merged Relation
 - Possible when both participations are total
- Relationship Relation
 - Not discussed



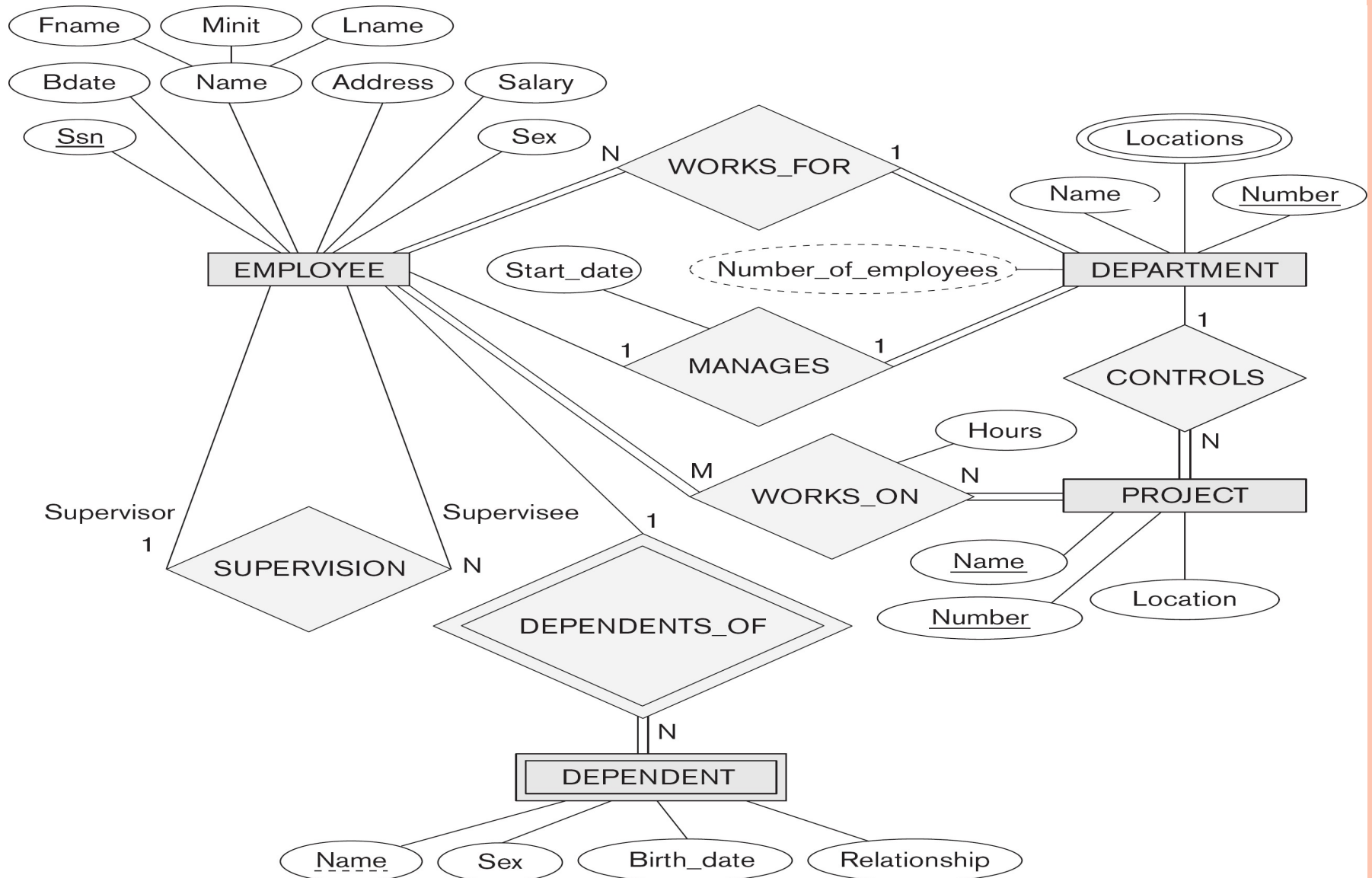
STEP 3: MAPPING BINARY 1-TO-1

Foreign Key

- i. Choose one relation as $S(dept)$, the other $T(employee)$
 - Better if $S(dept)$ has total participation (reduces number of NULL values)
- ii. Add to $S(dept)$ all the simple attributes of the relationship
- iii. Add as a foreign key in $S(dept)$ the primary key attributes of $T(employee)$



EXAMPLE OF ER - DIAGRAM



STEP 2 RESULT

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary
-------	-------	-------	------------	-------	---------	-----	--------

DEPARTMENT

Dname	<u>Dnumber</u>
-------	----------------



STEP 3 RESULT

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary
-------	-------	-------	------------	-------	---------	-----	--------

DEPARTMENT

Dname	<u>Dnumber</u>	Mgr_ssn	Mgr_start_date
-------	----------------	---------	----------------

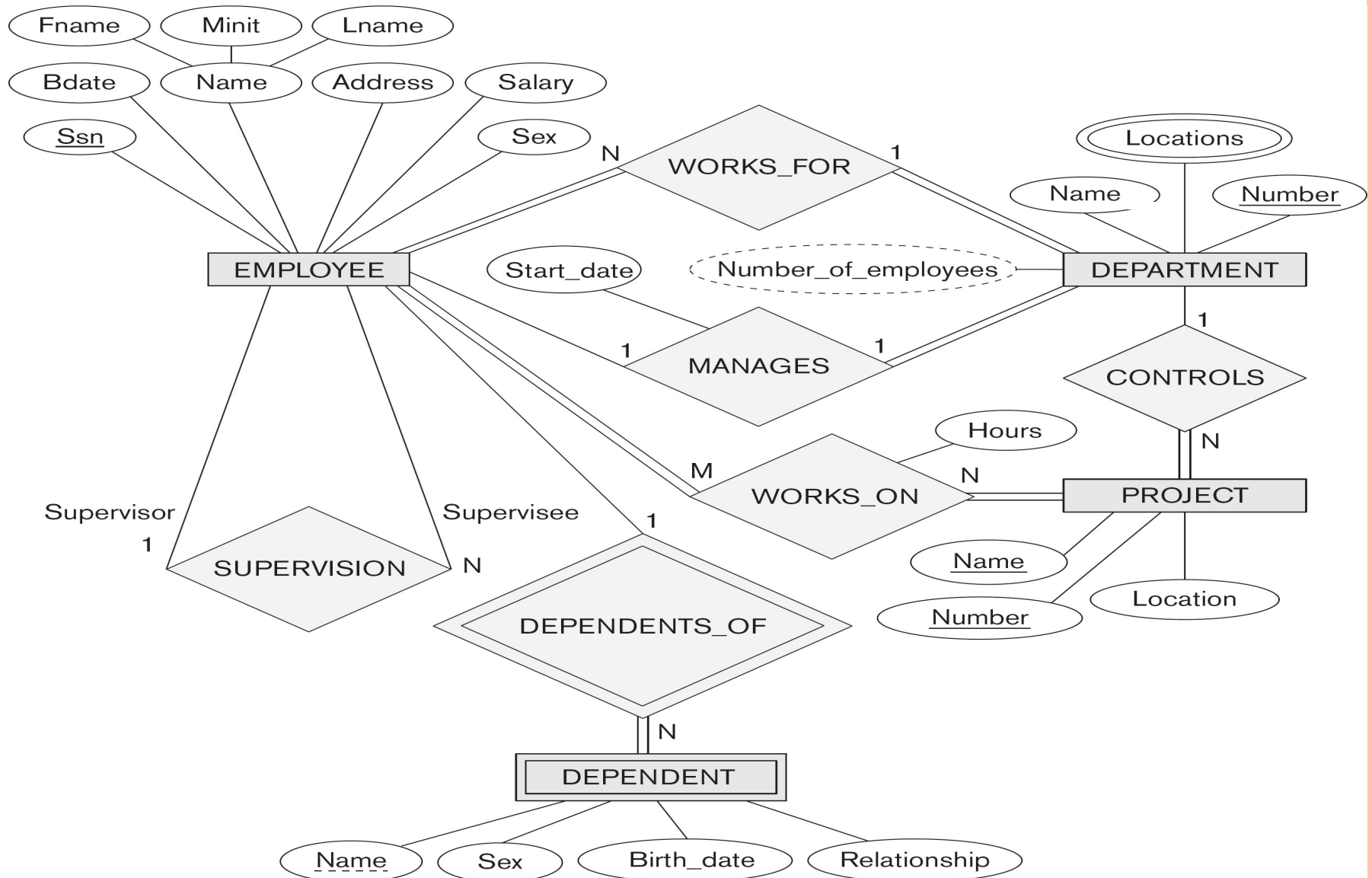


STEP 4: BINARY 1-TO-N

- i. Choose the $S(project)$ relation as the type at the N-side of the relationship, other is $T(dept)$
 - ii. Add as a foreign key to $S(project)$ all of the primary key attribute(s) of $T(dept)$
- Another approach: create a relationship relation



EXAMPLE OF ER - DIAGRAM



STEP 4 RESULT

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
-------	-------	-------	------------	-------	---------	-----	--------	-----------	-----

DEPARTMENT

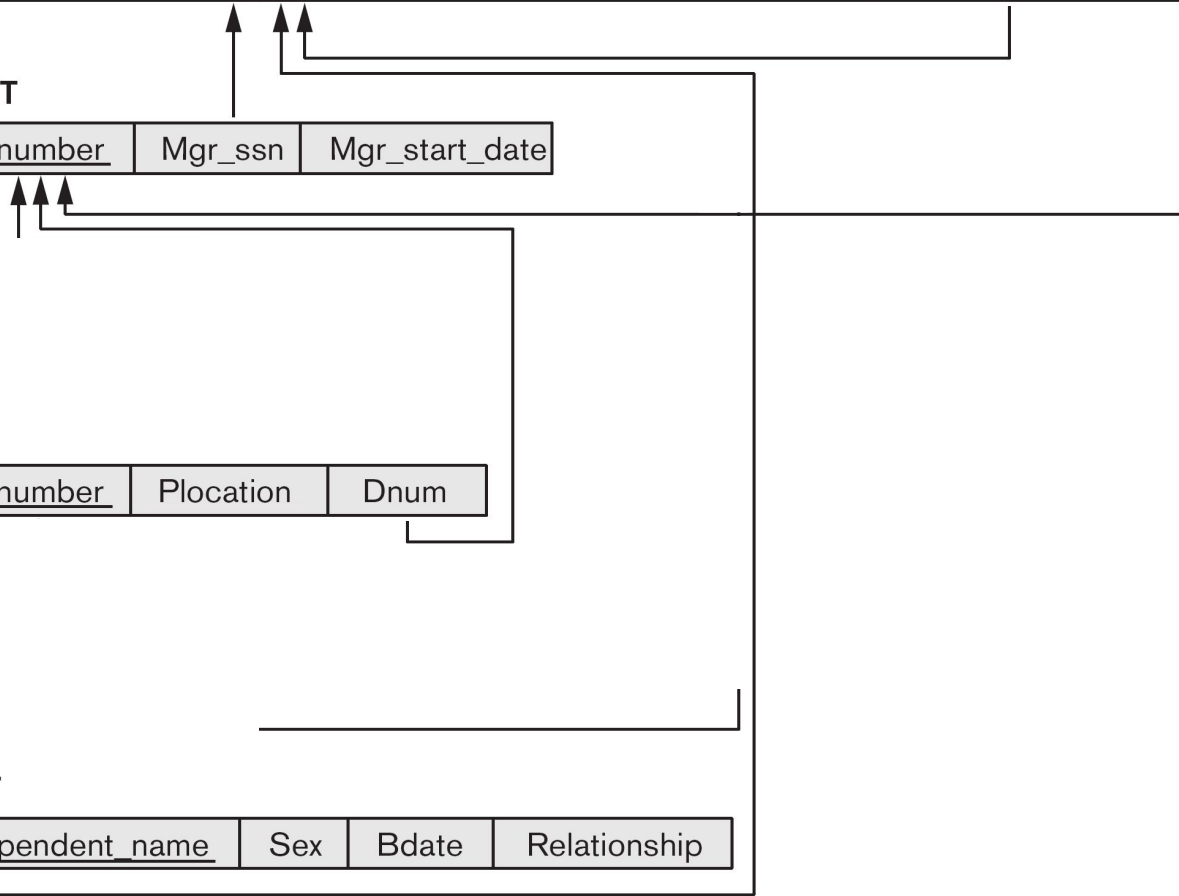
Dname	<u>Dnumber</u>	Mgr_ssn	Mgr_start_date
-------	----------------	---------	----------------

PROJECT

Pname	<u>Pnumber</u>	Plocation	Dnum
-------	----------------	-----------	------

DEPENDENT

<u>Essn</u>	<u>Dependent_name</u>	Sex	Bdate	Relationship
-------------	-----------------------	-----	-------	--------------

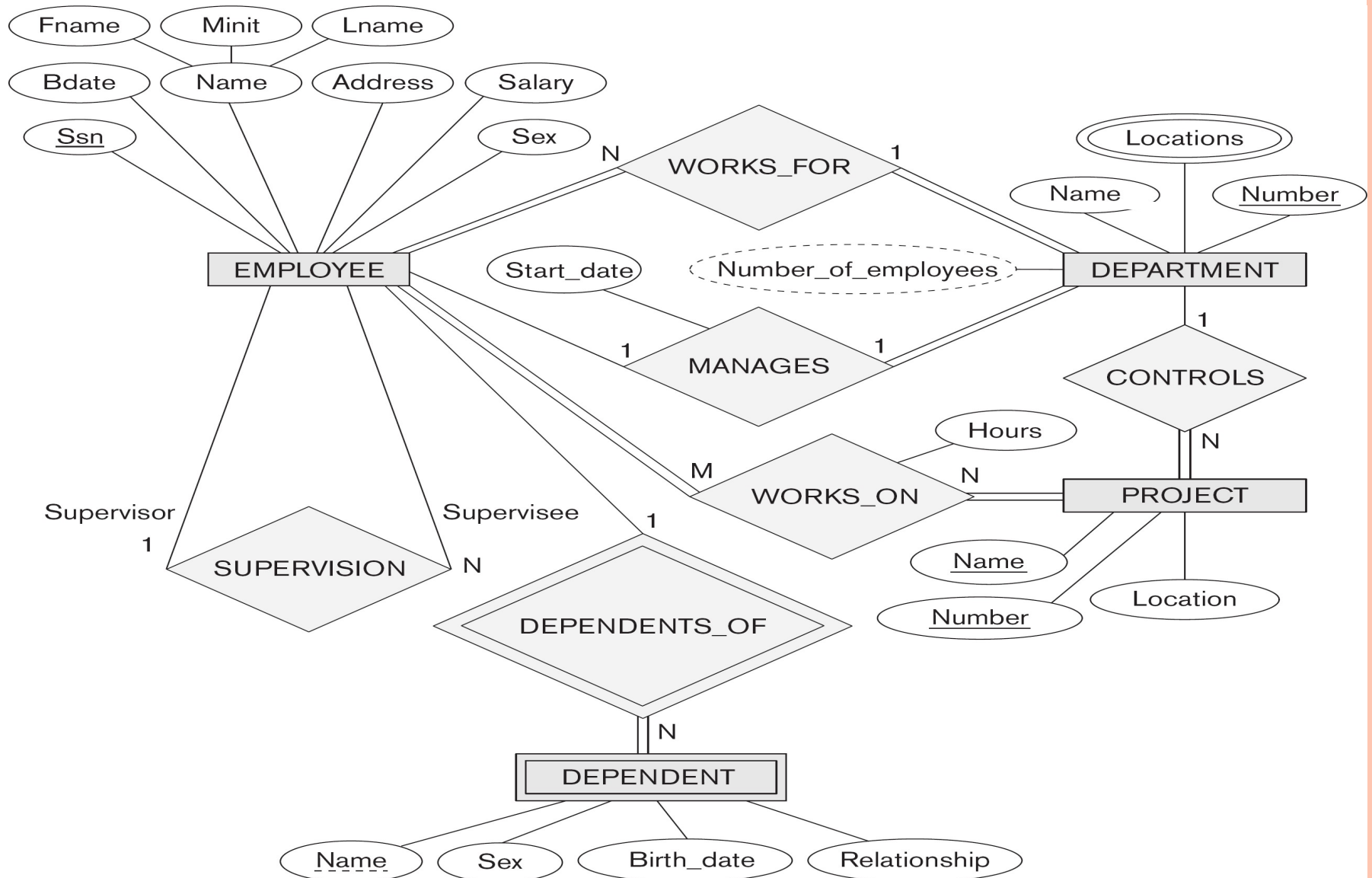


STEP 5: BINARY M-TO-N

- i. Create a new relation S (termed: *relationship relation*)
 - – In some ERD dialects, actually drawn in.
- ii. Add as foreign keys the primary keys of both relations; their combination forms the primary key of S
- iii. Add any simple attributes of the M:N relationship to S



EXAMPLE OF ER - DIAGRAM



STEP 5 RESULT

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
-------	-------	-------	------------	-------	---------	-----	--------	-----------	-----

DEPARTMENT

Dname	<u>Dnumber</u>	Mgr_ssn	Mgr_start_date
-------	----------------	---------	----------------

PROJECT

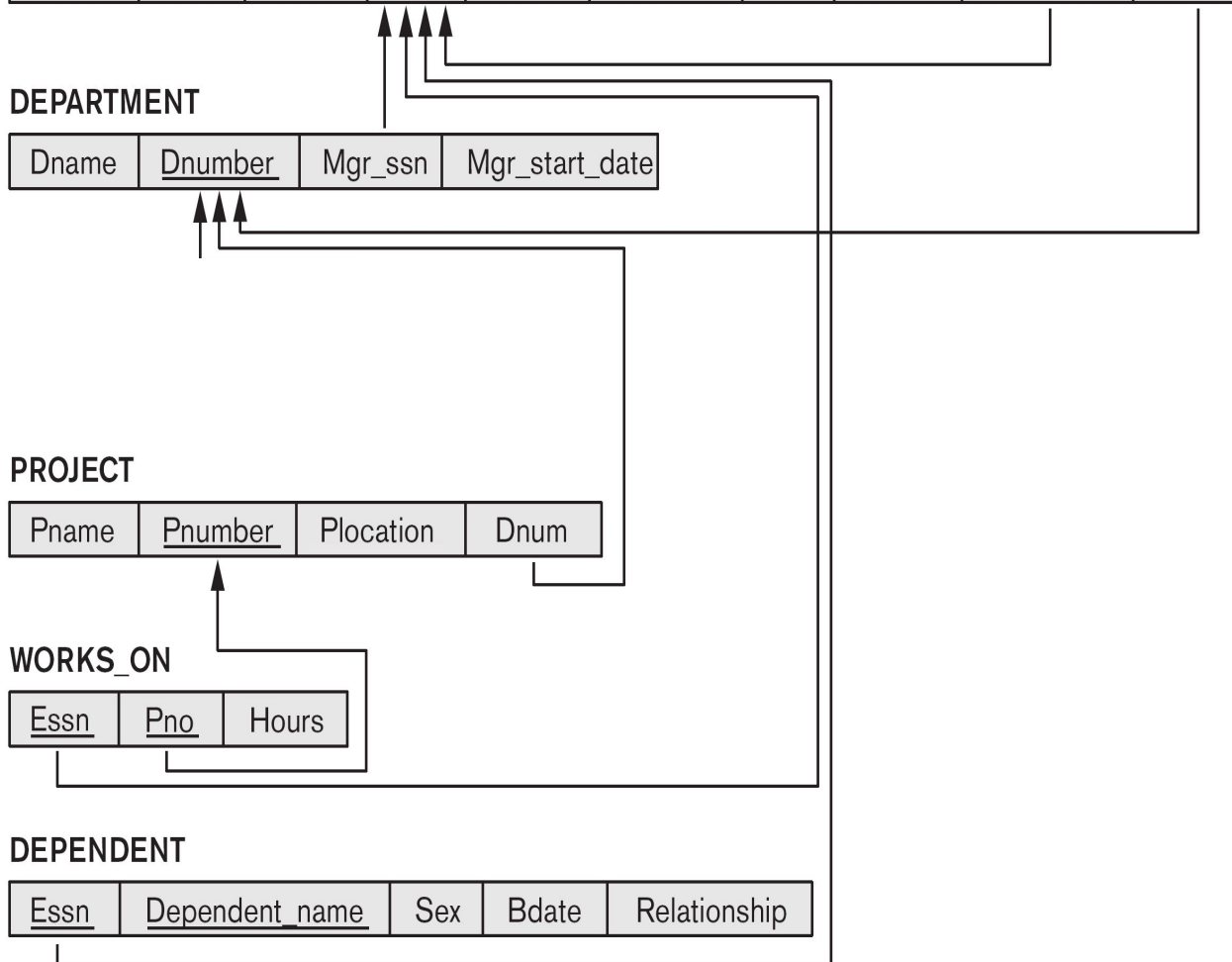
Pname	<u>Pnumber</u>	Plocation	Dnum
-------	----------------	-----------	------

WORKS_ON

<u>Essn</u>	<u>Pno</u>	Hours
-------------	------------	-------

DEPENDENT

<u>Essn</u>	<u>Dependent_name</u>	Sex	Bdate	Relationship
-------------	-----------------------	-----	-------	--------------

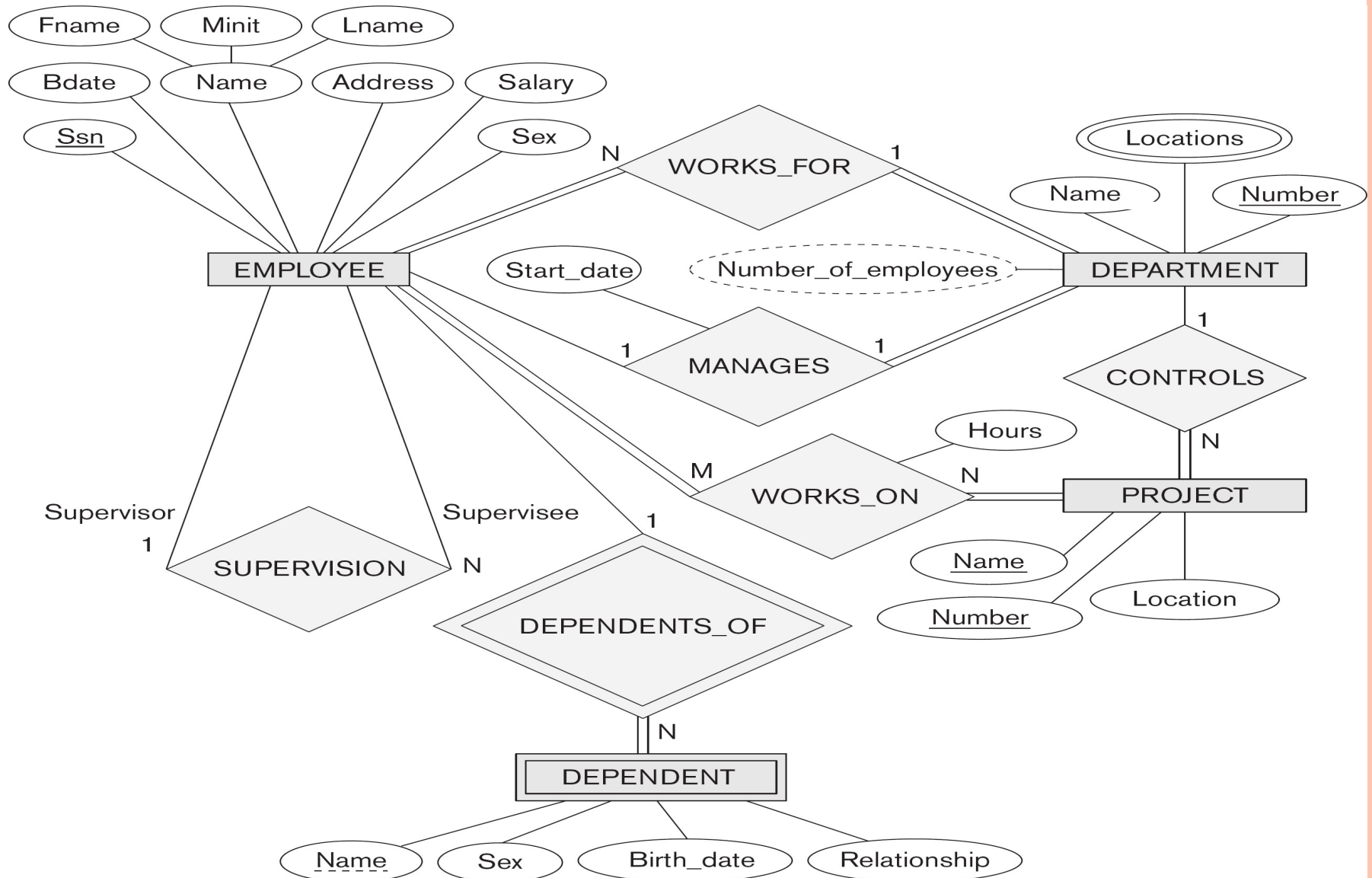


STEP 6: MULTIVALUED ATTRIBUTES

- i. Create a new relation $S(location)$
- ii. Add as foreign keys the primary keys of the corresponding relation
- iii. Add the attribute to $S(location)$ (if composite, the simple attributes); the combination of all attributes in $S(location)$ forms the primary key



EXAMPLE OF ER - DIAGRAM



STEP 6 RESULT

EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
-------	-------	-------	------------	-------	---------	-----	--------	-----------	-----

DEPARTMENT

Dname	<u>Dnumber</u>	Mgr_ssn	Mgr_start_date
-------	----------------	---------	----------------

DEPT_LOCATIONS

<u>Dnumber</u>	<u>Dlocation</u>
----------------	------------------

PROJECT

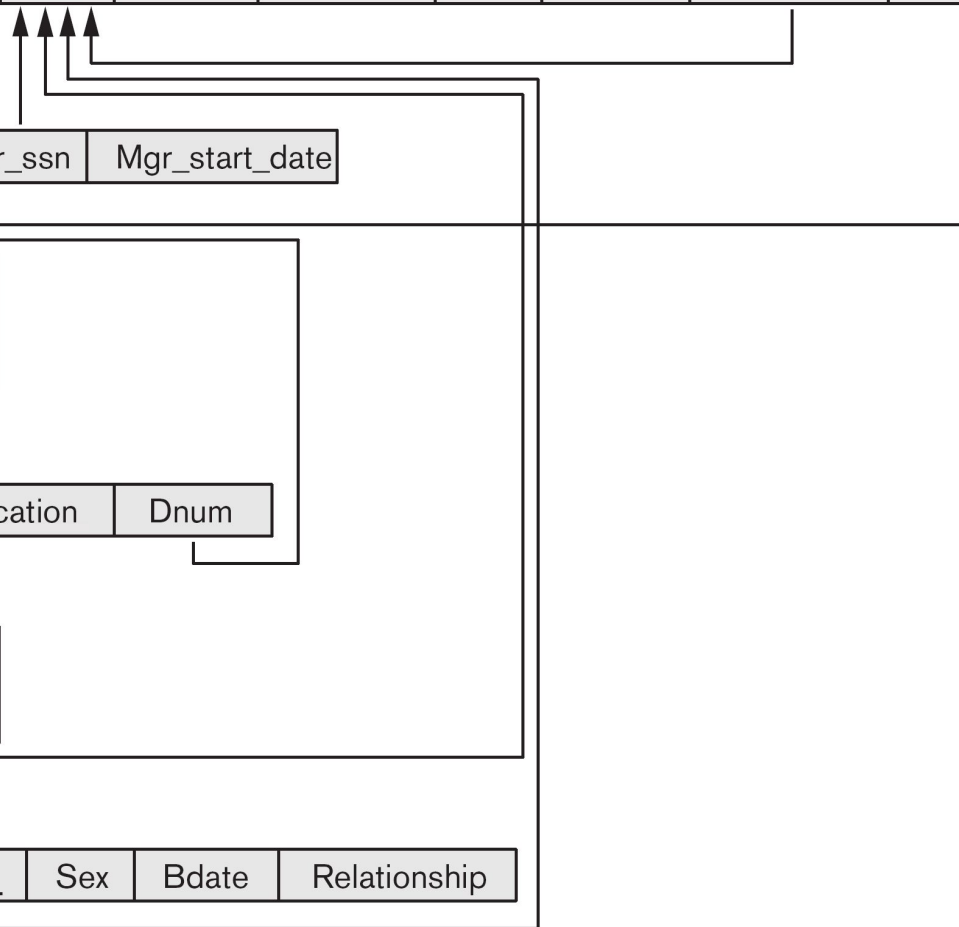
Pname	<u>Pnumber</u>	Plocation	Dnum
-------	----------------	-----------	------

WORKS_ON

<u>Essn</u>	<u>Pno</u>	Hours
-------------	------------	-------

DEPENDENT

<u>Essn</u>	<u>Dependent_name</u>	Sex	Bdate	Relationship
-------------	-----------------------	-----	-------	--------------



STEP 7: SPECIALIZATION/GENERALIZATION (AS OF NOW IGNORE)

- A. Multiple relations – subclass and superclass
 - Usually works (assumes unique id at parent)
- B. Multiple relations – subclass only
 - Should only be used for disjoint
- C. Single relation with one type attribute
 - Only for disjoint, can result in many NULLs
- D. Single relation with multiple type attributes
 - Better for overlapping, could be disjoint



SUMMARY

- Mapping from ERDs to relations is an algorithmic process
- Some choice points involve comparing time-space tradeoffs (more in physical design)
- The ER diagram may change from people to people depending on their understanding.

