REFLECTION REPORT

The first task of the Project involves combining four different datasets, each in different data formats into a single unified record. The four datasets given are in the following formats: CSV, XML, TXT and JSON. Each data file contains different customer data attributes for a thousand customer.

The CSV file which is a delimited text file that uses a comma to separate values and allows data to be saved in a tabular format contains vehicle information of the customers (Wikipedia, 2021a). The XML file which is an extensible markup language file, with structured tags and text contains job information of the customers (Wikipedia, 2021b). The TXT file which is a file with sequence of lines in plain texts gives Internal messages between colleagues, E-mail message sent to a customer and Confidential internal memo. Lastly the JSON file, a standard text-based format which is primarily used for sending data in web applications contains credit card information of the customers (Wikipedia, 2022).

My workflow/methodology for the task went as follows:

- Load and read the four data sets into a pandas data frame. The CSV and JSON files were easily parsed to a pandas data frame using the read_csv and read_json features of the pandas library respectively. The XML file was parsed using the xml.etree.ElementTree, a package that represent XML documents/files in tree formats allowing for easy parsing of xml files.

- After reading the respective datasets into a pandas data frame as stated above, a new column titled NAME was created for all data frames, the Name column contains the full name of all customers. The data frames were then sorted alphabetically based on the name column created, making each customer to bear the same index number across the three data frames.

- Since each customer now carries same index number across the three data frames, the data frames can now be merged into a singular record using the index number as a form of primary ID. The merge was done using another feature of the pandas package called concat which concatenates pandas objects along a particular axis. Multiple columns that arise due to joining the data frames were deleted.

- The text file was read manually to extract valuable information, the information extracted were updated into the merged data frame.

- The last aspect of the task was to write the merged data frame into a database using python Pony ORM.

From a data perspective, the main challenge encountered during the course of the project was in combining the different datasets all in different formats into a unified record. This challenge was solved by first putting all the data sets into similar format; pandas data frame. Another challenge then came up in converting the XML file into pandas data frame, while CSV and JSON have features in the pandas package that easily converts them into a data frame, the xml file had to be parsed using another package, xml.etree.ElementTree.

Finally, automating the process for future endeavors will be of great difficulty because of the text file present. Information in the text file can only be extracted manually and that is by reading.

PONY ORM

INTRODUCTION

Pony ORM is an object relational mapper library which can make development simple as we can communicate with the database using python, increasing the speed of development, avoids use of redundant code, and ease switching between database. However, in other cases, it could become problematic when it comes to schema evolution, scalability, performance.

Disadvantages: complex ORM calls can be inefficient, ORMs map the underlying data stored in a database into an object that can be used in our application. This can be inefficient when we need to fetch certain complex information from certain database. This is because while ORMs can generate queries, they are prone to fetching more information than required to perform the task, thereby increasing the cost of the generated query when compared to queries written specifically for the task required. Strictly using ORMs can limit the calls we make to the database. Some calls cannot be made with ORMs due to the complexities involved between the several layers to generate the required queries. In these situations, we need to generate the SQL query required by hand. This means our codebase will have a combination of ORM and hand-written SQL. The implication is that we now have a codebase complex as we're now using two languages to fulfil similar task, which defeats the purpose of ORMS.

ALTERNATIVES

1. Use of raw SQL.

2. Query builders: Query builders are designed to enhance productivity and simplify SQL query building tasks. Query Builder provides a graphical user interface for creating SQL queries. You can drag-and-drop multiple tables, views and their columns onto a visual designer to generate SQL statements. (Documentation, n.d.)

COMPARING AND CONTRASTING PONY ORM AGAISNT QUERY BUILDERS

Solution one utilizes pony ORM to map objects in their codebase to the database without writing SQL, however there is still a need to write SQL for some calls. Although it increases the speed of development compared to writing SQL all the time, this is done at the expense of complexity and the quality of their codebase. Query builders on the other hand generates SQL queries by interacting with the graphics user interface. They work directly with SQL queries which means they can avoid complexity of using additional dependencies in their codebase. The query builder is used as a matter of convenience to generate queries quickly.

RECOMMENDATION

There are certain tradeoffs of using different layers of abstracting database interactions such as ORMs, query builders. The SQL queries being generated in each level can be different and carry different running cost; including dynamic queries.

Furthermore, we realized that despite using ORMs we still have to generate certain queries by hand.

These issues can be fixed using query builders such as Syncfusion.

I recommend the use of query builders for expansion.

BIG DATA ISSUES

As the operations in a data driven organization scales up, the volume of data begins to accumulate which will facilitate detection of data patterns increase efficiency and service. This enormous quantity of data also means they have to develop a robust pipeline to collect, store and process the data. This pipeline must satisfy the legal requirements as well as utilization of a technological structure to serve the purpose.

TECHNOLOGICAL ISSUES

Storage and Processing Infrastructure:

The storage capacity and method of storage is an important issue for big data. As data is becoming more, local storage solutions such as flash storage, is becoming a requisite for competitive advantage etc.

The next issue is processing power. Data processing occurs when data is collected and translated into usable information. It is important to select the appropriate processing method in order to optimize the insight generation process

Big data processing is a set of techniques or programming models to access large-scale data to extract useful information for supporting and providing decisions.

The infrastructure required for both storage and processing are quite expensive and requires high technicality for optimal use. This can be replaced by the use of cloud computing options such as google cloud, amazon cloud.

LEGAL ISSUES

Data Protection and Privacy: Data protection involves safeguarding customer information from compromise, corruption and other activities that can pollute the integrity of the data.

"Of equal concern is the collection, use and sharing of personal information to third parties without notice or consent of consumers. 128 out of 194 countries had put in place legislation to secure the protection of data and privacy." (United Nations Conference on Trade and Development, n.d.). We can see that the importance of data privacy and data protection is increasingly recognized.

A potential solution is to encrypt or mask the data. However, this approach still leaves the integrity of the data at the mercy of the method of encryption.

Also, maintaining anonymity can potentially solve this problem because an anonymized data is no longer under a personal data.

Intellectual Property Rights: Intellectual property (IP) is simply the right of a creator to control his or her creations. It is necessary to the data being collected or how much of the data your company actually owns.

As the operations of a company expands, they have to ensure their IPR is protected in accordance with the IPR of the other countries they are operating in. This is because Patents and trademarks are territorial and must be registered in each country where protection is sought.

It may be helpful to seek legal counsel from a local expert to ensure proper conducts and potential solutions as Intellectual property laws can be complex.

Contractual liability: Contract liability is when one party to a contract agrees to reimburse any damages or losses suffered by another party.

In situations where third parties are involved, i.e., when analysis is done by the third party, both parties can sign contractual liability insurance where the supplier warrantees the data.

Contractual liability insurance protects against liabilities that the policyholder has assumed from entering into a contract of any nature. (contractscounsel., n.d.)

References

Wikipedia. (2021). Comma-separated values. [online] Available at: https://en.wikipedia.org/w/index.php?title=Comma-separated_values&oldid=1060298461 [Accessed 10 Jan. 2022].

Wikipedia. (2021). XML. [online] Available at: https://en.wikipedia.org/w/index.php?title=XML&oldid=1062340378 [Accessed 10 Jan. 2022].

Wikipedia. (2022). JSON. [online] Available at: https://en.wikipedia.org/w/index.php?title=JSON&oldid=1063500160 [Accessed 10 Jan. 2022].

contractscounsel., n.d. *Contract Liability: What is it.* [Online]
Available at: https://www.contractscounsel.com/b/contract-liability
[Accessed 16 January 2022].

Documentation, n.d. *dbForge Studio for MySQL.* [Online]
Available at: https://docs.devart.com/studio-for-mysql/building-queries-with-query-builder/query-builder-overview.html
[Accessed 16 January 2022].

United Nations Conference on Trade and Development, n.d. *Data Protection and Privacy Legislation Worldwide.* [Online]
Available at: https://unctad.org/page/data-protection-and-privacy-legislation-worldwide
[Accessed 16 january 2022].