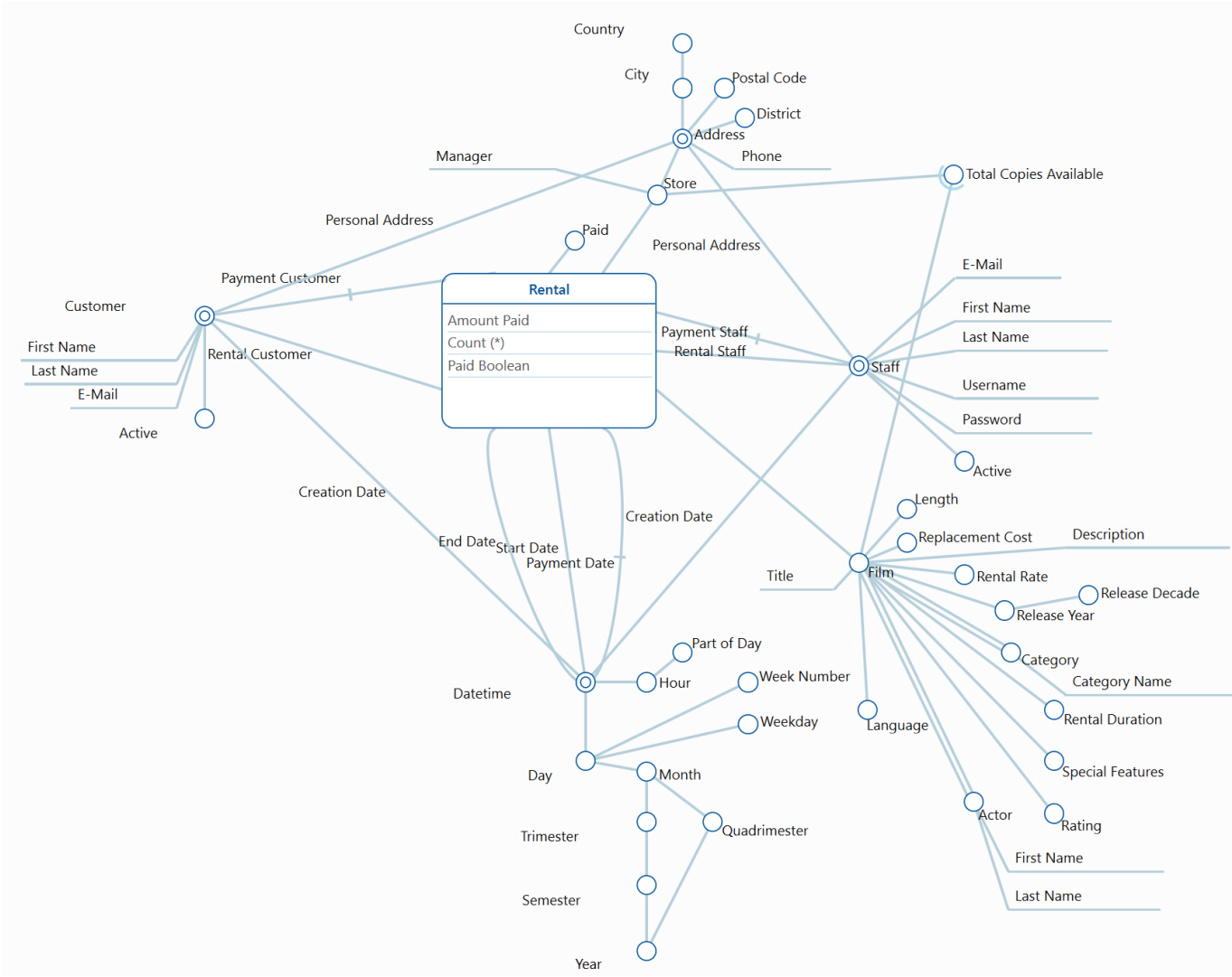
Data Warehousing Assigment - Part 1 Report

Abbas Al-Eryani 000439330
Geo Jolly 000439156
Alberto Parravicini 000437917
Almas Zhanibek 000439431

1 Dimensional Fact Model

Given the database specifications and the problem description, a Dimensional Fact Model (**DFM**) was created. The result is given in the figure below.

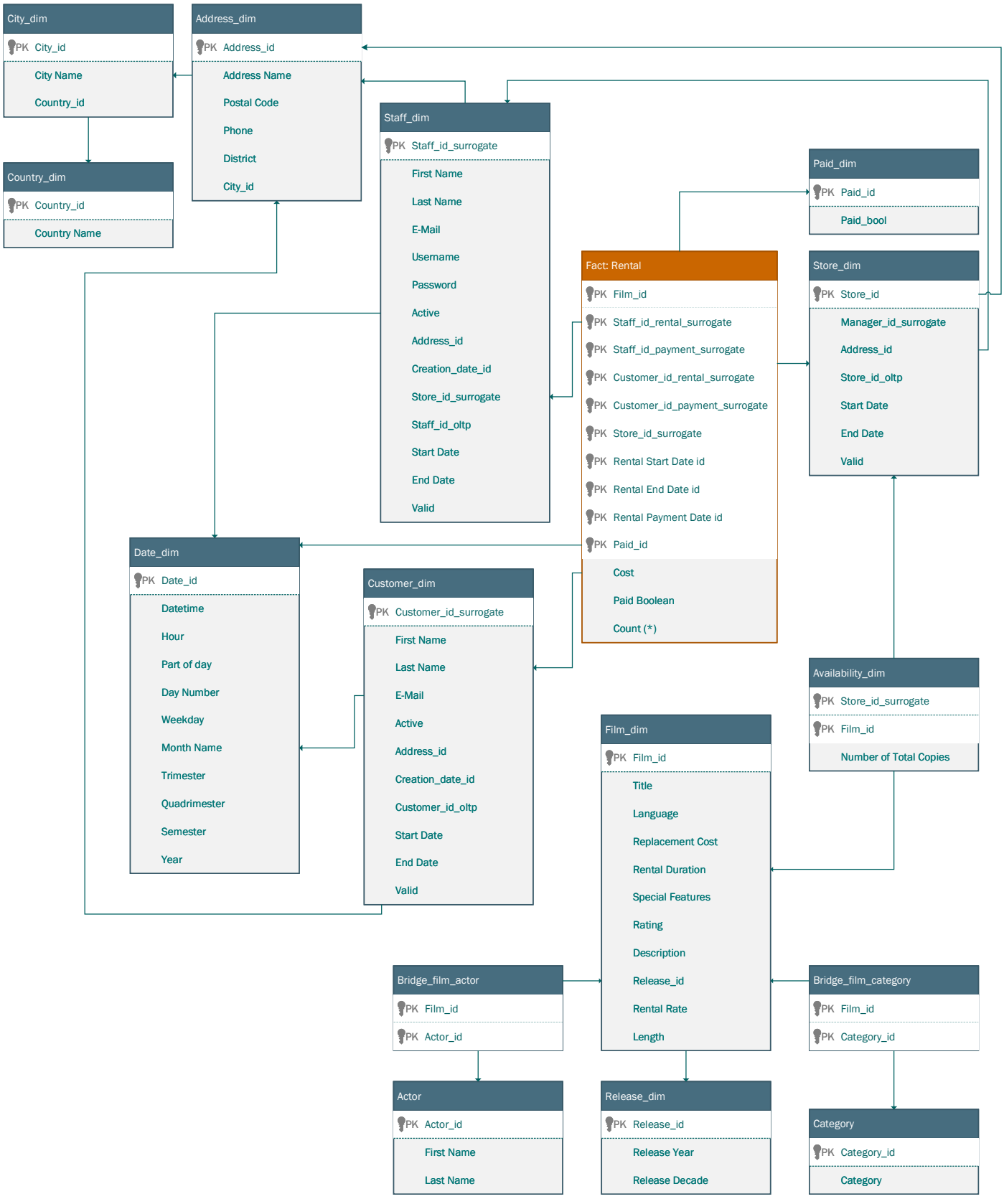


The *fact table* contains information about **Rentals** and **Payments**; from the original database schema, it is possible to see that the payment of a rental can be done at a different time than the rental itself. However, each payment refers to a single rental, so it is still possible to use a single fact table by making use of shared and optional dimensions, as explained below.

- **Measures:**
  - ***Amount paid:*** the measure contains how much a single rental was paid by a customer. As it is possible to have rentals for which the payment hasn't been done yet, it is suitable to have a specific value such as *"Not paid"* to handle this situation.
  - ***Count (*):*** this measure is simply used to count the number of rentals that are done, regardless of the payment. It could be used, for instance, to discover which are the most popular movies in each store or country.
  - ***Paid_bool:*** this boolean measure is used to distinguish between rentals for which a payment has been made or not. As an example, it can be used to measure the average payment rate for each store.

- **Dimensions:**
  - ***Paid_dim:*** whether a rental is paid or not isn't only useful as a measure, but also as a dimension, to analyze only the rentals that were paid.
  - ***Customer:*** it was decided to include many different descriptive attributes (such as *e-mail*) as the assignment requested to preserve all the information contained in the *OLTP* database. The fact table is connected to the customer dimensions though 2 different arcs: one for the customer who made the rental, and one for the customer that made the payment. The second arc is optional, as some rentals might not have a payment associated to them. Moreover, by querying the existing database, it is possible to see that the customer who made the rental and the one who paid might be different.
  - ***Staff:*** this dimension is structured similarly to *customer*. Once again, it is required to distinguish between the staff member related to the rental and the (optional) one related to the payment.
  - ***Store***, ***Customer*** and ***Staff*** are all connected to the ***Address*** dimension. It should be noted that the *District* doesn't belong to the hierarchy of *city* and *country*, as there are cities that belong to multiple districts and viceversa. It could be assumed that a district belongs to a single country, but from the given description it isn't possible to say so. On the same line of thought, it could be possible to build a hierarchy from *store* and *manager*, if it could be said that a manager was in the lead of multiple stores. This can't however be determined from the information at disposal.
  - ***Date:*** the basic element of the hierarchy is *datetime*, i.e. the full date with time and time-zone. From there, it is possible to infer the day, the month, etc.
    Similarly to *Customer* and *Staff* it is necessary to separate the dates of the rental and the one of the payment, which is optional.
  - ***Film:*** this dimension is linked to multiple others attribute, such as *Category* and *Language*; moreover, it is possible to build a small hierarchy made of *Release Year* and *Release Decade*. The *Special feature* attribute can be used to analyze whether having commentaries or trailers can impact the popularity of a movie. In practice, it would be better to convert the attribute to a set of binary attributes, so that it is possible to separate the single features from each other.
    The ***Total Copies Available*** dimension is used to keep track of how many copies of each movie are in the inventory of each store. This values aren't already in the *OLTP* database, but can be easily computed from the *inventory* table.

2 ROLAP Model



The *Dimensional Fact Model* was subsequently converted to a ***ROLAP*** model. The model was mainly built as a *Snowflake* schema, to reduce redundancy and make the relational dependencies look more evident. It was necessary to consider that some dimensions can change over time:

- *Store* is built as a type **2B** dimension, because the manager of the store can change over time and it is useful to keep track of the historical data.
- *Customer* and *Staff* are also built as type **2B** dimensions, as it can be useful to keep track of the *Addresses* over time. It is also interesting to monitor the evolution of *Active*, assuming that a customer can become active or inactive multiple times, like in the case of subscriptions. Fields such as *e-mail* and *username* can be considered as type **1**, and their values overridden without the need to insert new tuples.
- *Film* uses 2 bridge tables, to connect each movies to its categories and actors. There is also another bridge table used to model the number of copies of each movie owned by each store.
- The *Date* is not normalized for simplicity and ease of understanding, but in practice it can be normalized if lowering the redundancy of data is needed to save space.