Saaif Ahmed

saaifza2

CS:598 Foundations of Data Curation

Assignment 1 Part 1

1) For each of the files the best way to remark on the overall quality of the data is to describe them with a series of positive and negative remarks.
   a. File A: This file contains information regarding the cards such as model, year, and MSRP. This is uniquely row-defined by the VIN numbers in the second column of this data. Primarily beginning with the positives. This file is formatted as a ".txt" file which is at the very least easily readable and parsed by most modern programming languages. Therefore, if this file was sufficiently large, cleaning, editing, and storing the data and data revisions would be quite easy. Moving onto negatives, there is no schema description of the data present in the file. Only through knowledge of what generally is involved in car information is anyone able to understand what the columns refer to. There is seemingly duplicate data in row 1 as AWD and 4WD are present, but the AWD portion refers to model name. Row 2 has extra data on the engine displacement which no other car has, making the storage of the data, and the readability of the data extraordinarily difficult.
   b. File B: This file contains information about the sales made by the company, such as MSRP, car, and customer. This is uniquely identifiable by VIN and the sale date. For the positives of the file. The data is organized and formatted mostly cleanly. Opening in Excel it's easy to identify what's value applies to what column. As opposed to File A the naming convention appears to be consistent throughout the file, which is a big positive. Any missing data can be added in without any confusion on how to go about it. Lastly this is a .csv file so it is again useful and synergistic with programming languages. The negatives of this file are as follows. There is a significant amount of missing data. Some examples are missing states, MSRP values, and purchase prices. If this data set was sufficiently larger it would be an immense sink of funds in both money and time to fine-tune the data set. Another interesting part to note is that there are no default values and/or placeholders. This is fine but depending on the technology the data is meant to interface with, null values may not be accepted.
   c. File C: This file contains information surrounding customers, their professions, and other personal information relevant to the dealership. The positives are this file are few yet profound. Each data entry has all the information present for each entry. In a realistic scenario this is still very feasible as the amount of information is few per entry. There is a decent number of negatives to be had. Primarily the document type is the first type that isn't sufficient for programming as it is a .docx file. Word Documents are not easily interpretable by programming languages, making the data harder to verify, analyze, and manipulate. In addition, there are inconsistencies in the formatting making it harder to analyze through human interpretation. The last point to mention with regards to the data is the presumably "additional information" section for each customer. This is not

standardized at all, meaning that a schema for this needs to be developed if the data wants to be preserved and presentable in any fashion.

Lastly, for overlapping data between files. Names and Addresses are in B and C. VIN numbers and most car information excluding sale information are in A and B. Detailed car information is only in A, detailed sale information is only in B, and detailed customer information is only in C.

2) Refer to Excel workbook file.
3) Sheet 1 is the schema page. Sheet 2 has example tables. In the example tables **bold** refers to a primary key and *italics* refer to the foreign keys. On the schema the foreign key links are described.
4)

The schemas were made in this manner to support what seemed to be the company's main objective in their previous attempt at data management. Those being sales, customers, and inventory. Therefore, the databases were made as such with Car Information, Sales Information, and Customer Information. The attributes in the relational schema were designed in a manner that by asking any question starting in one table you could reasonably and easily answer the question by referencing the other tables. Each table has their set Primary/Primary Composite keys that facilitates the independence from row to row. The unique identifier among all the tables is the VIN as each vehicle must have a unique identifier. From that we can figure out the owner, the sale information, the car information. Adding additional values for the composite keys in the table allows us to easily search through the tables without having to reference others for common questions. For example, if you wanted to ask how many cars have a MSRP over 25,000 that is easily figured out.

No information was left out of the files. All information was storable, excluding duplicates. The reason for not excluding anything is directly in line with the principles of data curation. This information is now readily available for re-use, retrieval, and maintenance.

With Table 1 the reason for choosing the attribute and key values were as follows. To start, the VIN had to be included as there is no way for that to be duplicated among vehicles in circulation. Without it, two people with the same name could produce duplicate rows in the Customer Information table. The information should only be stored for completed sales and record keeping in addition to follow-up purchases. The other parts of the primary key for Table 1 are the first and last names of the customer and the middle initial. The attributes chosen for Table 1 are all information that is relevant to the customer's relationship to the dealership. Address, profession, and a comments space allows for heightened analysis when the time dictates it necessary.

Table 2 is the Car Information table and as such obviously has the VIN as the primary key. Otherwise, the information on the attributes in the table is largely there to support curating data surrounding each vehicle in inventory. Attributes such as maker, model, ear, engine, doors, MSRP are all obvious inclusions. Other columns can easily choose NULL as their value in a default scenario as their importance is limited to the overall needs of a dealership.

Table 3 is the Sales Information table and follows the previous tables where the VIN is part of the primary key as it is the only way to ensure uniqueness among the table's rows. However, the sale date must be included in the primary key as well because there is a possibility that the dealership sells the same car twice. Having the date column allows for that uniqueness to be preserved. There is a foreign key relation to the Customer Information table through the name of the customer and its corresponding columns. This is because you cannot have a sale without a customer, thus ensuring the integrity of sales throughout the history of the dealership. This means that the customer will have to be created first in the database, which is a dependency. The alternative scenario, and line of thinking where you cannot have a sale without a customer does mostly hold. However, this would allow for sales to be recorded without customers, which is a liability. Other attributes in this table are all necessary for the sale such as the sale value and the MSRP to calculate profit.

The hardest decision to make in this process was identifying what can and cannot be a foreign key. The foreign key constraints a necessary layer of redundancy and protection to the data, however the entire data editing process must be identified prior to setting anything in stone. There are no foreign keys in Table 2 as the dealership should be able to add cards to inventory without any dependency hardships. The dealership should also be able to add customers to the Table 1 data system without any concern. And with a customer in the system, a sale can then be logged and stored into Table 3. The two requirements for Table 1 and 2 are what led me to this line of thinking and this design decision. Initially it seemed desirable to maximize the number of foreign keys for optimized queries, however the integrity of that system had contradictions to the needs of a dealership.

Data independence is twofold, physical, and logical. The schema created supports data independence in both manners. Physical data independence is stored by the relational model of the schema. Because the model is simply a mapping from the data to the value it represents, so long as we can analyze the physical data the logical mapping can be adjusted to fit its needs. The relational model, by definition, is indifferent to the physical storage and processing. Thus, supporting the data's physical independence. Logical data independence is preserved by virtue of the solutions to physical data independence being the same as logical data independence. If the dealership decided to delimit information in a different manner, the data retrieval logic would have to be adjusted but the data itself in the relation model would not change at all. The integrity of the data is preserved eternally through the relational model.

The schema does support the overarching goals of data curation in numerous ways. To begin take discoverability as an objective of data curation. Developing queries through the relation model is made incredibly easily using keys and attributes. For example, you can develop a query between the Car Information Table and the Sales Information table to determine how many cars in inventory made before 2020 sold after 2020 through use of the attributes: VIN, Sale Date, and year. Other queries can be created in a similar fashion. Organization is another example of a data curation goal that this schema supports. The schema has an outline that defines the tables, attributes with descriptions, and their data and key types. This is a record of changes to the schema thus maintaining the standard for data curation. The storage of this data, as it is a relational model, can be handled by any SQL service for example. This handles the storage goal of data curation. Lastly these tables are designed with the sole purpose of solving real world problems for a real-world dealership. All the relevant information is stored for each table making it easy to use the semantics to answer specific questions. Additionally, the setup of the attributes and primary keys allow for easy traversal across the tables as one can use the VIN number to determine uniqueness among all tables.

There is much to be praised and criticized about the schema design, and those will be outlined here. The negatives surrounding this design are as follows. The lack of extremely efficient foreign key design is a liability. While the queries themselves can be sufficiently fast, there is room for improvement in their speed. Designing a schema that can hold its standards of response times as the database grows larger is good future proofing. Not to mention it's also related to the communication of the data. Furthermore, some of the tables are large in attribute numbers. With these large attributes queries must exclude lots of information which can make data processing harder than desired. Either combining attributes or creating more tables with more specific purposes can be beneficial to the long-term usage of the schema. However, there are positives to this schema's design that still allow it to be sufficiently useful. Primarily the design of the schema allows answering very common business questions incredibly easily. For example, imagine the dealer wanted to know how many customers in Chicago bought a 4 door Toyota vehicle in 2020. This a question that traverses across all three 3 tables but does so very easily. The retrieval of this information is as simple as writing a query for these exact 3 attributes. In addition, that query writing can be optimized to increase performance. In addition, all the data from the previous files were able to be stored effectively within the schema. Therefore, integrating this system into the dealership will be just as easy. Anyone looking for specific data they had stored can use a logical query to retrieve that information.

To sustain the database for future discovery and new purposes there are curational activities that can be done. Starting with policy and policy creation. The data was not presented in a fashion that matched any reasonable, or readable standard. In the future, it is critical that policy action is taken to define the format of dates, vehicle characteristics, and customer characteristics. This will fine-tune the database to handle new entries as time moves forward. Process, as a curatorial action, is another activity that can enhance the database. Process would be defined by, "the ensuring of success and efficiency in your organization, by managing the development of appropriate organizational units and roles, providing training, advocating for change, and managing curatorial activities." This is a new database for a small dealership. If time is invested now into developing the processes for training, management, and organization of the database team the database will avoid catastrophic failure. The dealership should take note of the importance of this database and schema and begin advocating for the proper process. Lastly there is a recommendation for analysis. As the dealership grows and changes, and the database becomes sufficiently large, the needs that the schema must meet will change. The development of relevant data models as time goes on, and the need to update or correct data is something to consider.