Individual Coursework Assignment 2

Name

Student Number

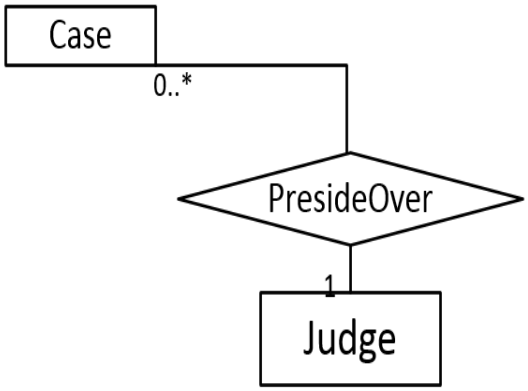
Date

Introduction

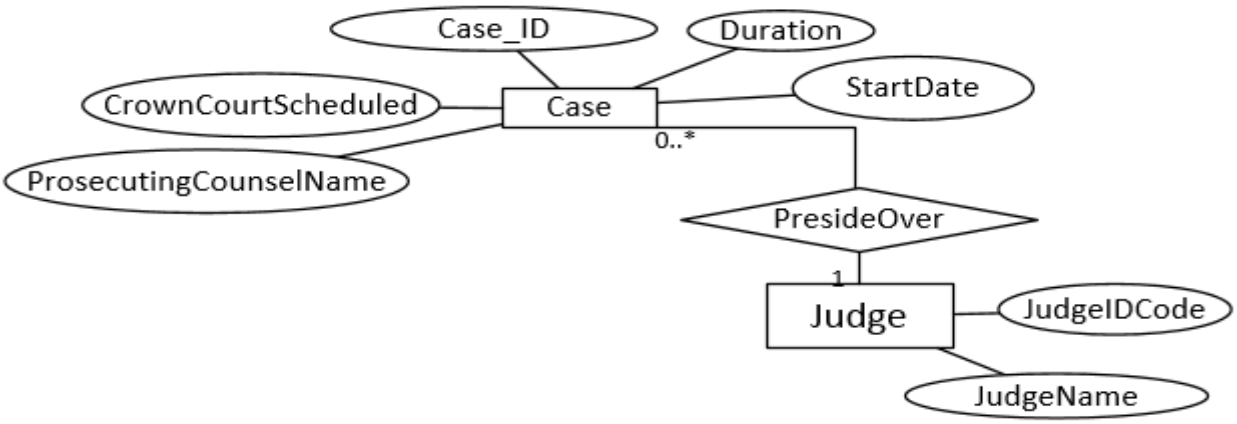
This paper presents the design of a database based to maintain the data on criminal court cases in the UK. The database is designed based on the information from the requirement elicitation document. The paper presents the Entity-Relationship Diagram (ERD), a determinancy diagram as well as a relational schema of the database. These models describes the structure of a relational database, enhancing the understanding of the structure of such databases as well as the relationship between the data within such databases (Dietrich, et al, 2015). How the requirements have been translated into the diagram has also been derailed below.

Entity-Relationship Diagram (ERD)

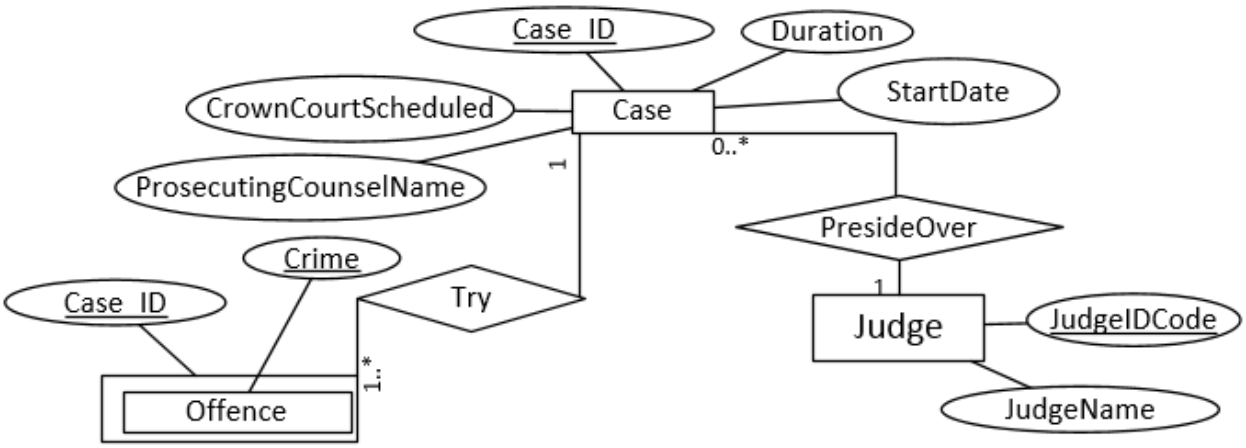
To examine the overall structure of the required database based on the provided information, an ER Diagram of the database has been produced. From the requirement elicitation document, each judge has a list of outstanding cases to preside over, but only a single judge presides over each case. It can be assumed that there can be scenarios in which a judge has no outstanding court case, like in the case of a newly posted judge. Judge is therefore an entity, with a "Preside Over" relationship with "Case", and a partially optional multiplicity as shown below.



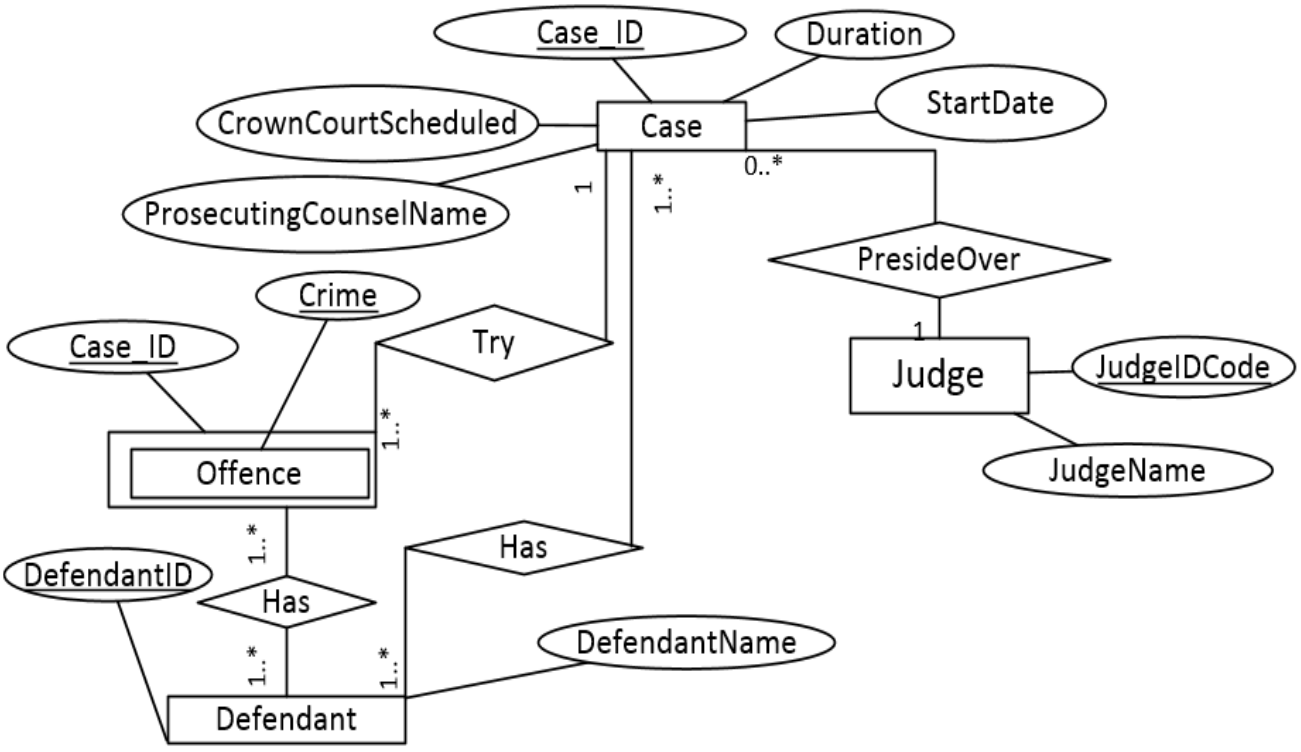
It can be assumed that each judge has a unique identity "Judge ID Code" with which they can be identified in the database. The database should also store the name of the judge as an attribute of the entity Judge. The requirements mention factors which can be designated as attributes of case, these includes the "Start Date" of the case, estimated "Duration" of the case, "Prosecution Counsel Name", and the "Crown Court" in which the case is scheduled. Prosecution Counsel has been taken as an attribute of the entity "Case" rather than an entity of its own because it bhas no other attributes and it has no any other relationship with other entities in the database. To facilitate the unique identification of each case in the database, the cases can be assigned "Case ID". The two entities with their attributes are as shown below.



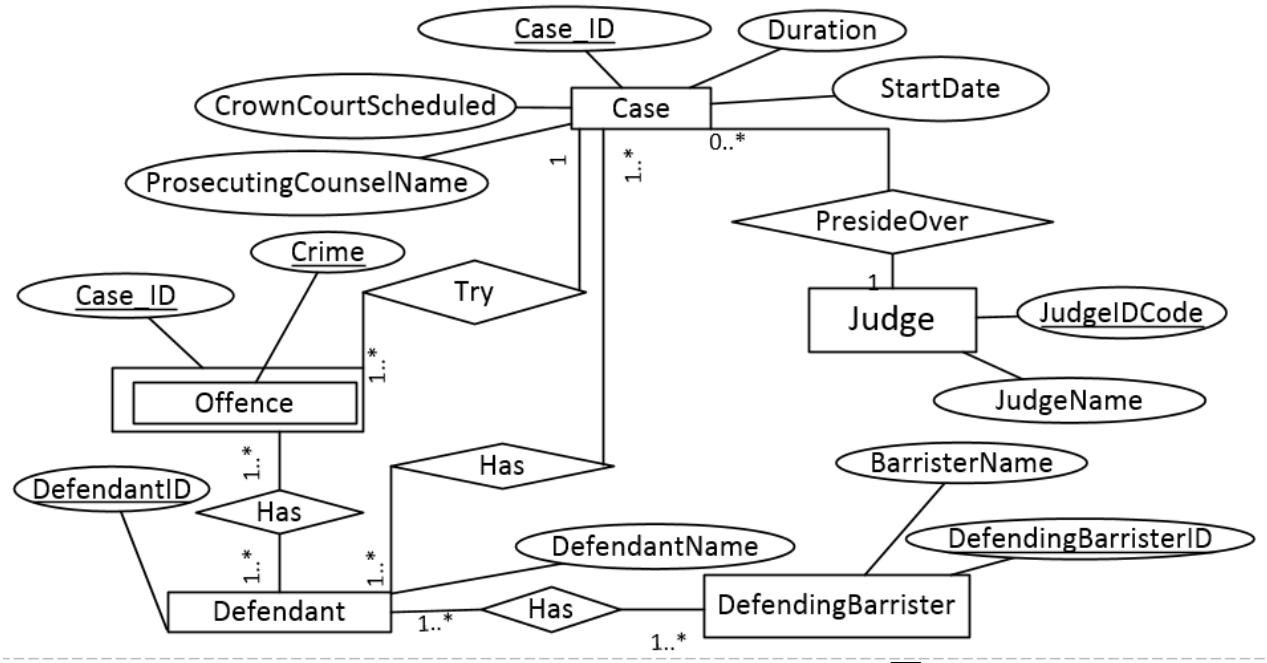
The requirement elicitation document also states that a case can try more than a single offence, and that each offence is for a single crime in law. Without a case, there would be no offence, therefore offence is a weak entity. “Case” has a “Try” relationship with “Offence”, and the relationship has 1 to many multiplicity. “Crime” committed is an attribute of the weak entity “Offence”. Crime, in this case, is taken to represent the law violated that makes an action of a defendant be considered an offence. The “Crime” and the “Case_ID” are the composite primary keys for the weak entity offence. The relationship and attributes of these entities can therefore be represented as follows:



It is also stated that each offence can have either one or more defendants, therefore the entity "Offence" has a "Has" relationship with the entity "Defendant". Since each offence can have one or more defendants, and it can be assumed that an offence cannot be brought to the court without the associated defendant, the relationship between these two entities in many to many. For identification purposes, each "Defendant can be assigned a unique identifier "DefendantID", which thus becomes the primary attribute of the entity. The DefendantName can also be store in the system for identification, thus this becomes another attribute of the entity. The relationship between this entity and other entities can be represented as follows.

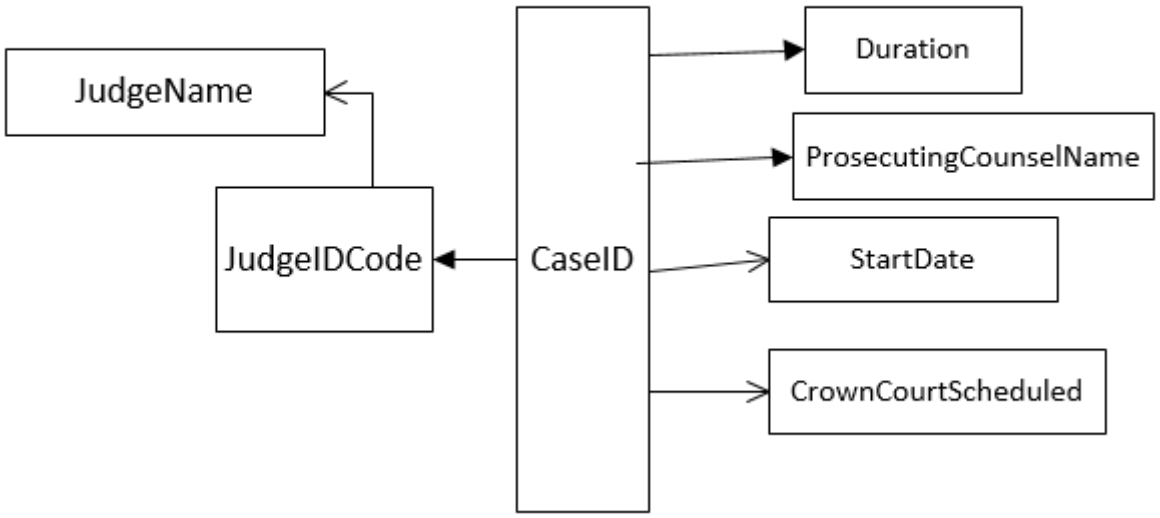


The requirement elicitation document also specify that each of the defendants can have either a single defending barrister or multiple defending barristers. The entity "Defendant" has a "Has" relationship with the entity "Defending Barristers" and the relationship has "many-to-many' multiplicity. It can be assumed that the Defending Barristers are identified in the database using their names and a unique identifier, therefore the entity "Defending Barristers" can be assigned the attributes "Barrister Name" and "Defending Barrister ID". Further it has been specified that the defendants may have more than a single outstanding case against them, therefore, this entity has a "Has" relationship with the entity "Case", and the relationship has a "many-to-many" multiplicity. The ERD for the entire database is therefore represented below.

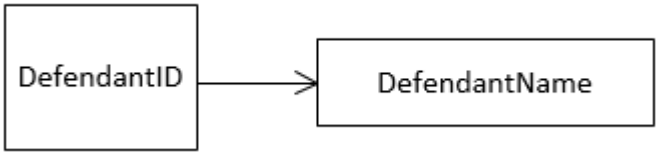


Determinancy Diagram

To determine the dependencies between the various data items in the data base, a dependency diagram of the database can be used. For a relationship with attributes (a, b….), attribute b is considered to be functionally dependent on the other attribute a, if and only if for each value of a, there is a precisely one b value at any particular time (Date and Darwen, 2007). Based in this definition, it can be noted that "Case ID" depends on "Duration", "Prosecuting Counsel Name" "Start Date" and Crown Curt Scheduled" because for each Case ID, there is precisely one of these attributes. These attributes can also be considered to be functionally dependent on case ID. Similarly, considering that each case is presided over by precisely one "Judge" there is also a functional relationship between "case" and "Judge ID Code". Further still, each Judge is expected to have a name, as such "Judge ID Code" depends on "Judge Name" This relationship is shown in the determinacy diagram below.



Although there is a “Has” relationship between the Case and the defendant, “Case ID” cannot be considered to have a functional relationship with “Defendant ID”. This is because, for every “Case ID” there can either be one or multiple “Defendant IDs”. Similarly, although there is a “has” relationship between the defendant and the “Defending Barrister”, the “Defending Barrister ID” is not functionally related to the “Defendant ID”. This is because for every “Case ID”, there is no precisely one Defendant, nor is there precisely one defending Barrister. “Defendant ID” however has a functional relationship with the defendant name, since each defendant is expected to have precisely one name. The Diagram below illustrates that “Defendant ID” depends on “Defendant name.



PAGE 7 REMOVED

Relational Schema for the Database

The structure of the database can be represented in the relational schema as demonstrated below. This involves translating the various entities in the database directly into schemas based on the information obtained on these entities from the ERD and the dependency diagram above.

To begin with, all the items that are functionally dependent on "Case ID" that have been identified in the dependency diagram above can be coupled together to form the entity "Case". By so doing, the various attributes of the entity "Case" are presented in the schema. This results in a

Case(CaseID, Duration, ProsecutingCounselName, StartDate, CrownCourtScheduled, JudgeIDCode)
Judge(JudgeIDCode, JudgeName)
Defendant(DefendantID, DefendantName)
Offence (Case_ID, Crime)

schema that represents the structure of the entity "Case" that is represented in the ER diagram above. Similarly, the schema of the entity "Judge" can be developed based on both the information in the ER diagram and the information on the Determinancy Diagram. Based on such information, "Judge ID Code" is the primary attribute of the entity while the only other attribute of the entity is the "Judge name". The relational schema for the four entities in the database are demonstrated in the figure below.

Conclusion and Recommendation

Relational databases are widely used in the storage and management of data, and will effectively help in capturing the data associated with the criminal cases in the UK. This paper has relied on the information obtained from the requirement elicitation document to develop an ERD, a Determinancy Diagram and a relational schema of the database. From the ERD, the database will require to have four entities and one of these four entities is a weak entity. Based on the provided

PAGE 9 REMOVED