

IT420: Database Management and Organization

Entity-Relationship Model to Relational Model
20 January 2006

Adina Crăiniceanu
www.cs.usna.edu/~adina

1

Database Design Process

- Requirements analysis
- Conceptual design: Entity-Relationship Model
- **Logical design: transform ER model into relational schema**
- Schema refinement: Normalization
- Physical tuning

2

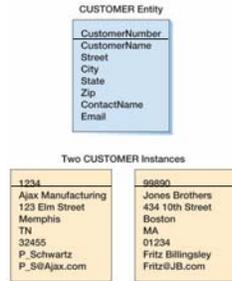
Goals

- Transform ER model to relational model

3

ER Model - Entities

- Attributes
- Identifiers: no two instances with the same value for identifier
- Weak entity: its existence depends on another entity
- ID-Dependent entity: weak + identifier contains another identifier



4

ER Model - Relationships

- HAS-A
 - Maximum cardinality
 - Minimum cardinality
- IS-A (super-type/sub-type)
 - Exclusive/inclusive

5

Relational Model

- Tables
- Integrity constraints
 - Primary key
 - Candidate key
 - Foreign key

6

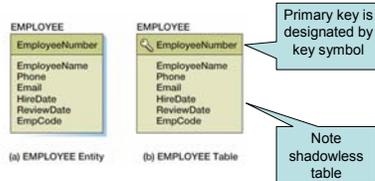
ER to Relational

- Transform entities in tables
- Transform relationships using foreign keys
- Specify logic for enforcing minimum cardinalities

7

Create a Table for Each Entity

EMPLOYEE (EmployeeNumber, EmployeeName, Phone, Email, HireDate, ReviewDate, EmpCode)



8

Select the Primary Key

- The ideal primary key is short, numeric and fixed
- Surrogate keys meet the ideal, but have no meaning to users



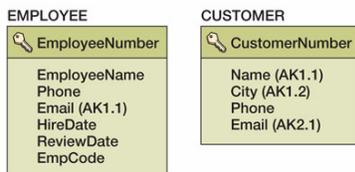
9

Specify Candidate (Alternate) Keys

- **candidate key = alternate key**
- **Candidate keys:** alternate identifiers of unique rows in a table
- ERwin uses **AK n . m** notation, where n is the number of the alternate key, and m is the column number in that alternate key

10

Specify Candidate (Alternate) Keys



11

Specify Column Properties: Data Type

- Generic Data Types:
 - CHAR(n)
 - VARCHAR(n)
 - DATE
 - TIME
 - MONEY
 - INTEGER
 - DECIMAL



12

Specify Column Properties: SQL Server Data Types

Data Type	Description
Binary	Binary, length 0 to 8,000 bytes.
Char	Character, length 0 to 8,000 bytes.
Datetime	8-byte datetime. Range from January 1, 1753, through December 31, 9999, with an accuracy of three-hundredths of a second.
Image	Variable length binary data. Maximum length 2,147,483,647 bytes.
Integer	4-byte integer. Value range from -2,147,483,648 through 2,147,483,647.
Money	8-byte money. Range from -922,337,203,685,477.5808 through +922,337,203,685,477.5807, with accuracy to a ten-thousandth of a monetary unit.
Numeric	Decimal - can set precision and scale. Range -10 ³⁸ -1.
Smalldatetime	4-byte datetime. Range from January 1, 1900, through June 6, 2079, with an accuracy of one minute.
Smallint	2-byte integer. Range from -32,768 through 32,767.
Smallmoney	4-byte money. Range from 214,748.3648 through +214,748.3647, with accuracy to a ten-thousandth of a monetary unit.
Text	Variable length text, maximum length 2,147,483,647 characters.
Tinyint	1-byte integer. Range from 0 through 255.
Varchar	Variable-length character, length 0 to 8,000 bytes.

13

Specify Column Properties: Oracle Data Types

Data Type	Description
BLOB	Binary large object. Up to 4 gigabytes in length.
CHAR(n)	Fixed length character field of length n. Maximum 2,000 characters.
DATE	7-byte field containing both date and time.
INTEGER	Whole number of length 38.
NUMBER(n,d)	Numeric field of length n, d places to the right of the decimal.
VARCHAR(n)	Variable length character field up to n characters long. Maximum value of n = 4,000
VARCHAR2(n)	

14

Specify Column Properties: Null Status

- Null status indicates whether or not the value of the column can be NULL

EMPLOYEE
EmployeeNumber: NOT NULL
EmployeeName: NOT NULL
Phone: NULL
Email: NULL (AK1.1)
HireDate: NOT NULL
ReviewDate: NULL
EmpCode: NULL

15

Specify Column Properties: Default Value

- A **default value** is the value supplied by the DBMS when a new row is created

Table	Column	Default Value
ITEM	ItemNumber	Surrogate key
ITEM	Category	None
ITEM	ItemPrefix	If Category = 'Perishable' then 'P' If Category = 'Imported' then 'I' If Category = 'One-off' then 'O' Otherwise = 'N'
ITEM	ApprovingDept	If ItemPrefix = 'I' then 'SHIPPING/PURCHASING' Otherwise = 'PURCHASING'
ITEM	ShippingMethod	If ItemPrefix = 'P' then 'Next Day' Otherwise = 'Ground'

16

Specify Column Properties: Data Constraints

- **Data constraints** are limitations on data values:
 - **Domain constraint** - Column values must be in a given set of specific values
 - **Range constraint** - Column values must be within a given range of values
 - **Intra-relation constraint** – Column values are limited by comparison to values in other columns in the same table
 - **Inter-relation constraint** - Column values are limited by comparison to values in other columns in other tables

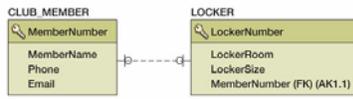
17

Create Relationships: 1:1 Strong Entity Relationships

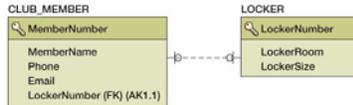
- Place the key of one entity in the other entity as a foreign key:
 - Either design will work – no parent, no child
 - Minimum cardinality considerations may be important:
 - O-M will require a different design than M-O

18

Create Relationships: 1:1 Strong Entity Relationships



(a) With Foreign Key in LOCKER



(b) With Foreign Key in CLUB_MEMBER

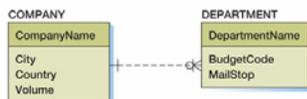
19

Create Relationships: 1:N Strong Entity Relationships

- “Place the key of the parent in the child”

20

Create Relationships: 1:N Strong Entity Relationships



(a) 1:N Strong Entity Relationship

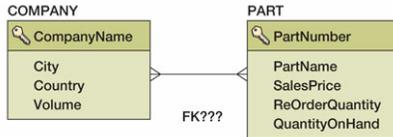


(b) Placing the Key of the Parent in the Child

21

Create Relationships: N:M Strong Entity Relationships

- In an N:M strong entity relationship there is no place for the foreign key in either table:
 - A COMPANY may supply many PARTs
 - A PART may be supplied by many COMPANYS



22

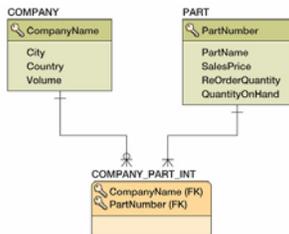
Create Relationships: N:M Strong Entity Relationships

- Create an **intersection table**:
 - The primary keys of each table → **composite primary key** for intersection table
- Each table's primary key becomes a foreign key linking back to that table

23

Create Relationships: N:M Strong Entity Relationships

COMPANY_PART_INT (*CompanyName, PartNumber*)



24

Relationships Using ID-Dependent Entities: Four Uses for ID-Dependent Entities

- Representing N:M Relationships
 - We just discussed this
- Association Relationships
- Multivalued Attributes
- Archtype/Instance Relationships

25

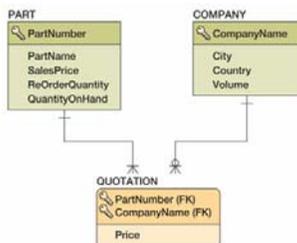
Relationships Using ID-Dependent Entities: Association Relationships

- An intersection table:
 - Holds the relationships between two strong entities in an N:M relationship
 - Contains only the primary keys of the two entities:
 - As a composite primary key
 - As foreign keys
- An **association table**:
 - Has all the characteristics of an intersection table
 - **PLUS it has one or more columns of attributes specific to the associations of the other two entities**

26

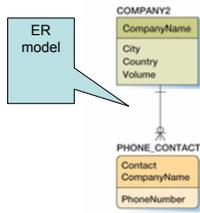
Relationships Using ID-Dependent Entities: Association Relationships

QUOTATION (*CompanyName*, *PartNumber*, Price)



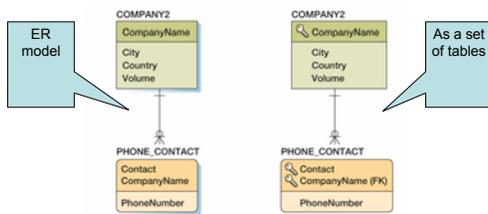
27

Relationships Using ID-Dependent Entities: Multivalued Attributes



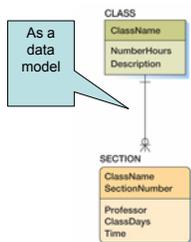
28

Relationships Using ID-Dependent Entities: Multivalued Attributes



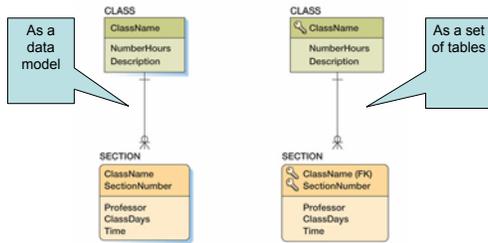
29

Relationships Using ID-Dependent Entities: Archetype/Instance Pattern



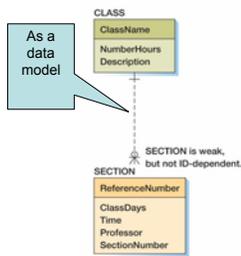
30

Relationships Using ID-Dependent Entities: Archetype/Instance Pattern



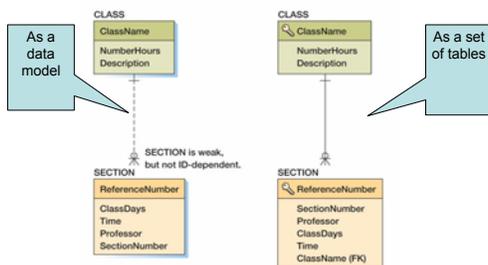
31

Relationships Using Weak Entities: Archetype/Instance Pattern



32

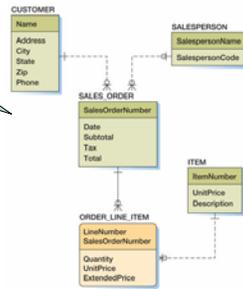
Relationships Using Weak Entities: Archetype/Instance Pattern



33

Mixed Entity Relationships: The Line-Item Pattern

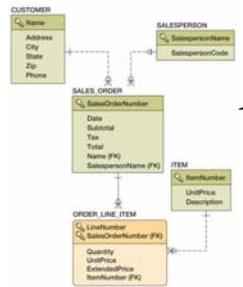
As a data model



34

Mixed Entity Relationships: The Line-Item Pattern

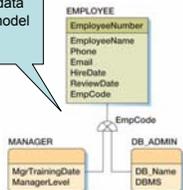
As a set of tables



35

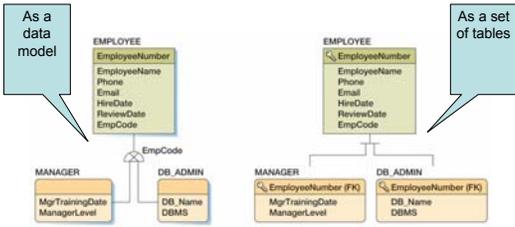
Subtype Relationships

As a data model



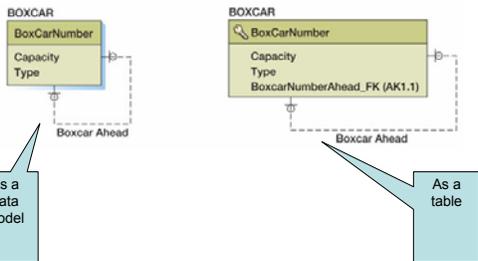
36

Subtype Relationships



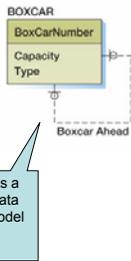
37

Recursive Relationships: 1:1 Recursive Relationships



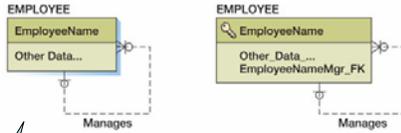
38

Recursive Relationships: 1:1 Recursive Relationships



39

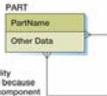
Recursive Relationships: 1:N Recursive Relationships



As a data model

As a table

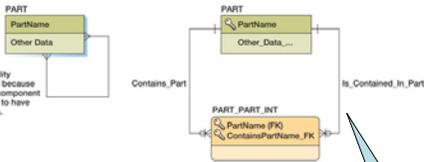
Recursive Relationships: N:M Recursive Relationships



The minimum cardinality is optional to optional because a part need not be a component part, nor does it need to have any subordinate parts.

As a data model

Recursive Relationships: N:M Recursive Relationships



The minimum cardinality is optional to optional because a part need not be a component part, nor does it need to have any subordinate parts.

As a data model

As a set of tables

Minimum cardinality: Parent Required

DEPARTMENT ([DepartmentName](#), BudgetCode, ManagerName)
EMPLOYEE ([EmployeeNumber](#), EmployeeName,
[DepartmentName](#))

- On Insert: child OK
- On Delete:
 - delete child (cascade) or prohibit
- On Update:
 - update child (cascade) or prohibit

43

Minimum cardinality: Child Required

DEPARTMENT ([DepartmentName](#), BudgetCode, ManagerName)
EMPLOYEE ([EmployeeNumber](#), EmployeeName,
[DepartmentName](#))

- More difficult to enforce (write code)
- Tricky:
 - A department must have some employee
 - EMPLOYEE has DepartmentName as FK, NOT NULL

44
