

CS3101 - Databases  
Assignment: P1 - DB Design

Matriculation Number: 160001362

29/10/2018

## Contents

<b>1</b>	<b>E-R Model</b>	<b>4</b>
1.1	E-R Diagram . . . . .	4
1.2	Underlying Assumptions . . . . .	4
1.3	Justification of Design Decisions . . . . .	4
1.3.1	Primary Keys . . . . .	4
1.3.2	Types of Attributes . . . . .	5
1.3.3	Participation and Cardinality . . . . .	5
1.3.4	Modelling Relationships . . . . .	6
1.3.5	Placement of Attributes . . . . .	6
1.3.6	Specialisation/Generalisation . . . . .	6
<b>2</b>	<b>Relational Schema</b>	<b>7</b>
2.1	Derived Relational Schema . . . . .	7
2.2	Justification of Design Decisions . . . . .	8
2.2.1	Type of Attributes . . . . .	8
2.2.2	Placement of Foreign Keys . . . . .	8
2.2.3	Translation of Specialised/Generalised Entities . . . . .	9
2.2.4	Comparison With E-R Diagram . . . . .	9
<b>3</b>	<b>Normalisation</b>	<b>9</b>
3.1	Design Methodology . . . . .	9
3.2	Functional Dependencies . . . . .	10
3.3	Discussion of Functional Dependencies . . . . .	10
3.4	Discussion of Normal Form . . . . .	11
3.4.1	First Normal Form (1NF) . . . . .	11
3.4.2	Second Normal Form (2NF) . . . . .	11
3.4.3	Third Normal Form (3NF) . . . . .	11
<b>4</b>	<b>Word Count - 2421</b>	<b>12</b>



Figure 1: E-R diagram modelling ConMan conference management system.

# 1 E-R Model

## 1.1 E-R Diagram

See figure 1 for E-R diagram

## 1.2 Underlying Assumptions

- All layouts can be utilised by all rooms (with varying capacities).
- An individual can have a multiple affiliations at a time.
- The 'chair' and 'author' entities are distinct from the roles carried out by an organiser.
- A session is held in a single room and that a room can only hold one session at once.
- Every country uses a postcode.
- A session can only take place within a single day and not run on to another day.

## 1.3 Justification of Design Decisions

### 1.3.1 Primary Keys

- The candidate key 'conf.code' and 'year' was selected as the primary key for conference due to conference codes being unique year to year. The conf.code attribute allows any conference to be unique within a year, and the year allows it to be identifiable across any year.
- The 'name' attribute was selected as the primary key for the venue entity as it is unique according to the specification.
- The entity set 'room' is weak. The attribute 'name' has been taken to be the discriminator/partial key. The primary key of room is the discriminator 'name' combined with the name of the 'venue', as the specification states that a room has a unique name in the context of its venue.
- The 'type' attribute of layout has been made the primary key of the entity, as it seems unrealistic to have two layouts with the same type.
- The candidate key of 'institution', 'name' and 'location', has been made the primary key, as presumably there can be no institution with the same name in the same location. For example two University of St Andrews in the United Kingdom.
- An id attribute has been added to the 'session' entity as the primary key to avoid having an additional weak entity set whose relations would have to reference many attributes.

- The entity set 'activity' is weak. The discriminator 'position' has been selected as the partial key as position is unique in the context of an academic session.

### 1.3.2 Types of Attributes

- The discriminator 'name' has been chosen in the 'room' entity as each room has a unique name in the context of the venue.
- A composite attribute has been used for the address of the 'venue' entity, as address can compose values such as name/number, street, city, province, country and postcode.
- The 'role' attribute of the relationship 'organises' has been made multi-valued, as an organiser can have multiple roles within a single conference.
- The 'duration' attribute of 'session' is an attribute derived from start\_time and end\_time.

### 1.3.3 Participation and Cardinality

- The relationship 'holds' between venue and conference is one-to-many. This is because a single venue may hold many conferences, but a conference must be held at a single venue.
- The relationship 'has' between venue and rooms is one-to-many, as a venue must have one or more rooms but a room must belong to one venue.
- The relationship 'held in' between 'session' and 'room' is many-to-one, as many sessions can use a single room throughout the day but a room does not have to host a session. A session must be held in a room.
- The relationship 'configured in' between 'room' and 'layout' is many-to-many as a room can be configured into many layouts and a layout can be used by many rooms. A room must be configured into a layout but a layout does not have to be used by a room.
- The relationship 'chooses' between 'session' and 'layout' is many-to-one, as many sessions must choose a single layout but a layout does not have to be chosen by a session.
- The relationship 'organises' between 'organiser' and 'conference' is many-to-many as an organiser must organise one or more conferences (in order to be considered an organiser) and a conference must be organised by at least one organiser.
- The relationship 'affiliated with' is a ternary relationship between institution, individual and conference, and in all cases is many-to-many, as an individual can optionally have many affiliations with institutions and can

attend many conferences, a conference can be affiliated with many institutions via the individuals that attend, and institutions can be affiliated with many individuals and many conferences via its attending staff.

- The relationship 'belongs to' between session and conference is many-to-one. This is because a unique id is assigned to session so it must belong to a single conference, and a conference must be made up of a series of sessions.
- The 'chairs' relationship between 'chair' and 'academic' is a one-to-many relationship. An academic session must have exactly one chair leading it, but a chair may lead many academic sessions (and must lead at least one, in order to be a chair).
- The relationship 'has' between 'academic' and 'activity' is one-to-many as each academic session must be associated with one or more activities, and an activity must be associated with one session.
- The relationship 'written by' between 'activity' and 'author' is many-to-many. An author must have written one or more activities (in order to be an author) and an activity must have one or more authors.

#### **1.3.4 Modelling Relationships**

- The role of an organiser has been made an attribute of the relationship 'organises' between the organiser and the conference as it is related to both entities.
- The ternary relationship 'affiliated with' has been used to model the relationship between institutions, individuals, and conferences, in order to allow an individual to be affiliated with many institutions at once and across time while enabling the conference to still be associated with the individual and their old affiliation.

#### **1.3.5 Placement of Attributes**

- The attribute 'capacity' has been placed on the relationship 'configured in' as it relies on a room and the layout that it utilises.
- The multivalued attribute 'role' has been placed on the relationship 'organises' as roles relate both to the organiser who will undertake them but also to the conference which creates the need for the role.

#### **1.3.6 Specialisation/Generalisation**

- The entities chair, organiser, and author are all specialisations of the individual entity representing people involved in the conference. The completeness constraint for the inheritance is total, as all of the individuals

modelled in the conference data in the specification are either chairs, authors, or organisers. Therefore there is no need to allow individuals to be instantiated. The inheritance is overlapping as entities can belong to more than one subclass entity set - this is modelled using multiple arrows.

- The entities 'academic' and 'social' are specialisations of 'session' and have disjointed inheritance as a session can be either academic or social. The completeness constraint of the inheritance is total, as all sessions are academic or social sessions. This inheritance makes use of one arrow which splits off to academic and social as a result of the disjointed inheritance.

## 2 Relational Schema

### 2.1 Derived Relational Schema

The following relational schema differs slightly from the E-R diagram. The E-R diagram was an attempt to clearly capture the scenario laid out in the specification. The relational schema aims to more closely match the low level implementation of the database, so some minor changes have been made to reflect this.

Relational Schema		
Schema	Attributes	Types
conference	<u>conf_code</u> , <u>year</u> , name, description, venue_name*	integer, integer, string, string, string
venue	<u>venue_name</u> , building_number/name, street, city, province, country, postcode	string, string, string, string, string, string
room	<u>venue_name</u> *, <u>room_name</u>	string, string
configured in	<u>venue_name</u> *, <u>room_name</u> *, <u>layout_type</u> *, capacity	string, string, string, integer
session	<u>session_id</u> , title, start_time, end_time, venue_name*, room_name*, layout_type*, conf_code*, year*	integer, string, date/time, date/time, string, string, string, integer, integer
academic_session	<u>session_id</u> *, academic_session_type, individual_id*	integer, string, integer
layout	<u>layout_type</u>	string
individual	<u>individual_id</u> , name	integer, string
institution	<u>inst_name</u> , <u>inst_location</u>	string, string
affiliated_with	<u>inst_name</u> *, <u>inst_location</u> *, <u>conf_code</u> *, <u>year</u> *, <u>individual_id</u> *	string, string, integer, integer, integer
organises	<u>individual_id</u> *, <u>conf_code</u> *, <u>year</u> *, <u>role</u>	integer, integer, integer, string
written_by	<u>position</u> *, <u>session_id</u> *, <u>individual_id</u> *	integer, integer, integer
activity	<u>session_id</u> *, <u>position</u> , title	integer, integer, string

## 2.2 Justification of Design Decisions

### 2.2.1 Type of Attributes

- The composite attribute 'address' in the 'venue' entity has been flattened down into the attributes 'building\_number/name', 'street', 'city', 'province', 'country' and 'postcode'.
- The multivalued attribute 'role' has been made the discriminator (partial key) of the schema representing the 'organises' relationship set. The organises schema contains identifiers for individuals and conferences, and 'role' acts as an identifier for the specific role within a conference which the individual has taken on.
- The attribute 'duration' in the 'session' entity set has been left out of the 'session' schema as it is a derived attribute and would therefore be better represented as a function in the database implementation.
- The values in the relational schema have been assigned one of the following types - integer, string, or date/time. Arguably codes/IDs could also be represented using strings if they allow alphabetic characters rather than integers.

### 2.2.2 Placement of Foreign Keys

- Many-to-many relationship sets - Where entity sets share a many-to-many relationship, a schema is created for the relationship between the entities, containing foreign keys to each entity and any attributes that are associated with the relationship.
- Many-to-one/one-to-many relationship sets - Where entity sets share a many-to-one or one-to-many relationship, the foreign key of the 'one' schema is placed in the 'many' schema. For example 'individual\_id' is placed into the 'academic\_session' schema as a foreign key as a single individual chairs an academic session.
- One-to-one relationship sets - There are no one-to-one relationship sets in this representation of the scenario, but if there were, the foreign key could be placed in the schema representing either of the two entity sets.
- Weak Relationship Sets - For all relationship sets where one entity is a weak entity set, the foreign key of the identifying entity set is placed in the schema of the weak entity set. For example to identify a room in the context of a venue, the venue name must be in the schema of room as a foreign key.



### 2.2.3 Translation of Specialised/Generalised Entities

- Individual - In the E-R diagram, the specialisations of individual (author, chair, and organiser) were specified in order to more clearly model the scenario. When modelling the schema, it is clear that the only necessary schema is that of individual, as the specialisations have no additional attributes and have overlapping inheritance. Not having the total inheritance constraint be enforced is a worthwhile trade off in this scenario to avoid having data duplication of individuals who fulfill more than one specialisation of the entity set 'individual'.
- Session - In the E-R diagram, the distinction between academic sessions and social sessions is explicit. This clearly models the scenario, but is not required for the relational schema. As social sessions do not have any additional attributes compared to the 'session' generalisation, it can be assumed that any session that does not have the 'type' attribute of an academic session is a social session, removing the need for disjointed inheritance. Therefore in the schema, the inheritance constraint of the session is actually partial instead of total, and the two required schema are session and academic\_session.

### 2.2.4 Comparison With E-R Diagram

The E-R diagram for the most part has been directly translated into the relational schema. The only major departures, as discussed previously, are the changes made to the inheritance associated with the 'individual' and 'session' entity sets. Minor departures include more descriptive attribute names.

## 3 Normalisation

### 3.1 Design Methodology

The design used during the modeling of this scenario was a top-down approach. An E-R diagram was produced, followed by a relational schema, and then an identification of functional dependencies during the normalisation stage. This likely resulted in the schema being in second normal form, as discussed below, rather than in a higher form such as third normal form or Boyce-Codd normal form. Given more time, a more iterative, mixed approach would be adopted which would involve each stage of the design process being revised to achieve a higher form.

When using the database scenario section of the specification, noun identification was used to identify entity sets, verb identification was used to identify relationships between entities, the attributes were decided upon based on whether a noun had ownership over something else (e.g. described using the word 'has'), and specialisation/generalisation was used when an entity logically

fit the description as a subset of something else (e.g. an academic session is a subset of session).

### 3.2 Functional Dependencies

Full Functional Dependencies		
Schema	Determinant	Dependent Set
conference	conf_code, year	name, description, venue_name
venue	venue_name	building_number/name, street, city, province, country, postcode
venue	building_number/name, street, city, province, country, postcode	venue_name
venue	country, postcode	province
venue	building_number/name, street, province, postcode	country
venue	building_number/name, street, city, province, country	postcode
configured in session	venue_name, room_name, layout_type session_id	capacity title, start_time, end_time, venue_name, room_name, layout_type
session	conf_code, year	venue_name
academic_session	session_id	academic_session_type, individual_id
individual	individual_id	name
activity	session_id position	title

### 3.3 Discussion of Functional Dependencies

- conference - The 'name', 'description', and 'venue\_name' are fully functionally dependent on 'conf\_code' and 'year'. A 'conf\_code' is only unique within a given a year so removing 'year' would result in the 'name', 'description', and 'venue\_name' of conferences with the same code from different years being returned. The 'year' alone obviously cannot determine the attributes of a conference, therefore 'conf\_code' and 'year' must be the determinant.
- venue - The attribute 'venue\_name' is unique so can fully determine every attribute in the 'venue' entity set. The total address and postcode can also fully determine the 'venue\_name'. Parts of the address/postcode can also determine other parts of the address using UK postcode rules. However these functional dependencies are debatable based on national postal rules.
- configured in - The 'capacity' of a room configured in a layout is fully functionally dependent on 'venue\_name\*', 'room\_name\*', and 'layout\_type\*' as the capacity of a room depends on the layout it has been configured in, and 'venue\_name' is required to identify room.

- session - The title, start\_time, end\_time, venue\_name\*, room\_name\*, layout\_type\*, conf\_code\*, and year\*, of a session are fully functionally dependent on session\_id as it is universally unique. The conf\_code and year can also fully functionally determine the venue of the conference.
- academic\_session - academic\_session\_type and individual\_id\* are fully functionally dependent on session\_id\* as it is universally unique and so can be used to uniquely identify the chair and type of the current session which otherwise could be the same as other academic sessions.
- individual - the attribute 'name' is fully functionally dependent on individual\_id, as multiple individuals can have the same name which require the unique ID to be differentiated.
- activity - The title of an activity is fully functionally dependent on the session\_id and position of the activity, as title is not unique and so two different activities in the same session could in theory have the same title.
- The schema 'layout', 'institution', 'affiliated\_with', 'organises', and 'written\_by' contain only trivial functional dependencies so have not been included in the table of functional dependencies despite being fully functionally dependent.

### 3.4 Discussion of Normal Form

The derived relational schema is in second normal form:

#### 3.4.1 First Normal Form (1NF)

First normal form requires a relational schema where all of the domains of attributes are atomic. The relational schema meets these requirements as composite attributes such as 'address' have been flattened into individual attributes, and other attributes such as individual name do not need to be reduced into first and last name for this context.

#### 3.4.2 Second Normal Form (2NF)

Second normal form requires a relational schema to comply with 1NF and to satisfy the condition that every non-prime attribute be fully functionally dependent on a candidate key. As demonstrated by the functional dependency table, the relational schema satisfies this requirement by having a suitable candidate key for every schema that can identify all non-prime attributes. The trivial dependencies not listed in the table can clearly also identify themselves.

#### 3.4.3 Third Normal Form (3NF)

Third normal form requires a relational schema to comply with 2NF and to satisfy the condition that no non-prime attribute be transitively dependent on

a candidate key. Unfortunately this relational schema does not satisfy this requirement, as the venue schema has transitive dependencies from `venue_name` to `province` and `country`, and the session schema has a transitive dependency from `session_id` to `venue_name`.

## **4 Word Count - 2421**