

# Huffman trucking: database design and development

[Design](#)



Running head: HUFFMAN TRUCKING Huffman Trucking: Database Design and Development Huffman Trucking started out as a single owner, single truck and trailer, operating in the Cleveland Ohio area back in 1936 doing local contract hauls. Today, Huffman Trucking is a National carrier with 1, 400 employees, 800 tractors, 2, 100 trailers, and 260 roll-on/roll-off units, operating from 3 logistical hubs located in Los Angeles, California, St. Louis, Missouri, and Bayonne, New Jersey and its central maintenance facility located in Cleveland Ohio (Apollo Group Inc. , 2005).

With the growth through the years, Huffman Trucking has maintained their competitiveness by being an industry leader in leveraging technology to the maximum to provide customer service and business efficiencies (Apollo Group Inc. , 2005). In the means to maintain this competitiveness, Huffman Trucking hired Smith Systems Consulting to develop a report of entities and attributes that will be needed for a Fleet Truck Maintenance Database. Upon receipt of Smith's report detailing the entities and attributes needed, our IT Manager submitted a Service Request SR-ht-003 to design a Fleet Truck Maintenance Database.

In the following paragraphs LTA will discuss the database architecture briefly and primary keys, which play a vital role in an Entity-Relational Database. The discussions of the different types of mistakes that are made in the design phase that led to a poor database design are also discussed. Mistakes include the lack of careful planning, proper normalization of data, poor naming conventions, lack of sufficient documentation and extensive testing. The ERD for the database will be revealed along with the choice of the

program to manage the database and allow for versatility for various platforms, applications, and features.

Huffman Trucking's fleet truck maintenance records are fairly straightforward, therefore, a basic database design architecture is recommended as a start in the entry of information, and importing of current database records into the new basic database. By starting simple, this database can be upgraded over time, as the company grows and the fleet grows. The important items to consider when designing a new database include: ease of use for the users, the production of query reporting, as well as financial records, parts orders, maintenance records, and purchase orders. "A good model and a proper database design form the foundation of an information system. Building the data layer is often the first critical step towards implementing a new system, and getting it right requires attention to detail and a whole lot of careful planning. A database, like any computer system, is a model of a small piece of the real world. And, like any model, it's a narrow representation that disregards much of the complexity of the real thing" (Malone, 2007). A primary key, which is a record or an attribute, uniquely identifies a table.

Primary keys make mapping relational data simple, in order to uniquely identify each entry in the database. The concept of some sort of unique value is common in database designing — using account numbers to identify part numbers, vendor numbers, and maintenance work orders. These are also known as natural keys, common entities that are used to uniquely identify objects. Generally, if the data that is being modeled has a decent

natural key, or identifier, that information should not be used as a primary key.

Natural keys should not be used as primary keys, as the purpose of the primary key is to uniquely identify a value in a database record. Several primary key characteristics are the primary key must be able to identify each row in a table. The primary key should not describe the characteristics of the entity. A part number ID of " 2566" is usually preferred over " Air Filter. " The value of a primary key should never change. Changing a primary key value means changing the identity of an entity. Changing the identity is not advised. Non-intelligent keys are preferred because they are less likely to change.

For example, the part number 2566 for an Air Filter for one model of truck, and the part number of 2560 would be an Air Filter for another model of truck. To have just a part number of " Air Filter" would be too ambiguous, and could result in lost time trying to locate the correct air filter for a specific model of truck. Those part numbers, in general would most likely never change over time, therefore, are best to use as primary keys in a database of part numbers. Primary keys should have the smallest number of attributes possible.

It is easier to manage unique keys that are numeric. Items to Consider During Design Phase Several things that are easy to overlook during the database design process include design and planning of the database, normalization of data, insufficient naming conventions, documentation, and testing. A brief run-down of these common errors during the design phase of a database is listed below. By listing them now, it can be used as an effective <https://assignbuster.com/huffman-trucking-database-design-and-development/>

guideline to follow when designing the database for Huffman Trucking's Fleet Maintenance. Design and Planning of the Database

Good databases are designed with careful thought, and with the proper care and attention given to the needs of the data that will be part of it. Since a carefully constructed database is at the heart of every business project, insufficient planning and detailing of the needs of the project could cause the whole project to lose its direction and purpose. Additionally, by not taking the time at the beginning, any changes in the database structures that may be needed to be made in the future could cause devastating consequences on the whole project, and greatly increase the likelihood of the project timeline slipping.

If the planning phase is rushed, problems will inevitably arise, and because of the lack of proper planning and design, there is usually no time to go back and fix any issues properly. "That is when the 'hacking' starts, with the veiled promise to return and fix things later, something that happens very rarely indeed" (Davidson, 2007). Normalization of Data Normalization defines a set of standards to break down tables into their basic parts until each table represents only one thing, and its columns fully describe the only thing that the table represents.

Normalizing the Huffman Trucking's data is important to ensure proper performance and ease of future development projects. Insufficient Naming Conventions Naming conventions are most the important line of documentation for any application. What is important to consider is the importance of consistency. Names should be kept simple while at the same time, identifying their purpose to the data being entered. Documentation Not <https://assignbuster.com/huffman-trucking-database-design-and-development/>

only will a well-designed database conform to certainly quality tandards, it will also contain definitions and examples about its tables, so that its purpose is clear to everyone on how the tables, columns and relationships are intended to be used. The goal of proper documentation should be to provide enough information for a support programmer to find any bugs and fix them easily. Testing As many Information Technology professionals know, the first thing to be blamed when a business system starts running slow is because the database can get bogged down with fragmented information, or too much information.

Deep knowledge of the system is the best way to dispel this notion Unfortunately, testing is the usually one of the things to go in a project when time starts to run out. What is important to consider in this whole process is that deep system testing is done to make sure that the design is implemented correctly. The first real test is for any database is when it goes into production, and users attempt to do real work. And if the system does not work fast enough, or contains bugs when it goes live, then more work will have to done on a live system, which could inherently cause the loss of revenue of any company.

By insisting on strict testing as an important aspect of database development, then perhaps the day will come when the database will not be the first thing to be pointed out when the system slows down. In order to accomplish the goal of establishing a functional database that Huffman can use now and in the future to effectively manage their data, it is recommended that Huffman Trucking decide to use MySQL. There are many

great things about MySQL, including the fact that MySQL is very popular among web applications and acts as a database for a multitude of platforms.

Some of these platforms include FreeBSD, BSDi, AIX, HP-UX, Linux, Novell NetWare, OS/2 Warp, Solaris, i5/OS, Windows 95, Solaris, Windows 98, SunOS, Windows ME, Windows 2000, Windows XP, and Windows Vista. MySQL is popular among open source code and bug tracking tools such as Bugzilla as well. MySQL is written in C and C++. Libraries that are used to access MySQL databases can be found in many of today's programming languages by using language specific API's. There is also an Open Database Connectivity (ODBC) that allows additional programming languages to communicate with MySQL, including ColdFusion or ASP.

MySQL features options that are not in many other RDBMSs. One feature that is not included in many RDBMSs is multiple storage engines. This feature allows for a user to select the most effective storage engine for each table in the application. Another great feature that MySQL offers is native storage engines. These are storage engines that are developed by MySQL and are optimized for specific application storage domain. They offer data warehousing, data archiving, high availability clustering, and many more features. MySQL recently developed a new advanced transactional storage engine called Falcon.

Falcon was designed for modern day corporations and web applications which makes it perfect for Huffman Trucking. One feature not to be overlooked is the availability of Partner-developed storage engines. Search engines that are partner developed are developed buy outside companies, but they are then highly tested by MySQL in order to ensure workability and <https://assignbuster.com/huffman-trucking-database-design-and-development/>

compatibility with MySQL. MySQL also has open source programmers that are independent and develop storage engines. These are used as well, but only after they pass MySQL rigorous inspection and testing.

Customers are even developing and designing community storage systems. Commit grouping is a MySQL feature that allows for the gathering of multiple transactions. This is done from a multitude of connections in order to increase the number of commits per second. Conclusion In conclusion the Fleet Truck Maintenance Database will be easy to use, provide effective tracking of finances, maintenance, and queries. The primary key(s) used in the database will have the following characteristics: be a single attribute, uniquely identify an entity, be non-intelligent, not change over time and be numeric.

This will ensure the ease of normalizing the database during the design phase to prevent update anomalies when database is implemented. LTA discussed several mistakes that occur during the design phase in order to avoid the same mistakes. These mistakes include poor design/planning, ignoring normalization, poor naming standards, lack of documentation and testing. The DBMS of choice for Huffman Trucking is MySQL. MySQL will effectively manage our data while allowing many different platforms to interact with the database.

MySQL is written in C and C++. However, MySQL offers much versatility in programming languages by using language specific API's or ODBC to allow additional programming languages such as ASP or Coldfusion. MySQL has many options that other RDBMSs do not possess such as multiple storage engines, open source programmers, commit grouping and more. Bottom line <https://assignbuster.com/huffman-trucking-database-design-and-development/>



is that MySQL offers versatility for our database to allow for continued growth, updates and changes in our company's needs. References Apollo Group Inc. (2005). Huffman Trucking.

Retrieved October 1, 2008, from Huffman Trucking Intranet: [https://ecampus.phoenix.edu/secure/aapd/CIST/VOP/Business/Huffman/HuffmanHome002.](https://ecampus.phoenix.edu/secure/aapd/CIST/VOP/Business/Huffman/HuffmanHome002.htm)

htm Davidson L. , (2007). Ten Common Database Design Mistakes. Simple-Talk. com. Retrieved September 29, 2008, from <http://www.simple-talk.com/sql/database-administration/ten-common-database-design-mistakes/>

Malone M, (2007). I'm Mike. Database Design: Choosing A Primary Key. Retrieved October 1, 2008, from [http://immike.](http://immike.net/blog/2007/08/14/database-design-choosing-a-primary-key/)

[net/blog/2007/08/14/database-design-choosing-a-primary-key/](http://immike.net/blog/2007/08/14/database-design-choosing-a-primary-key/)