Nicholas Catani

THE MAKING OF A DATABASE

TABLE OF CONTENT

BRIEF INTRODUCTION

Sub Pop is a record label founded in 1986 by Bruce Pavitt and Jonathan Poleman. The company achieved fame in the early 1990s for signing Seattle bands such as Nirvana, Soundgarden, and Mud honey, central players in the Seattle sound movement. After years of success, Sub Pop wants to change the use of pen and paper to store records and file contracts, thus, the owners are looking for something futurist or alternative to replace with. As an independent database engineer, the idea is to construct an intuitive, individual database using access that collects all the records in one single bucket and walkthrough the company to its information technology standardization & compliance.

BENEFITS ANALYSIS

Before digging into the ERD I've designed, perhaps is necessary to give solid reasons why the company should move to use a database management system.

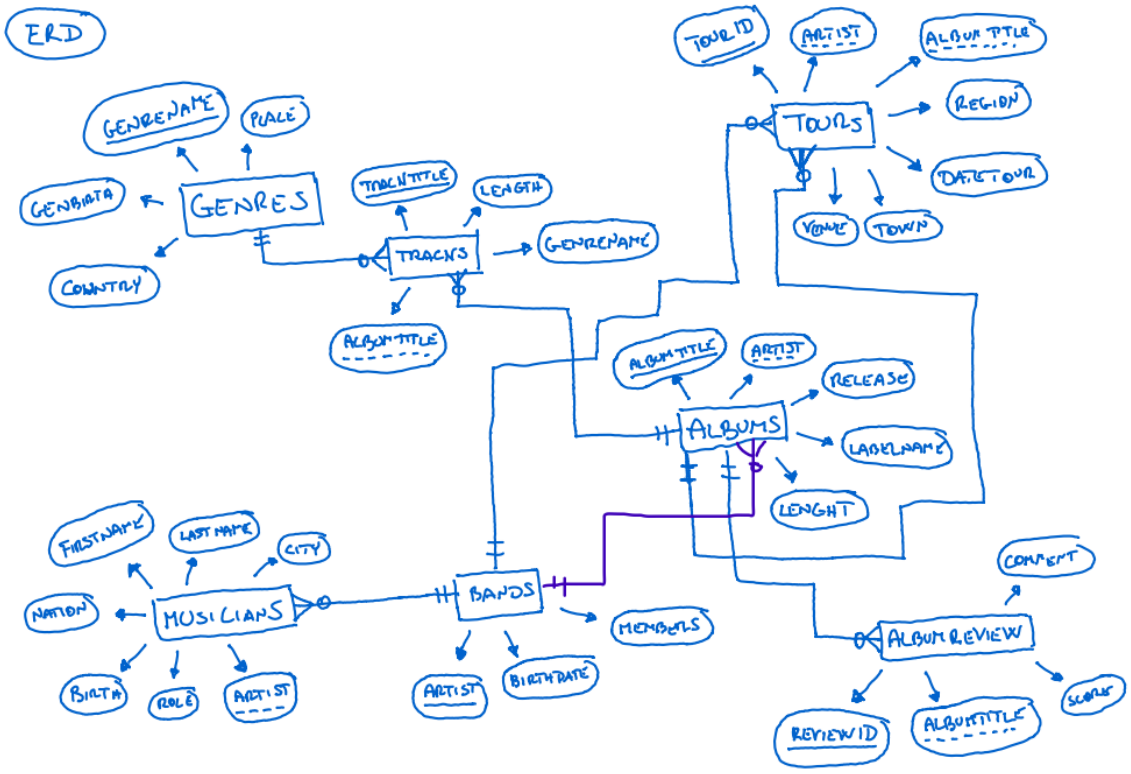
Data security is open-ended. With a database in place, the company can encrypt files and restrict data access to them. Also, it can assign to different accounts special sign-on procedures with encryption to protect sensitive data while logging in. I think it is perfect for GLBA-compliant offices, or anyone with sensitive information to protect. – for not experts, the GLBA compliance requires that companies develop privacy practices and policies that detail how they collect, sell, share, and otherwise reuse consumer information. On top of that, consumers must be given the option to decide which information, if any, the company is permitted to disclose or retain for future use.

A database can allow data backup and disaster recovery, something unachievable if operations are still written on a piece of paper. What if someone spills water or café on an important document? Or it has been shredded accidentally? A database is designed to create backups in different locations so that in case the data is stolen, ransomed, or corrupted, Sub Pop will have that extra copy. As regards disaster recovery, it offers recovery from corruption caused by a storm or a server breakdown.

Data integrity is the result of the two above-mentioned benefits, but also it can be achieved by reducing data redundancy. What it means is that the data itself is accurate and consistent. – not inaccurately entered. A database can achieve so by filtering out human errors and making it easier to detect further errors – Just to mention, there's a rule called Referential Integrity which protects relations between tables from bad data.

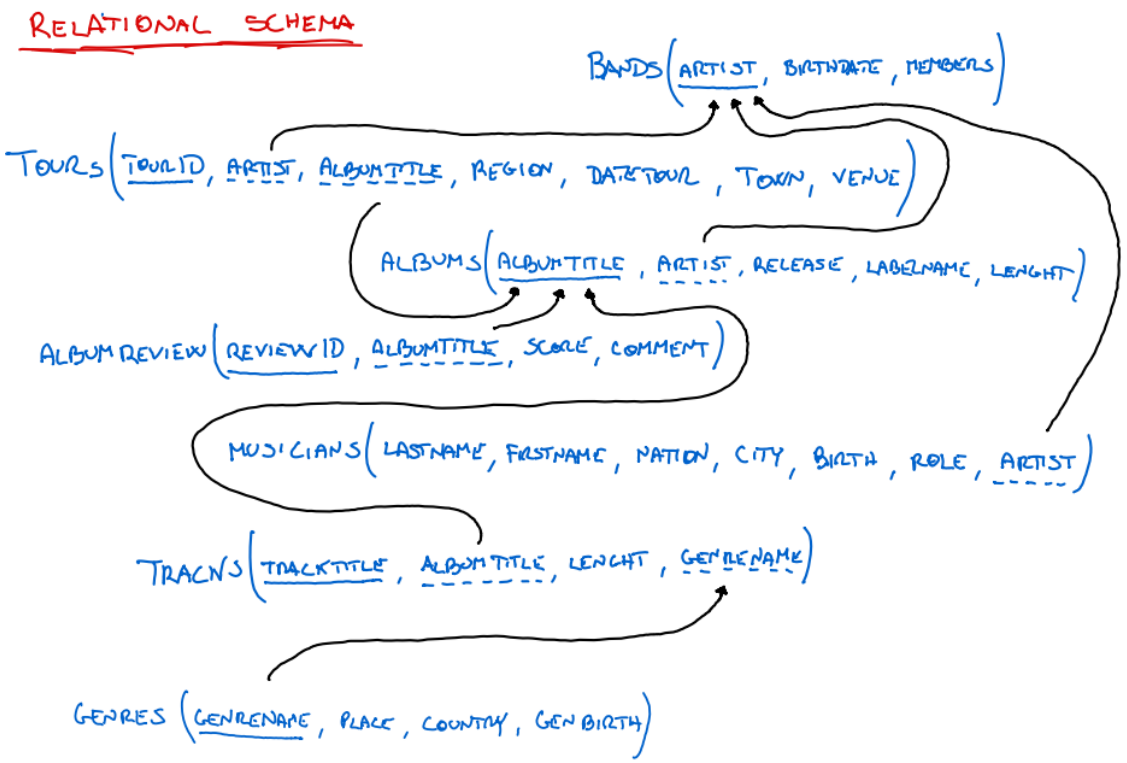
When the data is so organized and structured, it is facile to answer questions related to the information stored. Using SQL, the user can retrieve what he/she is looking for at lightning speed. In addition, a database can be designed to reduce redundancies in the data. Helps to avoid duplicates. Sub Pop might use this feature to its advantage since needs to pull forms, reports, and mailing lists. If implemented, the company workflows will be faster and more efficient, which consequently will improve decision-making processes.

ENTITY RELATIONSHIP DIAGRAM



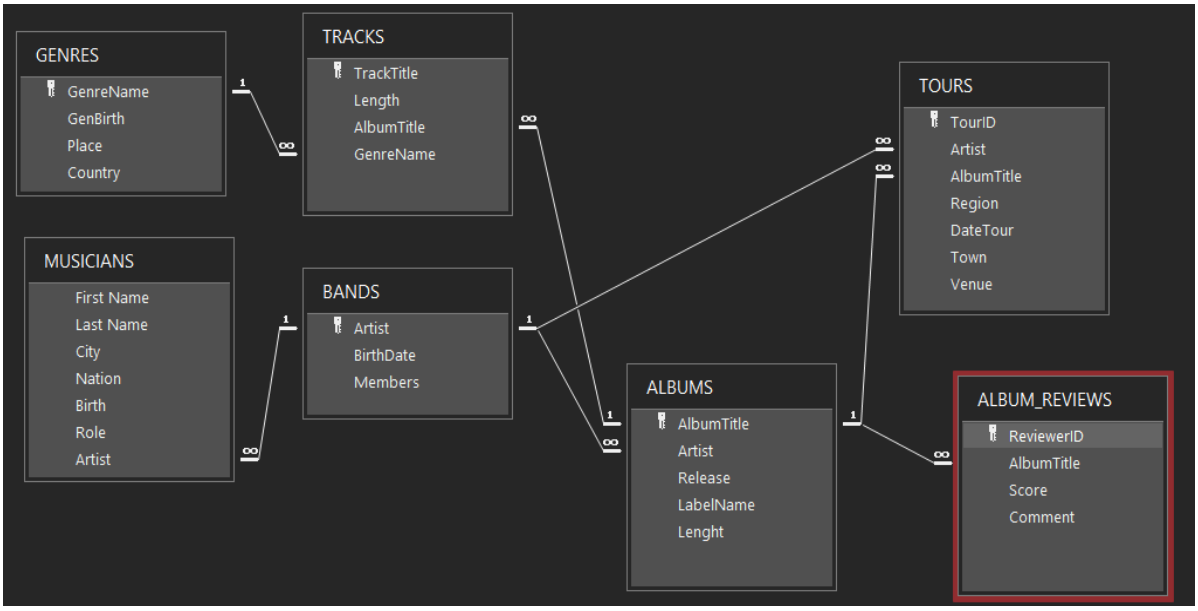
- PDF file bundled in the final submission.

RELATIONAL SCHEMA



- PDF file bundled in the final submission.

CONCEPTUAL TO PHYSICAL



The ultimate database contains 7 tables designed to suit SUBPOP's needs to store band data. It contains more than 500 track records and 80 albums from 25 top bands.

WHAT DID GO WELL?

One thing I recall is perhaps I had enough time to fill the tables from scratch without using a pre-packed dataset by loading a comma-separated value file. After having designed the entity relationship diagram and the schema, I spent the first 10 days of this project focusing on data entry. Since we are talking about a music database, it has not been that difficult to research and collect the necessary data to get the job done. The experience brought me to a point where I asked myself, where did professionals use to retrieve their data before Google or Wikipedia became so popular and accessible to everyone?

The second thing is probably the chance I had to have a clear idea in my mind since the beginning of the project. I think the ability to identify and articulate a neat vision is one of the key ingredients to carry out it until completion. The vision creates a congruence between the actual project deliverables and the strategic objectives of the project itself. In addition, it helped me to stay focused on keeping up with the plan and to mitigate the waste of time that would have occurred because of indecision.

WHAT DID GO WRONG?

When it comes down to data collection, retrieving correct information can be tedious. I think data inaccuracies are common, and this can make it difficult to get the most out of the valuable data I've collected. Unfortunately, the database contains missing and incomplete data due to a lack of information from the source. I'm experienced enough to understand that could lead to poor data analysis later down the road as the records in the database grow. Without sufficient data, analyses may have to be conducted with assumptions or extrapolations that will be less accurate, impacting the business decision-making processes.

I've found difficulties in implementing a macro feature that extracts the number of concerts made by a preselected musician "considering the band where he belongs" from a form. I tried several times to pull it off using the formula =forms!().text.value but without any success.

WHAT WOULD YOU DO DIFFERENTLY?

I would like to try a different software such as MS SQL Server for a bunch of reasons. First, Access can only support a small amount of traffic and may still return errors when the users try accessing the database simultaneously. Whereas SQL Server is designed for enterprise applications and handles hundreds or even thousands of users concurrently. From the SUBPOP perspective, the latter gives more scalability/reliability for employees and customers. Second, I figured out that MS Access is a good option for smaller applications, as it offers plenty of features regardless of its size – the creation of reports and forms, and updating data through DML queries, and macros. In contrast, MS SQL is a more robust database solution that supports advanced features critical for companies such as SUBPOP – reporting service, R server, Data Quality service, or Analysis service, just to mention a few.