Chapter 5: ER-and-EER-to-Relational Mapping

Outline:

5.1 ER-to-Relational Mapping

- Step 1: Mapping of Regular Entity Sets.
- Step 2: Mapping of Weak Entity Sets
- Step 3: Mapping of Binary 1:1 Relationship Sets
- Step 4: Mapping of Binary 1:N Relationship Sets.
- Step 5: Mapping of Binary M:N Relationship Sets.
- Step 6: Mapping of Multivalued attributes.
- Step 7: Mapping of N-ary Relationship Sets.
- Summary of Mapping constructs and constraints

5.2 EER-to-Relational Mapping

- Step8: Options for Mapping Specialization or Generalization.
  - Option 8A: Multiple relations-Superclass and subclasses.
  - Option 8B: Multiple relations-Subclass relations only
  - Option 8C: Single relation with one type attribute.
  - Option 8D: Single relation with multiple type attributes.

5.1 ER-to-Relational Mapping

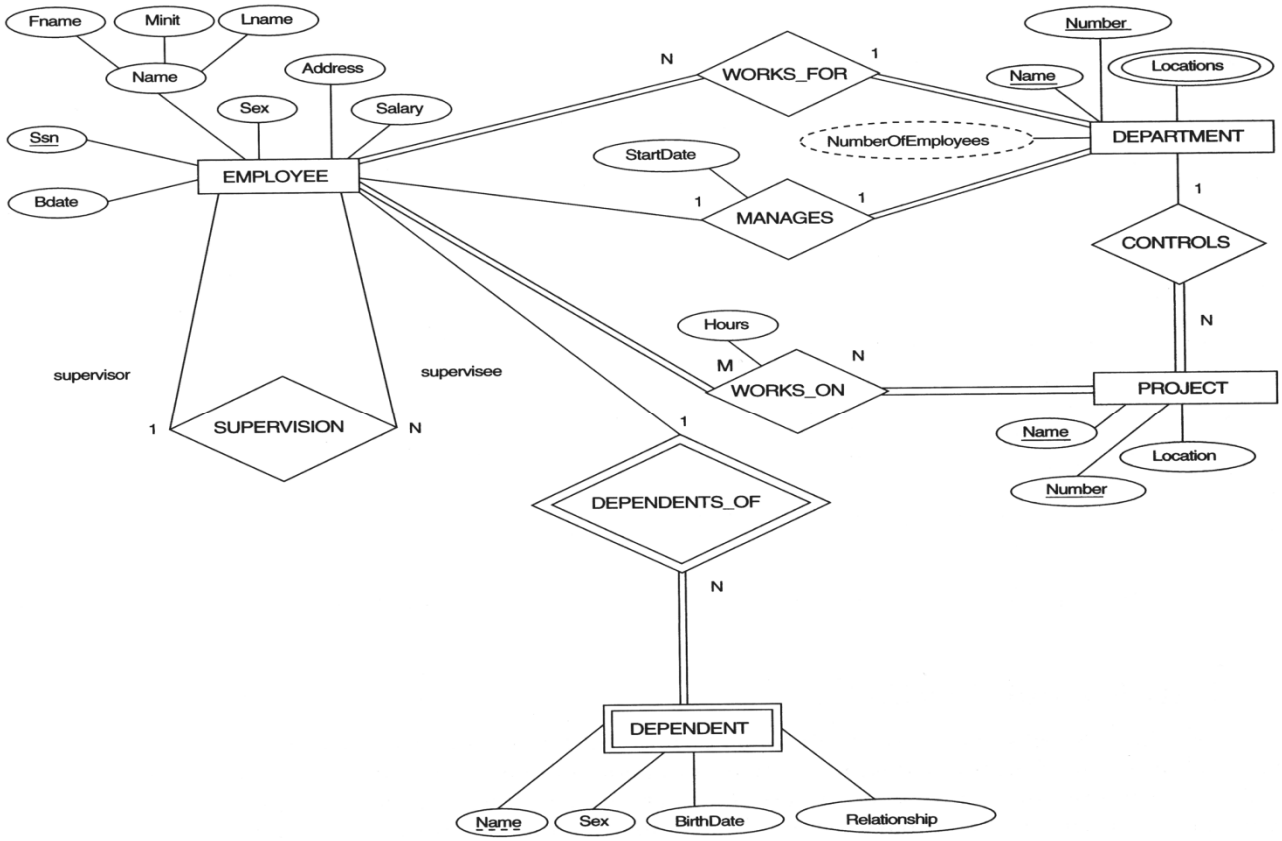


Figure 5.1: The ER conceptual schema diagram for the COMPANY database.

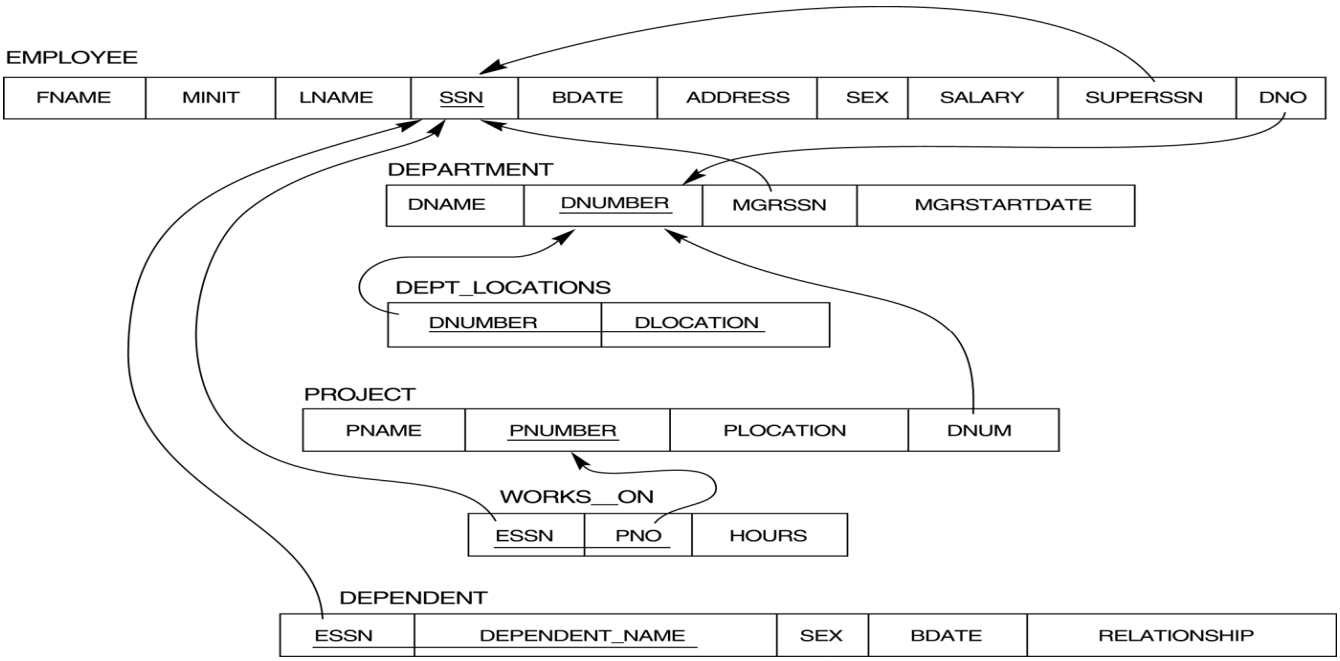


Figure 5.2: Result of mapping the COMPANY ER schema into a relational schema.

Step 1: Mapping of Regular Entity Sets.

- For each regular (strong) entity set E in the ER schema, create a relation R that includes all the simple attributes of E.
- Choose one of the key attributes of E as the primary key for R. If the chosen key of E is composite, the set of simple attributes that form it will together form the primary key of R.
- ***Example:*** We create the relations EMPLOYEE, DEPARTMENT, and PROJECT in the relational schema corresponding to the regular entities in the ER diagram. SSN, DNUMBER, and PNUMBER are the primary keys for the relations EMPLOYEE, DEPARTMENT, and PROJECT as shown.

Step 2: Mapping of Weak Entity Sets

- For each weak entity set W in the ER schema with owner entity set E, create a relation R and include all simple attributes (or simple components of composite attributes) of W as attributes of R.
- In addition, include as foreign key attributes of R the primary key attribute(s) of the relation(s) that correspond to the owner entity set(s).
- The primary key of R is the *combination* of the primary key(s) of the owner(s) and the partial key of the weak entity set W, if any.

- ***Example:*** Create the relation DEPENDENT in this step to correspond to the weak entity set DEPENDENT. Include the primary key SSN of the EMPLOYEE relation as a foreign key attribute of DEPENDENT (renamed to ESSN).
  The primary key of the DEPENDENT relation is the combination {ESSN, DEPENDENT_NAME} because DEPENDENT_NAME is the partial key of DEPENDENT.

- ***Note:*** It is common to choose the CASCADE option for the referential triggered action on the foreign key in the relation corresponding to the weak entity type, since a weak entity has an existence dependency on its owner entity. This can be used for both ON UPDATE and ON DELETE.

Step 3: Mapping of Binary 1:1 Relationship Sets

- For each binary 1:1 relationship set R in the ER schema, identify the relations S and T that correspond to the entity sets participating in R.
- Choose one of the relations-S, say-and include a foreign key in S that references the primary key of T. It is better to choose an entity set with *total participation* in R in the role of S.
- Include any simple attributes (or components of composite attributes) of the 1:1 relationship set as attributes of S.

- ***Example:*** 1:1 relation MANAGES is mapped by choosing the participating entity set DEPARTMENT to serve in the role of S, because its participation in the MANAGES relationship set is total.

- ***Note:*** an alternative mapping of a 1:1 relationship type is possible by merging the two entity types and the relationship into a single relation. This is appropriate when *both participations are total*.

Step 4: Mapping of Binary 1:N Relationship Sets.

- For each regular binary 1:N relationship set R, identify the relation S that represent the participating entity set at the N-side of the relationship set.
- Include as foreign key in S the primary key of the relation T that represents the other entity set participating in R.
- Include any simple attributes (or components of composite attributes) of the 1:N relationship Set as attributes of S.

- ***Example:*** 1:N relationship sets WORKS_FOR, CONTROLS, and SUPERVISION in the figure. For WORKS_FOR we include the primary key DNUMBER of the DEPARTMENT relation as foreign key in the EMPLOYEE relation and call it DNO.

Step 5: Mapping of Binary M:N Relationship Sets.

- For each regular binary M:N relationship set R, create a new relation S to represent R.
- Include as foreign key attributes in S the primary keys of the relations that represent the participating entity sets; their combination will form the primary key of S.
- Also include any simple attributes of the M:N relationship set (or simple components of composite attributes) as attributes of S.

- ***Example:*** The M:N relationship set WORKS_ON from the ER diagram is mapped by creating a relation WORKS_ON in the relational database schema. The primary keys of the PROJECT and EMPLOYEE relations are included as foreign keys in WORKS_ON and renamed PNO and ESSN, respectively.
  Attribute HOURS in WORKS_ON represents the HOURS attribute of the relation set. The primary key of the WORKS_ON relation is the combination of the foreign key attributes {ESSN, PNO}.

- ***Note:*** we can always map 1:1 or 1:N relationships in a manner similar to M:N relationships. This alternative is particularly useful when few relationship instances exist, in order to avoid null values in foreign keys. In this case, the primary key of the "relationship" relation will be only one of the foreign keys that reference the participating "entity" relations. For a 1:N relationship, this will be the foreign key that references the entity relation on the N-side. For a 1:1 relationship, the foreign key that references the entity relation with total participation (if any) is chosen as primary key.

Step 6: Mapping of Multivalued attributes.

- For each multivalued attribute A, create a new relation R. This relation R will include an attribute corresponding to A, plus the primary key attribute K-as a foreign key in R-of the relation that represents the entity set of relationship set that has A as an attribute.
- The primary key of R is the combination of A and K. If the multivalued attribute is composite, we include its simple components.

- ***Example:*** The relation DEPT_LOCATIONS is created. The attribute DLOCATION represents the multivalued attribute LOCATIONS of DEPARTMENT, while DNUMBER-as foreign key-represents the primary key of the DEPARTMENT relation. The primary key of R is the combination of {DNUMBER, DLOCATION}.

Step 7: Mapping of N-ary Relationship Sets.

- For each n-ary relationship set R, where n>2, create a new relation S to represent R.

- Include as foreign key attributes in S the primary keys of the relations that represent the participating entity sets.
- Also include any simple attributes of the n-ary relationship set (or simple components of composite attributes) as attributes of S.
- ***Example:*** The relationship set SUPPLY in the following ER diagram. This can be mapped to the relation SUPPLY shown in the relational schema, whose primary key is the combination of the three foreign keys {SNAME, PARTNO, PROJNAME}

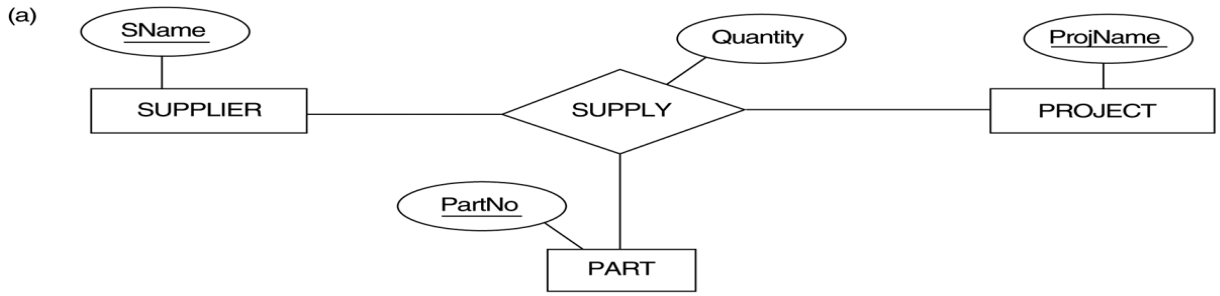


Figure 5.3: Ternary relationship sets. (a) The SUPPLY relationship.

SUPPLIER

SNAME	. . .

PROJECT

PROJNAME	. . .

PART

PARTNO	. . .

SUPPLY

SNAME	PROJNAME	PARTNO	QUANTITY

Figure 5.4: Mapping the n-ary relationship set SUPPLY from Figure 3.a.

Summary of Mapping constructs and constraints

ER Model	Relational Model
Entity set	"Entity" relation
1:1 or 1:N relationship set	Foreign key (or "relationship" relation)

M:N relationship set	"Relationship" relation and two foreign keys
n-ary relationship set	"Relationship" relation and n foreign keys
Simple attribute	Attribute
Composite attribute	Set of simple component attributes
Multivalued attribute	Relation and foreign key
Value set	Domain
Key attribute	Primary (or secondary) key

Table 5.1: Correspondence between ER and Relational Models

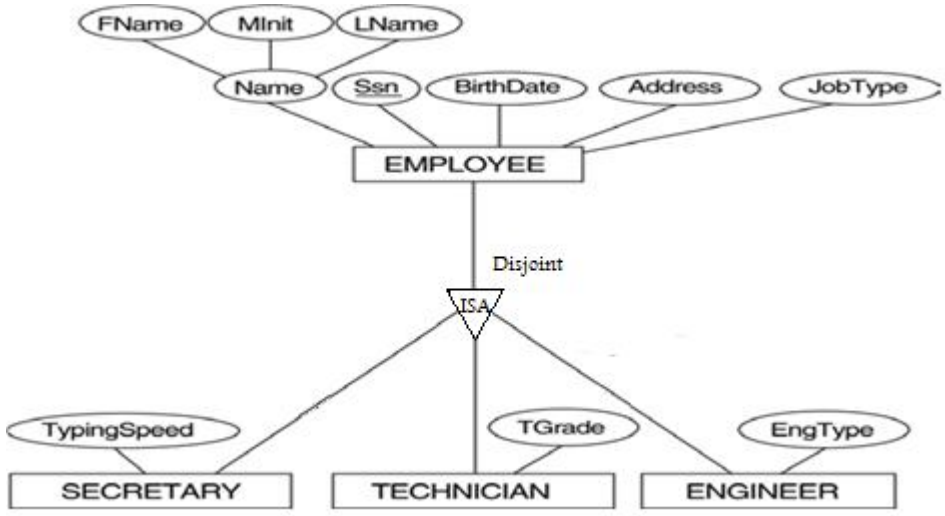
5.2 EER-to-Relational Mapping

Step8: Options for Mapping Specialization or Generalization.

- Convert each specialization with m subclasses {S1, S2,….,Sm} and generalized superclass C, where the attributes of C are {k,a1,…an} and k is the (primary) key, into relational schemas using one of the four following options:

  - ***Option 8A: Multiple relations-Superclass and subclasses.***

    - Create a relation L for C with attributes Attrs(L) = {k,a1,…an} and PK(L) = k. Create a relation Li for each subclass Si, 1 < i < m, with the attributesAttrs(Li) = {k} U {attributes of Si} and PK(Li)=k. This option works for any specialization (total or partial, disjoint of over-lapping).



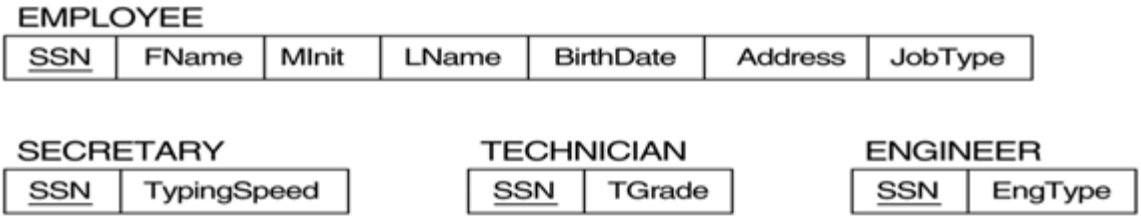
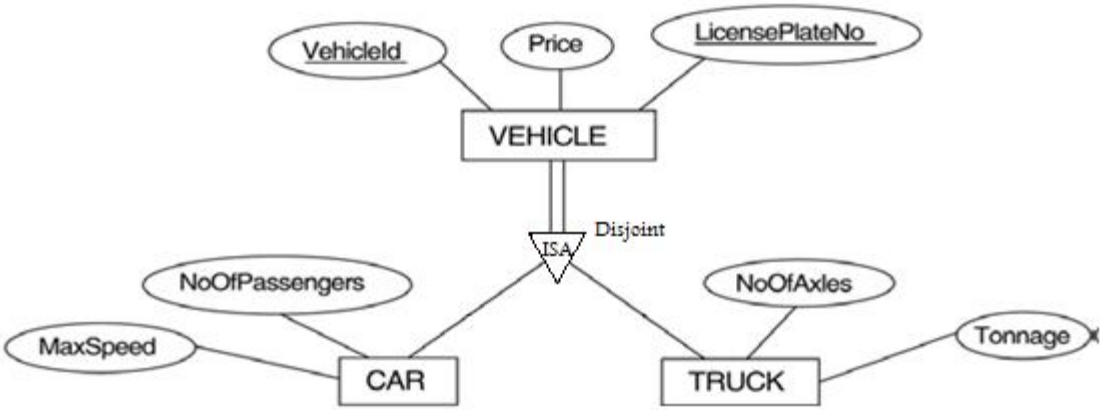


Figure 5.5: Option 8A Example.

  - ***Option 8B: Multiple relations-Subclass relations only***

- Create a relation Li for each subclass Si, 1 < i < m, with the attributes Attr(Li) = {attributes of Si} U {k,a1…,an} and PK(Li) = k. This option only works for a specialization whose subclasses are total (every entity in the superclass must belong to (at least) one of the subclasses).



CAR

VehicleId	LicensePlateNo	Price	MaxSpeed	NoOfPassengers

TRUCK

VehicleId	LicensePlateNo	Price	NoOfAxles	

Figure 5.6: Option 8B Example.

- ***Option 8C: Single relation with one type attribute.***

  - Create a single relation L with attributes Attrs(L) = {k,a1,…an} U {attributes of S1} U…U {attributes of Sm} U {t} and PK(L) = k. The attribute t is called a type (or discriminating) attribute that indicates the subclass to which each tuple belongs

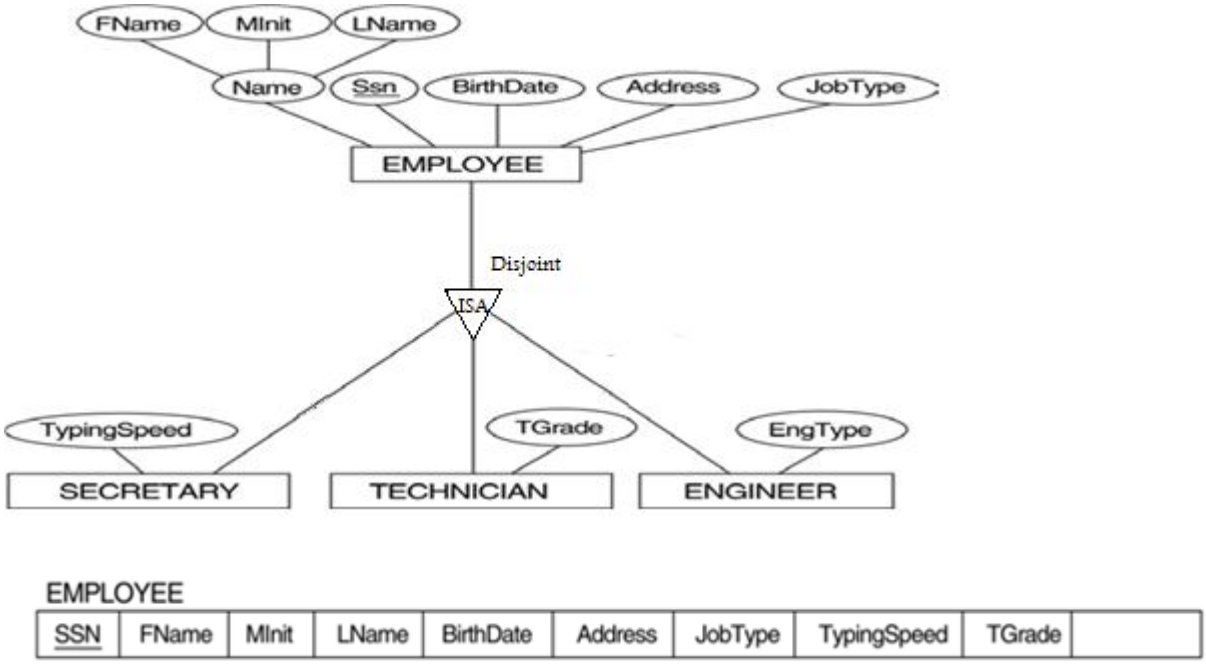
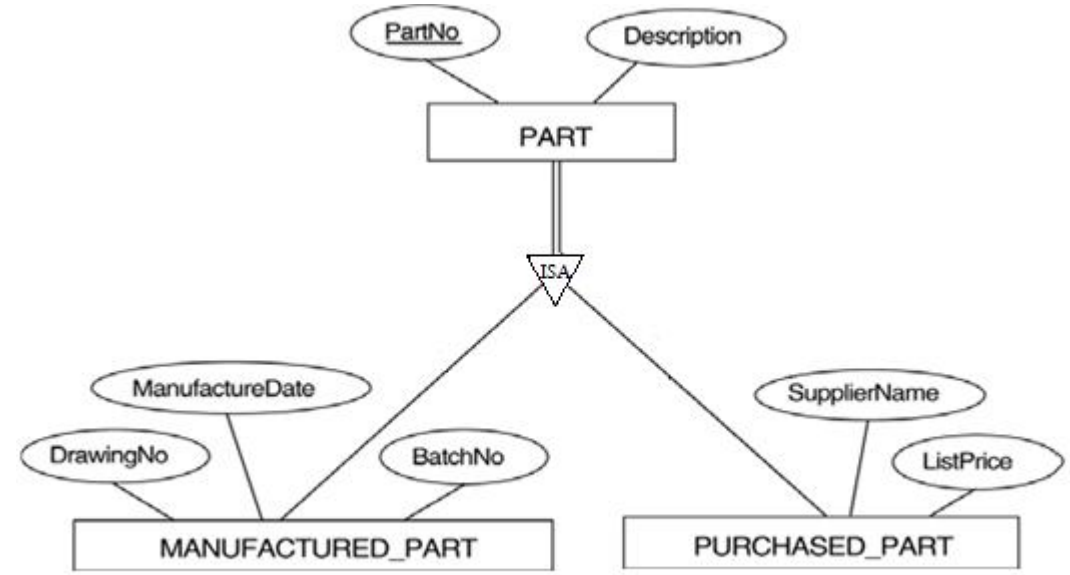


Figure 5.7: Option 8C Example.

- ***Option 8D: Single relation with multiple type attributes.***

    - Create a single relation schema L with attributes Attrs(L) = {k,a1,…an} U {attributes of S1} U…U {attributes of Sm} U {t1, t2,…,tm} and PK(L) = k. Each ti, 1 < I < m, is a Boolean type attribute indicating whether a tuple belongs to the subclass Si.



PART

PartNo	Description	MFlag	DrawingNo	ManufactureDate	BatchNo	PFlag	SupplierName	ListPrice

Figure 5.8: Option 8D Example.