Migração de dados para a Cloud em implementações ERP

JOÃO RUI DA SILVA AMORIM
outubro de 2020

Data Migration to Cloud in ERP Implementations

João Rui Da Silva Amorim

**A dissertation submitted in partial fulfillment of
the requirements for the degree of Master of Computer Science,
Specialisation Area of Software Engineering**

Supervisor: Nuno Silva
Co-Supervisor: Carlos Ferreira

Dedicatory

"To my parents, my uncles, my cousins, my friends, all my co-workers and teachers who have always supported and encouraged me throughout this rewarding academic journey. Without them, nothing of this would be achieved."

Abstract

The concept of Cloud Computing has evolved constantly, in strand of service models, based on the creation and share of several technological resources. Increasingly, it has used virtualization technology to optimize resources, which are shared by all accounts, in a self-service format. All these features result in a flexible and progressive behavior of resources. The management of the service provided, is made based on the service level agreement established between the client and the cloud provider, and the constant technological developments can quickly change depending of the requirements. That said, knowing the current state of Wipro with respect to the concepts of data migration and cloud being combined into one, it is very challenging to design and build a possible process to help the company make this transition. Especially, when there is already a tool that has been used for several years and it is intended to be part of this integration with a new possible solution that is described along this document. The study, of qualitative nature, is guided by different case studies when it comes to the processes being used to migrate data into Cloud. The main objectives are to find new solutions that increase productivity of the company, save human resources that can be reallocated to other tasks, ending up to be considered innovative solutions, with rapid implementations and most importantly with low cost. What ends up in the overall objective of this dissertation that is to examine the feasibility of the adoption of Cloud Computing in Wipro Portugal through two main points:

- The migration of data into Cloud;

- Integration with Data Conversion Tool (DCT).

We believe that this approach is very meaningful towards encouraging greater productivity and obtaining new achievements. Concerning the empirical study, there is a big number of tools that can be investigated later as possible solutions for other kind of implementations than Oracle Retail. For now this dissertation focus in the current OR business approaches and points for Oracle Cloud as the main Cloud Computing Service due to its partnership with Wipro. Both solutions that were implemented, SQL Loader and Golden Gate, seem viable and versatile as they can be integrated with the current tool, DCT and are capable of loading several amounts of data without any issues. In terms of performance Golden Gate seems to be a few steps above of SQL Loader, but requires deeper analysis when using multi threading as an option in both methods, and the containerization of the databases can be very relevant regarding the times of loading. In general, good solutions are available and need to be taken into consideration by the company as they can help to leverage its resources in a more efficient way and the main objective of having data in the Cloud was reached, having gathered knowledge about the behaviour of Oracle Cloud and some of the services.

Keywords: Data Migration, Cloud, System Integration, Database, Performance and Innovation

Acknowledgement

"I thank all those who supported and contributed to the project developed; to my parents and uncles who supported me unconditionally, providing help whenever possible for any task that had to be solved and who tried and are trying to get the best of me; to my friends who are the support, one of the biggest pillars behind this academic path, who were with me in both good and bad times, without them none of this would be the same; to all the teachers who accompanied me during my degree and masters, specially my supervisors Nuno and Carlos, for making me always aim more and who were fantastic and supportive during this stage of my life. Thank you very much and I wish you all the best."

Contents

List of Figures

List of Tables

List of Acronyms

ADW	Autonomous Data Warehouse.
ATP	Autonomous Transaction Processing.
AWS	Amazon Web Services.
CDB	Containerized Database.
CS	Cloud Service.
DB	Database.
DBaaS	Database-as-a-service.
DCT	Data Conversion Tool.
DCTDC	DCT Control Schema.
DCTDR	DCT Repository Schema.
DDL	Data Definition Language.
DMA	Data Migration Assistant.
DML	Data Manipulating Language.
DMS	Data Migration Service.
DR	Data Replication.
DSRM	Design Science Research Methodology.
ELT	Extraction, Loading and Transformation.
ERP	Enterprise Resource Planning.
ETL	Extraction, Transformation and Loading.
GG	Golden Gate.
GPU	Graphical Processing Unit.
HR	Human Resources.
IS	Information systems.
OCI	Oracle Cloud Infrastructure.
OR	Oracle Retail.
ORMFCS	Oracle Retail Merchandise Foundation Cloud Service.
PaaS	Platform-as-a-service.
PDB	Plugable Database.
QEF	Quality Evaluation Framework.
RAM	Random-access Memory.

ReSA	Retail Sales Audit.
RPM	Retail Pricing Management.
SaaS	Software-as-a-service.
SFTP	SSH File Transfer Protocol.
VC	Value for the Customer.

Chapter 1

Introduction

This chapter provides an overview of the key concepts that led to the development of this dissertation, including a short description of the main topic on which it was based. A company review is done in order to understand which business and market Wipro Portugal faces, giving a context about its actual position when it comes to data migration and the importance of Cloud in the same. Finally, a practical approach on how to execute a data migration will be described, integrating it with an already used tool in the company that is going to be detailed further ahead. The following points will be focused with more detail to ensure which path should be taken:

- Problem;
- Objectives;
- Preconized Approach;
- Structure of the document.

1.1 Company Review

Wipro Portugal is a subsidiary of Wipro Limited, which is a multinational information technology company, with consulting services. This is a company recognized globally for its comprehensive services portfolio, strong commitment to sustainability, good corporate citizenship, with more than 160,000 employees dedicated to serving customers on six continents. Wipro Portugal provides consulting services for the retail area, namely Enterprise Resource Planning (ERP) software. Strongly oriented towards the international market, Wipro Portugal is proud to serve retailers and wholesalers who are also international leaders, such as Sonae (Wire n.d.). This company was born in 1997 under the name of Enabler after a spin-off from the IT Department of the company Sonae / Continente. In 1999, won his first international contract at DeSpar in Italy, in that same year, started co-developing the Oracle ERP deployment capabilities for Sonae. Between 2000 and 2006 it broadens its horizons for LATAM, US and Europe, opening offices in Braga, Reading, Paris, Dusseldorf, Curitiba and Minneapolis and manages Human Resources and Finance processes in six countries (Wire n.d.). In 2006 Enabler was acquired by Wipro Limited for its extensive knowledge in retail and implementation of the Oracle Retail ERP package. Between 2006 and 2008, mergers and acquisitions are carried on, incorporating Wipro HR and Finance. A brief history can be seen in the following Figure 1.1.
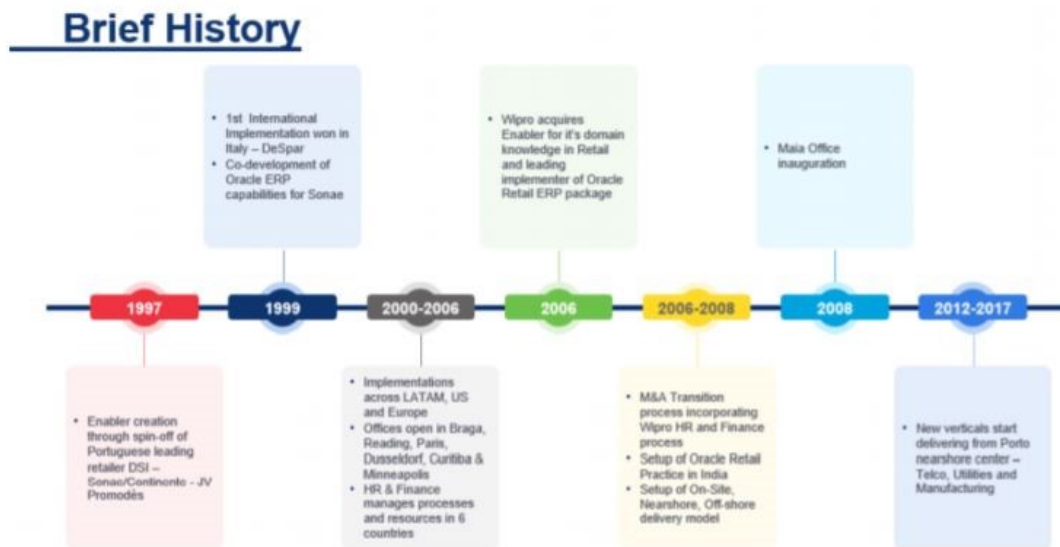
Brief History

Figure 1.1: Brief History of Wipro Portugal

1.2   Context

Wipro covers a large market related to Retail, and it is well known for its Oracle Retail implementations. These implementations can offer several applications that can manage efficiently every entity of merchandise, including stores, inventory, items, transfers, shipments, prices and many others. One of the biggest challenges, which has a huge cost for ERP implementations, is the ability to move data among different applications. Basically ERP is a business process management software that allows an organization to use a system of integrated applications to manage business and automate many different operations that require specific services or even human resources (Beal 2019). Data from the existing old or legacy application needs to be cleaned up and organized to ensure its integrity in the new system. Data migration is a complex process that describes the data transfer between computer storage types or file formats. It is a key consideration for any system implementation, upgrade, or consolidation. It is usually performed programmatically to achieve an automated migration, freeing up human resources from tedious tasks (Taylor 2019). Data migration occurs for a variety of reasons:

- Server or storage equipment replacements;

- Maintenance or upgrades;

- Application migration;

- Website consolidation;

- Data centre relocation.

Data migration can be challenging when sources and targets are numerous and heterogeneous, this will trigger many changes when it comes to the syntax and semantics of the data. To achieve an effective data migration procedure, data on the old system is mapped

to the new system using a specific design for data extraction and data loading. The design relates old data formats to the new system's formats and requirements. In terms of programming, data migration may involve many phases but at least includes data extraction where data is read from the old system and data loading where data is written to the new system (Pearlman 2019). In Wipro, investment is being made in the development of preconfigured utilities and tools that are instrumental in ensuring that the data is migrated easily and without issues. One of the tools already being used in the current data migration process is the Data Conversion Tool, also known as DCT, which represents an important part of our final solution but cannot perform data migrations to the Cloud. So it's urgent to see how we can use this knowledge about data migration and integrated it with the current available cloud computing services, and for that to be achieved the following key concepts need to be investigated:

- Cloud and Cloud Data Migration;

- Market Trends;

- Integration with Wipro Business Process;

- Website consolidation;

- Data centre relocation.

Finally it's important to check which actions could improve and be integrated with Wipro's current business process when doing Oracle Retail implementations. In short, with this knowledge and investigation, Wipro's clients will be more satisfied and able to have their applications in the Cloud, and Wipro in the other side will be on track ready and prepare to execute cloud data migrations, improving the scalability and reliability of its implementations.

1.3 Problem

The current Wipro process of data migration is based on the extraction and loading of data between on-premises systems with the help of the DCT. But it's not versatile when it comes to being performed into any kind of platform because it's not compatible with the cloud. This does not represent so much of a problem, but it's a huge concern for Wipro due to its low scalability and low efficiency when it comes to take advantage of the available resources. And with the evolution of technology, all the clients request for more speed, scalability and low cost to develop and deploy their projects. With this said, the Cloud starts looking as a more viable and cheaper solution but it requires the design and implementation of a stable process that can perform cloud data migration. So it's urgent for Wipro to find a way of making the current migration process more versatile, integrating the Cloud as part of their final solution. Therefore, it's crucial to exploit a way of solving this problem, and discover how can the current process of data migration be adapted to a better one. It will use the latest technologies to be as versatile as possible, but we need to be aware that as these changes are implemented, several problems may appear and we will have to actively respond to. For now, all data migrations that are being done are from one on premise system to another, and many of the issues can be covered with the main process that is used, but with the change to the Cloud many questions will be raised. For instance, focusing in the Oracle Retail Business Process, there are several applications that need to be deployed into the Cloud and for each one, questions will popup such as:

- For each environment, will it have all the applications and the respective databases?

- As Wipro, will partner Oracle provide an environment for each Client with no costs?

- Will the solution involve a Platform as a Service or a Software as a Service or even an integration of several services?

- Which permissions and restrictions are set?

- Are there any limits when it comes to:

  - System scalability;

  - Amount of data to be migrated.

These are the most important configuration questions that need to be answered, as they are crucial points to achieve in order to perform a data migration to the cloud during an Oracle Retail implementation. These questions represent some of the problems related to the initial deploy of the systems that we are trying to migrate to Cloud. The middleware that is currently being used in Wipro's methodology, the DCT also brings up some restrictions when it comes to establish and perform migrations to the Cloud. There is not a way defined to load data to the Cloud yet, and this is the other side of the future solution that needs a deep look. The following issues need to be solved too:

- Which configurations will be required to integrate DCT with the cloud computing services?

- If a third-party application is used, how to integrate it with DCT and Cloud?

To ease the approach that is going to be taken it's very important to know the differences of each cloud computing service that is available and if there is any part of the current Wipro's solution that can be reused to speed up the configuration of the migration process and its performance. In the Figure 1.2 an overview of the current solution and its configuration is showed in a simple component diagram, that represents a data migration between on-premises using DCT as middleware. Highlighting that this diagram will suffer changes while the new process is being developed and improved due to the possible use of third-party applications or services.
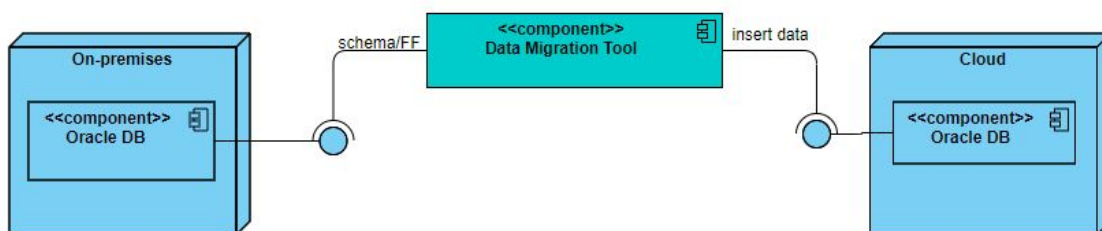
Figure 1.2: Current State of Migration

1.4   Goals

The main goal of this thesis is to develop a system that is capable of migrating data to the cloud and this includes a good preparation for all stages:

- Data Extraction;

- Data Loading (Staging);

- Data Validation;

- Data Loading (Target).

But most importantly, it has to be able to integrate the current Wipro migration process, having the possibility of loading data to the desired system hosted in the cloud or to an on-prem. However it's crucial to take into consideration all the different cloud computing services available, and understand which one stays within Wipro's standard procedures, paying deep attention to data protection and security. On other hand, we will need to get all the possible knowledge about the advantages/disadvantages achieved by each one of these services, in order to establish the best approach. However, part of the research will have to be devoted to study and configure services from Oracle Cloud, because it is one of the key partners of Wipro and already has products aligned with Oracle Retail implementations. The refinement of Wipro's proven Data Conversion Tool will be marked as an extra objective, that if it is achieved, it will allow to streamline the migration process when it comes to data loading and validation, increasing the process overall performance. But for now our main focus is to migrate data to the Cloud which already requires deep investigation and analysis. Considering the points described above, the set of following goals are established for this project:

- Improvement of current data migration process;
    - Compatibility with Cloud;
    - Integration with existing DCT;
    - Integration with third-party application if needed;
    - Performance enhancement;
    - Define the better set of cloud databases provided.
- Enhance of DCT.
    - Increase performance of scripts generation.

1.5   Contribution

From the different objectives outlined above, we can really establish what is the intended and the expected result. All the objectives focus in one point, the ability to migrate data into the Cloud while taking into account tools that are already available in the market and choosing the one that best fulfill our purpose in the OR business. Finally integrate it with DCT, in order to streamline operations such as data loading and validation and connect it to the respective cloud computing service. In this case, we can mostly expect to focus in Oracle Cloud Services that are available, because they are oriented to the solutions Wipro provide, and everyone has deeper knowledge about them, or at least about the Oracle Retail applications/services that have been used till now. As stated before in the Problem, the concept of Cloud is getting bigger and it's taking control of the market, so it's crucial for Wipro as a company to start taking measures in that area. This is why the solution here

documented is a great start to put Wipro well positioned in the retail market when it comes to unifying Cloud Computing and Data Migration. In this moment, all the projects are being based on on-premises systems, but this change and integration of the current data migration process with Cloud will allow to have more stable, cheaper, faster and versatile solutions, which fulfills the current and future market requirements. Furthermore, the new process will increase product quality and reliability, adding value to all the propositions Wipro makes to its customers, putting Cloud and Cloud Data Migration as its new achieved assets. The value analysis of the concerned solution will be detailed in chapters 2 and 4.

1.6   Preconized Approach

Every good solution to a problem, including in the area of Software Engineering, relies on a well structured research process. Usually, two methodologies are used in the research process to define the correct path to obtain knowledge about the issue concerned: behavioral science and design science. In the Figure 1.3 we can have an overall view about the differences and symbiosis.
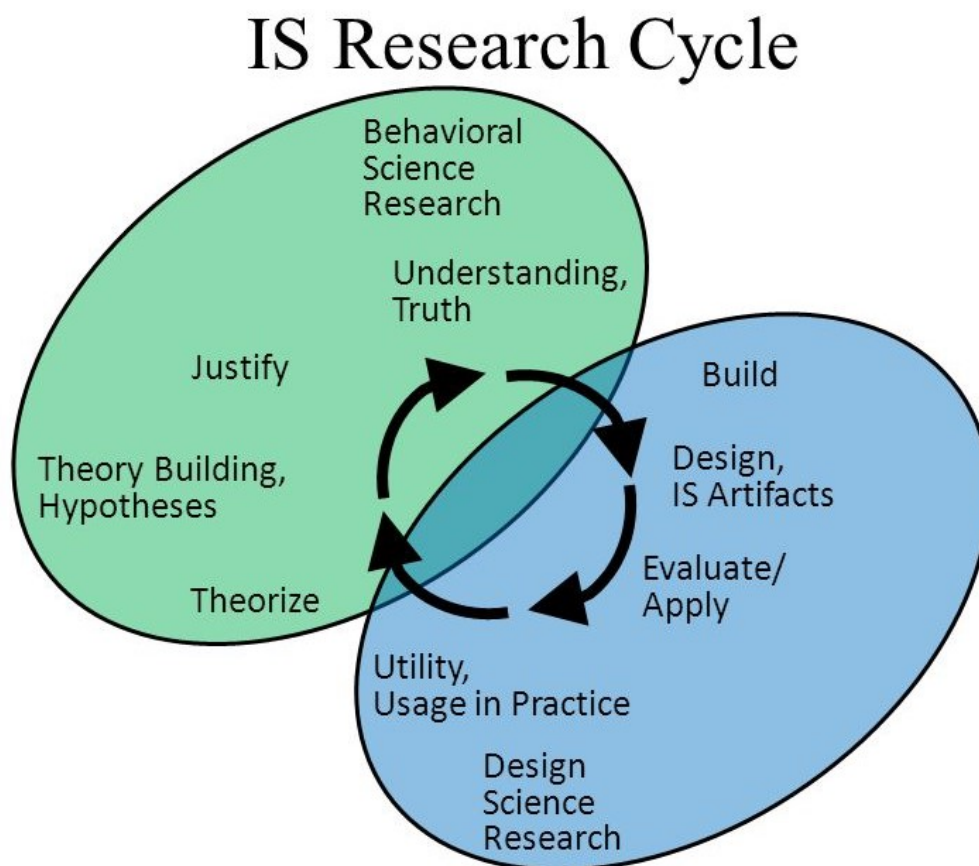
Figure 1.3: Base Research Methods(Jarvis 2014)

The first one is associated to theories creation and tries to justify a concept based on the understanding of the matter, while the second is associated with creation of artefacts and represents the practical way of proving a certain solution with different evaluations or

tests.(Goldkuhl 2016). This dissertation presents a research focused on solving a real problem and concern in the organizational context of Wipro Portugal, so it is possible to conclude that this work follows the design science methodology. Design Science Research Methodology (DSRM) is a well known and accepted framework to implement design science research in Information systems (IS) research. This methodology will be used on the development of this dissertation's work and it's composed by six steps (Peffers and Tuunanen 2006):

1. Problem identification and motivation: "Definition of the problem and justification of the value of a solution. This justification allows to accept the results and to know researcher's understanding of the problem." The problem concerned in this dissertation is presented in this first chapter in section 1.2. Part of its contribution and value analysis is also presented in this chapter and detailed in chapter 2;

2. Define the objectives for the solution: "Definition of the objectives of the solution from the problem concerned. These objectives can be quantitative or qualitative and require knowledge of the state of the problem and other solutions." The goals of this dissertation's solution are presented in section 1.3 of this chapter;

3. Design and development: "Involves the design of the artifact. This step includes the identification of functionality and architecture, then it is developed the designed artifact." The design of the solution is discussed in chapter 5 of this document. Its implementation is described in chapter 6.

4. Demonstration: "Prove that the artifact can solve the problem or parts of it. This demonstration is usually accomplished through experimentation, simulation, case study or similar activities, that in this project were performed at Wipro." In the solution presented in this document, its value analysis can provide some clarifications on how it can solve the problem concerned. Furthermore, the next point (evaluation) is important to give more strength to that value analysis;

5. Evaluation: "Analyze the efficiency of the solution to solve the problem. The objectives are compared to the observed results in the demonstration activity. After evaluation, the researcher can choose to go back to a previous activity, if the results didn't match the stated objectives, or to advance to the next activity." The evaluation of the this document's solution is presented in chapter 7;

6. Communication: "Communicate to other researchers and audiences the problem, the artifact, its design and its results demonstrated." This phase will be composed by the presentation of the work developed to an evaluation committee.

1.7 Document Structure

This document structure starts with the introduction chapter, where it is presented to the reader the conceptualization of the subject, the problem that this project will try to solve and the stated goals to achieve. In the following chapter, the context is defined and detailed. Several points are described that represent the current problem, how we are going to address it in order to obtain the best possible solution and possible issues will be raised and answered like:

- The engineering purpose;

- Assumptions/Implications;

- Point of view;

- Possible concepts.

A brief description of this dissertation contribution is presented in order to conclude this chapter. In chapter 3, it's presented the State of the Art, where a detailed view of all possible solutions is acquired. Theoretical concepts are detailed and then their correspondent technological background is analyzed. In the same chapter, methods and approaches to execute a data migration are described, and the value analysis of the current project is studied and detailed. All kinds of data migrations are discussed, into the Cloud or not, because it's important to distinguish their differences and how the "raw" process of it is reproduced. Chapter 4 shows an analysis of the solution, contextualizing it with the current state of Data Migration at Wipro Portugal and then makes a comparison and evaluation with the methods and approaches addressed in the previous chapter. In chapter 5, the design of the documented solution is presented. It describes the approach and methodology that was taken pictured with high-level diagrams to have a better understanding of the process flow. The next chapter, chapter 6, contains all the information about the implementation process, presenting all the strategies used to develop the necessary tests, all the steps to perform a cloud data migration, and finally explaining the overall workflow of this new process. In chapter 7, it is presented the factors that will be used to evaluate this solution, including hypothesis and statistical tests and evaluation methodologies. This chapter's goal is to evaluate results of the solution and write some conclusions for each factor analyzed. The last chapter, 8, presents the conclusions of all the work developed in this project, and how the research and development of the solution progressed over time.

Chapter 2

Business and Problem Analysis

In the chapter of Business and Problem Analysis is presented an overview about the market opportunities, how this solution would enhance the customer relationship, checking every point of view in order to understand correctly which factors potentiate the development of this solution. To be able to make the best approach when developing the solution, every assumption and implication is defined in order to always pay attention in specific points that are detailed below. Finally, it includes a section dedicated to Value Analysis, which includes a comparison with several tools that stand off in the current market, highlighting the ones that can possibly be included in the project final product.

2.1 Purpose

2.1.1 Market Opportunities

The rapid evolution of technology has led to the emergence of new innovative concepts that have caught the attention of countless companies. They feel the need to adapt and change for the better, to be updated with the latest services and technologies, and the Cloud is able of providing and replicating all types of services, that already exist or even new ones. So it's very important to highlight the cloud computing services and show the advantages they bring such as (Allen 2019):

- Lower capital costs;

- Lower IT operating costs;

- Ease of installation and maintenance;

- Optimized IT infrastructure;

- Scalability and maintainability.

The company doesn't need to do any physical hardware investments or maintenance and the personnel training can be discarded, allowing an improved and more efficient management of the resources. And to make the process even more viable, services can be configured to scale resources when needed. Reliability and mobility are key characteristics too, because Cloud can be accessible from anywhere and allows to define fault domains and backups in order to establish the minimum downtime possible for every type of application (*Advantages and Disadvantages Of Cloud Computing* n.d.). But for this transition to happen, it is necessary to outline the correct steps and have the necessary resources for such a sudden and innovative transition. That's where this strategy/solution for data migration to the

Cloud comes in. Finally, this solution will open the range of Wipro to new opportunities in the market, new customers and even improvements for customers already acquired. Wipro will be able to perform data migrations into fully established Cloud systems and current client implementations can be migrated or upgraded into the Cloud easily, saving resources and providing scalability.

2.1.2 Customer Relationship

Through the use of this product, the biggest "customer" would be the company itself, Wipro, which could provide data migration to the Cloud. From then on customers or partners of Wipro would be interested in putting their data and respective applications in the Cloud, in order to be able to take advantage of the advantages listed in the previous sub-section. But it will have as potential customers all companies that are related to the retail business or that have already acquired Oracle Retail implementations.

2.1.3 Adaptation of existing solutions

The adaptation of an existing solution, that is in the market, it's highly possible because there are already some applications/services that are able to do data migrations to the Cloud. In the case of Wipro that has Oracle as partner, there are already three tools developed by them: ORMFCS, Data Transfer and Golden Gate that will be detailed and presented in 3. In the most real scenario that is going to be designed and built, the new process needs to integrate DCT with a third-party tool. For now, DCT has been used in data migrations in Wipro and it already has a great performance while migrating between on-premises systems. With this said, for the realization of this project it will be necessary to consider the use of new technologies or services that divide into:

- SaaS or PaaS provided by the Cloud;

- Third-party tools that already exist in the market;

However, most of them can be discarded due to certain conditions driven by Oracle such as:

- OR Services only available in Oracle Cloud;

- Most available tools have a low compatibility with Oracle Cloud;

- Deployment of Oracle applications in other Clouds can be complex;

- Low quality technical support.

All these questions will be better described in chapter 3.

2.2 Personal point of view of the problem

The main process of data migration at Wipro doesn't cover data migrations into the Cloud, so the developed solution should be able to perform data migrations from on-premises to systems/applications that are deployed in the Cloud. A special focus has to be considered related to Oracle Retail, because this is the main area that Wipro Portugal is focused in doing implementations and represents its core business. When it comes to this solution, the

time-to-market is very important due to the fact that most companies and potential Wipro customers want to make this move to the Cloud as soon as possible. So, this solution needs to be developed by Wipro as soon as possible in order to cover this gap in future Oracle Retail implementations. But until then, Wipro will have to stick to on-premises systems, which could eventually drive some potential customers away, hence the opportunity, but also a challenge for Wipro.

2.2.1 Personal point of view from the interested parts

The parties interested in the development of this project are:

- Wipro;

- Wipro customers.

Basically, every company related to the Retail Business that is interested in migrating their business process into Cloud will be interested in this solution. Because in the end, Wipro customers will benefit from more cheaper, and depending of their retail workflow, they are able to increase its efficiency and replicate some services without any issues through the use of scalable solutions.

2.2.2 Sellers point of view

The Cloud provider, which in this case will be Oracle, has an interest in the development of this project since it would require the acquisition of its multiple resources that divide into:

- Computing resources (Virtual machines);

- DBaaS or Autonomous databases;

- Object Storage to save backups;

- Oracle Retail SaaS (*Oracle Retail Merchandising Foundation Cloud Service* n.d.):

  – Oracle Retail Merchandising Foundation Cloud Service;

  – Oracle Retail Pricing Cloud Service;

  – Oracle Retail Allocation Cloud Service;

  – Oracle Retail Invoice Matching Cloud Service;

  – Oracle Retail Integration Cloud Service.

- PaaS solutions to integrate applications.

Gaining access to the Oracle Cloud represents a crucial point in the development of this solution because it already offers many services that in one way mirror OR implementations already done by Wipro and provides the database services that are required for this dissertation implementation. And there are two ways of implementing OR solutions in the Cloud:

- Use of OR SaaS (referenced above);

- Customized deployment of OR applications (complex process but with full control);

- Integration through the use PaaS (Golden Gate) or SaaS (Oracle Retail Integration Cloud Service);

- PaaS solutions to integrate applications;

This would allow Oracle to have a constant profitability, and also improves its partnership with Wipro.

2.3   Assumptions

2.3.1   Migration process assumptions e Operational Conditions

For now we are going to focus Oracle Cloud, because Oracle represents one of Wipro's biggest partners, and in Wipro Portugal the work is mainly focused in Oracle Retail (OR) implementations. We are assuming of course that Oracle has some restrictions to their products being deployed in other Cloud then theirs and they already provide Oracle Retail Cloud Services as a replace of on-premises implementations, that's why this choice was made. A little review was done in 3 about other possible cloud computing services because they can be very helpful in the future. Lastly, it is assumed that the database services provided in Oracle Cloud are accessible from any migration tool on the market.

2.3.2   Security Assumptions

Related to the security in terms of migrating data into the Cloud, this point will mainly depend of the cloud computing service being used, having SLA's, legal compliances and access control. This is entirely defined by the company that provides the service. When it comes to the Data Conversion Tool, it's Wipro's property so any data operation is safely executed in a secure environment created by Wipro's technical team.

2.3.3   Maturity level of emerging technologies

There is a high level of maturity when it comes to one of the used tools during a migration that is the Data Conversion Tool. It consists on well-developed technologies with many years of use in the global market. On the other side, the new "parts" that have to be included in the new migration process seem to be quite recent, some of them not very stable which makes the implementation of this solution a little volatile and dependent on constant revisions to be 100% operational.

2.3.4   Assumptions made for an optimal solution

The created solution should be developed according to the parameters of efficient systems development, with a high performance and versatile. These parameters assume that business processes are quick to execute, as well as their configuration and operation are adaptable, with a rollback process. Thus, the criteria that characterize the best solution are the fact that the system is efficient and versatile.

2.3.5   Technical skills assumptions

There may be training for the technicians who create the environments with the DCT application, necessary databases, cloud services and configuration of any third-party applications(migration tools). There may also be training with the Wipro/customers themselves, explaining how the new OR cloud services work, detailing the new important features.

2.4   Implications

2.4.1   Use of Information

The realization of this project will be supported by several types of material. Instruction/information manuals that provide enough information to understand which services are available in the chosen cloud computing service, how they work and their full configuration. There are a few points that need special attention such as:

- Types of data loading methods;

- Accessibility and connection to cloud provided services;

- Security in Cloud;

The different types and more efficient data loading methods are shared and of public knowledge, so this information can be obtained by studying the documentation that is available in the Internet or books. In relation to Cloud security, there are scientific articles that report the details in building and structuring a secure environment, that can change depending from one cloud computing service to another. Accessibility and ways of connection to DBaaS need to be discussed with the providers and requires deep investigation to be sure it's possible to reach them in order to fulfill the process of this solution. It is necessary to study and analyze the architecture of these recent SaaS, just to make sure that the final solution stays stable when it comes to the use and migration of data into those services.

2.4.2   Experiments to be carried out

In this dissertation, in order to evaluate the final solution several experiments will be performed. They will divide into sections such as:

- Test of third-party migration tools;

- Test of the databases hosted in the Cloud.

All the experiences to be made will be shared and detailed in chapter 7. What will influence most the experiences will be the chosen data. The migration process will be done with large data sets in order to have robust tests to check correctly the state of the new solution. These tests will involve generation of scripts, loading/migrating data from on-premise to Cloud, covering every situation and analysing the differences.

2.4.3   Studies, previous solutions and problems

It is necessary to carry out a study of previous / related solutions in order to identify possible problems, some are described in 3 and analysed in 4. This study will allow to design and implement a versatile data migration process, as well as to define a possible configuration that allows reducing the time expended in this complex operation. Other key points that need attention could be having a backup of data in case of any migration problems (application down/slow, databases down). Finally, one of the main problems happens with the connections to DBaaS, which details are available for us, users, and if there are any other restrictions when it comes to accessibility as said in previous sub-section.

2.5   Possible solutions reflection

2.5.1   Possible improvements

This process / solution is composed by several stages and respective applications, depending of the state of each one it's possible to be in a continuous improvement when it comes to enhance the data migration process. It should be noted that this happens regarding products that are Wipro's property and can be upgraded, in any other case a request can be made to the third-party application proprietary to create an improved version for us, which will bring increased costs. Increasing the automation of the process would be an important point to take into account, since several points are manual and related to configurations, this would speed up a lot the current process.

2.5.2   Implications in case of product failure

Product failure can only be distinguished as failure of software. The software failure can imply a software failure regarding DCT, as it is constantly evolving and there may exist some points that are not fully developed, leading to some failures in our process. In the event of a software failure, it is necessary to identify the problem and use preventive measures to ensure data consistency. These measures may represent the use of data backups, so that we can normalize the system's operation as quickly as possible without loss of data. Regarding to software failures provoked by the cloud computing services and their applications, their management is usually carried out by the company that manages the service.

2.6   Value Analysis

Value Analysis is an organized process of analysis and evaluation of a product or service. Its goal is to minimize the cost of a product/service, increasing at the same time its value without losing quality. It uses a specified group of techniques and methods to identify unnecessary costs that don't have a positive influence in the product and does not improve its quality, efficiency, appearance, use or costumer satisfaction (CustomerThink 2016).

2.6.1 Value concepts

Value

The word value comes from the Latin "Valere", that means to be worth something. Value is defined as the worth, usefulness or importance of someone or something. Can be represented in a physical or psychological context, so a person or thing can be considered important or beneficial in some way, in other words, valued(Brainly.com n.d.).

Value for the customer

A simple way to understand what does value means for the customer is quoted: "Customer Value is the perception of what a product or service is worth to a Customer versus the possible alternatives. Worth means whether the Customer feels he got benefits and services over what he paid."(CustomerThink 2016). In simple words, value for the customer is the balance between benefits and costs(sacrifices). Concepts as quality, customization, responsiveness, reliability, image and trust are seen as benefit drivers, while price, time/effort/energy and conflict are seen as sacrifice drivers. In a moral way, values are what someone or a firm stands for: Honesty, ethics, sustainability, integrity, trust and many others. All these factors can influence the customer's decision. When these benefits and sacrifices are well combined or there is an considerable amount of benefits, and the costumer precepts advantage on an organization's offer, there is Value for the customer (VC). Another way of creating Customer Value is evaluating and knowing the customer's satisfaction/experience. This can be positive, the customer is happy with the service which increases the customer value, or it can be the other way around, decreasing the existing customer value or even denigrate the name of a brand. If the service/product is good and the customers are happy and satisfied with it, the company or product is backed up with loyalty, market share and price, which enforce them to reduce any pending errors, and to increase efficiency. Higher market share and better efficiency leads to higher profits(CustomerThink 2016).
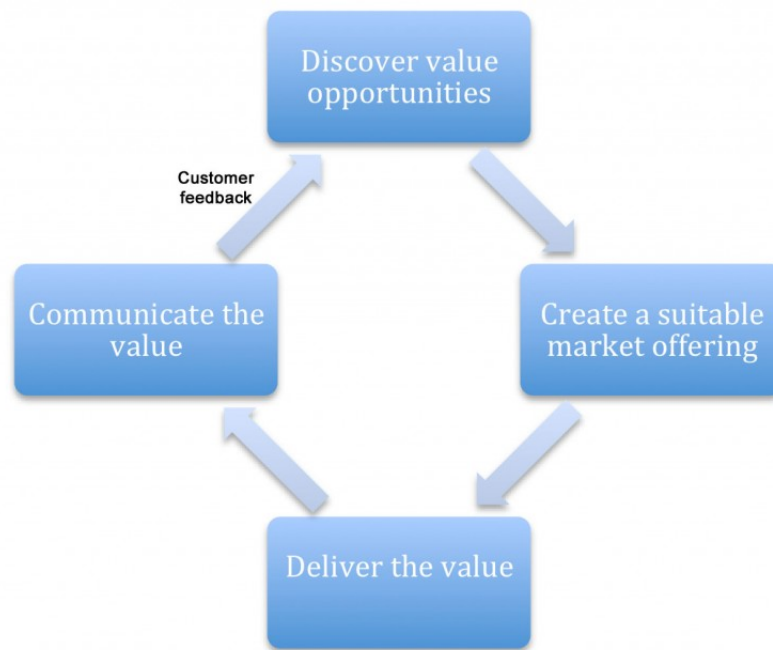
Figure 2.1: Value Cycle (Mansfield 2018)

Perceived Value

In this case, we need to understand correctly the Customer Value concept, what a Customer perceives as value, how does this value changes over time, and the influence of the Customer feedback. Everyone that buys a product or service can increase or decrease value over other competing options. To understand well the concept of perceived value there are many questions that helps to recognize what a Customer sees and makes valuable, taking in attention all competing brands/companies(CustomerThink 2016). "What is important to the Customer in his buying decision? Is price more important or are benefits? Are you good at delivering what the Customer believes is important? Are you able to deliver more than your competition on these factors?", answering these questions will allow to perceive correctly what does value means for the customer in each possible case. Basically, all the benefits and sacrifices of a product/service will influence the value perceived by the costumer. Different costumers have different perceived values of the same products or services. So, the perceived value is the general evaluation that a costumer gives to a product/service utility, based on perceptions of what he receives and what it gives.

2.6.2   Benefits and Sacrifices

For the client, Wipro and its customers,there are some benefits and sacrifices that our product/solution gives and demands. For this, let's take a look into the article "Conceptualising 'Value for the Customer': An Attributional, Structural and Dispositional Analysis" of Tony Woodall, that shows a longitudinal perspective on VC and how it changes over time. In the Figure 2.2 can be seen the four distinct temporal positions for VC (Woodall 2003).
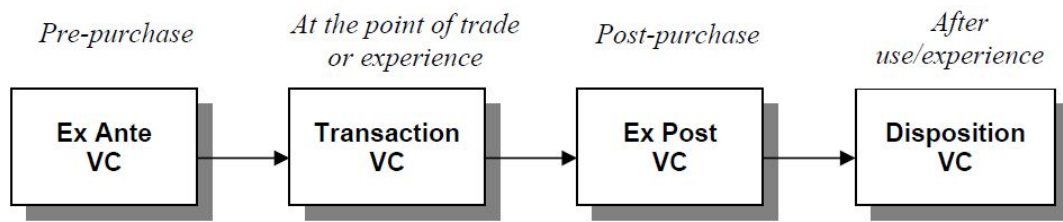
Figure 2.2: Longitudinal perspective on VC(Woodall 2003)

The solution described in this dissertation represents the improvement of Wipro's data migration process, including Cloud as a possible option. The following table 2.1 presents the benefits and sacrifices from pre-purchase phase till after use/experience phase for costumers, in this specific case to Wipro.

Table 2.1: Value Analysis - Benefits vs Sacrifices

Phase	Benefits	Sacrifices
Ex Ante VC	• Operational Benefits; • Product Benefits.	• Human Energy; • Time; • Efforts.
Transaction VC	• Trust.	• Monetary Cost; • Time.
Ex Post VC	• Responsiveness; • Technical Competence.	• Conflict; • Time/Energy/Effort (meetings).
Disposition VC	• Service Quality; • Time Saving; • Product Quality (improvement of Wipro process).	

2.6.3 Value Proposition

Value Proposition (VP) is crucial and essential to alert new costumers of the value of an organization's products and services. As Osterwalder said when building a business model, "A Value Proposition is an overall view of a company's bundle of products and services that are of value to the customer.", and it's connected to other important pillars such as: customer interface (target customer, distribution channels and relationship), infrastructure management and financial aspects(cost, revenue)(Osterwalder 2004). The value proposition must be clear and incisive about the product or service that's being provided, its target costumers, all the inherent value provided and its unique value. The service documented in

this dissertation has as its target the organization Wipro and maybe some of the current/future clients. This process/solution enriches the organization's current data migration process, including the implementation of some improvements in an already used tool the DCT. Allows the organization and the projects where it is implemented to deliver products to its costumers with more quality and faster, because the process is more versatile and can adapt to different options. The planning and approach here documented does not exist, so it's going to be unique in the organization and it is expected to have good feedback and to be part of future Oracle Retail implementations. In the following Figure 2.3 we can take a look into the design of a Business Model and check the different pillars around the VC concept. The combination of all these well-filled points contributes to a good business model, which can be portrayed in a Business Model Canvas, that is easier to understand and very interactive(Alexander Osterwalder 2015).
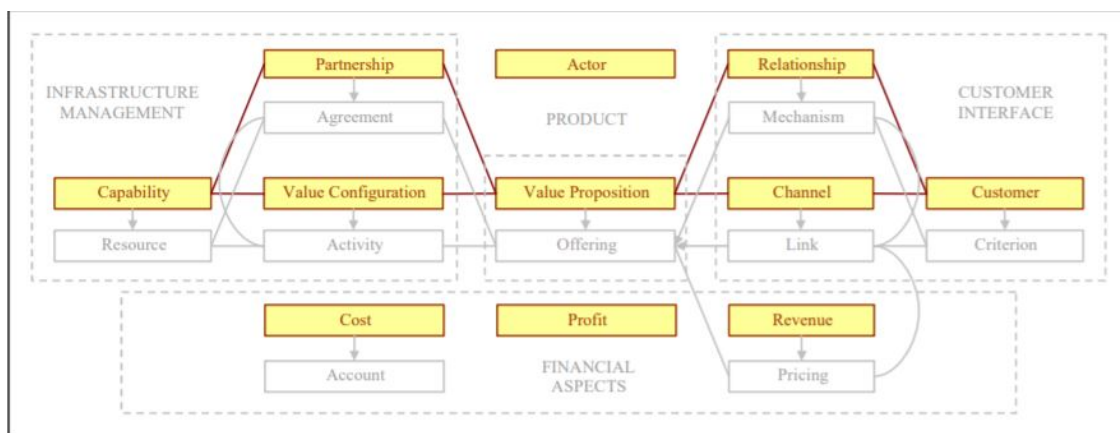
Figure 2.3: Business Model Ontology(Osterwalder 2004)

2.6.4   Business Model Canvas

The business model can be analysed and checked in Figure 2.4. This model addresses several concepts like:

- Key Partners, Resources and Activities;

- Value Proposition;

- Cost Structure and Revenue streams;

- Customer Segments, Relationships and Channels.
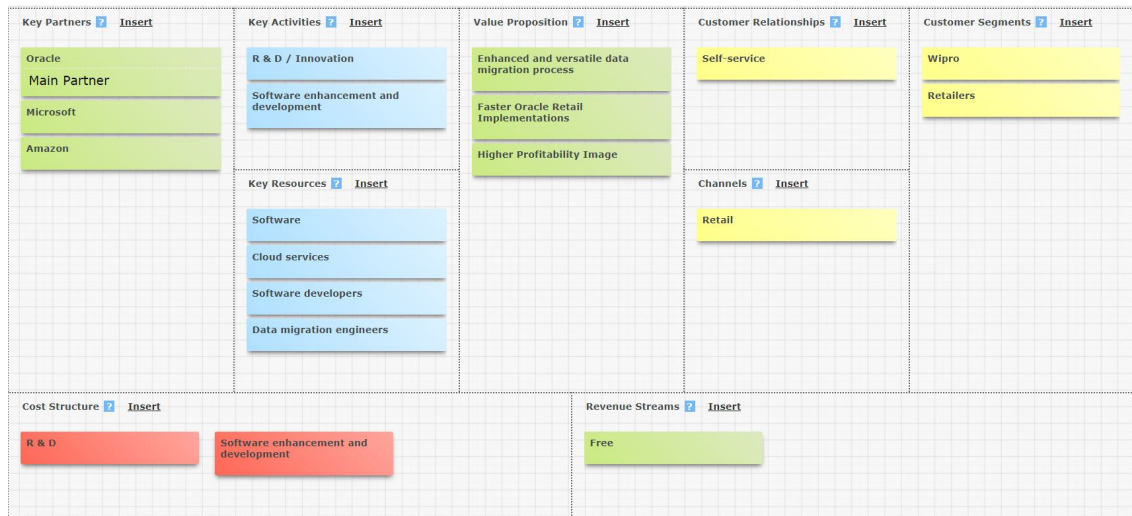
Figure 2.4: Business Model Canvas

2.6.5 Value Analysis Approaches

In general, a problem involving the need to make a specific decision that involves criteria and different degrees of importance for the decision maker requires the use of methods multi-criteria. A business value analysis depends of the value for the customer, customer's perception, existing processes, or even other options with a subjective character (Mocenni n.d.). For this case, we will be presenting two different methods:

- AHP Methodology;
- FAST Design.

The first one, AHP methodology, will be describing and comparing the different available tools/services available for performing cloud data migrations according well defined criteria/-parameters. Finally it's shown the FAST approach to understand the full picture and process of our solution when it comes to its different stages and necessary tasks to be performed at each one.

AHP Method

The AHP method will allow to check the relationship between the defined criteria, the respective sub-criteria and the possible solutions that best suit those criteria. We can easily choose the software/tool that needs to be part of Wipro's new migration process. In the following Figure 2.5 it's easy to understand the base of an AHP Process and it suits perfectly in this particular case.
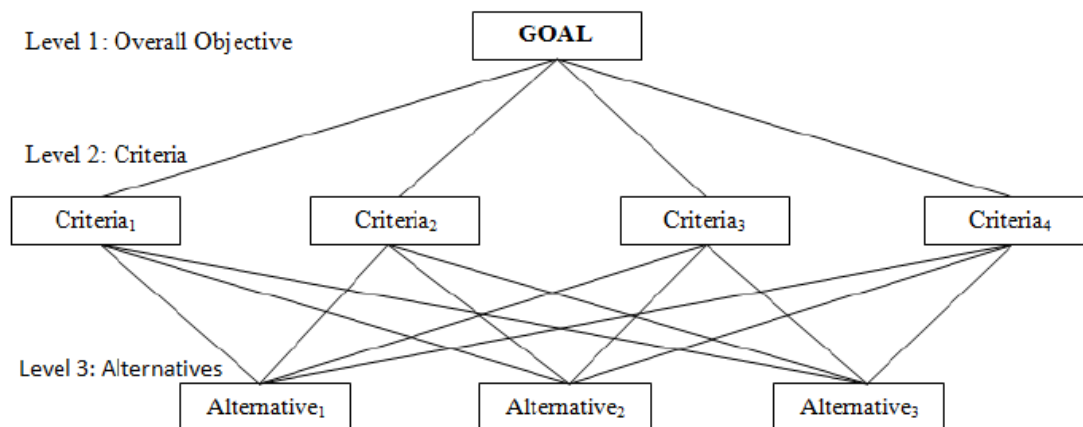
Figure 2.5: AHP Process(Prince Agarwal 2014)

Regarding data migration, the number of criteria varies depending on the objectives to be achieved and the resources available. In this case, the criteria number can be divided into cloud compatibility, database compatibility(sources/targets), cost and operations availability. For each one of these points we need to define correctly the relevance/effort. In the following displayed pictures, we are going to divide our analysis into two areas:

- Oracle's technologies;

- Other Enterprise Solutions.

This division will allow to easily compare tools from different owners and see which ones better adapt to our process taking into consideration all the points described in the 1 and 2. For the first AHP diagram, a comparison is done using 3 major Oracle tools recognized in the market:

- ORMFCS;

- GoldenGate;

- SQL Dev.

And we can easily choose GoldenGate as the most effective one having several options available such as schema replication, import of data from files, accepts several SaaS as sources and has a large compatibility when it comes to database and cloud concept. Noticing that it follows an ETL approach which is the one specified in our process alongside DCT.
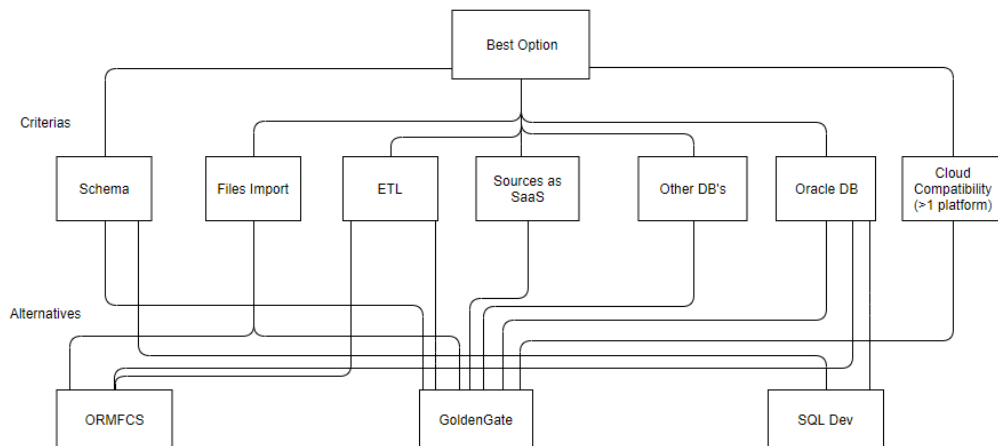
Figure 2.6: AHP Comparison with Oracle Technologies

In this case, we are comparing three tools from tech giants such as:

- AWS DMS;

- Qlik Replicate;

- Stitch Data Loader.

After doing a detailed comparison we are restrained to two tools, Qlik Replicate and Stitch Data Loader, but due to licence restrictions and undisclosed cost, we will stick to Qlik Replicate if needed when projecting our solution design in 5.
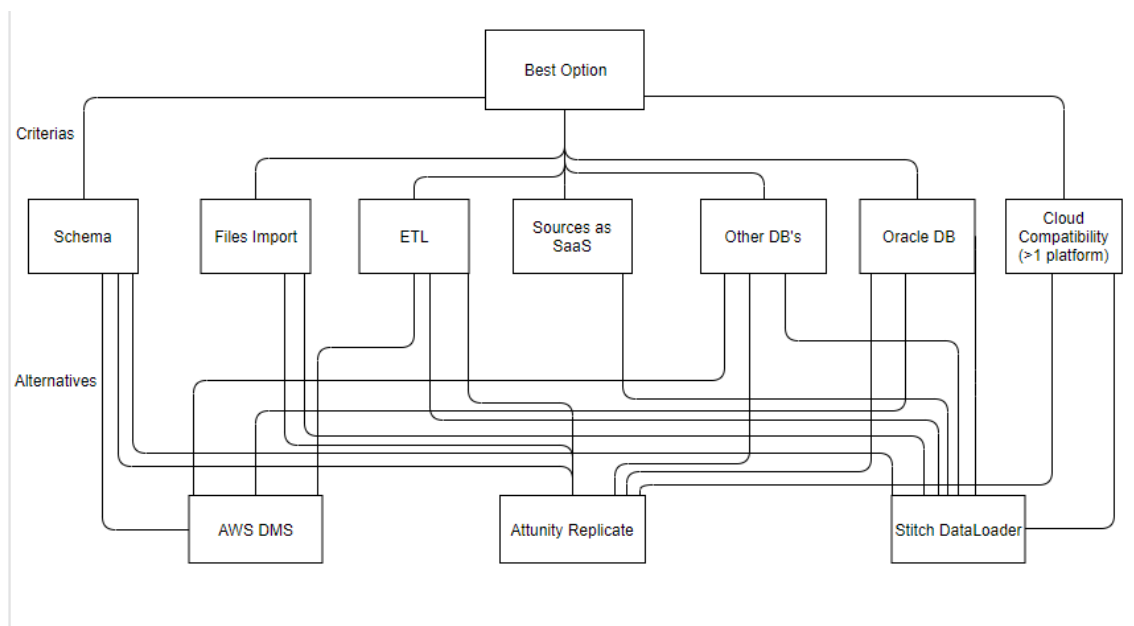
Figure 2.7: AHP Comparison with Other Technologies

In short, Oracle's GoldenGate and Microsoft Qlik Replicate are the main tools with the best attributes when it comes to data migration and integration with Wipro's current business process. But due to Oracle's Cloud restrictions, only Oracle's Golden Gate technological approach can be used, it seems to be the best to help in the adaptation of Wipro's process to perform Cloud data migrations, while respecting all the different criteria displayed above. But in a short future Qlik Replicate is a great tool to keep in mind if Wipro intends to expand it's cloud compatibility.

FAST

This representation shows the functions that need to be performed by the solution/service following a How?/Why? relationship, always in a logic and integrated way. It verifies if the designed solution achieves the needs of the company, and if necessary identifies duplicated or missing functions in its process(Value Analysis n.d.).
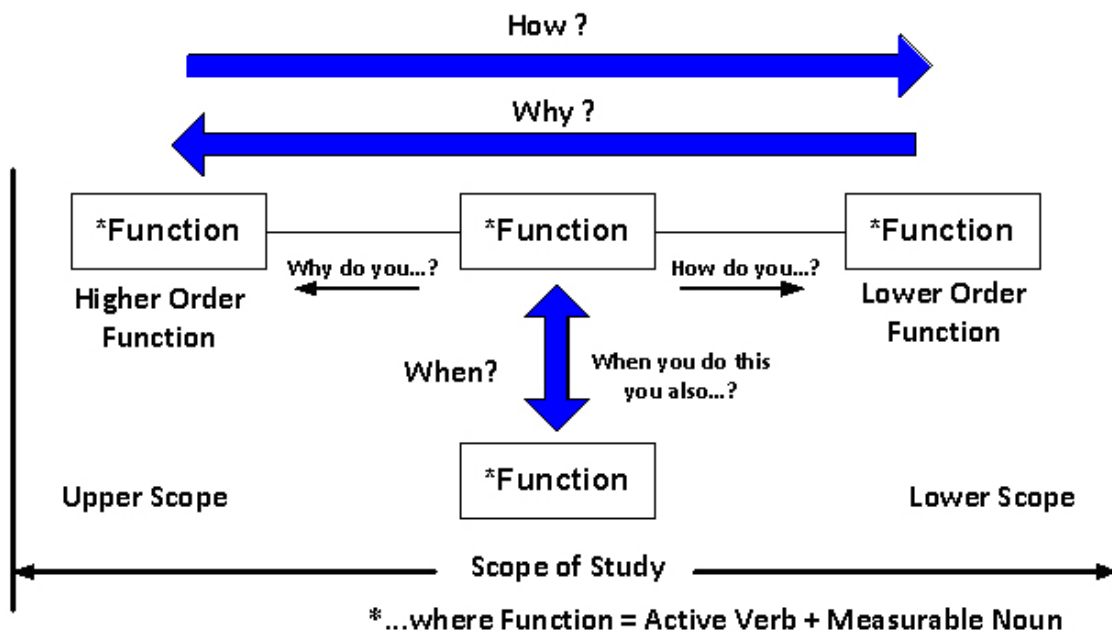
Figure 2.8: FAST Diagram Structure(Value Analysis n.d.)

In Figure 2.9 it is described the different functions that belong to this solution and make it work. Provides an overall view of the service in analysis, being able to track the critical path. The critical path represents the more important functions of the process. Besides this, also shows supporting functions that even though they are not in the critical path, their bad performance can harm the normal course of all operations, influencing the full data migration process. For almost every operation of this complex process there are tools that make their implementation easier. The solution process documented in this thesis doesn't exist in the current organization. It is difficult to compare the overall process with others implemented by another organization or developer because it is composed by several steps that change depending of the approach or business of each one. But it's possible to discuss and compare the different tools that can be used in each function of the process to achieve the best possible results. This information is presented in the next chapter 4. The tools and methods chosen to resolve each step of the process will influence the value of the solution,

so their analysis is very important to choose the most appropriated one depending of the business process and discussions made internally.
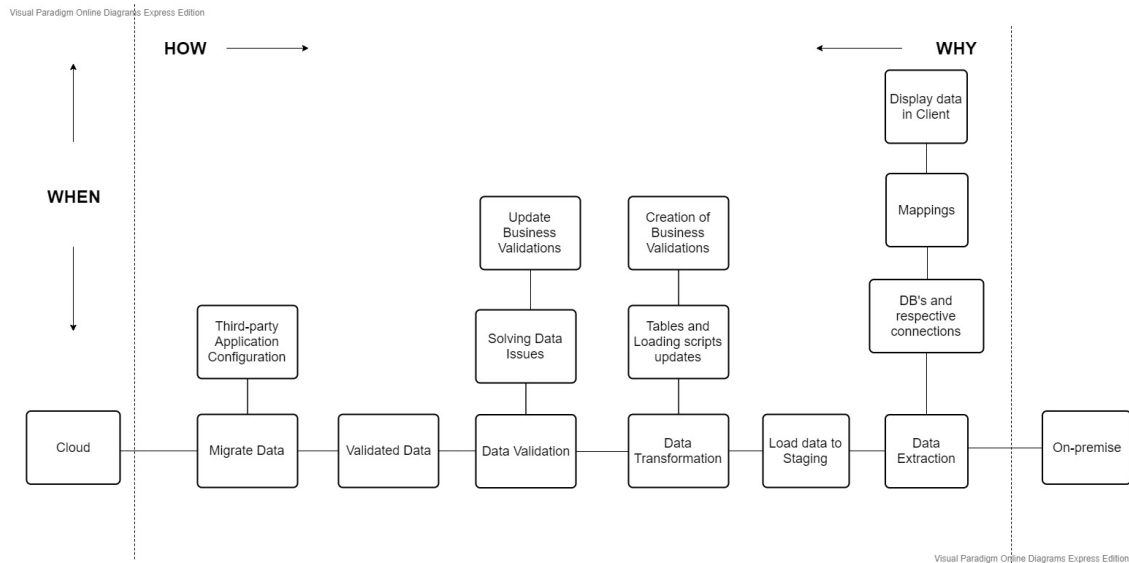
Figure 2.9: FAST Migration Diagram

It's possible to compare this solution with the current state of data migration process, having in mind the benefits and sacrifices in 2.1. In the current state of the migration process, the Data Conversion Tool is responsible for preparing the staging environment with all the data being already transformed and validated, but this stage can be improved when it comes to performance and efficiency parameters, and this is going to be one the main points of this solution. On the other hand, the process only allows migration between on-premises, which turns out to be a huge constraint regarding the current market due to the technology evolution. The solution documented will be more versatile, allowing migrations to any type of platform and with an improved DCT all the process will be faster and more efficient. Besides the sacrifices to the organization presented above, as human energy, some monetary cost, the overall benefits are more valuable and the final result will make up for them. The solution will increase the value/power of Wipro in the current market of Oracle Retail, having a huge availability when it comes to offer solutions on-premise or in the Cloud, gaining even more trust from its customers.

Chapter 3

State of the Art

In the chapter of the State of the Art is presented an overview of the project, are discussed important concepts and technologies related to the problem in the Theoretical Background and Technological Background, respectively. Also, it presents the known methods and approaches to perform a data migration and how can be adapted in order to address the main problem of this dissertation, including possible risks and challenges when implementing the solution.

3.1 Theoretical Background

Data migration is a crucial process for all existing companies today. In short, is the process of moving data from one system to another. It can be done in different ways, changing from company to company, and of course the resources that each one has available are a huge factor. But it has to be present, it's known that any person / application needs to have their data saved. So this process can be very useful in different situations such as (Taylor 2019):

- Migrating storage;

- Migrating databases;

- Migrating applications;

- **Migrating to the Cloud**.

Storage migration refers to a storage technology refresh, so all the data has to pass from a certain unit of storage to another, it's more of a hardware problem and with the data growth it's necessary to have more and bigger storage units. Database migration it's usually related to customer requests, that involves changes when it comes to the current system database, to have another kind of database, for example from Oracle to MySQL. Result of changes in the business process or need for evolution in the system, obtaining higher performance. The third point, applications migration, happens when it's needed to move from one application to another, because of a vendor or platform change, and data migration may be required to be performed depending of the compatibility of the new platform, in order to have the most recent data. Finally, there is Cloud data migration, that needs to have a deeper look, as it will be the situation highlighted throughout all this document, which is moving data from on-premises to the Cloud, never discarding that it is also possible to move data from one cloud to another(Taylor 2019). This point can refer to every situation described above which includes: storage, database and application migration. To have a better insight of

the problem that it's being addressed, and why this solution has to be developed, let's take a look to the evolution of the Cloud and what has changed in the Deployment across the years. In the Figure 3.1 we can check the percentages of Deployment Type being used along the years, comparing Cloud to Traditional(On-premises).

Worldwide Cloud IT Infrastructure Market Forecast by Deployment Type 2015-2020 (share based on Value)
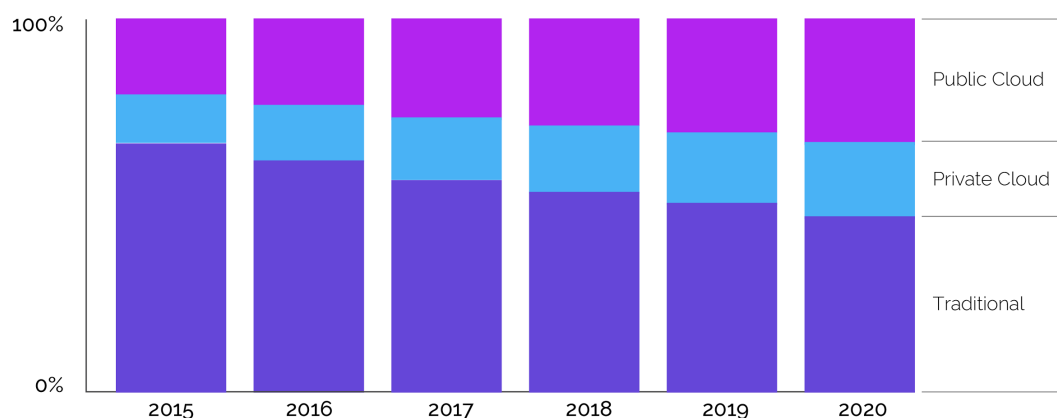
Figure 3.1: Prediction of Cloud Market(Amar 2018)

It is easy to see from the chart provided, that the Cloud has been standing out in the IT market, and does not tend to slow down a bit. This is due to several reasons, that can be divided by:

- Cost efficiency;

- Performance;

- Security;

- Scalability.

Maintaining on-premises computing requires many conditions, which have very high costs, like: space, electricity, hardware, networking operating systems, physical security and others. All these expenses can be shifted to a Cloud service in order to save money. Related to the performance, Cloud services offer their customers, the best equipment available in the market, processors, RAM memory, GPU's and even cloud programmable hardware. Not forgetting about managed databases, warehouses and orchestration platforms, everything that can ease the life of an IT professional. When working on an on-premises system we are stuck to the technology since the beginning and every change will allocate many resources, that sometimes cannot be spent. Last but not least, the security concept, it's true that it is hard to be more secure and in control than housing your data center yourself. Nonetheless security on-premise is only as reliable as the security policies put in place and enforced by the person or company in charge. On the other hand every Cloud service is required to meet the highest security standards in the industry that are set by several institutions. So, this point pretty much depends of the security established in the on-premises infrastructure,

because we cannot have 100% control of the cloud environments(Amar 2018). Depending on the Cloud service that is chosen, many options are offered, including services with low-high prices that adapt to every kind of customer, taking into account every possible need. In short, some major differences between both are represented in the following Figure 3.2.
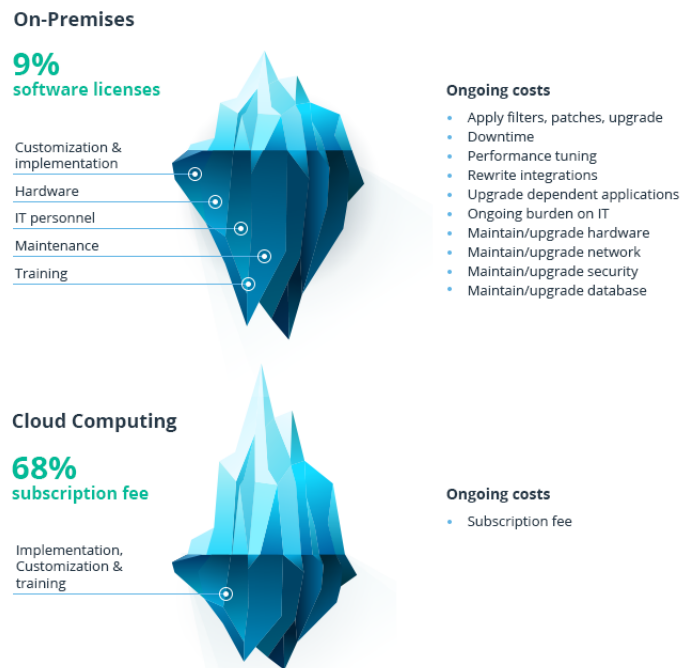
Figure 3.2: Cloud Computing vs On-Premises (Oleksiuk 2019)

3.2 Methods and Approaches

This section starts with an overview from the concept of data migration, taking into account all the details from each different stage that composes it. After this, the available strategy types and some good practices are described, ending with possible challenges/risks that need to be considered. Next, the main process of implementing a data migration into Cloud, using different Cloud computing services will be discussed to understand the best way to design and construct the solution.

3.2.1 Data Migration Process and Stages

A good data migration strategy must always be defined, in order to improve its performance and competitiveness. If a strategy is not well defined, it is more likely to generate more problems than solving them, that can result in missing deadlines and exceeding budgets for the project in question. To avoid these problems, some important steps need to be followed:

- Knowledge about the data;

- Data cleanup/cleansing;

- Data maintenance/protection;

- Governance.

To start a data migration, it's important to **know the data** we are working with to avoid unexpected issues, doing a full data audit would be a great start. Second step, **data cleanup**, it's performed when any kind of issues is reported, it should be done based on discussions with the client, this is mainly addressed to problems with missing data or the opposite and may require third-party resources. Finally, the two last steps are more related with data management, focusing in maintaining its quality: **maintenance/protection** and **governance**. As time goes by, data undergoes some degradation, and it becomes unreliable, so there is the need for control in place to maintain data quality, discarding any "garbage". Throughout this migration process, it's highly recommended to track and report on data quality, to be always updated with the latest state and integrity of the data(Pearlman 2019). Data migration, in the case of Wipro, requires always an important stage known as **Data Conversion**. Data conversion is the transformation of data from a legacy application to an updated or new application. This phase it's included in the **ETL** process, also known as: extract, transform and load. To be a little more explicit, during extraction are used scripts to get the important information from the model of the legacy application, and then transformation is crucial to do a correct mapping of all the information to the new data model of the application to be deployed, and if any third-party resource is necessary and how to adapt the process. Finally we proceed to the load of the information using SQL Loader, that speeds up the loading into an Oracle DB (Taylor 2019). The full context of data migration includes other processes that can be seen in the Figure 3.3.
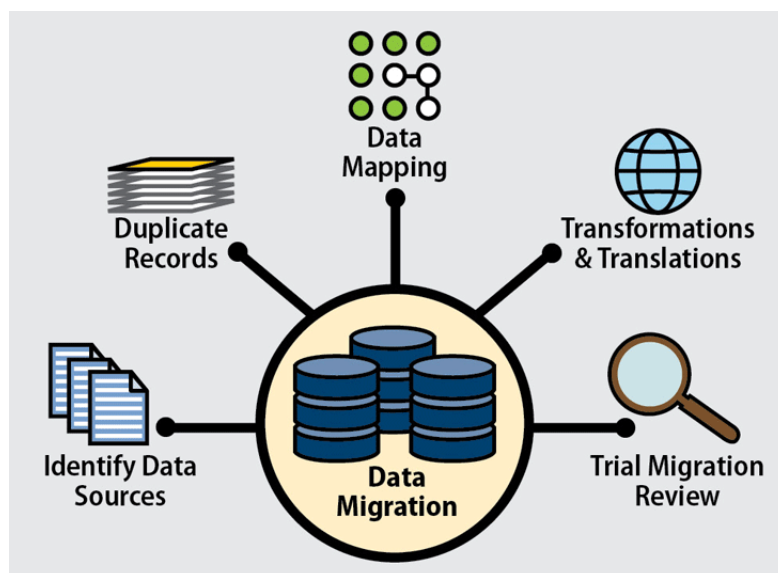
Figure 3.3: Data Migration Processes (Taylor 2019)

3.2.2   Data Migration Strategies

There are two types of migration strategies:

- "Big Bang" Migration;

- "Trickle" Migration.

In "**Big Bang**" migration, it's executed and completed a full transfer within a limited window of time. That results in live systems to experience some downtime while the data goes through the ETL processing, explained above, and transitions to the new application database. On the other hand, we can opt for a "**Trickle**" migration, that allows to perform a migration process through several phases. The old system and the new run in parallel, which eliminates any downtime or operational interruptions, and data is continuously migrating (Pearlman 2019).

Best Practices

Regardless the type of migration strategy that is chosen, there are some good practices that we should always follow. Before a migration should always be done a **backup** of data, because if there is a problem, data can be lost which is one of the worst possible scenarios. During the migration, we should stick to the defined strategy even when it's not going as expected. The process can be complicated but it's worst to keep changing the plan without trying for real till the end. Lastly, never forget to do several tests, many rehearsals to be sure that the desired result is eventually achieved without any issues, this can be done with the use of test environments (Pearlman 2019).

3.2.3 Set of viable solutions

For this case, several solutions can be designed because there are many tools that are capable of doing a data migration, to the Cloud or not, but we need to take a deeper look to Wipro's situation. Wipro already has a data migration tool, DCT, but it cannot reach the Cloud, so an integration has to be developed with some other tool. These data migrations need to be highlighted as they are related to Oracle Retail implementations, which have several restrictions, and may lead to subsequent integrations. So the most viable solutions would be done considering as integration tools, products of Oracle that are capable of performing some adjustments for the Cloud. Depending of the steps used during a data migration, other applications can be used instead of only Oracle's.

3.2.4 Challenges and Risks

A data migration is always a complicated and risky process. It's time consuming as it requires a lot of planning and many implementation steps. Just to raise awareness, the following points represent some of the possible issues or challenges we can face during a data migration (Taylor 2019):

- Data loss;
- Compatibility issues;
  - Operating systems;
  - Databases;
  - Cloud Services;
  - Complex user access to source and target systems.

- Poor execution impacts the business;

With this said, not all solutions represented in section 3.3 are appropriate due to several risks as mentioned above. The following list presents the reasons that allow to reject some of the available solutions as a viable one:

- Use of cloud data migration tools that aren't compatible with Oracle applications/ Oracle Cloud;

- Use of any other cloud computing service than Oracle Cloud.

Related to the first point, the main data migration tool at Wipro is the Data Conversion Tool and this component will not be changed, as it covers a huge part of the process already and several people as experience on it. The data migration tools that can be used, from Oracle or not, is still a point that cannot be completely closed. There are several details that need to be analyzed and will be reviewed in the later in this chapter and in chapter 4, having as crucial point the compatibility with Oracle Cloud and its services. Finally, when it comes to cloud computing services, we will only be able to opt for the Oracle Cloud, since it is the one that provides us with the necessary services (SaaS) for Oracle Retail implementations and the database services that are essential for our proof of concept. Denoting that if wanted, other Cloud services can still be used to deploy Oracle applications but it's more relevant to start making use of the latest applications that Oracle has to offer in its Cloud instead of staying attached to older ones.

3.3   Technological Background

In this section, some tools and applications that are responsible for data conversion, or allow the implementation of data migrations into Cloud are presented, not forgetting the possible Cloud computing services that are available. It will be discussed some of the trends on those technologies, highlighting the most popular ones. The tools that are going to be important for this project and that are already in use in the organization will be described in more detail. Small context about possible alternatives to the same will be seen, but note that, for now, won't replace anything.

3.3.1   Data Migration Tools

In the last years, many data migration tools have emerged, including open source tools. In this case, because it's being discussed a solution for migration of data into Cloud, a detailed analysis of migration tools will be done that divides into two distinct groups:

- Wipro's Data Conversion/Quality Tools;

- Available Cloud Migration Tools;

Maybe it's not clear why an investigation is done for simple data migration tools because focus is needed in a Cloud perspective, but this involves a larger than expected process, due to the context in which Wipro finds itself. Wipro already has a data migration tool, as previously mentioned, which is DCT, this tool eases the entire process of data migration between on-premises, but it is necessary to adapt this process for Cloud. So it's necessary to do a research that covers data migration tools that are able to process data into Cloud.

It should be noted that in the following sections, tools such as IQNxt and DCT will be highlighted as they have already been or are used in the current business context. DCT will remain one of the key tools in the process designed as the final solution, and it is presented with more detail in the following chapters.

Migration Tools of Wipro

Nowadays due to high paced IT trends, every company is expanding or trying to expand, and this, requires a huge focus on data migration (Help 2019). In order to respond to this high demand some tools were developed. In this section we will take a closer look to some of them such as: IQNxt, Data Conversion Tool.

IQNxt

IQNxt is a Wipro integrated data quality management and transformation solution that provides a platform for the same operations. It brings some new aspects of governance, continuous quality monitoring and business enablement. The primary and significant areas for IQNxt are data migrations and quality management. It offers many different functionalities inside its process such as:

- Addresses data quality for migration and on-going monitoring;

- Library of pre-built DQ rules by industry specific use cases (FATCA, UDI etc);

- Out of the box integration and transformation templates for select platforms;

- Integrated governance controls(workflows, audit and versioning);.

An overview of IQNxt can be seen in the Figure 3.4 below.
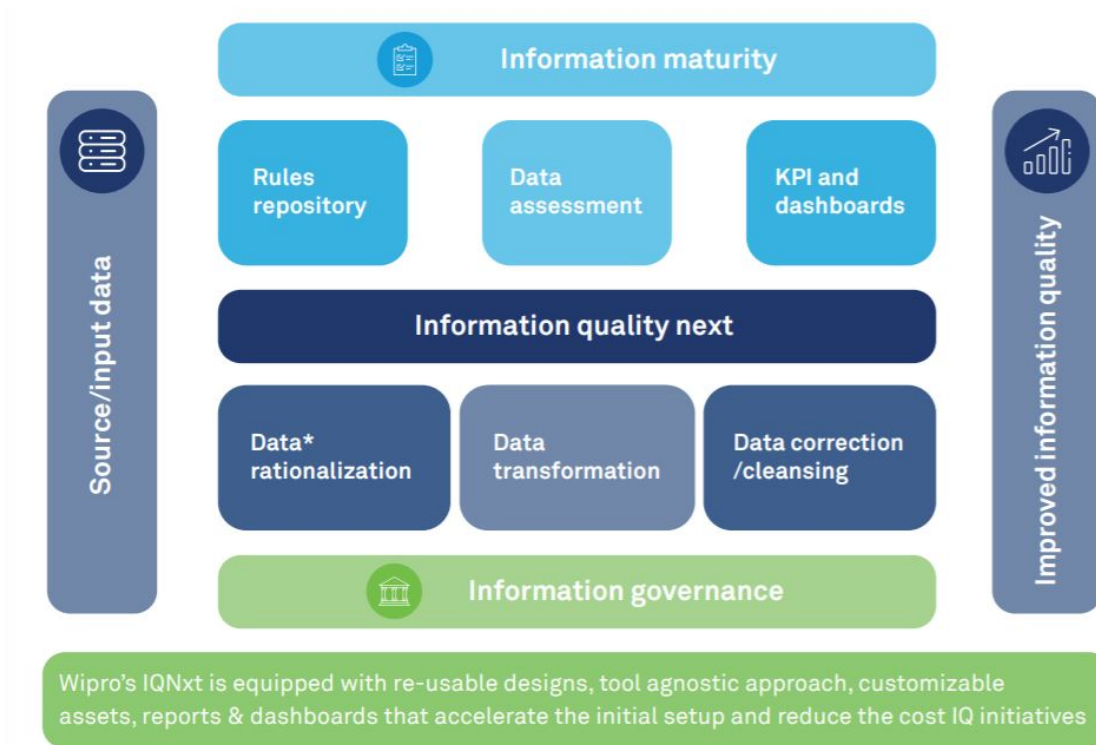
Figure 3.4: IQNxt - Process by blocks (Limited 2018)

In conclusion, this Wipro tool has several differentiating factors, on one hand it is already a patented framework for data migrations. It can convert any source of information, it is reusable and it has a library with specific rules for certain use cases from different industries. It also has centralized control, allowing you to define privileges for different tasks, workflow management and data security. In a business context it brings several positive aspects such as the reduction in the cost of transforming information by 20%. Less risk of failure or problems due to the fact that it is patented and in terms of implementation and prices is very flexible. Overall the feedback is quite positive (Limited 2018).

Data Conversion Tool

The Data Conversion Tool is the solution that is currently being used at Wipro Portugal for Oracle Retail Solutions. As its name says, the tool does the conversion of data, more specifically, allows us to do the necessary transformations to it, including many other points such as:

- Tables creation;

- Keys, Indexes creation;

- Extra validations to the data;

- Generation of loading scripts with multi-threading.

The tool doesn't stop here and has many other features, that only with a small demo would be understood. A new version of DCT is being developed, a web-App that has smoother interface than the current application and it's more configurable. As an addition to our final solution, we can pick it up where it was left and solve part of the problems that it still has.

We are talking about a more recent application, which is not in a stable version so it can bring a few challenges when testing it.

Cloud Migration Tools

These are available options that can be used to perform data migrations to the Cloud and have to be analysed in detail. They are on-premise cloud data migration tools that have the capability of combining all the data sources from within the enterprise and move it to the desired location of a specific cloud service. They offer a number of security restrictions to safeguard processes, systems, and data during migration as it will be seen in the following sections (Alooma 2018).

Oracle Retail Merchandising Foundation Cloud Service

Related to a specific service available from Oracle, the ORMFCS, it is capable of providing end-to-end merchandising operations that delivers the foundation of a certain retailer's operations. Merchandise operations, importing management and sales auditing solutions all together in a single-cloud service. This software has a functionality for data conversion that allows to take two different approaches. They can be divided into: normal procedure and mass uploads(Oracle 2018a).

Standard ORMFCS Migration Process

Migration process starts with the import of files, then it goes through some steps of validation, ending with the migration to the destiny of all the accepted records. In the first stage, data import from a file, we can choose the group or interface, followed by the selection the desired entity, ending with the chosen file name and respective description. This will do a basic validation of the data, comparing the type of fields imported to the columns of the chosen entity. Second stage it's data validation, basically all the records imported to the staging area will get their check constraints validated, preparing them for final stage, business validation. Business validation does the final validations related to the business model, involving primary, unique and foreign keys. If the data is correct, it is migrated to the primary tables on the database in the Cloud. One of the stages of the migration process can be seen below in the Figure 3.5.
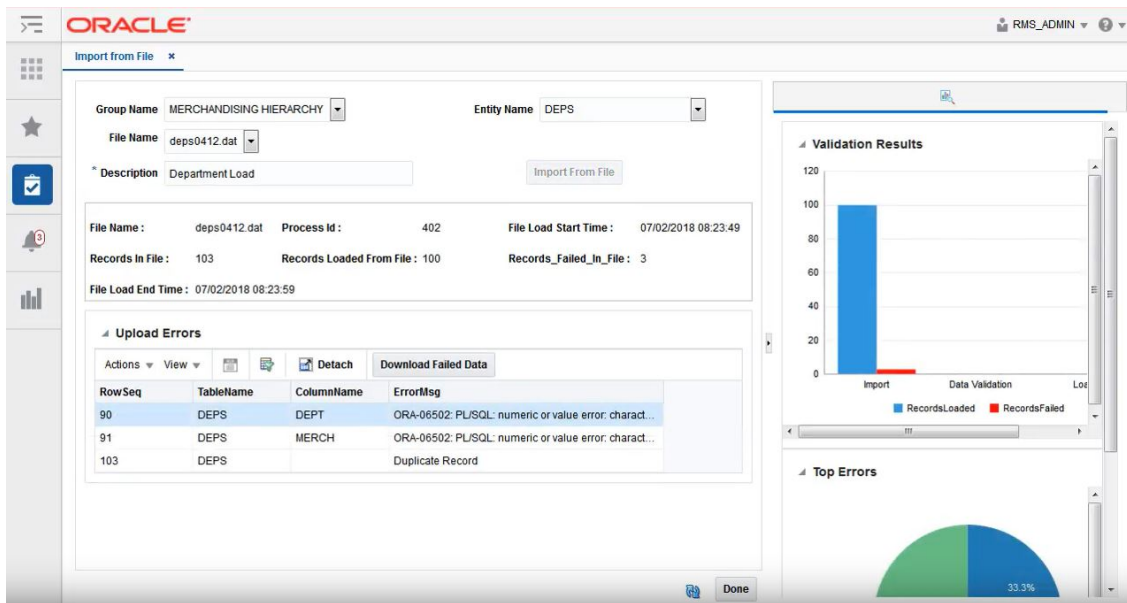
Figure 3.5: ORMFCS - 1st Stage (Oracle 2019b)

Mass Uploads

It's very similar to the process described above, but is able to do as the name says, mass uploads of many entity groups at the same time, and allows to skip the business validation, migrating data directly to primary tables which saves a lot of time during a migration. The other steps such as importing data from files and its validation are done automatically by the system. In the end of this operation we can check all the errors that result from any kind of validation, and the respective files that can be modified to start a fresh migration. This process would be one of the best to integrate with the tool used by Wipro, the Data Conversion Tool, that already does all the needed validations. The possible solution with the use of Mass Uploads will be discussed in the next Chapter 4. A simple screen of mass uploads functionality is exposed in the Figure 3.6.
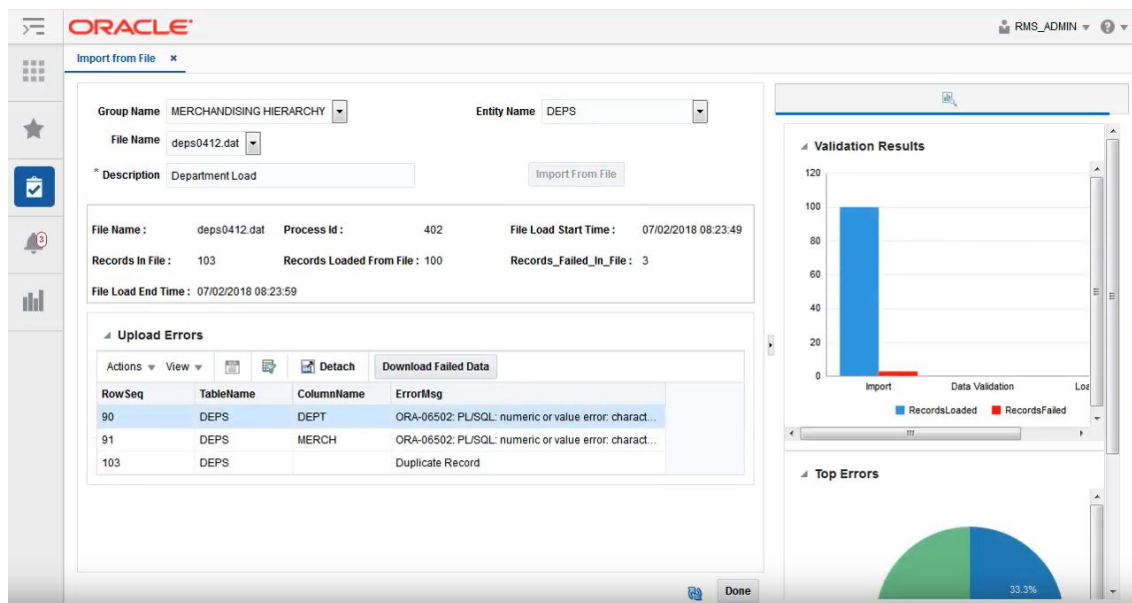
Figure 3.6: ORMFCS - Mass Uploads (Oracle Docs)

Oracle Zero Downtime Migration

This is one of the tools that Oracle provides that enables easy and efficient migration of on-premises databases to Oracle Cloud. It leverages Oracle MAA (Maximum Availability Architecture) technologies such as Oracle Active Data Guard and Oracle Golden Gate to minimize the maximum possible of downtime. For this case, it can establish as source databases the following (Oracle 2017):

- 12c, 18c, 19c (CDB/PDB);

- 11g, 12c, 18c, 19c (Non-CDB).

As targets, possible as Oracle DBaaS, Exadata Cloud Service or at customer:

- 11g, 12c, 18c, 19c.

SQL Developer

This one is a known integrated development environment that allows the development and management of Oracle databases in both traditional and Cloud deployments. It offers many helping features such as E2E development of PL/SQL applications, worksheet to run queries and scripts, a DBA console to manage the database, reports interface and a complete data modeling solution. But the one feature that is of high relevance in this case is the migration platform, that allows to "move" the databases to the Oracle Cloud. Noting that this is the crucial point along this thesis, sending the data to the Cloud integrating this step with the process already being in use by Wipro(Oracle 2017).

For this case, it can establish as source databases the following:

- 12c, 18c, 19c (CDB/PDB);

- 11g, 12c, 18c, 19c (Non-CDB);

- AWS Reshift;

  - AWS RDS.

As targets, there are several possibilities for 11g, 12c, 18c and 19c databases on:

  - Oracle Database Cloud Services Bare Mteal;

  - Oracle Database Cloud Services VMs;

  - Exadata Cloud Service;

  - Exadata Cloud at Customer;

  - Autonomous Data Warehouse;

  - Autonomous Transaction Processing.

SQL Loader

SQL Loader tool is used to load from any other data source into Oracle. But it's only able to read data from flat files. The most common procedure to load data from a third party database into Oracle using SQL Loader is detailed in the next steps (*Using Oracle SQL Loader with Examples* n.d.):

  - Extract data into Flat files from the third party database;

  - Create the table structure in the Oracle Database target using appropriate data types;

  - Write a control file, describing how to interpret the flat file and options to load the data;

  - Execute SQL Loader utility specifying the control file in the command line argument.

It's worth highlighting that this can be easily done using SQL Developer or Wipro's DCT. They already have a functionality that generate shell scripts that invoke the SQL Loader, based in the files or interfaces defined in the process of creation.

Oracle Golden Gate

Oracle GoldenGate is a known tool at Wipro, and it was already used in a few past projects which can be an advantage because there a few coworkers with experience using it. Golden Gate enables real-time data integration and replication in different IT environments, allowing homogeneous or heterogeneous migrations. It follows an ETL approach, divided in different stages such as:

  - Data Capture or Extraction;

  - Data Transformations;

  - Data Replication;

  - Track of operations and analytical information.

The most recent versions of Golden Gate bring extreme performance with simplified configuration and management, tighter integration with Oracle Database, support for cloud environments, expanded heterogeneity, and enhanced security. In addition to the Oracle GoldenGate core platform, Oracle provides a Management Pack to monitor Oracle Golden-Gate deployments and Veridata that allows high-volume data comparison between two in-use

databases (Oracle n.d.[b]). When it comes to the available and compatible databases, the following are denoted and represented in the figure 3.7:

- Amazon Relational Database Service;

- SQL Server;

- Oracle databases (matching versions or higher ones);

- Autonomous Database;

- Azure SQL Database.
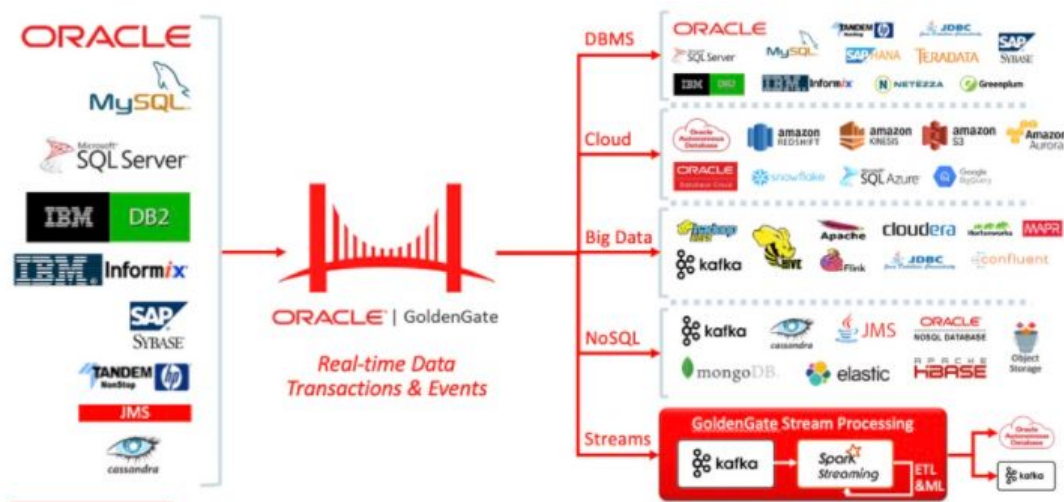
Figure 3.7: Golden Gate Compatibility (Vengal 2019)

Oracle Data Transfer and Appliance

This is one of Oracle solutions that allows to migrate data to the Oracle Cloud Infrastructure. It's basically an offline data transfer that works in both ways, from on-premise to Cloud and vice-versa. Due to several possible reasons of moving data through public internet such as high network costs, unreliable network connectivity, long transfer times and other security concerns, this can be a good solution for several companies and offers faster upload and are easy to use(Oracle 2020c). There are still some regions restrictions for this solution, so it may not be a good option for Wipro in a bigger picture. This solution offers three distinct options: disk-based data transfer, appliance-based data transfer and appliance-based data export. In the first case, the data is sent on encrypted disks to an Oracle transfer site. Then the staff at Oracle site upload the files into the desired repository. Requires the client to purchase the disks that are used to transfer data, in the end they are shipped back to the client. The remaining options are appliance-based and allows transfer and exportation of data. The process is similar to the first one, but are used Oracle's devices instead of our own. There are a few constraints related to the limits on Data Transfer Service Resources that have to be discussed with Oracle if an increase of limit is needed, depending of the business size of the company in question.

Database Migration Assistant | Qlik Replicate

These are two examples of tools created by Microsoft that allow to do data migrations into

Cloud. The first one, Data Migration Assistant, allows customers/companies to upgrade to a modern data platform and is able to detect any compatibility issues that occur during this transition to a newer version of SQL Server or Azure SQL Database. Gives performance and reliability improvements options for the target environment, allowing to move the schema, data and other objects from the source to the target server(Docs 2019). All the breaking, behavior or deprecated changes are reported to the user, and DMA offers a set of recommended options or possible approaches to mitigate them. Important points have to be noted related to the supported source and target versions:

- Sources;
  - SQL Server 2005;
  - SQL Server 2008;
  - SQL Server 2008 R2;
  - SQL Server 2012;
  - SQL Server 2014;
  - SQL Server 2016;
  - SQL Server 2017 on Windows;
- Targets.
  - SQL Server 2012;
  - SQL Server 2014;
  - SQL Server 2016;
  - SQL Server 2017 on Windows and Linux;
  - SQL Server 2019;
  - Azure SQL Database single database;
  - Azure SQL Database managed instance;
  - SQL Server running on an AVM(Azure Virtual Machine).

On the other hand, Microsoft has another tool, Qlik Replicate, that has similar features to DMA described above, but allows to migrate from an Oracle database to an Azure SQL Database. This can be more useful for Wipro, when it comes to the source system being used that is Oracle and if the client desires to migrate the data to Azure (Microsoft 2019).
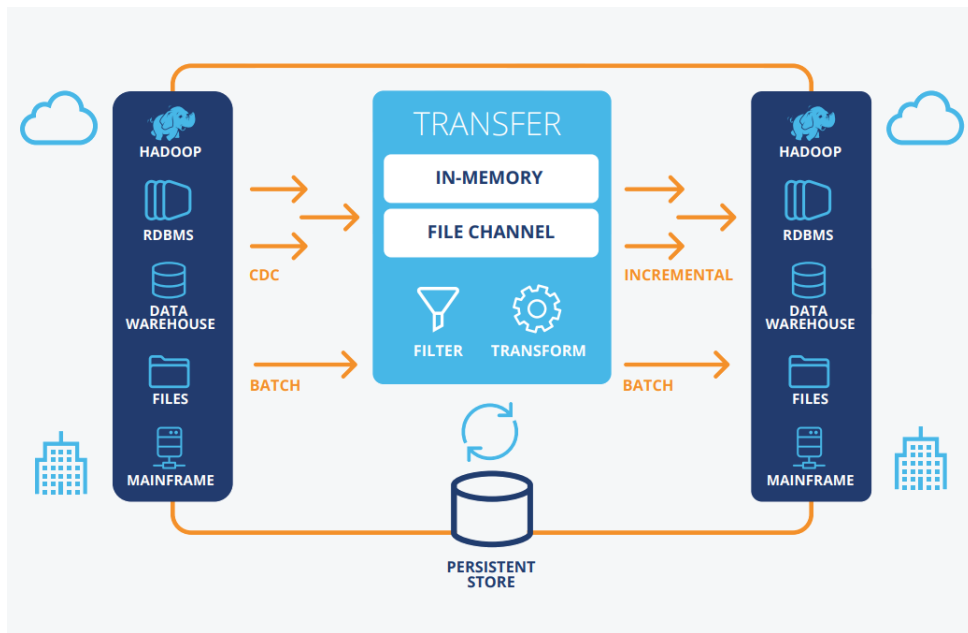
Figure 3.8: Qlik Replicate (Techniques 2020)

But, although it has a large database compatibility including on-prem or cloud DBaaS, it still cannot perform migrations to Oracle Cloud which is big setback. In the figure 3.9 we can take a closer look into all the options that are available.
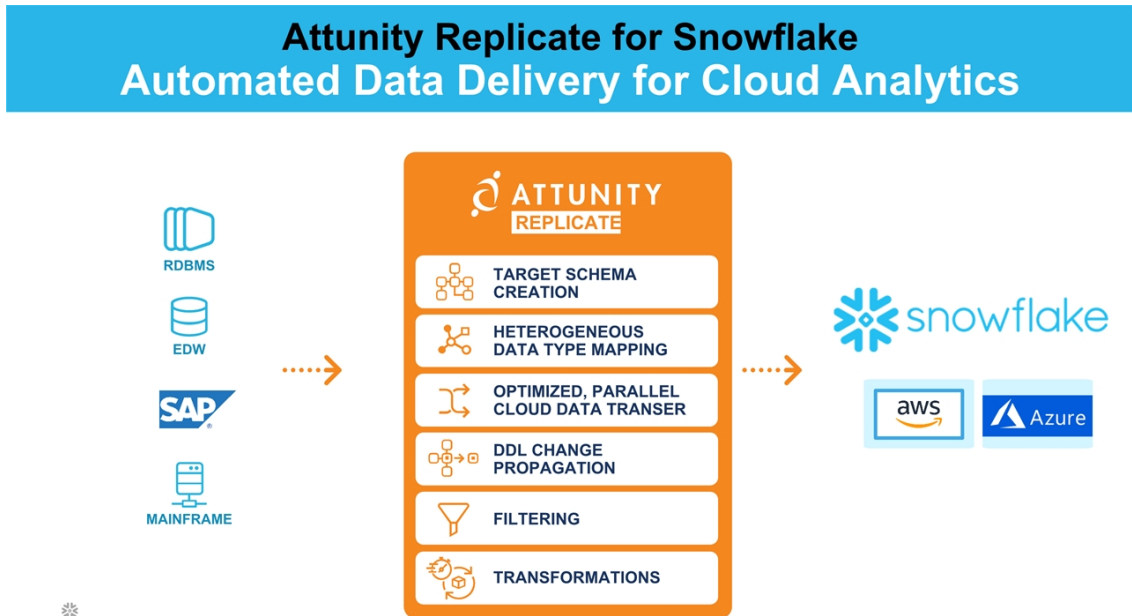
Figure 3.9: Qlik Replicate Compatibility (Alon 2018)

AWS Database Migration Service

AWS Data Migration Service helps to migrate databases to AWS quickly and securely. It's very simple to use, and lets the source database be fully operational during the migration,

minimizing the downtime of any application that relies on it. This service is very versatile allowing migrations from almost every kind of database (commercial or open-sourced). Mostly important supports homogeneous migrations (Oracle to Oracle), as well as heterogeneous migrations between different database platforms, for example Oracle or SQL Server to Amazon Aurora. All the data can be easily replicated into data warehouses by streaming it to Amazon Redshift and Amazon S3, allowing to have data secure, safe and backed up (AWS 2019). The figures shown below represent examples of data replication and homogeneous migration respectively.
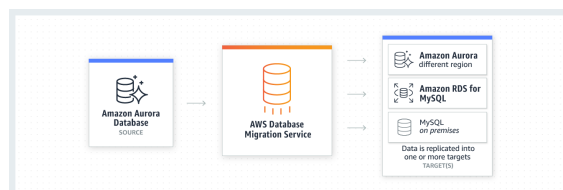
(a) Data Replication (AWS 2019)

(b) Homogeneous Migration (AWS 2019)

Figure 3.10: Homogeneous Migration and Data Replication

3.3.2 Cloud Computing Services

In this section, we will describe the most popular cloud computing services that may be part of our final solution. Although the most certain thing is to stick to Oracle since it is one of the key partners of Wipro. On the other hand, all applications derived from Oracle Retail are only found on the Oracle Cloud as SaaS. If it's decided to put the applications on the Cloud without making the transition to the services already provided, it would be necessary to analyze any restrictions imposed by Oracle with respect to its services/applications on a Cloud other than theirs. An overall state of each Cloud computing service can be checked in the Figure 3.11(Solutions 2018).

Figure 3.11: Cloud Computing Services Positioning (Solutions 2018)

Amazon Web Services

Basically the world's largest online book store also happens to be the world's largest public cloud provider. AWS is able to provide PaaS (platform as a service), IaaS (infrastructure as a service), serverless computing, rounding over 1000 different services, and is continuously releasing new features every year(INQDO 2019). The most commom services are divided into:

- VPC (Virtual Private Network);

- EC2 (Elastic Compute Cloud);

- RDS (Relational Database Service);

- IAM (Identity and Access Management).

Figure 3.12: Amazon Web Services (Sharma 2019)

Microsoft Azure Cloud

Microsoft Azure is the second largest Cloud platform in the world. It's basically a set of cloud services that help every organization to achieve their business challenges. Allows to build, manage and deploy applications on a global network using the desired tools and frameworks(Azure 2019a). Microsoft focus on being continuously innovating, easing all the development nowadays, preparing new solutions for the future in parallel. Provides a hybrid cloud, which allows to meet any client requirements/requests, integrating on-premises with cloud applications, supplied with top security backed up with experts and proactive compliance that is already trusted by several companies and even governments.
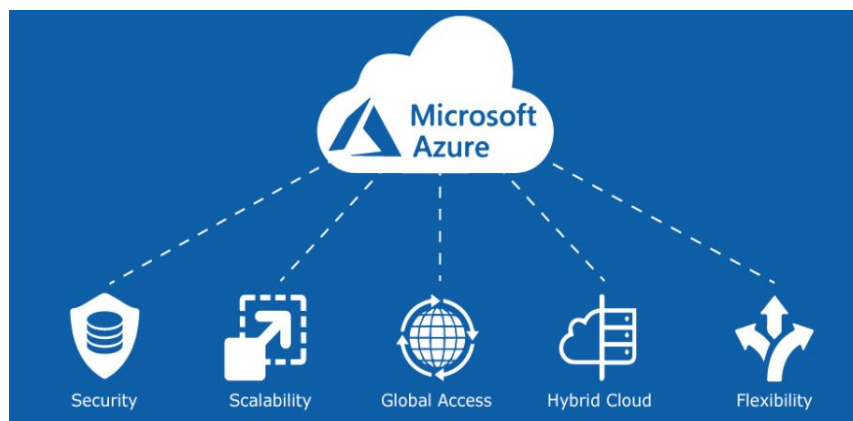
Figure 3.13: Azure Advantages (Kumar 2018)

Oracle Cloud

Finally the last cloud computing service that will be described, Oracle Cloud, it's very significant because it will be the most probable service to be used as part of the final solution. Like the others it offers three types of services:

- Software as a service;

- Platform as a service;

- Infrastructure as a service.

Oracle as part of software as a service, it offers cloud applications that connect business processes across enterprises. Covers many areas such as: customer experience, enterprise resource planning, in this case focusing Retail business, capital management and others. All of this allows business to be personalized, connected and secure. As examples of Oracle Cloud SaaS for Retail we have: Oracle Retail Merchandising Cloud Services, Oracle Commerce Cloud and Oracle Retail Customer Engagement Cloud Services and many others. Oracle Cloud's PaaS is the industry's leading enterprise cloud platform and like the offered SaaS helps organizations achieve innovation and business transformation with ease. Like Amazon Web Services, Oracle's IaaS offers elastic computing and storage, running any workload necessary for the business in question. However distinguishes itself from the others services available because it offers a fully integrated environment that is optimized for the cloud, has a unified security model and discards the use of any third-party software(Azure 2019b).

Chapter 4

Analysis

In this chapter, an overall analysis of the current migration process will be done in order to understand correctly how it stands in this precise moment. This will allow to check every data migration phase in detail, which brings us to the last section where we are going to focus on describing the new data migration process including Cloud as an option, and every sub-process associated to it.

4.1 Current State of Data Migration

The current state of data migration at Wipro follows the ETL (extract, transform, load) approach explained in the previous chapter 3 in section 3.2. First, a deep analysis of the system to be migrated is done. It can be a Legacy system from the client or an old version of an Oracle Retail Application. After the analysis, an extraction is done into several files with all the information from the old system database. The extraction should be controlled and divided into interfaces in order to enhance the performance of the process and ease future change requests. In this case, interfaces represent different business areas of Oracle Retail, and are used in every stage of the data migration which allows a better understanding of the overall process. The second stage is where our Data Conversion Tool will begin to work with the files extracted during the first phase. Two environments need to be set up, locally and on-site, each one with a respective database, working as a big staging environment for data processing. The first one is responsible for the preparation of the entire migration which includes the creation of tables, columns, keys, restrictions, business validations and the generation of loading/validation scripts, which is crucial for the third stage. Finally, in the third stage is where the second environment is going to be used. The generated scripts need to be passed via SFTP using a third-party application to the machine, then the files are loaded and validated by interfaces, as previously described, in the DCT database. If the data passes all the validations, we proceed to the last step, loading all the validated data into the client's new solution DB. Highlighting that we are stepping over a few steps, as for example, disabling and enabling constraints/keys during the migration to increase the performance. An overview of the current state of data migration in Wipro can be seen in the following Figure 4.1.
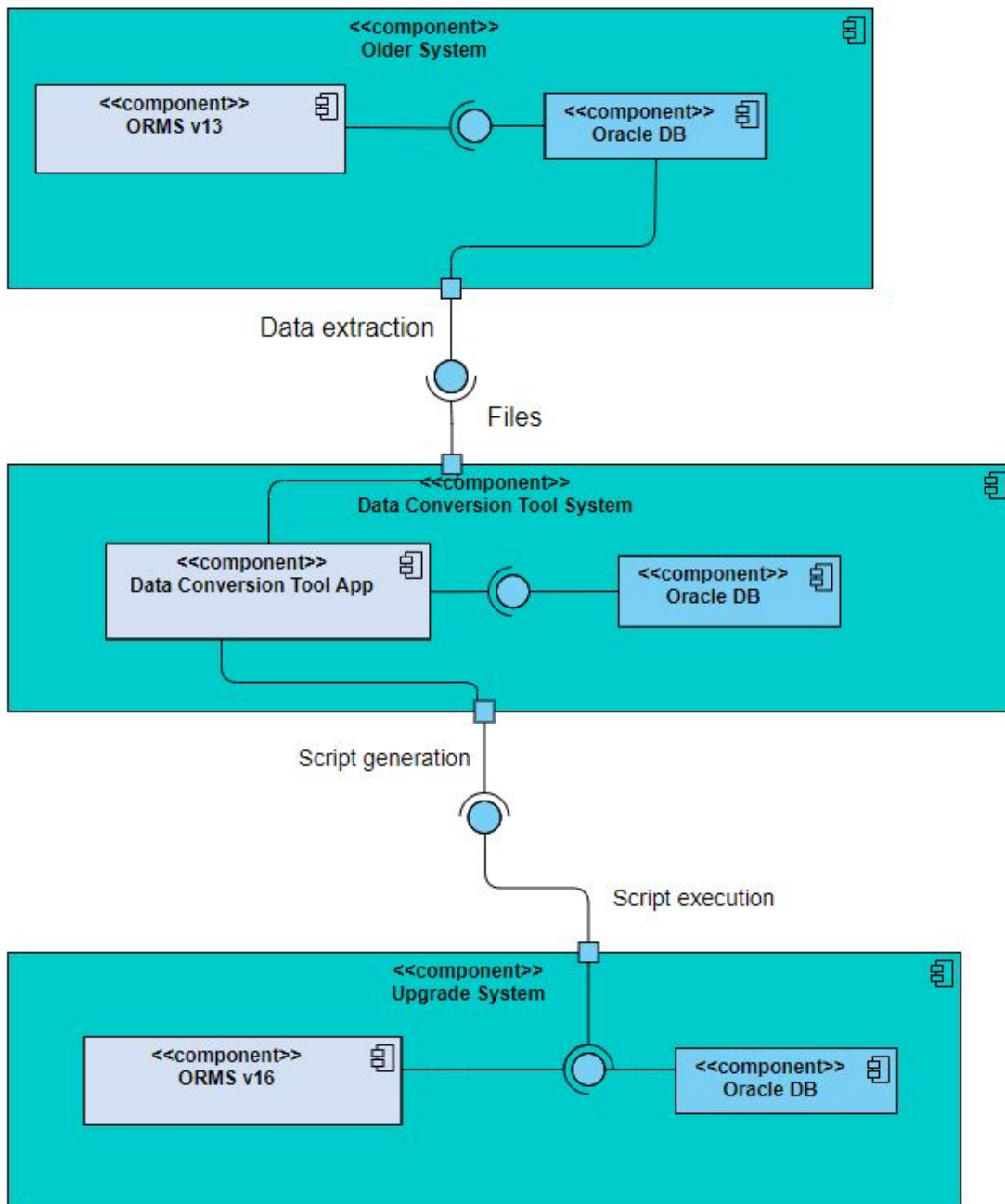
Figure 4.1: Current Data Migration Process

4.2   Migration Approach

This section describes the three phases of the proposed migration process:

- Migration Planning;
- Migration Execution;
- Migration Evaluation.

Each one of these phases has several sub-processes that need to be detailed and correctly analysed and planned. This process will take into account the use of Golden Gate and SQL Loader tool as middleware. But we need to emphasize that Golden Gate will have a higher impact in the current process taken by Wipro, that depending of the client requirements will require more or less steps. In the figure 4.2 we can take a closer look into all phases that form the basis of our proof of concept.
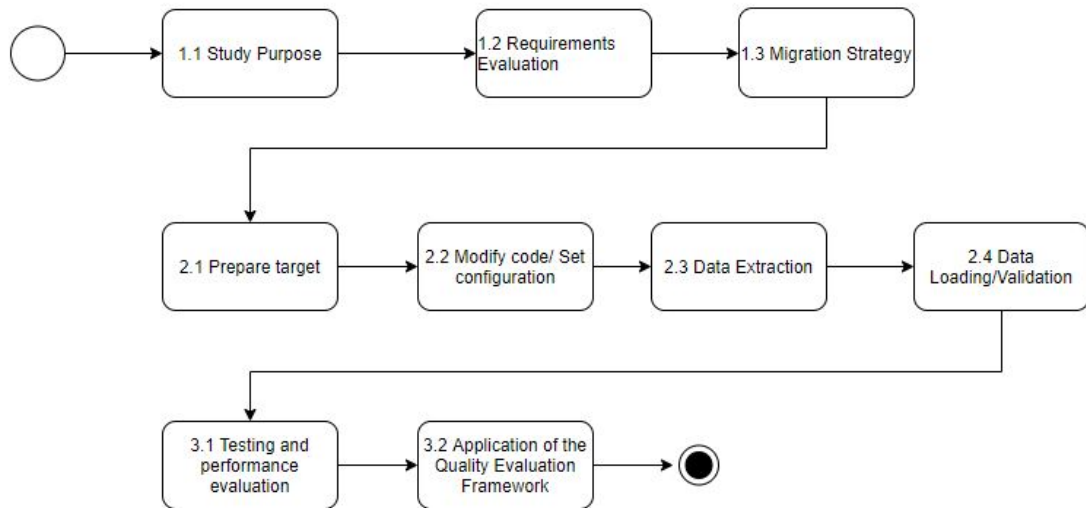
Figure 4.2: Migration Phases

4.2.1 Migration Planning Phase

The first phase that needs to be defined in a data migration is the planning one. The best way to plan is designing a sequence of sub-processes that represent the flow of operations that the migration scenarios, referenced in the previous sections, will go through. This flow will be divided into different points shown in the figure 4.3.
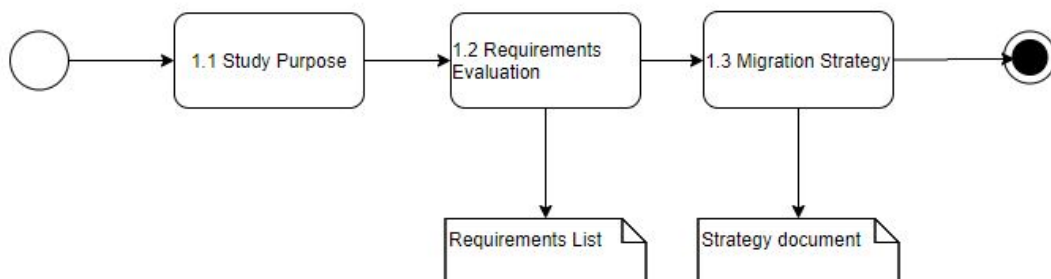
Figure 4.3: Migration Planning Phase

Purpose of Migration Stage

In the beginning of our planning phase, we need to draw the main objectives of our migration. Deep knowledge and understanding of the data migration scenario is a crucial point in order to avoid any setbacks later when already executing the data migration. The type of data migration needs to be correctly defined because the process will adapt depending if it's a migration from on-premise to Cloud or on-premise to on-premise.

Requirements Analysis Stage

In this stage, normally it's created a catalog with every subsystem or entities that are going to be migrated, which ends up being very helpful. This is a very technical procedure, but for our test cases we won't need those documents, it's preferable to give a high view of the systems that are present in the migration and the tables that will be used. After the migrations are performed, many points will be pending such elasticity, high availability or even technical support because it's driven by Oracle team and not Wipro's.

Develop Migration Strategy Stage

We will be following the strategy that brings the most benefits, that in this case is the "*Big Bang*" strategy with ETL approach that is described in 3, specified in the section 3.2.2. The chosen approach is ETL because it's the one mainly used in Wipro implementations. For the scenarios referenced above, we need to keep in mind that the data is already prepared in the source environment and we have the flat files ready too. Finally different procedures will take place:

- Migration to an Oracle Autonomous Database (Cloud);

- Migration to an Oracle Database as a Service.

Regardless of the procedure that is followed, it's always important to make backups of the data before any data migration execution.

4.2.2   Migration Execution Phase

In this process, tasks such as data extraction, target system setup/preparation and middleware configuration is essential. It's very important to understand the solution architecture, as well as the modifications applied in the target and staging environment. The steps that precede the current one provide the basic information for the execution of the migration, and which data is going to be delivered in the end of the process. We can check the flow of our migration execution in the Figure 4.4 below.
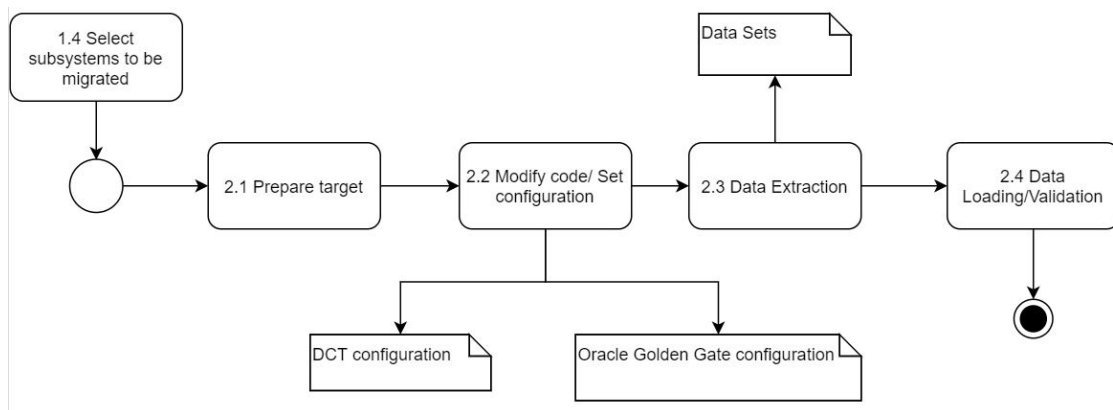
Figure 4.4: Execution Stage

Staging and Target Setup

In this stage, all steps must be performed with attention and awareness. It's preferable to use test environments that are already prepared when it comes to the DCT, Golden Gate, SQL Loader installation and respective databases preparation. The same applies for the cloud environment that hosts the Golden Gate Cloud Service and the target databases. Not forgetting that special privileges need to be set in order to achieve the desired result of the process. The possible migration scenarios will be described in the following chapter 5 but for each one the following steps are required:

- DCT Environment(source/staging);
  - DCT Application installed;
  - DCT Database installed;
  - SQL Loader installed;
  - Golden Gate On-premises.
- Cloud Environment(target).
  - Golden Gate Cloud Service Setup (target);
  - Autonomous Database Setup;
  - Database as a Service Setup.

DCT and Golden Gate configuration

Now this is a stage that needs detail, as it requires the full installation of Golden Gate in the our staging environment and in the target environment. For this purpose we will follow the Oracle documentation that is available and any specification added will be described in order to have a good understanding of the process. DCT is already installed and working in our test environment, which eases our process of implementation and configuration. So the majority of the focus will be in Golden Gate setup, but DCT will suffer small changes in its setup too in order to use SQL Loader to load data into the Cloud. As said before, several scenarios will be designed, referenced in chapter 5 and for each one it's important to make

specific configurations. Highlighting that DCT will have the same configuration for every kind of scenario, only Golden Gate will have variations but the files that are required to be configured are exactly the same:

- DCT configuration with SQL Loader;

    - mig_setup configuration;

- Golden Gate configuration for on-premises:

    - Manager configuration;

    - Extract configuration;

    - Pump configuration.

- Golden Gate configuration for cloud:

    - Manager configuration;

    - Replicat configuration.

Data Extraction

First of all we need to perform all the necessary extractions from the source system that has data to be migrated. In this case, the data extractions will be divided into:

- Golden Gate extraction from source database;

- Extraction into flat files for SQL Loader.

The first point is done automatically by a process configured in Golden Gate On-premises that is called Extract, that can be Classic or Integrated, but those points will be described in the next chapter. This process is able to connect to the database, extract data from the desired tables in a schema and saves it into files on Golden Gate file system. The SQL Loader scenarios, require a previous extraction into flat files, that is done with our predefined extraction scripts, but ends up using more human resources.

Data Loading

After all the preceding steps are done and validated, we proceed with the data loading. In this step, two kinds of loading will be performed:

- Through the use of Oracle Golden Gate to Oracle's ADW and Oracle's DBaaS;

- Through the use of SQL Loader that is invoked by the DCT generated scripts.

Heavy dummy files will be used in order to test the process at its limit and see if there any major differences while loading its data using SQL Loader compared to Golden Gate process that replicates the data from the database schema. It needs to be distinguished that in the case of SQL Loader the whole process will remain almost the same as previously said, and only the last step representing the connection to the target database in Cloud will be adapted. With the use of Golden Gate replication functionality, we won't need to generate in DCT the scripts to load the target system. We will only generate the scripts that load and validate data in the staging database and time will be spared. The main objective of

Golden Gate is to use our repository schema present in the staging database and replicate all the data inside it.

4.2.3   Migration Evaluation Phase

Notice that between on-premises the process will remain the same and will require only changes to be able to perform data migrations to Cloud, being more versatile and reusing what already exists. To make a good evaluation of the solution, performance tests will be carried to check the difference between integrating DCT with SQL Loader or with Golden Gate in order to perform data migrations to the Cloud. Finally a QEF will be designed to evaluate all the specifications of Golden Gate. In the figure 4.5 are displayed the steps and the obtained artifacts. All the details regarding each one of these evaluation steps are detailed in chapter 7.
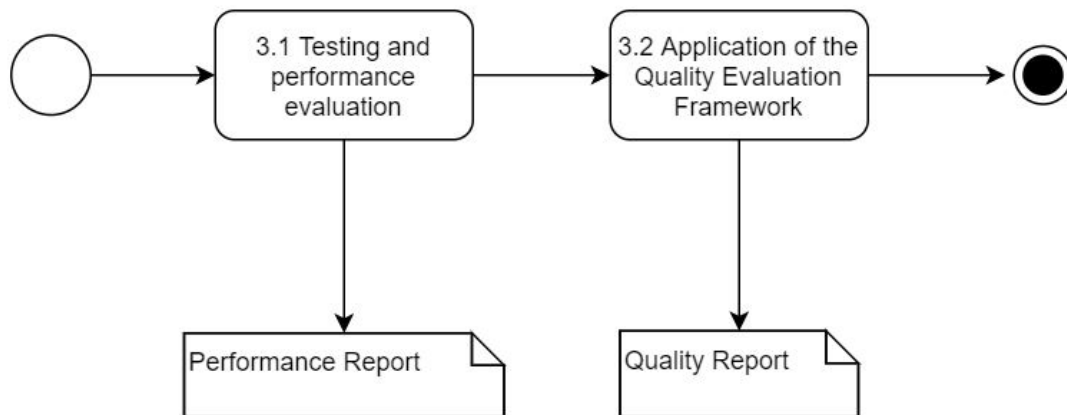
Figure 4.5: Evaluation Stage

Chapter 5

Design

In this chapter, it is presented the design of our proof of concept. It is necessary to highlight several parts of this solution that complete our data migration into Cloud. They are described in the following sections:

- Architectural Design;

- Cloud Integration with DCT using Golden Gate;

- Cloud Integration with DCT using SQL Loader;

First, an overview about the current and future solution architecture will be shown, allowing to understand the differences between each one. The different cloud data migration tools that were detailed in the chapter 3 will now be compared between each other, verifying carefully which one will adapt better to the conditions of the company's current process. Because Oracle as Wipro's partner can raise some restrictions to the use of their applications in other clouds than their own, so the use of each cloud computing service needs to be analysed carefully. In the last two sections, the focus will be to detail and explain each phase of the new migration processes. This will be divided by two possible approaches that include cloud integration using Golden Gate or SQL Loader as the chosen middleware.

5.1 Architectural Design

In this section, the methods/approaches and technologies discussed in the chapter 3 in section 3.3 are evaluated to conclude which is the best one for the current solution. In this case, as said before, we need to take special attention to possible restrictions that can be imposed by Oracle related to their applications in other Clouds than their own and using other technologies.

5.1.1 Possibles approaches to perform Cloud Data Migration

The section 3.3 of the chapter 3 presented different approaches/tools to include in Wipro's current data migration. To evaluate these approaches the following metrics need to be considered:

- Time/Performance;

- Effort;

- Costumer Satisfaction;

- Reliability;

- Customization.

A separate presentation explaining the different tools that were found in chapter 3 will be done in the following sub-sections, that will allow to distinguish the ones that are viable to this dissertation scenario and compliant with Oracle Cloud. Although, it's important to emphasize that they must respect homogeneous migrations between Oracle databases as they are the ones used in the DCT and in the OR solutions.

Solutions Restricted to Oracle Cloud

Taking into account that Oracle may impose some restrictions to their products being used in other Clouds, there were found some tools that can provide assistance for moving data into the Cloud. In this case, all the tools were developed by Oracle which are:

- Oracle Zero Downtime Migration;

- SQL Developer Migration Assistant;

- Oracle SQL Loader;

- Oracle Retail Merchandising Foundation Cloud Service;

- Oracle Data Transfer and Appliance.

The last option will not be used as it refers to physical transportation of the data with the use of our own or Oracle provided storage disks and that's not the solution we are looking for this project. This is followed up by Oracle Zero Downtime Migration and SQL Developer Migration Assistant that need to be discussed internally as viable options as they allow to perform database migrations quickly and with ease between on-premises Oracle Databases and Oracle DBaaS/Exadata Cloud Services (Oracle 2017). These would be possible tools to be used, if migration process used the DCT repository schema (DCTDR) or one without temporary(transformation) tables to be migrated. It would require small changes in the current migration process, which implies several discussion sessions to ensure that it is the right decision. Which brings us to Oracle Retail Merchandising Cloud Services, in this case specifically to the ORMFCS (Foundation), as it's the only one that has documentation, examples and proofs of allowing certain activities. Internally, it's a service already known, not in detail, but it has gain the trust of some coworkers. Oracle basically offers an enhanced OR solution with a feature that allows importing data from flat files, which then goes through a group of stages as mentioned in the chapter 3 in section 3.3. This solution has been already discussed, because it covers part of the necessities of Wipro and fits in the current process of data migration. However it also shows performance flaws because it's quite limited when it comes to importing high numbers of records, being retained in the amount of 50000. And the interaction with the user could be improved compared to the DCT. On the other hand, there are other services such as ReSA CS and RPM CS where there isn't enough information in order to check if the respective Cloud services have the same feature for data migration. Finally, we have the SQL Loader utility tool that has been used for a long time in Wipro's data migration methodology. During a migration,the DCT is responsible for generating scripts that make use of the SQL Loader which automates every process of creating and inserting data into staging/target tables. This tool needs to be part of our final solution because it's currently being used to do data migrations between on-premises

systems and it has a great performance. It's urgent to know how to enhance it's integration with DCT in order to load data into the Cloud. For now it's possible to see and understand its distribution during a cloud data migration through the deployment view represented in the Figure 5.1. It's important to highlight that SQL Loader is installed inside the DCT environment. Then, the new configured DCT will invoke SQL Loader that will be able to perform data migrations into the target databases hosted in the Cloud.
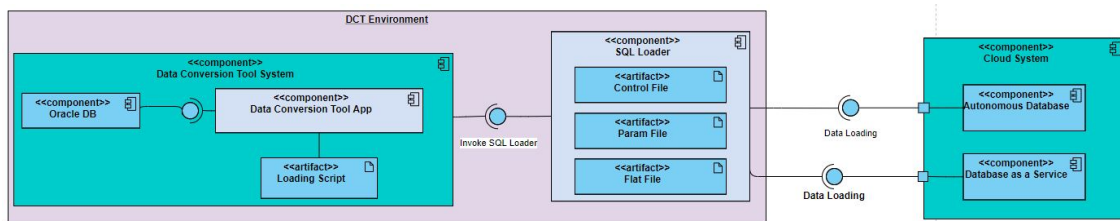
Figure 5.1: SQL Loader Deployment View

And through the following sequence diagram, it's clear how the DCT and SQL Loader "communicate" with each other to perform data migrations. SQL Loader ends up using the input files extracted in the first stage of the data migration and the control files generated by the DCT. In the end, after the loading is performed or interrupted, it generates several files that are divided into:

- Log File - track of operations performed;

- Bad File - records that cause errors;

- Discard File - records not selected for loading.
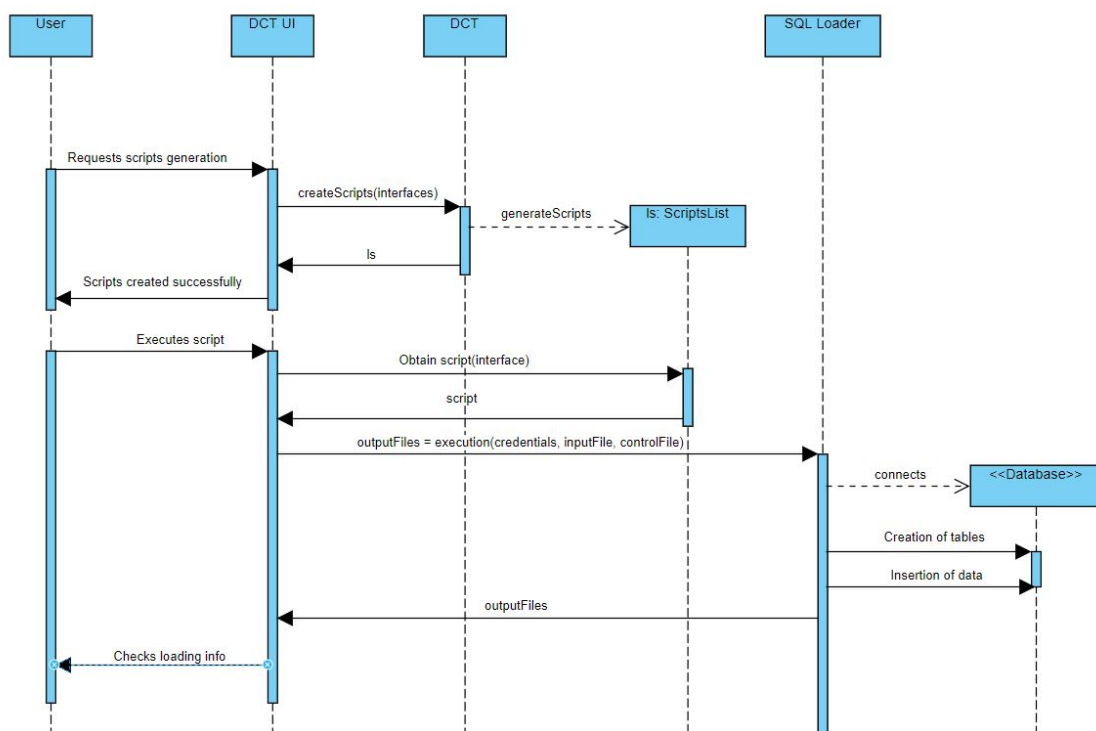
Figure 5.2: SQL Loader Sequence Diagram

Non-restricted Solutions

Having the possibility of using other cloud computing services without Oracle imposing any restrictions related to their products, there are other tools that need to be referenced as a possible piece to integrate the current data migration process with Cloud. From Microsoft there are two possible tools: **Data Migration Assistant**, also known as DMA, and **Qlik Replicate**. On the other hand, Oracle also provides one dynamic tool that is **Oracle Golden Gate**. **DMA** has many good features such as describing possible compatibility issues and offering recommendations against them, on the other hand allows to migrate from SQL Server as Source to SQL Server or Azure SQL Database in Azure Cloud. Although it's not a very viable option in this case and it ends up being a poorly dynamic tool due to two conditions:

- Type of database (SQL Server, Azure SQL);

- Compatible with Azure Cloud only.

However Microsoft's Qlik Replicate it's a way more versatile tool, which allows to work into Amazon's, Google's and Microsoft's cloud, and allows migrations in which the source can be Oracle's DB. This represents an important point because most of the used databases in OR are Oracle, and it has a wide range of options when it comes to choose a type of DB for the source or target, allowing relational and non-relational models. In short, this would be a great tool to integrate in the current process of data migration to cover the Cloud concept and it's very dynamic covering crucial Cloud platforms. Then we have the AWS migration service, which allows to migrate data to the Amazon Cloud, being very versatile regarding possible sources and destinations to perform the data transfer. It also allows homogeneous migrations between Oracle databases like some tools explained previously, which in this case is quite important. It has many other advantages such as low cost, easy to set up in AWS Management Console and simple to use. Finally, there is **Oracle's Golden Gate** that is the one we are going to explore in the following sections and in chapter 6, implementing the solution "around" it. It should be noted that this application and cloud service is complex to install and to configure, but ends up having many pros such as:

- Allows homogeneous and heterogeneous migrations;

- Track and history of all operations done during the migration;

- Supports heavy data loading;

- Control of DML operations;

- High compatibility with cloud services, including Oracle Cloud.

In this list of possible tools, there is one that stands out that is Golden Gate. In the Figure 5.3, it's represented a possible system architecture that would integrate DCT with Oracle Golden Gate to perform cloud data migrations. This view is divided into two environments, source and target. The source environment will have DCT, its database and Golden Gate on-premises installed, on the other hand, at the target environment (cloud) we will have the target databases prepared and Golden Gate service running. Noting that this system can suffer some changes later, depending of the chosen technology/tool or even cloud computing service. This choice will be definitive in the section 5.1.1 in order to proceed with the correct design and implementation.
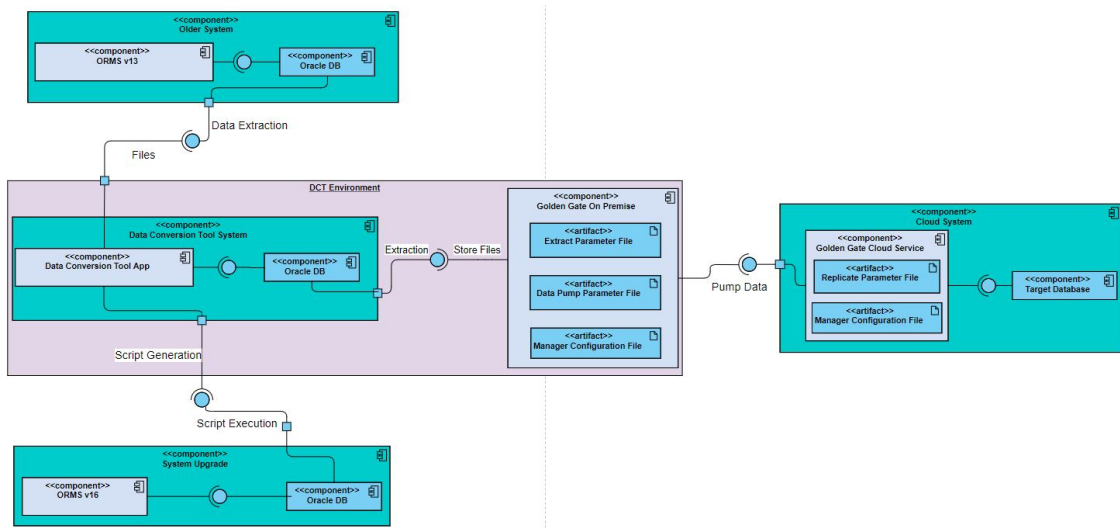
Figure 5.3: New Data Migration Process

Through the following sequence diagram, it's clear how DCT and Golden Gate "communicate" with each other to perform data migrations. This diagram focus in the operations done by each process executed by Golden Gate, in order to understand the flow of a data migration when using it as a middleware. Golden Gate on-premises will be responsible of extracting data from the source database, and then it will transfer all the extracted files into Golden Gate that is hosted in Oracle Cloud. Finally the Golden Gate that is in the target environment will replicate the data to the database hosted in the Cloud.
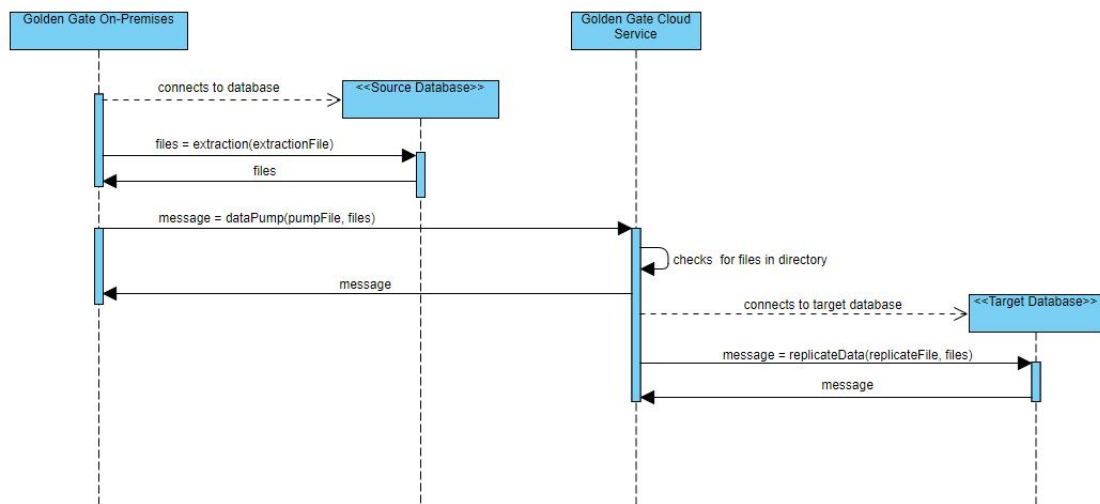
Figure 5.4: Golden Gate Sequence Diagram

Migration Tools Analysis

In the following table 5.1 several possible cloud migration tools are compared with each other in order to check the main differences from one to another when it comes to the concept of data migration. But for this project, the solution will reside in Oracle's Cloud. For that reason we can focus in other tools that the market offers or in the service already

Table 5.1: Migration Tools Comparison

Attributes	ORMFCS	Microsoft DMA	AWS DMS
Files	Yes	No	Yes
Schema	No	Yes	Yes
Oracle DB	Yes	No	Yes
Other DB's	No	Yes	Yes
Price/License	Free/On-Demand	Free/On-Demand	Free/On-Demand
Cloud Services	Oracle	Azure	Amazon

provided by Cloud mentioned above. All these conditions mentioned above allowed to design and apply correctly the AHP method, in section 2.6.5, when it comes to the choice of a middleware able of fulfilling the needs of the company. For this investigation to be complete and detailed, two tables were created in order to describe specific tools from Oracle and other companies. In the following tables 5.2 and 5.3 several possible tools are compared with each other taking into consideration crucial stages/attributes of a data migration (Stitch 2020).

Table 5.2: Oracle's Solutions Comparison

Attributes	SQL Developer	Golden Gate	DCT
Focus	DR [1]	DR, ETL	ETL
Files Import	No	Yes	Yes
Schema	Yes	Yes	No
SaaS sources	No	Yes	No
SLAs support	No	Yes	Yes
Security certifications	Yes	Yes	No
Oracle DB	Yes	Yes	Yes
Other DB's	No	Yes	No
Price/License	Free	Free/On-demand	Free
Cloud Compatibility	Oracle	Oracle, Amazon, Azure	None

[1] Data Replication;

In summary, many tools are available on the market and most of them meet the minimum requirements for a data migration, which can include data extraction, transformation / validation and final loading to the destination. But depending on the customer's requirements and specifications, there is always one that becomes the most suitable. For example, if we wanted to migrate to Azure Cloud, we would have DMA and Qlik Replicate which are compatible and support multiple databases. On the other hand, if a data migration to AWS was necessary, Amazon itself provides a tool, DMS, which also supports numerous databases and has a good performance. In this case, and in order to improve the current Wipro data migration process, the following tools/services are the ones that need to be highlighted:

- ORMFCS;

- SQL Developer;

- Golden Gate;

Table 5.3: Other Tools Comparison

Attribute	Qlik Replicate	Stitch Data Loader	DCT
Focus	DR [2]	ELT [3], DI	ETL
Files Import	Yes	Yes	Yes
Schema	Yes	Yes	No
SaaS sources	No	Yes (>100)	No
SLAs support	Yes	Yes	Yes
Security certifications	No	Yes	No
Oracle DB	Yes	Yes	Yes
Other DB's	Yes	Yes	No
Price/License	Not disclosed	Not disclosed	Free
Cloud Compatibility	Azure, Amazon, Google	Azure, Amazon, Google and Oracle	None

[1] Data Replication;
[2] Data Ingestion;
[3] Extract, Load and Transform;
[4] Not compatible with Oracle Cloud;

- SQL Loader.

But only two of them end up serving as possible solutions, which are Golden Gate and SQL Loader. They end up being the ones that best adapt to the described process in the beginning of this chapter. They allow an easy integration with the tool that is currently being used in the company, the DCT. And two scenarios are possible:

- SQL Loader integrated with DCT to load data into Oracle Cloud;

- Golden Gate integrated with DCT to load data into Oracle, Azure or Amazon Cloud.

So, both can be used, but the chosen solution is dependent of the requirements, time and effort of the implementation in question. So, as said before, in the following sections, the detailed design for each one of the solutions will be presented, but the major focus will be on Golden Gate tool since SQL Loader has already been used in the company and its integration is mostly fulfilled but lacks configuration. In chapter 7 tests will be made in order to evaluate the behaviour and performance of each one in different migration scenarios.

5.2 Cloud Integration with DCT using Golden Gate

As explained previously in our value analysis, presented in the chapters 2 and 4, for one solution of our proof of concept we will be using Oracle's Golden Gate tool to be responsible of performing data migrations into the cloud. Golden Gate is a very powerful tool created by Oracle that allows to replicate/migrate data between different systems, and for this case study, is the tool that best suits Wipro's desires and future implementations due to Oracle dependencies. We can check its main process in the following Figure 5.5 below.
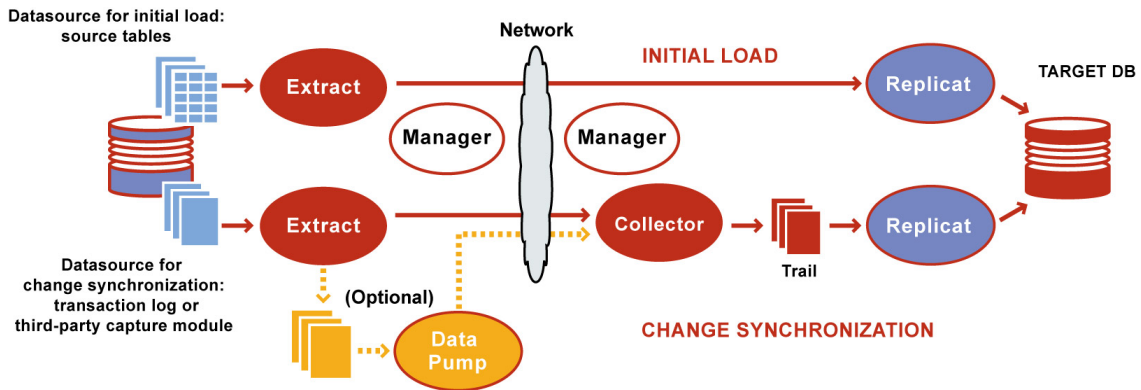
Figure 5.5: Golden Gate Procedures (Oracle 2020a)

This process diagram makes it clear to understand in what stages a data migration will be divided when using Golden Gate as middleware. It also highlights the different steps that are taken in each one, underlining that both Golden Gate need to be running in order to act as described below.
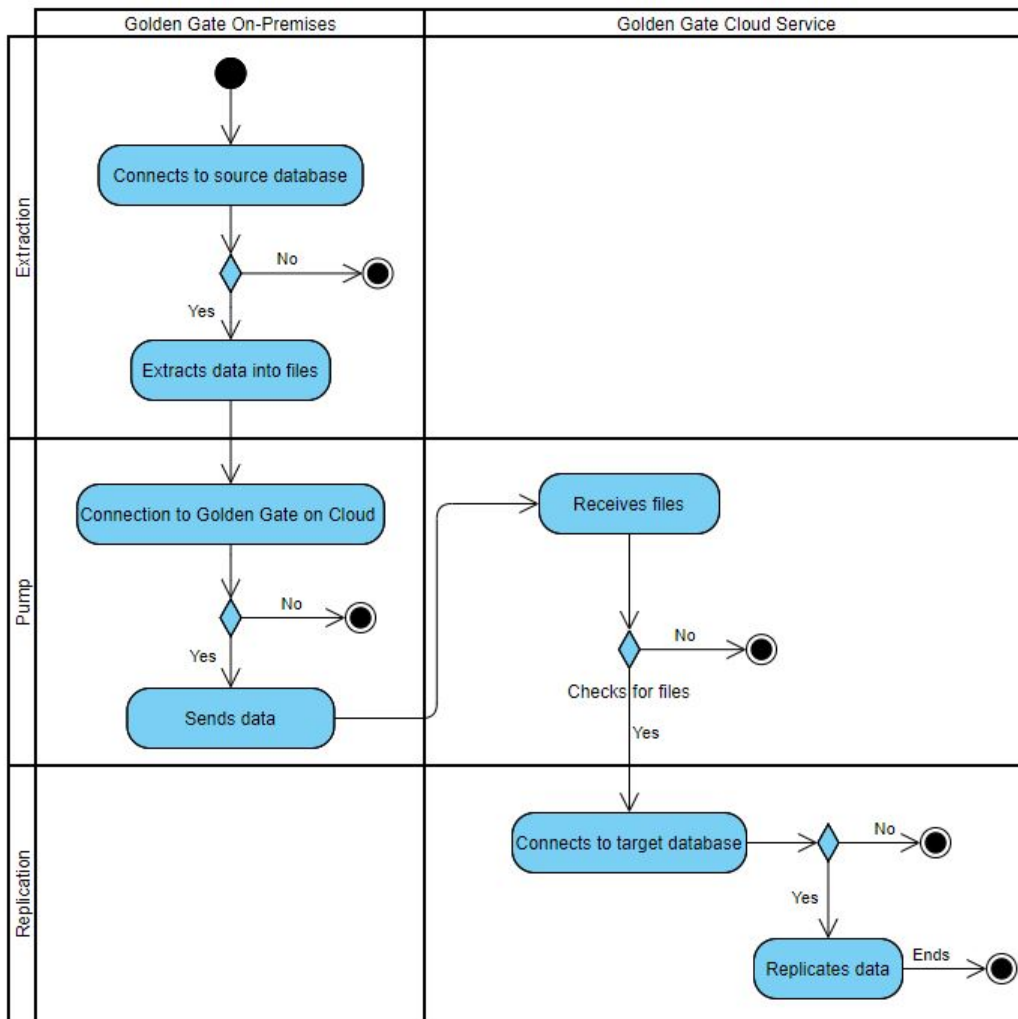
Figure 5.6: Golden Gate Process Diagram

First, it's crucial to understand the impact in Oracle Golden Gate implementation while using the databases above described. For that, every restriction is detailed in the following section 5.2.1, and the processes that Golden Gate can use in each situation will be analysed and we will see what difference it has in a data migration. In short, the following scenarios will be comparing key variables such as the type of database that is being used and the process of extraction/replication defined in Golden Gate. After an analysis of the restrictions, it will be designed the full behaviour of Golden Gate when it crosses three different scenarios:

- Loading from Non-Container Database into Non-Container Database which is represented as an Autonomous Database;

- Loading from Non-Container Database into Plugable Database (PDB) which is represented as a DBaaS;

- Loading from Plugable Database into Plugable Database in Cloud (PDB) which is represented as a DBaaS.

5.2.1 Database Restrictions and Golden Gate Functions

Before proceeding to the installation of Golden Gate, we need to have a good insight about Golden Gate functionalities and types of extracting/loading and take into consideration possible database restrictions when it comes to those same processes. Golden Gate application has several different processes that need to be carefully applied depending of the migration specifications. In the image 5.7 are represented every process and its evolution from one version to another, in this case it's being used one of the latest versions.
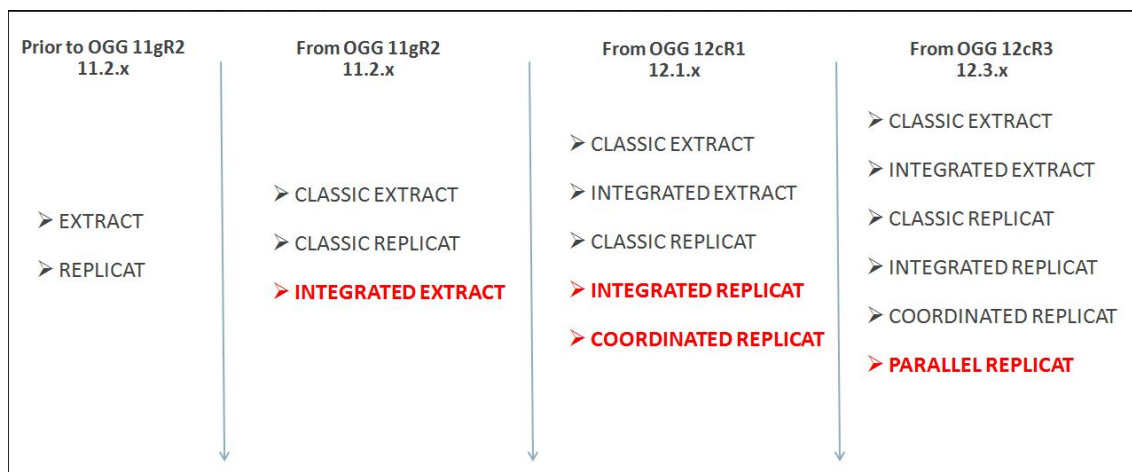
Figure 5.7: Golden Gate Processes (Veeratteshwaran 2018)

Each one of these methods has its advantages and disadvantages, and they are specific for certain version of databases and only allow specific data types too. So it's essential to have good knowledge about the data model and which data types are going to be used in order to filter the possible procedures. Then, we need to define whether the databases, among which the data migration will occur, are containerized or not. Whether it is a dedicated database or if we will have several pluggable databases on the same container. For a non-CDB database, there are several restrictions, and the Golden Gate's available and the number of

usable processes are very low (Oracle 2020b). This case applies to our migration into Oracle Autonomous Database scenario, where we can only use the following methods:

- Integrated/Classic Extract depending if the source database is containerized or not;

- Classic Replicat when it comes to the target (always Non-Integrated).

One important note is that these Classic methods are recommended to be used in databases which versions are below 11.2, so for the future it's better to avoid these procedures as they will be deprecated soon and aren't very good when it comes to performance. All the other processes support more recent database versions since the version 11.2.0.3, and only apply to Container Databases (CDB) or Multi Tenant environments. Therefore, for the other scenarios that have the DBaaS (PDB) as a target, we will be using classic/integrated extracts depending if the source database is containerized or not just like for the Autonomous Database scenario. Although, when it comes to data replication, integrated replicates will be used, as they are much more efficient, allow parallelism, multi-threading and are the only ones supported by containerized databases. An important aspect of integrated extracts/replicates it's that they use more recent procedures to obtain and load data. To get a deeper insight, it's displayed in the following list some attributes that make an integrate process much more complete than classic (Oracle 2020b):

- Integrated capture enables faster filtering of tables;

- Integrated capture handles point-in-time recovery and RAC integration more efficiently;

- Integrated capture features integrated log management. The Oracle Recovery Manager (RMAN) automatically retains the archive logs that are needed by Extract;

- Integrated capture uses the database log mining server to access the Oracle redo stream;

- Integrated capture is the only mode that supports capture from a multi-tenant container database;

- One Extract can mine multiple pluggable databases within a multi-tenant container database;

- Constructs logical change records (LCR) that represent source database DML transactions (in committed order).

All the files used in the migration scenarios are available in the appendix C in order to have a better understanding and knowledge of the commands used in each separate case. Below in figure 5.8 it's an example of the flow of a full integrated migration process. The steps that are represented in the figure are almost the same as the ones described in the previous process diagram, only the type of extraction and replication is changed. When it comes to the extraction, it's done by using the database log mining server, on the other hand, replication is done by using a lightweight streaming API that creates LCR's in representation of the DML commands.
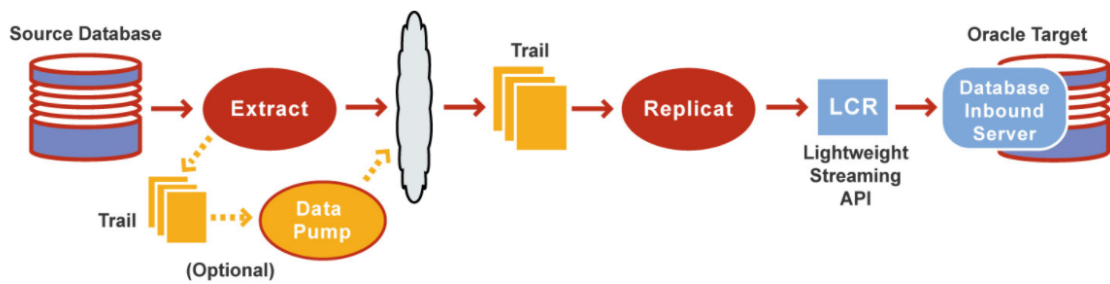
Figure 5.8: Golden Gate Integrated Process (Oracle 2020b)

5.2.2 Using Golden Gate to load data into an Autonomous Database

The first scenario it's related to the use of Golden Gate to load data into an Autonomous Database. For this case, it's important to highlight that the source and target are non-container databases which will have a negative impact in the performance of the data migration in this scenario. In the figure 5.9 we are able to see how the different systems will communicate with each other and how they are distributed. Above we have our source environment, that has a server with the DCT and Golden Gate On-premises installed. It's in this server that the classic extraction process will be executed by Golden Gate and when it's completed, the pumping process will be able to send all the generated trail files to the Golden Gate Server that is hosted and running in the Cloud. Finally, while the data transfer proceeds, the replicat process that is configured in the Golden Gate Cloud Service, will start picking the transferred files that are ready and will load its transactions into the Autonomous Database.
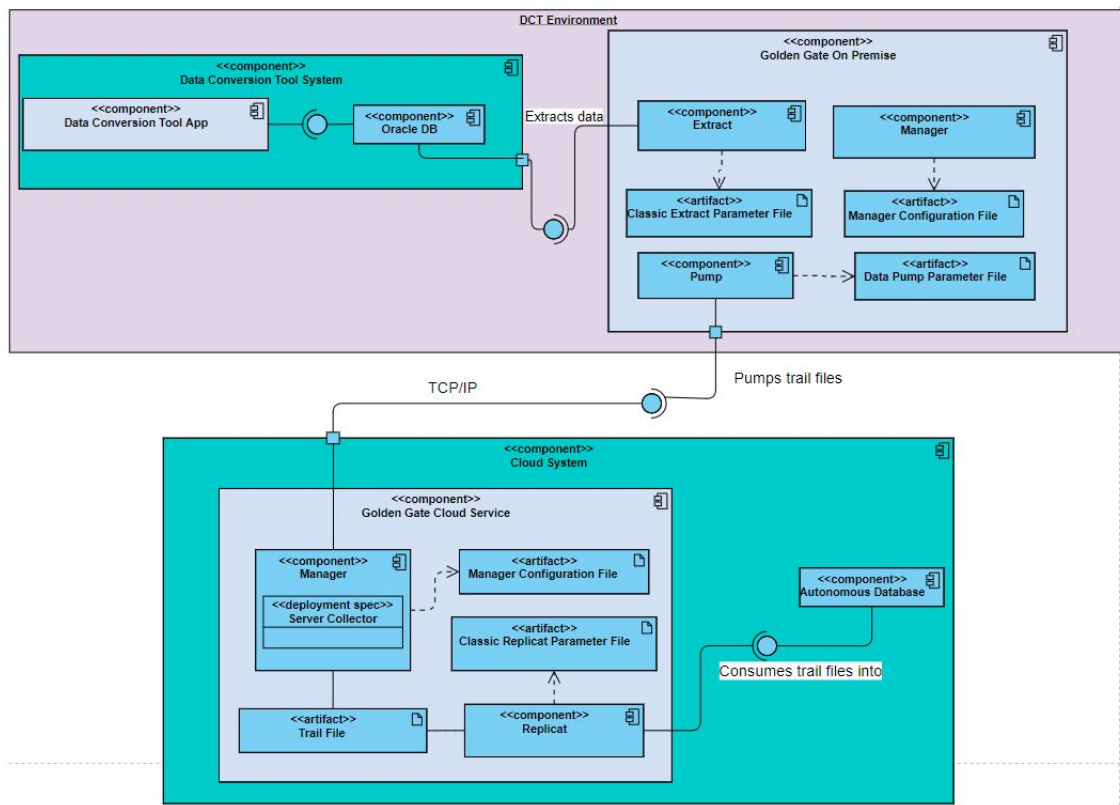
Figure 5.9:   Golden Gate to Autonomous Database Physical
View

It's easier to discern all the steps that are taken by Golden Gate during a data migration
into an Autonomous Database in the Figure 5.10 below. Basically, Golden Gate on source
environment connects to the database, extracts the data and sends it to Golden Gate on
target environment. Then the Golden Gate hosted in Cloud connects to the target database
and loads the data. It's important to highlight that because our source and target are non-
container databases, Golden Gate is only able to perform a classic extraction and replication.
This will have an impact in the data migration performance that will be analysed in chapter
7. That's why the next scenarios involve non-container and plugable databases in order
to cover all the possible entries when it comes to implement a solution using Golden Gate
integrated with DCT.
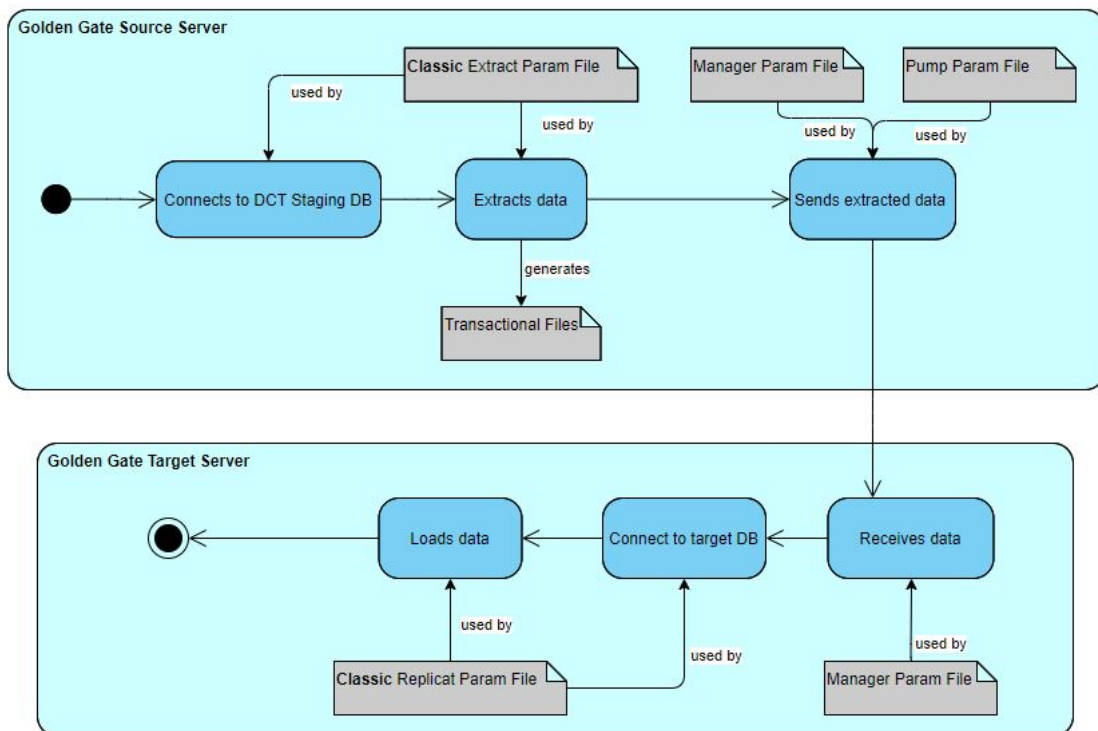
Figure 5.10: Golden Gate Classic Activity Diagram

5.2.3 Using Golden Gate to load data into Oracle's DBaaS

The second scenario it's related to the use of Golden Gate while loading data into a DBaaS. For this case, the data migration is being performed from a non-container database into a plugable database as a target. This scenario will be using an integrate replicat and and supposedly improves the performance of the data migration. In the figure 5.11 we are able to see how the different systems will communicate just like in the previous scenario. Above we have our source environment, that has a server with the DCT and Golden Gate On-premises installed. A classic extraction process will be executed by Golden Gate on source and when it's completed, the pumping process will be able to send all the generated trail files to the Golden Gate Server in Cloud. Finally, while the data transfer proceeds, the integrated replicat process, that is configured in the Golden Gate Server, will start picking the transferred files that are ready and will load its transactions into the Database as a Service. Important of noting that this case already applies the other type of processing when replicating data, that is the integrated one, although this does not have critical changes in our diagram because the main difference is portrayed in the configuration file of the tool.
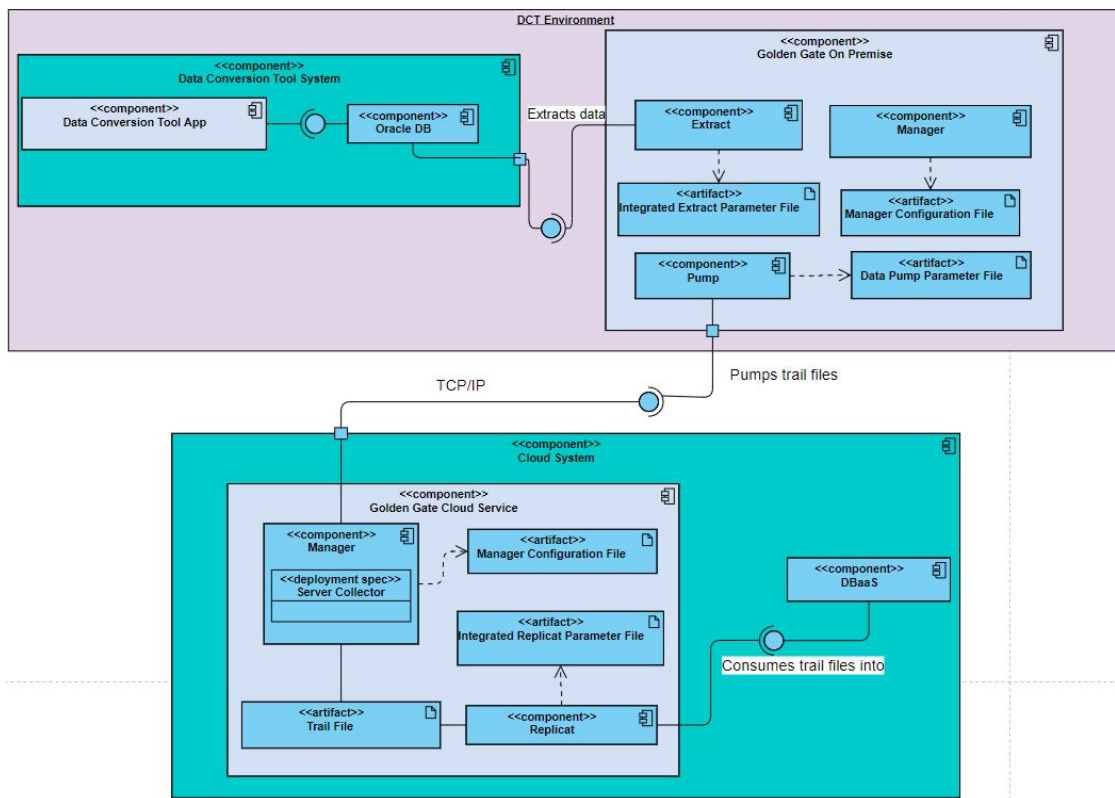
Figure 5.11: Golden Gate Fully Integrated Deployment View

Finally, for our last scenario involving Golden Gate, it's going to be repeated the previous process of loading data into Oracle's Database as a Service. This is a particular case, because both source and target are plugable databases which allow us to perform a fully integrated migration. In the figure 5.12 it's represented the physical view once again, all the steps executed are similar to the ones described in the past scenario, but as we said it's an integrated process. Golden Gate Server that is running in the source environment will perform an integrated extract, which is supposedly better than the classic one executed previously. The replication is integrated too which ends up being favorable to the data migration. Although, it's essential to compare the times obtained in each separate scenario in order to check which one has the better performance, this information is detailed in the chapter 7 .
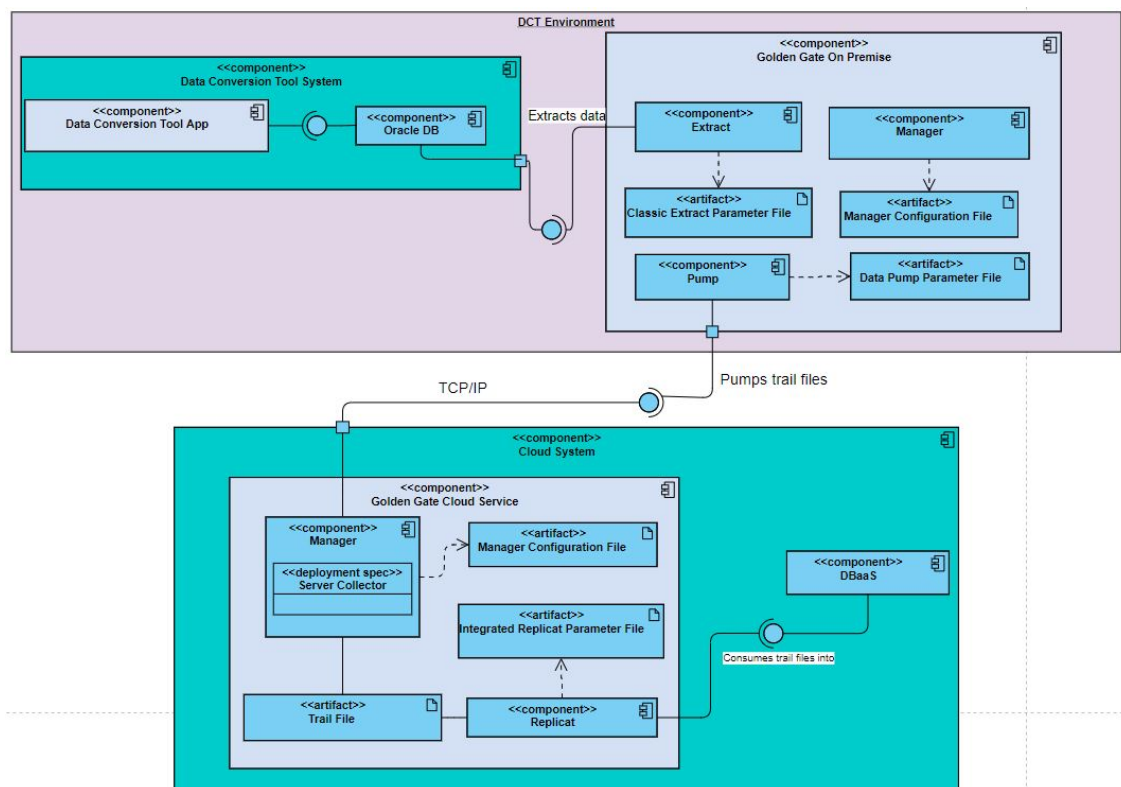
Figure 5.12: Golden Gate Classic/Integrated Deployment View

In the end, we are able to discern more easily all the steps that are taken by Golden Gate during a data migration into a DBaaS by checking the activity diagram represented in 5.13. Basically, Golden Gate on source environment connects to the database, extracts the data and sends it to Golden Gate on target environment. Then it establishes connection to the target database and loads the data. For the scenarios showed above, a partial and fully integrated migration was done. This means that in one case, Golden Gate was able to perform a classic extraction and then an integrated replication. But in the other case, it performed an integrated extract and replication. These scenarios will have a big impact in Oracle's Golden Gate performance that will be analysed in chapter 7.
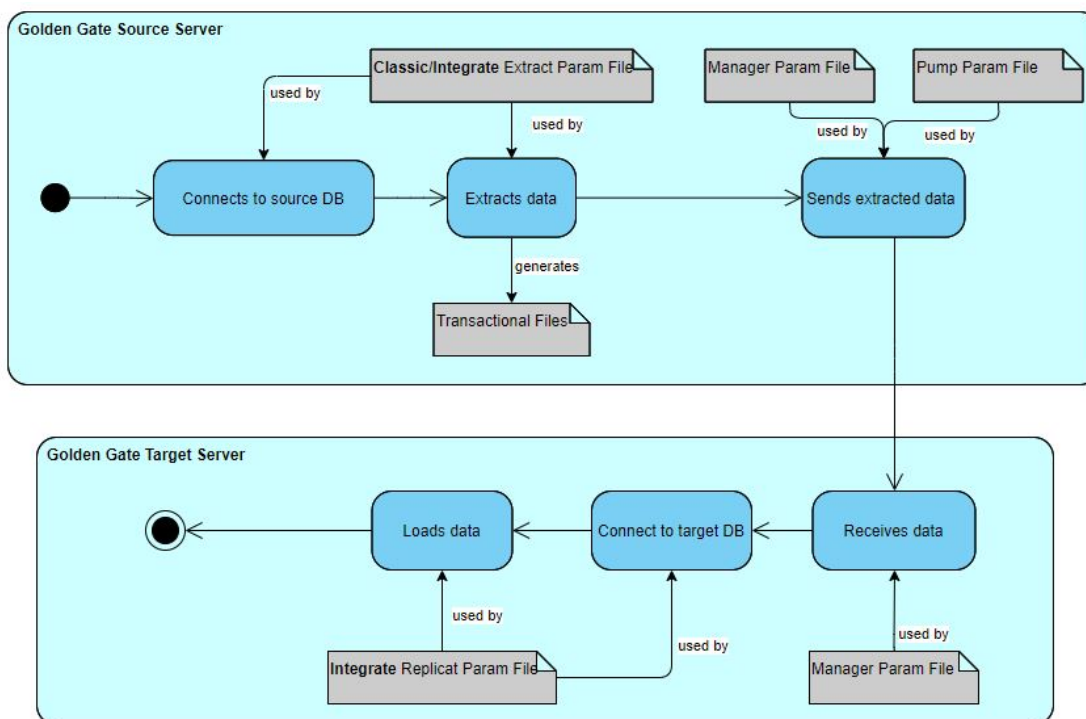
Figure 5.13: Golden Gate to DBaaS Activity Diagram

5.3   Cloud Integration with DCT using SQL Loader

For this section, the focus will be in describing the behaviour of the possible scenarios when using SQL Loader as our main tool to load data into an Autonomous Database or a DBaaS. As it can be seen in the following sub-section, it was only created one deployment view, because the process it's very similar and the main difference it's the target component, which is the database that's being used in each case.

5.3.1   Using SQL Loader to load data into Oracle Cloud

For the scenarios that use SQL Loader to load data into Oracle Autonomous Database and a DBaaS, we only need to focus in the target database that is being used. There is a particular difference, because the extra process of extraction that is performed by Golden Gate is going to be ignored, because SQL Loader is able to pick up the files directly from the DCT Server. In the figure 5.14 we are able to see how the SQL Loader consumes directly the files in the DCT Server, and then performs the loading directly to the target database. In this case, is important to know that the scripts generated by the DCT invoke SQL Loader, and then its able to pick up the correct files due to the information store in configuration files that were already explained in chapter 4. It's important to know how the scripts are generated in order to understand how the connection is established to the correct address, this will be detailed in chapter 6.
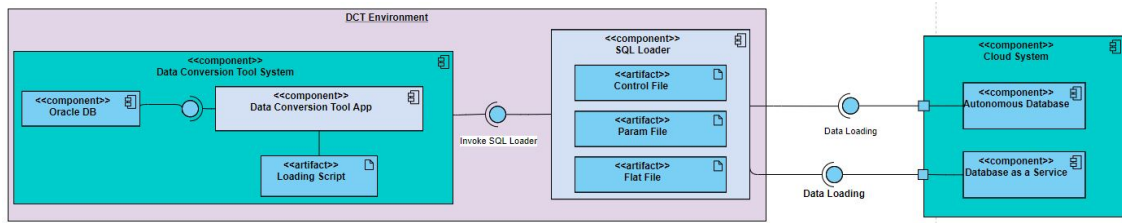
Figure 5.14: Deployment Diagram for SQL Loader Migration
into Oracle Cloud Services

In the activity diagram represented in figure 5.15 it's easy to understand how SQL Loader works, which files are are used by each process, when it comes to reading and writing data.
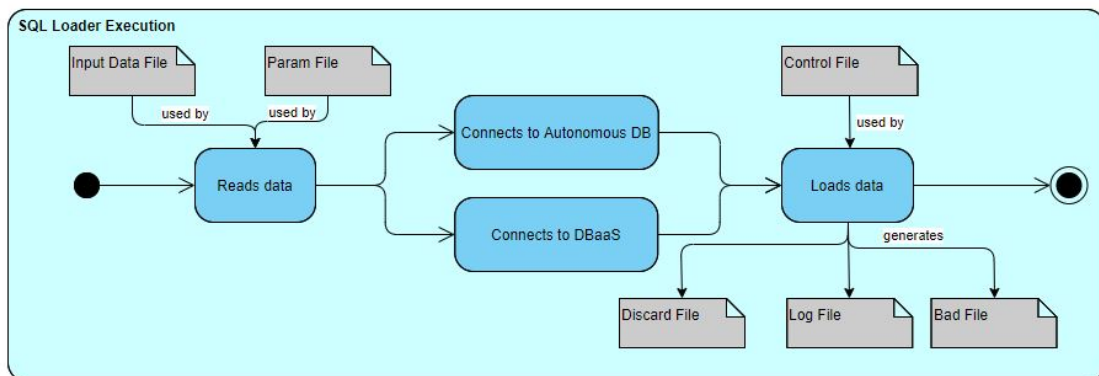
Figure 5.15: Activity Diagram for SQL Loader Migration into
Oracle Cloud Services

It's important to highlight that this integration is much more comprehensive and easier to understand due its low amount of steps, and Wipro already has been using a similar process to perform data migrations between on-premises. With the implementation of these scenarios using SQL Loader, we are going to be able to do two distinct comparisons:

- SQL Loader vs Golden Gate;

- Database comparison;

In short, it can be a viable option depending of its performance compared to Golden Gate scenarios that were explained previously.

Chapter 6

Solution Construction/Implementation

In this chapter, it's presented the construction of our solution. In order to have a good understanding of the steps that were taken to implement it, this chapter was divided into three sections that are:

- Cloud Databases Preparation;

- SQL Loader Implementation;

- Oracle Golden Gate Implementation.

The implementation is going to start with the preparation of the target databases in Oracle Cloud, in order to be ready to receive data from the source database. The last two sections are mainly dedicated to the implementation and configuration of the tools chosen in the previous chapters during the analysis and design of the solution. Therefore, two solutions will be presented, one that is already used by DCT, and it's mainly dedicated to small volumes of data, SQL Loader. On the other hand, that is more appropriate for large volumes of data, we will use Oracle's Golden Gate.

6.1 Cloud Databases Preparation

This section details the creation of distinct target databases. For our migration scenarios it will be created an Oracle Autonomous Database, improved for loading several transactions at the same time. As a secondary option, an Oracle Cloud Database Service instance (12c) will be created in order to see the difference between an Oracle DBaaS and an Oracle Autonomous Database.

6.1.1 Oracle Autonomous Database Setup

In the image below, 6.1, we can see the initial page of Oracle Cloud and the step that it needs to be followed in order to create an Autonomous Database.
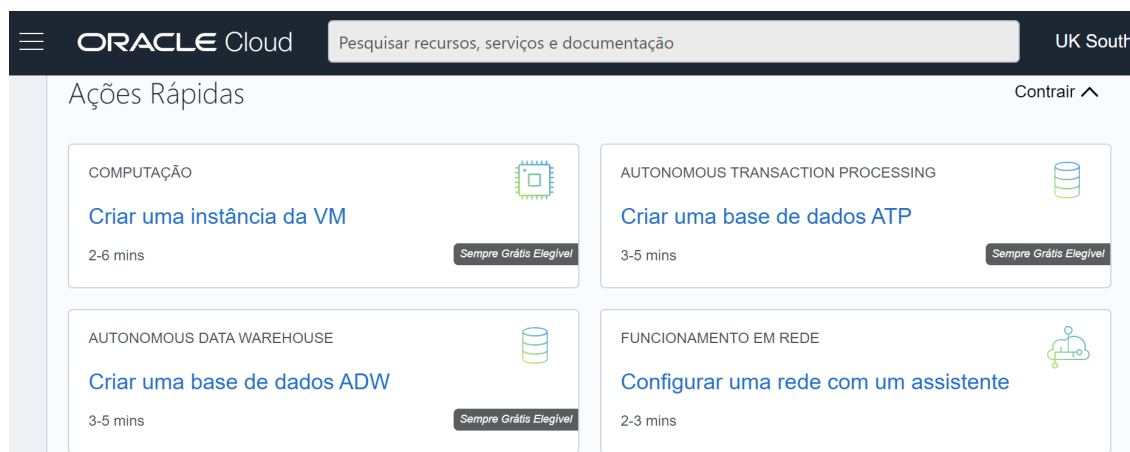
Figure 6.1: Oracle Cloud Initial Page

After we press "*Criar uma base de dados ATP*", we are redirected to the Autonomous Database creation page and it's required to give a presentation name, a database name and the transactions type of volume. In this test case, the best is to opt for transaction processing as it uses concurrency at its maximum taking advantage of DCT that uses several threads to load efficiently the data with SQL Loader.
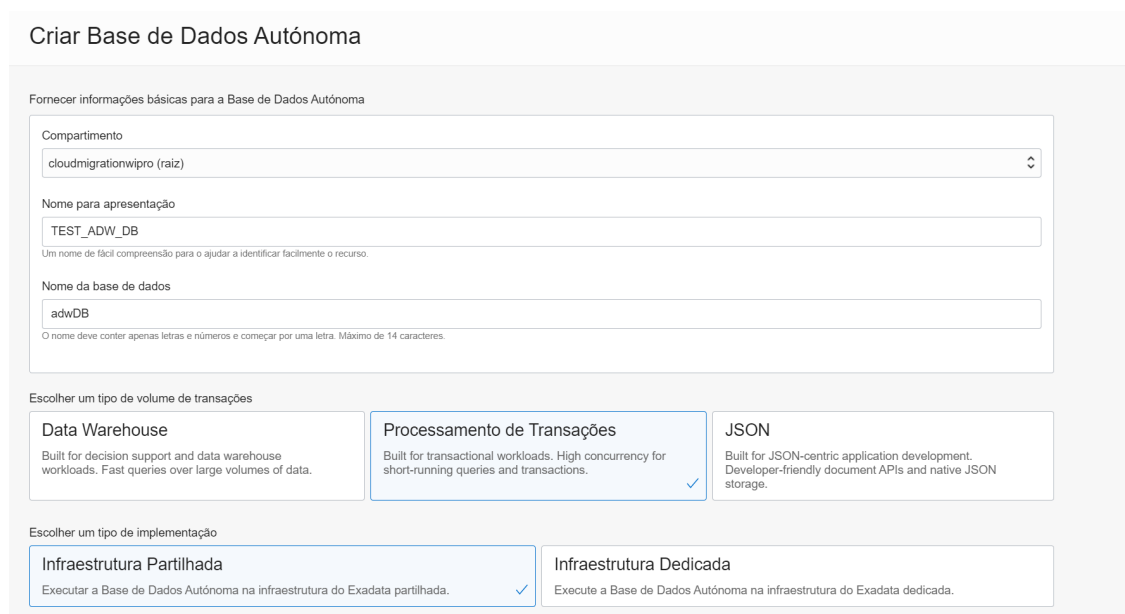
Figure 6.2: Autonomous Database Creation

All the details of the database are available, including storage size, number of Oracle computing units (OCPU) and the type of transactions it suits best. In the figure 6.3, every detail is specified, including connection details (wallet), service console, tools such as web SQL Developer and boards to have a better analysis of all the data being processed. It's important to highlight that the wallet contains the necessary info including credentials and security certificate in order to establish a connection to the autonomous database and load the data, but this will be detailed in the following sections while configuring DCT and Golden Gate.

Figure 6.3: Autonomous Database Details

6.1.2 Oracle Cloud Database as a Service Setup

After the creation of the autonomous database is successful we proceed to create our next testing database that will be a normal DBaaS. In the first scenario we have a self-securing and self-repairing database. We have full control and responsibility when it comes to managing the database with Oracle Autonomous Database Cloud, like a traditional solution. But the autonomous database is able to leverage advanced machine learning systems that apply updates, patches and simplifies the provision process. For this second scenario we will have a managed cloud database service available as a fee-based subscription that can save money. The database management and administration activities can be performed by the enterprise, which ends up being a good option for those companies wishing to outsource their database administration tasks (*Oracle Cloud Database vs On-Premises - Understanding the Differences* n.d.).The chosen database version will be 12c, the same as our source of migration. For this we will need to create a new database system that you can see in the image 6.4 displayed below.

Figure 6.4: Database as a Service

The next steps to follow are very similar to the ones showed in the previous subsection 6.1.1, where we choose the credentials, the volume and type of transactions that best fit

our database. With the databases all set, it's time to design one of our possible solutions when it comes to integrating DCT with Golden Gate in order to load into the Cloud. This will be detailed in the next section 5.2.

6.2   SQL Loader Implementation

SQL Loader tool has an important role in Wipro's current migration process, as we saw and analysed in chapter 5 in section 5.1.1. SQL Loader is the one responsible for loading data into staging/target tables. With this said, it's crucial to find out how to make use of the code already generated by DCT and be able to load data to an Oracle Autonomous Database and an Oracle DBaaS. Several steps need to be fulfilled, and to help our implementation, an Oracle's lab was used in order to accomplish this (Oracle 2019a).

6.2.1   Oracle Autonomous Database

First we need to download the wallet, that is saved with a .zip format, from the Oracle Cloud Infrastructure, from within our previously created autonomous database. Then we move the wallet to the DCT Server through WinSCP and then unzip it into the /wallets folder. The following command should be issued:

- $ unzip <your_wallet_file.zip> -d /wallet.

Then, to be able to connect to the target database, we need to copy the entries from the wallet's tnsnames.ora to the tnsnames.ora file that resides in our DCT Server. It's important to have the tnsnames.ora file configured correctly because this file defines all databases addresses we wish to connect to. This step is most commonly performed inside the following path, noting that it's being used an environment variable, ORACLE_HOME, that was defined in the server to ease the configuration:

- $ORACLE_HOME/network/admin/tnsnames.ora.

The next step is to copy the details from sqlnet.ora extracted file into the sqlnet.ora file in the server as was done in the previous step with tnsnames.ora. For that, we need to export the desired directory of the wallet and then replace the path value inside sqlnet.ora file as shown below:

- export TNS_ADMIN=$HOME/wallet;

- value "?/network/admin" replaced by $TNS_ADMIN);

When it comes to java applications, it's optional because it doesn't apply to our scenario, but it's only needed to uncomment the following lines in ojdbc.properties file:

- javax.net.ssl.trustStore=$TNS_ADMIN/truststore.jks;

- javax.net.ssl.trustStorePassword=Password1234(Password set when downloading wallet);

- javax.net.ssl.keyStore=$TNS_ADMIN/keystore.jks;

- javax.net.ssl.keyStorePassword=Password1234.

. After the connection is configured we should test the ADB (Autonomous Database) service using sqlplus. The below command allows to test the connection easily:

- $ sqlplus ADMIN@<ADB_Service_Name>;

. After the connection is tested successfully, we need to give permissions to every generated script using the following command:$ chmod +x loading_script.sh.

6.2.2 Oracle Database as a Service 12c

For Oracle's DBaaS a few steps have to be replicated from the above, including appending the details from OCI database connection to DCT tnsnames.ora file. After that, we need to pay attention to the values of of the variable TNS_ADMIN, because we will have two different cases, the migration into Oracle Autonomous database will use a wallet and the for the DBaaS will be by connection details of tnsnames file. So sqlnet.ora and tnsnames.ora files need to be carefully configured in order to avoid any issues while performing the data migrations. In the figure 6.5 it's described the configuration for the connection to DBaaS.

Long	cdb_ip_root	(DESCRIPTION=(CONNECT_TIMEOUT=5)(TRANSPORT_CONNECT_TIMEOUT=3)(RETRY_COUNT=3)(ADDRESS_LIST=(LOAD_BALANCE=on)(ADDRESS=(PROTOCOL=TCP)(HOST=10.0.0.11)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=DBaaS GG_lhr1tw.subnet.vcn.oraclevcn.com))) Hide Copy

Figure 6.5: DBaaS TNS Details

This configuration will be used when invoking the loading scripts, generated by DCT, to load data into the target database that resides in Cloud. The same strategy applies when loading data into an autonomous database too.

6.2.3 DCT Configuration

Now we need to configure our DCT in order to connect to the autonomous database without any issues because for now it's only prepared for on-premises migrations. In the figure 6.6 there are two distinct methods:

- run_sqlldr();
- run_sqlldr_dest().

. These methods are invoked in each loading kornshell for every par file passed as parameter, the first one is pointing to our staging/dct database, and the second one to the on-prem target database. The code shown is basically the configuration of the connection command that should be used when calling sqlldr.

```
# Function: run_sqlldr
# Sintax: run_sqlldr <par1> [<par2>...<parn>]
# Result: EXECUTES SQL LOADER statements
run_sqlldr()
{
sqlldr userid=$U$DBLINK/$P silent=header,feedback parfile=$* << EOF
EOF
}
#
################################################

#
################################################
################################################
#
# Function: run_sqlldr_dest
# Syntax: run_sqlldr <par1> [<par2>...<parn>]
# Result: EXECUTES SQL LOADER statements
run_sqlldr_dest()
{
sqlldr $UP silent=header,feedback parfile=$* << EOF
EOF
}
```

Figure 6.6: DCT Methods

All the user ids and wallets are described in the figure 6.7 below, and this information is
present in the mig_oracle_setup kornshell. This script is stored in the DCT Server, and it
has the main parameters of its configuration. For our on-prem targets, due to database lack
of permissions, our Tech team is responsible for generating wallets to ease our connection
to the database with the necessary permissions. As for our staging database, we use the
MIG_USER and MIG_PASS and the db link that is defined in our tnsnames.ora file present
in the server. With this data we are able to fulfill without any issues our current process of
migration.

```
######################### DB USERS ##############

export MIG_USER=DCTDC
export MIG_PASS=dct4n4DC

# Connection configurations
export U=$MIG_USER
export P=$MIG_PASS
export DBLINK=@RETAIL

# DCT Repository Schema
export MIG_DB=DCTDR

# Client Name (Optional)
export CLIENT=WIPRO_RETAIL

# Oracle Wallet RMS
export TNS_ADMIN=/home/dct/.wallet
export UP=/@dsRMSAlias2
export UP_PS=/@dsRMSAlias2
```

Figure 6.7: DCT Users and Connections

But in order to enhance our DCT to be able to accomplish a data migration into the Cloud we will need to add a new user. This user should have all the necessary permissions defined in the autonomous database, focusing on DDL and DML operations to save data into the database. The db link will be the one extracted in the beginning of our process, that is present in the downloaded wallet from Oracle Cloud. Below are defined the new user and db link as needed, they can be seen in 6.8

```
######################### DB USERS ##############

export MIG_USER=DCTDC
export MIG_PASS=dct4n4DC
export MIG_CLOUD_USER=ADMIN
export MIG_CLOUD_PASS=Password1234
export MIG_CLOUD_DBaaS_USER=sqlloader
export MIG_CLOUD_DBaaS_PASS=oracle_ORA_123

# Connection configurations
export U=$MIG_USER
export P=$MIG_PASS
export DBLINK=@RETAIL

]#ADW Configurations
-#DBLINC in tnsnames.ora downloaded from wallet
export UC=$MIG_CLOUD_USER
export PC=$MIG_CLOUD_PASS
export DBLINKC=@adwdb_medium
]#DBaaS Configurations
-#Prepare tnsnames.ora with connection too
export UCS=$MIG_CLOUD_DBaaS_USER
export PCS=$MIG_CLOUD_DBaaS_PASS
export DBLINKCS=@140.238.65.41:1521/DBaaSGG_lhr1tw.subnet.vcn.oraclevcn.com
```

Figure 6.8: DCT User and Connection for Cloud

Our method to load the on-prem target suffered little changes to have room for possible data migrations to the Cloud. For that we make use of the interfaces, defined in the DCT as a representation of business areas, giving a specific interval for the ones that will be going to the Cloud and the ones that aren't. In the image 6.9 the interval is between 9100 and 9200 but it can take any values depending of the number of targets that the project requires. So, by doing a simple filter by interface passed in the name of the parameter we can proceed with an on-prem or cloud loading. 6.9

```
# Function: run_sqlldr_dest
# Syntax: run_sqlldr <par1> [<par2>...<parn>]
# Result: EXECUTES SQL LOADER statements
run_sqlldr_dest()
{
interface=$(echo "$1" | sed 's/_T.*//' | sed 's/.*_I//')
echo $interface
if [[ $interface > 9100 && $interface < 9200 ]];
then
    echo "For Cloud ADW Target"
    sqlldr $UC$DBLINKC/$PC silent=header,feedback parfile=$* << EOF
EOF
else if [[ $interface > 9200 && $interface < 9300 ]];
then
    echo "For Cloud DBaaS Target"
    sqlldr $UCS$DBLINKCS/$PCS silent=header,feedback parfile=$* << EOF
EOF
else
    echo "For On Prem"
    sqlldr $UP silent=header,feedback parfile=$* << EOF
EOF
fi
}
```

Figure 6.9: DCT User and Connection for Cloud

The loading kornshells invoke the run_sqlldr() or run_sqlldr_dest() methods depending of the interfaces, that could match staging or target areas. In the figure 6.10 we can look at the code to have a better understanding of what happens for each interface.

```
# ================================= SEASONS =====================================
# Processing LOADING [SQL*LOADER] Script (mig_I9988_Tseasons_load.par)
if [[ -f "mig_I9988_Tseasons_load.par" ]]; then
    # Update processing flag
    proc_ind=1

    # Forcing temp files removal (if they exist!)
    rm -f mig_I9988_Tseasons_load.par.log
    rm -f mig_I9988_Tseasons_load.par.bad
    rm -f mig_I9988_Tseasons_load.par.dsc

    echo "[ RUNNING SQL*LOADER PROGRAM USING (mig_I9988_Tseasons_load.par) ] ..."
    run_sqlldr_dest mig_I9988_Tseasons_load.par
```

Figure 6.10: Loading Kornshell Example

Now that the configuration is complete, it's important to emphasize that this process should be taken into consideration when loading small amounts of data. That's why we are reviewing another possible approach with Golden Gate in the following section 6.3 which is recommended for large amounts of data.

6.3 Oracle Golden Gate Implementation

To start we need to do have the following details ready and prepared as pointed:

- Setup Oracle Database 12c On premise for Golden Gate;

- Oracle GoldenGate On Premises installation/configuration;

- Configure Oracle Autonomous Data Warehouse and DBaaS;

- Configure Oracle Golden Gate Cloud Classic/MS Service for 12c Instances and ATP.

All these points are detailed in the chapter 5, in order to understand the flow of the Golden Gate process during a data migration, and to see how every component communicates between themselves. The first two steps are relevant to our migration process when it comes to the source migration environment, and we need to keep in mind that the installation of Golden Gate needs to be done in the same machine that has the database installed. Several steps will be followed as described in (Oracle 2018b) and (Class n.d.). The version of Golden Gate is going to be 12.3.0.1.2 or later as they support delivery to Cloud.

6.3.1   Source database - Oracle 12c

For this section, a simple installation of an Oracle database 12c was done in the personal machine in order to ease any configurations needed when it comes to administrator privileges. The following documentation (Oracle n.d.[a]) was followed to setup the database. Then a user was configured, in order to create the dummy tables referenced more ahead in the next sections. Noting that this was installed as non container database as it's the type that allows classic extracts/replicates, which is supported by our Autonomous Database too. For other cases, it's preferable to have a CDB(container database) as it can host several pluggable databases making resource management more efficient.

6.3.2   Database as a Service Configuration for Replication

Starting with the Oracle DBaaS, we need to create an Ingress Network Rule when it comes to the port that is being used, to be able to connect to the database using SQL Developer. The rule is described in the picture 6.11, where the port is being open for all addresses. For this action was performed using the OCI user interface as it was more intuitive.

Figure 6.11: Network Rule

Now, connected to the database we need to set some parameters in order for the communication with Golden Gate to work correctly without any issues. For this configuration the following steps were performed:

- Enable system to be able to perform a golden gate replication;

- Create target user with default tablespace users;

- Grant target user with connect, resource and create table;

- Create target golden gate admin user with unlimited tablespace;

- Grant dba to admin user;

- Run procedure dbms_goldengate_auth.grant_admin_privilege('oggadmin','apply').

Now both users are ready, and we can create our target tables, and the golden gate user has enough privileges to perform the migration from one database to another when required. In order to have a better understanding of this step, all the scripts used are displayed in the appendix B, which contain the necessary info when it comes to the configuration of the users and source/target database.

6.3.3   Oracle Autonomous Data Warehouse Configuration for Replication

For this case we need to configure our Oracle Autonomous Database in Cloud to be able to use Oracle GoldenGate On Premises and perform the tasks as described below:

- Configure Oracle Autonomous Database for Replication;

- Unlock the user ggadmin in Oracle Autonomous Database;

- Prepare Database (schema and target tables);

- Create new application user user_target.

The Oracle Autonomous Database already has a pre-existing user created for Oracle GoldenGate On Premises called that is called ggadmin. The ggadmin user needs to be granted the right set of privileges for Oracle GoldenGate On Premises Replicate to work. By default, this user is locked. To unlock the ggadmin user, connect to your Oracle Autonomous Data Warehouse Cloud database as the ADMIN user using any SQL client tool. First we need to run the alter user command to unlock the ggadmin user and set the password for it, and then we can create a new application user called goldengatecloud in this case, and give him the necessary grants to prepare the database. (Oracle 2018b).

- alter user ggadmin identified by password account unlock;

- create user goldengatecloud identified by password;

- grant create session, resource, create view, create table to goldengatecloud.

Then we change to our recently created user, goldengatecloud, and we can create the replicate tables that are needed for the test cases as it's shown in the figure 6.12.
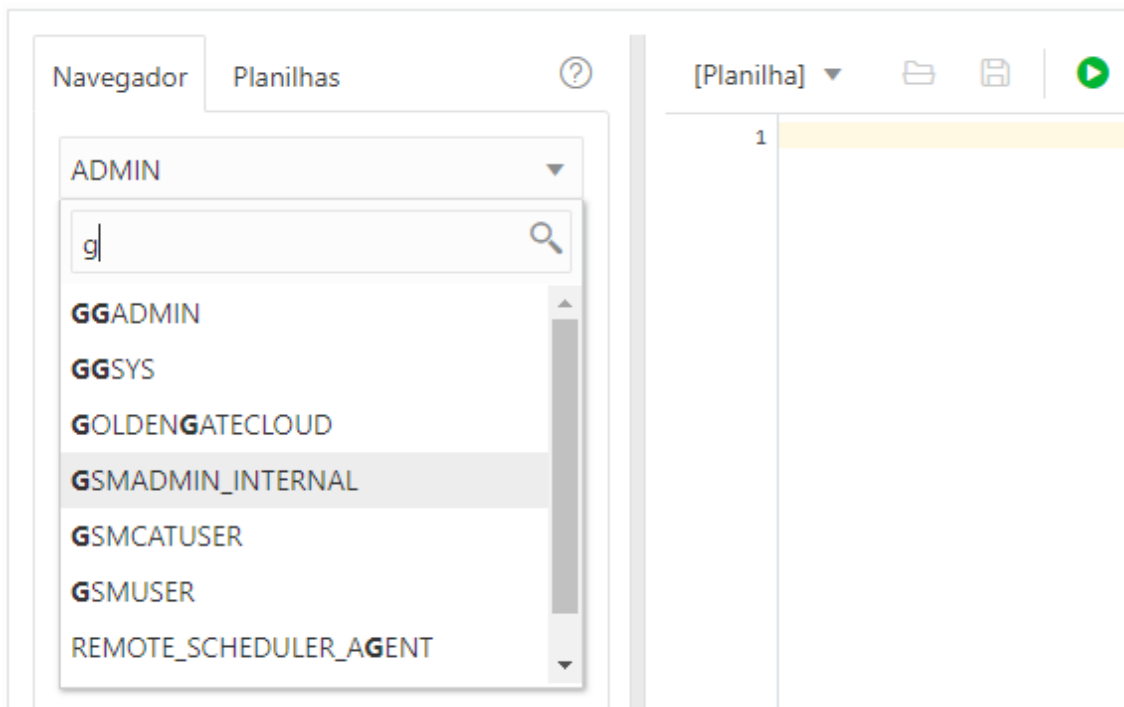
Figure 6.12: User GoldenGateCloud

For every test case, we are going to focus only in one table, for example the Account table, rounding up to one million records, to see and analyse its behaviour when working with a huge amount of records. The performance will be compared to the use of SQL Loader and Golden Gate Classic/MS Cloud Service when replicating into Oracle 12c databases in Cloud. The configuration of the tables can be seen in the "replicate_tables_creation.sql" file. As we saw before in the case of SQL Loader, all the details of the Autonomous Database are needed, and they are available in the wallet that it's downloaded easily from our database instance page 6.13. There are several important files such as ojdbc.properties sqlnet.ora and tnsnames.ora that are essential to establish a secure connection. These last two files, sqlnet.ora and tnsnames.ora, are required to configure Oracle GoldenGate On Premises to work with Oracle Autonomous Database.
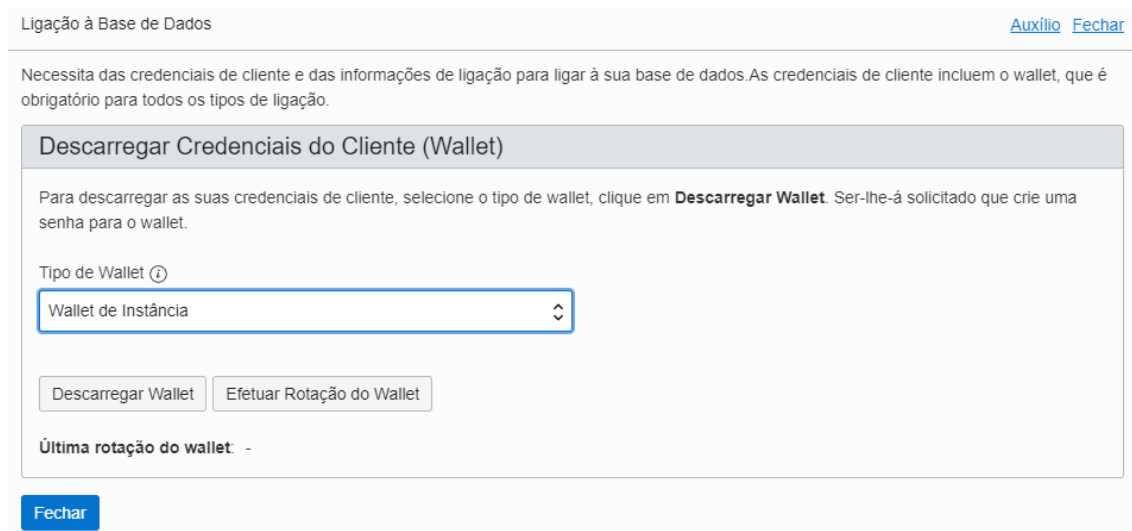
Figure 6.13: Wallet Download

6.3.4 On premise Golden Gate version 19

The application of Golden Gate needs to be installed of preference in the source and target database location. So in this case, we installed it within the same $ORACLE_HOME as our source database (12c). For this we just have to follow the steps described in the installation guide that comes with the download app. But the configuration of its parameters is crucial for a good performance of our data migration. Golden Gate has different types parameters that need to be configured according the specifications of our migration and the databases that are going to be used in it. In the list below are the respective files:

- Extract;
- Pump;
- Replicate;
- Manager;
- Globals.

To take a closer look, these files can be seen in the directory ogg_files if needed with simple configurations that were applied in our first scenario that involves the Autonomous

Database. In order to have a better understanding, an overview is going to be done for each with an image representing the flow. For our golden gate source setup we need to configure always three files that are: extract, pump and manager. In a simple extract file, we need to setup a group for it, the database user, a directory for extraction trail files, report timer (optional) and finally the tables that are going to be migrated, below it's a list of commands:

- extract ext1;

- userid user_src, password user_src_password;

- exttrail ./dirdat/in;

- reportcount every 10 minutes, rate;

- table user_src.* (this will extract all tables in user_src schema).

Keep in mind that this process implemented in the DCT environment needs to have an extra configuration, as we are using the repository schema where the last two columns are related to validations and we don't need to extract them. We need to append to our **table user_src.* the command colsexcept(col1, col2)**, this will save up time and enhance our performance during the migration. For the pump file, as for the extract file, we defined a group, the database user, passthourgh command to ignore table name validations, the remote host (golden gate cloud service public address) and the its port, the rmttrail path that is going to receive the files extracted in "ext1" by TCP/IP and finally we define the tables to be migrated again as we can see:

- extract ext1pump;

- userid user_src, password user_src_password;

- passthru;

- rmthost 132.145.66.153, mgrport 7809;

- rmttrail /u02/trails/pn;

- table user_src.*.

After understanding the concepts of the extracts and pumps, we just need to run as administrator the golden gate executable and then we start the manager, add the respective extracts and pumps with the correct paths for each one and register/attach the extract to the non-CDB or PDB if required. When the replication is required, we just need to activate/start each process while the manager is running and we can check the performance and statistics of the loading. These final steps are described in the list below:

- add extract ext1, tranlog, begin now (integrated keyword can be added depending of the cases);

- add extract ext1, integrated tranlog, begin now;

- add exttrail ./dirdat/in, extract ext1, megabytes 50;

- add extract ext1pump, exttrailsource ./dirdat/in;

- add rmttrail /u02/trails/pn, extract ext1pump, megabytes 50;

- add trandata <schema>.<tables>;

- register extract ext1 database add container(<pdb_name>);

- register extract ext1 database container(<pdb_name>);

- start mgr;

- start ext1;

- start ext1pump.

In the end is just missing the manager configuration, that will operate the processes configured above and report any issues that happen while the migration is performed. This setup is very similar to the one installed in the Cloud, we need to setup a port and a dynamic port list to ease the connection in case any other port is already being used in other operation. An access rule needs to be added in order to allow other servers to do inbound communications, very important if we are using Golden Gate to do bi-directional updates/refreshes. The other commands are optional and were added in order to ease the management of the server when it comes to storage, and a fast process in case any operation fails to restart automatically.

- port 7809;

- dynamicportlist 7810-7815;

- accessrule,prog *,ipaddr 132.145.66.153,ALLOW;

- autostart extract *;

- purgeoldextracts ./dirdat/in*, usecheckpoints, minkeephours 1;

- autorestart er *, retries 3.

6.3.5 OCI - Golden Gate Classic

In this case we will do the configuration for Oracle GGCS do be able to communicate with Oracle Golden Gate On Premises, Oracle DBaaS 12c and Oracle Database 12c on premises. We will follow a certain sequence of steps that are described in (Oracle n.d.[c]):

- ssh -i <private-key-filename> opc@<public-id-address>;

- alter sqlnet.ora and tnsnames.ora files to have connections to source and target databases in '/u02/deployments/<deployment_name>';

- add credentialstore to connect to DBaaS;

- alter credentialstore add user oggadmin@migration.subnet.vcn.oraclevcn.com:1521/DBaaSGG_lhr1tw password oracle_ORA_123 alias oggadmin123 domain OracleGoldenGate;

- test connection with "ggsci > dblogin useridalias oggadmin123 domain OracleGoldenGate";

- Repeat steps above to establish connection to the source database.

For the source database in specific do the following:

- ggsci > dblogin useridalias <alias_name> domain OracleGoldenGate;

- add trandata <schema>.<table>.

.

For the target database in specific do the following:

- $ cd /usr/local/bin $ ./ggsci oracle<databaseversion>;

- dblogin useridalias oggadmin123 domain OracleGoldenGate;

- "ggsci > add checkpointtable <schema>.checkpointtable".

.

For this golden gate application deployment we need to configure at least two files that are: replicat and manager. In a simple replicat file, we need to setup a group for it, the database user, report count (optional) and finally the tables that are going to be migrated with the keyword "map", below it's a list of commands:

- replicat repdw;

- useridalias ggadminalias;

- reportcount every 5 minutes, rate;

- map user_src.*, TARGET goldengatecloud.*.

. This setup for the manager is the same as described before for the source environment. The list of commands is displayed below just in case:

- PORT 7809;

- dynamicportlist 7810-7815;

- accessrule,prog *,ipaddr 132.145.66.153,ALLOW;

- purgeoldextracts /u02/trails/pn*, usecheckpoints, , minkeephours 1.

As explained for source environment. now we just need to run as administrator the golden gate cloud service passing as parameter the version of the target database (i.e. oracle19) and then we start the manager, add a checkpoint table into ggadmin schema of our target database in order to control the flow of the migration, this step requires dblogin from golden gate console. Then the replicat is added to the target database, notice that can be several types of replicats, below are three main options: Classic, Parallel and Integrated. When the replication is required, we just need to start the replicat process while the manager is running, extracts in source must be running or else there won't be any transactions to be processed. These final steps are described in the list below:

- add checkpointtable oggadmin.ckptab;

- add add replicat repdw, exttrail /u02/trails/pn, checkpointtable oggadmin.ckptab;

- add add replicat repdw, parallel, exttrail /u02/trails/pn, checkpointtable oggadmin.ckptab;

- add replicat repdw, integrated, exttrail /u02/trails/pn, checkpointtable oggadmin.ckptab.

To finish this chapter, we can conclude that both solutions have been successfully implemented and can be applied, if this is what Wipro desires. They share different levels of complexity when it comes to the configuration of each one but they have the same ultimate goal, which is to load the data to the target database hosted in the Cloud. The integration

with DCT ends up being easier while using SQL Loader, due to its use in on-premises migrations in Wipro. So it was more of a direct implementation, which focused on changing the basic configuration of the DCT regarding the execution of the generated scripts. Not forgetting that all the connection details used during the loading needed to be configured according the service details provided by Oracle Cloud. Regarding Oracle Golden Gate tool, its integration was much more complicated and demanding, not so much for the technical part, although it has several steps to be performed, but more for the knowledge that had to be acquired overall. Because its integration with DCT it's only possible when its inner migration process is well understood. For this case, a deep study and investigation was done in order to achieve the best possible solution. In the next chapter, 7, each solution is going to be evaluated in order to check its performance, and its behaviour will be analysed when migrating data into the databases referred in the beginning of this chapter.

Chapter 7

Solution Evaluation

For the problem that is being addressed in this thesis, we can highlight several different approaches to obtain a complete and detailed assessment of the solution. The main concept that will be subjected to an evaluation is the final migration process, that is divided into two distinct approaches:

- Migration with DCT using SQL Loader;

- Migration with Golden Gate.

Two different processes will be used in order to make a good and concise evaluation of these procedures. First a QEF scenario will be designed and created in order to understand correctly the raised requirements and have an accurate assessment/measurement of compliance percentage of each one, taking into consideration their relevance. Noting that a comparison will be done between the possible solutions to be used: DCT and Golden Gate. Then, hypothesis testing will be done in order to do a good analysis of DCT plus SQL Loader and Golden Gate loading capabilities into Oracle Cloud services. These tests will rely in all the data gathered during migrations scenarios that are present in appendix A. The evaluation depends upon the parameterization of:

- Oracle Autonomous Database vs Oracle DBaaS;

- Golden Gate configurations;

- SQL Loader vs Golden Gate.

Finally a comparison with traditional on-premise migration will be shown.

7.1 Quality Evaluation Framework

To set up a good QEF quality scenario for the Golden Gate solution, first it's essential to define correctly the required dimensions and factors that in this case will be:

- Functionality;
    - Functional;
    - User Interaction.
- Adaptability/Consistency;
    - Versatility;

      – Pedagogical Aspects;

      – Accuracy;

      – Content Quality;

      – Maintenance.

- Relevancy.

      – Visual Quality;

      – Navigation.

For each one of these dimensions, are defined several factors with a certain weight, and each factor will have the necessary requirements associated to it with a fulfilling percentage (Asif Laghari 2018). It's important to highlight that these dimensions represent different areas of action. The first one, **Functionality**, represents all the functionalities that Golden Gate performs, including extraction and loading of data for example, and how the interaction with the user it's done, when it comes to process or permissions configuration. Then, we have **Consistency**, that involves concepts like:

- Cloud Compatibility;

- Content Quality;

- Maintenance.

This dimension is crucial to check how versatile Golden Gate is, checking its reliability when a process is interrupted, if the log files displayed during its execution are helpful and if it's compatible with other cloud computing services than Oracle's. Finally, there is the **Relevancy** dimension, that represents the user experience when navigating through Golden Gate system, which focus points like:

- Intuitive User Interface;

- Responsible User Interface;

- Displaying methods;

- Run-time errors treatment.

The Golden Gate QEF document is represented and detailed in the appendix D and every concept is referenced in (Thion et al. 2007) and (T N and Hegadi 2013).

7.2   Research hypothesis specification

A hypothesis test may or may not support a null hypothesis (H0), compared to an alternative hypothesis (H1). A null hypothesis generally describes a situation where there is no difference between two evaluations (Barsukov 2015). In this case, the following null hypotheses were defined:

- "There is no significant difference in the loading time between the SQL Loader and the new golden gate integration";

- "There is no significant difference in the loading time between a classic or integrated configuration on Golden Gate";

- "There is no significant difference in the loading time between an Autonomous Database (non-CDB) or simple Oracle DBaaS (PDB)".

In the hypotheses that were defined, it was established that the null hypothesis is rejected if p-value is less than 5% having a confidence level of 95%. In more detail, and performance related, hypothesis testing is a great way to compare and analyse the loading speed of our Data Conversion Tool generated scripts with Golden Gate application, using the same sets of data in each scenario. The evaluation will be based in two different scenarios when it comes to technologies that are:

- Using Oracle SQL Loader directly from the scripts generated by DCT;

- Using Oracle Golden Gate.

And for each one of these tools we will see which database is faster while all the DML's and DDL's commands are processed for insertion of data. This will allow to restrict our solution into a good third party application if needed and a recommended database in order to improve overall performance of our data migrations, and any retail solution that is implemented in the future. Noting that these evaluations can comprise numerous types of tests to be performed. But for the scenarios specified above, the best metrics to be used are: performance and efficiency through time. For specific enhancements regarding the complete migration process, we could go deeper and compare even other tools mentioned in the chapter 3 and other Cloud services to see which ones could respond better to our necessities in terms of time and quality, but for now the scope will be restricted to the tests mentioned above. Finally, in order to explain and discuss correctly part of the problem, these three hypothesis will be tested in the form of a proof of concept focusing the performance/time metric. The program R will be used to make the necessary statistical calculations and tests with the obtained sets data. In the Figure 7.1 it's possible to understand all the scenarios that need to be evaluated taking into consideration the restrictions described in section 5.2.1 in chapter 5, which lead us to the hypotheses referenced above.

Method	Size	Nº of tables	Source DB (On Premises)	Source DB (On Premises)	Target DB (On Cloud)	Target DB (On Cloud)
			DB-1	DB-2	DB-1	DB-2
			Non-CDB (Oracle 12c)	PDB (Oracle 12c)	Non-CDB (Autonomous DB)	PDB (Oracle 12c)
Golden Gate Process			Classic	Integrated	Classic	Integrated
DCT + Golden Gate (extraction + loading)	1M	1 table	X		X	
	1M	1 table	X			X
	1M	1 table		X		X
DCT + SQL Loader (loading)	1M	1 table	-		X	
	1M	1 table	-			X

Figure 7.1: Evaluation Scenarios

7.3   Description of the evaluation methodology

Improving program performance is the process of changing the software system so that it works more efficiently and faster. This improvement can be made by monitoring and analyzing the performance of an application, identifying ways to improve it. Performance optimization can be performed at several levels, in this case the focus will be on the new data migration processes, which includes the DCT and Golden Gate. When measuring execution times, it is necessary to take into account certain points: data sets that follow normal distribution are unlikely, and their analyses are often not appropriate due to their small size. Analysis of big samples is recommended as it is more robust and can cover more problems, in this case it will round one million records for each sample, being repeated for

at least 20 times. Regarding statistical tests, the size will be equal to or greater than 20 and it is assumed that in both cases, the execution times will be close to an average value, presenting a normal distribution, allowing a more robust analysis. As such, it was decided to use the parametric t-test to compare the two samples in each case, and the ANOVA test if the possibility of adding more samples is provided. In the case of the ANOVA test, samples must obey a normal distribution, otherwise Friedman non-parametric test will be used (Kassambara n.d.). In the scope of testing two independent variables followed by the consequent t-test, the next steps should be taken into consideration (Soetewey n.d.):

- Get a sample of 20 executions of a given independent variable, loading the same data into each specific database with Golden Gate and SQL Loader, writing down the respective time for each attempt;

- Run Shapiro-Wilk test to confirm that the samples follow a normal distribution;

- Otherwise, 10 more samples (total of 30) will be obtained and the parametric test is carried out;

- Execute t-test;

If the test includes more than two variables (Soetewey n.d.):

- Obtain a sample of 10 executions of a given independent variable, generating/loading the same data, writing down the respective time for each attempt;

- Run Shapiro-Wilk test to confirm that the samples follow a normal distribution;

- Otherwise, if normal distributions are not respected, the Friedman test should be used instead of ANOVA;

- Check that the sample groups are mutually independent through the chi-square test;

- Do the Levene test to check whether the population variances are homogeneous;

- Execute ANOVA/Friedman test;

- Can check where the difference relies with Tukey post-hoc / Nemenyi tests.

7.4   Preparation of the evaluation

7.4.1   Golden Gate Samples

So following the steps described above, all the samples are in document referenced in the beginning of this chapter, and we will start with the comparison of Golden Gate procedures and the different types of databases used as they are implicit with each other. In order to verify the normal distribution of each sample taken for each use case of Golden Gate, we make use of Excel and R Studio to perform the tests described above and to create the graphs. This first sample, represented in figure 7.2, makes reference to the migration scenario from a non containerized database to a database as a service, by looking to the values obtained is safe to say that it follows a normal distribution as we can see below, and the metric here was time in seconds.

Figure 7.2: Golden Gate - Scenario 1

For the second sample, in 7.3 makes references the migration scenario from a pluggable database (CDB) to a database as a service, and we can say that it follows a normal distribution too, in this case the metric used was the same as the first sample.

Figure 7.3: Golden Gate - Scenario 2

Finally for the third sample that references the migration scenario from a non containerized database to an autonomous database, it follows a bit of a skewed distribution as we can see

in 7.4. In order to ensure it obeys a normal distribution, a Shapiro-Wilk test was added that is represented in figure 7.5. The metric used was time in minutes because it has a huge difference of performance related to the other test cases.

Figure 7.4: Golden Gate - Scenario 3

Below it's safe to say that our third sample follows a normal distribution and is ready to be used in a t-test because the value of p-value is superior to 5% what means that we can't reject the null hypothesis and the evaluation can be proceeded with all the samples.

Figure 7.5: Shapiro Test

7.4.2   SQL Loader Samples

As was done with Golden Gate samples, all the samples related to SQL Loader are in the document referenced in the beginning of this chapter, and we will start with the comparison of using SQL Loader to load data into different types of databases. In order to verify the normal distribution of each sample taken for each use case of SQL Loader, we make use of R Studio and Shapiro-Wilk test. This sample makes reference to a migration scenario from a pluggable database (PDB) to a database as a service and follows a normal distribution as we can see in the image 7.6, and the metric used here was time in minutes.

DBAAS LOADING

Figure 7.6: SQL Loader - Scenario 1

For the second sample that references the migration scenario from a pluggable database (PDB) to an autonomous database, which follows a normal distribution too, the metric used was the same as the first sample.

AUTONOMOUS DATABASE LOADING

Figure 7.7: SQL Loader - Scenario 2

7.5 Evaluation Analysis

The evaluation will be divided in three analysis due to the use of different procedures and applications to load data into Oracle Cloud. Different databases were used to fulfill all the migration scenarios described initially in this chapter and in chapter 6. In short the analyses divide into the following sets:

- Golden Gate Process Analysis;

- SQL Loader versus Golden Gate Performance;

- Database Performance Analysis with SQL Loader.

In the end, a little reflection is made about the designed QEF related to Golden Gate application.

Golden Gate Process Analysis

For the first scenario, we are going to evaluate the differences between using a Classic or an Integrate Replicat process with Oracle Golden Gate. As an Integrate process is usually more efficient, this will be correctly tested while loading data into the Autonomous Database and DBaaS. Because the Autonomous Database it's a non-container database and does not support integrated replicats, the data will have to be loaded by using a classic replicat. Meanwhile the DBaaS will be using an integrate replicat. Below in figure 7.8 the results can be measured, and we can check that the p-value is inferior to 5% having a confidence interval of 95%. This means that there is a huge discrepancy between the times of loading of each scenario, so for this analysis it's safe to assume that it's more reliable and efficient to use an integrated process than the classic one. If we pay more attention to the samples, it's important to denote that difference is rounded to 25 minutes between them, which in a migration it's very relevant because we are talking about many millions of records.

```
        Welch Two Sample t-test

data:  dataMin$Non_CDB_DBaaS and dataMin$Non_CDB_ADW
t = -53.066, df = 19.039, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -25.57446 -23.63387
sample estimates:
mean of x mean of y
  0.45500  25.05917
```

Figure 7.8: T Test for Non CDB Migration to Non CDB Autonomous Database

The secondary case that can be analysed is the comparison between a full integrated process and a partial one (classic extract and integrated replicat), with Oracle Golden Gate. For this it will be used an Oracle 12c database non containerized and a plugable one to check the behaviour of the different scenarios. In the figure 7.9 the p-value is superior to 5% what implies that we can't reject the null hypothesis and the difference between samples is not significant and all the values are very similar. With this result, it's possible to say that

having a classic or integrate extraction is not relevant for our migration using Golden Gate application.

```
        Welch Two Sample t-test

data:  dataMin$Non_CDB_DBaaS and dataMin$PDB_DBaaS
t = -0.67243, df = 35.046, p-value = 0.5057
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.06698182  0.03364849
sample estimates:
mean of x mean of y
0.4550000 0.4716667
```

Figure 7.9: T Test for Non CDB and PDB Migration to Non CDB Autonomous Database and DBaaS

SQL Loader versus Golden Gate Performance Analysis

For this case is better to show a diagram of all times processed during every test for each possible scenario loading data into Oracle's Database as a Service, that in this case use:

- SQL Loader;

- Golden Gate.

In the following bar graph it's represented the mean times for each migration scenario to be executed. To be noted that when executing migrations with SQL Loader method, no division by threads or parallelism is being used, so it can be enhanced in order to ease future migrations. After a good analysis of the figure 7.10, two comparisons are essential in order to obtain the method that best suits a migration into Oracle's Autonomous Database or a DBaaS.

Figure 7.10: Graph with mean time for each scenario

For this we had to compare the scenario of using SQL Loader and Golden Gate to load data into DBaaS, and as a second one compare the same tools while loading to the Autonomous Database provided by Oracle. In the first scenario there is a huge gap between SQL Loader and GG when loading into a DBaaS that sums up to at least 30 minutes which is very relevant in a migration. And the results of the t-test in figure 7.11 ensure this because the p-value it's very low below 5% which allow us to reject the null hypothesis of not existing a difference between those two tools when migrating data into a database as a service. Emphasizing that this conclusion can only be drawn, because although Golden Gate performs more operations as shown in the chapter 5, it is capable of being more efficient than SQL Loader having a much shorter loading time.

```
        Welch Two Sample t-test

data:  dataComp$SQLLDR_2_DBAAS and dataComp$GG_PDB_DBaaS
t = 112.04, df = 19.193, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 30.81788 31.99045
sample estimates:
 mean of x  mean of y
31.8758333  0.4716667
```

Figure 7.11: T Test for SQL Loader and GG procedures to load
DBaaS

In the second scenario the gap between SQL Loader and GG when loading into the Autonomous Database is not so significant as in the previous scenario, and rounds to 5 minutes

more and less. But to ensure if there is a relevant difference, we need to proceed with the t-test. And the results of the t-test in figure 7.12 ensure yet again the same conclusion as before because the p-value obtained it's very low below 5% which allow us to reject the null hypothesis of not existing a difference between those two tools when migrating data into Autonomous Database. Important of saying that this case in specific can be better explored as SQL Loader was not being used at its max so this scenario can be seen as future work if required. Emphasizing once again, as it was said in the previous scenario, that this conclusion can only be drawn, because although Golden Gate performs more operations than SQL Loader, it is more efficient having a much shorter loading time.

```
          welch Two Sample t-test

data:  dataComp$SQLLDR_2_ADW and dataComp$GG_Non_CDB_ADW
t = 17.161, df = 24.122, p-value = 5.049e-15
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 5.787387 7.369280
sample estimates:
mean of x mean of y
 31.93750  25.35917
```

Figure 7.12: T Test for SQL Loader and GG procedures to load Autonomous Database

Database Performance Analysis with SQL Loader

When it comes to the concept of database performance, there are two ways of analyse the problem because there are two different migration scenarios available that involves DCT and SQL Loader together or only Golden Gate through the use of extraction and replication. First with Golden Gate procedure, after checking the samples and doing the t-test that is displayed in the figure 7.13, we can validate that the p-value is inferior to 5% which implies that there is a significant difference between the times taken from each migration scenario and in a certain way lead us to think that a classic DBaaS in Oracle Cloud is more efficient than an Autonomous Database.

```
          welch Two Sample t-test

data:  dataMin$Non_CDB_DBaaS and dataMin$Non_CDB_ADW
t = -53.066, df = 19.039, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -25.57446 -23.63387
sample estimates:
mean of x mean of y
  0.45500  25.05917
```

Figure 7.13: T Test for Non CDB Migration to Non CDB Autonomous Database

But this can be probably happening due to the use of different procedures in Golden Gate. For the first scenario, migration from Non-CDB to the classic DBaaS from Oracle were used the following processes:

- Classic Extract;

- Integrate Replicat.

For the other scenario, that migrates data to the Autonomous Database, were used:

- Classic Extract;

- Classic Replicat.

And we know that Integrate Replicat is faster and more efficient than the Classic one, so to be totally sure that the databases behave differently we need to compare the samples obtained from loading both DB's with the use of DCT/SQL Loader too. This way we will have two different scenarios of comparison to ensure if they perform differently or not according to the target database being containerized or not. After fulfilling the t-test to compare the SQL Loader scenarios, it's safe to assume that both databases, being containerized or not, they continue to have a similar behaviour. In figure 7.14, the p-value is superior to 5% what makes impossible to reject the null hypothesis for this case. There is not an impacting difference while using SQL Loader to load data into Oracle's Autonomous Database or DBaaS.

```
           Welch Two Sample t-test

data:  dataMinSqlldr$sqlldr_dbaas and dataMinSqlldr$sqlldr_adw
t = -0.19911, df = 27.203, p-value = 0.8437
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.6969073  0.5735740
sample estimates:
mean of x mean of y
 31.87583  31.93750
```

Figure 7.14: T Test for SQL Loader Scenarios

GG - Quality Evaluation Framework

When it comes to our final evaluation using the QEF methodology, after developing the document and giving the correct evaluations to every requirement of each dimension, we came across a specific value as we can see in appendix D. The final result was 71% what means that the application used in the final solution is a reasonable option when it comes to this scenario of migrating data into the Cloud, but can be improved in some questions such:

- Data Transformation;

- Rollback or backup points defined in the process;

- More explicit and well defined documentation about possible implementations;

- Better display and interface.

But overall is a good tool that does the job and fulfills the main objectives of a data migration and could be even used to do bidirectional replications, depending of the client requirements.

Chapter 8

Conclusion

8.1 Final Considerations

After a deep study of the Cloud it is concluded that this concept of Cloud Computing is becoming more and more important as time goes by, and for companies it's even more relevant as they are always looking for better, cheaper and highly scalable solutions. In general it's not very disruptive when it comes to the different types of technology being used and the ways of implementation (containerization or virtualization), but has a large impact in all kinds of market due to its form of use. The essential idea with this project was to change the current business model, allowing cloud computing to be in the same model as traditional computing. That in this case, includes the migration of data into Oracle Cloud from on-premises systems, having a low consumption of computational resources always through the use of cloud services. When it comes to cloud computing services, they will keep expanding its services and launching new applications because the world of technology is in a continuous evolution and Cloud represents a big piece of it. In this thesis, Oracle's Cloud environment was the one focused due to to Wipro partnership with it, and a deep investigation was done when it comes to some of the services provided by it:

- Virtualization and containerization;
- DBaaS;
- Autonomous databases;
- Object Storage;
- Migration tools (PaaS);
- Networks configuration;

After searching and designing what were the possible solutions, several points needed to be raised and discussed in order to check if our initial goals were achieved. Overall Oracle Cloud is very well organized, easy to follow up, very accessible and ends up providing one of our solutions which is Oracle Golden Gate (On-premise and as a Cloud Service). So it is a good cloud computing service to start using and studying as years go by because it already provides OR solutions and easy integration with other tools. It would allow a major save when it comes to resource management, physical and human, due to an automation in virtual machines preparation and provides analytic data that allows to easily check if the resources are being used efficiently. Finally for data migration tools, many were checked and detailed but due to Wipro current business and implementation process, our final options were restricted to Golden Gate and SQL Loader due to best compatibility with Oracle Cloud

and with DCT that is being used in Wipro. As a conclusion we can say that both are great solutions but Golden Gate can be highly superior in specific scenarios that were discussed in the previous chapter 7. Compared to the old process, the performance of cloud data migrations is almost similar to the one between on-premises when using Golden Gate, so it's possible to say that this solution has a great performance and is more versatile than the old one, although it's never a direct comparison due to all the variables that stand between the different processes. Emphasizing that the best approach would be to use Golden Gate for cloud data migrations, but for now, in order to start taking smaller steps into a new and improved solution, it's best to keep DCT with the new SQL Loader configuration as it allows a smoother transition. Finally in the next section 8.2, it's analysed and left open what can be done in the future, which tools can be even more useful and easier for other situations because it can be a little harsh for a company to stay out of Cloud services like Azure, AWS or Google Cloud. With this said, a company like Wipro should always keep an open mind when it comes to new technologies and opportunities that can expand its business model and increase significantly its revenue as time goes by.

8.2   Future Work Perspective

I believe that when, and if, the transition to this new approach and new services is accepted and planned, a new analysis of the solution will have to be reconsidered, and it becomes necessary a full review of the steps explained in chapters 5 and 6 due the different specifications and implementations that the clients can ask for. After the evaluation was done in chapter 7, it was highlighted that Golden Gate performance was superior to SQL Loader, but it's important to keep in mind that the evaluation didn't consider the use of multi-threading which can improve the performance substantially and maybe overcome the results obtained before with Golden Gate. So, this is an important point to raise later when implementing a possible final solution. But the world of data migration is very wide, with some limitations when it comes to Cloud, but there are other tools easier to manage and that support other types of databases that need deeper investigation, for example the Qlik Replicate from Microsoft that was described and analysed in sections 3.3.1 and 5.1.1 respectively. If possible, it would be very rewarding to analyze its performance compared to the other approaches described, always knowing that it will have advantages and disadvantages. It is expected that there will be a need for adaptations when it comes to security and configuration of the applications, because our data is never 100% safe in the Cloud, so this concept will require a lot of attention later. But in short, we believe that this is the time for Wipro to study business opportunities that include Cloud, because it will transform the way it works and expand its business.

Bibliography

Advantages and Disadvantages Of Cloud Computing. (Accessed on 10/08/2020). url: `https://www.guru99.com/advantages-disadvantages-cloud-computing.html`.

Alexander Osterwalder Yves Pigneur, Alan Smith (2015). "Value Proposition Design". In: 1.1. (Accessed on 02/10/2020), pp. 1–619. url: `https://www.academia.edu/37165637/Value_Proposition_Design_-_A._Osterwalder`.

Allen, Clifton L. (2019). "The Benefits and Risks of Cloud Computing". In: *Operational Guidance* 1.1, p. 1. url: `https://www.missionbox.com/article/489/the-benefits-and-risks-of-cloud-computing`.

Alon, Shimon (2018). *https://www.sec.gov/Archives/edgar/data/893821/000117891318002529/exhibit_1.htm*. (Accessed on 10/17/2020). url: `https://www.sec.gov/Archives/edgar/data/893821/000117891318002529/exhibit_99-1.htm`.

Alooma (2018). *Cloud Migration Tools*. (Accessed on 01/31/2020). url: `https://www.alooma.com/blog/cloud-data-migration-tools`.

Amar, Rami (2018). *Best Practices for Migrating from On-Prem to Cloud | Alooma*. (Accessed on 01/30/2020). url: `https://www.alooma.com/blog/best-practices-for-migrating-data-from-on-prem-to-cloud`.

Asif Laghari, Hui He (2018). "Quality of Experience Framework for Cloud Computing (QoC)". In: (Accessed on 02/15/2020). url: `https://www.researchgate.net/publication/327093313_Quality_of_Experience_Framework_for_Cloud_Computing_QoC`.

AWS (2019). *AWS Database Migration Service - Amazon Web Services*. (Accessed on 02/05/2020). url: `https://aws.amazon.com/dms/?nc=sn&loc=1`.

Azure, Microsoft (2019a). *What is Azure—Microsoft Cloud Services*. (Accessed on 02/05/2020). url: `https://azure.microsoft.com/en-us/overview/what-is-azure/`.

– (2019b). *What is Azure—Microsoft Cloud Services*. (Accessed on 02/05/2020). url: `https://azure.microsoft.com/en-us/overview/what-is-azure/`.

Barsukov, Nikita (2015). *Using T-tests to benchmark page load performance*. (Accessed on 02/15/2020). url: `https://medium.com/gotoassist-product-blog/using-t-tests-to-benchmark-page-load-performance-6f45faff091e`.

Beal, Vangie (2019). *Best ERP Software 2019: Comparison & Reviews*. (Accessed on 02/22/2020). url: `https://www.webopedia.com/TERM/E/ERP.html`.

Brainly.com. *which is the best definition of "value"?* (Accessed on 02/07/2020). url: `https://brainly.com/question/1476533`.

Class, DBA. *install oracle goldengate 12c on linux 7 - DBACLASS DBACLASS*. (Accessed on 08/29/2020). url: `https://dbaclass.com/article/install-oracle-goldengate-12c-on-linux-7/`.

CustomerThink (2016). *What is Customer Value and How Can You Create It?* (Accessed on 02/07/2020). url: `http://customerthink.com/what-is-customer-value-and-how-can-you-create-it/`.

Docs, Microsoft (2019). *Overview of Data Migration Assistant (SQL Server) - SQL Server*. (Accessed on 02/05/2020). url: `https://docs.microsoft.com/en-us/sql/dma/dma-overview?view=sql-server-ver15`.

Goldkuhl, Göran (2016). "Separation or unity? Behavioral science vs Design science". In: *AIS SIGPRAG* 12.1, p. 1. url: http://www.vits.org/publikationer/dokument/804.pdf.

Help, Software Testing (2019). *13 Best Data Migration Tools For Complete Data Integrity*. (Accessed on 01/31/2020). url: https://www.softwaretestinghelp.com/data-migration-tools/.

INQDO (2019). *AWS explained: the basics*. (Accessed on 02/05/2020). url: https://www.inqdo.com/aws-explained-the-basics/?lang=en.

Jarvis, Katelyn (2014). *IS Research Development an International Perspective - ppt download*. (Accessed on 10/17/2020). url: https://slideplayer.com/slide/976925/.

Kassambara, Alboukadel. *Normality Test in R: The Definitive Guide - Datanovia*. (Accessed on 09/24/2020). url: https://www.datanovia.com/en/lessons/normality-test-in-r/.

Kumar, Manish (2018). *Why place your bets on Microsoft Azure Cloud Services? - Zenesys Blog*. (Accessed on 10/17/2020). url: https://www.zenesys.com/blog/why-place-your-bets-on-microsoft-azure-cloud-services.

Limited, Wipro (2018). "IQNxt: Make your data work". In: *Oracle Cloud Infrastructure Documentation* 2.1, pp. 1–2. url: https://www.wipro.com/content/dam/nexus/en/service-lines/analytics/offerings/iqnxt-make-your-data-work.pdf.

Mansfield, Dani (2018). *Marketing theory: understanding customer value*. (Accessed on 10/17/2020). url: https://builtvisible.com/understanding-customer-value/.

Microsoft (2019). *Microsoft Data Migration Guide - Migrate Oracle to Azure SQL Database*. (Accessed on 02/05/2020). url: https://datamigration.microsoft.com/scenario/oracle-to-azuresqldb?step=1.

Mocenni. "The Analytic Hierarchy Process". In: (Accessed on 05/03/2020). url: https://www3.diism.unisi.it/~mocenni/Note_AHP.pdf.

Oleksiuk, Anna (2019). *Cloud Computing Vs On-Premises [Comparison Guide] - Intellias*. (Accessed on 10/17/2020). url: https://www.intellias.com/cloud-computing-vs-on-premises-comparison-guide-2019/.

Oracle. *Installing Oracle Database 12c on Windows*. (Accessed on 09/13/2020). url: https://www.oracle.com/webfolder/technetwork/tutorials/obe/db/12c/r1/Windows_DB_Install_OBE/Installing_Oracle_Db12c_Windows.html.

– *Oracle GoldenGate 19c*. (Accessed on 10/12/2020). url: https://www.oracle.com/middleware/technologies/goldengate.html.

– *Replicate On-Premises Data to Cloud With Oracle GoldenGate Cloud Service*. (Accessed on 09/02/2020). url: https://www.oracle.com/webfolder/technetwork/tutorials/obe/cloud/ggcs/Replicate_On-Premises_Data_to_Cloud_with_OGGCS/index.html?fbclid=IwAR3CneJTIn7w-yXeIlka9EqvjJPDKWV6pSXN7AbRWYg3wlo82hA2PDbGIXM#overview.

– (2017). "Migrating to Oracle Database: Your Guide to a Successful Migration, On-Premises or in the Cloud". In: *Migrating to Oracle Database: Your Guide to a Successful Migration, On-Premises or in the Cloud* 17.1, pp. 1–17. url: http://www.oracle.com/us/products/database/migrating-to-oracle-database-wp-3610526.pdf.

– (2018a). "Oracle Retail Merchandising Cloud Services - Data sheet |Oracle". In: *Oracle Retail Merchandising Cloud Services - Data sheet |Oracle* 2.1, pp. 1–2. url: http://www.oracle.com/us/industries/retail/retail-merchandise-cs-ds-3654456.pdf.

– (2018b). *Replicating Data to Oracle Autonomous Data Warehouse Cloud*. (Accessed on 08/29/2020). url: https://docs.oracle.com/goldengate/c1230/gg-winux/GGODB/replicating-data-oracle-autonomous-data-warehouse-cloud.htm#GGODB-GUID-808668AB-68B0-4D4A-96C4-A18FC06D1299.

- (2019a). *ATPLabs - Oracle Autonomous Database Hands-on Workshop*. (Accessed on 08/29/2020). url: `https://objectstorage.us-ashburn-1.oraclecloud.com/n/oraclepartnersas/b/ATPWorkshop/o/ATPLabs.pdf`.
- (2019b). *Retail Documentation–RMFCS: Data Conversion Migration Process - YouTube*. (Accessed on 10/17/2020). url: `https://www.youtube.com/watch?v=3QmJJ85lQLE&ab_channel=OracleLearning`.
- (2020a). *1 Introduction to Oracle GoldenGate*. (Accessed on 10/17/2020). url: `https://docs.oracle.com/goldengate/1212/gg-winux/GWUAD/wu_about_gg.htm#GWUAD117`.
- (2020b). *Choosing Capture and Apply Modes*. (Accessed on 09/15/2020). url: `https://docs.oracle.com/goldengate/1212/gg-winux/GIORA/process_mode.htm#GIORA212`.
- (2020c). "Overview of Data Transfer Service". In: *Oracle Cloud Infrastructure Documentation* 1.1, p. 1. url: `https://docs.cloud.oracle.com/en-us/iaas/Content/DataTransfer/Concepts/overview.htm`.

Oracle Cloud Database vs On-Premises - Understanding the Differences. (Accessed on 10/11/2020). url: `https://www.centroid.com/blog/oracle-cloud-database-vs-on-premises/`.

Oracle Retail Merchandising Foundation Cloud Service. (Accessed on 10/14/2020). url: `https://docs.oracle.com/cd/F11448_01/index.html`.

Osterwalder, Alexander (2004). "THE BUSINESS MODEL ONTOLOGY A PROPOSITION IN A DESIGN SCIENCE APPROACH". In: 1. (Accessed on 02/10/2020), pp. 1–172. url: `http://www.hec.unil.ch/aosterwa/PhD/Osterwalder_PhD_BM_Ontology.pdf`.

Pearlman, Shana (2019). *Data Migration: Strategy and Best Practices - Talend*. (Accessed on 01/30/2020). url: `https://www.talend.com/resources/understanding-data-migration-strategies-best-practices/`.

Peffers, Ken and Tuure Tuunanen (2006). "THE DESIGN SCIENCE RESEARCH PROCESS: A MODEL FOR PRODUCING AND PRESENTING INFORMATION SYSTEMS RESEARCH". In: *THE DESIGN SCIENCE RESEARCH PROCESS: A MODEL FOR PRODUCING AND PRESENTING INFORMATION SYSTEMS RESEARCH* 24.1, pp. 84–101. url: `https://pdfs.semanticscholar.org/e1fa/ec8846289113fdeb840ff3f32d102e46fbff.pdf`.

Prince Agarwal Manjari Sahai, Vaibhav Mishra (2014). *General hierarchy structure of AHP | Download Scientific Diagram*. (Accessed on 10/17/2020). url: `https://www.researchgate.net/figure/General-hierarchy-structure-of-AHP_fig2_264436283`.

Sharma, Aashish (2019). *AWS Cloud - All About Amazon Web Services Cloud*. (Accessed on 10/17/2020). url: `https://www.entrepreneuryork.com/technology/cloud-computing/aws-cloud-all-about-amazon-web-services-cloud/`.

Soetewey, Antoine. *How to do a t-test or ANOVA for more than one variable at once in R and communicate the results in a better way | by Antoine Soetewey | Towards Data Science*. (Accessed on 10/12/2020). url: `https://towardsdatascience.com/how-to-do-a-t-test-or-anova-for-many-variables-at-once-in-r-and-communicate-the-results-in-a-6defaa712e5`.

Solutions, SaM (2018). *Oracle Cloud vs. AWS [Comparative Analysis+Examples]*. (Accessed on 02/06/2020). url: `https://www.sam-solutions.com/blog/amazon-aws-vs-oracle-cloud-platform-a-comparative-analysis/`.

Stitch, Talend Company (2020). *Apache Airflow vs. Qlik Replicate vs. Stitch - Compare features, pricing, services, and more*. (Accessed on 05/10/2020). url: `https://www.stitchdata.com/vs/airflow/attunity/#data-src-dest`.

T N, Manjunath and Ravindra Hegadi (2013). "Data Quality Assessment Model for Data Migration Business Enterprise". In: *International Journal of Engineering and Technology* 5.

Taylor, Christine (2019). *Data Migration: the Process and Steps to Succeed*. (Accessed on 01/30/2020). url: `https://www.enterprisestorageforum.com/storage-management/data-migration.html`.

Techniques, Internation Software (2020). *Attunity Replicate | IST*. (Accessed on 10/17/2020). url: `https://ist.com.gr/page/112/attunity-replicate`.

Thion, Virginie et al. (2007). "An Evaluation Framework for Data Quality Tools". In:

Using Oracle SQL Loader with Examples. (Accessed on 10/12/2020). url: `https://www.oracle-dba-online.com/sql_loader.htm#:~:text=SQL20LOADER%5C%20utility%5C%20is%5C%20used,the%5C%20data%5C%20into%5C%20Oracle%5C%20Tables..`

Value Analysis, Canadian Society of. *Function Analysis system Technique (FAST)*. (Accessed on 02/18/2020). url: `https://www.valueanalysis.ca/fast.php`.

Veeratteshwaran (2018). *1 Introduction to Oracle GoldenGate*. (Accessed on 10/17/2020). url: `https://www.oracle-scn.com/oracle-goldengate-parallel-replicat/`.

Vengal, Thomas (2019). *Oracle GoldenGate for Big Data 19.1 is Released | Oracle Data Integration Blog*. (Accessed on 10/17/2020). url: `https://blogs.oracle.com/dataintegration/oracle-goldengate-for-big-data-191-is-released`.

Wire, Business. *Wipro Acquires Leading European Retail Solutions Provider, ENABLER; Acquisition to Strengthen Wipro's Worldwide Retail Solutions Portfolio as Well as Global Delivery Footprint*. (Accessed on 06/10/2020). url: `https://www.businesswire.com/news/home/20060601005476/en/Wipro-Acquires-Leading-European-Retail-Solutions-Provider-ENABLER-Acquisition-to-Strengthen-Wipros-Worldwide-Retail-Solutions-Portfolio-as-Well-as-Global-Delivery-Footprint`.

Woodall, Tony (2003). "Conceptualising 'Value for the Customer': An Attributional, Structural and Dispositional Analysis". In: *Academy of Marketing Science Review* 1. (Accessed on 02/10/2020), pp. 1–45. url: `https://www.researchgate.net/profile/Tony_Woodall/publication/228576532_Conceptualising_'Value_for_the_Customer'_An_Attributional_Structural_and_Dispositional_Analysis/links/556c7c1708aeab7772231484/Conceptualising-Value-for-the-Customer-An-Attributional-Structural-and-Dispositional-Analysis.pdf`.

Appendices

A Data Migration Scenarios - Statistics

Entity	N° of records
ACCOUNT	1048575
HISTORY	1048575

A.1 Data Migration Scenario - SQLLDR to ADW

Start Time	End Time	Minutes	Seconds	N° records	Replication
24/09/2020 11:30	24/09/2020 12:03	33,46666666	2008	1048575	1
24/09/2020 12:13	24/09/2020 12:44	30,4	1824	1048575	2
24/09/2020 12:45	24/09/2020 13:17	31,34999999	1881	1048575	3
24/09/2020 13:20	24/09/2020 13:52	32,21666666	1933	1048575	4
24/09/2020 13:55	24/09/2020 14:26	31,03333334	1862	1048575	5
24/09/2020 14:28	24/09/2020 15:02	34,00000001	2040	1048575	6
24/09/2020 15:06	24/09/2020 15:36	30,8	1848	1048575	7
24/09/2020 15:39	24/09/2020 16:11	32,51666667	1951	1048575	8
24/09/2020 16:30	24/09/2020 17:03	33,11666667	1987	1048575	9
24/09/2020 17:10	24/09/2020 17:40	30,63333333	1838	1048575	10
24/09/2020 19:10	24/09/2020 19:43	33,05	1983	1048575	11
24/09/2020 19:45	24/09/2020 20:17	32,4	1944	1048575	12
24/09/2020 20:27	24/09/2020 20:57	30,61666667	1837	1048575	13
24/09/2020 20:59	24/09/2020 21:32	32,91666666	1975	1048575	14
24/09/2020 21:36	24/09/2020 22:09	32,88333333	1973	1048575	15
24/09/2020 22:14	24/09/2020 22:44	29,5	1770	1048575	16
24/09/2020 22:47	24/09/2020 23:20	32,56666667	1954	1048575	17
24/09/2020 23:41	25/09/2020 00:13	32,61666666	1957	1048575	18
25/09/2020 00:20	25/09/2020 00:51	30,85	1851	1048575	19
25/09/2020 00:38	25/09/2020 01:08	30,58333333	1835	1048575	20

A.2 Data Migration Scenario - SQLLDR to DBaaS

Start Time	End Time	Minutes	Seconds	N° records	Replication
26/09/2020 21:48	26/09/2020 22:21	32,83333334	1970	1048575	1
26/09/2020 22:23	26/09/2020 22:55	31,26666667	1876	1048575	2
26/09/2020 22:58	26/09/2020 23:31	32,35	1941	1048575	3
26/09/2020 23:34	27/09/2020 00:06	32,03333333	1922	1048575	4
27/09/2020 09:15	27/09/2020 09:48	32,68333332	1960,999999	1048575	5
27/09/2020 09:52	27/09/2020 10:23	31,55	1893	1048575	6
27/09/2020 10:26	27/09/2020 10:57	31,9	1914	1048575	7
27/09/2020 11:01	27/09/2020 11:33	32,41666667	1945	1048575	8
27/09/2020 11:36	27/09/2020 12:07	31,28333333	1877	1048575	9
27/09/2020 12:10	27/09/2020 12:43	32,9	1974	1048575	10
27/09/2020 12:46	27/09/2020 13:17	31,33333333	1880	1048575	11
27/09/2020 16:12	27/09/2020 16:43	31,40000001	1884	1048575	12
27/09/2020 16:47	27/09/2020 17:19	31,95	1917	1048575	13
27/09/2020 17:23	27/09/2020 17:54	31,45	1887	1048575	14
27/09/2020 18:01	27/09/2020 18:33	31,68333333	1901	1048575	15
27/09/2020 18:35	27/09/2020 19:08	32,81666667	1969	1048575	16
27/09/2020 19:10	27/09/2020 19:41	31,51666666	1890,999999	1048575	17
27/09/2020 19:44	27/09/2020 20:15	31,06666666	1863,999999	1048575	18
27/09/2020 20:17	27/09/2020 20:50	32,51666666	1951	1048575	19
27/09/2020 20:55	27/09/2020 21:27	31,80000001	1908	1048575	20

A.3   Data Migration Scenario - Golden Gate from Non-CDB to ADW Database

Start Time	End Time	Minutes	Seconds	N° records	Replication
14/09/2020 18:01	14/09/2020 18:27	25,55	1533	1048575	1
14/09/2020 20:50	14/09/2020 21:19	28,73333333	1724	1048575	2
14/09/2020 21:22	14/09/2020 21:46	23,86666667	1432	1048575	3
14/09/2020 21:53	14/09/2020 22:16	22,36666667	1342	1048575	4
14/09/2020 22:17	14/09/2020 22:42	24,79999999	1488	1048575	5
15/09/2020 13:35	15/09/2020 14:00	24,81666666	1489	1048575	6
15/09/2020 14:17	15/09/2020 14:45	28,08333334	1685	1048575	7
15/09/2020 14:46	15/09/2020 15:11	25,68333333	1541	1048575	8
15/09/2020 15:15	15/09/2020 15:39	24,46666667	1468	1048575	9
15/09/2020 15:51	15/09/2020 16:16	24,85	1491	1048575	10
15/09/2020 16:28	15/09/2020 16:53	25	1500	1048575	11
15/09/2020 16:59	15/09/2020 17:26	26,99999999	1620	1048575	12
15/09/2020 17:34	15/09/2020 17:59	24,98333334	1499	1048575	13
15/09/2020 20:51	15/09/2020 21:16	25	1500	1048575	14
15/09/2020 21:47	15/09/2020 22:12	25	1500	1048575	15
16/09/2020 13:35	16/09/2020 14:01	25,81666667	1549	1048575	16
16/09/2020 14:35	16/09/2020 14:58	22,48333334	1349	1048575	17
16/09/2020 15:02	16/09/2020 15:30	27,31666666	1639	1048575	18
16/09/2020 15:35	16/09/2020 16:02	26,56666666	1594	1048575	19
16/09/2020 16:17	16/09/2020 16:42	24,79999999	1488	1048575	20

A.4 Data Migration Scenario - Golden Gate from Non-CDB to DBaaS

Start Time	End Time	Minutes	Seconds	N° records	Replication
16/09/2020 12:00	16/09/2020 12:00	0,483333331	28,99999986	1048575	1
16/09/2020 12:09	16/09/2020 12:09	0,383333329	22,99999972	1048575	2
16/09/2020 12:17	16/09/2020 12:18	0,45	27,00000003	1048575	3
16/09/2020 12:17	16/09/2020 12:18	0,45	27,00000003	1048575	4
16/09/2020 12:23	16/09/2020 12:23	0,483333331	28,99999986	1048575	5
16/09/2020 12:26	16/09/2020 12:26	0,583333333	35,00000001	1048575	6
16/09/2020 12:29	16/09/2020 12:30	0,549999992	32,99999954	1048575	7
16/09/2020 12:32	16/09/2020 12:32	0,516666672	31,00000033	1048575	8
16/09/2020 13:26	16/09/2020 13:27	0,43333333	25,99999979	1048575	9
16/09/2020 13:38	16/09/2020 13:39	0,550000003	33,00000017	1048575	10
16/09/2020 13:45	16/09/2020 13:46	0,399999999	23,99999995	1048575	11
16/09/2020 13:55	16/09/2020 13:55	0,43333334	26,00000042	1048575	12
16/09/2020 14:13	16/09/2020 14:13	0,383333339	23,00000035	1048575	13
16/09/2020 14:19	16/09/2020 14:19	0,333333338	20,00000027	1048575	14
16/09/2020 14:28	16/09/2020 14:29	0,41666667	25,00000019	1048575	15
16/09/2020 14:30	16/09/2020 14:30	0,366666669	22,00000011	1048575	16
16/09/2020 14:32	16/09/2020 14:32	0,43333333	25,99999979	1048575	17
16/09/2020 14:34	16/09/2020 14:35	0,500000002	30,0000001	1048575	18
16/09/2020 14:38	16/09/2020 14:38	0,500000002	30,0000001	1048575	19
16/09/2020 16:17	16/09/2020 16:18	0,45	27,00000003	1048575	20

A.5 Data Migration Scenario - Golden Gate from PDB to DBaaS

Start Time	End Time	Minutes	Seconds	N° records	Replication
18/09/2020 11:30	18/09/2020 11:31	0,550000003	33,00000017	1048575	1
18/09/2020 11:38	18/09/2020 11:39	0,699999996	41,99999976	1048575	2
18/09/2020 11:42	18/09/2020 11:43	0,483333331	28,99999986	1048575	3
18/09/2020 11:44	18/09/2020 11:45	0,499999991	29,99999947	1048575	4
18/09/2020 11:48	18/09/2020 11:48	0,583333333	35,00000001	1048575	5
18/09/2020 11:50	18/09/2020 11:50	0,383333329	22,99999972	1048575	6
18/09/2020 12:57	18/09/2020 12:57	0,41666667	25,00000019	1048575	7
18/09/2020 12:58	18/09/2020 12:59	0,45	27,00000003	1048575	8
18/09/2020 13:00	18/09/2020 13:00	0,399999999	23,99999995	1048575	9
18/09/2020 13:01	18/09/2020 13:01	0,349999998	20,99999988	1048575	10
18/09/2020 13:02	18/09/2020 13:03	0,466666671	28,00000026	1048575	11
18/09/2020 13:04	18/09/2020 13:04	0,516666672	31,00000033	1048575	12
18/09/2020 13:05	18/09/2020 13:05	0,383333339	23,00000035	1048575	13
18/09/2020 13:06	18/09/2020 13:07	0,566666673	34,0000004	1048575	14
18/09/2020 13:08	18/09/2020 13:09	0,43333334	26,00000042	1048575	15
18/09/2020 13:10	18/09/2020 13:10	0,566666673	34,0000004	1048575	16
18/09/2020 13:11	18/09/2020 13:11	0,466666671	28,00000026	1048575	17
18/09/2020 13:13	18/09/2020 13:13	0,41666667	25,00000019	1048575	18
18/09/2020 13:14	18/09/2020 13:14	0,349999998	20,99999988	1048575	19
18/09/2020 11:13	18/09/2020 11:14	0,45	27,00000003	1048575	20

B SQL Code

```
alter session set "_ORACLE_SCRIPT"=true;
CREATE user user_src IDENTIFIED BY user_src_password;
grant dba, connect, resource to user_src;
EXEC DBMS_STREAMS_AUTH.GRANT_ADMIN_PRIVILEGE('user_src');
EXEC DBMS_STREAMS_AUTH.GRANT_ADMIN_PRIVILEGE('ggadmin');
exec dbms_goldengate_auth.grant_admin_privilege('user_src',container=>'all')
ALTER DATABASE ARCHIVELOG;
select count(*) from user_src.history;
ALTER SYSTEM SET ENABLE_GOLDENGATE_REPLICATION=TRUE SCOPE=BOTH;

GRANT SELECT ANY DICTIONARY TO user_src;
GRANT UNLIMITED TABLESPACE TO user_src;
GRANT CREATE ANY TABLE TO user_src;
BEGIN
DBMS_GOLDENGATE_AUTH.GRANT_ADMIN_PRIVILEGE
(grantee=>'user_src',
privilege_type=>'capture',
grant_optional_privileges=>'*');
END;

CREATE TABLE user_src.ACCOUNT (ACCOUNT_NO number(10,0), BALANCE number(18,2),PRIMARY KEY (ACCOUNT_NO));
CREATE TABLE user_src.HISTORY (SEQUENCE_NO number(10,0), ACCOUNT_NO number(10),PRIMARY KEY (SEQUENCE_NO));
alter database add supplemental log data;
ALTER SESSION SET CONTAINER=pdborcl;
ALTER SESSION SET CONTAINER=cdb$root;
ALTER PLUGGABLE DATABASE OPEN READ WRITE;
SHOW CON_NAME
alter database add supplemental log data;

ALTER SYSTEM SET ENABLE_GOLDENGATE_REPLICATION=TRUE;
EXECUTE IMMEDIATE 'alter pluggable database all open';
CREATE or REPLACE trigger OPEN_ALL_PLUGGABLES
    after startup
    on  database
BEGIN
    execute immediate 'alter pluggable database all open';
END open_all_pdbs;
alter pluggable database pdborcl save state;
ALTER SESSION SET CONTAINER=pdb1 SERVICE='pdbservice';
```

Figure 1: Source Database - Main Configuration

```sql
alter session set "_ORACLE_SCRIPT"=true;
desc dba_users;
ALTER SYSTEM SET ENABLE_GOLDENGATE_REPLICATION=TRUE SCOPE=BOTH;
Select username, default_tablespace from dba_users;
]/**
Create oogtrg User to prepare replication environment
.*/
create user oggtrg identified by oracle_ORA_1234 default TABLESPACE USERS temporary TABLESPACE temp;
alter user oggtrg identified by oracle_ORA_1234 account unlock;
grant connect, resource to oggtrg;
alter user OGGTRG quota unlimited on USERS;
]/**
Create oggadmin User to set up golden gate config in db
.*/
create user oggadmin identified by oracle_ORA_1234 default tablespace USERS temporary TABLESPACE temp;
alter user oggadmin identified by oracle_ORA_1234 account unlock;
grant dba to oggadmin;
grant connect, resource to oggadmin;
GRANT CREATE SESSION TO oggadmin;
GRANT UNLIMITED TABLESPACE TO oggadmin;
GRANT CREATE SESSION, ALTER SESSION, RESOURCE, CONNECT, SELECT ANY dictionary TO oggadmin;
GRANT CREATE TABLE   TO oggadmin;
exec dbms_goldengate_auth.grant_admin_privilege('oggadmin','apply');
exec dbms_goldengate_auth.grant_admin_privilege('oggadmin',container=>'pdb1');
]/**
Create sqlloader User to use in sql loader
.*/
create user sqlloader identified by oracle_ORA_123 default tablespace USERS temporary TABLESPACE temp;
grant dba to sqlloader;
exec dbms_goldengate_auth.grant_admin_privilege('sqlloader','apply');
SELECT name, pdb
FROM   v$services
ORDER BY name;
ALTER SESSION SET CONTAINER=pdb1;
SHOW CON_NAME;
select count(*) from oggtrg.account;
truncate table oggtrg.account;
truncate table oggtrg.history;
select count(*) from oggtrg.history;
```

Figure 2: Target Database - Main Configuration

```
DROP TABLE ACCTN;
CREATE TABLE ACCOUNT
(
   ACCOUNT_NO                      NUMBER      (10,0)      NOT NULL
,  BALANCE                         NUMBER      (18,2)         NULL
,  CONSTRAINT PK_ACCTN
   PRIMARY KEY
   (
     ACCOUNT_NO
   )
    USING INDEX
);


DROP TABLE HISTORY;
CREATE TABLE HISTORY
(
   SEQUENCE_NO                     NUMBER      (10,0)      NOT NULL
,  ACCOUNT_NO                      NUMBER      (10,0)      NOT NULL
,  CONSTRAINT PK_HISTORY
   PRIMARY KEY
   (
    SEQUENCE_NO
   )
   USING INDEX
);
```

Figure 3: Target Database - Tables Creation

C Golden Gate Param Files

C.1 Golden Gate with Autonomous DB Param Files

```
extract extl
userid user_src, password user_src_password
exttrail ./dirdat/in
table user_src.ACCOUNT;
```

Figure 4: Golden Gate to Autonomous DB - Classic Extract

```
extract extlpump
userid user_src, password user_src_password
RMTHOST localhost, MGRPORT 7809
rmttrail ./dirdat/pn
table user_src.ACCOUNT;
```

Figure 5: Golden Gate to Autonomous DB - Pump

```
PORT 7809
Dynamicportlist 7810-7820, 7830
ACCESSRULE, PROG COLLECTOR, IPADDR 132.145.66.153, ALLOW
PURGEOLDEXTRACTS ./dirdat/pn*, USECHECKPOINTS, MINKEEPHOURS 1
AUTORESTART EXTRACT ext*, RETRIES 4, WAITMINUTES 4
AUTORESTART REPLICAT rep*, RETRIES 4, WAITMINUTES 4
```

Figure 6: Golden Gate to Autonomous DB - Manager in Source

```
PORT 7809
DYNAMICPORTLIST 7810-7815
ACCESSRULE, PROG *, IPADDR *, PRI 1, ALLOW
PURGEOLDEXTRACTS /u02/trails/pn*, USECHECKPOINTS, MINKEEPHOURS 1
```

Figure 7: Golden Gate to Autonomous DB - Manager in Cloud

```
replicat repdw
USERIDALIAS ggadminalias
REPORTCOUNT EVERY 30 SECONDS, RATE
MAP user_src.*, TARGET goldengatecloud.*;
```

Figure 8: Golden Gate to Autonomous DB - Classic Replicat

C.2 Golden Gate with DBaaS - Integrate Param Files

```
extract ext1
userid user_src, password user_src_password
TranlogOptions IntegratedParams (max_sga_size 256)
exttrail ./dirdat/in
LOGALLSUPCOLS
UPDATERECORDFORMAT COMPACT
DDL INCLUDE ALL
REPORTCOUNT every 10 minutes, RATE
SOURCECATALOG pdb1
table user_src.ACCOUNT;
```

Figure 9: Golden Gate to DBaaS - Integrate Extract

```
extract ext1pump
userid user_src, password user_src_password
PASSTHRU
RMTHOST 132.145.66.153, MGRPORT 7809
rmttrail /u02/trails/pn
SOURCECATALOG pdb1
table user_src.*;
```

Figure 10: Golden Gate to DBaaS - Pump

```
PORT 7809
DYNAMICPORTLIST 7810-7815
ACCESSRULE, PROG *, IPADDR *, PRI 1, ALLOW
PURGEOLDEXTRACTS /u02/trails/pn*, USECHECKPOINTS, , MINKEEPHOURS 1
```

Figure 11: Golden Gate to DBaaS - Manager in Source

```
replicat repdw
USERIDALIAS oggadmin
DBOPTIONS ENABLE_INSTANTIATION_FILTERING
ASSUMETARGETDEFS
DDL INCLUDE ALL
REPORTCOUNT EVERY 1 SECONDS, RATE
MAP PDB1.user_src.*, TARGET PDB1.oggtrg.*;
```

Figure 12: Golden Gate to DBaaS - Integrate Replicat

C.3 Golden Gate with DBaaS - Classic/Integrate Param Files

```
extract ext1
userid user_src, password user_src_password
exttrail ./dirdat/in
table user_src.ACCOUNT;
```

Figure 13: Golden Gate to DBaaS - Classic Extract

```
extract ext1pump
userid user_src, password user_src_password
RMTHOST localhost, MGRPORT 7809
rmttrail ./dirdat/pn
table user_src.ACCOUNT;
```

Figure 14: Golden Gate to DBaaS - Pump

```
replicat repdw
USERIDALIAS oggadmin
DBOPTIONS ENABLE_INSTANTIATION_FILTERING
ASSUMETARGETDEFS
DDL INCLUDE ALL
DDLERROR DEFAULT IGNORE
REPORTCOUNT EVERY 1 SECONDS, RATE
MAP user_src.ACCOUNT, TARGET oggtrg.ACCOUNT;
```

Figure 15: Golden Gate to DBaaS - Integrate Replicat

D   Golden Gate Quality Evaluation Framework

q	D	q_i	Dimension	Q_j	W_{ij} (Factor Weight j in Dim i) [0,1]	Factor	rw_{jk} (requirement weight k in Factor j) {2, 4, 6, 8, 10}	Requirement	wf_k % requirement fulfillment k) [0,100]
71%	0,63	74,28	Functionality	84,44	0,40	Functional	10	FF01 - Data replication	100
							10	FF02 - Data extraction	100
							8	FF03 - Data Transformation	50
							8	FF04 - Data Validation	80
				67,5	0,60	User Interaction	2	FUI01 - All the applications present a similar explorer experience	60
							8	FUI02 - Application processes are organized by a stage/step view related to migrations	100
							6	FUI03 - Permissions are guaranteed	0
							6	FUI04 - Easy to configure main features	100
							2	FUI05 - Applications support offline operations	0
							8	FUI06 - All statistics gathered by the applications processes are well organized and present relevant information on the situation analysis	80
		60	Adaptability/Consistency	100	0,43	Versatility	4	AV01 - Applications are present in their environments in a native way	100
							6	AV02 - Applications allow to configure some options	100
							10	AV03 - Application compatibility with Cloud	100
				70	0,14	Pedagogical aspects	2	APA01 - Applications can mantain their users interested without create anxiety situations	70
				50	0,14	Accuracy	4	AA01 - Number of incidents or malfunctions/ Reliability	50
								AA02 - Comparison with reality (applied scenarios)	50
				0	0,14	Content Quality	8	FCQ01 - All application information is well organized	20
								FCQ02 - Quality of log messages displayed	50
				0	0,14	Maintenance	10	AM01 - Applications have resume or restart option maintaining consistency over the data	0
		58	Relevancy	20	0,20	Visual quality	4	RVQ01 - Applications are intuitive and of easy understanding	50
							6	RVQ02 - Display method (cmd, webpage, app GUI)	20
				67,5	0,80	Navigation	2	RN01 - Simple way to access the main procedures	60
							4	RN02 - Application user interface is quick and fast responsible	50
							10	RN03 - Application runtime does not have errors, and unexpectable errors should be treated	50
							8	RN04 - Login and Access security	100

Figure 16: Golden Gate QEF Overall Rating

Dimension	Functionality			
Factor	Functional			
		Wk - Fulfilment (%)		
Requirement	Metric Evaluation	0	50	100
FF01 - Data replication	User can configure a data replication procedure	No access to functionality	Partial access to the functionality	Full access to the functionality
FF02 - Data extraction	User can configure a data extraction procedure	No access to functionality	Partial access to the functionality	Full access to the functionality
FF03 - Data Transformation	User can configure different types of transformations, specify table parameters for entities	No access to functionality	Partial access to the functionality	Full customization when it comes to data transformation
FF04 - Data Validation	User is allowed to set data validations, constraints, data types for entities	No access to functionality	Partial access to the functionality	Every kind of validation is possible

Figure 17: QEF Functionality Dimension – Functionality

	Dimension	Functionality			
	Factor	User Interaction			
				Wfk – Fullfilment (%)	
Requirement	**Metric Evaluation**		**0**	**50**	**100**
FUI01 - All the applications present a similar explorer experience	The navigation between processes present the same experience. Execution commands, files configuration, displayed file system.		No	Partially equal	Yes
FUI02 - Application processes are organized by a stage/step view related to migrations	All the processes should be in separated windows in order to allow the user a better understanding of the migration scenario.		Only one window	Some stages mixed but at least 2 windows are displayed	Every appropriate process is separated in a new window
FUI03 - Permissions are guaranteed	Functionalities must be available only to apropriated users.		No permissions when running any process	-	Permissions granted to any process
FUI04 - Easy to configure main features	Minimum number of lines to configure a process. Depends of the process type.		>8 lines	-	<= 4 lines
FUI05 - Applications support offline operations	When running in offline mode, it's possible to perform operations.		No	-	Yes
FUI06 - All statistics gathered by the applications processes are well organized and present relevant information on the situation analysis	Statistics are gathered and displayed by each process and it's divided depending of the database operations that are replicated and by each table.		Only displays number of operations	Number of operations by each table	Number of operations divided by database DML and by table

Figure 18: QEF Functionality Dimension – User Interaction

Dimension	Consistency				
Factor	Versatility, Motivational aspects, Pedagogical aspects, Accuracy, Content quality, Maintenance				
			Wfk – Fulfilment (%)		
Requirement	Metric Evaluation	0	50	100	
AVO1 - Applications are present in their environments in a native way	All the applications must apply the same patterns refering audio and video. Document refering audio and video specifications can be consulted in the QA manual - Audio and Video section.	No	-	Yes	
AVO2 - Applications allow to configure some options	Application allows to set times to refresh logs, automatic files purges, set inbound rules, ports	No	-	Yes	
AVO3 - Application compatibility with Cloud	Application is compatible with different Cloud Computing services	0	1	>=2	
APA01 - Applications can mantain their users interested without create anxiety situations	Login exceptions (unknown username/password) must present user a message indicating the cause of unperformed login. Only registered users can make login.	No	Partially clear, detailed	Yes	
AA01 - Number of incidents or malfunctions./ Reliability	Unexpected errors must be presented to the user with a clear message about the error.	No	Partially clear, not very detailed	Yes	
	Number of issues during scenario execution	<3	<6	>=6	
AA02 - Comparison with reality (applied scenarios)	Scenario applied can be easily compared a reality case.	No	Large amount of data but simple case	Yes	
FCO01 - All application information is well organized	Information well displayed. If several processes are running to perform migrations with different scenarios.	No	-	Yes	
FCO02 - Quality of log messages displayed	Log messages are intuitive and displayed in order of operations by certain intervals. Reports can be taken.	No	Only simple log file displayed	Yes	
AM01 - Applications have resume or restart option maintaining consistency over the data replicated	Automatic actions are done. No need to set up processes or databases again.	No	-	Yes	

Figure 19: QEF Consistency Dimension

Dimension	Relevancy			
Factor	Visual quality, Navigation			
		Wfk – Fullfilment (%)		
Requirement	**Metric Evaluation**	**0**	**50**	**100**
RVQ01 - Applications are intuitive and of easy understanding	All the applications must be of easy understanding, intuitive and easy of control. Documentation is available in Oracle Docs.	No	After obtainning knowledge from documentation is simple to manage the apps.	Yes
RVQ02 - Display method (cmd, webpage, app GUI)	If the application is displayed in the command line or in a webpage or a simple GUI	CMD	-	Webpage or GUI
RN01 - Simple way to access the main procedures	How many steps should be taken to see results from main procedures.	>8	>4	<=2
RN02 - Application user interface is quick and fast responsible	Time to start any process in seconds	<1	<5	>=5
RN03 - Application runtime does not have errors, and unexpectable errors should be treated	Login exceptions (unknown username/password) must present user a message indicating the cause of unperformed login. Only database users with correct permissions can make login.	No	-	Yes
RN04 - Login and Access security	All of the actions must be made by user interaction. Automatic actions doesn't exist.	No	-	Yes

Figure 20: QEF Relevancy Dimension