Database Systems 2

Lecture 10

Database Design Methodology

Conceptual Design

Conceptual Database Design

1 Build local conceptual data model for each user view.

1.1 Identify entities

1.2 Identify relationships

1.3 Identify and associate attributes with entities or relationships

1.4 Determine attribute domains

1.5 Determine candidate and primary key attributes

1.6 Specialise/generalise entities (optional)

1.7 Draw Entity-Relationship diagram

1.8 Review local conceptual data model with user

1.1 Identify entities

Look for objects that have an existence in their own right.

Look for nouns.

Beware of synonyms and homonyms.

This is a subjective process.

- Is marriage an entity, a relationship or an attribute?

One set of requirements can give rise to several equally valid set of entity types.

You are not seeking out ultimate truth – you are trying to come up with a good enough set of entities for the purpose at hand.

<u>Document Entities</u>

Assign meaningful names and record names and descriptions in a data dictionary. Also aliases or synonyms.

1.2 Identify relationships

Again look for nouns in the requirements spec.

Also phrases such as ‘Branch has staff’ or ‘Staff manages Property’.

Some relationships that exist in real life may not be required by the database. The list of transactions or outputs may help to determine this.

Don’t assume that all relationships are binary ones.

Use ER diagrams to sketch out entities and relationships.

Document Relationships

For each relationship record: name, cardinality and participation constraints in the data dictionary.

1.3 Identify and associate attributes with entities or relationships

What information are we required to hold on…..?

Distinguish between simple and composite attributes.

- Will your database need to hold an address as one data item or several data items?

Identify derived attributes if possible.

Keep a look out for attributes that appear to be associated with more than one entity.

- Does this indicate the need to specialise or generalise?
- Does it indicate the existence of a relationship.

Document Attributes

Name and description (inc aliases or synonyms)
Data type and length
Default values / Nulls allowed?
Whether composite / derived (what from?)

1.4 Determine attribute domains

What is the valid list of values for each attribute.

- Branch number can be a three character variable length string consisting of one letter and a two digit number between 1-99.
- Telephone numbers are a fixed length 13 digit string.

Document Attribute Domains

Record the following the data dictionary:

- Allowable set or range of values for an attribute.
- Size and formats of attribute fields.

1.5 Determine candidate and primary key attributes

Identify the candidate keys for each entity.

Chose one to be the Primary key, based on:

- The CK that is less likely to have its values changed.
- The CK that is less likely to lose its uniqueness in the future.
- The CK with the fewest characters.
- The CK which is easiest to use from the users point of view.

If we are unable to identify a PK for an entity it is probably a weak entity.

Document Primary Keys

Record them in the data dictionary.

1.6 Specialise/generalise entities (optional)

Identify superclass and subclass entities.

- Do we represent Property_for_Rent and Property_for_Sale as two separate entities or as subclasses of one entity?
- How many attributes do they have in common?
- Do they have dramatically different relationships with other entities?

Be guided by the readability of the diagram and the clarity by which it models the enterprise.

1.7 Draw Entity-Relationship diagram

Note that you may have sketched out many draft ERDs by the time you get to this stage.

Also remember that this only represents **one (group of) user's view** of the enterprise.

1.8 Review local conceptual data model with user

Get user feedback.

Modify as appropriate – possibly repeating previous steps.

Get user to sign off the model.