

Module_7: SQL Server Parallel with Data Warehouse

7.1 Introduction and Background

Microsoft SQL Server Parallel Data Warehouse (PDW) DBMS system is specifically designed for data warehousing workloads. PDW is a highly scalable and massively parallel processing (MPP) solution that follows a "share nothing" architecture. It is packaged as a database appliance built on industry-standard hardware. In this overview, we will delve into the key software and hardware components of the PDW system and provide an example that illustrates its query processing approach.

A data warehouse is an element of computing systems considered a core part of many businesses. They are mainly used for data analysis. A data warehouse combines information from multiple different places electronically into one single all-inclusive database. For example, companies may combine sales from the cash registers in a store, online sales, and company-to-company orders all together as one database. Data warehouses are made for this purpose.

In recent years, the OLTP (Online Transaction Processing) and DW (Data Warehousing) market segments have experienced significant growth, enabling the development of specialized database systems tailored to meet the unique requirements of each workload [3]. During the early 2000s, numerous start-ups emerged within the database industry, offering scale-out, Massively Parallel Processing (MPP) database engines specifically designed to cater to the demands of large-scale DW workloads. Notable examples include ParAccel, Netezza, Greenplum, Exasol, DatAllegro, and Vertica.

These specialized engines introduced a major disruption in the DW market segment by achieving significantly higher performance compared to general-purpose MPP database engines. They accomplished this by capitalizing on advancements in hardware, such as larger and more affordable RAM, multi-core processors, and enhanced network and memory bandwidths. Additionally, they leveraged innovative storage formats like column stores and data compression, as well as query processing techniques finely tuned for DW workloads.

Another noteworthy development was the rise of the database appliance, which revolutionized the way database systems were delivered. These appliances provided complete hardware and software solutions, packaged together for seamless deployment and operation. This approach offered streamlined implementation and reduced complexity for users.

Overall, the convergence of technological advancements and the specific demands of DW workloads led to the emergence of specialized database engines that outperformed their general-purpose counterparts. Furthermore, the introduction of database appliances brought about a new level of simplicity and efficiency in deploying database systems.

A data warehouse is a collection of data that is non-updateable, time-variant, integrated, and subject-oriented that is used in the support of decision-making management processes and business intelligence. The meaning of each of these terms is important for defining what a data warehouse is. The term non-updateable means the information in the data warehouse can be refreshed and loaded from operational systems, but cannot be updated by end users. The term time-variant means the information in the data warehouse contains a dimension of time so that they can be used to study changes and trends. The term integrated refers to the data in the data warehouse are characterized using consistent formats, encoding structures, related characteristics, and naming conventions gathered from many internal or external systems of record. The term subject-oriented means the data warehouse is organized around key subjects,

or higher-level entities, of the enterprise. These major subjects may include students, patients, products, or time. The data warehouse is not a mere consolidation of all of the operational databases of an organization. Because of a data warehouse's focus on business intelligence, time-variant data, and external data, a data warehouse is distinct from a simple database. Data warehousing is a process where organizations build and maintain data assets. Making successful a data warehouse requires proven practices of data warehousing, strong organizational commitment, sound project management, and making the right technology decisions.

Putting all this information into one place makes it a lot easier for people to look at and evaluate that information. It also makes the jobs of data miners and data analysts much easier, since they can find everything all in one place. It can even help companies realize higher profits and make sales a lot easier.

Data warehouses have many other advantages that come from combining data from several locations into one central database. Data warehouses prevent database isolation sometimes caused by running very long queries. Even if source transaction systems fail, the data warehouse will still maintain data history. They also much more consistently provide data to the end user, helping to prevent mistakes. A data warehouse can also reorganize all of the information found to make it much easier to read for analysts and to help them more clearly understand the content. Reorganizing the information properly can also increase the performance for queries very noticeably. It can even make the queries much easier to construct in many instances.

Although there are many positives to data warehouses, they have their cons. They can often make certain employers have to do even more work. This would include gathering data from other workers and customers to help maintain the warehouse. Another common issue with data warehousing is the cost-to-benefit ratio. This is a huge project for businesses, and it could take up thousands of hours to establish. Sometimes it just isn't used enough to make up for the enormous implementation costs. This is before even mentioning the huge amount of money it takes to maintain the data warehouse and keep it up and going. You also have to update it and do a lot of maintenance for it as your business grows which can be a pain at certain times. The price for a simple one-terabyte data warehouse with around one hundred thousand queries per month can cost as much as four hundred and sixty-eight thousand dollars a year. For big companies, this might not be a huge deal, but this is definitely not affordable for smaller businesses. Businesses must carefully consider whether they truly need a data warehouse.

The privacy and security of your data for a data warehouse is also only as good as your cloud services, so for smaller businesses, data may not be safe. If a corporation's warehouse is accessed and leaks customer information, it could be terrible for the company. It would give out possibly customers' credit card information and damage the company's reputation. Although there are many things that can go wrong with a data warehouse, if you know how to use one correctly the positives outweigh the negatives.

There are many ways to create a data warehouse and several different ways to upload all the information quickly. First, consider all of the data from each individual source: for example, all data from the cash registers, online sales, and company sales. Next, remove all of the redundant data, so there is no repetition to corrupt the information provided. Lastly, reorganize all of the data obtained into a consistent format so that it can all be queried. A query is a request for information from a specific table inside of a database.

There are three main types of data warehouses. First is the Enterprise Data Warehouse, which is a central database. It has decision support services. Its approach is unified for keeping the data in the warehouse organized, which helps present the data easier. Users can also sort the data more easily according to category.

Next, we have the Operational Data Store. The Operational Data Store (ODS) is nothing but data being stored for when there is no organization needed. ODS is refreshed in real time. It is highly recommended to use ODS over the other two methods for more simple things like storing employees' hours or pay to keep things simpler.

The last form of data warehouse is a Data Mart. Data marts are basically just a subset of the actual data from the data warehouse. It was made precisely for sales and financing businesses. If the data mart is self-regulating, then data can be composed straight from the founding source. Basically, any data warehouse that is mostly used for money-related purposes should be created as a data mart.

There are multiple different components of a data warehouse. First, the Load Manager is also known as the "front component" of the warehouse. It is called the front component because it works with the main extraction of data to load it all into the data warehouse. Next we have the Warehouse Manager, which is used to manage the data inside of the warehouse. It helps to investigate the data inside the warehouse to make sure it stays consistent throughout. The Query Manager is another key component to the warehouse, its main job being to manage the queries for the warehouse. Since it manages the queries of the warehouse it is also known by some as the "backend component." Lastly, we have the end-user access tools, which are assembled into five different categories. The first category is responsible for reporting the warehouse data to management. The second category is used to store the tools for the queries. The third category is for tools that help application development. The fourth category is for EIS tools. Lastly, the fifth category is for OLAP and data mining tools.

Next, we discuss the architecture of a data warehouse. A simple data warehouse design is where all the different forms of data are stored in the central repository of the data warehouse. The repository receives information from each source and then is used for data analysis and mining. Next we have simple architecture design with a staging area included. All of the non-required data must be prepared before going to the warehouse, because only the necessary data is wanted. The staging part of the warehouse prepares data before it actually goes into the data warehouse. The next form is called hub-and-spoke warehousing. It works similarly to a staging area, but instead adds a data mart after the central repository to further customize the data warehouse, which may be needed in certain types of business applications. After the data is sorted, cleaned, and ready to use, it is moved into the responding data mart. Lastly, sandboxes are very private and protected data stores, which make them very efficient and allow new ways of analyzing the data sets without having to follow the basic rules of the data warehouse.

Many tools and utilities are required to effectively use a data warehouse. These can be very useful when navigating the warehouse. Data Extraction is used to get data from several different outside sources and present them. Data Cleaning tools can also be very useful for finding the errors inside of the data set, even correcting them for the user. This can also be used to get rid of duplicate data. Data Transformation tools are used for converting the format type of the data sets. For example, if the date is in a legacy format, it can easily be transformed to warehouse standards. Data Loading is used mostly for sorting and checking reliability of data. The refresh tool is just for simply updating the sources into the warehouse format.

There are three main ways to test a data warehouse. The first method is called Unit Testing and consists of testing each individual part of the warehouse in turn. This test is usually done by the developer. Next is Integration Testing, in which different parts of the application are formed into one, then tested compared to the number of inputs put in. This method is mainly used to make sure all of the parts work together after combining them. Lastly, System Testing tests all the data in the entire data warehouse together. This is the final check to make sure that everything is working together as a final product. This kind of testing is the testing done by the whole team.

Most organizations have a plethora of data but cannot access it effectively and efficiently. There are multiple reasons why there is a knowledge gap between so many organizations. The first reason is because organizations have a fragmented way in which they created their information systems and databases. Because these organizations have time constraints and resource constraints, they have been developing information systems one at a time. This creates inconsistent and uncoordinated databases. Databases are usually based on different purchased applications, software platforms, and hardware that may have been brought in from a variety of organizational acquisitions, mergers, and reorganizations. From these circumstances, it can be very difficult or nearly impossible for managers to find and use accurate information that has to be synthesized across different systems.

Most systems are created to help operational processing, in which there is little to no thought is given to information and analytical tools needed to make decisions. Operation processing, also known as transactional processing, takes, stores, and changes data to help daily operations of an organization. It usually focuses on optimizing access to a small set of data that relates to a specific transaction—for example, a customer, an order, and associated product data. Information processing is an analysis of data or other forms of data to help decision making. Large swaths of data are needed to derive information, such as sales of total products, across several years, from all sales regions. Systems that are mostly developed internally or bought from outside vendors were designed to help operational processing, and very little thought is given to information processing.

The next trend is that multiple systems are not synchronized. It may be difficult, maybe impossible, to make separate databases consistent. Even if the metadata are controlled and created by only one data administrator, the values of data for the same attributes may not agree. That is because of a difference in update cycles and of the separate places where the same data is captured for all systems. Therefore, to get a single view of the organization, the data from separate systems has to be periodically separated and synchronized into one additional database.

Many organizations use some form of a balanced scorecard-metrics, which show organization results in customer satisfaction, product quality, human, financial, and other terms simultaneously. To make sure that this multidimensional view of the organization shows consistent results, a warehouse of data is necessary. When questions rise on the balanced scorecard, analytical software that the data warehouse may be used to “slice and dice,” “drill down,” visualize, and in many other ways mine business intelligence. The next trend is customer relationship management. This is when organizations in all sections are realizing there is value in having a total picture of interactions with customers across all of the touch points. For example, for a bank, the touch points are ATMs, online banking, electronic fund transfers, tellers, loans, and investment portfolio management, all supported by consolidated operational systems. Therefore, without having a data warehouse, a teller does not know to try to cross-sell a customer one of the bank’s mutual funds if the large, atypical automatic deposit transaction shows up on the teller’s screen. To have a total picture of the activity by a given customer requires a consolidation of data from many different operational systems.

Another trend is supplier relationship management. The process of managing the supply chain has become critical in raising product quality and reducing costs for many organizations. Organizations desire to make strategic supplier partnerships that are based on a total picture of activities and suppliers, from billing, to the meeting of delivery dates, to support, to pricing, to quality control. Data about these different activities may be locked inside separate operational systems: for example, accounts payable, shipping and receiving, maintenance, and product scheduling. ERP systems also have improved this situation by bringing much of this data into a

single database. However, many ERP systems tend to be designed to optimize operational, instead of informational or analytical, processing.

There is a need to separate informational and operational systems. An operational system must be a system which is used to run a business in actual time, based on current data. Some examples of operational systems are sales order, patient registration systems, and reservation systems. Operational systems have to process large volumes of relatively simple write/read transactions and provide fast responses. Operational systems can also be called systems of record. Informational systems were designed to support decision making based on historical prediction and point-in-time data. Also, they were designed for data-mining applications or complex queries. Some examples of informational systems would be customer segmentation, systems for sales trend analysis, and human resources planning.

The main differences between informational and operational systems are defined by six key characteristics. These characteristics are primary purpose, type of data, primary users, scope of usage, design goal, and volume. For the first characteristic, the primary purpose of an operational system is to run the business on current basis, but for informational systems, the primary purpose is to support managerial decision making. The second characteristic is type of data. The type of data that operational systems use is a current representation of the state of the business. The type of data that informational systems use is historical point-in-time, snapshots, and predictions. The third key characteristic is the primary user. The primary users of operational systems are administrators, clerks, and salespersons. The primary users of informational systems are managers, customers, and business analysts. The fourth key characteristic of the differences is the scope of usage. The scope of usage for operational systems is planned, narrow, and simple updates and queries. The scope of usage for informational systems is ad hoc, broad, and complex queries and analysis. The fifth key characteristic the design goal. The design goal of operational systems is that performance is throughput and the availability. The design goal of informational systems is the ease of flexible access and use. The sixth key characteristic of the difference is volume. The volume of operational systems consists of multiple constant updates and queries on one or a few table rows. The volume of informational systems consists of a periodic batch of updates and queries requiring many or all rows.

This shows that the two types of processing have vastly different characteristics in almost every category of comparison. What is particularly noticeable is that they have very different communities of users, because operational systems are used by salespersons, administrators, clerks, and others who need to process business transactions, while informational systems are used by executives, business analysts, managers, and increasingly by customers who are in search of status information or those who are decision makers.

The need to separate informational systems and operational systems is based on three key factors. The first factor is that a data warehouse tries to centralize data that is scattered throughout different operational systems and makes them easily available for decision support applications. The second factor is that a properly designed data warehouse can add value to data by increasing their quality and consistency. The third factor is that a separated data warehouse negates much of the contention for resources that would result when informational applications are competing with operational processing.

The success of data warehousing is not guaranteed. Data warehousing projects fail about forty percent of the time. This is because warehousing data is complex and requires cooperation and synchronization across an organization. For example, when errors become discovered in data that is loaded into the warehouse, the correct place to fix the errors will be in the source systems so bad data is not loaded again later. But the errors might be acceptable in the source

system or even not considered errors by the unit of business for the source system. Many professional organizations sponsor yearly award programs to show the best data warehousing practices. The Data Warehousing Institute Awards is one of the most prestigious yearly award programs. There are many impressive winners from this award program. One of these winners is Continental Airlines, which won the award for the best enterprise data warehouse. The continental data warehouse had an actual-time architecture and had automated data transformations. What this did was simplify consolidated data from various source systems. A cross-unit-business system steering committee developed standard data definitions, also known as metadata. The uses of and changes to the warehouse have to be justified by profitability and revenue projections.

Ease of access is a necessity for companies that use collected data for their business. Many businesses function by using user data to create value for themselves or for other companies. Companies such as Google use user search data to tailor ads to the user's interest. The ease of use provided by data warehouse interfaces allows for companies to perform their tasks more efficiently than if they had to access the data by hand. They do this by using an algorithm that queries their database through the interface and then analyzes the user search history to show ads to the user that might provoke a response. Some companies just collect the data and store it, selling it to advertising companies who use the data to send relevant information to users who fit certain criteria. These are just a couple of examples of what is possible with data warehouse queries and analysis.

The architecture of a data warehouse is broken down into three levels; Data Sources, Warehouse, and users. The user-level of the data warehouse is where all of the analysis, reporting, and data mining happens. This is the level where most businesses operate, searching for and using the data for their intended purposes. The warehouse level is where the metadata, summary data, and raw data are stored. This allows for usage from the user level and for data deposits from the data sources level. This is the level that receives the most traffic because it handles the movement of data and the resorting after the reception and removal of data. The data sources level is where the data is generated, either through digital or real-world collection, and then stored in the warehouse level.

Each piece of data is categorized based on various criteria, depending on the type of data. For example, if the data came from a financial transaction, it would be sorted based on date, amount, sender, receiver, and status. The data is then put with data that has the same criteria, typically sorted by date. Once the data is stored it can be easily retrieved and aggregated into a list of data that all matches a list of requirements for analysis. This allows for the data to be easily accessed and sorted.

These warehouses hold information that has become important to the function of most companies, schools, banks, militaries, and governments. Data warehouses store information about equipment purchases, student class lists, transaction history, payment history, payroll, inventory listings, and employee information. Without efficient, organized ways to store all this information, companies and organizations wouldn't be able to function. Stores wouldn't be able to keep track of their stock so that they can remain stocked. Companies wouldn't be able to manage and maintain payroll for their employees, keep track of what equipment is at what location, or what businesses placed orders for how much. Data storage, management, and organization have grown in importance to the point where most of modern society is dependent on the storage and reproduction of data.

Concise Summary

The SQL Server Parallel Data Warehouse (PDW) is an appliance that employs a shared-nothing, Massively Parallel Processing (MPP) approach to deliver exceptional performance and scalability for data management in Data Warehouse (DW) workloads. This appliance comprises a distributed cluster of computer servers (known as nodes), each serving a distinct role in providing the solution. The system's capacity can scale from tens of terabytes to over 500 terabytes of raw tabular data.

The SQL Server PDW appliance integrates software, hardware, high-speed networking, and storage, all pre-configured and balanced to efficiently process statements typically encountered in DW workloads. The appliance is designed to be plug-and-play, eliminating the need for specialized tuning or configuration, ensuring a low total cost of ownership for customers.

To achieve high performance and scalability, SQL Server PDW employs data partitioning techniques across and within computer nodes. This enables optimal utilization of every component of the system during the execution of SQL statements. By effectively distributing user data and meticulously designing each system component, SQL Server PDW delivers impressive performance and scalability. Data warehousing is an extremely important component of modern business, forming the technical backbone of any given company. Data warehouses are used to store, manage, and access the data that the companies use. Data warehouses process many different types of data, varying from purchase data to internet ad data. Data warehouses operate with the same general structure. The data sources put the data into the warehouse system, and the users access the warehouse through the user level.

Data warehouses are created in a very particular way so that users and data sources can interact with it at the same time. The physical structure of data warehouses consists of a series of computers connected together in such a way so that they share storage space, allowing them to distribute the data for easier access and storage. This system of distributed storage allows for the users to access the data at the same time as the data sources, in addition to having multiple users able to access the data simultaneously. These data warehouses also make it easier to combine multiple data sets into one place, sorting the data sets into a format that is easily searchable and accessible.

While data warehouses are useful systems, allowing users to collate data and search it to be processed, implementing a data warehouse is a very big project. Creating a data warehouse requires a lot of time, money, and resources in order to do properly. A company will have to pay for the computer hardware and networking as well as having to pay employees or contractors to set up the hardware. Once the hardware is set up, the company has to pay their employees to set up the system before it can be used. This results in a time span where they are paying for a system that is not providing any return until it is complete. Depending on the size of the warehouse, this can result in loss of profit versus the cost of set up.

Data warehouses are an exercise in cost-benefit analysis in most situations, usually ending up being more beneficial in the long run. The utility of a system that automatically sorts the data stored within based on input criteria is quite valuable. Once the data warehouse is set up, it ends up resolving a lot of problems that come from not having the sorted collection of data. It allows for employees across the company, and across the country, to access the same data and collaborate in the data analysis. The collaboration ends up saving the company time and effort, eventually resulting in more work done for less effort. All in all, data warehouses expand the availability of data and make data manipulation significantly easier.

Extended Resources

1. Data Warehouse Tutorial For Beginners | Data Warehouse Concepts | Data Warehousing | Edureka. Data Warehousing & BI Training: <https://www.edureka.co/data-warehousi...>
This Data Warehouse Tutorial For Beginners will give you an introduction to data warehousing and business intelligence. You will be able to understand basic data warehouse concepts with examples. The following topics have been covered in this tutorial:

1. What Is The Need For BI?
2. What Is Data Warehousing?
3. Key Terminologies Related To DWH Architecture:
 - a. OLTP Vs OLAP
 - b. ETL
 - c. Data Mart
 - d. Metadata
4. DWH Architecture
5. Demo: Creating A DWH

<https://www.youtube.com/watch?v=J326LIUrZM8>

2. Data Warehouse Concepts | Data Warehouse Tutorial | Data Warehouse Architecture | Edureka

Data Warehousing & BI Training: <https://www.edureka.co/data-warehousi...>

This tutorial on data warehouse concepts will tell you everything you need to know in performing data warehousing and business intelligence. The various data warehouse concepts explained in this video are:

1. What Is Data Warehousing?
2. Data Warehousing Concepts:
3. OLAP (On-Line Analytical Processing)
4. Types Of OLAP Cubes
5. Dimensions, Facts & Measures
6. Data Warehouse Schema

<https://www.youtube.com/watch?v=CHYPF7jxlik>

3. What is a Data Warehouse - Explained with real life example | data warehouse vs database (2020)

About this video - In this video, we will understand what a data warehouse is using a very simple really life example. A data warehouse is nothing more than a storeroom of your house. Data warehouse stored all your enterprise data into a centralized location and hence called EDW (Enterprise Data Warehouse). So, in this video we will cover

- 1) What is a Warehouse?
- 2) Real-life example to explain datawarehouse ?
- 3) Various data warehousing stages
- 4) Characteristics of a datawarehouse

https://www.youtube.com/watch?v=jmwGNhUXn_o

4. Microsoft Azure Analytics - Azure Synapse Analytics is a limitless analytics service that brings together data integration, enterprise data warehousing, and big data analytics. It gives you the freedom to query data on your terms, using either serverless or dedicated resources—at scale. Azure Synapse brings these worlds together with a unified experience to ingest, explore, prepare, manage, and serve data for immediate BI and machine learning needs.

<https://azure.microsoft.com/en-us/services/synapse-analytics/>

5. What is Data Warehouse-as-a-Service?

Data Warehouse-as-a-Service (DWaaS) is a modern solution to address the data management challenges of today's companies. Data is critical to how modern companies operate, from providing actionable analytics and insights to fueling digitally transformed business processes.

Companies generate tremendous amounts of data each day, but to translate this resource into value, a company needs a place to aggregate, store, organize and analyze the data – that is a data warehouse. As one might imagine, data warehouses can be quite large and costly to build and maintain. Data Warehouse-as-a-Service addresses this challenge by providing the full-featured capabilities companies need, without much of the administrative overhead.

-
6. A data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management's decision-making process.— W. H. Inmon

<https://www2.cs.sfu.ca/CourseCentral/741/jpei/slides/Data%20warehousing%202.pdf>

7. Data Warehouse: From Architecture to Implementation - Data warehousing is one of the hottest topics in the computing industry today. For business executives, it promises significant competitive advantage for their companies, while information systems managers see it as the way to overcome the traditional roadblocks to providing business information for managers and other end users.

With the publication of this book comes the most comprehensive, practical guide to designing, building, and implementing a data warehouse on the market today. Barry Devlin - one of the world's leading experts on data warehousing - is also one of the first practitioners in this area. In this book, he distills the insights and experiences gained over 10 years of designing and building data warehouses.

<https://dl.acm.org/doi/book/10.5555/548222>

8. Building the data warehouse – Author: Stephen R. Gardner

Publication: [Communications of the ACM](#) September

1998 <https://doi.org/10.1145/285070.285080>

<https://dl.acm.org/doi/abs/10.1145/285070.285080>

9. Microsoft SQL Server Parallel Data Warehouse: Architecture Overview, _Blakeley J., A., Paul A. Dyke, César Galindo- Legaria,
file:///C:/Users/snorth/Documents/SarahsFiles/~KSU/ALG/1-ALG-Round%2022-2022-2023-DB/ALG_CS%203410-2023-/ALG%20Modules-CS_3410/DB%20Textbook/Mod_7/Textbook.pdf

10. SQL Scorecard for Improved Stability and Performance of Data Warehouses, Nayem Rahman, 2016, <https://www.igi-global.com/article/sql-scorecard-for-improved-stability-and-performance-of-data-warehouses/157277>

11. A Study on Big Data Integration with Data Warehouse, T.K.Das and Arati Mohapatro, School of Information Technology & Engineering, VIT University, Veldore, India, <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=db5ee8a8f6e0b3316d691bbad41533683c6f7351>

References

- Hoffer, J. A., Ramesh, V., & Topi, H. (2011). *Modern database management*. Upper Saddle River, NJ: Pearson.
- Burnside, K. (2017, November 21). The Disadvantages of a Data Warehouse. Retrieved from <https://smallbusiness.chron.com/disadvantages-data-warehouse-73584.html>
- Blakeley J., A., Paul A. Dyke, César Galindo-Legaria, Enabling Real-Time Business Intelligence, 2012, Volume 126, ISBN : 978-3-642-33499-3
- Data Warehousing - Concepts. (n.d.). Retrieved from https://www.tutorialspoint.com/dwh/dwh_data_warehousing.htm
- Data Warehousing Concepts. (n.d.). Retrieved from https://docs.oracle.com/cd/B10500_01/server.920/a96520/concept.htm
- Data Warehousing. (n.d.). Retrieved from <https://investinganswers.com/dictionary/d/data-warehousing>
- Data warehouse. (2020, February 7). Retrieved from https://en.wikipedia.org/wiki/Data_warehouse#Benefits
- 2017, September 27. The True Cost of Building a Data Warehouse. Retrieved from <https://www.cooladata.com/cost-of-building-a-data-warehouse/>
- Data Warehouse Architecture, Concepts and Components. (n.d.). Retrieved from <https://www.guru99.com/data-warehouse-architecture.html>
- Golfarelli, M., & Rizzi, S. (1970, January 1). Figure 1 from WAND: A CASE Tool for Data Warehouse Design: Semantic Scholar. Retrieved from <https://www.semanticscholar.org/paper/WAND:-A-CASE-Tool-for-Data-Warehouse-Design-Golfarelli-Rizzi/4eef34cdd2505cb5bcb553496ab654fa0cd1c475/figure/0>
- Pearlman, S. (n.d.). What is a Data Warehouse? - Talend. Retrieved from <https://www.talend.com/resources/what-is-data-warehouse/>
- Walls, D. (2019, January 15). 7 Steps to Data Warehousing. Retrieved from <https://www.itprotoday.com/sql-server/7-steps-data-warehousing>
- What is a Data Warehouse? (n.d.). Retrieved from <https://www.oracle.com/database/what-is-a-data-warehouse/>
- What Is Data Warehousing? Types, Definition & Example. (n.d.). Retrieved from <https://www.guru99.com/data-warehousing.html#4>