Seminar 2 - Logical/Physical Model

Data Storage Paradigms, IV1351

November 2024

Project members:
[Adrian Boström, adrbos@kth.se]
[Viktor Sandström, visand@kth.se]
[Lucas Rimfrost, rimfrost@kth.se]

1 Introduction

After making a Conceptual model for Soundgood Music school in the previous seminar which creates a good baseline for creating a logic and physical model. The logical model is a key step to achieve a fully functioning data base in our physical model. As the school increases in size its requirements grow and a solid structure is in order for an efficient system.

2 Literature Study

In preparation our group attended lectures and looked at online lectures for arguments we forgot, thus creating a good baseline to start with the task at hand. In comparison to the conceptual model the logical model is not so different. The main difference is that the logical model is a lot more formally rigid, seeing as the implementation of the conceptual model is not concerned with the how and what of the database, it mainly gives a rough overview of the key requirements of a system.

Now with the logical model we will structure the database in such a way that we can achieve all the requirements given to us by the Soundgood Music School in a good way. This is done by implementing the theories of primary keys, foreign keys and a lot more dependency.

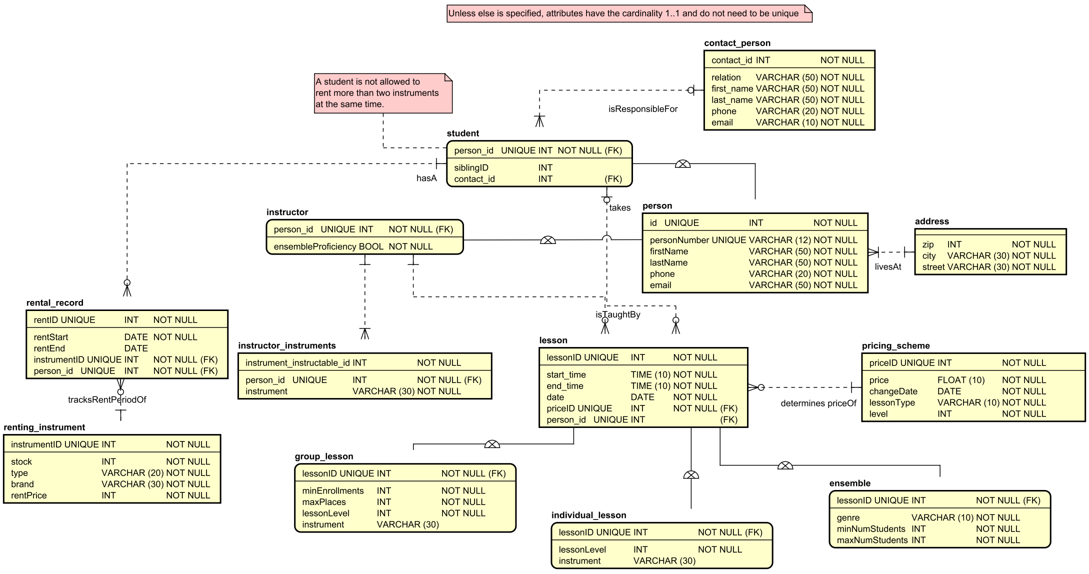
3 Logical model

The first step in achieving a logical model is by reviewing our conceptual model and adhering to normalize it and set primary keys and make inheritance crystal clear. When reviewing this we noticed a few issues with our current approach.

These were the issues we noticed:

- The payment for students and teachers can just be derived instead of its own table.
- Review which attributes can be NULL(for example siblingID)
- Have a relation made to hold history and current pricing scheme.
- Connect the previously mentioned pricing scheme to lesson instead so the price can be calculated through there due to inheritance.

When seeing these issues with combination with the new changes which need to be made like including data types and solving cardinality issues we know that we will need to make changes to our model. Astah Professional was used when improving the model and creating the logical model for the improvement. This was the improved model:



Now we see elements such as subclasses, primary keys and foreign keys. This fulfills the needs to now create a functional database independent of which DBMS we use.

This model is also quite normalized which means its a lot more data space efficient and removes redundancy. The system for an employee is pretty streamlined.

In the pricing scheme relation we can see change dates to allow for history of pricing changes such that we can charge the correct price for lessons which happened in the past.

The main challenges which came with making the logical model is dealing with derived data and finding a balance in normalization and practically. This is due to the fact that as you subdivide a database the more abstract and ambiguous.

We also struggled as a team to communicate our ideas on how we thought the database would look like and discussing pros and cons with each approach. In the end we have an approach which satisfied the whole team and met all the requirements.

4 Physical model

For the physical model on the hand its very hands on in terms of its approach, we chose to use PostgreSQL as our DBMS to implement it.

When converting the model to a physical one we need to decide the order which they are created due to constraints of foreign keys and connections.

For keys we need to ensure their uniqueness, otherwise the system does not work.

When solving issues with primary keys/ foreign keys it becomes apparent that they form a sort of web shape in their inherent dependency on each other.

When comparing to the conceptual model it is clear that this one is a lot more suitable to implement on a database and made it a lot easier to actually start writing the SQL code.

We generated code with the help of online tools but also some hand made data, This includes 3 teachers, 3 students, one lesson and a pricing scheme.

5 Discussion

There were a number of interesting difficulties and educational opportunities during the process of turning our conceptual model into logical and physical models. This conversation will examine our design choices, their effects, and possible substitutes.

5.1 Design Decisions and Challenges

Finding the ideal balance between normalization and usefulness was our main obstacle. We had to make certain practical choices even though our goal was to preserve appropriate normalization in order to get rid of repetition. In order to simplify queries and preserve data integrity, for instance, we decided to use change dates in the pricing scheme instead of distinct history tables.

5.2 Normalization Trade-offs

Our normalization strategy tried to find a balance between practical requirements and data integrity. We carefully considered where strict normalization could limit the system's usability, while we worked toward appropriate standardization to remove redundancy and preserve data consistency. The pricing scheme relation serves as an example of this balance; instead of constructing distinct historical tables, which would have achieved higher normalization at the expense of more complicated queries and possibly decreased performance, we preserved historical pricing data through change dates.

5.3 Implementation Insights

We ran into a number of technical difficulties when implementing it physically. Careful planning of the creation sequence was necessary to manage the interconnections across tables, especially for linkages involving lessons, students, and instructors. To maintain system stability, extra care had to be taken while handling NULL values, particularly in situations like sibling relationships.

6 Conclusion

The process of creating this physical and logical model for Soundgood Music School has been instructive and has brought to light the significance of careful database design. Our difficulties with normalization, dealing with NULL values, and putting historical data monitoring into practice gave us real-world database development insights that exceeded theoretical understanding. There is always space for improvement as the system grows, even though our current implementation satisfies the needs of the school and offers a strong basis for development in the future. The finished model is well-suited to Soundgood's requirements since it achieves a good compromise between ensuring data integrity and enabling realistic commercial processes.