[Group 20] Coursework 1

College Library Database Management System

ECS740P - Database Systems

Anthony Stockman

[Group 20]

| Vahesian Logan Vasudeva | David Haunschild | Edward Monah | Mohsen Razvi |

Table of Contents

Introduction

This paper aims to outline the design of a database system for a college library based on accepted database design principles to fulfil the requirements for Coursework 1 for the module ECS740P - DATABASE SYSTEMS, and the following scenario:

"A college library provides various resources for students and staff, including books, videos, DVDs and CDs. It is common that several copies are kept of some resources, for example, recommended books for courses. The usual loan period of a resource is 2 weeks, but some resources are available for short loan only (2 days) and some other resources can only be used within the library. The library consists of 3 floors. Resources are stored in the library on shelves. To locate a specific item in the library a combination of floor number and shelf number are used. In addition to this, a class number system is used to identify in which subject area a particular item belongs, for example all resources concerned with Database Systems will have the same class number. Students hold library cards which identify them as valid members of the Library. Students can loan a number of different resources at one time, but the total number of resources they may borrow at a given time must never exceed 5. Staff members at the College also hold library cards and are allowed to loan up to 10 different items at one time. The library charges fines for resources that are loaned for longer than the time allowed for that resource. For each day a resource is overdue the member is fined one dollar. When the amount owed in fines by a member is more than 10 dollars, that member is suspended until all resources have been returned and all fines paid in full" (Stockman, 2020).

This report solely contains E.R conceptual design of the database system visualised as an Enhanced ER diagram using Chen Notation (Li and Chen, 2009), and the relational database schema derived from said EER diagram normalised to third normal form. Assumptions are clearly stated where they have been made.

From the coursework case, as stated above (Stockman, 2020) it can be inferred that the system needs to be able to fulfil the following criteria:

- Assist in the borrowing/lending of resources from the library by verifying outstanding copies in their location, as well as distinguishing between resources by type and subject
- Determine the maximum length of the loan and how many resources a particular member can loan at a given time.
- Register which member currently holds/previously held a resource, including the loan time period on both outstanding and resolved loans.
- Calculate the debt owed by any member if the loan period has elapsed, issue a fine according to the number of expired resources and days overdue, and suspend the member if fines exceed $10 threshold.

The following sections of the report will go on to comprehensively detail the process of developing the functional design of the database system.

Entity-Relationship Model

Initially, assumptions have to be made to clarify the database's final objective, its practical functions, and how it can be realistically expected to be used. Entities and relationships are then determined with functional requirements and memory optimisation in mind.

These entities and relations are shown in the format of an Enhanced Entity-Relationship diagram using Chen Notation (Li and Chen, 2009). This diagram maps out the relationships between the entities and the database system and states their attributes. Firstly, the entities were identified and defined. The interactions between the different entities are then stated as the relations connecting them, before determining the cardinality between the entities.

In the case of a database for a college library, only a few necessary entities were found, the key ones being: 'Resource'[1], and 'Member'.

Regarding the 'Resource' entity, when listing the type of resource - this was done in the form of a total disjoint specialisation (Lemahieu et al., 2018) with 'Resource' as the superclass and the subclasses of: 'Book', 'Video/DVD', and 'CD'. This was done as different types of resources require different information to be stored to describe them.

Assumptions include (Table 1)[2]:

- Only titles of resources stored in the database are titles that are a library resource **[1]**
- There is no distinction between different copies of the same resource **[2]**
- All resources as defined by "Resource_ID" are of the same edition **[3]**
- Many resources can be loaned to many members **[4]**

Considering these assumptions, the 'Resource' entity was allocated the following attributes: "Resource_ID", "Borrowable_Time", "Type", "No_of_Total_Copies."These attributes are all single. 'Resource' has been divided into three disjoint subclasses: 'CD_Title', 'Book_Title' and 'DVD/Video_Title'.

The following assumptions have been made about the elements of the subclasses (Table 1):

- CDs are normal music albums/spoken words CDs **[5]**
- DVD/Videos are of normal films and TV shows **[6]**

[1] Throughout this report single entities will be denoted by single quotation mark 'Entity', while attributes will be denoted by double quotation marks "Attribute."
[2] Table 1 refers to the link to the Assumption Key in Appendix I and can be selected for direct navigation.

Some attributes to these weak entities store information including their name, creator and when they are released/published. These improve search functionality by expanding search criteria of the database.

There are two dedicated entities ‘Class’ and ‘Location’ related to the Resource Entity:

The ‘Class’ entity contains only single attributes and uniquely identifies the resource’s relevant academic subject, taught by a particular department. As attributes we have chosen: “Class_ID”, “Department” and “Subject.” We have therefore made the following assumptions (Table 1):

- There is no distinction between different year groups **[7]**
- The “Class_ID” is a number which determines the subject the resource is for **[8]**
- Any resource belongs to only one subject **[9]**
- There may be subjects that have no resources in the library **[10]**

Staff and students can find materials exclusively for their course via this database.

The ‘Location’ entity contains the attributes: “Location_ID”, “Floor_no” and “Shelf_no” as identifiers for the physical location of the resource. The following assumptions were made (Table 1):

- If the shelf and floor numbers are known, then that is enough information for the resource to be located **[11]**
- All resources with the same “Class_ID” are not necessarily found in the same location **[12]**

The attributes for the ‘Member’ entity contain all relevant details for the member. The attribute for a member’s name is stored as this is key to understanding who the account belongs to, other necessary personal details are taken such as: phone number, email address, date of birth, and address. These are taken in order to have means of getting in touch with the member regarding anything going on at the library, or if they have any fines due, and the date of birth is taken in order to verify if the member is old enough to withdraw age-restricted resources. The database will also record whether the member is a student at the college or a staff member, as this affects the number of resources they are allowed to have on loan at any given time. This will be stored as an attribute, meaning both types of member will be stored in the same table on the database, furthermore meaning ‘a member cannot have the status of both a staff member and a student simultaneously’ **[13].** It is also assumed that (Table 1):

- Both staff and students are affected by the restrictions on borrowable time in the same way **[14]**
- Both types of members are subjected to the same fines and possible suspension due to overdue resources **[15]**
- Regarding the address, all members will have a UK address, which is subsequently possible to determine from the house number/name and postcode **[16]**

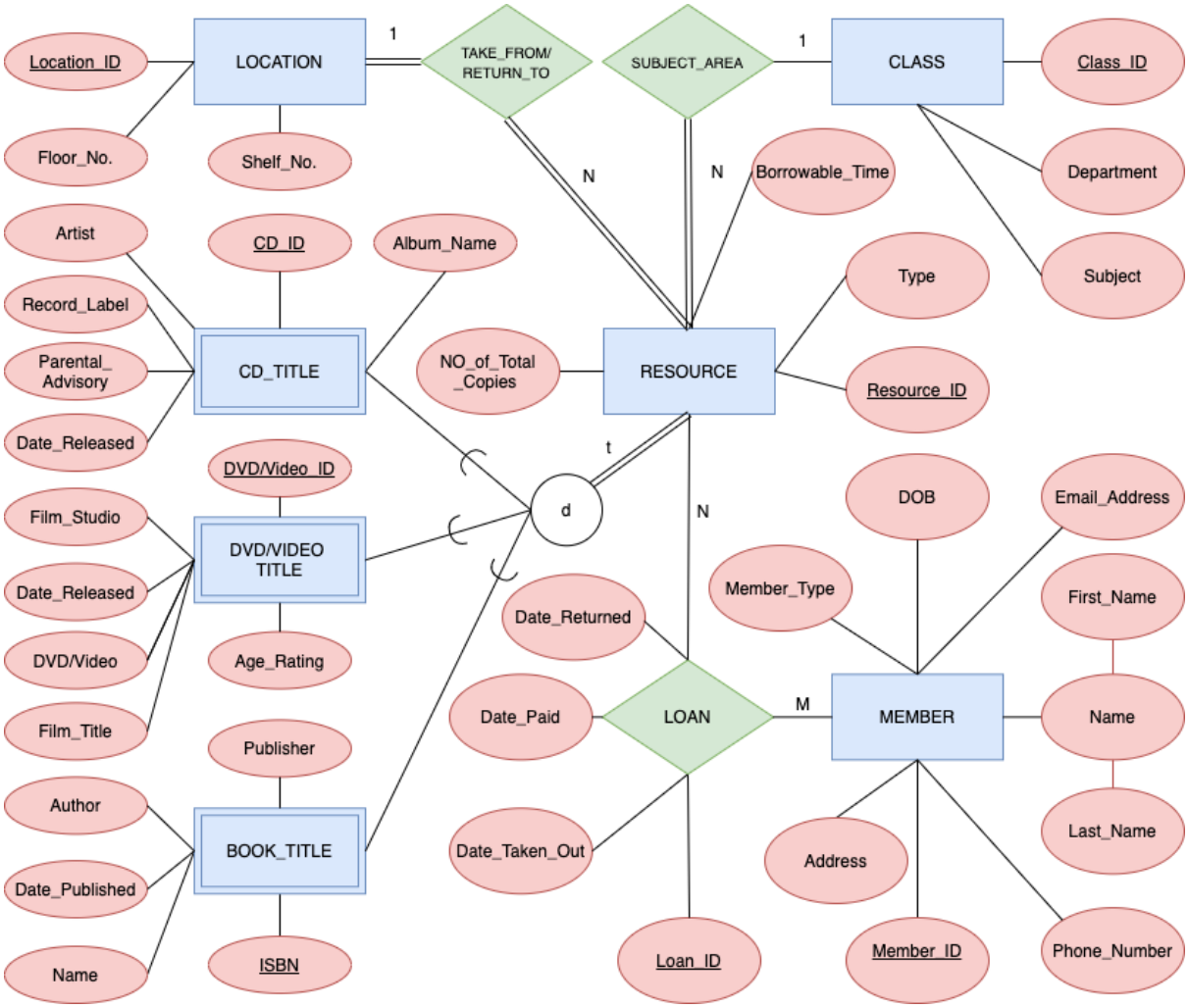
It was decided that ‘Loan’ could be placed on the diagram as the relationship connecting the ‘Member’ and ‘Resource.’ It has some attributes attached to it that will be used to describe and uniquely identity the loan. Essentially the ‘Loan’ relation only needs four attributes: the unique “Loan_ID”, and the dates the loan was taken out (“Date_Taken_Out”), the date the resource was returned (“Date_Returned”), and the date the fine was paid, if applicable (“Date_Paid”). From these attributes in combination with attributes such as “Borrowable_Time”, the due date, and the potential fine can be calculated. Regarding loans, it is assumed (Table 1):

- Reference books cannot be loaned by either students or staff, as these are meant to stay within the library - borrowable time is 0 in this case **[17]**
- A single loan will correspond to only one resource to one member at one time **[18]**
- Each copy of the same resource is borrowable for the same length of time **[19]**

Cardinality

The nature of the relationships between entities have been mapped onto the ER diagram as the following: ‘Resource’ to ‘Member’ is many to many, as members can take out multiple resources, and resources can be loaned to many members.
‘Resource’ to ‘Location’ is many to one as there are multiple resources that would be stored in the same location. It is assumed that: ‘every location that is stored in the database since has at least one resource assigned to it’ **[20]** (Table 1), so this is a total participation. ‘Resource’ to ‘Class’ is many to one as many resources will be for the same subject. This is a partial participation, i.e. not every subject will necessarily have a resource (though, every resource will have a subject). ‘Resource’ to ‘Resource Titles’ is a total disjoint specialisation.

Enhanced Entity-Relationship Diagram



Created with draw.io by authors.

Derived Attributes

In this Entity Relationship model the 'Fine' functionality is implemented as a derived attribute from the attributes of the 'Loan' relationship. This is because the amount due from a fine can simply be calculated based upon how long the item can be borrowed for, when it is taken out and the current date. The number of borrowable resources is also a derived attribute as it is totally dependent on the member type. Staff can loan up to 10 books at a time, whereas students can only borrow 5. There are other unincluded derived attributes such as the suspended status of a member. This is once again derived as the member becomes suspended once their fines total to $10, so it is simply calculated by the total of all their fines over all their loans. Using a simple database query, a list of suspended members can be found.

Relational Database Schema

It is now possible to map out the relationship schema from the ER diagram. In the following relations, the primary key will be underlined while any foreign keys will be italicised. Composite attributes are shown in square brackets as follows: Composite attribute [Part 1, Part 2].

Member Relation

The first relation to make concerns the 'Member' entity.

Member (Member_ID, Email_Address, Name [First_Name, Last_Name], Address [House_Number_or_Name, Street, County, Postcode], Phone_Number, Member_Type, DOB)

A unique "Member_ID" will be given to each library member. The member's email address will be the primary method of contact. While it is true that an email address would be a unique identifier, in normal operation of the library (taking out loans etc) employees would not need to know a member's email address. In fact, it can be argued that it is better that the email address is hidden from employees for privacy reasons, unless it is absolutely necessary to contact a member. Employees will refer to a member by their "Member_ID."Other attributes such as "Address" and "Phone_Number" are useful for contacting the member if required, while "DOB" is good for deriving the age for any age-restriction or benefits. "Member_Type" states whether the member is a staff member or student member.

Class Relation

The 'Class' relation is as follows:

Class (Class_ID, Subject, Dept_No, Department) .

"Class_ID" is a number that refers to a particular subject taught by a particular department in the college. It is assumed that there is only one department for each subject and that there is no targeted year group for a particular resource. Each department is assigned an integer.

Location Relation

The 'Location' relation is as follow:

Location (Location_ID, Shelf_No, Floor_No)

"Location_ID" is a unique number referring to a particular shelf on a particular floor. Floors are numbered and shelves are numbered in a sensible way such that members and employees of the library are easily able to find the required shelf on a floor when given the "Shelf_No." The "Shelf_No" numbering starts from 1 on each floor.

Resource Relations

Resource (Resource_ID, Borrowable_Time, Type, No_of_Total_Copies, Location*_ID, Class_ID)*

"Resource_ID" is a unique number referring to each resource of any type. Every copy of the same resource shares its ID. Each member borrows each resource for a "Borrowable_Time" which is the maximum period permitted for the loan. "Type" of resource refers to 1 of the 3 types of available resources ('Book', 'DVD/Video' and 'CD'). Each copy of the resource has the same "Resource_ID" and "No_of_Total_Copies" denotes how many copies the library owns. Every copy of a certain resource is borrowable for the same duration i.e. each copy of a certain resource is treated identically. "Location_ID" is chosen as a foreign key to obtain the whereabouts of each resource's physical location. This is useful for the library staff and users in replenishing or collecting the books from this point. "Class_ID" is an apt foreign key here in order to identify the subject to which the resource is relevant.

DVD/Video_Title (Resource_ID, Film_ID, Film_Name, Date_Released, Film_Studio_Name, Country_Name, Age Rating, DVD/Video)

"Film_ID" is assigned to each individual film. "Film_Name" is the name of the film. "Date_Released" and "Film_Studio_Name" all indicate further details about the resource on record. "Country Name" indicates the country of origin for the film studio(s) involved in film. The "Age_Rating" of this type of resource is important for checking whether a film is suitable for a member. "DVD/Video" states whether the resource is a DVD or a video.

CD_Title (Resource_ID, CD_Name_ID, Album_Name, Date_Released, Artist_Name, Country_Name, Parental_Advisory)

"Album_Name", "Date_Released" and "Artist _Name" all indicate further details about the resource on record. "Country_Name" indicates the primary nationality of the artist. "Parental_Advisory" denotes whether the album contains explicit content.

Book_Title (Resource_ID, Book_Name, Author_First_Name, Author_Last_Name, Country, Date_Published, ISBN)

"Resource_ID" is a unique number referring to each exclusive resource of any type. "Book_Name", "Author_First_Name", "Author_Last_Name" and "Date published" all indicate further details about the resource on record. "ISBN" is included in order to process specific requests from members for one resource with a global identifier (and not exclusively internal) against the database. "Country_Name" indicates the primary nationality of the author.

The specialist attributes for the three subtypes are useful for better sorting and searching criteria whilst querying the database.

Loan Relation

Loan (Loan_ID, Date_Taken_Out, Date_Returned, Date_Paid, *Member_ID, Resource_ID*)

"Loan_ID" is a unique number that refers to one resource borrowed by one member on a particular date. Both dates of resource exit/entry from/to the library are recorded through "Date_Taken_Out" and "Date_Returned."Each loan has a fine associated with it, if applicable. This amount is derived from "Date_Taken_out", "Borrowable_Time" and one of the current date or "Date_Returned", depending on whether the resource has been returned or not. "Date_Paid" denotes when a fine is paid, if applicable. As foreign keys: "Member_ID" and "Resource_ID" correspond to the borrower and the borrowed item.

Normalisation

Normalisation has been carried out below up to third normal form (3NF). The stages of normalising the relations are laid out below.

First-Order Normalisation

The concept of first normal form (1NF) dates back to 1971 when it was defined by Edgar Codd in a conference paper (Xiao et al., 2011). It is defined as: "The first normal form states that every attribute type of a relation must be atomic and single-valued. Hence, no composite or multi-valued attribute types are tolerated." (Lemahieu, vanden Broucke and Baesens, 2018)
This will be applied to the relations in the following section.

Member Relation

Member (Member_ID, Email_Address, Name [First_Name, Last_Name], Address [House_Number_or_Name, Street, County, Postcode], Phone_Number, Member_Type, DOB).

This relation (above) is not normalised in the 1NF. There is a composite attribute "Name."To put this in the 1NF, this attribute will be split into "First_Name" and "Last_Name.""Address" is also a composite attribute. It can be simplified to just the following two: "House_Number_or_Name" and "Postcode." As mentioned earlier, it is assumed that all addresses are in the UK where only these two attributes are required to obtain a full address. Storing address in this way is viable as it is rare that the library will need a member's address, as primary communication will be via email. The address can be obtained via an external database or module in the event it is required. Only one address is stored for each member. While it is possible that a member may have multiple phone numbers, the library will only ask for and store one phone number for each member. A member can only be one of students or staff and therefore the aforementioned attributes are all atomic. "DOB" is obviously atomic. The library only stores one email address per member, to ensure the attribute is atomic. So, in 1NF, the relation becomes:

Member (Member_ID, Email_Address, First_Name, Last_Name, House_Number or Name, Postcode, Phone_Number, Member_Type, DOB)

Class Relation

Class (Class_ID, Subject, Dept_No, Department)

As previously mentioned, each "Class_ID" refers to one subject and a taught subject is exclusive to one department, so therefore this is fully atomic and is already in 1NF.

Location Relation

Location (Location_ID, Shelf_No, Floor_No)

As previously mentioned, "Location_ID" refers to one shelf on a particular floor so this relation is already in 1NF

Resource Relations

Resource (Resource_ID, Borrowable_Time, Type, No_of_Total_Copies, Locatio*n_ID, Class_ID)*

A resource can only ever be placed in one location and is only for one subject therefore "Location_ID" and "Class ID" are atomic. "Borrowable_Time" in days can be one of 14, 2, or 0 i.e. a reference resource, so the attribute is atomic. "Type" refers to whether they are a book, DVD/video or a CD and is obviously atomic. "No_of_Total_Copies" refers to the number of copies the library owns which is obviously atomic. Therefore, this relation is in 1NF.

DVD/Video_Title (Resource_ID, Film_ID, Film_Name, Date_Released, Film_Studio_Name, Country_Name, Age Rating, DVD/Video)

It's assumed the "Film_Name" is atomic as it is the name given by the studio. As all the copies are of the same edition, "Date_Released" will always be the same for all copies of a resource and therefore is atomic in this relation. If there's any ambiguity regarding the Date released, the British release date will be used. "Age_Rating" is obviously atomic and the "DVD/Video" attribute states whether the resource is a DVD or video. As "Resource_ID" refers to a unique resource, even if the library had both DVD and video versions of a film, the "DVD/Video" attribute would still be atomic.

Multiple film studios can be involved in a film. Therefore, "Film_Studio_Name" is not atomic. It is assumed that film studios are based in one country. Therefore, some normalisation is required as follows:

DVD/Video_Title (Resource_ID, *Film_ID*, Film_Name, Date_Released, Age_Rating, DVD/Video)
Film_Studio (*Film_ID*, *Film_Studio_ID*)
Film_Studio_Name (Film_Studio_ID, Film_Studio_Name, Country_Name)

CD_Title (Resource_ID, CD_Name_ID, Album_Name, Date_Released, Artist_Name, Country_Name, Parental_Advisory)

It's assumed the "Album_Name" is atomic as it is the name given by the record label. "Date_Released" is singular as the library will only stock one edition. "Artists_Name" could be multivalued and "Country_Name", the main nationality of the artist, is linked to "Artist_Name." Therefore, this is not 1NF. The relation will be split as follows.

CD_Title (Resource_ID, *CD_Name_ID*, Album_Name, Date_Released, Parental_Advisory)
CD_Title_Artist (*CD_Name_ID*, *Artist_ID*)
Artist (Artist_ID, Artist_Name, Country_Name)

Book_Title (Resource_ID, Book_Name, Author_First_Name, Author_Last_Name, Country, Date_Published, *ISBN*)

The “Book_Name” is assumed to be atomic. “ISBN” is unique and atomic and will be used to determine which book it is. “Date_Published” is atomic but there could be multiple authors. “Country” is the main nationality of the author (only one value for nationality will be stored). Therefore, the relation is split as follows becomes:

Book_Title (Resource_ID, Book_Name, Date_Published, *ISBN*)
Book_Author (*ISBN*, *Author_ID*)
Author (Author_ID, Author_First_Name, Author_Last_Name, Country_Name)

Of course, there may be multiple tuples for the same “ISBN” in the “Book_Author” denoting multiple authors for one book. This is why this relationship consists of a composite primary key with Author ID. The same principle applies for “CD_Title_Artist” relation and the “Film_Studio” relation.

The final first order resource relations are below:

Resource (Resource_ID, Borrowable_Time, Type, No_of_Total_Copies, Location*_ID, Class_ID)*
DVD/Video_Title (Resource_ID, *Film_ID*, Film_Name, Date_Released, Age_Rating, DVD/Video)
Film_Studio (*Film_ID*, *Film_Studio_ID*)
Film_Studio_Name (Film_Studio_ID, Film_Studio_Name, Country_Name)
CD_Title (Resource_ID, *CD_Name_ID*, Album_Name, Date_Released, Parental_Advisory)
CD_Title_Artist (*CD_Name_ID*, *Artist_ID*)
Artist (Artist_ID, Artist_Name, Country_Name)
Book_Title (Resource_ID, Book_Name, Date_Published, *ISBN*)
Book_Author (*ISBN*, *Author_ID*)
Author (Author_ID, Author_First_Name, Author_Last_Name, Country_Name)

Loan Relation

Loan (Loan_ID, Date_Taken_Out, Date_Returned, Date_Paid, *Member_ID, Resource_ID*)

As mentioned previously, a member can only take out one copy of the same resource on a given day. Using a “Loan_ID” primary key that increases by one for each loan of a resource, this ensures that this relation is in the 1NF. Each “Loan_ID” only refers to the loan of one resource to one member on one given date, i.e. if a member were to loan multiple resources in one day, it would add multiple rows to the ‘Loan’ table with the “Loan_ID” increasing by one for each loan.

Second-Order Normalisation

A relation in second normal form (2NF) can be described as: “A relation R is in the 2NF if it satisfies 1NF and every non-prime attribute type A in R is fully functionally dependent on any key of R” (Lemahieu et al., 2018).

Therefore, 2NF requires all of the attributes of an entity to be fully functionally dependent on the primary key. “To avoid update anomalies in database schemas containing functional dependencies, 2NF was introduced by Codd in” (Arenas, 2009).

Relations

Member (Member_ID, Email_Address, First_Name, Last_Name, House_Number or Name, Postcode, Phone_Number, Member_Type, DOB)
Class (Class_ID, Subject, Dept_No, Department)
Location (Location_ID, Shelf_No, Floor_No)
Resource (Resource_ID, Borrowable_Time, Type, No_of_Total_Copies, *Location_ID, Class_ID)*
DVD/Video_Title (Resource_ID, Film_*ID*, Film_Name, Date_Released, Age_Rating, DVD/Video)
Film_Studio_Name (Film_Studio_ID, Film_Studio_Name, Country_Name)
CD_Title (Resource_ID, *CD_Name_ID*, Album_Name, Date_Released, Parental_Advisory)
Artist (Artist_ID, Artist_Name, Country_Name)
Book_Title (Resource_ID, Book_Name, Date_Published, *ISBN*)
Author (Author_ID, Author_First_Name, Author_Last_Name, Country_Name)
Loan (Loan_ID, Date_Taken_Out, Date_Returned, Date_Paid, *Member_ID, Resource_ID*)

Since all the above relations have a single primary key and are in 1NF, these are all in 2NF.

Film_Studio (*Film_ID, Film_Studio_ID*)
CD_Title_Artist (*CD_Name_ID, Artist_ID*)
Book_Author (*ISBN, Author_ID)*

As the above relations solely consist of composite foreign primary keys, which are in 2NF.

Third-order Normalisation

Lemahieu, Vanden Broucke and Baesens (2018) state that a relation is in the third normal form (3NF) if it satisfies 2NF and no non-prime attribute type of R is transitively dependent on the primary key. By transitive functional dependency, it is meant we have the following relationships in the table: A is functionally dependent on B, and B is functionally dependent on C. In this case, C is transitively dependent on A via B (“Third Normal Form (3NF) - Database Normalisation,” n.d.).

Member Relation

Member (Member_ID, Email_Address, First_Name, Last_Name, House_Number or Name, Postcode, Phone_Number, Member_Type, DOB)

This relation is already in 3NF as none of the attributes are transitively dependent on the primary key.

Class Relation

Class (Class_ID, Subject, Dept_No, Department)

The "Department" a specific "Class_ID" refers to is dependent on the "Subject" therefore this is not in 3NF. The relation is split into:

Class_Number (Class_ID, Subject, *Dept_No*)
Dept_Number (Dept_No, Department_Name)

Location Relation

Location (Location_ID, Shelf_No, Floor_No)

The above relation is already in 3NF as per method of definition.

Resource Relations

Country (Country_ID, Country_Name)

The reason for creating the 'Country' relation is to improve data storage efficiency as "Country_Name" is an attribute present in many other relations. It is already in 3NF.

Resource (Resource_ID, Borrowable_Time, Type, No_of_Total_Copies, *Location_ID, Class_ID)*

It can be assumed that resources with the same "Class_ID" are not necessarily found in the same location. This relation is in 3NF.

DVD/Video_Title (Resource_ID, Film_*ID*, Film_Name, Date_Released, Age_Rating, DVD/Video)
Film_Studio (Film_ID, *Film_Studio_ID*)
Film_Studio_Name (Film_Studio_ID, Film_Studio_Name, Country_Name)

The attributes “Date_Released” and “Age Rating” are transitively dependent on “Resource_ID” via “FIlm_ID.” Therefore, these relations can be split as follows:

DVD/Video (Resource_ID, DVD/Video, *Film_ID*)
Film (Film_ID, Film_Name, Date_Released, Age_Rating)
Film_Studio (*Film_ID*, *Film_Studio_ID*)
Film_Studios_List (Film_Studio_ID, FIlm_Studio_Name, *Country_ID*)

CD_Title (Resource_ID, *CD_Name_ID*, Album_Name, Date_Released, Parental_Advisory)
CD_Title_Artist (CD_Name_ID, *Artist_ID*)
Artist (Artist_ID, Artist_Name, Country_Name)

The attributes “Date_Released”, “Parental_Advisory”, and “Album Name” depend on “CD_Name_ID.” These relations are split as follows:

CD (Resource_ID, *CD_Name_ID*)
CD_Name(CD_Name_ID, Name, Date_Released, Parental Advisory)
CD_Artist (*CD_Name_ID, Artist_ID*)
Artist (Artist_ID, Artist_Name, *Country_ID*)

Book_Title (Resource_ID, Book_Name, Date_Published, *ISBN*)
Author (Author_ID, Author_First_Name, Author_Last_Name, Country_Name)
Book_Author (ISBN, *Author_ID*)

“Book_Name” and “Date_Published” depend on “ISBN” so these relations are adjusted as follows:

Book (Resource_ID, *ISBN*)
ISBN (ISBN, Book_Name, Date_Published)
Book_Author (*ISBN*, *Author_ID*)
Author (Author_ID, Author_First_Name, Author_Last_Name, *Country_ID*)

Loan Relation

Loan (Loan_ID, Date_Taken_Out, Date_Returned, Date_Paid, *Member_ID, Resource_ID*)

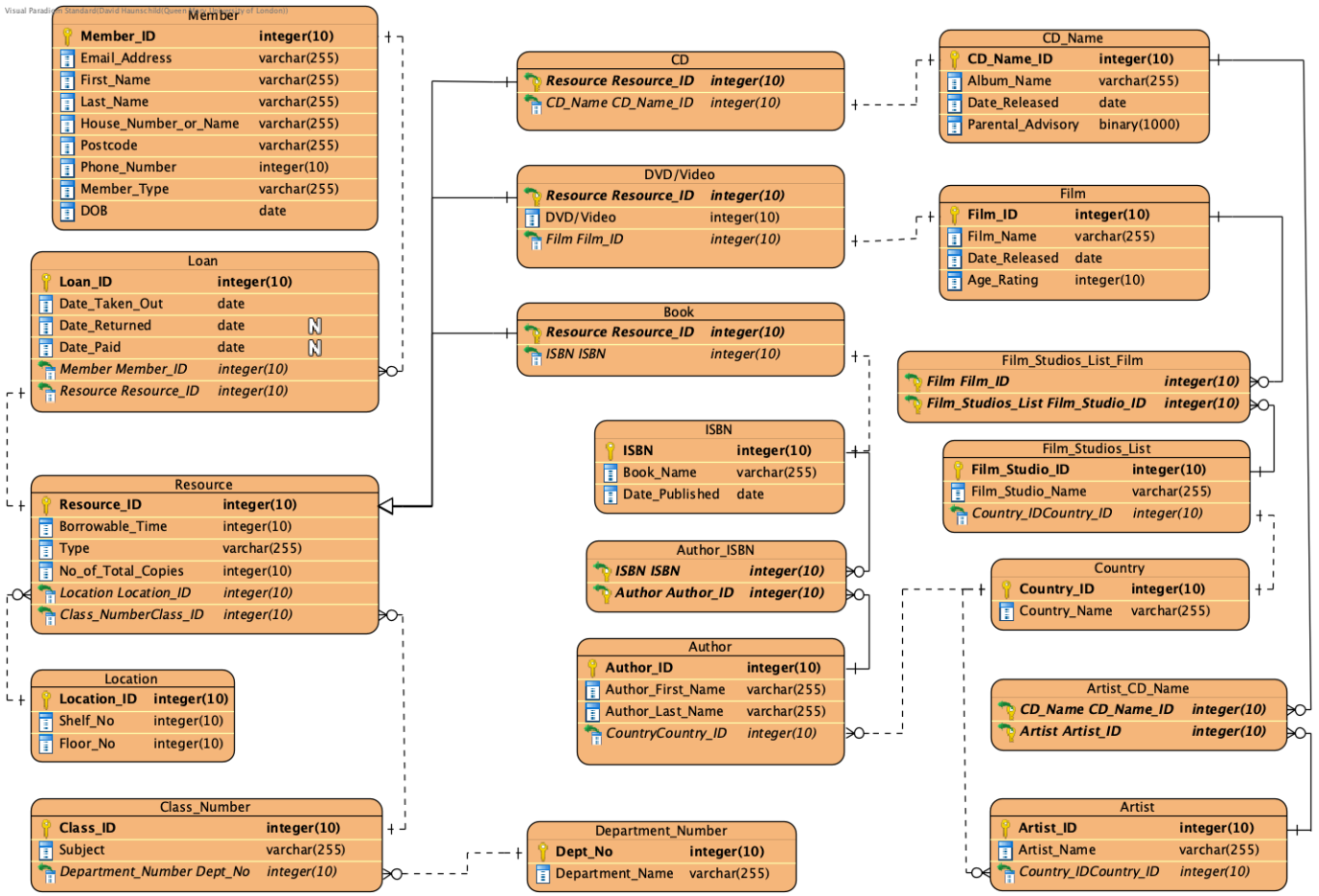
This is already in 3NF.

Final 3NF normalised relations

Member (Member_ID, Email_Address, First_Name, Last_Name, House_Number or Name, Postcode, Phone_Number, Member_Type, DOB)

Class_Number (Class_ID, Subject, *Dept_No*)
Dept_Number (Dept_No, Department_Name)
Location (Location_ID, Shelf_No, Floor_No)
Country (Country_ID, Country_Name)
Resource (Resource_ID, Borrowable_Time, Type, No_of_Total_Copies, *Location_ID, Class_ID)*
DVD/Video (Resource_ID, DVD/Video, *Film_ID*),
Film (Film_ID, Film_Name, Date_Released, Age_Rating)
Film_Studio (*Film_ID*, Film_*Studio_ID*)
Film_Studios_List (Film_Studio_ID, Film_Studio_Name, *Country_ID*)
CD (Resource_ID, *CD_Name_ID*)
CD_Name(CD_Name_ID, Album_Name, Date_Released, Parental Advisory)
CD_Artist (*CD_Name_ID, Artist_ID*)
Artist (Artist_ID, Artist_Name, *Country_ID*)
Book (Resource_ID, *ISBN*)
ISBN (ISBN, Book_Name, Date_Published*)*
Book_Author (*ISBN*, *Author_ID*)
Author (Author_ID, Author_First_Name, Author_Last_Name, *Country_ID*)

Normalised Relational Schema Diagram in Visual Paradigm



Created with Visual Paradigm by Authors.

Conclusion

Having been given the task to design a database for a college library, an ER diagram was designed based on what was deemed intuitive and necessary. Following this the relations were determined and then normalised to third normal form.

In summary, the seven entities initially chosen are considered sufficient to meet system requirements. Compared to the original relational schema, the normalised design resulted in some attributes being created and some being removed. This was done to ensure that attributes were atomic and there was no transitive dependency present, fulfilling third order normalisation criteria.

The normalised relational schema is shown in both text and diagrammatic form. With this current normalised design, all design requirements have been met. On top of this, it is very easy to add additional functionality such as tracking which employee assigns a loan or other

functionality as per the library's requirements. This design is ready to be implemented into a physical database system.

Bibliography

Arenas, M., 2009. Second Normal Form (2NF), in: LIU, L., ÖZSU, M.T. (Eds.), Encyclopedia of Database Systems. Springer US, Boston, MA, pp. 2522–2522. https://doi.org/10.1007/978-0-387-39940-9_1263

Lemahieu, W., Vanden Broucke, S., Baesens, B., 2018. Principles of Database Management: The Practical Guide to Storing, Managing and Analyzing Big and Small Data.

Li, Q., Chen, Y.-L., 2009. Entity-Relationship Diagram, in: Li, Q., Chen, Y.-L. (Eds.), Modeling and Analysis of Enterprise and Information Systems: From Requirements to Realization. Springer, Berlin, Heidelberg, pp. 125–139. https://doi.org/10.1007/978-3-540-89556-5_6

Stockman, T., 2020. Coursework 1 Specification.

Third Normal Form (3NF) - Database Normalization [WWW Document], n.d. URL https://www.1keydata.com/database-normalization/third-normal-form-3nf.php (accessed 11.5.20).

Xiao, H.Y., Zhou, P., Zhao, C.L., 2011. The Breakthrough of Database Design Normalization Theory [WWW Document]. Adv. Mater. Res. https://doi.org/10.4028/www.scientific.net/AMR.204-210.1615

Appendix I

Table 1: Assumption Key

Assumption Number	Assumption Description
1	Only titles of resources stored in the database are titles that are a library resource
2	There is no distinction between different copies of the same resource
3	All resources as defined by "Resource_ID" are of the same edition
4	Many resources can be loaned to many members
5	CDs are normal music albums/spoken words CDs
6	DVD/Videos are normal films and TV shows
7	There is no distinction between different year groups
8	The "Class_ID" refers to the department and subject the resource is for.
9	Any resource belongs to only one subject area.
10	There may be subjects that have no resources in the library
11	If the shelf and floor numbers are known, then that is enough information for the resource to be located
12	All resources with the same "Class_ID" are not necessarily found in the same location
13	A member cannot have the status of both staff member and a student simultaneously
14	Both staff and students are affected by the restrictions on borrowable time in the same way
15	Both types of members are subjected to the same fines and possible suspension due to overdue resources
16	Regarding the address, all members will have a UK address, which is subsequently possible to determine from the house number/name and postcode.
17	It is assumed reference books cannot be loaned by either students or staff
18	A single loan will correspond to only one item
19	Each copy of the same resource is borrowable for the same length of time
20	Every location that is stored in the database has at least one resource assigned to it

Appendix II

Assignment Description

This task is to fulfil the requirements for designing a database application in this first half of the coursework through: creating a conceptual schema in the form of an ER (or UML) diagram, deriving the relational schema from this ER diagram and subsequently normalising the relations in order to complete this design. In this case, the concept of a relational database schema is designed for a college library.