



Ellen Luna

Process Reinforcement (Application Migration and Version Update)

Metropolia University of Applied Sciences

Master of Engineering

Information Technology

Master's Thesis

27 May 2024

Abstract

Author: Ellen Luna
Title: Process Reinforcement (Application Migration and Version Update)
Number of Pages: 56 pages
Date: 27 May 2024

Degree: Master of Engineering
Degree Programme: Information Technology
Professional Major: Networking and Services
Supervisors: Sami Sainio, Principal Lecturer
Ville Jääskeläinen, Principal Lecturer

The emergence of errors during application migration or system updates on both the server and user stations presents significant challenges for the company, resulting in deviations from the allocated time for the activity. Due to this occurrence, the technical team's problem-solving skills and expertise were not only tested but also provided a valuable opportunity to innovate and refine the migration and version update processes. Furthermore, undertake a thorough root cause analysis, implement enhanced testing and deployment methodologies, and establish a streamlined framework for future updates to guarantee long-term efficiency and quality.

The issue at hand revolves around the challenge of transitioning the applications from one server to another or updating the current setup, both of which are causing errors on the server. Consequently, the generation of scheduled tasks from the server becomes unsuccessful, resulting in the non-delivery of anticipated reports to the users or clients. In which affecting the client's overall experience and the efficiency of their decision-making processes.

Hence, we took the initiative to enhance and reinforce its processes by establishing a well-defined targeted turnaround time. This modified approach aims to ensure efficient outcomes for the organization. By integrating and adding new procedures to its main process and

installing the refined framework, the company will not only strengthen its operational resilience but also pursue optimized performance and results.

Following the implementation of the enhanced framework, notable improvements were observed, particularly in the significant reduction of issues associated with application migration and version updates. Subsequently, instances of migration-related problems, when logged, were promptly addressed within a shorter timeframe.

These challenges presents a valuable chance to investigate the underlying issues more thoroughly, strengthen the testing and deployment practices, and establish a foundation for future updates that are both smooth and efficient. By embracing these challenges as opportunities for learning and advancement will undoubtedly contribute to the company's success and set the stage for continued innovation.

Keywords: Application migration, version update

Table of Contents

1.	Introduction	8
1.1	Business Context	9
1.2	Design and Structure	10
2	Project Specification	11
2.1	Current State Analysis (Problem statement)	11
2.2	Scope	13
2.3	Objective	13
2.4	Expectations	14
3	Methodology	14
3.1	Research Design	15
3.2	Tools	15
3.3	Approach Method	16
3.4	Data Collection	16
3.5	Data Analysis Procedure	18
3.6	Ethical Considerations	18
3.7	Limitations	19
3.8	Key findings	19
4	Theoretical Background	22
4.1	Migration	22
4.2	Categories of migration techniques	23
4.3	Main stages of migration	24
4.3.1	Pre-Migration Assessment	24
4.3.2	Actual Migration	25
4.3.3	Post-migration	26
4.4	Various Type of Migration Technology	27
4.4.1	Data Migration:	27
4.4.2	Application Migration:	27
4.4.3	Infrastructure Migration:	27
4.4.4	Platform Migration:	28
4.5	Migration Challenges	28
4.5.1	Compatibility Issues:	29

4.5.2	Data Loss and Integrity:	31
4.5.3	Downtime and Disruption:	32
4.5.4	Resource Constraints:	33
4.5.5	User Adoption and Training:	33
4.5.6	Security Risks:	33
4.5.7	Vendor or Supplier Dependencies:	33
4.5.8	Performance and Scalability:	34
4.5.9	Regulatory and Compliance Considerations:	34
4.6	The process modifications	34
4.7	Defining process performance	35
4.8	Process improvement and innovation	36
5	Implementation	37
5.1	Objective and Scope	37
5.2	Earlier Migration Process	38
5.3	Enhanced process (Implemented)	39
5.4	Detailed of the newly added process	39
5.4.1	File Inventory Procedure:	39
5.4.2	Sanity checking	40
5.5	Case Studies	40
5.6	Case Study 1	41
5.7	Analysis on gather data	45
5.7.1	Issue Graph:	46
5.7.2	SLA(Service Level Agreement) Graph:	46
5.8	Case Study 2:	47
6	Summary	51
6.1	The study	51
6.2	The challenges	52
6.3	Future improvements	53
	References	57

List of Abbreviations and Definition

2D (two-dimensional) - It refers to objects or representations that have only two dimensions: height and width.

3D (three-dimensional) - depict objects or scenes in three-dimensional space by adding depth for a realistic and immersive experience, mimicking how our eyes perceive spatial relationships.

Dynamic Link Library (DLL)- a file format and a collection of code and data that multiple programs can use simultaneously. It contains functions, procedures, and routines that software applications can call upon during execution.

ICT (Information and communication technology) - refers to all technologies used to handle telecommunications, broadcast media, audio-visual processing and transmission systems, intelligent building management systems, and network-based control and monitoring functions.

SaaS (Software as a Service) - is a cloud-based method of providing software to users.

SLA (Service Level Agreement) - is a formal agreement between a service provider and a customer that outlines the expected level of service, including specific standards, responsibilities, and measures for performance.

SQL (Structured Query Language) - a standard programming language designed for managing and manipulating relational databases.

TAT (Turnaround Time)-refers to the amount of time it takes to complete a process or fulfill a request. It is often used in business and service contexts to measure the time it

takes for a task to be completed, starting from the initiation of the request until its resolution or completion.

UAT (User Acceptance Testing) - it is the final phase of the software testing process, where the intended users test the software to ensure it can handle required tasks according to specifications.

1. Introduction

In the ever-evolving landscape of technology, organizations face the constant challenge of keeping their software applications up-to-date to ensure they meet the needs of their users and remain competitive in the marketplace. For FincX company an application migration and version updates are critical processes that play a crucial role in optimizing software performance, enhancing security, and introducing new features that improve user experience.

A recent study by Lawrence and Simon in 2023, found that software issues were the most common cause of (software system service) downtime in 2023, accounting for a staggering 65% of all outages [11]. This highlights the importance of implementing a well-defined process for application migration and version updates to minimize disruptions and maintain business continuity.

Migration in simple terms means relocating from one location to another. It may also be a process of moving data or application from another system or platform. There were numerous reasons for an organization to undertake application migration or version update, nevertheless, one of the paramount objectives is to improve performance and efficiency of the company's system. More specifically the reason for migration could be more diverse like, improving security and governance, changing devices to prevent data loss, keeping up with current technologies to avoid obsolescence, gain superior value in business, to name a few examples.

In its early stages, the organization adopted a straightforward approach to software deployment and updates, relying on manual processes and basic tools. This rudimentary methodology served its purpose during the organization's formative years, when the volume of software deployment and update tasks was fairly easy to handle.

With fewer applications and a smaller IT infrastructure, any errors that arose could be promptly identified and addressed without causing significant disruptions to business operations.

As the organization's growth resulted in an expansion of its custom-developed and in-house software applications, the number of accompanying tasks also increased. To accommodate this growing workload, additional servers were procured and allocated among the different departments. However, as business operations flourished, the shortcomings of the existing process model became increasingly apparent, necessitating the development of a more sophisticated and scalable approach.

The primary goal of this study is to considerably improve the existing process model to make it more flexible, efficient, and adaptable to accommodate changes in the organization's landscape. The enhanced process model will undergo thorough real-world testing using different software applications and infrastructure configurations to validate its effectiveness in meeting the changing requirements of a business environment.

1.1 Business Context

The main organization, FincX Financial Management Company, headquartered in Singapore, commands a prominent position in the global financial landscape. Its strategic foothold in both London and China allows it to specialize in advanced fund management services. With a diverse clientele spanning some parts of Asia and Europe, the company delivers tailored investment solutions that precisely meet the unique requirements of each market. Fueled by a steadfast commitment to excellence, innovation, and client satisfaction, FincX adeptly navigates the complexities of international finance. This dedication ensures that clients benefit from unmatched financial management expertise, seamlessly delivered across continents.

1.2 Design and Structure

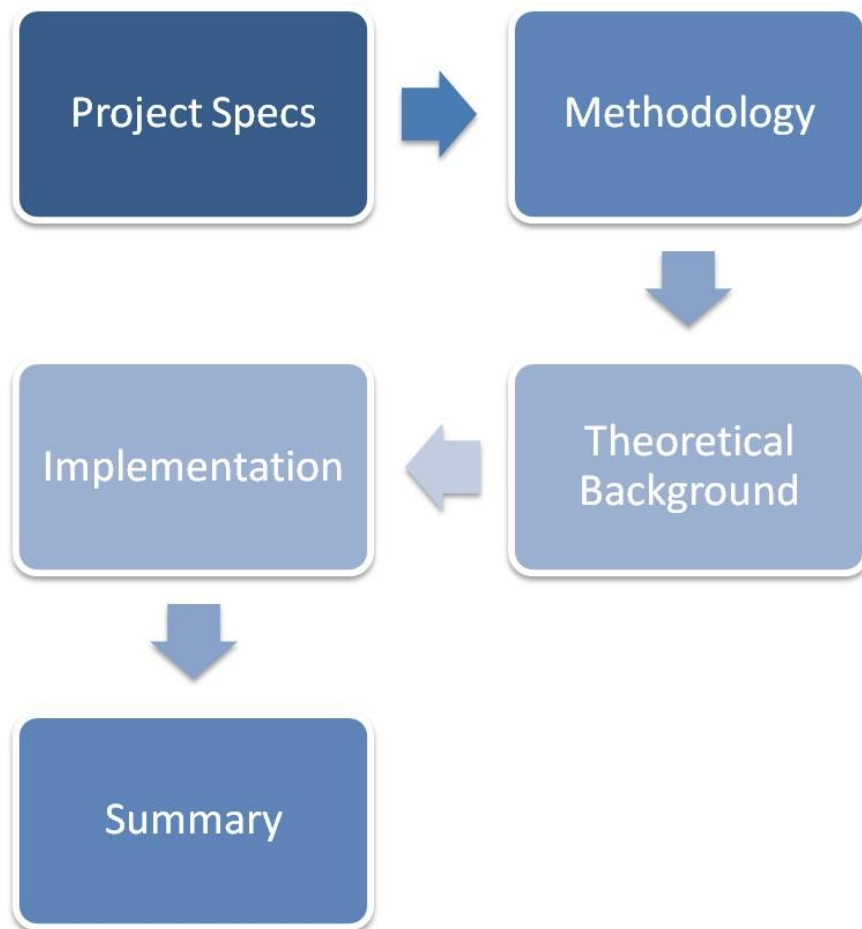


Figure 1: Thesis design and structure

The diagram above shows straightforward visual overview of a thesis's organization. It acts as a roadmap, revealing crucial sections like project specification, methodology, theoretical background, implementation, and conclusion. The design is crafted to lead readers systematically through the research process, proving a comprehensive understanding of the topic under study.

2 Project Specification

The purpose of this chapter is to define the problem statement, objectives, scope, and expectations associated with a project. These will serve as a roadmap for all stakeholders involved, aiming for a common understanding of the project's goals and constraints.

2.1 Current State Analysis (Problem statement)

Migration technology plays a vital role in the modern technology landscape, enabling organizations to transition their applications from one environment to another or upgrade its current setup. Although migration encompasses various types such as platform, database, and cloud migration, the main emphasis and focus of this study is on application migration and version updates process.

Initially, the financial company's application migration and version update process appear to be proceeding smoothly. However, there were issues or delays in solving the issues that needs to be addressed immediately. The company had to explore strategies that would improve the outcomes. The goal was that there would be a smooth transition with minimal disruptions, an error-free result that meets both server requirements and end-user expectations.

The problem at hand revolves around application migration and or version update, specifically the challenges and complexities organizations face when transitioning their applications from one server or station to another or sending an update to its current setup.

Visual illustrations and representations have been provided to offer an understanding and insight into the current issues:

1. New environment (scenario) –this will happen when bringing Application A to station01 from station02, or introducing Application A to a new station.

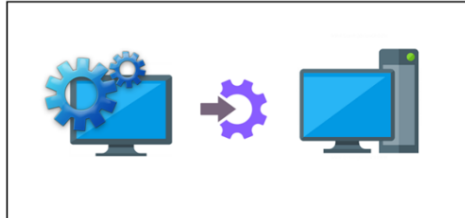


Figure 2: Application to new environment

2. Version update (scenario) –Application A resides on a station which is updated and this has an impact on Application A.

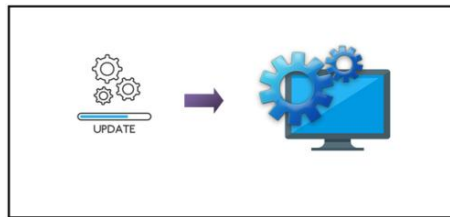


Figure 3: Application version update

During the migration process there could be issues related to compatibility, data integrity, performance, and user experience. These challenges lead to prolonged downtimes, decreased productivity, and increased costs for organizations. It is crucial to deal with these problems effectively to ensure a smooth and successful application migration or update, enabling businesses to leverage new technologies, enhance operational efficiency, and meet evolving client demands.

2.2 Scope

This project, concentrating solely on addressing issues related to application migration and version updates within a particular financial company's server infrastructure and user stations. The intention is to tailor the solutions and strategies specifically to the unique requirements and challenges faced by this company, taking into account the unique characteristics and complexities of its industry. Consequently, exploring other forms of technological migration or different sectors may not yield relevant insights or practical benefits for this study, as the methodologies and considerations may differ significantly. The challenges also do not extend to hardware upgrades or network restructuring. Instead, the emphasis is placed on enhancing software applications and refining user interfaces experience within the company's existing technological framework. By narrowing the scope to the targeted company and its specific needs, the project can create customized solutions that are optimized for the financial industry, ensuring effective and efficient application migration and version updates.

The project's anticipated outcomes are positive, with the newly developed process or techniques integrating to the main procedure, expected to lead to substantial improvements by effectively addressing post-migration issues and challenges. The primary goal is to achieve a significantly enhanced migration experience and elevate organizational efficiency and productivity.

2.3 Objective

The objective of this project is to enhance a comprehensive process that effectively eliminates post-migration issues. The process will encompass a series of well-defined steps and methodologies aimed at addressing common challenges and concerns that arise after the version update or migration of applications. By implementing this process, the goal is to minimize disruptions, ensure smooth transitions, mitigate any

potential issues that may arise during or after the migration process, and rapid issue resolution or within a conservative timeframe.

The developed process will involve careful analysis and assessment of the application's requirements, dependencies, and compatibility with the target environment. It will also include rigorous testing procedures to validate the functionality, performance, and security to all working environment. This approach ensures prompt and efficient identification and resolution of inconsistencies, compatibility gaps, and or configuration errors that may possibly arise during the migration and update process.

2.4 Expectations

The expectation is to smoothly transfer and update financial applications across servers and user stations, ensuring uninterrupted services, enhanced data security, fast issue resolution and improved user experience. This effort aims to align technology with domain expertise, fostering operational efficiency, regulatory compliance, and client satisfaction. The desired outcome is a future-ready infrastructure that bolsters the company's competitiveness and innovation in the financial sector.

3 Methodology

This chapter outlines the research methodology employed to investigate and execute the process of introducing Application A to the latest version or to a new and fresh environment. The methodology describes the approach and steps taken to ensure a smooth application migration and successful system update while minimizing disruptions and maintaining its functionality.

3.1 Research Design

The research employed a comprehensive approach of quantitative analysis and in-depth case study methodologies to thoroughly examine the research objectives related to the challenges, strategies, and outcomes associated with system updates or migrations.

3.2 Tools

JIRA, MS Studio, and 3D Max are essential tools for assessing the effectiveness and efficiency of a carefully designed process, as they each provide valuable functionalities for different aspects of the evaluation.. It ensures that the results achieved closely match the intended objectives and benchmarks. In the following list the benchmark and the objective are explained in detail.

- a) MS Studio – An internally developed application system using MS Visual Studio and the C# programming language will be employed, to assess its smooth operation following the migration or update of the operating system.
- b) Autodesk 3D Max – This was employed to examine and validate cases related image processing. Utilizes 3D computer graphics and animation software provided that concerns such as compatibility with textures and materials, dependencies on plug-ins, and compatibility with file formats do not arise.
- c) JIRA - is an issue tracking and project management software which feature reporting dashboard that provides a quick overview of the project's status. These dashboards serve as a valuable data source for our process improvement graph.

3.3 Approach Method

Qualitative research methods and case studies are employed as an aim to explore, understand, and explain complex situation within their natural contexts. This shows depth, richness, and context-specific insights that are valuable for theory development, decision-making, and enhancing our understanding of actual situation.

3.4 Data Collection

Gather information from various two sources to be used for research.

1.1.1 Primary data collection :

These insights have been gathered through firsthand experience and in-depth discussions with various individuals actively engaged in migration and update tasks.

Table 1: Summary of the activity for data collection for case study 1(FinC X)

Respondents	Topic discussed	Date	Mode
Project Manager	Migration project management	Jan 2022	In-person
Infrastructure Engineer	New environment security, stability, performance and setup	Jan 2022	In-person
System Administrator	Handling on technical aspect on migrating server	Jan 2022	In-person
Database Administrator	Data Integrity, database performance and resolving any database-related issues.	Jan 2022	In-person
Developers/Programmers	Application compatibility to new environment	Jan 2022	In-person
Network Engineer	Network configuration	Sept 2022	Online
Security Expert	Security concern related to migration	Sept 2022	Online

End User	Feedback on migration (UAT and application requirements)	Sept 2022	Online
----------	--	-----------	--------

After the initial discussion and data gathering, I conducted two face-to-face meetings and three online discussions with the infrastructure engineer to delve deeper into the current process and assess the progress of the enhanced process.

Table 2: Summary of the data collection report on case study 2 (Company Y)

Respondents	Topic discussed	Date	Mode
Manager	Project details	Mar 2023	In-person
3D Artist (design team)	3D Techniques, Tools use and experience	Mar 2023	In-person
End User	Experience of the application	Mar 2023	In-person
3D Artist (design team)	User's feedback	Sept 2023	Online
End user	Process implementation follow-up	Sept 2023	Online

A face-to-face meeting with three construction company staff addressed the issue and introduced a solution. Online follow-ups were conducted to gather input on the proposed solution.

1.1.2 Secondary data collection:

The collected data was extracted from JIRA system log, a widely used project management platform that provides comprehensive functionality for issue tracking, task management, and bug tracking. This data source proved to be an invaluable resource for gaining insights into the migration and update tasks process.

3.5 Data Analysis Procedure

In this section, we outline the systematic procedures employed for data analysis in our research study. The collected data underwent careful analysis to draw meaningful insights and support the research objectives.

1.1.3 Data Preparation - Raw data imported from the log tracker into Excel was meticulously processed, guaranteeing uniformity and accuracy. A comprehensive data cleaning procedure was conducted, focusing on resolving duplicates, errors, and missing values to ensure data integrity.

1.1.4 Data Exploration - Sorted and filtered data to target specific subsets, used PivotTables to condense large datasets, and created charts to visualize trends and patterns.

1.1.5 Descriptive Statistics - Employed Excel functions such as SUM, AVERAGE, COUNT, MIN and MAX for basic statistical analysis.

1.1.6 Documentation and Reporting - Thoroughly documented the analysis procedures, including applied methods, formulas, and assumptions made during the process. Prepared comprehensive reports with clear visual representations, making the results accessible and understandable.

3.6 Ethical Considerations

When conducting this research, data privacy and confidentiality were considered, ensuring that sensitive data is protected according to legal and company's regulations.

The anonymity of the research participants, including employees and clients, was

preserved by refraining from using real names or any personally identifiable information in research materials.

3.7 Limitations

While doing company research, challenges like data sensitivity, regulatory compliance, limited access, trust concerns, resource constraints, data quality issues, ethical dilemmas, technological limitations, and organizational resistance impacted the study. In order to overcome these challenges demands careful planning, communication, and flexibility from researchers. Potential challenges were identified and addressed with alternative data collection methods and clear communication strategies to build trust, while maintaining flexibility to adapt research methods and explore alternative data sources based on available data and restrictions.

3.8 Key findings

The study highlights the pivotal role of migration technology in the contemporary technological landscape, particularly focusing on application migration and version updates.

The research initially noted smooth progress in the company's application migration and version update processes. Project size has probably the greatest influence on the process [2]. While a few minor glitches arose at the beginning, and they were easily addressed due to manageable volume of tasks and minimal concerns. However, as time went on, the number of applications increased significantly, leading to a corresponding rise in the number of issues, requiring urgent solutions and actions.

During the study, it was observed that the company's proactive approach involved continuous exploration for application migration and version update strategies, aiming for an admirable outcome. An execution that shows steady transition with at least

minimal disruptions, making an error-free result, aligning with both server requirements and end-user expectations.

The main challenge identified revolved around application migration and version updates, specifically the difficulties encountered when transitioning applications between servers or stations and updating the existing setup. These complexities often lead to problems related to compatibility, data integrity, performance, and user experience, resulting in prolonged downtimes, decreased productivity, and increased costs for the organizations.

Addressing these challenges effectively emerged as a urgently need. Successful resolution enabled the organizations to navigate these complexities, ensuring a smooth process. This achievement holds significant value as it empowers the businesses to embrace new technologies, optimize operational efficiency, and respond proficiently to evolving client demands.

As part of the ongoing efforts to refine the company's procedures to migration and updates, they're actively exploring enhancements, and one of their key initiatives involves integrating a new process, which is the file inventory system into the workflow, and next to that is enhancing the process by implementing checkpoints to ensure file sanity. The additional foundational steps incorporated into the continuous improvement process have yielded numerous valuable benefits, including:

Compatibility - The files that caused the compatibility issues were easily identified, and they were also quickly diagnosed and resolved.

Reduced downtime - Early identification of files related to the application and resolution of compatibility issues help to prevent them from causing downtime for systems.

Data integrity - identify and resolve potential data integrity issues, maintain secure transfers, and establish a stronger audit trail, ensuring continuous monitoring and validation throughout the migration process. Maintaining data integrity by providing visibility into file changes, detecting unauthorized modifications, preventing file conflicts, ensuring compliance with data integrity standards, reducing data loss risk, enhancing data management efficiency, and supporting informed decision-making.

Performance - Effective resolution of issues during migration is achieved through the implementation of a newly integrated process that utilizes file analysis based on size, usage patterns, and dependencies. Additionally, this process contributes to improved system performance by identifying and eliminating unnecessary or duplicate files.

User experience- identifies files critical to user workflows and applications essential to daily operations.

Moreover, it was observed that the newly developed solution enhances troubleshooting capabilities, enabling the team to quickly identify and address issues, resulting in a more functional and efficient process improvement. The enhance process approach did not only optimizes the current operations but also shows a strong foundation for future innovations and organizational adaptability.

4 Theoretical Background

This where an in-depth discussion is presented on the various theories and conceptual frameworks that form the foundational basis for the research, outlining how each of these theoretical elements contributes to the overall understanding and direction of the study.

4.1 Migration

In the context of technology, migration refers to the process of transferring data, applications, or systems from one environment to another. It involves moving resources from an existing infrastructure or platform to a new one, often with the goal of improving performance, scalability, efficiency, or compatibility [13].

Business requirements often serve as the driving force behind migration processes, which aim to achieve objectives like reducing costs, enhancing performance, improving scalability, strengthening security, or adopting innovative technologies. To ensure a seamless transition with minimal impact on operations and data integrity, meticulous planning, comprehensive testing, and well-executed implementation strategies are vital components of successful migration [14].

Undertaking technology migration is a complex process, as applications interact with other applications, and each one has its own data model [15], that brings forth challenges such as compatibility issues, the potential for data loss, and the risk of downtime. As a result, organizations often opt to engage seasoned professionals or rely on specialized migration providers to ensure a smooth and prosperous transition.

4.2 Categories of migration techniques

There are common application migration patterns, categorized to provide a spectrum of approaches for technology migration, allowing organizations to choose the technique that best suits their needs, objectives, and available resources [16]. The choice of migration technique depends on factors such as the complexity of the application, desired outcomes, budget, timeline, and the extent to which the organization wants to leverage the capabilities of the new technology platform. The following strategies are widely adopted by the organization due to their alignment with the company's needs and objectives.

1.1.7 Rehost (Lift and Shift): This technique involves moving an application or system from one environment to another without making significant changes to its architecture or functionality. It focuses on migrating the existing infrastructure, configurations, and data to a new environment. Lift and shift migrations are often employed when organizations want to migrate quickly or minimize disruptions, but they may not take full advantage of the capabilities of the new technology[16].

1.1.8 Replatforming: Replatforming involves migrating an application or system to a new environment while making some modifications to take advantage of the target platform's features and capabilities. It typically involves adapting the application or system to work optimally in the new environment, such as updating software dependencies, reconfiguring components, or optimizing performance. Replatforming strikes a balance between migration speed and leveraging the benefits of the new technology [16].

1.1.9 Refactoring or Re-architecting: This technique involves making significant changes to the application or system architecture during the migration process. It may involve rewriting or reengineering parts of the application to align with the new technology platform, taking advantage of its advanced features, scalability, or performance. Refactoring or re-architecting migrations require more time and effort but can result in a more optimized and efficient system in the long run [16].

4.3 Main stages of migration

The migration process typically involves multiple stages such as assessment, migration execution and optimization[22] this is to ensure a smooth and successful transition from one system, platform, or environment to another [50]. While the specifics may vary depending on the nature and scope of the migration, the following outline provides a general framework for the phases of migration:

4.3.1 Pre-Migration Assessment

is a critical step in the process of migrating data, applications, or entire systems from one environment to another. It involves a thorough evaluation of the current state of the systems, applications, and data that are to be migrated.

Planning and assessment, including (1) defining migration goals, objectives, and requirements, (2) assessing the existing system or environment to be migrated, (3) identifying dependencies, constraints, and potential risks, and (4) developing a detailed migration plan and timeline;

Preparation, which encompasses (1) gathering and preparing the necessary resources, tools, and documentation, (2) establishing

communication channels and coordination among stakeholders, (3) preparing the target environment or system for migration, and (4) creating backups to ensure data integrity;

Data Analysis and Mapping, which involves (1) analyzing the data to be migrated, including its structure, format, and quality, (2) mapping the source data to the target data model or schema, and (3) defining transformation rules, if required, to ensure data compatibility.

Migration is a highly complex undertaking, therefore it requires significant upfront thought and in-depth planning to ensure effective resource use and risk management, on-time and within budget implementation, and ultimately operational success[8].

4.3.2 Actual Migration

Once the migration planning and assessment, preparation, and data analysis and mapping are finished, the actual migration process gets underway. This phase typically includes:

Data Extraction and Conversion, (1) the data is extracted from the source system or environment, (2) converted to the format or structure required by the target system, and (3) then validated and cleaned to ensure its accuracy and integrity.

Data Loading and Integration, the transformed data is (1) loaded into the target system or environment, (2) necessary data validations and integrity checks are performed, and (3) the migrated data is integrated with existing systems or applications.

Testing and Validation, from this phase (1) functional and performance testing is conducted on the migrated system or environment to (2) validate the data's integrity, consistency, and correctness, and to (3) identify and address any issues or discrepancies.

User Acceptance and Training, (1) end-users are involved in the testing and validation process, their (2) feedback is obtained and any concerns are addressed, and (3) training and support are provided to help users adapt to the new system or environment.

Deployment and Go-Live phase, the (1) final migration cutover is planned and executed, the migration (2) process is monitored to ensure minimal disruption, (3) post-migration checks and validations are performed, and (4) a transition to the new system or environment is completed.

4.3.3 Post-migration

To ensure the Migration achieves its objectives and the new system functions optimally, the Evaluation and Optimization phase meticulously (1) evaluates its effectiveness and success, (2) identifies areas for improvement, (3) optimizes performance and functionality, and (4) provides ongoing support and maintenance.

It's important to note that these phases are iterative and interconnected. Close collaboration between stakeholders, effective project management, and continuous monitoring are critical for a successful migration process.

4.4 Various Type of Migration Technology

4.4.1 Data Migration:

Morris (2020, p. 9) describes it as “the process of transferring data from one system, storage medium, or repository to another, often as a one-off movement from old systems to a new repository”[9],[8]. It may include restructuring or reformatting data to fit the target system's requirements, ensuring data integrity and minimizing disruptions during the transition. Data migration requires careful planning, attention to detail, and rigorous testing to minimize the risk of data loss, ensure data integrity, and maintain business continuity [19]. It is often facilitated by specialized tools and technologies designed to streamline and automate the migration process. It is typically performed when organizations upgrade their systems, change databases, consolidate data from multiple sources, or transition to new platforms or environments.

4.4.2 Application Migration:

The migration of system applications is increasingly emerging as a significant concern fraught with various challenges. This migration involves transitioning from one operational environment to another, with the goal of maintaining similar functionality. The migration process encompasses several crucial aspects, including ensuring the full utilization of the features offered by the new environment, maintaining existing settings without necessitating alterations, and ensuring seamless continuity of operation for current applications [10].

4.4.3 Infrastructure Migration:

Moving an entire IT infrastructure, including servers, networks, storage, and other hardware components, to a new location or platform. It involves

transitioning from on-premises infrastructure to a cloud-based infrastructure. This requires meticulous planning, risk assessment, and collaboration between different teams involved in the process. It is essential to consider factors such as compatibility, scalability, security, and business continuity to ensure a successful and seamless migration with minimal impact on operations [17].

4.4.4 Platform Migration:

The process of transitioning from one technology platform to another. It involves moving an application, system, or entire infrastructure from the current platform to a new platform that offers different features, capabilities, or benefits. For example, migrating a website or web application from one content management system (CMS) to another, or transitioning from one development framework or programming language to another. Platform migration requires careful planning, coordination between different teams, and thorough testing to ensure a successful transition [18]. It is important to consider factors such as compatibility, functionality, scalability, security, and long-term viability when selecting a new platform.

4.5 Migration Challenges

There were several challenges and obstacles that organizations encounter when transitioning from one technology platform, system, or environment to another [20]. It encompasses the difficulties, complexities, and potential problems that arise during the migration process. These issues can arise due to various factors such as technical constraints, compatibility challenges, resource limitations, user adoption concerns, and security risks [21], [39].

4.5.1 Compatibility Issues:

When migrating from one technology platform to another, compatibility issues may arise. This can include software or hardware incompatibilities, different data formats, or changes in programming languages. Ensuring seamless integration between the existing and new technology is crucial to avoid disruptions or loss of functionality[22].[39].

Common compatibility issues that can be encountered during migration:

- a) *Software Version Incompatibility:* When migrating software applications, compatibility issues can arise if the target environment requires a different version of the software [23]. This can result in incompatibilities with the application code, dependencies, libraries, or APIs. Upgrading or modifying the software to align with the target environment may be necessary.
- b) *Hardware Compatibility:* Migration to a new hardware platform can present challenges if the target hardware has different specifications, architectures, or operating systems [24],[25]. Incompatibilities may arise with device drivers, peripheral connections, or system configurations. Ensuring hardware compatibility and, if necessary, obtaining or updating drivers or hardware components may be required.
- c) *Data Format and Structure:* Migration of data between systems can encounter compatibility issues when the source and target systems have different data formats, structures, or encoding methods. This can result in data loss, data corruption, or inaccessibility[21]. Data transformation, mapping, and validation processes need to be implemented to ensure data compatibility during the migration.

- d) *Network and Protocol Compatibility*: In cases where migration involves network-dependent systems or communication protocols, compatibility issues may occur. Differences in network configurations, protocols, or security settings can lead to communication failures or performance issues. Adjustments to network settings or implementing protocol translation mechanisms may be necessary [44].

- e) *Legacy System Support*: When migrating from legacy systems to modern technologies, compatibility issues may arise due to outdated software, proprietary formats, or unsupported features. Ensuring that the target environment supports legacy system integration or finding suitable replacements or workarounds may be required [45], [46].

- f) *Third-Party Software Dependencies*: Migration may involve software applications or systems that rely on third-party libraries, plug-in, or components. Incompatibilities can occur if the target environment does not support or has different versions of these dependencies. Ensuring compatibility of third-party software or finding alternatives that are compatible with the target environment may be necessary [47].

- g) *Security and Access Control*: Migration can introduce security and access control challenges, especially if the target environment has different authentication mechanisms, encryption standards, or security protocols. Ensuring data security, user access, and compliance with security policies may require adjustments to security configurations or implementing compatible security measures [48].

4.5.2 Data Loss and Integrity:

During the migration process, Nauduri, Laszewski, and Williamson, stated that "as data volumes have increased and data quality has become an issue,"[3] there is a risk of data loss or corruption. Therefore, it is essential to implement robust data backup and recovery mechanisms to minimize the chances of data loss and ensure data integrity during the migration.

Common causes of data loss and integrity issues during migration:

- a) *Insufficient Planning*: Inadequate planning and preparation for the migration process can lead to data loss or corruption. Without a well-defined migration strategy, data may be mishandled or overlooked, resulting in incomplete or inaccurate transfers [37],[38].
- b) *Incompatible Data Formats*: Differences in data formats between the source and target systems can cause data loss or corruption during migration. If the migration process does not account for these format discrepancies or lacks proper data transformation mechanisms, data integrity may be compromised [38].
- c) *Inadequate Data Validation*: Insufficient validation processes during migration can result in the transfer of incorrect or inconsistent data. Failure to perform thorough data validation checks can lead to data loss or integrity issues that may go unnoticed until after the migration is complete [38].
- d) *Inadequate Backup and Recovery Procedures*: Lack of proper backup and recovery mechanisms can increase the risk of data loss during migration [39].

Without reliable backup systems or recovery strategies, it becomes difficult to restore data in case of accidental deletion, corruption, or other unforeseen events [40].

- e) *Security Breaches*: Data loss and integrity issues can also arise due to security breaches during migration. Unauthorized access, data interception, or tampering can compromise the confidentiality, integrity, and availability of the data being transferred [41].
- f) *Human Errors*: Human errors, such as incorrect data mapping, accidental deletion, or mishandling of data, can contribute to data loss or integrity issues during migration. Lack of attention to detail or inadequate training on migration processes can increase the likelihood of such errors [42].
- g) *Inadequate Testing*: Insufficient testing of the migration process can lead to unforeseen data loss or integrity issues [39]. Without thorough testing, it becomes challenging to identify and address potential problems before the migration is implemented [43].
- h) *Resource Limitations*: Inadequate resources, such as insufficient storage capacity or network bandwidth, can contribute to data loss or integrity issues during migration. If the migration process exceeds the available resources, data transfers may be incomplete or compromised [49].

4.5.3 Downtime and Disruption:

Technology migration often requires downtime or temporary disruptions in service. It is important to plan and manage the migration process carefully to

minimize the impact on operations and end-users[26]. Effective communication and coordination with stakeholders can help mitigate the negative effects of downtime.

4.5.4 Resource Constraints:

Technology migration may require significant resources, including financial resources, technical expertise, and infrastructure. Organizations need to allocate adequate resources to ensure a smooth migration process and address any resource constraints that may arise [27].

4.5.5 User Adoption and Training:

Introducing new technology to users requires proper training and support. Resistance to change or lack of understanding of the new technology can affect user adoption and productivity. Providing comprehensive training and support to end-users can help them adapt to the new technology effectively [28],[29].

4.5.6 Security Risks:

Technology migration can introduce new security risks if not properly managed. It is important to assess and address potential vulnerabilities in the new technology and ensure that appropriate security measures are in place to protect data and systems[30],[31],[39].

4.5.7 Vendor or Supplier Dependencies:

Migration to a new technology platform may involve dependencies on specific vendors or suppliers. Organizations need to consider the long-term viability and support from these vendors to mitigate risks associated with vendor lock-in or the discontinuation of support [32],[33].

4.5.8 Performance and Scalability:

To achieve the performance and scalability requirements outlined in the scalability modeling process, it's essential to assess the scalability of the new technology platform. This assessment will ensure it can accommodate future growth and maintain optimal performance under varying workloads [34],[35].

4.5.9 Regulatory and Compliance Considerations:

Technology migration may involve compliance with specific industry regulations or data protection laws. It is important to ensure that the new technology meets the necessary regulatory requirements and safeguards sensitive data appropriately [36].

4.6 The process modifications

Refers to the act of changing, updating, or altering the existing procedures and methods within an organization's operations. This can involve introducing new technologies, redefining workflow steps, eliminating inefficiencies, or integrating new practices to improve performance, quality, compliance, or efficiency. The goal of process modification is often to enhance overall productivity, adapt to changing market demands, ensure compliance with updated regulations, or improve customer satisfaction. Most operational problems within an organization are linked to processes or a series of activities. Thus, solving these problems effectively requires a focused effort on improving these processes [4].

4.7 Defining process performance

Process performance refers to the measurement and assessment of a process's effectiveness and efficiency in meeting its intended objectives. It encompasses various dimensions, including the quality of outputs, the speed and reliability of process execution, the cost-effectiveness of operations, and the level of customer satisfaction generated. Essentially, process performance is about evaluating how well a process achieves its goals, how it utilizes resources, and how it contributes to the broader objectives of the organization [5].

Table 3. Key aspects of defining and assessing process performance [6]:

Key aspects of defining and assessing process performance	
1	Quality: The degree to which the process outputs meet the required standards or expectations. . This can involve error rates, defect levels, or adherence to specifications.
2	Speed and Timeliness :The duration it takes for the process to complete its cycle, from start to finish, including the punctuality of delivering outputs.
3	Cost :The financial measure of the process, encompassing the total costs involved in executing the process, with a focus on identifying opportunities for cost reduction and control
4	Flexibility: The ability of the process to adapt to changes, whether in volume, variety, or other external factors, without significant loss in performance

4.8 Process improvement and innovation

The transformative potential of Information and Communication Technology (ICT) in organizations extends beyond mere technological upgrades [7]. Additionally, posits that for ICT to catalyze genuine organizational transformation, it must be strategically integrated with process improvements and innovations that align with the organization's broader goals [7].

Some key principles for maximizing the transformative impact:

- (1) Integration of Process Improvement with ICT Investments: Simply investing in ICT is not enough. For organizations to truly transform and derive substantial economic benefits, a rigorous focus on improving processes is essential.
- (2) Collaboration Between ICT and Business Functions: There's often a disconnect between those working in ICT and those overseeing various business functions. Bridging this gap by focusing jointly on process improvement is essential for successful organizational transformation.
- (3) Lean Six Sigma as a Catalyst for Innovation: Adopting methodologies like Lean Six Sigma is highlighted as crucial for bringing about significant economic returns from ICT investments as these methodologies encourage efficiency and continuous improvement in processes.
- (4) Adoption of the Lean and Digitize Approach: Introducing a new methodology, Lean and Digitize, suggests that radical changes in perspective are necessary for maximizing the value of ICT investments. This approach underlines the importance of both improving processes and correctly implementing digitization.
- (5) Economic Returns Require More Than Just ICT: It underscores that without a concerted effort to improve processes, investments in technology alone cannot assure economic returns. Real and substantial benefits come from combining process improvement with strategic digitization.

Fundamentally, the takeaway is that entities must transcend mere capital allocation in technological assets and proactively participate in enhancing and innovating processes to achieve substantial economic advantages from Information and Communication Technology (ICT).

5 Implementation

This is a critical phase in the lifecycle of a project or plan, where the carefully crafted strategies, plans, and designs are put into action to achieve the desired goals and outcomes. Undertaking the real-world execution of the proposed process is absolutely essential, as it stands as a fundamental milestone in affirming its effectiveness. By actively putting the process into practice, it did not only generate tangible evidence but also validate its efficacy, unequivocally showcasing its capacity to achieve the intended goals and deliver the desired outcomes with undeniable certainty.

5.1 Objective and Scope

The objective of this phase is to evaluate the efficacy of the proposed solution in eliminating the problem associated with application migration and version updates that leads to the failure of some server jobs resulting inefficiency to serve its client. This test plan covers two use cases:

- Integration of the inventory process into the primary application flow.
- Execution of the inventory process as standalone functionality.

5.2 Earlier Migration Process

This is the earlier migration process as described, where the organization utilized a simplistic strategy for software deployment and updates, relying on manual intervention and fundamental tools. This initial approach met the needs of the organization during its early stages, when managing the volume of software deployment and updates was considerably straightforward.

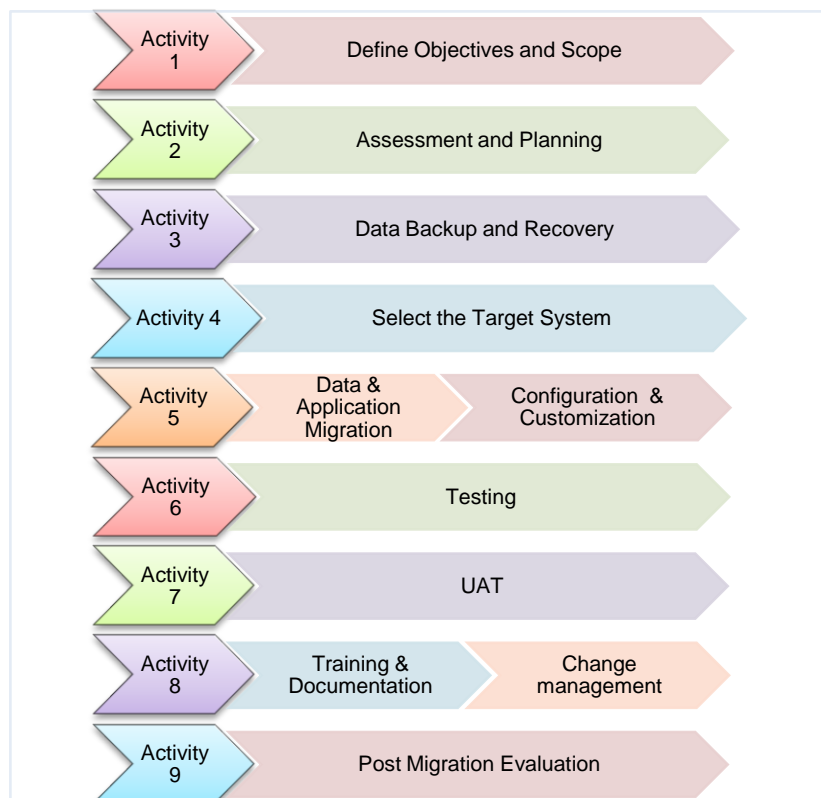


Figure 4: Pre-update process

Figure 3: Visual representation that illustrates the current process, detailing each step from initiation to completion and highlighting the sequence and interactions involved.

5.3 Enhanced process (Implemented)

“The complexity of the restructuring task is often underestimated”[1]. This underestimation can stem from a variety of factors, including a lack of understanding of the interconnectedness of organizational processes. Before we start working on the revision process, it's important to understand how things are currently done. We can do this by carefully examining the existing process step-by-step. Following a thorough review, we determined that integrating a new sub-process into the main workflow would be the best approach to address the identified issue. To strengthen the existing application migration process, we'll be adding two new steps: a file inventory procedure and sanity file check. The organization will follow this plan as part of the ongoing work to improve the system migration and update process testing.

5.4 Detailed of the newly added process

As part of strengthening the existing process, two additional procedures were integrated into the main workflow:

5.4.1 File Inventory Procedure:

This sub-procedure, integrated within the main process, establishes a structured method for systematically cataloging, overseeing, and tracking all application related files within the organization. It ensures that each file is accounted for, easily accessible, and properly managed to achieve optimal efficiency and security. This system serves as a central repository for details such as file names, locations, sizes, creation dates, owners, and other relevant attributes.

5.4.2 Sanity checking

This procedure was incorporated to verify the proper placement and versioning of the corresponding files.

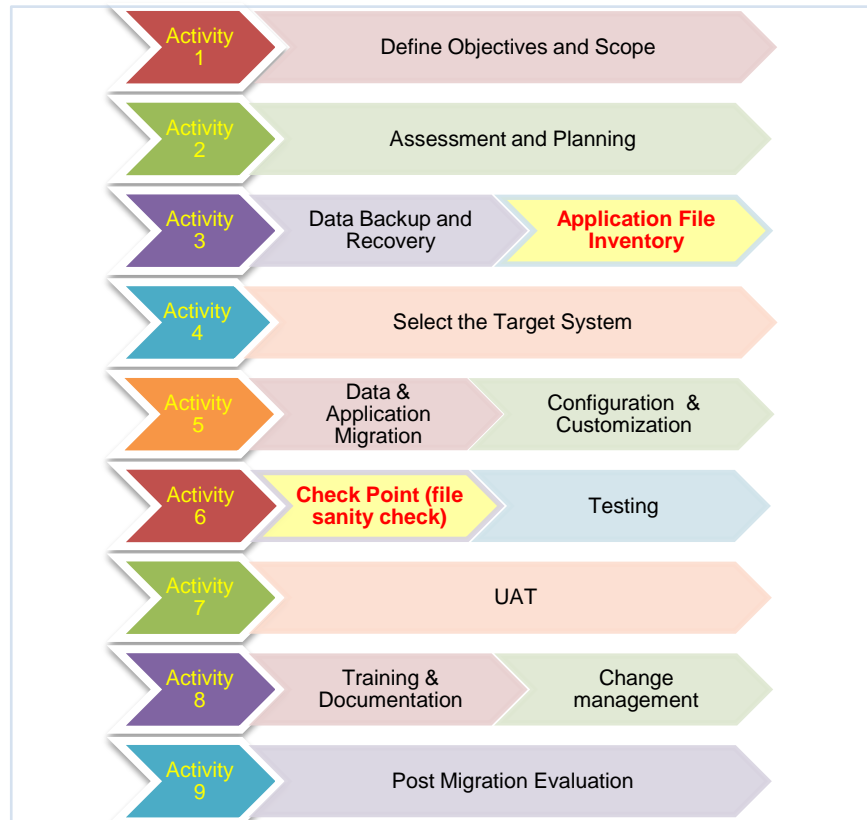


Figure 5: Enhance migration and update of application process

Figure 4 details the additional processes incorporated into the existing framework. These include two new procedures; file inventory procedure and sanity checking, which have been integrated into the main workflow to enhance its functionality and effectiveness in supporting the migration and version update activity.

5.5 Case Studies

In our current research, we are delving into two separate use-case scenarios, each associated with distinct sources (company), and characterized by its unique set of circumstances and requirements. However, it's noteworthy that despite these

differences, both scenarios are being approached with the same implementation process. This uniform approach allows for a consistent and comparative analysis, facilitating a comprehensive understanding of how the implementation process functions in varying organizational contexts. Furthermore, it underscores the significance of the file inventory activity within the broader context of issue resolution.

The incorporation of a checkpoint process into the procedural framework is an intentional augmentation aimed at elevating the holistic effectiveness of the entire operational sequence. This strategic addition serves a multifaceted purpose, encompassing the assurance of quality, systematic management of risks, meticulous monitoring of progress, facilitation of informed decision-making, optimization of resource utilization, cultivation of an environment conducive to continuous improvement, client satisfaction through iterative feedback loops, and the cultivation of transparent communication channels within the project team and with external stakeholders. The checkpoint, as an integral component, functions as a pivotal control point that not only validates the adherence to predefined standards but also provides a structured mechanism for ongoing evaluation, refinement, and alignment with the overarching objectives of the project. This comprehensive approach strengthens the commitment to excellence, risk mitigation, and stakeholder satisfaction, ultimately contributing to the overall success and efficiency of the undertaken process.

5.6 Case Study 1

For Case 1, we utilized one of the application system created in .Net with and interface to SQL . However, our focus was exclusively on the report generation module (report menu). This particular module was selected for its ability to illustrate the dependency on dynamic-link libraries (DLLs) during the occurrence of an action event, which is initiated by a single button click within the system menu. This scenario was chosen due

to its straightforward nature, representing a clear and direct instance of how action events interact with DLL dependencies. By focusing on this module, we aimed to provide a detailed examination of the system's functionality in handling dependencies, observing its capability to execute such operations efficiently and accurately in a user-friendly manner. This DLL file is cataloged within our inventory, alongside its associated application name and various other relevant properties.

Scenario: Version update

Description: The existing system, which had been operating under .NET Framework 2.0, underwent an upgrade to a more advanced version framework.

Use Case Name: Generate and Display Report in Reporting Dashboard

Actors:

- User (Primary Actor)
- Reporting Dashboard (System)

Preconditions:

- The Reporting Dashboard application is installed and running under .Net 2.0 framework version.
- The SQL database connection details and query parameters are configured in the application.

Main Flow of Events:

1. The user launches the Evaluation system application.
2. The user select the Employee Summary on Report menu.
3. The system automatically connects to the SQL database using the configured connection details.
4. The system prepares and executes a predefined SQL query.

5. The system retrieves the data from the database query result.
6. The system generates a Report form based on the retrieved data and a predefined report template.
7. The report is immediately displayed to the user in the Reporting Dashboard.

Employee ID	First Name	Last Name	Email Address	Job Title	Department	Start Date
				Accountant	Human Resources	20/10/2020
				IT Specialist	IT	17/08/2022
				Sales Representative	Human Resources	24/04/2023
				Data Analyst	Finance	15/05/2020
				Project Manager	Engineering	17/08/2022
				Data Analyst	Finance	17/08/2022
				Accountant	Sales	27/06/2020
				Customer Service Representative	Sales	21/11/2023
				Customer Service Representative	Finance	08/11/2020
				IT Specialist	Finance	22/06/2023
				Accountant	Finance	17/08/2022
				Data Analyst	Sales	12/08/2020
				Project Manager	Human Resources	25/04/2021
				Sales Representative	Sales	21/11/2023
				Marketing Manager	Marketing	04/05/2021
				Director of Engineering	Engineering	17/08/2022

Figure 6: Reporting dashboard

Prior to the framework update, the reporting dashboard successfully shows inactive employee data in its extractions, which constituted a sample set we used for report generation.

Post-Condition:

The user can view and interact with the generated report within the Reporting Dashboard.

Alternative Flow:

Report Generation Error (Step 2 or Step 5): If there are issues with generating or retrieving the report data, the system displays an error message and provides guidance on resolving the issue.

Issue / error message: The error message "Could not load file or assembly" occurs at runtime when application is executing and is unable to locate or load a specific assembly (DLL file) that it depends on.

File involve for the scenario: Dll file (System.Data.dll)

Additional information on error: When you encounter the error message "Could not load file or assembly" when running .NET code, it typically indicates that there is a problem loading or referencing a specific assembly (DLL file) that your application depends on.

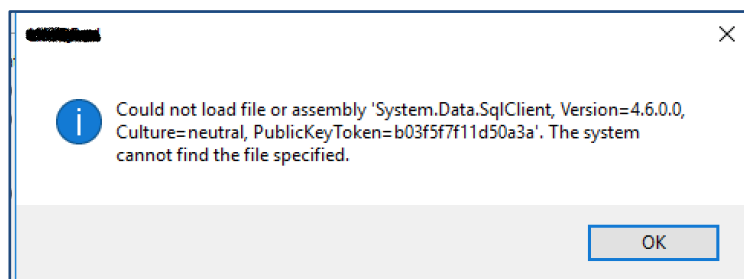


Figure 7: Error for file or assembly missing

Analysis

The application relies on external assemblies (DLL files) containing crucial code, resources, or components necessary for its operation. These assemblies may be either third-party libraries or custom assemblies tailored for the application's specific requirements. During runtime, these files can be either absent or incompatible, often as a result of version updates; this is the reason why we encounter the 'Could not load file or assembly' error.

Solutions or strategy to resolve the issue:

1. Ensure that the missing DLL file is present in the application's directory or in a directory specified in the application's search path. Refer to the inventory file.

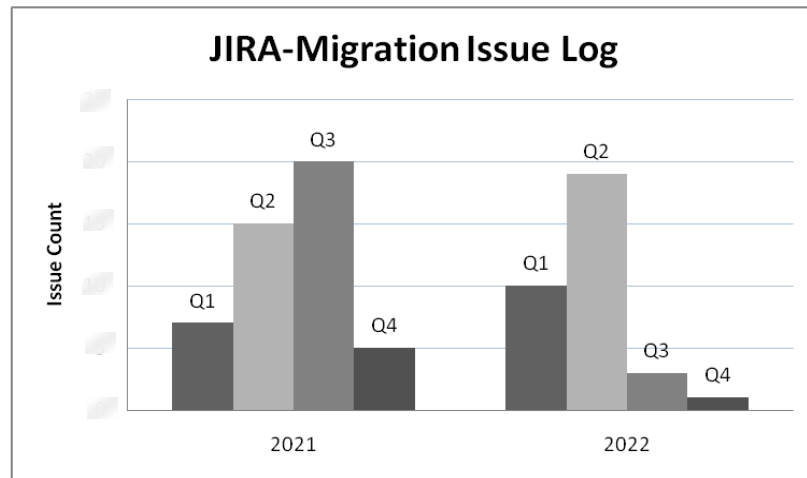
2. Verify that the assembly references in your application match the available versions of the assemblies in your runtime environment.
3. Use assembly binding redirects in your application's configuration file to redirect references from one version of an assembly to another compatible version.
4. Verify whether the missing DLL is included in the .NET Framework, and confirm that the appropriate framework version is installed. If not, copy or register the correct version from the file inventory location.

Therefore, when you encounter a "Could not load file or assembly" error, it indicates that your .NET application is unable to locate and load a specific assembly (DLL file) that is expected to be part of the application's dependencies. Resolving this issue underscores the utmost importance of conducting a meticulous file inventory, ensuring that all necessary DLLs are not only accessible but also accurately referenced within your application. This comprehensive inventory process serves as an essential step in troubleshooting and rectifying such errors to maintain the smooth operation in the application.

5.7 Analysis on gather data

To comprehensively assess the effectiveness of the process, we extracted logs from JIRA and conducted an analysis comparing the frequency of migration and update issues before and after implementing the process enhancements. It's important to note that while this analysis provides valuable insights; other factors may also influence the outcomes.

5.7.1 Issue Graph:



The graph illustrates the migration issue from 2021 to 2022. According to the data, the third quarter of 2021 had the highest number of migration issues within that year. Additionally, it is noteworthy that the second quarter had consistently high numbers in both years. However, in the third quarter of 2022, the incidents decreased significantly, even when compared to previous quarters within that year. This decline can be attributed to the improved implementation of the processes.

5.7.2 SLA(Service Level Agreement) Graph:

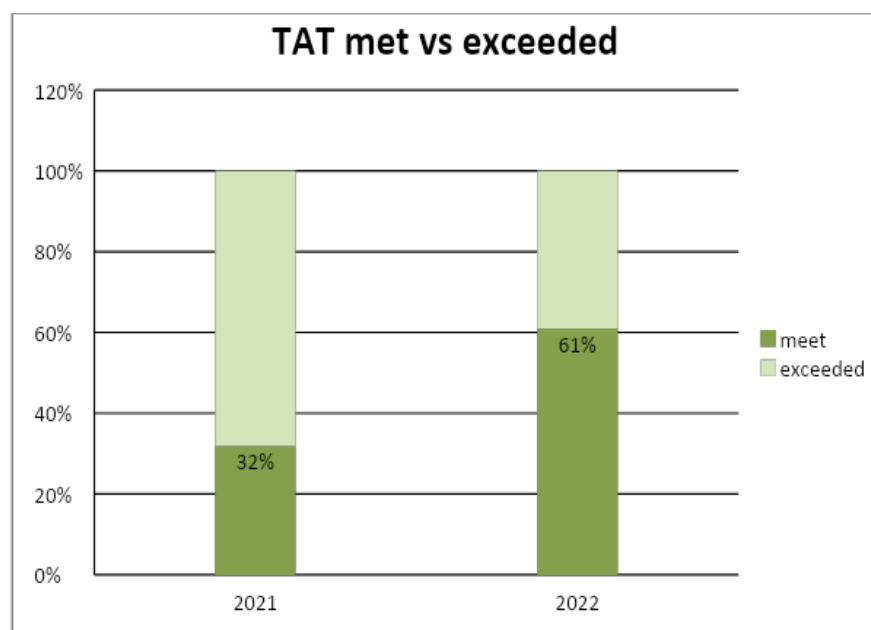


Figure 9: Turnaround Time

In 2021, the provider's adherence to the TAT stood at a mere 32%, with 68% of cases exceeding the agreed-upon service level standards, indicating a comparatively low compliance rate. However, there was a remarkable improvement in 2022, with the provider meeting the TAT in 61% of cases and exceeding it in 39% of instances. This highlights a substantial enhancement in their service delivery efficiency, underscoring their strong commitment to fulfilling the service level requirements outlined in the SLA.

5.8 Case Study 2:

In this case study, we explore the process of rendering a 3D image and transferring it to a new environment. This involves crafting the image in a 3D modeling software, with accurate textures, lighting, and details are incorporated. Once rendered, the image is then copied to a different environment while preserving its integrity and quality, ensuring it maintains its visual impact and effectiveness across different platforms.

Scenario: File migration to new environment

Use Case: Render, Transfer, and address texture issue with 3ds Max Image on
another Machine

Actors:

- 3D Artist (Primary Actor)
- 3ds Max Software (System)
- Second Machine User (Secondary Actor)

Preconditions:

- The 3D artist has a 3D scene or model created in 3ds Max with textures applied.

- The 3ds Max software is installed and configured on the artist's machine.
- The second machine user has a compatible image viewer or software installed.

Main Flow of Events:

- The 3D artist launches the 3ds Max software on their machine.
- Within the 3ds Max environment, the artist opens the 3D scene or model that they want to render, which includes textured elements.
- The artist sets up the scene, including camera angles, lighting, materials, and rendering settings.

Rendering the Image:

- The artist configures the rendering settings within 3ds Max, specifying output quality, resolution, and other parameters.
- The artist initiates the rendering process by clicking the "Render" button or using a rendering command within 3ds Max.
- Once the rendering is complete, the artist saves the rendered image, including the textures, to a location on their machine.



Figure 10: Successfully rendered image

Transferring the Rendered Image:

- The artist copies the rendered image file, to a storage medium (e.g., USB drive) or a cloud storage service to facilitate transfer to another machine.

Opening the Rendered Image on Another Machine with Texture Issue:

- The second machine user obtains access to the storage medium or cloud storage service where the rendered image file is stored.
- The second machine user copies the rendered image files from the storage medium to their own machine.
- The second machine user attempts to open the rendered image using a compatible image viewer or software on their machine.
- The second machine user encounters a texture issue, such as missing textures or incorrect texture paths.



Figure 8: Image with missing texture

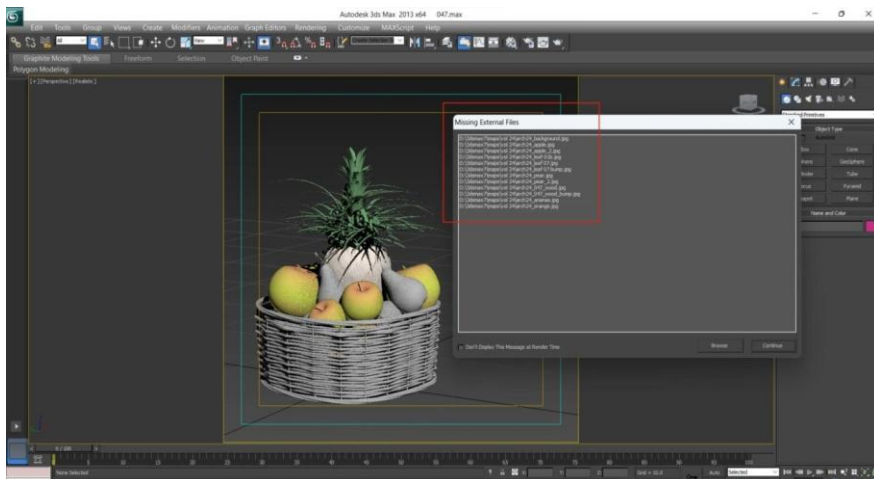


Figure 9: Show warning on missing texture or file mapping

Analysis:

By default, in 3ds Max, rendered image files and texture files are located in different directories. When you render an image in 3ds Max, the rendered image file is typically saved to a specific output directory that you specify during the rendering settings configuration. This is usually a separate directory from where your original 3D scene and texture files are located.

Texture files, on the other hand, are often referenced within your 3D scene, and their paths can be set to point to specific directories where the textures are stored. These texture files can be in various formats (e.g., JPG, PNG, TIFF) and are usually separate from the rendered image files.

Upon encountering the texture issue, the second machine user took proactive steps to identify and rectify the problem. They discerned that the root cause of the issue was the absence or incorrect paths of certain texture files referenced within the 3D scene.

Solution

To resolve this texture issue effectively, the user embarked on a file inventory task, meticulously searching for the missing or improperly referenced texture files. Once located, the user thoughtfully organized and placed these texture files in the correct directories on their machine.

The detailed file inventory and organizational procedure enabled the user to promptly and confidently resolve the texture problem. Subsequently, upon reopening the rendered image, the user was pleased to find that the image displayed correctly, with all textures applied as intended. This comprehensive solution demonstrates the user's

dedication to ensuring an unblemished rendering experience and underscores the importance of thorough file management when working with 3D images.

Suggestion:

In light of the significant advantages witnessed, the company is actively exploring the conversion of its existing inventory file process into a sophisticated application system. The plan to transition from manual inventory file management to an automated application system stems from the substantial benefits experienced by the organization while studying and implementing the process. Through meticulous analysis, the company has observed a remarkable improvement in operational efficiency, accuracy in data management, and timely decision-making.

6 Summary

This case study examines financial and construction company's approach to overcoming application migration and version update challenges and issues. It explores the implemented solutions, their impact on the organization's performance, and the lessons learned.

6.1 The study

When executing migrations or updates, inevitable challenges and issues arise. To conquer these hurdles swiftly and efficiently, it is imperative to develop robust strategies within your process. Rapid and effective resolution is not only possible but imperative. This can be achieved through keen analysis and systematic approach, organizations can navigate migration challenges effectively, ensuring a smooth transition while learning from diverse scenarios encountered during the process.

The case study emphasizes the context-specific nature of organizational processes. It acknowledges that while not all processes are universally applicable, there are generic parts of the processes adaptable across different context. This was achieved by studying different use cases and scenarios and applied the same methodological approach, ensuring flexibility and effectiveness in diverse situations. By observing how these processes function in specific situations, organizations can identify what works well and where improvements are needed. This direct evaluation offers valuable insights, helping organizations make well-informed decisions and strategic improvements to enhance overall organizational performance.

In the world of organizational processes, it's crucial to recognize the variety of organizational needs while also acknowledging the presence of adaptable, generic processes. To assess the effectiveness of these processes, practical scenarios play an essential role. These scenarios not only showcase the strengths of a developed process but also pinpoint areas for improvement. This valuable feedback empowers organizations to refine their methods and streamline operations more efficiently, contributing to ongoing organizational success.

6.2 The challenges

In the context of data collection for a financial company, two primary challenges emerge prominently—limited access and ethical considerations. The financial sector, characterized by sensitive and confidential information, often encounters obstacles in obtaining comprehensive datasets due to legal constraints, privacy regulations, and proprietary concerns. Limited access to financial records, transaction details, and proprietary algorithms poses a significant hurdle, impacting the depth and accuracy of research findings. Simultaneously, ethical considerations in handling personal financial data become paramount, requiring meticulous attention to informed consent,

transparency, and confidentiality. Striking a delicate balance between obtaining valuable insights and safeguarding individual privacy is crucial, necessitating the establishment of robust ethical frameworks and transparent data access policies within the financial industry.

The challenges encountered when considering a construction and interior design company for a case study include the intricate planning and coordination required due to the simultaneous management of both construction and design aspects, posing a significant challenge in project complexity. Additionally, the tight timelines associated with construction and interior design projects present difficulties in analyzing and documenting within specific timeframes for a case study, particularly when unexpected delays occur. Furthermore, maintaining client confidentiality emerges as a challenge, as many clients prioritize privacy in their projects, making it difficult for case studies to share specific project details or client testimonials without potentially breaching confidentiality agreements.

6.3 Future improvements

Case study 1:

Improve the approach for cloud computing migration. To strengthen the operations, it is strongly recommended to conduct a thorough assessment of the modified processes, specifically concentrating on how well they can adapt to and support cloud migration. This suggestion entails a meticulous review of our existing enhance procedures to gauge their alignment with the ever-changing requirements and functionalities inherent in cloud-based environments. This involves a careful examination of each process step, aiming to pinpoint potential enhancements or optimizations that seamlessly align with the fundamental principles and best practices crucial for a successful transition to the cloud. This rigorous analysis is geared towards ensuring that the chosen enhance

processes will not only cater to the immediate needs of the organization but also actively contribute to a more effective and streamlined shift to the cloud infrastructure.

Case Study 2:

Converting business process into an application system. In considering future improvements to our existing process, the client has recommended a transformative shift by converting it into an application system. This strategic move is envisioned to bring about enhanced efficiency, streamlined workflows, and improved data accuracy. The transition from a manual process to an application aligns with the client's forward-thinking approach and commitment to embracing technological advancements. The proposed application system promises benefits such as real-time tracking, automated data entry, and seamless integration with other essential business systems. By leveraging technology, this initiative seeks to optimize resource utilization, mitigate errors, and facilitate a more agile and responsive operational environment. Furthermore, the application system is anticipated to provide a user-friendly interface, fostering increased user adoption and satisfaction. This forward-looking approach not only addresses current operational challenges but also sets the stage for a more adaptive, technologically advanced, and efficient organizational framework.

Glossary

.NET Framework -a software framework developed by Microsoft that provides a comprehensive programming and runtime environment for building and running Windows applications.

3D Max - a professional 3D computer graphics software developed by Autodesk. It is widely used in various industries such as architecture, design, gaming, film, and visual effects for creating 3D models, animations, and renderings.

Actor - The external entity (such as a user or another system) that interacts with the system.

Alternate Flows - Variations or alternative paths that might be taken to achieve the goal, including exceptional or error conditions.

Application Inventory - is a dynamic document that catalogs all software applications within an organization, detailing their deployment across various environments (on-premises, cloud, SaaS), along with information on licensing, user data, and other relevant metadata, aiming to enhance operational security, efficiency, and compliance [12].

Assembly/ assemblies - fundamental unit of deployment and versioning. It encapsulates compiled code, metadata, and other resources necessary for a .NET application to run.

Cloud storage - service that lets you store and retrieve digital data, like files and photos, online. Instead of saving files locally, it uses remote servers, allowing access from any internet-connected device.

Jira - is a popular project management and issue tracking software developed by Atlassian.

Libraries - are collections of pre-written code and functionalities that developers can use to perform common tasks without having to write the code from scratch.

Goal or Objective - The specific task or goal that the user or system is trying to accomplish.

Main Flow - A sequence of steps or interactions that represent the normal or most common scenario for achieving the goal.

Pre-Conditions - Conditions that must be true before the use case is initiated.

Post-Conditions - Conditions that must be true after the use case is completed successfully.

Rendered image - is a visual representation of a 3D scene or model that has undergone the process of rendering.

Rendering - creating a 2D image or animation from a 3D scene involves simulating lighting, shadows, and materials.

Reporting dashboard - a visual representation of key performance indicators (KPIs), metrics, and data points that provide a quick and consolidated view of an organization's performance.

Texture (image) - refers to a 2D image or procedural pattern that is applied to the surface of a 3D model to enhance its visual appearance.

Scenario - specific sequence of actions or events that illustrates how a system interacts with an external entity, typically a user or another system, to accomplish a specific goal.

Stakeholders - are individuals, groups, or entities that have an interest or concern in a particular organization, project, or issue.

Use Case - In technology and software development, it is refer to a description of how a system interacts with an external entity (such as a user or another system) to accomplish a specific goal.

References

1. Cameron E., and Green M.(2004). Making Sense of Change Management.London: Kogan Page
2. Maciaszek L. (2001) Requirements Analysis and Systems Design (Developing Information Systems with UML). Canada: Pearson Addison Wesley
3. Nauduri P, Laszewski T, Williamson J.(2011). Oracle Information Integration, Migration, and Consolidation : The Definitive Guide to Information Integration and Migration in a Heterogeneous World. Birmingham: Packt Publishing, Limited;
4. Rama,S.(2008). Process Improvement Using Six Sigma.: A DMAIC Guide.Milwaukee: ASQ Quality Press
5. Jan vom B., Rosemann M.(2014). Handbook on Business Process Management 2 : Strategic Alignment, Governance, People and Culture. Springer Berlin : Heidelberg, (page 288)
6. Jan vom B., and Rosemann M.(2014). Handbook on Business Process Management 2 : Identification of Performance Indicators: Strategic Alignment, Governance, People and Culture. Springer Berlin : Heidelberg. (page 231)
7. Nicoletti B.(2012), Lean and Digitize : An Integrated Approach to Process Improvement, Taylor & Francis Group, (page 28)

8. Kearns K. D.(2017). Planning & Management Methods for Migration to a Cloud Environment, MITRE
9. Morris J.(2020). Practical Data Migration, BCS Learning & Development Limited UK
10. Shahbaz G., Razzaq A., Ullah S., and Salman A.(2018). Matrix Clustering based Migration of System Application to Microservices Architecture International Journal of Advanced Computer Science and Applications; West Yorkshire Vol. 9, Iss. 1, DOI:10.14569/IJACSA.2018.090139
11. Lawrence A. and Simon L.(2023). Annual outages analysis 2023:The causes and impacts of IT and data center outages. Retrieved January 05, 2024, from <https://datacenter.uptimeinstitute.com/rs/711-RIA-145/images/AnnualOutageAnalysis2023.03092023.pdf>
12. Faddon.(2021). Application Inventory: Understanding the Importance. Retrieved January 23, 2024, from <https://faddom.com/application-inventory/>
13. Faddon.(2021). Application Migration: Risks, Strategies, and Best Practices. Retrieved January 05, 2024, from <https://faddom.com/what-is-application-migration/>
14. Faddon.(2021). Application Migration: Risks, Strategies, and Best Practices. Retrieved January 05, 2024, from <https://faddom.com/what-is-application-migration/#one>

15. Gillis, S. A.(2024). Data Migration. TechTarget. Retrieved January 10, 2024, from <https://www.techtarget.com/searchstorage/definition/data-migration>
16. IBM.(n.d). What is application migration?. Retrieved December 21, 2023, from <https://www.ibm.com/topics/application-migration>
17. Y Soft(2024). 5 Major consideration for migrating from server to cloud. Retrieved March 14, 2024, from <https://www.safeq.com/blog/considerations-for-migrating-from-server-to-cloud>
18. [18] Gregory, J.(2023) Platform Migration: Strategy, Reasons, Challenges. SaM Solutions. Retrieved March 22, 2024, from <https://www.sam-solutions.com/blog/platform-migration/>
19. [19] IBM.(n.b) What is data migration?. Retrieved March 22, 2024, from <https://www.ibm.com/topics/data-migration?>
20. [20] Smallcombe, M.(2022) . Top Challenges of Data Migration. Integrate.io. Retrieved March 22, 2024, from <https://www.integrate.io/blog/what-is-data-migration/>
21. Segner, M. (2023). Risks Associated with Data Migration: Data Migration Risks and the Checklist You Need to Avoid Them.MonteCarlo. Retrieved April 23, 2024, from <https://www.montecarlodata.com/blog-data-migration-risks-checklist/>
22. Merak(2020).Stages of a Migration. Retrieved February 11, 2024, from <https://www.meraksystems.com/marketing/stages-of-a-migration.html>

23. Bougnague,S.(2023). Migration: 10 Common Data Migration Challenges and How To Overcome Them. Cloudficient. Retrieved March 22, 2024, from <https://www.cloudficient.com/blog/10-common-data-migration-challenges-and-how-to-overcome-them>
24. ManageEngine(2022) Software deployment Version Incompatibility Error. Retrieved March 22, 2024, from https://www.manageengine.com/products/desktop-central/software_installation_version_incompatibility_error.html
25. Graff,E.,Wen,A.,Paetsch,S.,Baxter,J. (2021) .Windows Hardware Compatibility Program. Microsoft. Retrieved March 22, 2024, from <https://learn.microsoft.com/en-us/windows-hardware/design/compatibility/>
26. Linthicum, D. (2024). CLOUD COMPUTING Cloud migration is still a pain: Infoworld. Retrieved March 22, 2024, from <https://www.infoworld.com/article/3712782/cloud-migration-is-still-a-pain.html>
27. ImagineIT(n.b)The Biggest Hurdles A Migration Faces: The 9 Critical Steps To Migrate From An Outdated Legacy IT System. Retrieved March 22 2024, from <https://imit.com/migrating-from-legacy-it-systemsto-a-modern-cloud-based-infrastructure/>
28. Jones,R.(2023).The Importance of Measuring User Adoption of New Technology.TrainingFolks. Retrieved March 22, 2024, from <https://www.trainingfolks.com/blog/the-importance-of-measuring-user-adoption-key-metrics-to-track-for-new-technology>

29. Fong, M.,Peuker,S.(2023). Information management user adoption explained. Open Text. Retrieved March 29, 2024, from <https://www.opentext.com/assets/documents/en-US/pdf/opentext-information-management-user-adoption-explained-en.pdf>
30. [30]TRT(n.b.), What Are The Security Risks Of Migrating To Cloud?: The Security Risks of Migrating To Cloud.Retrieved March 28,2024 from <https://www.trt.com/what-are-the-security-risks-of-migrating-to-cloud/>
31. Maayan, G.(2022). CLOUD SECURITY: Security risks of cloud migration. INFOSEC. Retrieved March 28,2024 from <https://www.infosecinstitute.com/resources/cloud/security-risks-of-cloud-migration/>
32. Ryzhokhin, A.(2023). Understanding Vendor Lock-In and Strategies to Avoid It.ARDAS.Retrieved March 28,2024 from <https://ardas-it.com/understanding-vendor-lock-in-and-strategies-to-avoid-it>
33. Datacore(n.b.).Break Free from Vendor Lock-in: Choose Flexibility and Innovation for Your Data.Retrieved March 28,2024 from <https://www.datacore.com/glossary/vendor-lock-in/>
34. Tekinerdogan, B. & Ozcan, O.(2017).Scalability Requirement:Architectural Perspective for Design and Analysis of Scalable Software as a Service Architectures. ScienceDirect. Retrieved April 28,2024 from <https://www.sciencedirect.com/topics/computer-science/scalability-requirement>

35. Smith, C. & Williams, L. (2000). Performance And Scalability Of Distributed Software Architectures: An SPE Approach. Journal of Parallel and Distributed Computing. 3.
36. Blusonic(2023).Governance And Compliance In Cloud Migration. Retrieved April 28,2024 from <https://www.blusonic.com/governance-and-compliance-in-cloud-migration>
37. Petersen, S. V.(2023). DATA MIGRATION PRACTICES:Avoiding Data Migration Strategy Mistakes: 6 Common Pitfalls and Proven Remedies.HOPP.Retrieved April 20,2024 from <https://hopp.tech/resources/data-migration-blog/strategy-mistakes>
38. Johnson, L. (2024).8 Most Common Pitfalls of Marketing Data Migration. Adversity.Retrieved May 24,2024 from <https://www.adverity.com/blog/8-most-common-pitfalls-of-marketing-data-migration>
39. Giersza, O.(2024).Data Migration: Challenges & Risks During Legacy System Modernization. BRAINHUB .Retrieved May 24,2024 from <https://brainhub.eu/library/data-migration-challenges-risks-legacy-modernization>
40. Cathey, J. M., & Phillips, R. H., Jr. (1994). Backup and recovery. Management Accounting [USA].Retrieved May 24,2024 from <https://link.gale.com/apps/doc/A16408460/AONE?u=anon~827c0065&sid=googleScholar&xid=46e120f9>
41. WIZ(2024).Understanding Cloud Security Risks.WIZ.Retrieved May 24,2024 from <https://www.wiz.io/academy/cloud-security-risks>

42. [42] Pocalyko, P.().Closing The Gap: Resolving Human Error in Cloud Security.Cyber Defense. Retrieved May 24,2024 from <https://www.cyberdefensemagazine.com/closing-the-gap-resolving-human-error-in-cloud-security/>
43. Freeman, M.(2024).Manual migration testing in the modern world: An unnecessary evil. Gable. Retrieved May 24,2024 from <https://www.gable.ai/blog/data-migration-testing>
44. [44]Ridwan, M.A.,Radzi, N. A. M.,Ahmad, W.S.,Abdullah, F., Jamaludin, M. Z., and Zakaria, M. N.(2019). Recent Trends in MPLS Networks: Technologies, Applications and Challenges. Retrieved May 25, 2024. from <https://doi.org/10.1049/iet-com.2018.6129>
45. Warren, I., & Ransom, J.(2002). Renaissance: A method to support software system evolution. Retrieved May 25, 2024. from https://www.researchgate.net/publication/3974556_Renaissance_A_method_to_support_software_system_evolution
46. Gholami, M. F., Daneshgar, F. , Beydoun,G. , Rabhi, F.(2017). Challenges in Migrating Legacy Software Systems to the Cloud - An Empirical Study. Retrieved May 25, 2024. from <https://doi.org/10.1016/j.is.2017.03.008>.
47. Al-Rubaye, H. A. T. (2020).Towards the Automation of Migration and Safety of Third-Party Libraries. Rochester Institute of Technology. (p19-32)

48. Sultan S. ,Ahmad I., & Dimitriou, T.(2019). "Container Security: Issues, Challenges, and the Road Ahead," in IEEE Access. Retrieved May 25, 2024 from doi: 10.1109/ACCESS.2019.2911732.
49. Cao K., Liu,Y., Meng G. and Sun Q. (2020). "An Overview on Edge Computing Research.Retrieved May 25, 2024 from doi: 10.1109/ACCESS.2020.2991734
50. Otmame, A., and Jha,M.(2021) "Without Data Quality, There Is No Data Migration." Big Data and Cognitive Computing. Retrieved May 25, 2024 from <https://doi.org/10.3390/bdcc5020024>.