DataWash: An advanced Snowflake Data Quality tool powered by Snowpark

Gerard Gallardo Mírez

Abstract — The increasing need for data accuracy and completeness in today's organizations has highlighted the importance of data quality management. To address this need, DataWash has emerged as an advanced data quality tool powered by Snowpark that provides organizations with a comprehensive solution for improving the quality of their data in Snowflake. This tool provides scheduled batch execution and ad hoc / on-demand analysis capabilities, generating a Power BI report for easy visualization of data quality metrics. The suite of modules provided by DataWash can handle a wide range of data quality issues, such as data duplication, inconsistencies, and compliance with data standards. In essence, this bachelor's thesis aims to develop DataWash as an advanced data quality tool in order to help organizations improve the accuracy and reliability of their data by exploring its capabilities and cost-effectiveness, evaluating its performance using real-world datasets, and benchmarking it against leading data quality tools on the market.

Index Terms — Data Quality, Snowflake, Snowpark, Data Cleaning, Data Profiling, Data Validation, Data Enrichment, Streamlit, Power BI, Data Integration, Data Governance, Data Visualization.

1 Introduction

Data quality is a crucial aspect of any data-driven organization, and ensuring the accuracy, completeness, and consistency of data is a challenging task. The proliferation of data sources and the increasing volume, velocity, and variety of data have made data quality management more complex and difficult.

The risks associated with poor data quality are multifaceted. Inaccurate or incomplete data can result in flawed business intelligence, leading to misguided strategies and ineffective operational decisions. Moreover, organizations heavily depend on data integration and data exchange with various systems and stakeholders.

Additionally, poor data quality can hinder regulatory compliance efforts. Many industries, such as healthcare, finance, and telecommunications, are subject to strict regulatory requirements regarding data accuracy, privacy, and security. Non-compliance with these regulations can lead to legal consequences, financial penalties, and damage to the organization's reputation.

That is where DataWash comes in, an advanced data quality tool that provides a comprehensive set of modules to detect and correct data issues in an automated and efficient way, helping organizations improve decision-making, increase operational efficiency, and reduce risks associated with poor data quality.

In this bachelor's thesis, we will explore the capabilities and features of this data quality tool that I have been developing over the last few months, its underlying technology Snowpark, and the benefits it offers to organizations seeking to improve their data quality management.

We will also discuss the state-of-the-art approaches to data quality management, compare them with DataWash, and conduct an evaluation of the tool's effectiveness and cost-effectiveness. Ultimately, we will provide recommendations for organizations seeking to adopt DataWash as part of their data quality management strategy.

2 State-of-the-art

The field of data quality has gained increasing importance in recent years as data has become a critical asset for many businesses. The use of big data and analytics has led to a growing demand for high-quality data to make accurate business decisions. As a result, data quality tools have become an essential component of any data management strategy [1].

There are several data quality tools available on the market that provide various functionalities, including data profiling, data cleansing, data enrichment, and data monitoring. Some of the popular data quality tools on the market include Soda [2], Great Expectations [3], dbt [4], Collibra [5], Ataccama [6] and Matillion [7].

However, these tools are often expensive and require significant expertise to implement and maintain. Additionally, they may not be optimized for cloud-based data warehousing solutions such as Snowflake.

To verify the above, a market study has been carried out between the set of data quality rules and checks offered by DataWash and those of leading data quality tool providers. More details on this market study can be found in the dossier, which contains the full analysis. The highlights of the comparison can be seen in the linked figure below (Image 1).

As can be seen in the comparison table, DataWash is different from the other major companies offering data quality tools in several ways. The set of modules offered by DataWash contains the most advanced modules on the market, as well as new modules that perform checks based on machine learning algorithms to provide the user with a global view of the quality of their data.

All this without forgetting those essential modules for a data quality tool such as the number of duplicate rows, null values, etc. This makes DataWash one of the most complete tools on the market today, offering all types of modules (from the most advanced to the essential ones) and providing new proposals for modules adapted to the current needs of the business world.

Apart from the above, DataWash differs from the major companies offering data quality tools in several other ways. Firstly, the cost of running DataWash on Snowflake is significantly lower compared to other data quality solutions, making it more accessible and affordable for small and medium-sized businesses.

In addition to being more economically advantageous, DataWash is also highly customizable, meaning that it can be tailored to meet the specific needs and requirements of each individual business. This is a major advantage, as other data quality tools often come with a set of predefined features and functionalities that may not be relevant or useful for every business. With DataWash, businesses can choose the modules that are relevant to their specific needs and configure them as required, thereby ensuring maximum efficiency and effectiveness.

Furthermore, DataWash is designed to support advanced use cases that are not commonly found in other data quality tools. This means that businesses with complex data requirements can benefit from using DataWash, as it is equipped to handle more challenging and demanding scenarios. For instance, DataWash can be used to validate large amounts of data, detect and correct data errors, and maintain the quality and integrity of data over time. These advanced use cases make DataWash a valuable solution for businesses looking to improve the quality and accuracy of their data and make better data-driven decisions.

It is therefore established that DataWash is a cost-effective, customizable, and advanced data quality solution that offers businesses a range of benefits and features not commonly found in other data quality tools. By choosing DataWash, businesses can improve the quality and accuracy of their data, making better data-driven decisions and achieving their goals and objectives more efficiently.

To put all this into practice, in the following sections we will work with a real use case in order to deeply analyze the full capabilities of DataWash.

3 USE CASE

A manufacturer that specializes in designing automobile components has chosen to store their company's data in Snowflake due to its many benefits, including its ability to handle large amounts of data, its ease of use, and its cost-effectiveness. The company is facing challenges with the quality of their data, which is causing issues in their business operations. Specifically, the company has identified several problems with their data, including:

- Inaccurate data from an API: The API they use to obtain the distances between warehouses for their MRO (Maintenance, Repair, and Operations) Inventory optimization is now providing the distances in miles instead of kilometers. This inaccurate data can cause problems with inventory optimization and potentially lead to delays in production or increased costs.
- Broken sensors: One of the sensors in the machine used for predictive maintenance has been broken and is now reporting 0 instead of the usual data. This can lead to inaccurate predictions and potentially lead to machinery failure or downtime.
- Data duplication: Due to the way their CRM (Customer Relationship Management) works, some new users are being stored twice with different email addresses. This can cause confusion in communication with customers and lead to potential business losses.
- Incorrect data entered manually in the HR source system: Some of the problems identified include missing values in crucial fields such as employee IDs and contact information, inconsistent formatting of names and addresses, outdated or incorrect employment dates, and incomplete or erroneous salary records. These issues undermine the reliability and integrity of data, hindering the effective management and use of human resources information.
- Data drift: A new integration of predictive maintenance is starting to misbehave due to data drift occurring in the sensor data. Such occurrences can result in unreliable forecasts and potentially contribute to equipment malfunctions or operational disruptions.

These issues are not isolated cases and are common in the organization. Sometimes, they go undetected for a while, making the issues much harder to fix once they are found. Therefore, it is essential for the company to have a solution that can continuously monitor their data and detect any issues promptly. This solution should be able to analyze the data in real-time and alert the relevant personnel if any anomalies are detected. By having such a solution in place, the manufacturer can ensure that they are able to resolve any issues quickly and prevent any potential harm to their equipment, processes, or personnel.

To address these challenges, the organization would need to implement a data quality management system that can monitor data quality metrics, detect anomalies and data drift, provide alerts and notifications when issues are detected, and define the actual data quality rules that need to be implemented and controlled.

They would also need to establish data governance practices to ensure the accuracy, completeness, and consistency of their data, as well as roles and responsibilities that need to be in place, so data quality is improved in a sustainable and efficient way. By doing so, the company can improve the quality of their data, reduce the risk of errors and inefficiencies, and ultimately enhance their business operations.

4 OBJECTIVES

As can be seen in the above use case, the company has identified several problems with their data that are impacting their business operations, including inaccurate data from an API, broken sensors, data duplication, and data drift. These issues can lead to delays in production, increased costs, confusion in communication with customers, and potential machinery failure or downtime. The following objectives are designed to address the data quality issues faced by the manufacturer and improve the company's data management.

4.1 Resolving inaccurate data in the API response

The aim is that by using DataWash, the manufacturer can set up data quality rules to check the format, units, and range of the API response data. For instance, they could specify that the distances should be provided in kilometers and flag any API responses that do not meet this requirement. DataWash can also monitor changes in the API response format and notify the relevant team members if there is a change that impacts their processes.

4.2 Troubleshooting sensor data

The purpose is to detect when a sensor is reporting zero values and raise alerts to the relevant team. This can be achieved by setting up data quality checks to monitor the sensor data for anomalies and deviations from the expected values. If the sensor data falls outside of the expected range, DataWash can raise an alert to notify the relevant team members.

4.3 Identification and removal of duplicate user entries

The goal is to allow the manufacturer to identify and remove duplicate user entries in the CRM system. This can be accomplished by setting up data quality checks to detect duplicate entries based on specific criteria, such as email addresses or other unique identifiers. Once duplicate entries are identified, DataWash can merge the duplicate records or delete the duplicates altogether.

4.4 Addressing manual data entry issues in the HR source system

The objective is to enhance the quality and reliability of the human resources data stored in the table by leveraging the advanced capabilities of DataWash. Through the implementation of comprehensive data cleansing and validation processes, we aim to identify and rectify the data quality issues present in the human resources table. This includes addressing problems such as missing or inconsistent data, incorrect formatting, and outdated or inaccurate information.

4.5 Minimizing data drift in predictive maintenance

The target is to reveal when there is data drift in the predictive maintenance system. This can be carried out by establishing data quality validation procedures to diligently scrutinize the data for irregularities and deviations from anticipated patterns. If DataWash detects data drift, it can raise an alert to notify the relevant team members so they can investigate and take appropriate action.

In essence, the main objective of this bachelor's thesis is to develop and implement an advanced data quality tool that can help the manufacturer detect and resolve data quality issues quickly and effectively, ultimately leading to better decision-making and operational efficiency, supporting long-term success and growth.

5 PLANNING

In order to accomplish the proposed objectives, the subsequent steps have been followed:

- Conduct research on the best data quality practices and tools currently available on the market.
- Gather requirements from potential users of DataWash, including data analysts, data engineers, and business stakeholders.
- Design the DataWash's architecture considering best practices and industry standards.
- DataWash development using Snowpark, ensuring it meets the requirements gathered.
- Development of a set of data quality rules and checks that can be customized to meet the specific needs of the organization.
- Design and implementation of a user-friendly interface for DataWash using Streamlit, which includes interactive dashboards for data analysis and reporting.
- Test DataWash thoroughly to ensure it works as expected and meets the requirements.
- DataWash deployment on a Snowflake instance and train users on how to use it effectively.
- Monitor DataWash's performance and update it as needed to ensure it remains effective and up to date.
- Document the configuration of each of the data quality rules that are part of the set offered by DataWash.
- Deploy DataWash on an Azure Container Instance using Docker and the Azure Container Registry.
- Leverage Apache Airflow to efficiently orchestrate DataWash's data quality rules.

- Analysis of DataWash costs and creation of a license that allows the code not to be distributed beyond the customer.

The planning diagram (Image 4) that was drawn up at the beginning of the project can be seen in the appendix section.

6 Methodology

To achieve the aforementioned objectives, a meticulous and systematic methodology has been employed, encompassing a series of well-defined steps and processes:

- Requirement gathering: The first step in the development of DataWash has been to gather requirements from potential users and stakeholders. This has involved conducting surveys, interviews, and focus groups to identify the specific data quality issues that organizations are facing. As well as carrying out market research on the main companies offering data quality tools.
- Implementation: The implementation phase has entailed the actual development of DataWash, including the integration with Snowflake. This has required the use of Snowpark, Snowflake's advanced data processing engine, to provide a comprehensive suite of data quality modules that can be easily integrated into Snowflake pipelines. In addition to the above, a Power BI report containing an in-depth analysis of the results obtained by each module and the warnings received during each execution has also been developed.
- Design: Once the requirements have been gathered and the implementation of the set of data quality rules has been successfully carried out, the next step has been to design the user interface that will allow us to configure the parameters of each data quality rule. To this end, wireframes and prototypes of the interface have been created using Streamlit in order to incorporate the set of data quality rules and checks that can be customized to meet the needs of each organization.
- Testing and validation: Once developed, DataWash has been thoroughly tested and validated to ensure its accuracy, reliability, and efficiency. This has involved both automated and manual testing, as well as user testing to gather feedback and identify areas for improvement.
- Cost analysis: An analysis of the costs of implementing DataWash in Snowflake has been carried out considering the pricing information available on the Snowflake website. To perform this analysis, we compared the execution time taken by DataWash for different input data sizes.

7 Architecture

The architecture of DataWash (Image 6) is designed to be a comprehensive and flexible solution for ensuring high-quality data in Snowflake. It is composed of several components that work together to provide an end-to-end data quality solution.

In essence, it is engineered to be modular and scalable, enabling organizations to tailor the solution to their specific data quality needs and easily expand as their data volumes and complexity grow.

7.1 What is Snowflake and what benefits does it bring to DataWash?

Snowflake is a cloud-based data warehousing platform that provides scalable, high-performance storage and processing of large datasets [8]. It is designed to handle a wide range of workloads, from traditional data warehousing and business intelligence to advanced analytics and machine learning. Snowflake's architecture (Image 7) is based on a multi-cluster, shared-data approach that separates storage from compute, providing a flexible, cost-effective solution for managing large amounts of data.

One of the key benefits of Snowflake is its ability to scale seamlessly to handle large, complex workloads. With Snowflake, users can easily add and remove computing resources as needed, and data is automatically distributed across the underlying storage infrastructure, ensuring high performance and reliability.

For DataWash, Snowflake provides several benefits. First and foremost, Snowflake serves as the data storage and management platform for the tool. This means that all data processed and analyzed by DataWash is stored in Snowflake, making it easy to access and manage. Snowflake also provides a high degree of security and data protection, ensuring that all data is kept confidential and protected against unauthorized access.

In addition to its data storage and management capabilities, Snowflake also provides a powerful computing platform for running complex data analytics and machine learning workflows [9]. DataWash leverages this computing capability to perform its data quality checks, which can be computationally intensive, particularly when analyzing large datasets. By running these workflows in Snowflake, DataWash can take advantage of Snowflake's scalable computing infrastructure, ensuring that the tool can process large amounts of data quickly and efficiently.

All things considered, Snowflake's scalability, ease of use, and security features make it a valuable resource for DataWash, helping to ensure that the tool can process and analyze data efficiently and effectively while always keeping it secure.

7.2 What is Snowpark and what is it useful for?

DataWash is built using Snowpark [10], a collection of

user-defined functions, user-defined aggregate functions, user-defined types, and procedures that runs inside Snowflake. Snowpark enables you to perform complex data processing and transformations within Snowflake without having to transfer the data to another service or tool.

Snowpark is also an open-source project, allowing users to write and run Spark-style computations in Snowflake. It allows developers to use familiar Spark APIs and programming languages like Scala, Java, and Python to build data processing applications on Snowflake's cloud data platform, as can be seen in the figure linked below (Image 8). It also enables fast and efficient processing with the ability to scale resources as needed, making it the perfect tool for building large-scale data processing pipelines in Snowflake.

Snowpark can be useful for many data processing and transformation tasks, such as data quality checks, data standardization, data enrichment, and data integration, among others [11]. By leveraging Snowpark, you can perform these tasks within Snowflake, greatly improving the performance and efficiency of your data processing pipelines as well as simplifying the management of your data infrastructure. Additionally, Snowpark can help you to build custom data processing and transformation logic that can be easily integrated into your existing Snowflake environment.

7.3 Why choose Snowpark to implement DataWash modules?

One of the main benefits of using Snowpark is that it allows for the creation of highly performant and scalable data quality modules. Snowpark provides several features and optimizations that help speed up the processing of large datasets, such as data caching and indexing. These features help to ensure that the data quality modules implemented using Snowpark can handle large amounts of data efficiently and with minimal latency.

It should be noted that Snowpark provides a familiar and flexible programming environment for working with Snowflake data. This helps to streamline the development and implementation of data quality modules, as developers can leverage their existing knowledge of Snowflake and Snowpark to build new functionality quickly and effectively.

In addition to performance and scalability, Snowpark also provides several tools and resources to help with the development and maintenance of data quality modules. This makes it easier for developers to build, test, and debug their modules, and to collaborate with other members of their team to deliver outstanding data quality solutions.

Taking everything into account, Snowpark is designed to be easy to use and integrate with existing data management systems. This makes it a versatile and flexible platform for building custom data quality solutions that can be tailored to meet the specific needs of each customer.

8 Data quality ruleset

DataWash is a powerful data quality tool that offers a set of modules to ensure high quality data for businesses, each of which is designed to perform a specific task. These modules are designed to help organizations identify and correct data inconsistencies, inaccuracies, and duplicates, among others. Below, we will explore the different modules provided by DataWash, their functionalities, and how they can be used to enhance the overall quality of your data.

8.1 Advanced data quality rules

- Outlier Detector: This module performs an advanced outlier detection algorithm that checks if the data is within a certain range. Instead of setting the range manually, the module learns the range based on the distribution of the collected data. The more data collected, the more accurate the range will be. In this way, it helps to detect outliers in the dataset, letting the company know whenever an outlier is detected. In terms of the use case, it can predict when a sensor is going to fail in order to alert the manufacturer as soon as possible and find a solution instantly.
- Correlation Check: The module is used to analyze the relationship between two columns of data. The user specifies the two columns to be analyzed and the software computes the correlation between them. Any row with an atypical correlation will be flagged as a warning for the Data Steward to review. For instance, if the user is analyzing a worker's salary and hours worked and there is a disproportionate relationship between the two columns, the system will detect this and flag it as a warning.
- Outlier Length: This module checks the length of the values in a specified column to see if they fall within an expected range. The expected range is determined by computing the mean and standard deviation of a fixed number of preceding values. This module sends a warning whenever the length of a value is outside the expected range.
- Distribution Check: It is used to determine whether the distribution of a column has changed over time. The user specifies the column to be analyzed and the software compares the distribution at two points in time using the Kolmogorov-Smirnov test [12], which is a non-parametric test that determines the goodness-of-fit of two probability distributions to each other. Any changes in the distribution of the data will be flagged as a warning for the Data Steward to review.
- Freshness Check: The module is used to monitor the frequency and volume of new data being added to a dataset. The user specifies the average frequency of data being added, and the software will flag any significant deviation from this average frequency as a warning for the Data Steward to review. For exam-

ple, if there is a sudden increase in new data being added or a period without any new data, the software will detect this and flag it as a warning.

8.2 Standard data quality rules

- Categorical Check: This module is used to detect outliers in categorical fields. The software identifies unique or rare values in a target column specified by the user. These values are flagged as warnings for review by the Data Steward, who is also responsible for setting a threshold (which can be an integer value or a percentage) for what constitutes a rare value, and any value that occurs less frequently than the threshold will be considered rare and flagged.
- Duplicate Check: The module is used to detect duplicate rows in a dataset. The software scans the entire dataset to identify any duplicate rows and flags them as warnings for the Data Steward to review.
- Expected Range: Used to verify that the data in a column falls within a specified range of values. The user specifies the column to be analyzed and the expected range of values. If a higher percentage of values fall outside the expected range, the system will flag this as a warning for review by the Data Steward.
- Like Count: The module is used to count the number of rows in a column that match a specified format. The user specifies the expected format in the column, and the software will count the number of rows in the column that match this format. The software will also flag any discrepancies in the data to be reviewed by the Data Steward, which will define the maximum allowed percentage of values that do not correspond to the specified format within the selected column, beyond which a warning shall be sent.
- Null Count: It is used to count the number of null values in a specified column. The user specifies the column to be analyzed and the software will count the number of null values. Any discrepancy or error in the data will be reviewed by the Data Steward, who will define the maximum percentage of null values allowed within the specified column, above which a warning will be sent.

8.3 Data quality metrics

- Numeric Metrics: The module is used to perform basic mathematical calculations on the data in a dataset. The user specifies the column to be analyzed, and the software will perform calculations such as average, maximum, minimum, mean, standard deviation, sum, and variance. The Data Steward can also set parameters for each calculation, so that any value outside the expected range will be flagged as a warning to be reviewed in the overall analysis.

8.4 Data quality verifications

- Verify Country: This module checks whether the values in a specified column correspond to actual country names. If any value is misspelled or does not correspond to a real country, the module will generate a warning alert, prompting the Data Steward to perform a verification and rectification process.
- Verify Element: The module checks whether a specified item or a list of items is contained within a column. The module will return a positive result if the element is found within the column or a negative result otherwise.
- Verify Email: This module checks whether the values in a specified column correspond to valid email addresses. If any value does not correspond to a valid email address, the module will trigger a warning alert, thereby prompting the Data Steward to undertake an authentication and adjustment process.
- Verify Phone Number: The module checks if the values in a specified column correspond to valid phone numbers. If any value does not correspond to a valid phone number, the module will activate a warning alert, thus initiating a meticulous validation and correction process by the Data Steward.
- Verify Website: Checks whether the values in a specified column correspond to existing websites or IP addresses. The module shall establish a connection to the specified websites or IP addresses and verify the response to determine whether they exist. If any value does not correspond to an existing website or IP address, the module will trigger a warning alert, subsequently prompting the Data Steward to engage in a thorough process of verification and amendment.

Taking everything into account, DataWash is designed to help cleanse and validate data, ensuring it is accurate and trustworthy. The tool's modular architecture allows Data Stewards to choose the specific modules they need to address their unique data quality challenges.

To be able to go into more detail about the features of each module and consult all its documentation, just click on the link to the following reference [13], which corresponds to the GitHub repository of the project.

Ultimately, it is also worth noting that the abovementioned set of data quality rules is highly scalable, offering flexibility and suitability for each use case. Therefore, DataWash is not only limited to these data quality rules, but goes much further, offering rules adjusted to the profile of each customer and with the possibility of even extending the set of data quality rules that make up DataWash.

9 Working modes

The tool consists of two working modes: Ad Hoc / On-Demand analysis using Streamlit (Configuration App) and Scheduled Batch Execution (JSON).

9.1 Ad Hoc / On-Demand analysis using Streamlit (Configuration App)

At the core of the DataWash architecture is the Streamlit web app [14], which provides an easy-to-use interface for configuring and running data quality checks. This component is ideal for organizations that require on-demand data quality checks.

This first working mode allows real-time configuration and execution of data quality modules through the Streamlit web app.

Figure 1. Result of Ad Hoc / On-Demand analysis via Streamlit (Configuration App) for the Outlier Detector module.

In this way, once the parameters of the module to be executed are configured, the graphs and results obtained specifically for the module we have just set up are displayed on the screen via the user interface.

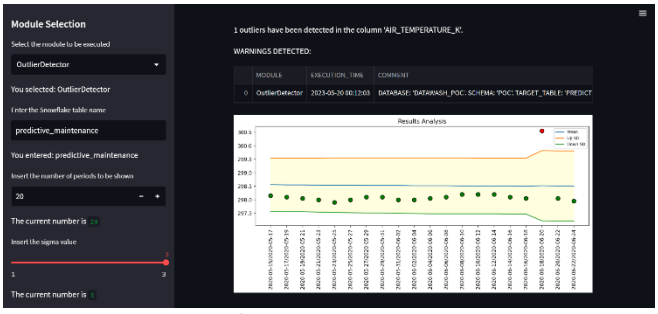


Figure 2. Ad Hoc / On-Demand analysis graphs via Streamlit (Configuration App) for the Outlier Detector module.

The idea is that, once the Data Steward has found the configuration and set of rules that best fit to monitor the organization's data quality, by simply clicking on the "Export Configuration" button in the user interface, the configuration of those data quality rules can be exported to the global repository (JSON) for scheduled batch execution.

The Ad Hoc / On-Demand working mode provides organizations with a flexible and customizable way for business and power users who might not have technical knowledge to create their own checks. The real-time configuration of the modules allows organizations to perform data quality checks whenever and wherever they are needed.

9.2 Scheduled Batch Execution (JSON)

This second working mode enables batch processing of data quality modules through a JSON file configuration. This working mode is ideal for organizations that require regular data quality checks. The parameters of each module are configured through a JSON file (which has previously been generated from the user interface), and each module has its own specific parameters to ensure its correct operation. The frequency of execution (daily, weekly, monthly, etc.) is determined by the customer's needs.

Once the execution is complete, a Power BI report containing the results of each module is generated [15], along with warnings detected in each module. This is one of the key features of DataWash, as it provides a comprehensive view of the data quality status and any issues that need to be addressed so that the Data Steward can carry out an in-depth analysis. This allows organizations to identify and resolve data quality issues before they become major problems.

It is worth mentioning that this report ingests through Snowflake the results obtained and the warnings detected that DataWash has previously persisted in table format at the end of its execution.

9.2.1 Power BI report automatically generated from each scheduled DataWash execution

The Power BI report generated by DataWash consists of three main sections: the landing page, the warnings page, and the module analysis pages. The landing page provides a brief overview of DataWash, including its purpose and key features. This page also includes a summary of the warnings generated during the last execution of DataWash, as well as a status indicator that shows whether the last execution was successful or not.

The overall warnings page (Image 9) provides a detailed analysis of all warnings generated during the last execution of DataWash. This page includes a table that lists all the warnings, along with information such as the module that generated the warning, the severity of the warning, and the description of the issue. Users can filter the warnings by severity or module, and they can also export the table to Excel for further analysis.

Additionally, the Power BI report generated by DataWash also includes module-specific pages that provide a detailed analysis of the execution of each module implemented in the tool. Each module-specific page includes information about the module's purpose, the tables it analyzed, and the warnings generated during its execution. The module-specific pages also provide visualizations that help to understand the distribution and severity of the warnings generated by each module.

Please note that for each of the module-specific pages that exist, different types of graphs have been generated depending on the needs of each data quality rule. In addition, each of them can be analyzed through the dossier, which contains the original Power BI report file.

Subsequently, in order to understand the structure of these module-specific pages, the analysis page corresponding to the Outlier Detector module will be shown in the following linked figure (Image 10).

By providing a comprehensive Power BI report, DataWash helps users easily visualize and analyze the results of their data quality assessments. This can help identify patterns and trends in data quality issues, which, in turn, can inform decisions about data quality improvement initiatives. Additionally, the report provides a clear and concise way to communicate the results of data quality assessments to stakeholders and decision-makers.

As has been demonstrated, DataWash provides organizations with a comprehensive and flexible data quality solution that can be tailored to their specific needs. The tool's two working modes enable organizations to perform periodic data quality checks as well as on-demand data quality checks to ensure the accuracy and reliability of their data.

10 Analysis of results: where is data persisted?

Once the modules have been properly configured and executed, the analysis of the results and any detected warnings are uploaded to Snowflake. Each module has its own analysis table, which allows users to drill down into the results and gain a more granular understanding of the data quality issues present. Additionally, there is a general warnings table that aggregates all the warnings detected across all modules.

DataWash meticulously adheres to a well-defined and systematic protocol for data persistence once the modules have been successfully executed. This procedure can be seen in detail below:

- Analysis Tables: For each module executed, DataWash creates an analysis table in Snowflake to store the results of the analysis. This table (Image 2) contains the data that has been analyzed and the results obtained, such as any outliers detected, length range violations, incorrect country names, incorrect element values, invalid email addresses, invalid phone numbers, or non-existent websites.
- General Warnings Table: In addition to the analysis tables, DataWash also persists in Snowflake a general warnings table, which is a unique table for all modules. This table (Image 3) stores all the warnings that have been generated by the different modules, allowing for a central repository of all the warnings generated during the data cleaning process. The general warnings table provides an overview of the data quality and helps the user to keep track of all the warnings generated by the different modules.
- Installation Table: A third type of table is also created, the installation table (Image 5) lists the version of the modules that have been executed. Additionally, it is worth mentioning that there is only one common installation table for each DataWash instance.

11 Deploying datawash on an azure container instance using docker and the azure container registry

To run the Streamlit web app, an Azure Container Instance is used, which allows the application to be deployed instantly without managing any server infrastructure. This is a significant advantage, as it eliminates the need for the user to worry about server management and maintenance [16].

Having said that, we will go through the necessary steps to deploy DataWash on an Azure Container Instance by creating a Docker container and the Azure Container Registry to be able to do so.

11.1 Creating the Docker Image

The first step in this process was to create a Docker image based on the DataWash code [17]. This was done by creating a Dockerfile, which specifies the dependencies and commands needed to build the image. The Dockerfile was based on the official Python image and included the necessary dependencies to run the DataWash code. Once the Dockerfile was created, the image was built using the Docker build command, which created the image based on the instructions in the Dockerfile.

11.2 Creating the Azure Container Registry

Subsequently, an Azure Container Registry was created to store the Docker image. This was done by using the Azure portal to create a new container registry resource, selecting the appropriate subscription, resource group, and name for the registry. Once the registry was created, it was secured with access keys and policies to control access to the registry.

11.3 Pushing the Docker Image to the Azure Container Registry

The next step was to push the Docker image that had been previously created to the Azure Container Registry. This was done by using the Docker push command, which pushed the image to the registry using the registry's login credentials. This process required ensuring that the Docker CLI was properly configured to authenticate with the Azure Container Registry.

11.4 Creating the Azure Container Instance

Finally, an Azure Container Instance was created to run the DataWash container. This was accomplished by leveraging the capabilities of the Azure portal to generate a new container instance resource. The process involved carefully choosing the relevant subscription, resource group, and a suitable name for the instance [18]. The image that had been pushed to the Azure Container Registry was selected as the image source for the instance, and the necessary environment variables were configured to run the DataWash code [19].

11.5 Advantages

In essence, the above process provided a scalable and easy way to run the DataWash code without managing

any server infrastructure. This procedure can be used to execute DataWash in a cloud-based environment, enabling greater flexibility and scalability for data quality monitoring and analysis. Furthermore, this allows customers who are inclined towards acquiring DataWash as a solution to their company's data quality issues to conduct direct testing of the tool.

12 Leveraging Apache Airflow to Efficiently Orchestrate DataWash's Data Quality Rules

To effectively orchestrate the execution of the different DataWash data quality rules, we have leveraged the capabilities of Apache Airflow in conjunction with Snowpark [20]. As the core of these data quality rules is developed using Snowpark, the goal was to seamlessly coordinate and manage the execution of Snowpark jobs to address this challenge.

Apache Airflow serves as a powerful workflow management platform that allows us to define, schedule, and monitor complex data pipelines [21]. By leveraging Airflow's flexible and scalable architecture, we have designed a comprehensive orchestration system for DataWash.

The orchestration process starts by configuring Apache Airflow to execute Snowpark jobs, which are responsible for performing the data quality checks and analysis defined by the DataWash rules. Each Snowpark job encapsulates the logic required to validate and cleanse the data according to the specific rule [22].

The subsequent step involved creating individual tasks within Airflow to represent each data quality rule. These tasks were then organized into workflows, where dependencies and the order of execution were defined. This allowed us to establish the sequence in which the Snowpark jobs should be executed to ensure the accuracy and efficiency of the data quality rules.

With Apache Airflow, we have been able to schedule the execution of these tasks based on predefined intervals or triggered events. This automated scheduling eliminated the need for manual intervention, ensuring that the data quality rules were executed consistently and in a timely manner.

Moreover, Apache Airflow provides rich monitoring and logging capabilities, allowing us to track the progress and status of each job. We can easily monitor the execution of DataWash data quality rules, view any error or warning messages, and take appropriate actions if needed.

13 The Cost-Effectiveness of DataWash

To determine the cost-effectiveness of DataWash, we analyzed the cost of running all its modules for different table sizes. The cost of Snowflake's computing resources was calculated using the pricing information available on the Snowflake website [23]. The following figure shows the cost of running all DataWash modules for different table sizes:

Table Size	Snowflake Computing Cost
100 records	$0.099
1,000 records	$0.198
10,000 records	$0.501
100,000 records	$1.336
1,000,000 records	$3.633

Figure 3. Analysis of DataWash costs based on the number of records per run.

Please note that the above computing cost has been calculated based on the runtime spent by DataWash considering the total number of records in the input data and comparing it with the pricing information available on the Snowflake website.

According to the information provided, a complete run of all DataWash modules using 100-record tables costs $0.099 of Snowflake's computing resources, while a complete run using 1,000-record tables costs $0.198 of Snowflake's computing resources. For tables of 10,000 records, the cost of a complete run of all DataWash modules increases to $0.501 of Snowflake's computing resources. The cost of running DataWash on 100,000-record tables is $1.336 of Snowflake's computing resources, and for 1,000,000-record tables, the cost is $3.633 of Snowflake's computing resources.

Based on the above analysis, we can conclude that DataWash is highly cost-effective. The cost of running DataWash for smaller tables is relatively low, with the cost increasing as the table size grows. However, even for a table size of 1,000,000 records, the cost is still reasonable, and the benefits of ensuring high data quality far outweigh the cost.

14 Conclusion

As has been shown in this paper, DataWash is a powerful data quality tool that can help organizations improve the accuracy and reliability of their data. With its modular architecture and ability to run on Snowpark, DataWash offers a flexible and customizable solution for a wide range of data quality needs.

The importance of data quality is only increasing as more and more businesses rely on data-driven decision-

making. The need for accurate and reliable data is critical, and DataWash provides a streamlined solution for ensuring that data is of high quality, which can ultimately improve business outcomes.

Furthermore, the ability to customize modules in DataWash to fit the specific needs of different businesses and industries makes it a highly adaptable tool. This flexibility, combined with the power of Snowflake, advanced machine learning techniques, and the simplicity of the Streamlit user interface, makes DataWash a strong contender for becoming a product in high demand by major companies.

As possible improvements and future lines of DataWash, it is worth mentioning that, due to the flexibility of its architecture, it is highly scalable for the integration of new data quality rules as well as the customization of the parameters of existing rules. This demonstrates that DataWash is easily expandable by adding new modules to customize the experience for each customer, thus becoming a solution for any type of customer, regardless of their background.

Overall, DataWash represents a significant step forward in the field of data quality management and has the potential to transform the way organizations handle their data. With ongoing development and improvements, it is likely to continue to be an essential tool for businesses across industries for years to come.

If you would like to find out more about DataWash and how it works, here are links to a mini-series of two blogs written and published by me on the ClearPeaks website. The first blog [24] explores the features, benefits and use cases of DataWash and compares it to other popular solutions on the market, while the second blog [25] looks at the technical details to see how it works behind the scenes.

15 Bibliography and References

[1] Data Quality Management For Data Warehouse Systems. [Consulted: March 2023]. Available on the Internet: https://ceur-ws.org/Vol-2351/paper_27.pdf

[2] Soda Data Quality Platform. [Consulted: March 2023]. Available on the Internet: https://www.soda.io/

[3] Great Expectations: The Data Testing Tool. [Consulted: March 2023]. Available on the Internet: https://greatexpectations.io/

[4] Data testing in dbt. [Consulted: March 2023]. Available on the Internet: https://www.getdbt.com/product/data-testing/

[5] Collibra Data Quality & Observability. [Consulted: March 2023]. Available on the Internet: https://dq-docs.collibra.com/

[6] Data Quality - Ataccama. [Consulted: March 2023]. Available on the Internet: https://www.ataccama.com/platform/data-quality

[7] Matillion Data Quality Framework. [Consulted: March 2023]. Available on the Internet: https://www.matillion.com/about/news/matillion-data-quality-framework/

[8] Snowflake Documentation. [Consulted: April 2023]. Available on the Internet: https://docs.snowflake.com/

[9] The Snowflake Elastic Data Warehouse. [Consulted: April 2023]. Available on the Internet: https://dl.acm.org/doi/pdf/10.1145/2882903.2903741

[10] Snowpark Developer Guide for Python. [Consulted: April 2023]. Available on the Internet: https://docs.snowflake.com/en/developer-guide/snowpark/python/index

[11] What is Snowpark — and Why Does it Matter? A phData Perspective. [Consulted: April 2023]. Available on the Internet: https://www.phdata.io/blog/what-is-snowpark/

[12] Kolmogorov–Smirnov test. [Consulted: April 2023]. Available on the Internet: https://en.wikipedia.org/wiki/Kolmogorov%E2%80%93Smirnov_test

[13] DataWash GitHub repository. [Consulted: March-June 2023]. Available on the Internet: https://github.com/ClearPeaks/snowflake-datawash

[14] Streamlit Documentation. [Consulted: April 2023]. Available on the Internet: https://docs.streamlit.io/

[15] Power BI Documentation. [Consulted: April 2023]. Available on the Internet: https://learn.microsoft.com/en-us/power-bi/

[16] Deploy Streamlit using Docker. [Consulted: May 2023]. Available on the Internet: https://docs.streamlit.io/knowledge-base/tutorials/deploy/docker

[17] How to "Dockerize" Your Python Applications. [Consulted: May 2023]. Available on the Internet: https://www.docker.com/blog/how-to-dockerize-your-python-applications/

[18] Quickstart: Deploy a container instance in Azure using the Azure portal. [Consulted: May 2023]. Available on the Internet: https://learn.microsoft.com/en-us/azure/container-instances/container-instances-quickstart-portal

[19] Set environment variables in Azure container instances. [Consulted: May 2023]. Available on the Internet: https://learn.microsoft.com/en-us/azure/container-instances/container-instances-environment-variables

[20] Using Airflow with Snowpark. [Consulted: May 2023]. Available on the Internet: https://www.mobilize.net/blog/using-airflow-with-snowpark

[21] Apache Airflow Documentation. [Consulted: May 2023]. Available on the Internet: https://airflow.apache.org/docs/

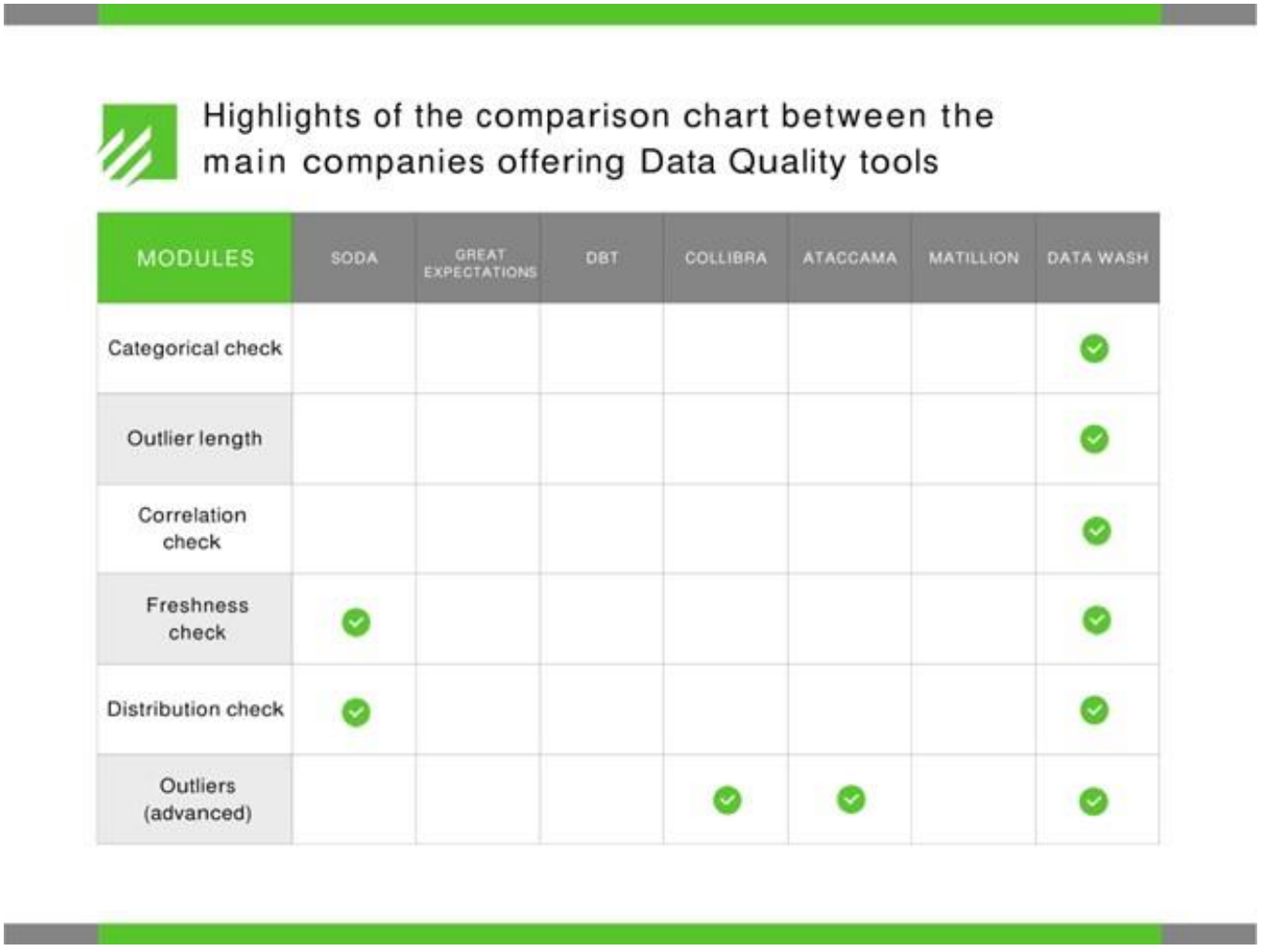
[22] Data Engineering with Apache Airflow, Snowflake & dbt. [Consulted: May 2023]. Available on the Internet: https://quickstarts.snowflake.com/guide/data_engineering_with_apache_airflow/

[23] Snowflake Pricing & Cost Structure. [Consulted: May 2023]. Available on the Internet: https://www.snowflake.com/pricing/

[24] DataWash: An Advanced Snowflake Data Quality Tool Powered by Snowpark – Part 1. [Published by me in May 2023]. Available on the Internet: https://www.clearpeaks.com/datawash-an-advanced-snowflake-data-quality-tool-powered-by-snowpark-part-1/

[25] DataWash: An Advanced Snowflake Data Quality Tool Powered by Snowpark – Part 2. [Published by me in May 2023]. Available on the Internet: https://www.clearpeaks.com/datawash-an-advanced-snowflake-data-quality-tool-powered-by-snowpark-part-2/

APPENDIX



Highlights of the comparison chart between the main companies offering Data Quality tools

MODULES	SODA	GREAT EXPECTATIONS	DBT	COLLIBRA	ATACCAMA	MATILLION	DATA WASH
Categorical check							✓
Outlier length							✓
Correlation check							✓
Freshness check	✓						✓
Distribution check	✓						✓
Outliers (advanced)				✓	✓		✓

Image 1. Highlights of the comparison chart.

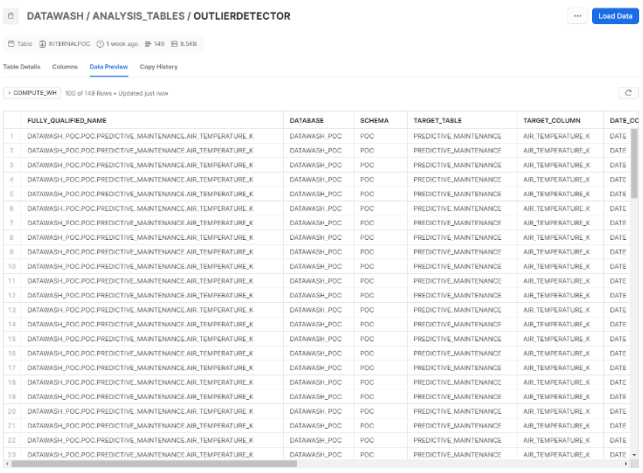


Image 2. Snowflake analysis table persisting the results of running the Outlier Detector module.

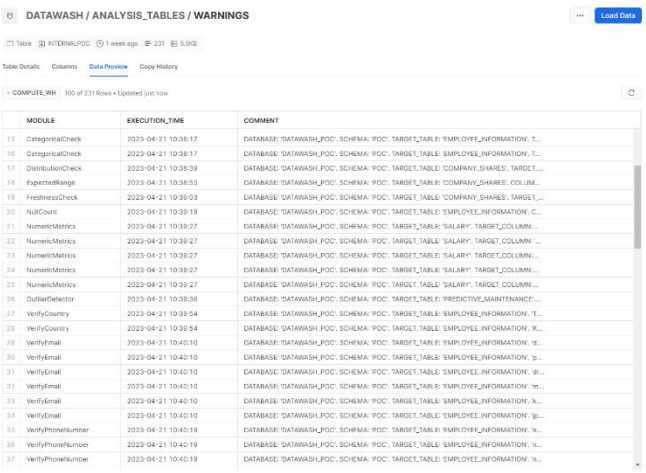


Image 3. Snowflake warnings table that persists the warnings sent by each module during DataWash execution.

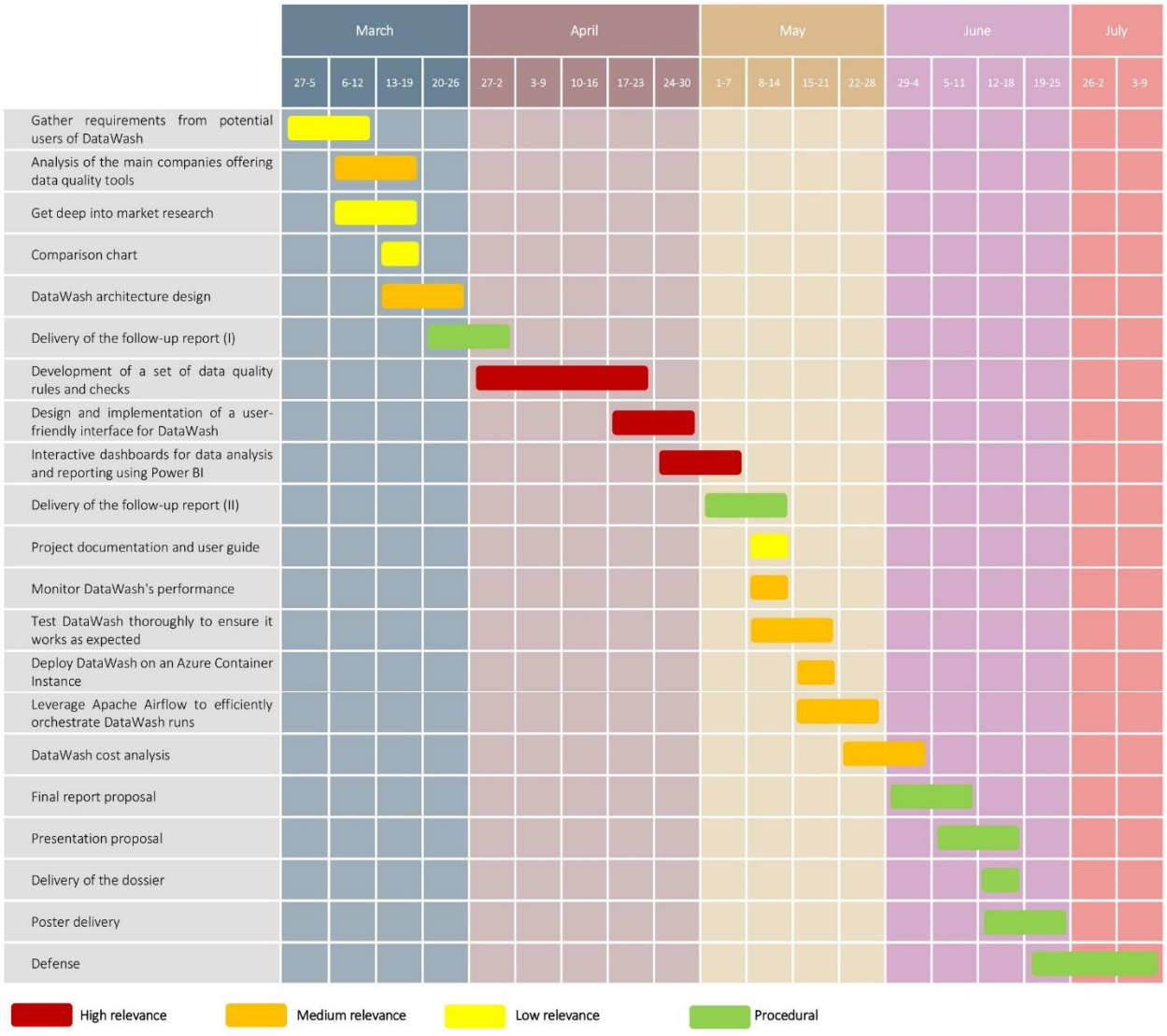


Image 4. Planning diagram.

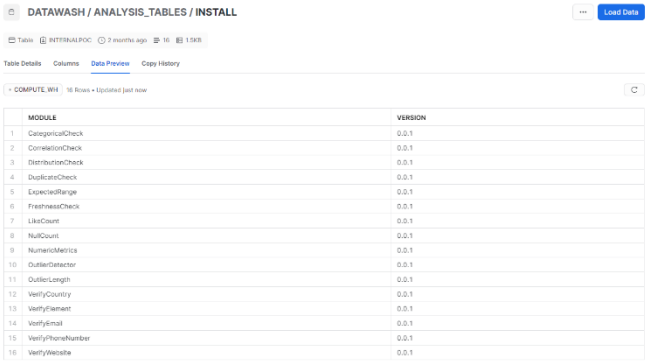


Image 5. Snowflake installation table that persists the version of the DataWash modules that have been executed.

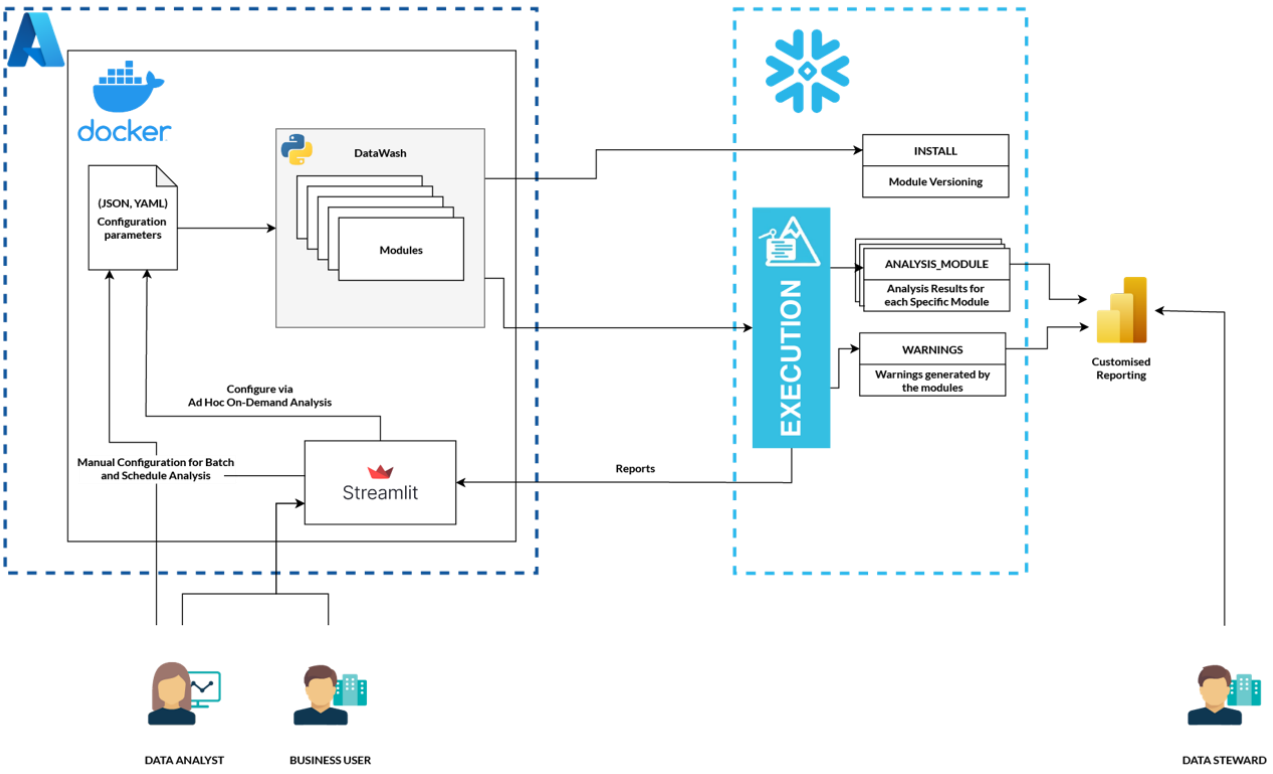


Image 6. Overview of the DataWash architecture: scalable and flexible cloud-based solution for data quality.

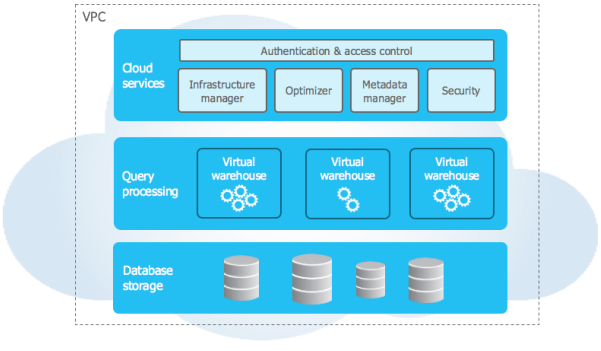


Image 7. Snowflake architecture.

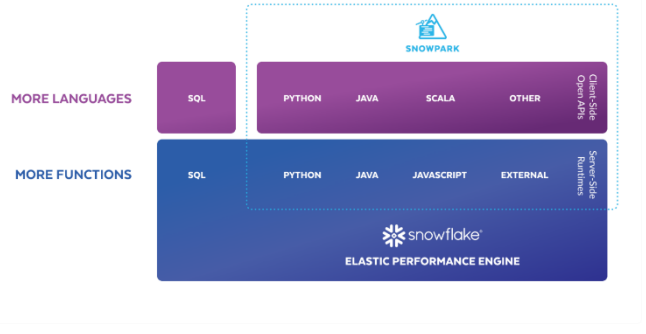


Image 8. Snowpark programming languages and computing engine.

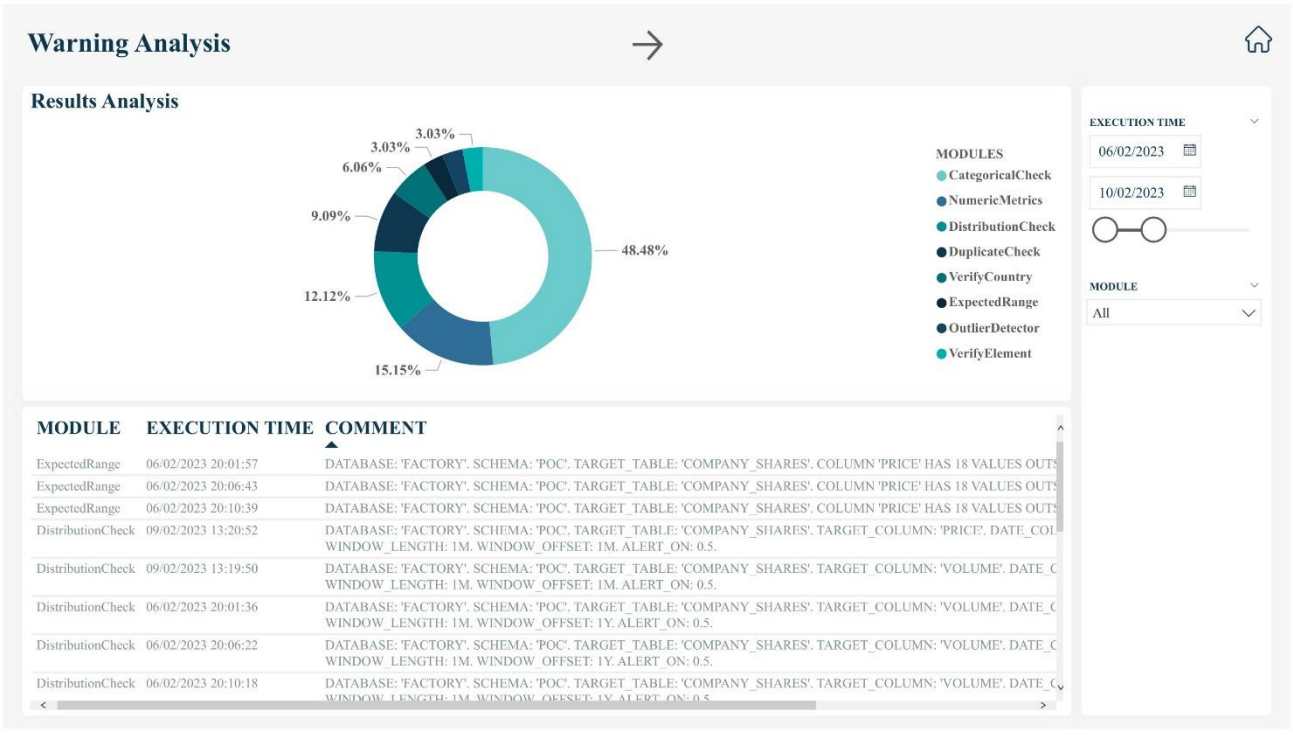


Image 9. Warning analysis page of the Power BI report generated by DataWash.

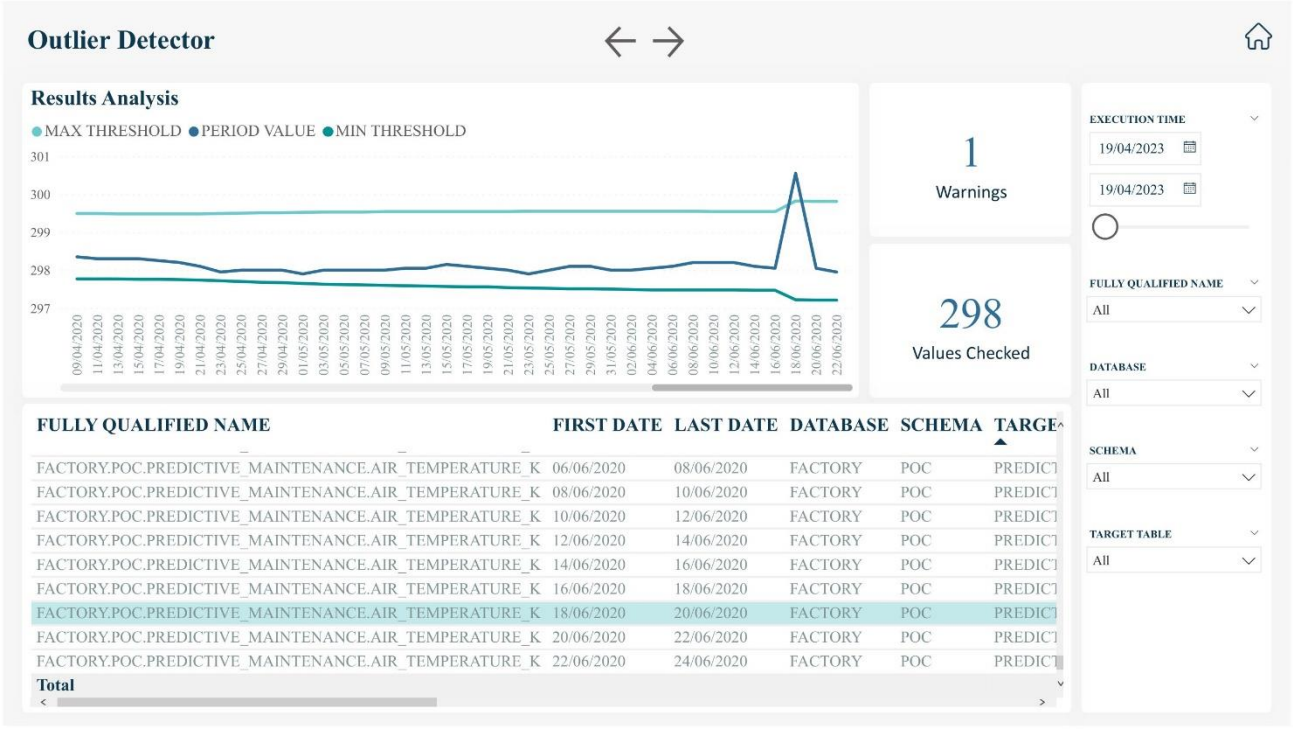


Image 10. Outlier Detector module results analysis page of the Power BI report generated by DataWash.