

G64DBS Database Systems

Lecture 4 Entity/Relationship Modelling

Tim Brailsford

Designing a database

- What is the database going to be used for?
- What tables, keys & constraints are needed?
- Need to build a model of the real world

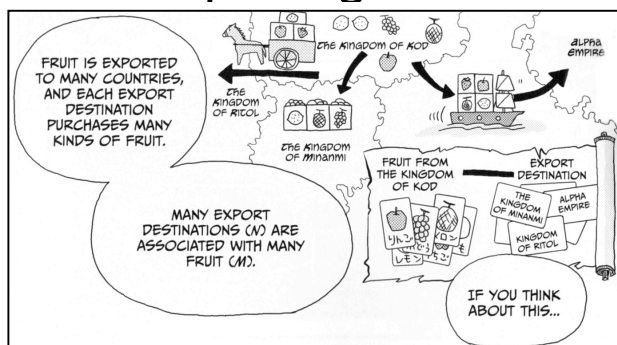
Designing a database



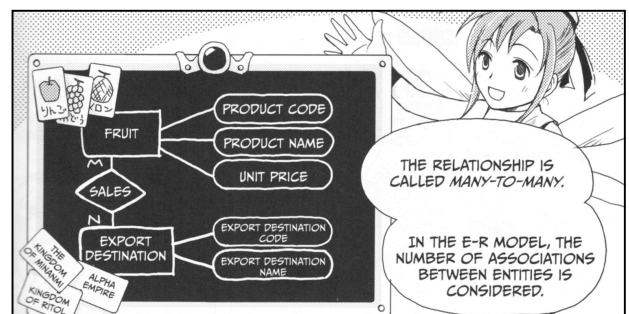
Entity/Relationship Modelling

- E-R Modelling
- Conceptual design
- Entities:
 - Objects or items of interest
- Attributes:
 - Facts about or properties of an entity
- Relationships:
 - Links between entities

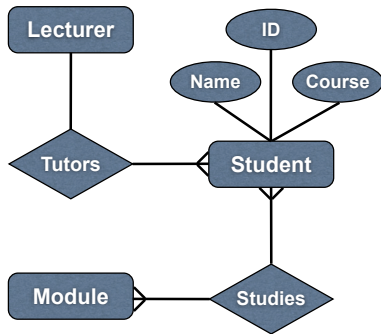
Example: Exporting Fruit



E-R Diagram

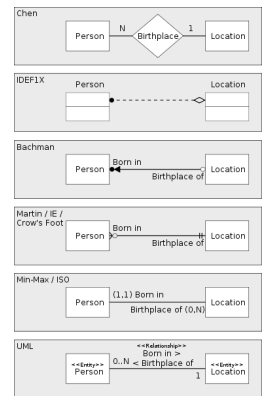


E-R Diagram



E-R Notation

- Various conventions
- Chen
 - Mostly American
 - Manga guide
- Crow's Foot
- Mostly British
- Many others

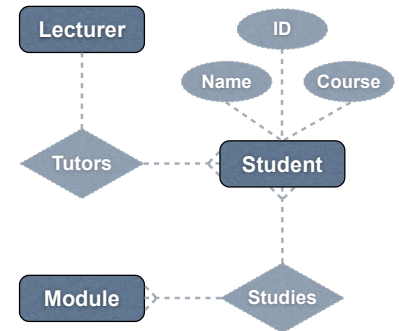


Entities

- Entities represent objects or things of interest
 - Physical things like fruit, students, lecturers etc.
 - More abstract things like export destinations, modules, courses
- Entities have
 - A type or class, such as Fruit
 - Instances of that type, such as Apple or Lemon
- Attributes

Visualising Entities

- In an E/R Diagram, an entity is usually drawn as a box with rounded corners
- The box is labelled with the name of the entity

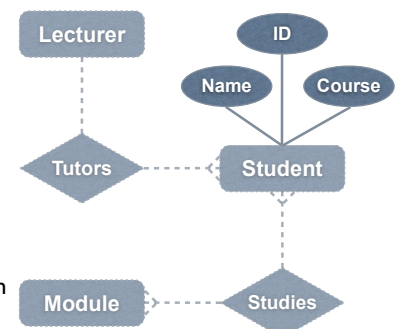


Attributes

- Attributes are facts, aspects, properties, or details about an entity
 - Fruit has ID, name ...
 - Students have IDs, names, courses, addresses, ...
 - Modules have codes, titles, credit weights, levels ...
- Attributes have:
 - A name
 - A domain of possible values
 - Each value from the domain represents an instance of the entity

Visualising Attributes

- In an E/R Diagram attributes are usually drawn as ovals
- Each attribute is linked to its entity by a line
- The name of the attribute is written in the oval



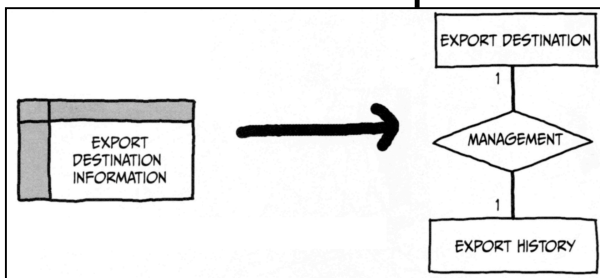
Relationships

- Relationships are an association between two or more entities
- Each Fruit has a single Price
- Each Student takes several Modules
- Each Module is taught by a Lecturer
- Relationships have
 - A name
 - A set of entities that participate in it
 - A degree - the number of entities that participate in it
 - A cardinality ratio

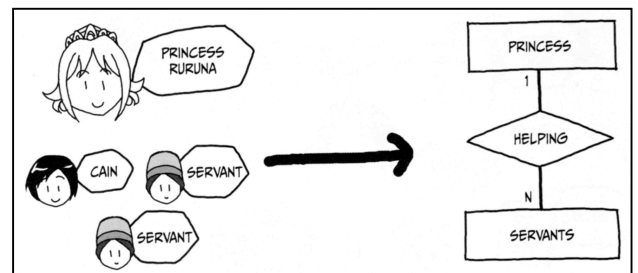
Cardinality Ratios

- Cardinality Ratio of a relationship describes the number of instances that can participate in the relationship
- NB the degree of a relationship describes the number of entities in the relationship
- One to one - 1:1
 - Each lecturer has a unique office
- One to many - 1:M (sometimes written as 1:N)
 - A lecturer may tutor many students, but each student has just one tutor
- Many to many - M:M (sometimes written as M:N)
 - Each student takes several modules, and each module is taken by several students

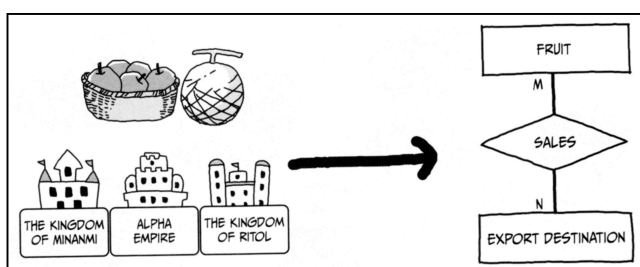
One to One Relationship



One to Many Relationship

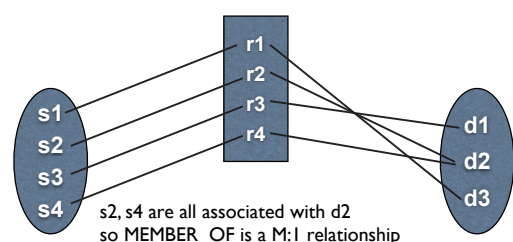


Many to Many Relationship



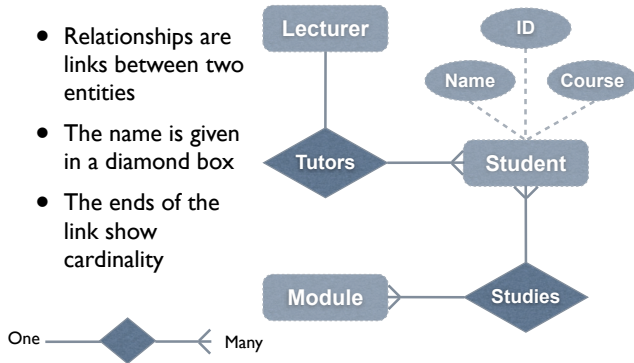
Relationship Example

(s1, s2, s3, s4) are STUDENT instances
 (d1, d2, d3) are SCHOOL instances
 (r1, r2, r3, r4) are MEMBER_OF instances



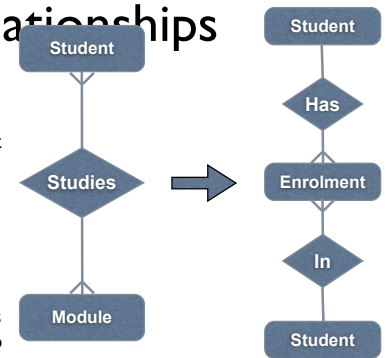
Visualising Relationships

- Relationships are links between two entities
- The name is given in a diamond box
- The ends of the link show cardinality



Removing M:M Relationships

- Many to many relationships are difficult to represent in a database
- We can split a many to many relationship into two one to many relationships
- An entity represents the M:M relationship



Conceptual Design using E-R Models

- Given a description of an application you need to identify:
 - Entities
 - Attributes
 - Relationships
 - Cardinality ratios
- General guidelines:
 - Since entities are things or objects they are often nouns in the description
 - Attributes are facts or properties, and so are often nouns also
 - Verbs often describe relationships between entities

Example

A university consists of a number of schools. Each school offers several courses. A number of modules make up each course. Students enrol in a particular course and take modules towards the completion of that course. Each module is taught by a lecturer from the appropriate school, and each lecturer tutors a group of students

Example: Entities

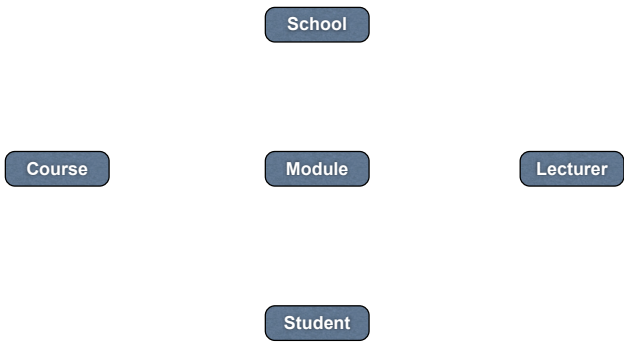
A university consists of a number of **schools**. Each school offers several **courses**. A number of **modules** make up each course. Students enrol in a particular course and take modules towards the completion of that course. Each module is taught by a **lecturer** from the appropriate school, and each lecturer tutors a group of students

Example: Relationships

A university consists of a number of schools. Each school **offers** several courses. A number of modules **make up** each course. Students **enrol in** a particular course and **take** modules towards the completion of that course. Each module is **taught by** a lecturer **from the** appropriate school, and each lecturer **tutors** a group of students

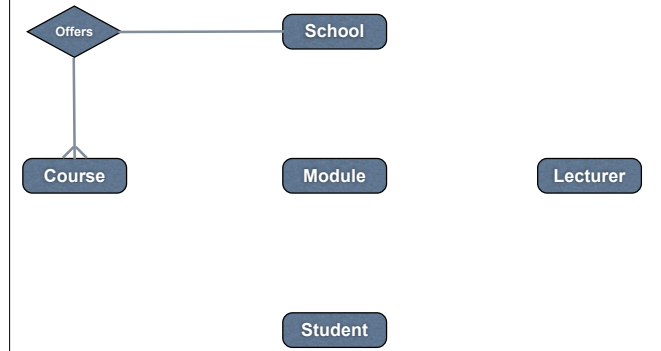
Example: E-R Diagram

Entities: School, Course, Module, Lecturer, Student



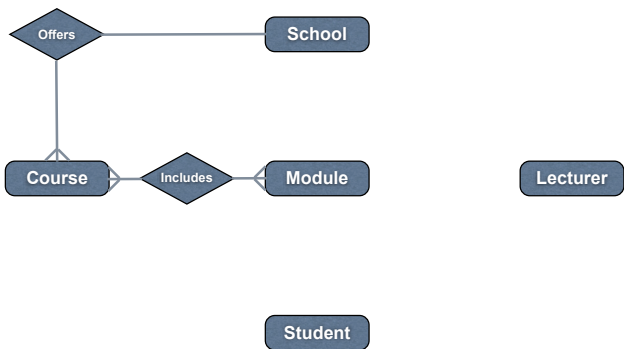
Example: E-R Diagram

Each school offers several courses



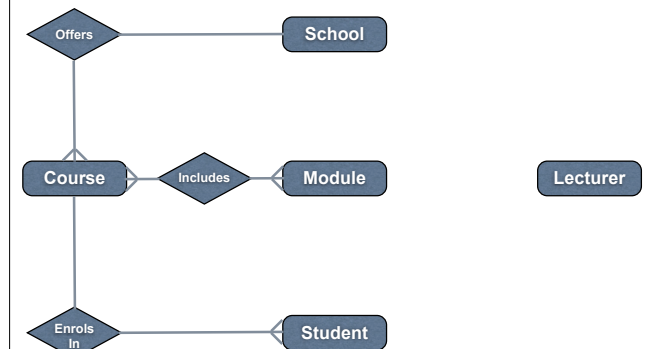
Example: E-R Diagram

A number of modules make up each courses



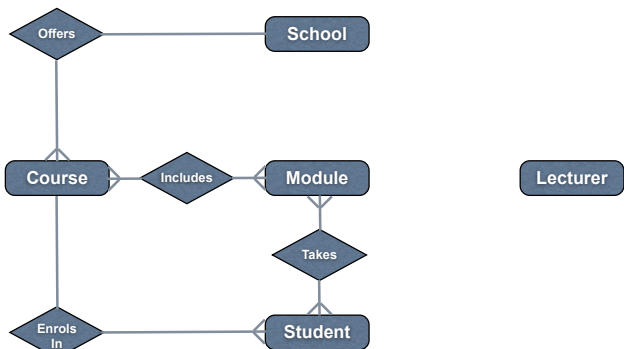
Example: E-R Diagram

Students enrol in a particular course



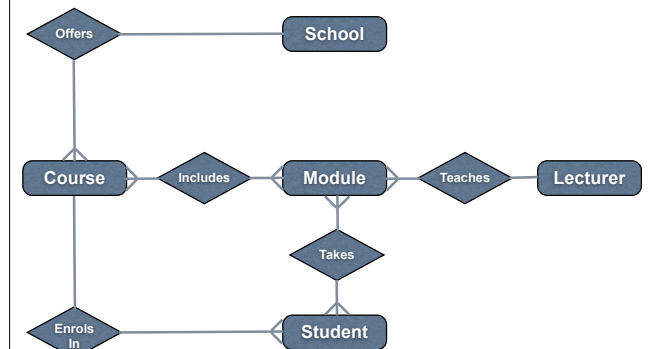
Example: E-R Diagram

Students take modules



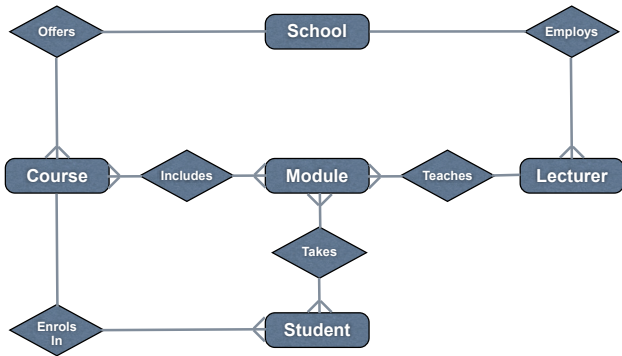
Example: E-R Diagram

Each module is taught by a lecturer



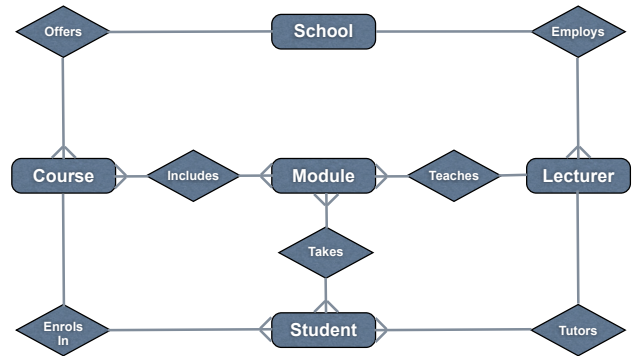
Example: E-R Diagram

The lecturer is employed by the appropriate school

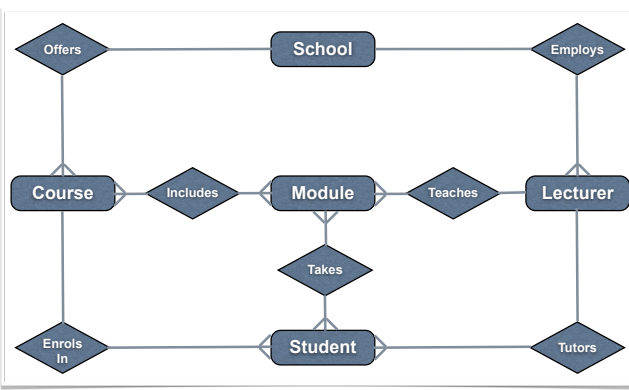


Example: E-R Diagram

Each lecturer tutors a number of students



Example: E-R Diagram



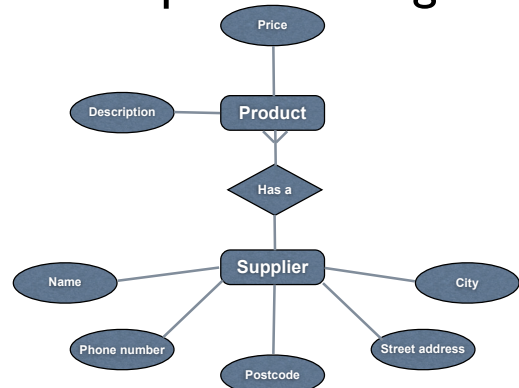
Entities and Attributes

- Sometimes it is hard to tell if something should be an entity or an attribute
 - They both represent objects or facts about the world
 - They are both often represented by nouns in descriptions
- **General guidelines**
 - Entities can have attributes but attributes have no smaller parts
 - Entities can have relationships between them, but an attribute belongs to a single entity

Example

We want to represent information about products in a database. Each product has a description, a price and a supplier. Each supplier has a name, street, city, postcode, and phone number.

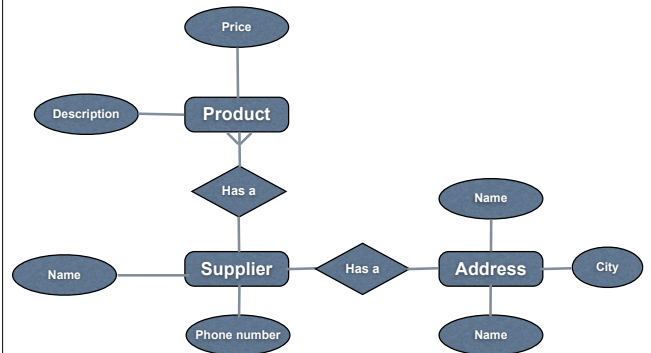
Example: E-R Diagram



Example: Relationships

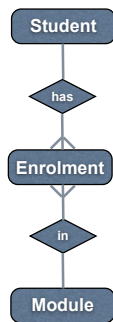
- Each product has a single supplier
 - A supplier might supply many products
 - Therefore Product–Supplier is a M:1 relationship
- Each supplier has an address
 - A supplier has a single address
 - It does not seem sensible for two different suppliers to have the same address
 - Therefore Supplier-Address is a 1:1 relationship

Example: E-R Diagram



Debugging Designs

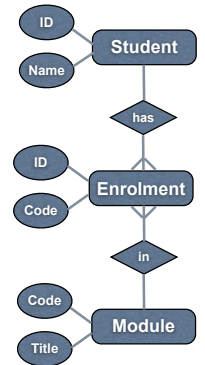
- With practice E-R diagrams can be used to plan queries
 - You can look at the diagram and figure out how to find useful information
 - If you can't find the information you need, you may need to change the design



How can you find a list of students who are enrolled in the "Database Systems" module?

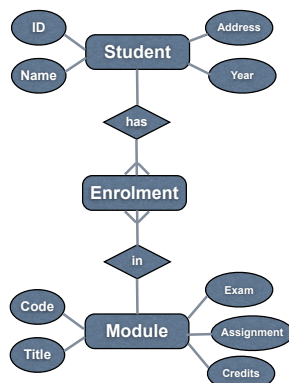
Debugging Designs

1. Find the instance of the **Module** entity with title "Database Systems"
2. Find instances of the **Enrolment** entity with the same Code as the result of (1)
3. For each instance of **Enrolment** in the result of (2) find the corresponding **Student**



Implementing E-R Designs

- Given an E/R design
 - The entities become database tables
 - Attributes of an entity become columns in the corresponding table
 - Relationships may be represented by foreign keys



Representing Relationships

- Table representing an entity at the M side of the relationship will have a foreign key representing the relationship
- The Enrolment table
 - Will have a foreign key to Student for the 'has' relationship
 - Will have a foreign key to Module for the 'in' relationship

