Information Systems 3

ISY300B

Cape Peninsula
University of Technology

Entity Relationship (ER) Modelling

ER Introduction

- ER diagrams revised
- Introduction to advanced ER
  - Specialization
  - Aggregation
  - Generalization

ER diagrams revised

- ERM forms the basis of the ERD
- ERD depicts the databases main components
  - Entities
  - Attributes
  - Relationships

Entities

- Object of interest to the end user
- At ERM level the entity refers to the entity set (Table)
- In UML notation divided into three parts
  - The top (entity name)
  - The middle (description of the attributes)
  - The bottom (list of methods)

Attributes

- These are characteristics of entities
  - E.g.: Supplier entity includes: SID, SNAME, SADDRESS
  - Domains
  - Identifiers
  - Composite Key (Uses more than one attribute)
  - Composite and Simple Attributes
  - Single-Valued Attributes
  - Multivalued Attributes

Relationships

- This is an association between two entities
  - (Also referred to a participant)

Multiplicity

- This refers to the number of instances one entity that are associated with one instance of a related entity.

Multiplicities

Existence Dependence / Independence

- An entity is said to be existence- dependent if it can exist in the database only when it is associated with another related entity occurrence.
  - (Consists of a mandatory foreign key that cannot be null)
- When an entity can exist without being associated to another entity it is said to be existence-independent.

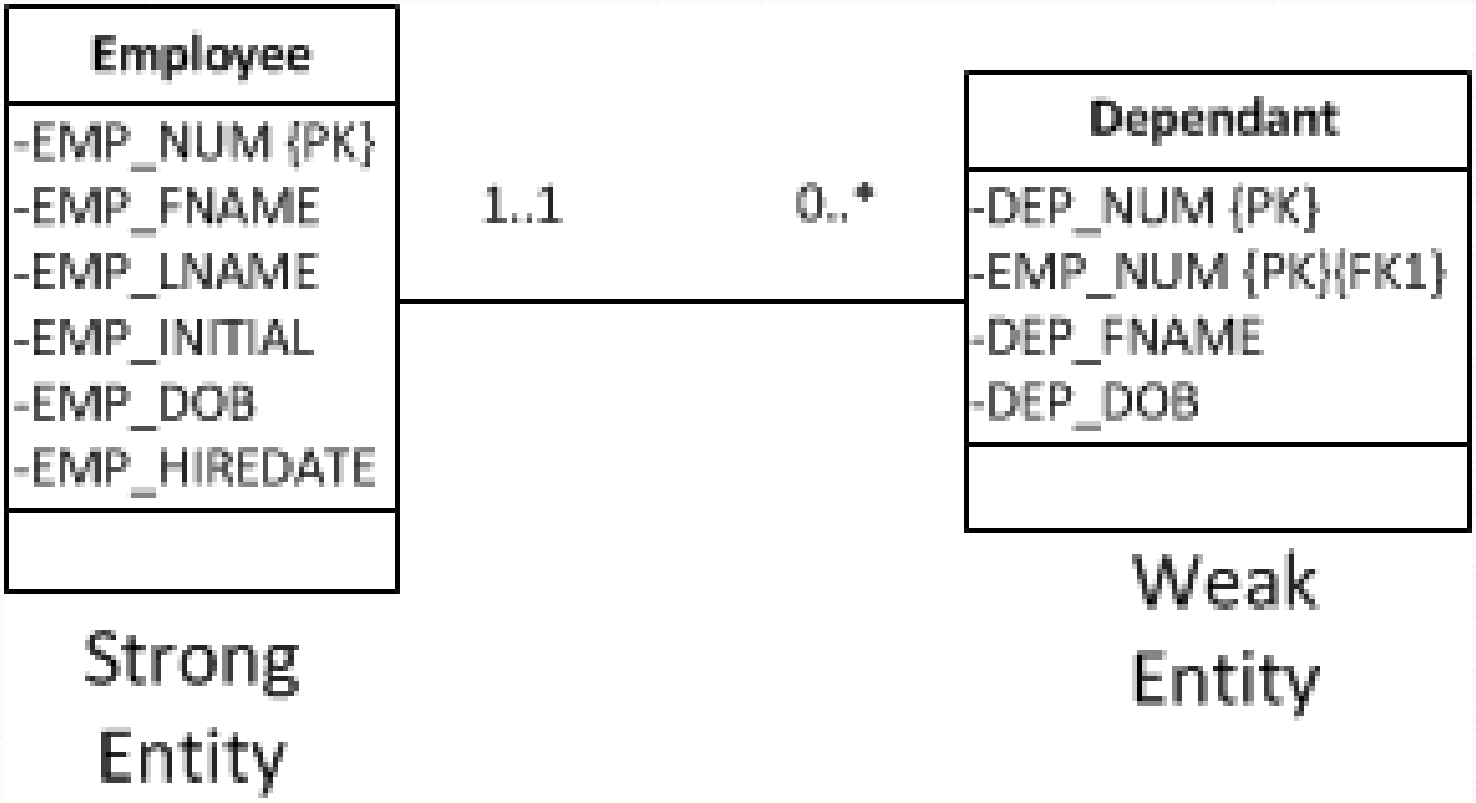
Relationship Strength

- Based on the how the primary key of a related entity is defined.
- Weak (Non Identifying Relationships) (pg. 167)
  - PK of the related entity does not contain the PK of the parent entity.
  - PK will the appear as a FK of the related entity.
- Strong(Identifying) Relationships (pg. 169)
  - PK of the related entity contains the PK of the parent entity.

Weak Entities

- It is existence-dependent (It cannot exist without the entity with which it has a relationship)
- It has a primary key that is partially or totally derived from the parent entity in the relationship.

Weak Entities



Relationship Participation

- Optional or Mandatory Participation
  - Optional one entity occurrence does not require a corresponding entity occurrence
  - Mandatory – an entity occurrence requires a corresponding entity occurrence in a particular relationship.

Relationship Degree

- This indicates the number of entities or participants associated with a relationship.
  - Unary relationship exists when an association is maintained with a single entity
  - Binary relationship exists when two entities are associated
  - A ternary relationship exists when three entities are associated

Recursive Relationship

- As was mentioned a recursive relationship is one in which a relationship can exist between occurrences of the same entity set.

Developing and ER Diagram

- Create a detailed narrative of the organisations description of operations
- Identify the business rules based on the descriptions of operations
- Identify all main entities from the business rules
- Identify all main relationships between entities from the business rules
- Develop and initial ERD

Developing and ER Diagram

- Determine the multiplicities and the participation of all relationships
- Identify the primary and foreign keys
- Identify all attributes
- Revise and review the ERD

Advanced Data Modeling

Cape Peninsula
University of Technology

Extended Entity Relationship Model

- Entity supertypes and entity subtypes
  - Supertypes
    - Entity types that has a relationship with one or more subtypes and it contains attributes that common to its subtypes.
  - Subtypes
    - These are subgroups of the supertype entity and have unique attributes, but they will be different from each subtype.

Specialisation Hierarchy

- The specialization hierarchy depicts the arrangement of higher-level entity supertypes (parent entities) and lower-level entity subtypes (child entities)
- Sometimes describes as an “IS-A” relationships

Inheritance

- This enables a subtype entity to inherit the attributes and relationships of the supertype.

Subtype Discriminator

- A subtype discriminator is the attribute in the supertype entity that determines to which entity subtype each occurrence is related. (Figure 6.2)

Disjoint and Overlapping constraints

- Disjoint subtypes (Non-overlapping subtypes) contain a unique subset of the supertype entity set
- Overlapping subtypes (non-disjoint subtypes) are subtypes that contain non-unique subsets of a supertype entity set.

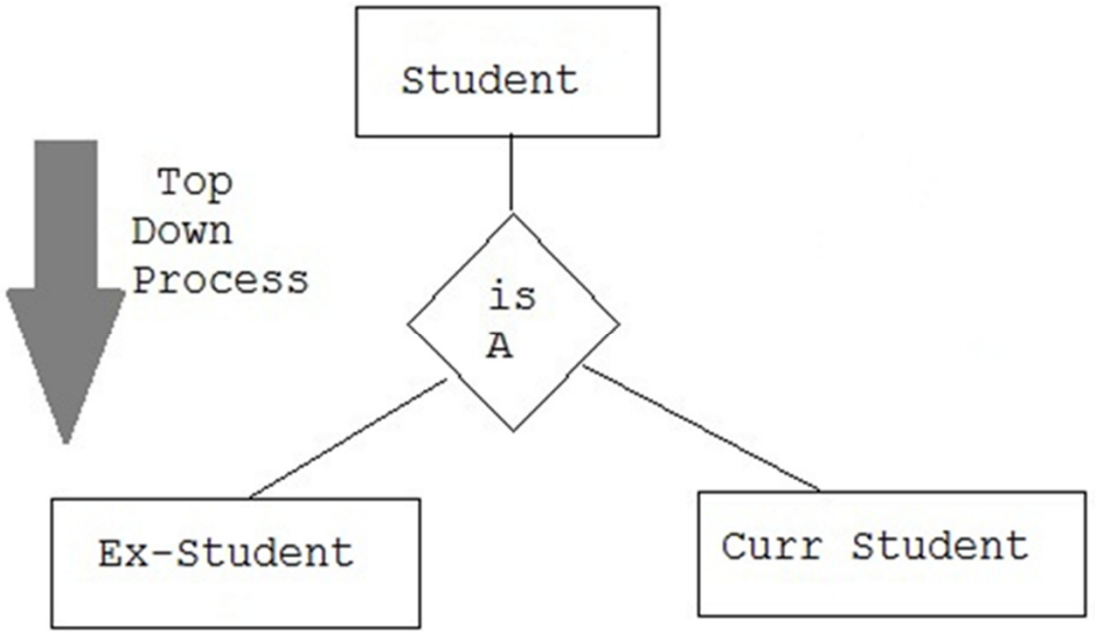
Completeness Constraint

- This type of constraint specifies whether each entity supertype occurrence must also be a member of at least one subtype.
  - **Partial completeness** – means that not every supertype is a member of a subtype
  - **Total completeness** – means that every supertype is a member of a subtype

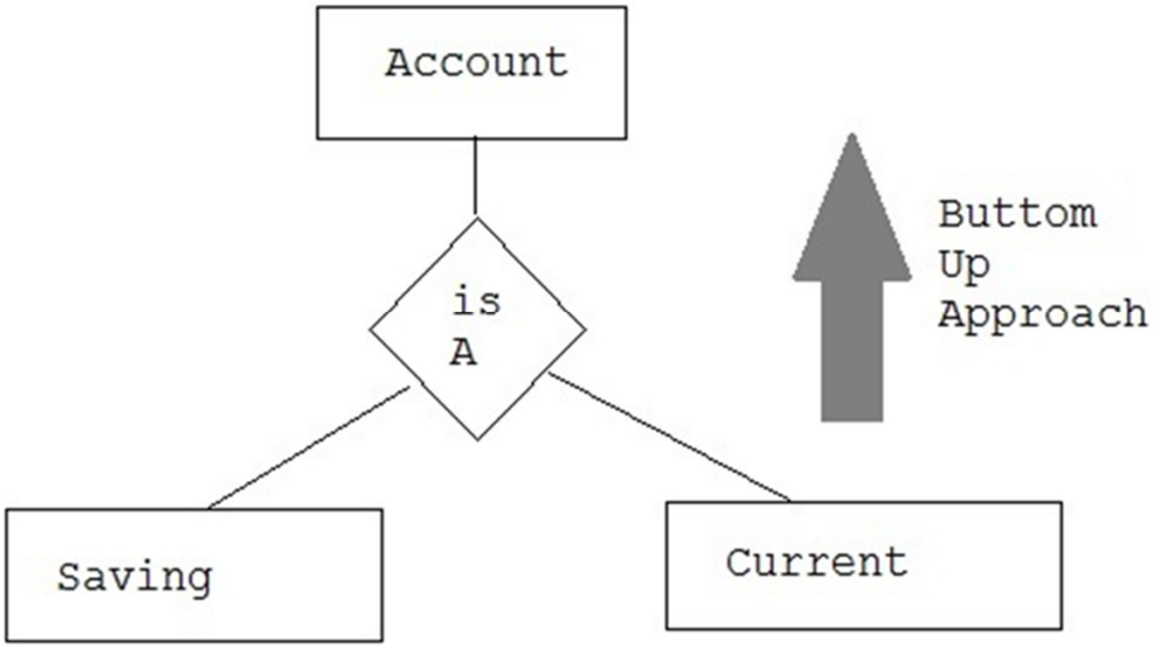
Specialization and Generalization

- Specialization is a top-down process of identifying lower level , more specific entity subtypes from a higher –level entity supertype.
- Generalization is the bottom-up process of identifying a higher level more generic entity supertype from the lower level entity subtypes.

Specialization



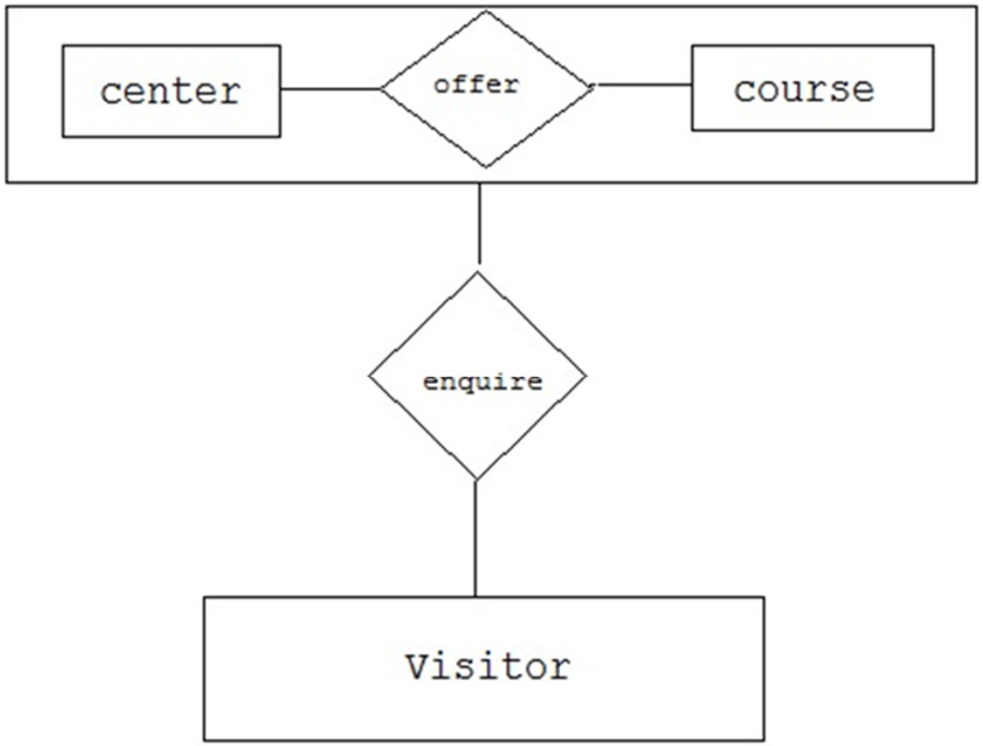
Generalization



Composition and Aggregation

- Aggregation – is used when an entity is composed of a collection of other entities. The relationship can be classified as a “has_a” relationship type.
  - Example:
    - A team has many players
    - A band has many musicians

Aggregation



Entity Clustering

- This is creating a virtual entity type, that represent multiple interrelated entities.

Entity Integrity: Selecting primary keys

- Natural Keys
  - These are real-world, generally accepted identifier used to distinguish-this is, uniquely identify-real-world objects
    - Example:
      - Class numbers to register classes, invoice numbers to identify specific invoices.
-

Primary Key Guidelines

- understand the function of the primary key.
  - this is to uniquely identify and entity instance or row with in a table.
  - primary keys and foreign keys are used to implement relationship amongst entities.

Primary Key Guidelines

- Unique Values
- Non intelligent
- No change over time
- Preferable single attribute
- Preferably numeric
- Security Compliant

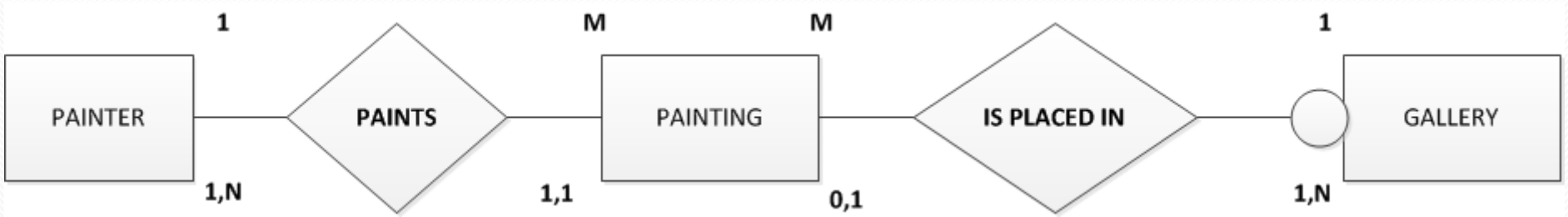
Composite and Surrogate Keys

- **Composite key**
  - A key that is composed of two or more attributes.
    - Used as identifiers of composite entities, where each primary key combination is allowed only once in a *:* relationship
    - As identifiers of weak entities, where the weak entity has a strong relationship with the primary entity.
- **Surrogate key**
  - A key with no business meaning.
    - This does not exist in the real world (in some instances)

Converting and advanced ER model into a database structure

Cape Peninsula
University of Technology

Converting and advanced ER model into a database structure



TABLES:
PAINTER, PAINTING, GALLERY

Converting and advanced ER model into a database structure

ARTIST

PTR_NUM	PTR_LASTNAME	PTR_FIRSTNAME	PTR_INITIAL	PTR_AREACODE	PTR_PHONE
1	Engel	Gino	G	031	123-4567
2	Ross	Celeste	C	011	234-5678
3	Davids	Robin	R	021	345-6789

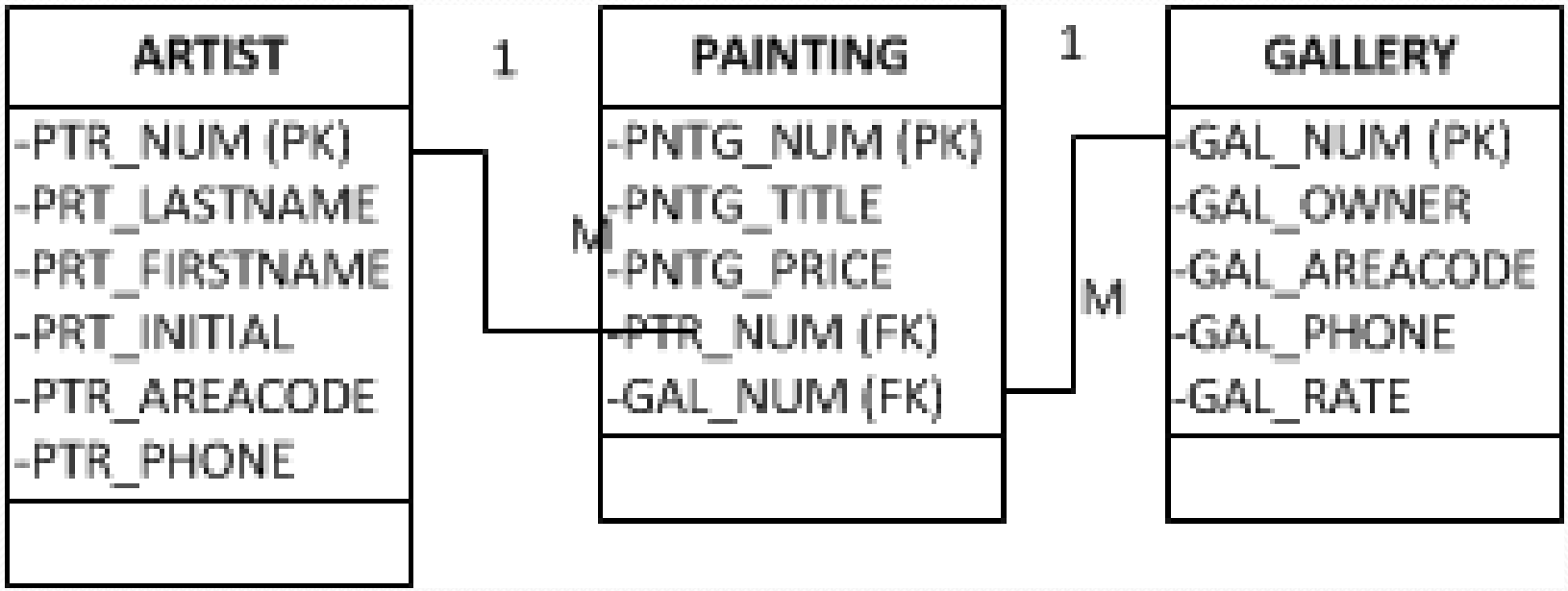
PAINTING

PNTG_NUM	PNTG_TITLE	PNTG_PRICE	PTR_NUM	GAL_NUM
112	Maximum	5050	1	9
113	Founders	6000	2	
114	Plastic Paradise	8450	3	8

GALLERY

GAL_NUM	GAL_OWNER	GAL_AREACODE	GAL_PHONE	GAL_RATE
8	L.Z.Daniels	031	112-4567	0.35
9	Dr. V.Steyn	011	113-9875	0.28

Converting and advanced ER model into a database structure



Converting and advanced ER model into a database structure

SQL Commands to Create the PAINTER Table

```
CREATE TABLE PAINTER (
PTR_NUM          CHAR(4)      NOT NULL    UNIQUE,
PRT_LASTNAME     CHAR(15)     NOT NULL,
PTR_FIRSTNAME    CHAR(15),
PTR_INITIAL      CHAR(1),
PTR_AREACODE     CHAR(3),
PTR_PHONE        CHAR(8),
PRIMARY KEY(PTR_NUM));
```

Converting and advanced ER model into a database structure

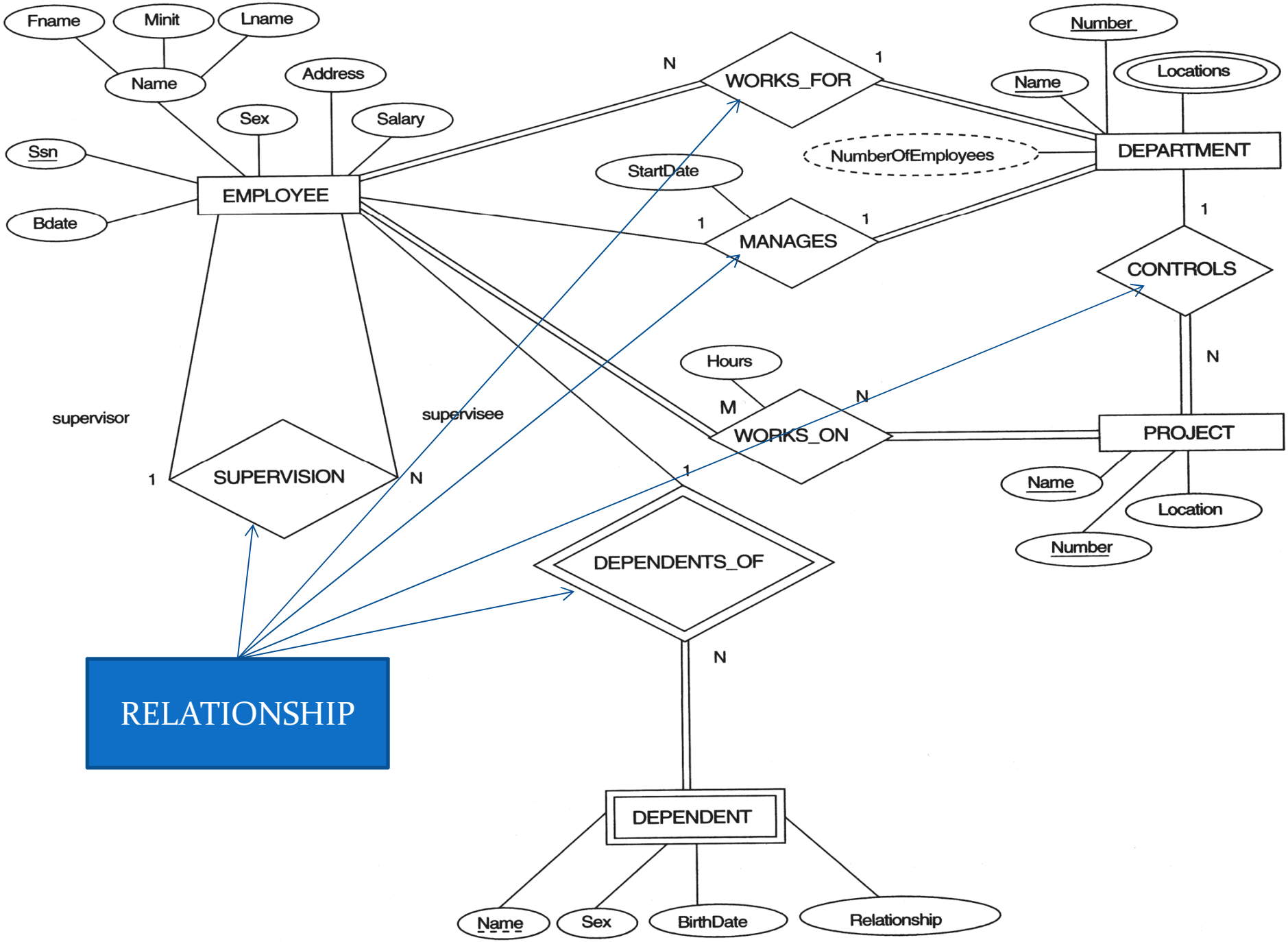
SQL Commands to Create the GALLERY Table

```
CREATE TABLE GALLERY (
GAL_NUM          CHAR(4)        NOT NULL    UNIQUE,
GAL_OWNER        CHAR(35),
GAL_AREACODE     CHAR(3)        NOT NULL,
GAL_PHONE        CHAR(8)        NOT NULL,
                 GAL_RATE       NUMBER(4,2),
PRIMARY KEY(GAL_NUM));
```

Converting and advanced ER model into a database structure

SQL Commands to Create the PAINTING Table

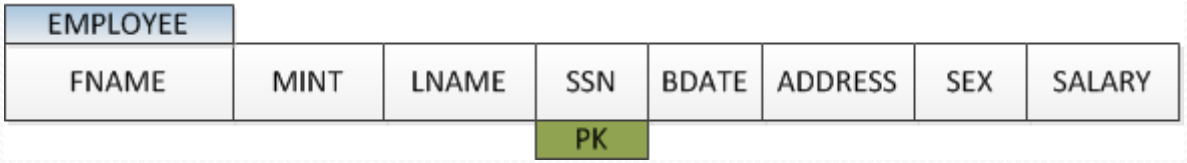
```
CREATE TABLE PAINTING (
PNTG_NUM            CHAR(4)          NOT NULL      UNIQUE,
PNTG_TITLE          CHAR(35),
PNTG_PRICE          NUMBER(9,2),
PTR_NUM             CHAR(4)          NOT NULL,
GAL_NUM             CHAR(4),
PRIMARY KEY(PNTG_NUM)
FOREIGN KEY(PTR_NUM) RERERENCES PAINTER
                    ON DELETE RESTRICT
                    ON UPDATE CASCADE,
FOREIGN KEY(GAL_NUM) REFERENCES GALLERY
                    ON DELETE RESTRICT
                    ON UPDATE CASCADE);
```

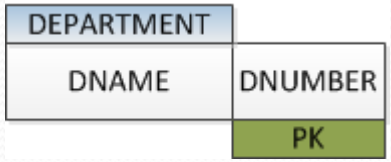

Fname
Minit
Lname
Name
Address
Sex
Salary
Ssn
Bdate
EMPLOYEE
N
WORKS_FOR
1
Number
Name
Locations
DEPARTMENT
NumberOfEmployees
StartDate
1
MANAGES
1
1
CONTROLS
N
Hours
M
N
WORKS_ON
PROJECT
supervisor
supervisee
1
SUPERVISION
N
1
Name
Location
Number
DEPENDENTS_OF
N
RELATIONSHIP
DEPENDENT
Name
Sex
BirthDate
Relationship

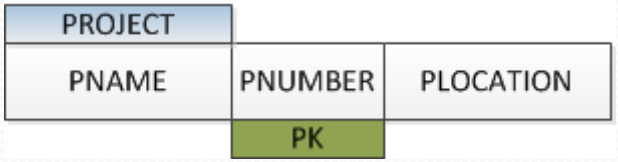
Converting and advanced ER model into a database structure – STEP 1

- Step 1: Mapping of Regular Entity Types
  - Create a table for every regular entity and include all simple attributes
  - Leave out multivalued attributes (Step 6)
  - Pick a Primary Key
  - Do not include Foreign Keys

Converting and advanced ER model into a database structure – STEP 1







Converting and advanced ER model into a database structure – STEP 2

- Step 2: Mapping of Weak Entities
  - Create a table for each weak entity and include all simple attributes.
  - Create foreign keys of the primary key attributes that corresponds to the owner entity.
  - The primary key for the weak entity will be a combinations of and a partial key of the weak entity.

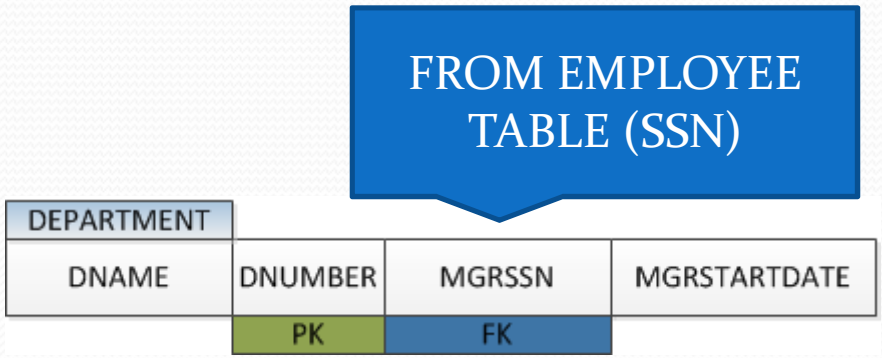
Converting and advanced ER model into a database structure – STEP 2

FROM EMPLOYEE TABLE (SSN)

Converting and advanced ER model into a database structure – STEP 3

- Step 3: Mapping of 1:1 Relation Types
  - Foreign Key approach:
    - Include the primary key of one relation as a foreign key in the other
    - Include attributes of the relationship with the foreign key
  - Merged Relationship
    - Merge the two relations
  - Cross Reference
    - Add a new table with the keys from both entities (see m:n for example of this)

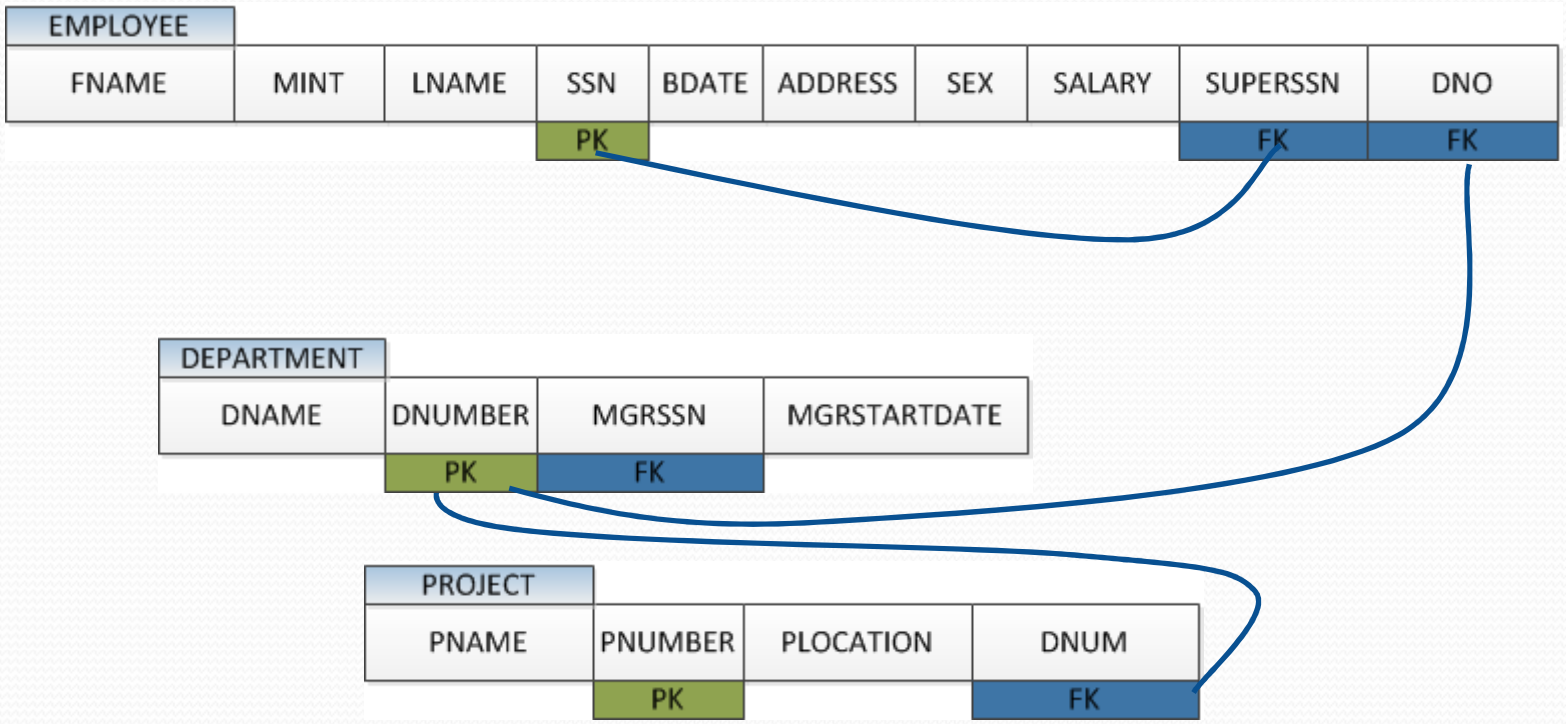
Converting and advanced ER model into a database structure – STEP 3



Converting and advanced ER model into a database structure – STEP 4

- Step 4: Mapping of Binary 1:N Relationships
  - Let R be the "1 side", S the "n side"
    - Include R's key as a foreign key in S
    - Include any attributes of the relation together with the foreign key in S
  - Or use the Cross Reference table approach (m:n approach)

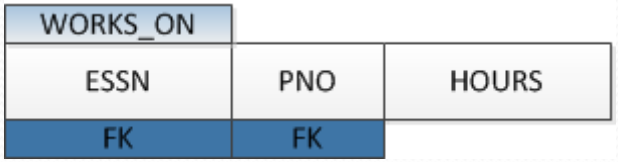
Converting and advanced ER model into a database structure – STEP 4



Converting and advanced ER model into a database structure – STEP5

- Step 5: Mapping of Binary M:N Relationship Types
  - Create a new relation to represent the relationship
  - Include the keys of the related entities as foreign keys in the relation
  - Make the key of the relation the combined keys of the participating entities
  - Include any simple attributes of the relationship as attributes of the relation
  - Include CASCADE for UPDATE and DELETE (typical)

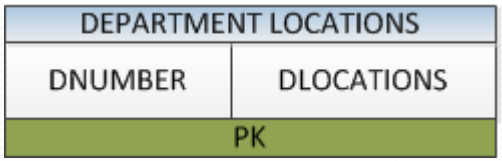
Converting and advanced ER model into a database structure – STEP5



Converting and advanced ER model into a database structure – STEP 6

- Step 6: Convert a multi-valued attribute into a relation with composite primary key consisting of the attribute value plus the primary key of the attribute's entity
  - Identify the corresponding relation (entity/table) .
  - Create a new relation (entity/table) representing the attribute.

Converting and advanced ER model into a database structure – STEP 6

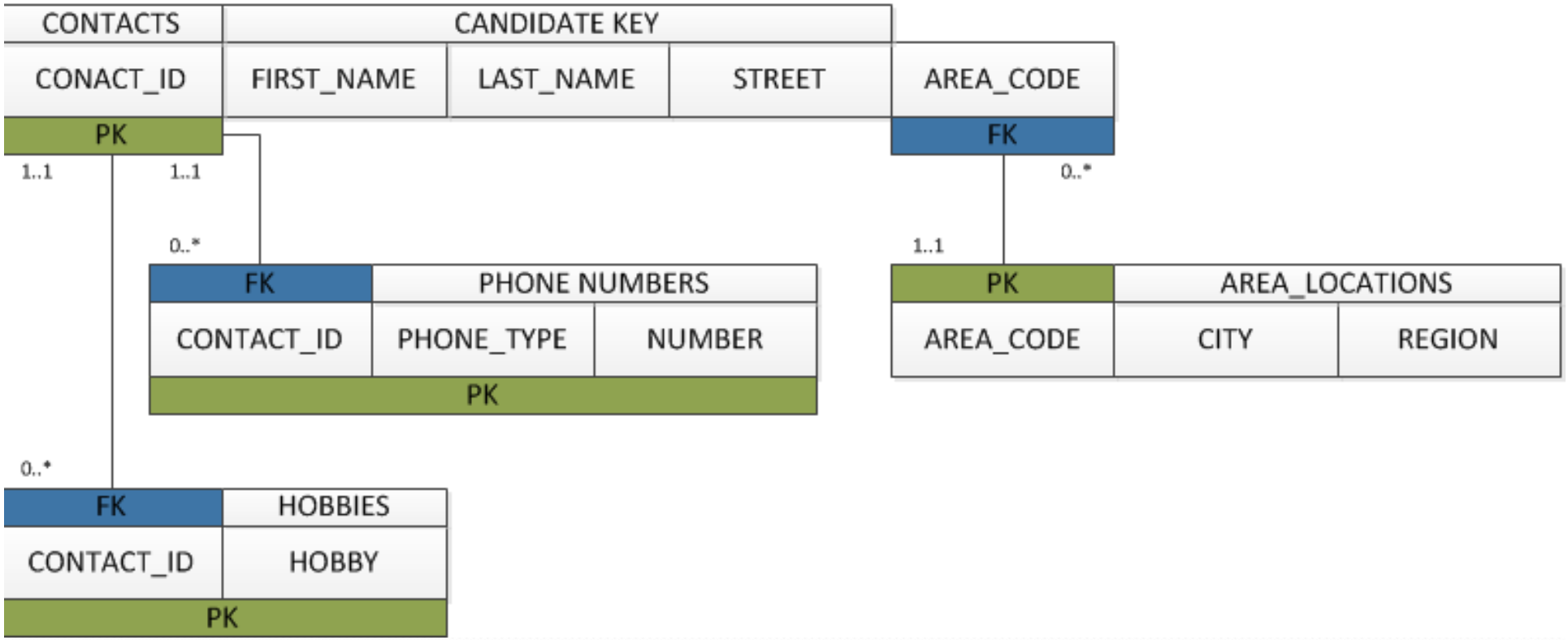


Converting and advanced ER model into a database structure – STEP 6

contactid	firstname	lastname	hobbies
1639	George	Barnes	reading
5629	Susan	Noble	hiking, movies
3388	Erwin	Star	hockey, skiing
5772	Alice	Buck	
1911	Frank	Borders	photography, travel, art
4848	Hanna	Diedrich	gourmet cooking

EXAMPLE 2

Converting and advanced ER model into a database structure – STEP 6

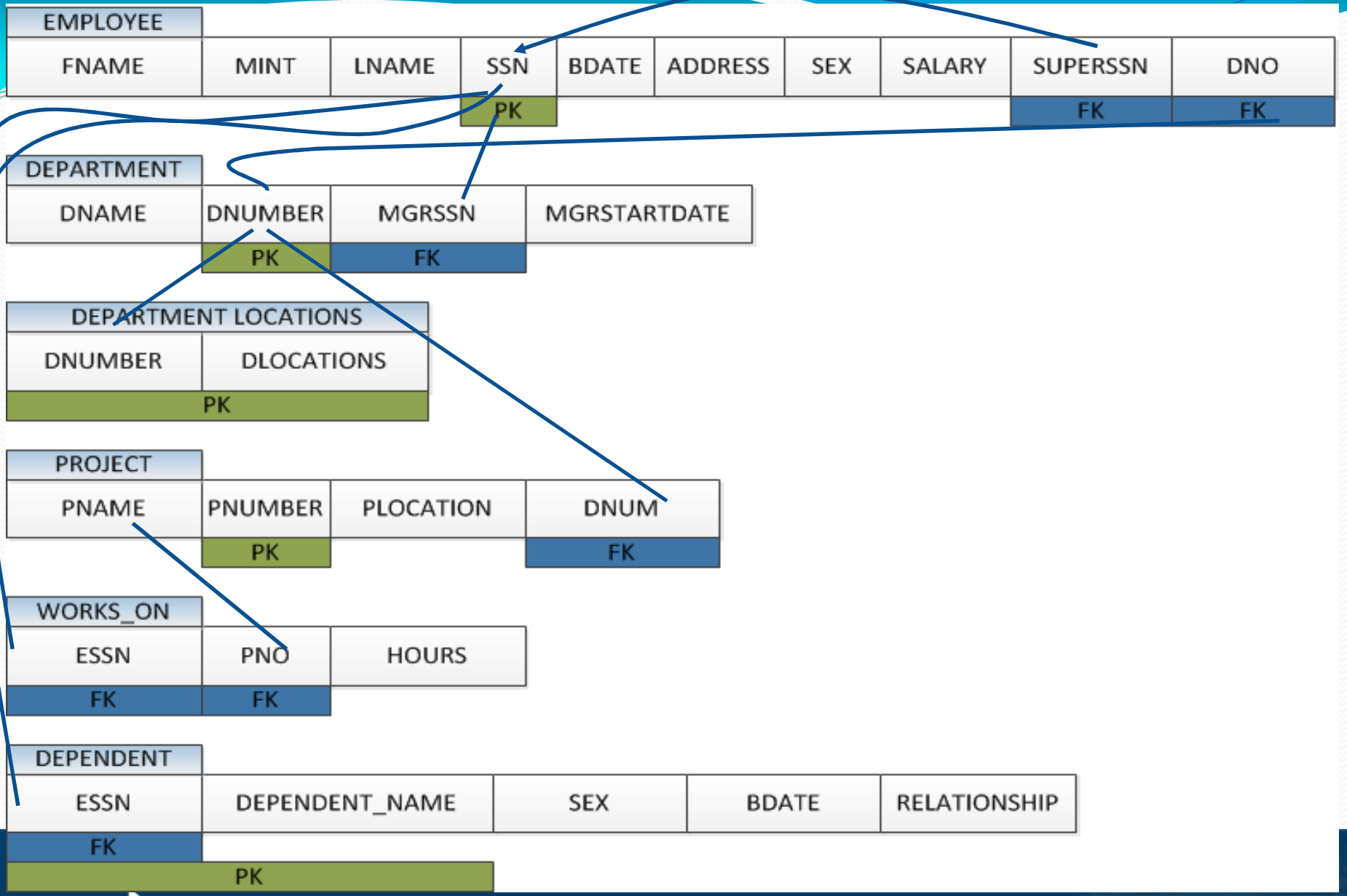


Converting and advanced ER model into a database structure – STEP 6

CONTACT_ID	HOBBY
1639	reading
5629	hiking
5629	movies
3388	hockey
3388	skiing
1911	photography
1911	travel
1911	art
4848	gourmet cooking

Converting and advanced ER model into a database structure

- Step 7: Convert *n*-ary relationships
  - Create a new relation to represent the relationship
  - Create foreign keys that reference the related entities.
  - Include any relation attributes


EMPLOYEE
FNAME
MINT
LNAME
SSN
BDATE
ADDRESS
SEX
SALARY
SUPERSSN
DNO
PK
FK
FK
DEPARTMENT
DNAME
DNUMBER
MGRSSN
MGRSTARTDATE
PK
FK
DEPARTMENT LOCATIONS
DNUMBER
DLOCATIONS
PK
PROJECT
PNAME
PNUMBER
PLOCATION
DNUM
PK
FK
WORKS_ON
ESSN
PNO
HOURS
FK
FK
DEPENDENT
ESSN
DEPENDENT_NAME
SEX
BDATE
RELATIONSHIP
FK
PK

References

- Learn Data Modeling. 2014. Supertypes and Subtypes. Pro Business Sytems LLC. [Online] Available at: http://www.learndatamodeling.com/dm_super_type.php. Accessed 2 February 2014.
- Agile Data. 2014. Choosing a Primary Key: Natural or Surrogate?. Abisoft Inc. [Online] Available at: http://www.agiledata.org/essays/keys.html#sthash.m8cLwwbX.dpuf. Accessed 2 February 2014
- Peter, R., Coronel, C. & Crockett, K. 2008. "Database Systems Design, Implementation and Management". Chapter 5 and 6