Lecture 4

Conceptual Design with the
Entity Relationship Data Model

Week 2

Overview

- The Entity Relationship Data Model
- Entities and Attributes
- Key attributes (candidate and primary keys)

2

The Entity Relationship Data Model

- Conceptual level modelling language to represent facts about the universe of discourse.
- The ER model represents fact *types*, not individual facts: individual facts will be stored in the database instance

  E.g. 'Employee' has Social Security Number (ssn) and Name – means that every entity of this type has an ssn and a name

3

Ontology of the ER data model

- The central concept of the ER data model is the *entity*. According to the world view of ER the world consists of entities which have properties.
- 'Idiosynchratic' properties are those which the entity has without reference to other represented entities
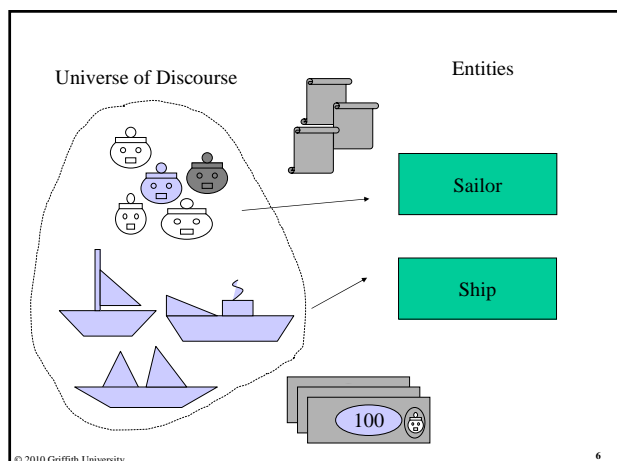- 'Relational' properties are those which the entity has in relation to other represented entities

4

Entities and Attributes

- When reading the English description of a Universe of Discourse we can identify objects of interest, or *entities*
- These 'objects of interest' generally appear as *nouns* in the text

5

6

1

How to identify entity types

noun

E.g. *'Sailors* have a name, country of origin and a registration number.'

- The database will have to represent facts about sailors, e.g.

  Joe Blogg, UK, registration # A1298

  Kris Haptelund, Norway, registration # N97664

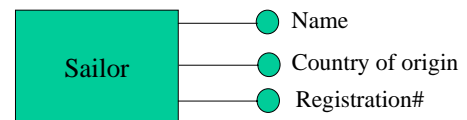- Here 'Sailor' is an entity type, while Joe and Kris are individual entities.

  Note the use of singular

- We represent entity *types* on ER schemas
- The name of an entity type is normally a noun in *singular*
- Any given entity type has an *entity set* associated with it. Elements of this entity set are the entities which have the given type.

  E.g. the entity set 'Sailors' consists of entities (like Joe and Kris), and the type of each member of this entity set is 'Sailor'
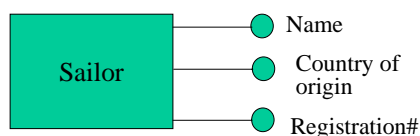

Attributes

- An attribute of an entity type represents a *property* that entities of that type may have
- Attributes have a *value*
- Values may not be arbitrary. For any attribute there is a set of possible values defined, called the *attribute domain*
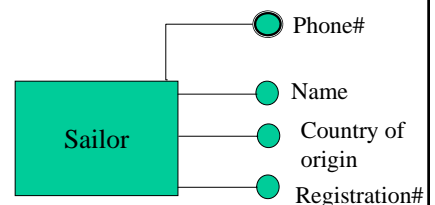

Sailor — Name, Country of origin, Registration#

Attributes

- Simple
- Set valued
- Complex

Sailor — Name, Country of origin, Registration#

Attributes

- Simple
- Set valued
- Complex

Sailor — Phone#, Name, Country of origin, Registration#

Attributes

- Simple
- Set valued
- Complex →

Sailor entity with attributes: postcode, city, Street addr, address (complex), Phone#, Name, Country of origin, Registration#

13

Derived attributes

- Some attributes are not independent from others, their value can be derived from the rest. E.g. 'Age' is a derived attribute of 'Person':

Person entity with Date of birth and Age (derived)
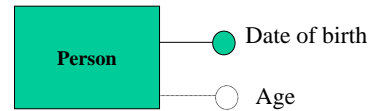
14

Key attributes
(candidate and primary keys)

- A *key* of an entity type is a set of attributes such that no two entities agree on all values of those attributes
- In the simple case we can find an attribute, such that each entity has a different value for that property
- E.g. for 'Sailor' the 'Registration number' may uniquely identify *a* sailor, thus it is a key attribute

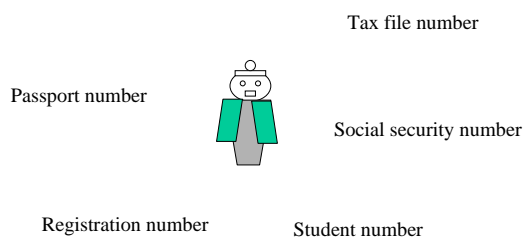**15**

Candidate keys

- There may be several sets of attributes unique to entities of an entity type
- Candidate keys of Sailor could be

  ck: {Registration#} ,
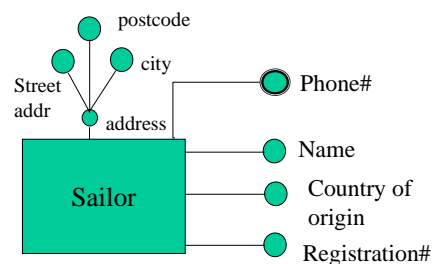   {Name, MothersMaidenName, Birthdate)

16

Made-up key attributes...

Tax file number, Passport number, Social security number, Registration number, Student number

17

Primary key

Sailor entity with attributes: postcode, city, Street addr, address, Phone#, Name, Country of origin, Registration# (underlined)

Entity integrity constraint: value for primary key *must* be known, and its value *must be unique* for each entity

18

Null values

- The ER data model does not prescribe that each attribute must have a value associated with it - if the value is unknown or not applicable, then we assign a 'NULL value' to that attribute (meaning there is *no* known value for that attribute)
- However, there must be at least one key set of attributes, called the *primary key*, for which values ***must*** be known.

19

Summary

The following concepts have been introduced:
- Entity
- Attribute (simple, set valued, complex)
- Attribute domain
- Key (candidate, primary)

Still to be covered:
Relationships, Participation and cardinality constraints, binary / ternary / higher order relationship types, Roles, Attributes of relationship types

20

The end

21