



Enterprise Solution Architecture Strategy Guide

A Roadmap to Transform, Migrate, and Redefine Your Enterprise Infrastructure
along with Processes, Tools, and Execution Plans



NITESH GARG
ATUL SHARMA



**Enterprise Solution
Architecture
- Strategy Guide**

*A Roadmap to Transform, Migrate, and Redefine
Your Enterprise Infrastructure along with
Processes, Tools, and Execution Plans*

Author

Nitesh Garg

Co-Author

Atul Sharma



www.bpbonline.com

FIRST EDITION 2021

Copyright © BPB Publications, India

ISBN: 978-93-90684-57-1

All Rights Reserved. No part of this publication may be reproduced, distributed or transmitted in any form or by any means or stored in a database or retrieval system, without the prior written permission of the publisher with the exception to the program listings which may be entered, stored and executed in a computer system, but they can not be reproduced by the means of publication, photocopy, recording, or by any electronic and mechanical means.

LIMITS OF LIABILITY AND DISCLAIMER OF WARRANTY

The information contained in this book is true to correct and the best of author's and publisher's knowledge. The author has made every effort to ensure the accuracy of these publications, but publisher cannot be held responsible for any loss or damage arising from any information in this book.

All trademarks referred to in the book are acknowledged as properties of their respective owners but BPB Publications cannot guarantee the accuracy of this information.

To View Complete
BPB Publications Catalogue
Scan the QR Code:



www.bpbonline.com

Dedicated to

*Sheer Will To Learn
Desire to Succeed
Never Ending Dream to be there
Sense of Accomplishment
And
Responsibility to Share with others...*

In Sanskrit, Mother of all...they say it like this....

वंदौ बाल रूप सोई रामू। सब सिधि सुलभ जपत नामू॥

“Vando bal roop soi ramu, Sab sidhi sulabh japat johi namu”

About the Authors

Author: Hey all! This is **Nitesh** I have done B.E. in Computers and possess close to 19 years of professional experience, primarily into Solution Architecture and as an Enterprise Solution Architect.

I have planned, designed and built Datacenters or Cloud Platforms or Application Interfaces across the world.

I have an expertise in IT Consulting, Transitions, Optimization, Transformation, Migrations and Build Projects and Pre-Sales (RFPs/Due Diligence/POCs). I have been in a customer-facing role and have been leading the Solution Defense all over.

I have also acted as a Guest Lecturer to educate college and technology-bound students about the current market trends and ways to start their career.

I am a joint owner of a YouTube channel wherein we publish latest technology videos on request. Here is the link:

https://www.youtube.com/channel/UCT-8sxXz7SBHhB8ZKnqnHw?view_as=subscriber

I am also associated with Professional Learning Platforms/NGOs to share my knowledge with the target audience through video journals and webinars. Here is a link to catch a glimpse of some of my Videos:

<https://www.antwak.com/author/2350-nitesh-garg>

Co-Author: Hello all! This is **Atul** I have done B.Tech. in Mechanical stream and possess close to 17 years of professional experience, primarily into Solution Architecture and as an Enterprise Solution Architect in Cloud and DevOps domains.

I have been working very closely with industry leading cloud provider(s) to plan, design and build DCs and platforms for Business Applications.

I have an expertise in Cloud Designs and Management, DevOps Functions and Scripts, IT Optimization, Transformation and Migration Projects. I have been in a customer-facing role right from the beginning of my career.

I have also acted as a Guest Lecturer to educate college and technology-bound students about the current market trends and ways to start their career.

I am a joint owner of a YouTube channel wherein we publish latest technology videos on request. Here is the link:

https://www.youtube.com/channel/UCT-8sxXz7SBHhB8ZKnqnHw?view_as=subscriber

About the Reviewers

Cherian Mathew has over 22 years of rich industry experience in designing cutting-edge data center solutions that helps increase the revenue, productivity and optimizes the cost. He is passionate about the continuously evolving cloud technologies and has helped many large enterprises and National Government projects to adopt and be successful with private, public and hybrid cloud services. With certifications from various data center technology OEMs, he has handled increasingly responsible roles around Operations, Delivery and Solution Architecture in the IT industry and has acquired exposure to various industry verticals. Currently, he is leading a team of solution architects for a large and well-known cloud software OEM.

Shiva Thakur is a Solution Architect with over 13 plus years of progressive work experience in the field of Enterprise Architecture, IT Operations, Cloud Adoption, Vendor Management and Automation. He is currently handling the role of a Consultant for an IT MNC. He has a proven track record of managing and delivering multiple infrastructure transformation projects.

He is a core member of various technical forums, communities and mentoring programs. He is a strong believer in serving and giving back to the community, thus always helping and empowering people to grow and realize their full potential.

Acknowledgement

We would like to thank a few people before we start.

We are grateful to all those seniors who taught us something when we joined as freshers. We would like to thank all those colleagues who told us what to do and how to behave. Special thanks to all the mentors who thought we can learn something and all the Supervisors who believed in us. Thanks to all those Webinars/Training Courses that instilled something in us! We are immensely obliged to those Management Visionaries who could see some potential in us. We express our gratitude to all those extra-ordinary guys who paved the path for a new career. Thanks to that one guy who gave us that One Chance...

Without you all, we wouldn't have reached where this book was conceptualized to be.

We thankfully acknowledge the services and editors from BPB. Thanks to the entire team of BPB for their invaluable support.

Preface

This book covers all aspects to plan, design, build, or migrate using Solution Architecture as the base in the IT Industry.

This book covers and unveils the very elite and exquisite quotient of IT Industry, The Solution Architecture. It presents and introduces the importance of Solution Architecture. It emphasizes the very basic concepts of Solution Architecture, logics behind, challenges and real time strategies that come very handy while dealing with any piece of IT Industry.

This book projects the inputs that are required to be a Solution Architect. It provides the basic understanding towards Enterprise Solution Architecture.

In addition to, it also depicts detailed and phase wise execution of two of the most critical IT projects i.e., IT Transformation and IT Migration. Elaborated description is aided by pictorial process flow charts, Visio based schematics/designs and lastly MPP based detailed project plan which also includes the technical steps.

This book provides a clear learning path for anyone who aspires to be a Solution Architect and understands its value. The chapters of this book will refresh every thread of your mind.

It takes a practical approach towards Solution Architecture. An effort has been made and a few real-time industry examples have

been embedded with in this book. It covers important aspects like Cloud Concepts, DevOps, and Critical Tools as well.

However, it will specially put an importance on how to become an Architect and how to handle projects like Transformation or Migration as an Architect. This book will guide you towards creating strategy and handling your clients. It will also list down all kinds of techniques, technologies and platforms to show how to migrate from where to where and how to exactly plan, design and build a datacenter or a cloud platform.

This book is divided into 12 chapters. They cover each and every concept of Solution Architecture as listed below:

[Chapter 1](#) begins with an Introduction to Solution Architecture and tells us why it is called an elite and astonishing quotient of the IT Industry. It highlights the process of Solution Architecture, such as, to design an underlying platform Infrastructure or Application itself.

This chapter also highlights a complete reference model of the Architecture world and the basic differences between all of them, while stressing on the real focus, which is to provide the overall solution.

[Chapter 2](#) covers all the Logics such as:

Architecture Principles

Architecture Purpose

Architecture Types

Architecture Levers

Architecture Hats

Architecture Processes

Chapter 3 ushers onto the IT Transformation Project in terms of Solutioning. This chapter focuses on the strategy, vision, purpose and challenges associated with an IT Transformation project.

Chapter 4 depicts and lists out the phases that are associated with an IT Transformation project. It lists down all the benefits along with the different types of lifecycles which a project can go through. It also provides a detailed demo of a complete lifecycle with integrated process flow charts of such a mammoth project.

Chapter 5 focuses on the real world execution of an IT Transformation project wherein all the associated tasks and activities are listed down. This chapter also mentions all the associated tools being used in the executions.

Chapter 6 is a specifically crafted version for our readers which highlights topics such as Engagement Roadmap, Solution Considerations, Benefits and Add-On Offerings.

[Chapter 7](#) introduces IT Migrations as a whole and elaborates about the Migrations Purpose, Migration Principles and Migrations Strategy. It also provides a comprehensive view about what exactly Migrations are and gives a snapshot of the core concepts.

[Chapter 8](#) provides a thorough know-how of various types of Migrations that are prevalent in today's IT world. There are scenarios including Datacenter, Cloud, or a DevOps environment. In addition to, this chapter also straightens the Migration Methodology and moves on to Application Stacks and Application Move Groups.

[Chapter 9](#) concentrates on a complete Migrations Project and lists down all the critical Migration Components. It further illustrates all the lifecycle phases that are associated with an IT Migrations project. All the phases have been elaborated separately in order to make it easier to understand.

[Chapter 10](#) explicitly explains about Migration Topology, Migration Milestones that are associated with each and every phase of a Migrations project. It also lists down the challenges while dealing with the migration of Legacy OS Hardware. Lastly, it lists down all the documents that are to be created during a Migrations Project.

[Chapter 11](#) talks about all the Deliverables that are associated with each lifecycle phase and milestones. It also highlights specific details about Migrations Testing and rollbacks.

[Chapter 12](#) summarizes all the previous chapters.

Downloading the coloured images:

Please follow the link to download the
Coloured Images of the book:

<https://rebrand.ly/528203>

Errata

We take immense pride in our work at BPB Publications and follow best practices to ensure the accuracy of our content to provide with an indulging reading experience to our subscribers. Our readers are our mirrors, and we use their inputs to reflect and improve upon human errors, if any, that may have occurred during the publishing processes involved. To let us maintain the quality and help us reach out to any readers who might be having difficulties due to any unforeseen errors, please write to us at :

errata@bpbonline.com

Your support, suggestions and feedbacks are highly appreciated by the BPB Publications' Family.

Did you know that BPB offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at www.bpbonline.com and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at business@bpbonline.com for more details.

At you can also read a collection of free technical articles, sign up for a range of free newsletters, and receive exclusive discounts and offers on BPB books and eBooks.

BPB IS SEARCHING FOR AUTHORS LIKE YOU

If you're interested in becoming an author for BPB, please visit www.bpbonline.com and apply today. We have worked with thousands of developers and tech professionals, just like you, to help them share their insight with the global tech community. You can make a general application, apply for a specific hot topic that we are recruiting an author for, or submit your own idea.

The code bundle for the book is also hosted on GitHub at In case there's an update to the code, it will be updated on the existing GitHub repository.

We also have other code bundles from our rich catalog of books and videos available at Check them out!

PIRACY

If you come across any illegal copies of our works in any form on the internet, we would be grateful if you would provide us with the location address or website name. Please contact us at business@bpbonline.com with a link to the material.

IF YOU ARE INTERESTED IN BECOMING AN AUTHOR

If there is a topic that you have expertise in, and you are interested in either writing or contributing to a book, please visit

REVIEWS

Please leave a review. Once you have read and used this book, why not leave a review on the site that you purchased it from? Potential readers can then see and use your unbiased opinion to make purchase decisions, we at BPB can understand what you think about our products, and our authors can see your feedback on their book. Thank you!

For more information about BPB, please visit

Table of Contents

1. Solution Architecture – The Astonishing Quotient

Introduction

Structure

Objective

Overview

The astonishing quotient

Different pieces of IT

Segregation in terms of roles and entity

Segregation in terms of technology

The journey and evolution

What is the worth

Conclusion

Glossary

2. Solution Architecture – The Logics

Introduction

Structure

Solution architecture

Solution architecture types

Solution architecture principles

Solution architect role

Solution architecture levers

Introduction of major processes

IT transformations

Purpose of IT transformation

How IT transformation provides benefits?

Phase 1 transformation

Phase 2 transformation

Phase 3 transformation

Phase 4 transformation

IT migrations

Principles of IT migrations

What is an IT migration plan?

Essential components of migration plan

Kinds of migration

Migration tools

Popular migration tools

Migrations process flowchart and lifecycle – a glimpse

Conclusion

3. The Transformations Project – Lets Usher

Introduction

Structure

Objectives

Generic world

Information technology world

How IT changes

A brief look back

Strategy and vision

Purpose

Challenges

Key steps

Conclusion

Points to remember

4. The Transformations Project – Lifecycle and Process Flow Chart

Introduction

Structure

Objectives

Lifecycle of a project

Types of lifecycle

Process flowchart and its importance

Process flowchart for a transformation project

Interpretation – Phase I

Interpretation – Phase II

Interpretation – Phase III

Interpretation – Phase IV

Conclusion

Points to remember

5. The Transformations Project – Tools and Executions

Introduction

Structure

Objectives

Transformation planning and best practices

Finalizing the stakeholders and point of contact

Transforming servers and applications

Dates and timelines

Transformation tools

Discovery and assessment tools

Agentless discovery.

Agent-based discovery.

Generic connector appliance architecture

Microfocus Platespin transformation manager

AWS application discovery.

Azure migrate server assessment tool

Transformation tools

Platespin migrate tool

Velostrata/Google migration services

Cloud endure

AWS Server Migration Service

AWS Database Migration Service

Azure Migrate Server Migration Service and Database Migration Service

Conclusion

6. The Transformations Project – Add-Ons

Introduction

Structure

Objectives

Engagement roadmap

Solution considerations

Benefits

Add-on offerings

Cost-reduction – on the ops

Conclusion

7. The Migrations – Introduction and Lets Usher

Introduction

Structure

Objectives

Generic world

Information technology migrations

Migrations impact

Purpose of migration

Migration principles

Migration drivers

Migration strategy

Strategic parameters

Benefits

Conclusion

Points to remember

8. Migrations Types – Application Stacks – Move Groups

Introduction

Structure

Objectives

Migration methodology

Types of migrations

Application stacks

Application move groups

Significance of application move groups

Conclusion

Points to remember

9. The Migrations Project – Process Flow-Chart

Introduction

Structure

Objectives

Migration components

Migration life cycle

Migration process flow

Migration Phase 1

Migration Phase 2

Migration Phase 3

Migration Phase 4

Migration Phase 5

Conclusion

Points to remember

Key terms

10. The Migrations Project –Topology, Milestone, Legacy OS-Hardware and Documents

Introduction

Structure

Objectives

Migration topology

Planning and discovery phase

Assessment phase

Migration strategy, planning, and design phase

Infra build/test migrations - POCs phase

Actual cutovers/migrations phase

Handover phase

Migrations milestones and journey

Migration overall approach

Migration approach for legacy hardware

Emulation software

Swing-kit migrations

Fork-lift migrations

Migration approach for legacy OS

Operating system upgrade

Re-platform

Reuse

Migration documents

Discovery tool report

Inventory details

Assessment report findings and observations

Workloads health and age report
Databases versions and instances reports
Application – server mappings
Application criticality index
Application dependency charts
Migration tool assessment report
Compatibility analysis
Physical servers movement
Reporting templates
Email templates
Escalation hierarchy
Hardware BOMs
Software BOMs
Move groups formulation strategy report
The finalized version of migration move groups
Detailed build plan
SOPs – DC build
MOPs – DC build
Application cookbooks
Application blueprint
Risk register
Risk mitigation plan
Pre-build checklist
Post-build checklist
Test environment build plan
Test migrations plan and steps
Test migrations results and reports
Migration plan document
Rollback strategy and plan

Detailed steps for move groups migration execution

Handover templates

Conclusion

11. The Migrations Project – Deliverables, Testing, and Rollbacks

Introduction

Structure

Objectives

Migration deliverables

Project plan

Vision diagram or a design

Test environments

Infrastructure tests

Application tests

Migration rollback

Conclusion

12. The Epilogue

Introduction

Objectives

Chapter 1: Solution Architecture - The Astonishing Quotient

Chapter 2: Solution Architecture - The Logics Behind

Chapter 3: The Transformations Project: Lets Usher

Chapter 4: The Transformations Project: Lifecycle and Process Flow Chart

Chapter 5: The Transformations Project: Tools and Executions

Chapter 6: The Transformations Project: Add-Ons

Chapter 7: The Migrations: Introduction and Lets Usher

Chapter 8: Migration Types – Application Stacks – Move Groups

Chapter 9: The Migrations Project: Process Flow-Chart

[Chapter 10: The Migrations Project: Topology, Milestone, Legacy OS-Hardware and Documents](#)

[Chapter 11: The Migrations Project: Deliverables, Testing and Rollback](#)

[Conclusion](#)

[Index](#)

CHAPTER 1

Solution Architecture – The Astonishing Quotient

Introduction

Defining solution architecture: So, what is solution architecture?
Let us get cracking...

Solution architecture is a piece of the Information Technology (IT) environment. It is that piece of IT that deals with defining or proposing a solution to host a new system or to upgrade it. A solution normally would consist of or will include various sections of an IT world as per the requirements.

In general, solution architecture is the process of designing an application (or applications) or to design the underlying infrastructure that hosts the designated application. It is the process that is responsible for the design of one or more applications or services within an organization or for an external client.

Structure

In this chapter, we will cover the following topics:

Overview

The astonishing quotient

Different pieces of IT

The journey and evolution

What is the worth

Objective

The primary objective of this chapter is to highlight the following:

A complete overview of the solution architecture

Understanding as to why solution architecture is called as astonishing

Understand various components of the IT world

Journey of IT and its evolution

And lastly, what is the value of solution architecture and IT

Overview

The main focus of any solution architecture process or a project is always on the technical decisions being made regarding the solution and how they are actually going to make an impact on the outcomes of the project in terms of business and its continuity.

Solution architecture is applied to those projects/programs of the IT world where in there is a requirement to:

Greenfield requirement

Upgrade the existing environment

Sneak in a new system that will fit into the existing environment

Migrate or move existing environment to a new location/environment

Repair or remove existing environment

Do consult to suggest ways to upgrade the environment

Resolve business-impacting issues and many more

Solution architecture is actually the initial step taken when an organization starts building a solution (for its end client) to create a set of enterprise-based solutions or designs, applications, and their associated processes that collaborate and integrate with each other. This is being done so as to address specific needs and requirements of the required solution and being carried out through the software architecture and its associated IT application architecture work which are integrated components of solution architecture.

Solutions and designs that are crafted through processes of solution architecture are jotted down in documents. These solutions highlight a great deal of insights and vision for all current and future solutions, applications, and processes that the organization has. They act as guidelines and as a baseline to ensure that they conform to set standards that make integration and communication easier and make the tracking of problems and inconsistencies between solutions easier as well.

All the solutions and designs documented in the solutions document along with all the instructions are then being carried over for POCs/implementations/builds/migrations, and so on.

Solution architecture, in simple terms, is the process of creating and developing new and innovative solutions, which are based on the parameters defined by the parent organization, associated guidelines along with industry best practices cum instructions. Overall objective is to fit a solution that matches and is feasible enough with the levels of enterprise architecture and is acceptable

to a client in terms of incurring cost and return of investment, information architecture, system portfolios, integration requirements, and many more.

Solution architecture processes can also be termed as concatenation and combination of various internal and external (client based) processes, amalgamation of many roles (to implement and execute the plans), and a lot of documents which includes SIPs, SOPs, design documents, and so on.

Solution architecture is an art in itself. It is an art being exercised and practiced by solution architect(s) to formulate respective solutions. It is an art because a solution architect needs to possess:

Technologies know-how (All)

Deep technical skills

Problem solving

A vibrant and positive attitude

Leadership skills

Decision making

Negotiation and objection handling

Customer conversation skills

Industry awareness and knowledge

Stakeholder management and much more

Solution architecture paves the way for a project as it forms the baseline and acts as a foundation stone. Each project is a result of a well-planned and well-executed solution architecture strategy. This is where solution architecture scores over all other processes and is considered to be a critical component and hence it is termed as an elite entity. Same has been depicted in the following figure:

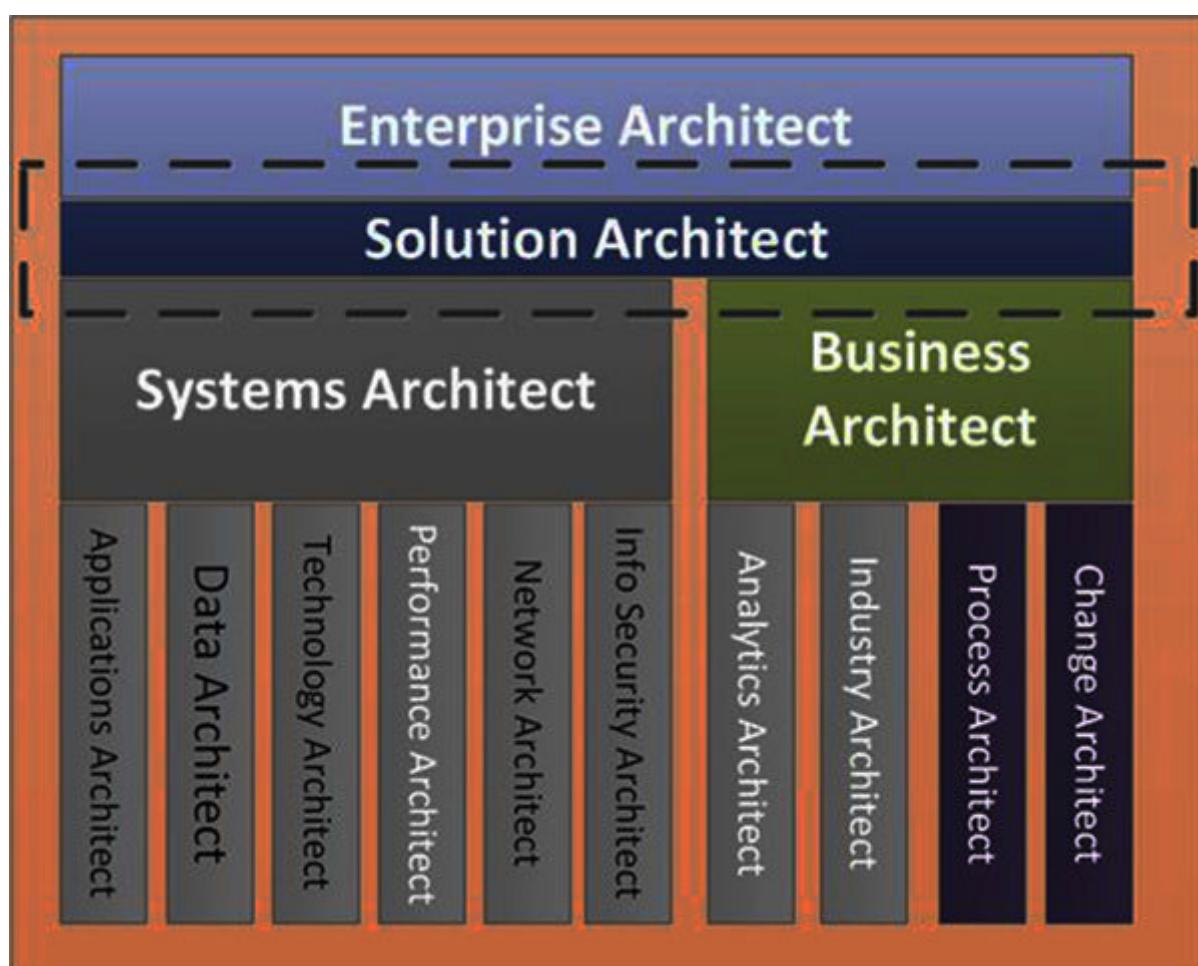


Figure 1.1: Solution architecture

An important point to be noted here is, a solution architect needs specific and significant skills to address various categories of people, technology, and processes. He has to be well versed with all the technologies, be it network, storage, security, database, cloud, DC, wintel, citrix, and even telecom technologies. This is where a solution architect scores over others.

Not only does he need to possess technical skills related to all relevant domains, while preparing a solution, he has to assume, think and assess the number of processes being pursued in his parent organization and importantly, in the client organization. A solution architect must exhibit a balanced mix of technical and business skills and will often work with an **enterprise architect** for strategic direction and decisions.

Additionally, an architect has to design tools to furnish and model the design. He then has to provide architectural inputs and views of the solution based on that model, thus, enabling solution views to different stakeholder groups and enabling impact assessment for future design changes.

The astonishing quotient

So, it is a piece of IT environment. Then why and where is the astonishing quotient? Let us break this myth as well...

The base model of the architecture is the key to all kinds of delivery models and the entire project's success relies on this very base model itself. This very architecture model itself provides much more value than a static design diagram. That is why solution architecture phase and all associated activities become very critical. Very often you will find that all the stakeholders that include your very own top management, Vendors along with client management which may include top business heads/CTOs or even CEOs sometimes, are anxiously and eagerly awaiting design inputs and draft version of the solution.

Thus, it makes the job of a solution architect very critical and very important indeed. A moment of lapse in concentration or a silly mistake here and there may end up costing millions or even the entire project. At the same time, a successful solution will bring in loads of accolades, name, fame, and glory. A careful examination of all available statistics and artifacts, discussions, and workshops to gain more information and an eye on the strategic roadmap will help a SA tremendously to hit bull's eye.

However, as I mentioned, delivering a successful solution in today's fast changing and highly complex and distributed IT

environments requires a lot of thinking in terms of architecture and designs. The SA has to focus on shaping and defining the overall solution and all its associated structures, all integrated components, their dependencies and responsibilities and precisely (and minutely) making core decisions related to impending design and all its associated aspects.

Lastly, a solution architect also needs to address various categories of people. In almost all the projects that a solution architect handles, they will need to do front ending in front of client and their own management to address, elaborate, and suggest a solution for any small or a complex in the simplest of words. problem in laymen's terms. They have to innovate and improvise many a times and find different ways to avoid repeating and convey the same message using a variety of words for different sections of stakeholders and audiences. Thus, it makes it imperative for them to have a deep understanding of all business process, technological depth, people management skills, time management, efficiency, acute precision, and a cohesive vision to create a reliable product.

That is how solution architecture and all the solution architects attain the coveted *Astonishing*

Thus, the ultimate goal with each project comprised of solution architecture processes is to successfully address specific business needs, requirements, or problems through the design and development of applications and information systems. A solution architect has to pay attention to below mentioned key aspects:

Provide a solution that must evolve to meet all future requirements

Provide a baseline that is easy to understand and well-crafted in such a way that it defines all the requirements, principles, guidelines, patterns as well as all the constraints

Provides a solution and design which clearly highlights any and every dependency

Provides insights as to which all process changes will have to be accompanied to get the solution validated and executed

Last but not the least provides a solution that is robust, scalable, feasible, efficient, highly available, and economical

Different pieces of IT

IT is a world of Information in itself. It is such a vast field that it cannot be run via a single entity or a single department. Thus, IT is divided into various pieces in terms of roles and entities. In terms of technology, IT is again segregated into numerous sections which sum up various fields. Let us have a look at them one by one.

Segregation in terms of roles and entity

In terms of roles and entity, infrastructure technology has been segregated mainly into below mentioned various fields:

Management

Sales/Pre-sales

Solution development

PMO

IT operations

Support functions (Finance – HR – Administration - Internal IT Support – Learning)

The following representation illustrates all the pieces of IT:

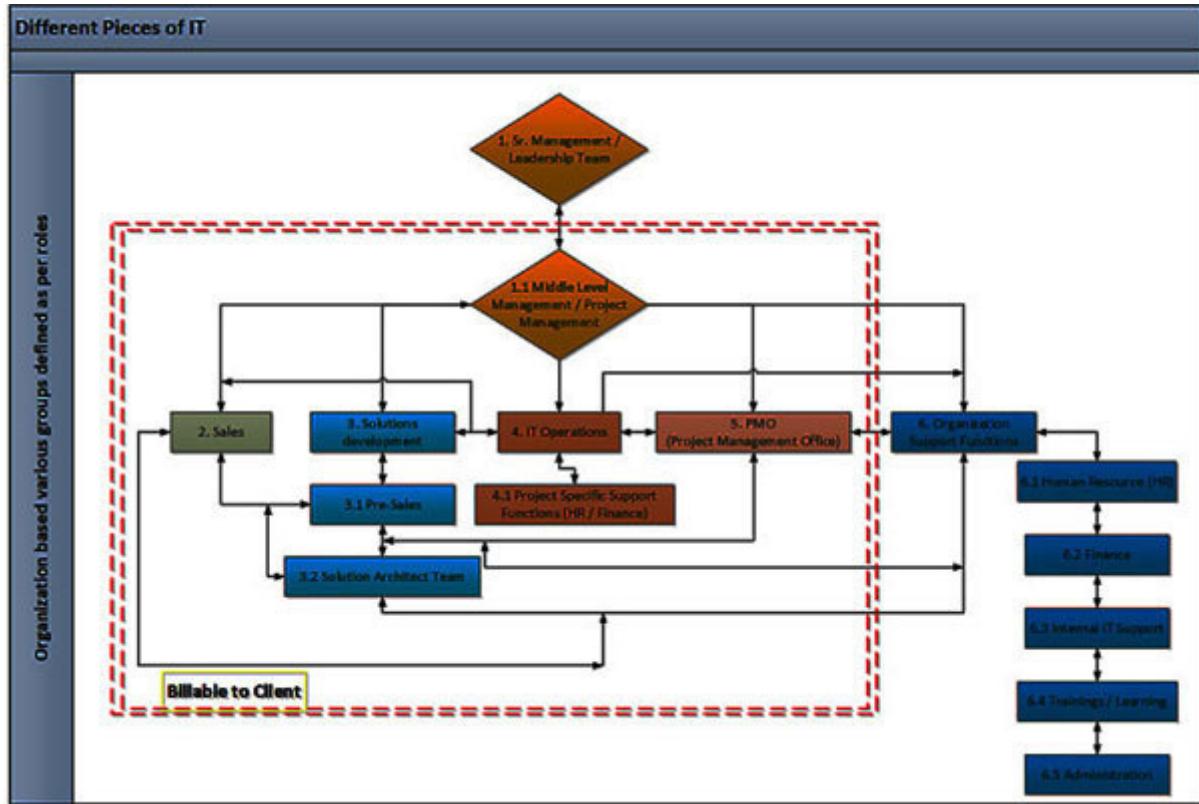


Figure 1.2: Different pieces of IT – roles and entity

As is evident, there are six main sections that primary control all the functions of the entire IT sector.

It starts with management which sits at the top (which is always the case, is not it - pun intended). The management piece is segregated into two various sections. There is **Senior Management** or **Leadership Team** which generally includes the likes of Chairman/Director/CEO/CTO/CFO/BU Heads/MD/VPs, and so on. This piece generally owns or controls the firm and is directly responsible for managing all internal and external affairs end-to-end which includes funding, media, handling client top brass, and so on. This layer is the decision-maker and controls funding and lays down policies and guidelines for the organization.

This layer has a sub-piece that is often called as the mid-level or middle level or the high-level or the project management. This layer normally includes Practice Heads/Business Unit Heads/Project Heads/Delivery Unit Heads/Senior Managers, and so on. This layer of management is responsible for running projects or program of projects, bring more business, align with sales to find new opportunities, handle client(s) for their respective project(s) and manage teams and resources with respect to all the relevant domains or tracks as per their project. One important aspect associated with this layer is to control the funding and maintain compliance to service levels that have been agreed with the client. Any deviations from the SLAs or funding gets escalated to the main management layer.

Sales is one of the most pivotal parts of any organization. They play a very important and critical part in the success of any organization. The success of any organization is mainly dependent on as to how their Sales team perform or how efficient they are. It is the Sales team that help an organization achieve and meet their revenue and fiscal targets.

It is the Sales team which helps to sell a business product or a service to an end user or a client. In an IT world, Sales is generally aligned with selling IT-enabled services or custom-made products to a client that might be in any part of the globe. Sales team are normally aligned with targets. Sales guys act as an intermediate between an organization and a potential client.

Sales guys are usually not technical but they are trained on all the products or services which their parent organization intends to sell to their customers.

Sales guys often need help from the solution development team in terms of technical inputs and this process is called as Pre-Sales. We will be having a separate and a very detailed section on sales and pre-sales in our next guide.

Solution and Software This piece of IT is incredibly critical as architects responsible for solution development (both infra and applications) are the ones who help sales team to successfully close the deal(s) as a RFP.

Then they take over the deal to plan, design, solution, implement, and then handover to operations team. The solution development team consists of pre-sales guys that are none other than solution architects only from both infra and application teams. This team works in junction with sales team while working on a RFP or any deal. They basically work from technical perspective and hold workshops/sessions with clients to capture precise requirements and provide responses to any technical query.

Additionally, the solution development team consists of solution architects which represent all or various domains of IT, for example, network, compute, databases, storage, and so on. All these architects while working on the solutioning part, they act on the requirements as per signed **Scope of Work** and work very closely with enterprise solution architects to design and solution enterprise-based applications. The entire business is based on

these applications which are normally considered business critical and are ranked as per their client age, revenue generation, and usage.

The **Project Management Office** or as per the abbreviation PMO that is now a global industry standard, is responsible or accountable for collecting data that is relevant to the project in flow. It is to be noted that although PMO is associated with IT operations or delivery teams, it also plays a major role in IT development projects like DC build or migrations.

PMO team is basically responsible for tracking the project, its current state, budget status, SLAs monitoring, management reporting, scheduling leadership or client calls, coordination with client, maintaining risk and compliance registers, and tracking economies associated with the project. Some of the reports and data are being prepared for the client as well.

They have to ensure that client-specific standards or organization-specific standards (whatever is applicable as per signed contract) are thoroughly maintained and there are no deviations whatsoever with those.

PMO team is responsible for raising alarms for even a hairline deviation from the defined standards.

IT IT operations or IT Ops as is the abbreviation being used across the industry, is the team that works on the client environment and provides IT support for the services that have

been agreed as part of the contract and project. IT ops team provides support via an incident management tool that covers incidents (defects) of the entire client IT landscape.

Ops team works on daily routine tasks that include technology-based support, reporting, and processes. Every process and activity being executed is tracked by PMO and via specific tools and dashboards are generated and shared with all required stakeholders.

For any client request which is a potential sale or needs support from solutions development team, specific teams are engaged. In such a case, IT ops team works with solutions team and provides all required inputs for the solution to be designed.

IT Ops team would often constitute dedicated Support functions such as Technical Support Team, HR Team, Finance Team, PMO Team, and Project Management to manage all affairs. However, this is applicable only for an Ops Outsourcing organization only, it can be different for a system Integrator or OEM.

Organization Support Organization-based support functions are the generic teams from various departments, such as, HR/Finance/Administration/Internal IT or Learning – Trainings department. It is not dedicated to any project or user but is specific to all the users and projects belonging to the organization. All these functions also provide support via an incident management tool. The primary motive of all these functions is to ensure that all the organization-based users and

projects are able to execute their tasks without any hassles or issues.

Segregation in terms of technology.

IT has always been segregated into two main categories since its introduction, namely: Hardware and Software. Ever since it was introduced, it has essentially segregated into below mentioned components:

Software

Databases

Human resources and procedures

Telecommunications

Hardware:

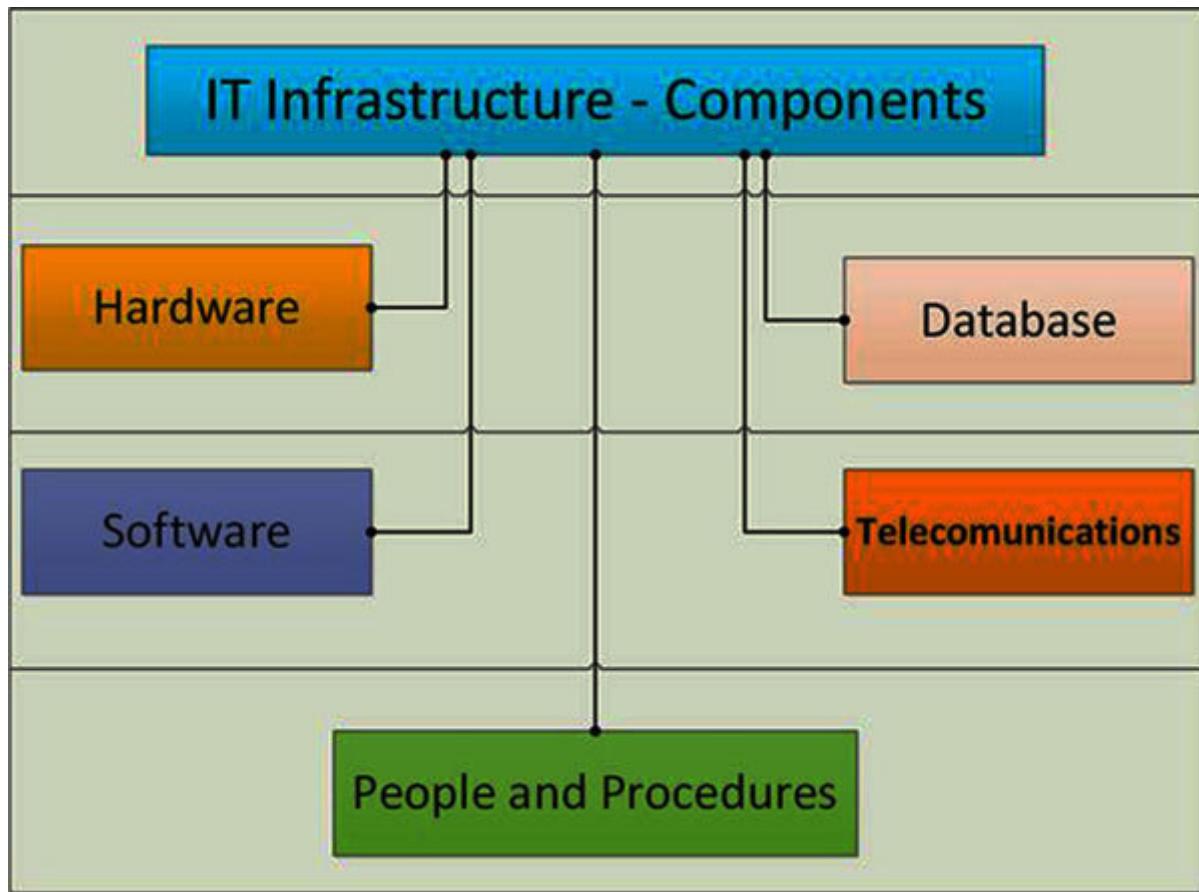


Figure 1.3: Different pieces of IT – Technology

Well, to be honest, this is how IT started out in the very first place. This piece of IT is basically dealing with the physical technology that is being guided by information residing on top of it. Physical hardware can be anything and may belong to any of the domains, for example, network devices (a router, firewall, and switch), compute (servers, chassis, and blades), storage (SAN, disks), telecommunications (fiber cables) and much more.

Hardware makes the base and tip of any IT environment which is being planned to be setup. It is always the starting point of any implementation. Hardware devices once setup correctly and being maintained, normally serve you for years and thus are always

treated with utmost care and are kept in specified, designated, and restricted areas only such as a Datacenter.

Someone needs to tell hardware as to what he has to do and that is actually the role of the software. Software consists of various programs and procedures. It coordinates with underlying hardware and makes it perform as per the requirements. software(s) are generally classified into two different categories: system software and application software. The system software is the base that amalgamates with the hardware. It is the first piece of software that is installed on top of the hardware. System software includes of any flavor such as Ubuntu, RHEL, and Windows, and so on. Application software is the one that is specifically designed to run specialized and some specific tasks only. They are custom-built applications or piece of software which are created to carry out some specific tasks only. An application software may have various sub-modules as well. For example, SAP HANA has as many as seven different modules and they all serve different purposes when being executed.

This component is more popularly known as network. This is where the entire world gets connected. This includes hardware means of using wires, devices, wireless or fibre optics, cables along with software means of the configuration of all these to form a one single network. For example, a piece of network is designed when you tie up a bunch of computers and brings them on a single network which is normally called as **Local Area Network** If you tie up computers across the cities, then it becomes **Wide Area Network** And let us not forget the network of networks!! The internet is the biggest of them all as every piece

of hardware and software operating in this world connects with it. No internet means no functionality. And it only takes a router for you to get there.

Database(s) are one of the most critical components of a business IT setup. Database is used to store date in a specific format, and it's not just store but organize as well. Database systems are designed for applications and they work injunction with them. Any application-specific transaction is being executed through databases only and all sorts of logs are stored along with the data. Any query through the data or any search requests is carried out by database only. If a database stops working, then the whole application module stops working. Thus, databases are setup with huge resources and are kept with the highest maintenance, monitored throughout through highly specialized tools. Some of the examples would be Microsoft SQL and Oracle.

People and This is the final component of the IT infrastructure and quite frankly, the most important one. Human factor, possibly has to be the most important component of all. It is always the people or the humans as you basically need people to run the show. And how would they run it, they will some sort of guidelines or instructions to perform their job which actually are procedures.

The journey and evolution

IT has really come of age today. It is really as good as anything and still evolving. However, it wasn't the same always. It has been close to some 100 years since IT was conceptualized first. It was more of a support to human kind in terms of technology early on and later on the evolvement was rapid. Specially, the evolvement has been pretty rapid if you compare in the last few decades and years.

What started off as a mere hardware has turned into a dynamic box hosting virtual environments. With the introduction of the cloud and now DevOps, IT has taken another leap into the future. Same infrastructure that was built by over five system administrators over a period of 5–10 days, is now built using scripts and codes in a matter of minutes. In fact, growth has been such you do not need to host an IT room or a datacenter or invest anything in hardware, you can simply just walk onto a cloud and run entire operations from there:

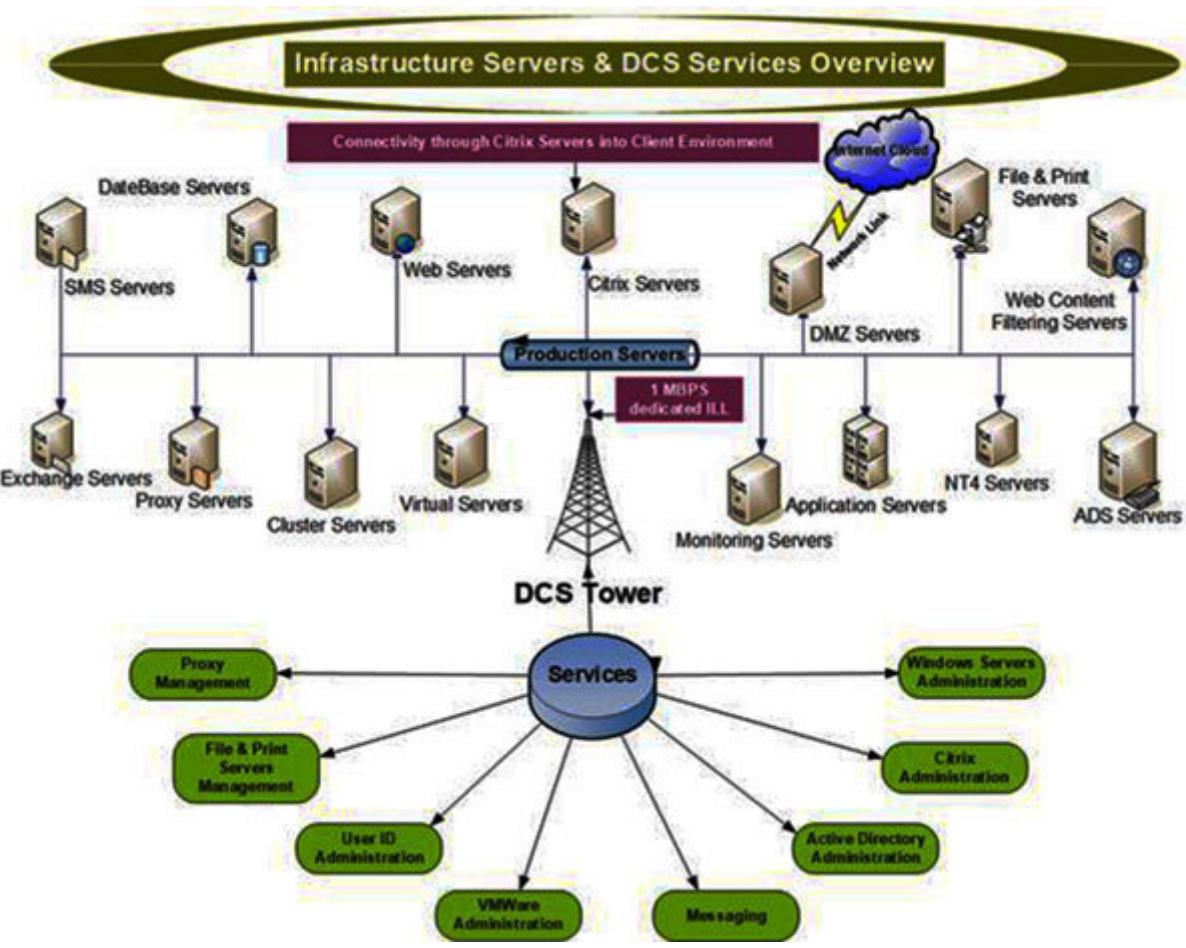


Figure 1.4: Traditional IT services

The preceding figure represents a design of a DC being planned to be hosted. This used to be the trend not so very long ago. The preceding figure shows as to how **Hardware** was the means of setup in IT world till early 2000. Physical hardware along with system software and then application software, this is how all setups used to be done.

However, this evolution changed with the introduction of virtualization technologies like VMware and Microsoft Hyper-V. With virtualization, entire IT landscape started changing. Now, hardware was designed in such a way that it would accommodate

more and more resources in terms of memory, storage, and processor so as to host a fleet of virtual servers on top of them. Thereafter, cloud burst onto the scene and IT changed its face completely. With cloud, you do not have to worry about buying hardware, its maintenance, upgrades, and so on Cloud provides you ready infrastructure and whole lot of tools and technologies and you just have to choose as to what you want and you are good to go. And it is cheaper too as it is a Pay as you go model. Thereafter, IT experienced another evolution that saw DevOps becoming the flavor. It is that part which was earlier elusive and was pretty much restricted to developers and script writers. However, with DevOps, all tasks and activities can be put into a code and can be executed with in a few seconds. With all this, solution architecture has become a very major force as IT environments are global in nature now and more complex than ever:

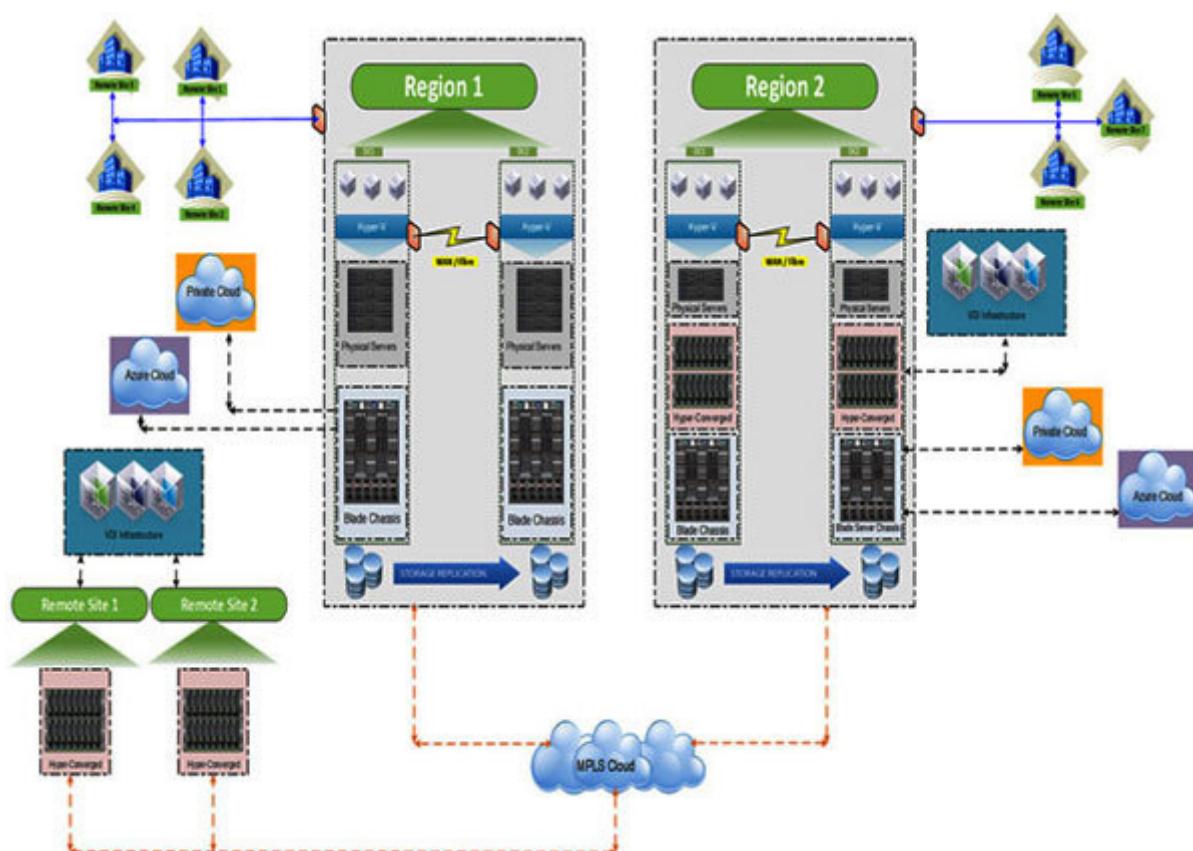


Figure 1.5: Traditional IT services

There is a requirement now to make applications available to a user in the USA and in China at the same time. Thus, solution architecture along with enterprise architecture drives the business in all of IT Landscape.

Lastly, on a closing note, it will not be an overstatement if I mention that IT is still evolving and it is the adaption of IT infrastructure which makes it so very reliant. It is an ever-changing environment now and the landscape is going for a makeover once again.

What is the worth

It is a proven point that application of solution architecture for any development project leads to much better project results. Various research programs and studies have analyzed, suggested, and confirmed that the approach of applying solution architecture to a project helps to achieve milestones in time via reduced throughput time. More importantly, it is quite an impact on the budget as well which is always the key factor. Additionally, solution architecture inputs help to a reduction of variance in time and budget as the project progresses. From a long-run perspective, initial application of solution architecture and all its components and processes to a project have proven to be very beneficial and has been earmarked as a quality improvement process.

Thus, no wonders that almost all the IT firms across the world holds a team of solution architecture in their kitty.

As for business, clients across the world prefer to communicate with a solution architect. That is why, workshops are arranged during RFP process and solution architects acting as pre-sales guys participate with their client counterparts to resolve all technical queries and provide clarification with the strategic roadmap.

If we talk from an architect's point of view, they see projects which are executed under solution architecture as a whole world in itself, each and every one of them. For an architect, every single of these projects presents tremendous opportunity for learning. It presents an exposure that is absolutely rare. Each such project shapes an architect in terms of technical knowledge, process knowledge and orientation, compliance awareness and controls structure, provides absolute mental toughness to shine under great pressure and it shapes the overall persona of an individual. Working as a solution architect is the ultimate thing in IT infrastructure field and attaches immense value to an architect's resume.

An architect attains completeness when he works as a solution architect and starts designing and solutioning.

Thus, any solution architecture project taken over by a solution architect working as part of the solution development team is like a dream come true.

Conclusion

This brings us to the end of our first chapter of this book. In this chapter, we understood as to how IT evolved and stood up. With the advent of new technologies, IT has now grown leaps and bounds. We also had a complete overview of the solution architecture and its overview and why it has a special place in IT world. Lastly, we understood various IT components and their associations with each other at a high level.

In the next chapter, we will focus on the topics such as solution architecture principles and its basic purpose. We will then have a detailed look at all the levers of solution architecture followed by introduction of major IT processes which are prevalent in the industry today.

Stay safe...see you in the next chapter....

Glossary

Glossary

Glossary Glossary

Glossary Glossary Glossary

Glossary Glossary Glossary

CHAPTER 2

Solution Architecture – The Logics

Introduction

This chapter will begin our journey into the world of solution architecture (SA). We will be getting acquainted with various types of SA and principles. We will be also be going through SA levers and will have a look at the major process of SA in terms of transformations and migrations.

Structure

We will cover the following topics:

SA principles

Solution architect roles

SA levers

SA processes

IT transformations – A glimpse

IT migrations – a glimpse

We will be also going through SA levers and will have a look at the major process of SA in terms of transformations and migrations.

Solution architecture

In general, principles help in decision-making for an organization and gives a way forward to how the organization's goals are achieved. Within an organization, sub principles need to be set such as principles related to architecture that uses organizational vision and organization principles. Architecture principles cover the governance of processes.

Based on the organization such as type or organization domain, organization, and architectural principles are developed.

Solution architecture types

Following are three types of architectures that are present in an organization:

Enterprise Enterprise architecture covers the entire organization. Enterprise architects use specific set of principles based on enterprise architectural planning that results in better consistency. It is oriented towards business architecture, application data, and technology architecture.

Solution A SA focuses on details while enterprise architects have an eagle view which looks at the higher perspective, i.e., from the organization goals perspective. Solution architects wear multiple hats in order to cater the needs of users, leadership and management, IT, security, and so on. In a nutshell solution, architects are technology leaders who provide technology vision to an enterprise.

Infrastructure Infrastructure architecture aims to cover the baseline or the underlying infrastructure that supports the overall business setup or applications. An infrastructure architect will support and design changes or upgrades in the base Infrastructure technology as per the evolving market trends and availability.

SA is further segregated into three different types:

Conceptual

Logical

Physical

Conceptual solution

It targets solution vision and its context. It is the most abstract layer and a very high position where one can view almost everything from business to technical. It represents entities and their relationship with each other. Conceptual architectural diagrams are generally structural models that show concepts and their relationships but not exactly the flow diagrams on how they work.

Logical solution

It is a middle layer between the conceptual and physical layers of SA. This describes how a solution works considering logical information and also the functionalities of each entity. It focuses on high-level and low-level details.

Physical solution

It is detailed oriented layer and is more implementation/build related. It is related to specific products, technical and data representations which is for setting up a technology solution.

Example: data center design, its integrations with other providers/vendors or cloud-based SA or their reference architectures.

Solution architecture principles

Enterprise Architecture (EA) principles are guidelines and rules created by enterprise architects in achieving the enterprise objective and goals. Principles are developed based on the respective domains such as business, data, application, and technology.

Business

Business principles should provide maximum return on investment to an organization or enterprise.

Business principles should be based on business continuity requirements. Operations should not hamper because of system interruptions.

Entire process should comply with laws and regulations.

Business principles should be developed to protect organizational intellectual property.

IT organization is responsible for setting up infrastructure and IT processes to meet **Service Level Agreements** user requirements, and delivery timings.

Reusable applications and assets should be developed in order to remove duplication and inculcate standardization in the organization.

Architecture should be based on a design of services that mirror real-world business activities comprising the enterprise (or inter-enterprise) business processes.

Data

Data is an asset, that is, real data that can be measured is valuable. It supports in decision making specifically if data is accurate and if it is timely it can help in taking timely decisions.

Data is shared. Appropriate data access is required for users for carrying out their roles and responsibilities. Accordingly, data is shared within and across organizations. Shared data is virtual single resource across the enterprise. This helps the enterprise in improved decisions since it is an ultimate single source of more accurate and timely managed data.

Data is accessible for users to perform their functions. However, access to data does not guarantee a user to modify and edit the data. Access in form of appropriate user rights and permissions decide the level of access on data. However, any information should not be misinterpreted by the user while analysing the data if it is not part of his job function.

Above three principles are interrelated and complement each other.

Trustee accountability is a must for data quality, that is, a trustee is responsible for the validity and accuracy of data. Data trustee makes sure that data content goes on a single virtual resource shared across and within the enterprise. Integrity of the data should be controlled by placing appropriate quality control measures.

Data is defined in a consistent manner across the enterprise. Available data definitions should be understandable by all the users. If a new data definition(s) is required, the definition effort needs to be coordinated and framed with the data descriptions

Unauthorized use and disclosure of data are protected. Pre-decisional, sensitive, source selection-sensitive, and proprietary information should be protected. Systems, data, and technologies must be protected from unauthorized access and manipulation. Headquarters information must be safeguarded against inadvertent or unauthorized alteration, sabotage, disaster, or disclosure.

Application

Applications are independent of the choice of specific technology and should be able to work on different technologies. This independence allows applications to be developed, designed, and used in the most effective way. Dependence on a particular technology provides higher chances for applications to be obsolete if the technology becomes obsolete.

Applications are user friendly. Ease of use reduces the training efforts for the users to understand the usability of the application. The usability of the application should be easily understandable.

Technology

Changes to applications and technologies are driven by business needs and their requirements. Business needs arise in form of improving the business process that can be met or fulfilled by making the changes to the underlying technology or making changes to the application itself such as changing the underlying platform for the application, say moving to the containerized application from the traditional application which sits on a specific workload.

Responsive change management is needed to implement changes in an effective and trackable manner. Based on the changes, the architecture artifacts are updated.

To reduce the non-trivial cost of maintaining expertise in and connectivity between multiple processing environments, technological diversity is a must. There is non-trivial cost of infrastructure required to support alternative technologies for processing environments. Limiting the number of supported components will simplify maintainability and reduce costs.

Interoperability for data, applications, and technology is promoted across the organization(s) via defined set of standards. This helps

in the usage and supportability of the product by multiple vendors in the market. This also provides extensibility of the applications on different technologies from different vendors.

Solution architect role

SA is a description of a discrete and focused business operation or activity and how IS/IT supports that operation.

A SA will help to deliver organization values and their definite purpose. Solution architects come into play during solution design and implementation. Prior to solution design and build, solution architects understand solution's business context. They then define the solution vision and its requirements. They participate in **Request for Proposal (RFPs)** or in proof of concepts during the design phase and develop a solution roadmap in line with business requirements. During the solution build phase, SA communicates and collaborates with stakeholders and guides the implementation team.

A solution architect will have to play the different role depending upon various sections of the business such as:

Business

Technical

Functional

Data

Build and implementation

Operational management

Solution architects work with enterprise architects for strategic direction. They are integrators between different entities within an IT system that meets the business requirements for the organization. They are responsible for selecting and finalising the appropriate and the best technology needed for the solution. Solution architects need to consider the below things while designing the solution:

Business and technology objectives: Technology to be used for the solution, that is, finalization of services and products provided by finalized vendor say from cloud perspective or data center perspective.

Which framework, vendor, cloud provider to be used for the solution In-line with business and client requirements

Open sourced versus closed sourced software

Resources required for implementing the solution

People and skills required to implement the solution

Scalability of the solution

Capacity planning of the solution

High availability and disaster recovery considerations

Security policies of the organization

Authentication, authorization, encryption of data in REST and transit and security aspect

Legal aspects considering the user base and the solution hosted

Solution should be cost effective and not over expensive

Maintenance and operational management for the solution once its setup and is in live state

Deciding time lines and work with the project manager so that the project is executed within timelines.

Solution architecture levers

SAs levers are used and applied for multiple tasks or domains such as below:

Infrastructure modernization which can cover migrations onto cloud platforms

Network refresh and upgrades

Embedding multiple layers of security and compliance

Implementing hybrid environments consisting of cloud, on-Prem DCs.

Data management that can cover database, storage, SAP, healthcare, and so on

Application development from monolithic to micro-services architecture using features such as DevOps, Serverless functions, and so on

Continuous improvements and continuous developments (CI/CD) via Cloud Development Platforms

Cloud Development and Integrations

Analytics such as AI, ML, Big Data, and many more:

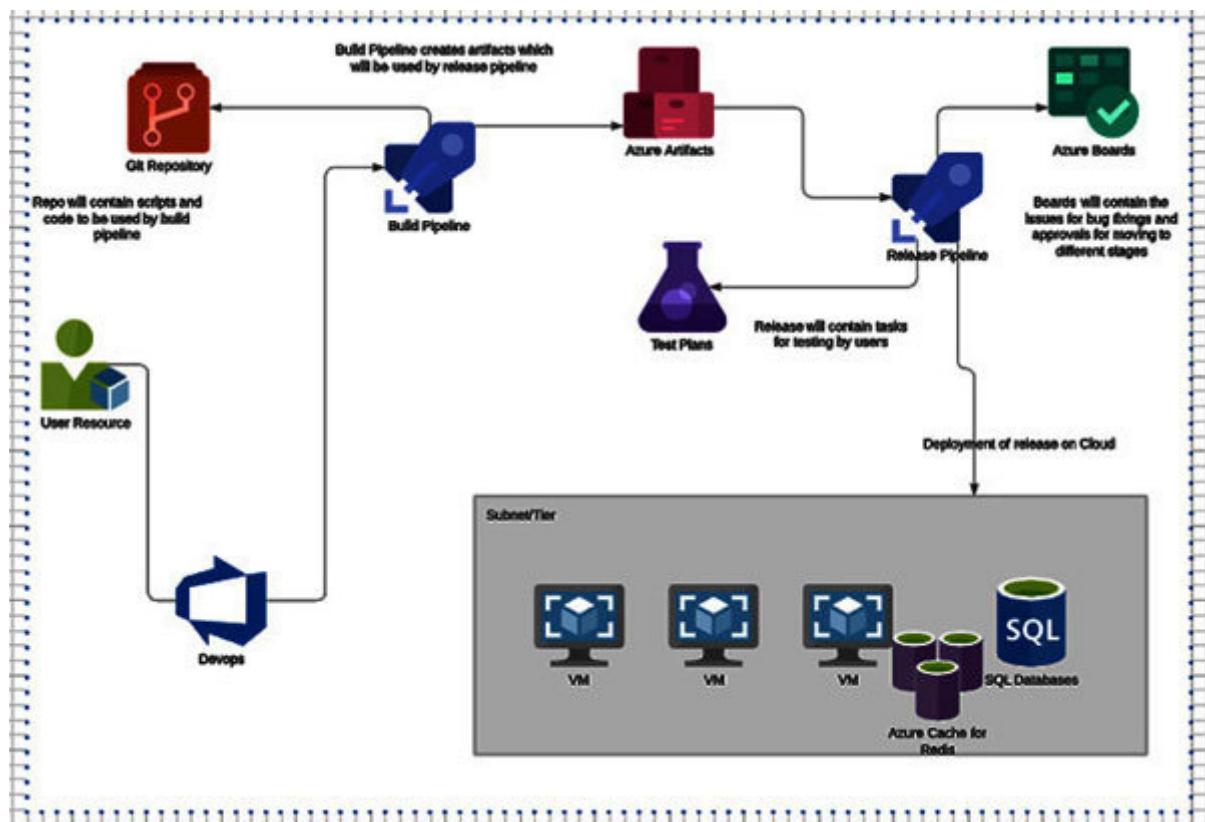


Figure 2.1: CI/CD – Cloud Tier

The above diagram has the following flow for the solution that displays the continuous integration and continuous deployment setup in the environment.

Here, the solution architect has chosen Azure as the solution for the deployment and its CI/CD.

Users are provided appropriate rights from security perspective to use Azure DevOps that is a DevOps service from cloud provider.

GIT repository is chosen to store code and scripts.

Build pipelines shall use code from GIT repository and create artefacts which shall be stored in the artefacts in this case SA chose Azure Artifacts.

Artefacts uses the feeds which is a form of code used by Release pipelines to deploy the infrastructure on cloud or on prem environment that will be consumed by the users.

Release pipelines contain multiple tasks in sequential order which shall be used to deploy infrastructure. Examples of such tasks

Pick up the environment say development environment as variables

Task to deploy the virtual machines using the artefacts feed generated via build pipeline based on the **Infrastructure as a Code** in form of JSON templates or scripts or Terraform code or any other means.

Task to deploy database on which the application servers are dependent.

Post deployment, there is a test plan task that will do the testing and store the results in Azure boards.

In case of errors, the errors need to be approved before the release is applied to the next stage, that is, say UAT and then production:

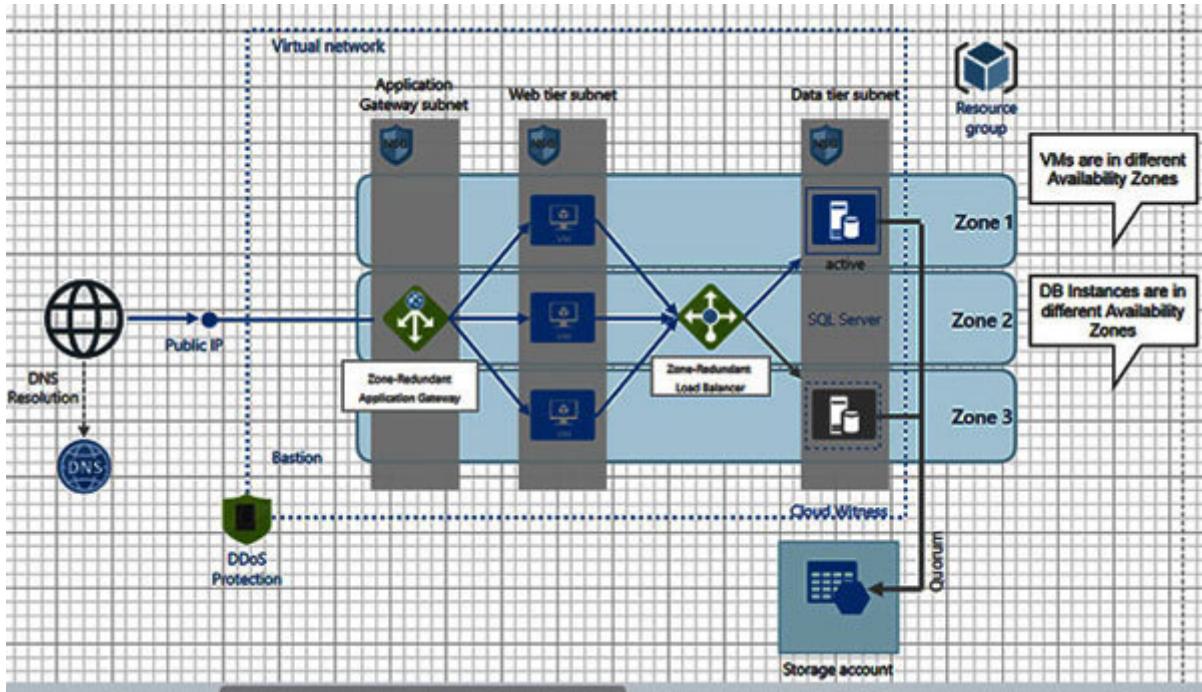


Figure 2.2: Web and DB tier

Solution architect covered high availability with resilience at every layer:

Web layer

Database layer

Application gateway/load balancer layer

Each zone in the cloud has a different physical location.

If one SQL server fails, other becomes functional in another zone.
Zone redundant load balancer will detect that and accordingly
route the traffic to other SQL DB internally for web servers.

Introduction of major processes

Introduction of major processes:

IT transformations

IT migrations

IT transformations

Transformation in itself is a very broad term. Basically, any idea that has been planned and executed, embedded with new features, keeping the original motive and scope alive and enhancing the mechanism of delivering that scope and with added value is called as a transformation of any product or service.

In other words, a transformation process is any activity or group of activities that takes one or more inputs, transforms and adds value to them, and provides outputs for customers or clients. It is the set of changes in the physical characteristics of materials or customers.

Thus, the process in which a product is renewed or is being refurbished as per current or future requirements is called as transformation process:

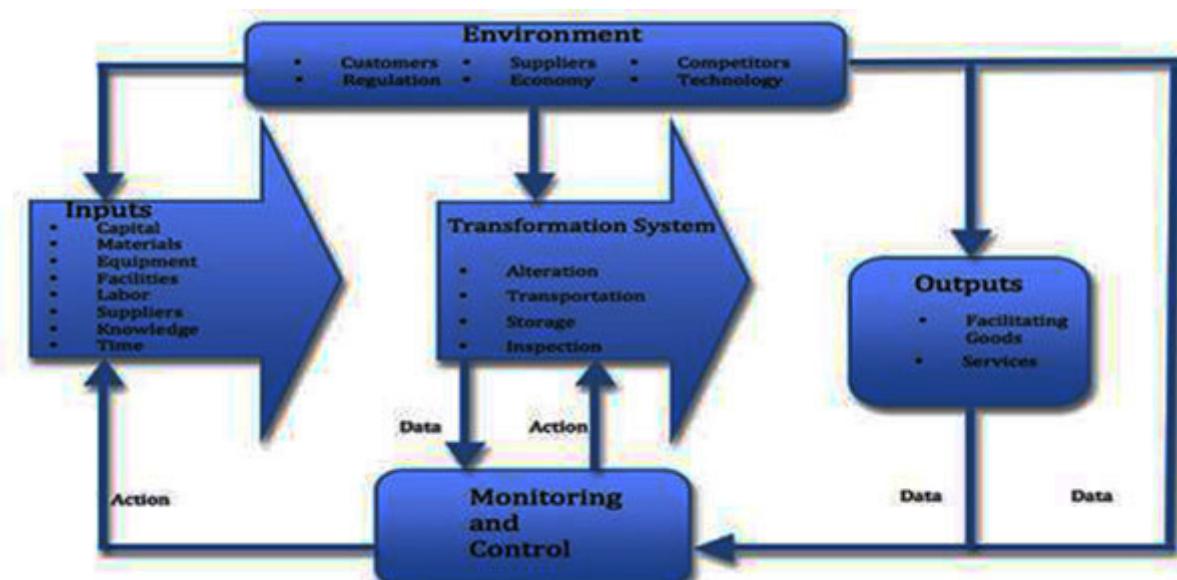


Figure 2.3: IT transformations sample

IT transformation is normally referred to as *rip and replace* (informally).

IT transformation is a process which is used to create/renew/rebuild/refurbish IT capability for a client/customer.

It is a complete reassessment and overhaul of an organization's information technology (IT) systems in order to improve the efficiency and delivery in a digital economy.

It can involve changes to -- and modernization of -- network architecture, hardware, software, IT service management, and how data is stored and accessed.

IT transformation basically forms the foundation of an organization's larger digital transformation strategy and is led by business leaders (such as the CIO / CTO).

Purpose of IT transformation

IT transformation aims to change the IT department from being a reactive, inflexible organization to being a proactive, flexible part of the business.

Transformation enables an organization to adapt and respond quickly to changing digital business requirements and make more informed decisions.

The ultimate goal of these efforts is to "reimagine IT development, delivery, and operating models, and to enhance IT's ability to collaborate effectively within the enterprise and beyond its traditional boundaries."

Also, it is mandatory that organizations must take steps to keep pace with an increasingly digital and competitive market by not only optimizing existing systems, but also creating and acquiring new applications and services that generate deeper insights into their business, industry, and customers.

How IT transformation provides benefits?

Successful IT transformation builds a solid foundational infrastructure on which to deliver automated services, cloud computing, and new operating models.

It also automates and accelerates the deployment of IT services and reduces risk during deployments.

IT transformation clears the path to deliver IT as a service that is more cost-effective, agile, and helps foster innovation.

IT transformation also ensures better business-IT alignment.

By optimizing traditional IT cost models, organizations free up IT budget from operational expenses and can designate more funds for digital transformation.

Let us have a look at the transformation project flowchart:

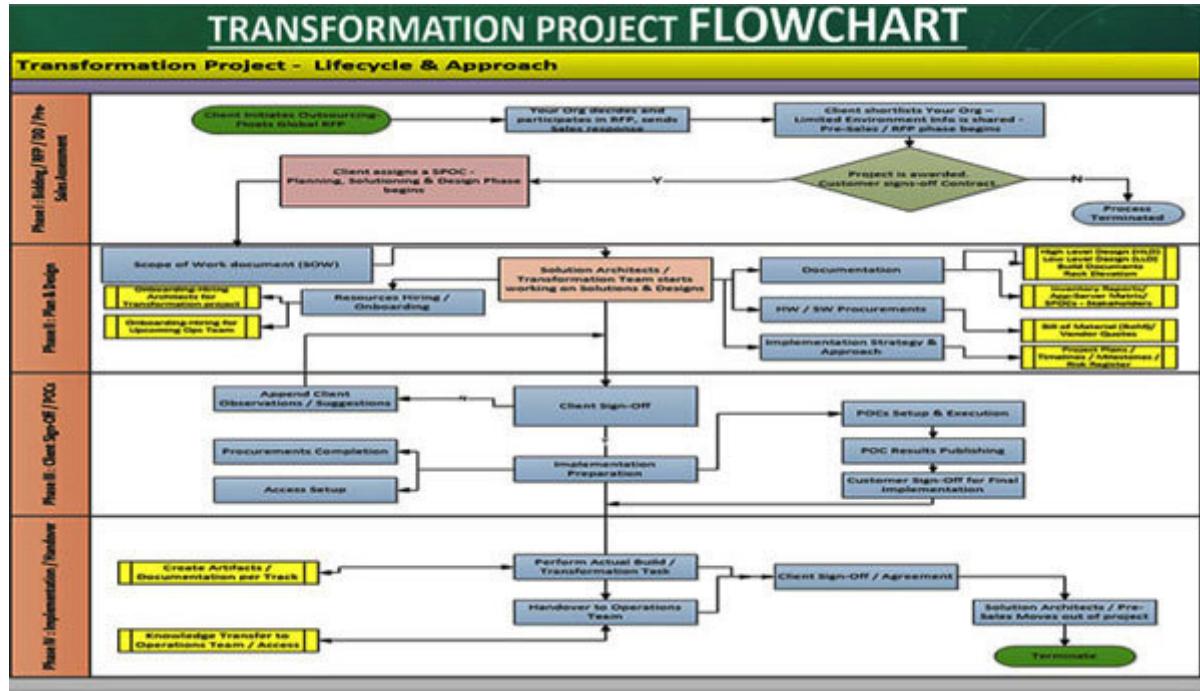


Figure 2.4: Transformation flow

Phase 1 transformation

Transformation Phase 1 starts with any client (in any part of the world) initiating an RFP. This means that client is willing or ready to outsource his business to a vendor.

Based on the RFP, all potential service providers respond to the RFP with a sales pitch.

Thereafter, the client shortlists a few vendors (based on experience, exposure, vendor org size and potential and off course cost) and shares a limited amount of information.

This information is used by sales/pre-sales team to prepare potential solution for client pain areas.

Once the deadline is over to submit the solutions, then client awards contract to a suitable vendor and assigns a SPOC who interacts with vendor org and helps them get access, required information, mitigate any risks or loopholes, and ensure a gap-free delivery within stipulated contract and assigned cost.

Phase 2 transformation

Transformation Phase 2 is for Plan and Design. This phase starts post contract signing and the transformation team starts working as per roles and responsibilities mentioned in the **Scope of Work** document.

A lot of activities are done in parallel in this phase, such as, resourcing, documentation, Procurements which includes bill of material and vendor quotes, defining transformation approach and strategy.

Documentation includes high-level designs, low-level designs, rack elevations sheets, and build documents. It also includes inventory along with app-server mapping.

Client SPOCs and stakeholder's identification is a very critical activity in this phase.

Phase 3 transformation

Transformation Phase 3 is for Client Sign-Offs on all the design documents followed by POCs setup and implementation.

This phase also includes procurements (both HW and SW) completion which is again a critical activity for transformation project.

Vendor access on the client environment is also processed in this phase.

POCs setup, results publishing, and customer acknowledgment leads to the next phase that is final implementation.

Phase 4 transformation

Transformation Phase 4 is the final phase in a transformation project.

In this phase, final implementation takes and every piece of IT being built or rebuilt is being documented as an artefact for customer sign-off.

Alongside, the transformation team involves operations team and starts imparting knowledge transfer to them.

Once implementations are over, then piece by piece (or track by track), customer validation and sign-off are sought and received.

Post completion of all builds, customer sign-off, and handover to ops team, transformation team moves out of project and process is completed.

IT migrations

Migrations is a dynamic term that is an act of moving from one place to another. IT migrations refer to a process that moves or converts data processing or information systems to a different technology.

It is a process of transferring business IT processes to a new hardware infrastructure or a different software platform.

In all kinds of migrations, the movement is done towards a system perceived to be better than the existing system and in the long run provide better value. Migrations generally involve downtime when the old system is replaced with a new one. Sometimes two systems run in parallel with one system in testing phase. When the cutover is finalized the old systems are switched off.

Principles of IT migrations

Every IT migration works on the below principles in order to strengthen, modernize, and revenue generation from the serving IT platform:

Replatform

Rehosting

Repurchasing

Refactoring

Retiring

Retaining

Decommissions

Consolidation

Centralizations

Upgradations

Future roadmap

ROI / TCO

Cost savings

What is an IT migration plan?

A migration plan is an essential initial step to carry out a successful migration. It lists out all the critical components associated with migration with respect to teams, resources, stakeholders, vendors, timelines, checklists, processes, technical SOPs, and rollback strategy if needed.

It includes end-to-end documentation such as plan, design, testing, execution and implementation, reporting, handovers, owners, and ongoing support.

It lists out each and every detail about migrating the workloads and associated applications.

It lists out phases in form of waves when the migration will take place, which environments will be targeted and when.

Essential components of migration plan

Every migration plan consists of below essential components. The project will be jeopardized if any of the below components are missed:

Identify the data format, location, and sensitivity: Identify what data you are migrating, what format it is currently in, where it is residing, what format it shall be in postmigration

Planning the size and scope of the project and migration plan.

Backup all the data: Prior to the migration, entire data and workloads to be migrated should be backed up. In case of any errors and issues in cutover, the roll back should be done via restore or alternate strategies.

Assess migration tools to be used and assess staff understanding about the migration tool. Staff should be provided training to understand and run the migration tools in order to carry out the migration.

Execution of the migration plan: ensure right permissions, controls, data protection and monitoring are in place during execution.

Testing of the final system: once migration is complete, ensure that there are no connectivity problems with source and target systems. This ensures that entire data is correct and intact and secure in the target location.

Follow-up and maintenance of the data migration plan: Recheck and re-audit all the data post-migration.

Kinds of migration

IT migrations vary from platform to platform. Which type of migrations is to be performed depends upon the target. Target can vary from datacentre to cloud.

Migrations strategies can be of the following type:

AS-IS Also called as lift and shift from one environment to other.

Here, the infrastructure is built in the target systems with new software and OS versions and the data is imported on top of it from the source location. Entire solution is not moved AS-IS.

There are many systems that an organization is unaware if they are being used in the environment but they are running in source location. Such systems are candidates for decommission or retired.

New and fresh Here, the entire platform is built and configured from scratch in the target location. Appropriate rights are given to user to use the new systems. This happens when the older systems in source location are outdated and cannot be upgraded directly.

Kinds of migration are as follows:

Data center to data center

Physical to physical (fork lift and swing kit migrations)

Physical to virtual server migrations

Virtual to virtual server migrations

Data center to cloud

Physical to virtual server migrations

Virtual to virtual server migrations

Cloud to cloud

Virtual to virtual server migrations:

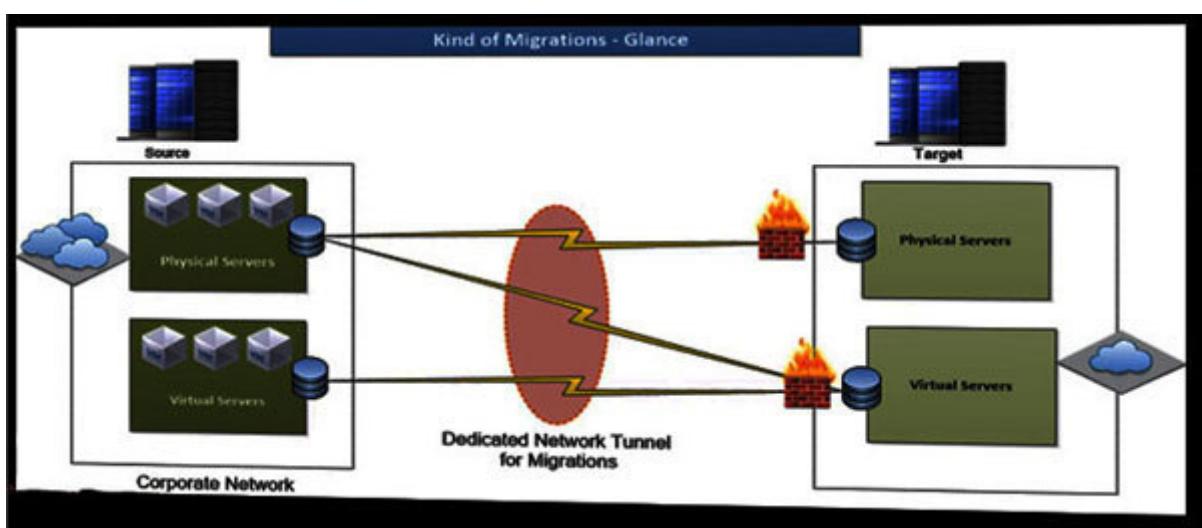


Figure 2.5: Migration types sample

Migration tools

Migration tools are used to migrate the data in form of servers and storage to the destination or target systems. These tools convert the entities from the source location and moves them over the network to target systems in appropriate format.

Migration tools perform the following functions:

Captures the state of the server (neither paused or shutdown state).

Pre-copies memory pages and references from source to destination host/location.

Any changes to memory pages during pre-copy process are also logged.

These changes are logged as delta which is less in size.

Post pre-copy completion.

Migration tool then copy delta changes and transfer ownership to target host.

Target host then bring the server back to operation.

Migration team performs postmigration checks. Performs any DNS changes, website certificates changes, reconfiguring the new load balancers or gateways.

Post that testing team carries out testing of the application.

Cutover is finished and the server at source location is shutdown.

Popular migration tools

All the following tools are license-based tools that are used for P2P, P2V, V2V based migrations.

Cloud endure (for Azure, AWS, and GCP)

Plate spin

Carbonite

Azure migrate and azure site recovery

Free conversion tools: vSphere converter for virtual to virtual migrations

Migrations process flowchart and lifecycle – a glimpse

IT process flowchart and lifecycle are explained on the basis of the following:

Phase-I is an assessment phase that involves assessing the environment and capturing inventory. In this stage, it is being identified that all which applications and their stack servers are potential candidates for migrations. Those who do not fit the bill are candidates for decommission. They are identified candidates, will be processed as a part of migration plan. They are also assessed if they are compliant enough with respect to anti-virus, patches, server health parameters, backups, and monitoring.

Phase-II of a migration process focuses on creating a migration plan. This stage also helps in identifying if a server is a candidate for physical to physical, physical to virtual, or virtual-to-virtual migration. Here migration tool is also identified.

Phase-III of the migration process stresses on the setting of test lab environment ding testing, publishing results and getting required approvals to proceed for actual migrations. This phase implements the plan, checklists mentioned in the migration plan that includes performing test migrations. Once testing is complete, results are published to the stakeholders:

MIGRATIONS PROCESS FLOWCHART & LIFECYCLE

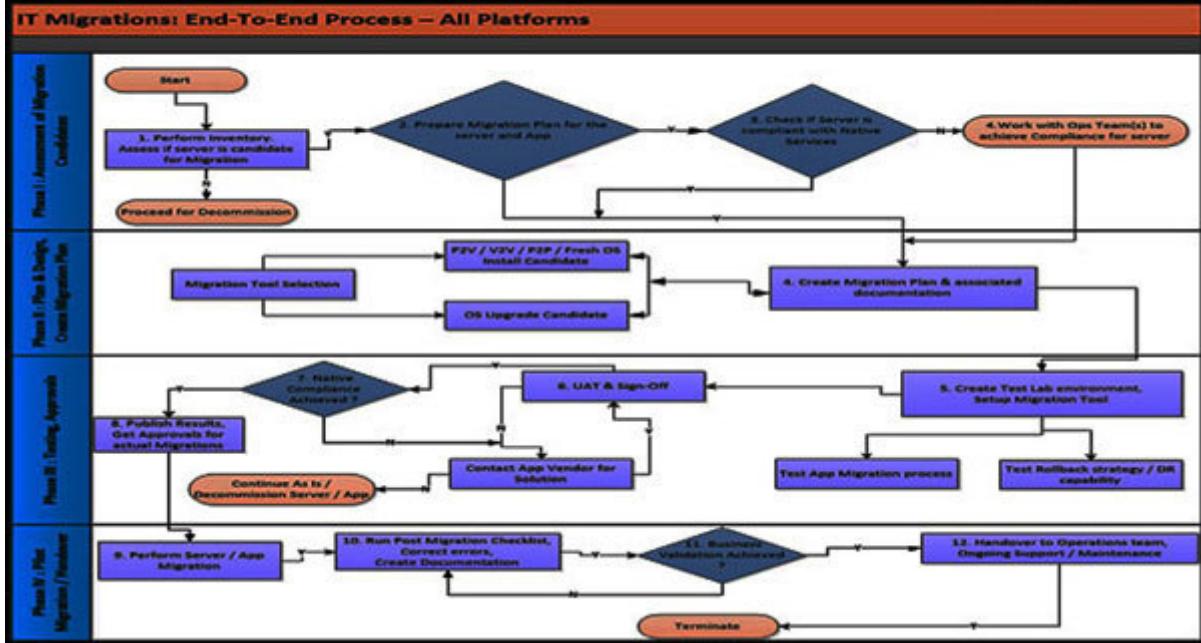


Figure 2.6: Process flowchart and lifecycle

Phase IV of the migration focuses on actual migrations.

Migrations are carried out using pre and postmigration checklist and using identified migration tools. All the migration steps are well articulated and documented in the migration plan.

Postmigrations validation is done by business and migrated server is handed over the operations team for daily operational activities.

In case of any errors, postmigration checklist is run again until business validation is achieved.

We will be covering IT migrations, process flow and entire lifecycle in much more details in the following chapters.

Conclusion

This marks the conclusion of this chapter on SA. We have had a brief look at various aspects of SA in terms of its principles, its types and its levers. We have also understood the role played by a solution architect. And we have had a nice glimpse of major IT processes related to SA which are IT transformation and IT migrations.

In the next chapter, we will start with transformations as a whole. We will go through transformations history, its strategy, vision, challenges, and much more. See you all in the next chapters and happy reading.

CHAPTER 3

The Transformations Project – Lets Usher

Introduction

An introduction to Information Technology transformations project.

What is the meaning of a Transformation in the first place...what does it symbolizes in a generic world and what does it ensemble inside an IT world.

A brief look at how IT changes and what are its strategy, vision, and purpose.

Along with, what are the various challenges and Key steps associated with an IT transformations project.

Structure

In this chapter, we will cover the following topics:

Generic transformations

Transformations history

Transformations strategy and vision

Transformation purpose

Transformation challenges

Transformation key steps

Summary

Objectives

In this chapter, we will:

Learn about transformations in general first

Learn about IT transformations

Have a brief look back

Understand what is strategy and vision

Get to the purpose see the challenges faced followed by key steps

Generic world

So, what are transformations? Transformations mean change as they bring a change to about everything they are applied to. And it is not just any change, they change a human, a life, a product, a process, a material, and an area altogether. Thus, it is simple to imply that transformations are applicable to every facet of life on this planet:



Figure 3.1: Generic transformation

Transformations in itself is a very broad term indeed and have an even deeper meaning. They are basically any idea or suggestion, which has been thoroughly brainstormed and then carefully planned, executed with an aim to enrich the old product with a set of new features to compete in the current market scenario or to adapt as per the new and given situations or circumstances.

Today's outlook has become very shaky indeed, business environments are highly unpredictable now and disruptions are

more frequent than anything else. In fact, disruptions have become the new norm these days and the world is still trying to learn to adopt a new normal. There is no such thing as *guaranteed* and nothing can be taken for granted. Such scenarios present a very complex question of existence in the market in front of all business and non-business entities across the world. This is where transformation comes in.

Transformations when applied to a business, they bring about a change in terms of:

Processes and how they are applied

The way an organization function

All the human resources

Approach and strategic changes

Off-course how the funds are spent

Overall product design, its nature, and outlook:

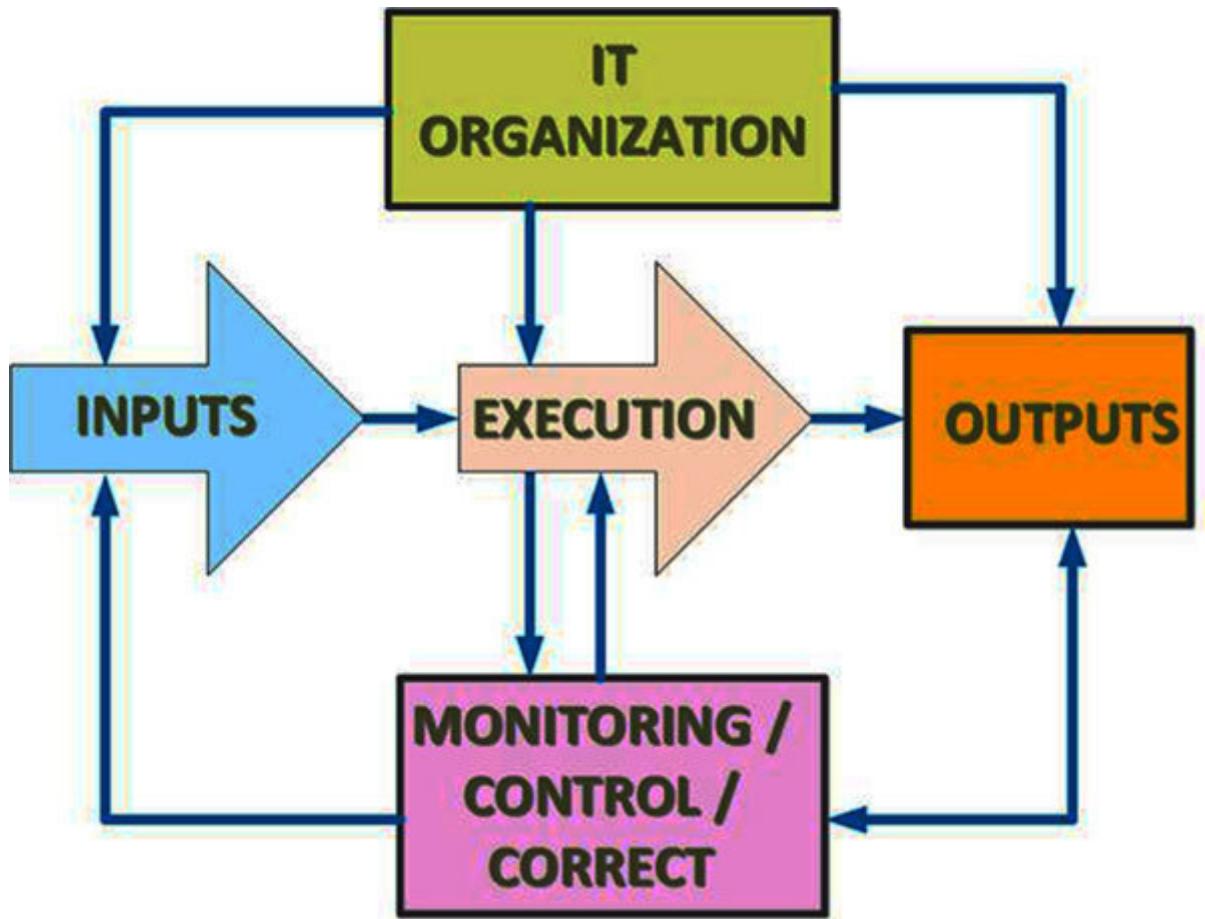


Figure 3.2: Business transformation sample

Information technology world

An information technology transformation is not much different from a generic transformation. Basic concept of an IT-based transformation is being derived from a real-life transformation only. There are many components in an IT Transformation which are pursued exactly in the same manner such as a goal and a motive, expectations, outcome as they are being pursued in a real-life or generic transformation. However, IT transformations differ in a way as to how it is being done. There are a lot of definitions, logics, and concepts attached with transformations. We will go through a few definitions through the course of this very chapter to understand what are transformations, their varied meanings, how they are interpreted in different contexts, and how they are relevant and connect through IT world.

Transformation is one of the most common and at the same time, one of the most overused word today's IT world. This particular word is used today in every aspect of IT environment. It is not just limited to upgrading hardware assets/software/applications but applies to every business that uses Information Technology as a platform and operates out of it. Believe it or not, but transformation is even applied to Annual Reports and even to Press releases, however, implying that its true meaning might have been muted amid the cacophony of buzzwords.

Getting back to IT world, IT transformations are popularly or generally called as *Rip and Replace* across the industry. Please note that this is not an official Tag and it is rather a nickname that has been associated with IT transformations due to the overall nature. It does not mean that transformation(s) aims to rip the entire environment or replace it entirely. It is a schematic process or a set of processes that gradually transforms an environment, basically giving a new birth to an already existing environment and loading it with a whole new set of features to cope with upcoming requirements.

An important point to mention here is that, transformation is not always done to reassess or to rebuild an environment. Today's IT market and world is as dynamic as it can be. With the advent of technology almost every other day, it has become a necessity for firms to catch up to stay relevant. A famous example of this would be a very famous mobile company, Nokia which was ahead of all its competitors in the late 1990s and early 2000. However, with the advancement and introduction of smartphone technology, Nokia became outdated and was eventually driven out of the race. Thus, it has indeed become a cat and mouse game and at times organizations go through transformation processes to add:

Value to their portfolio

Value to their stakeholders/clients

Value in their operations

At times, information technology transformations can be termed as a tool or a set of tools that are being put in use by an organization to achieve the desired milestones. It can be termed as a mechanism or a process that is being applied to:

Create

Renew

Rebuild

Refurbish

For me, information technology is also a capability service in itself. This service is being normally applied on an existing environment for an IT client or a customer or for an internal upgrade for a new application or software. It is a synonym of a study which is being carried out to assess the current situation of IT environment (which is in question) which belongs to a client, followed by a thorough evaluation if the environment can be upgraded and overhauled where in deciding about latest Technology or products to be used and then finally executing and implementing these plans to achieve the final state of the desired IT environment. Not to mention that the service itself makes use of all the latest technology trends and processes as per the Industry standards.

How IT changes

IT transformations normally involve fundamental level changes to an organization as it aims for modernization. It involves architecture level changes to all the aspects of an IT environment which includes:

IT hardware assets

Applications and software(s) running in the environment

All the support services and their respective management

Data accessibility and its storage

Overall product designing

Network upgrades:

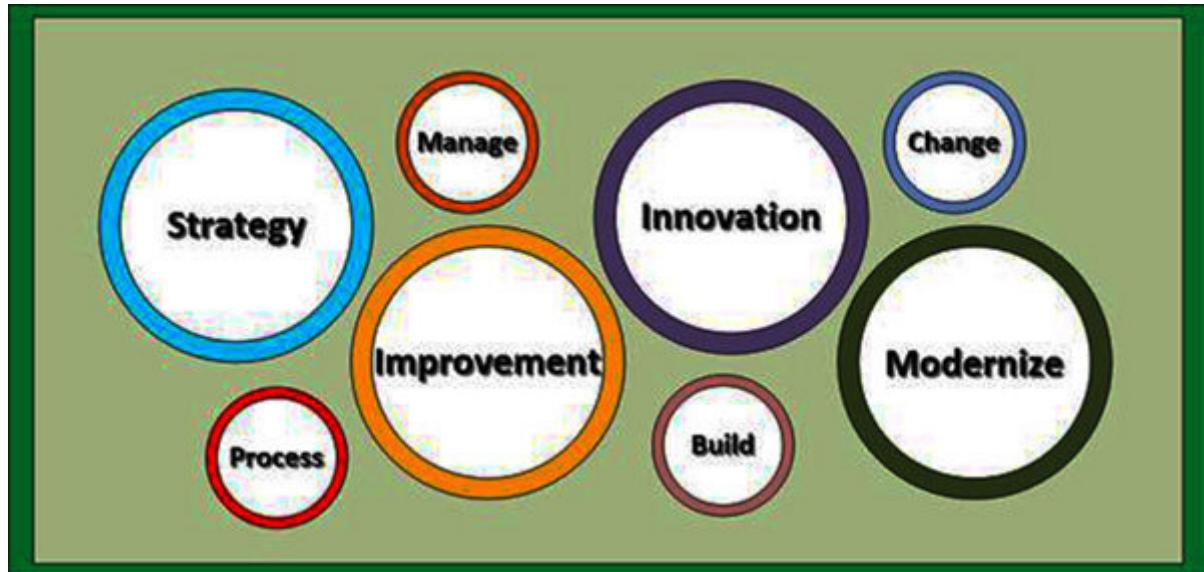


Figure 3.3: Transformation fundamentals

All these components generally form the backbone for any IT organization and thus making changes to any of these involves a lot of effort and hence a lot of planning. As is evident, this also involves a lot of costs as procurements also run as part of transformation projects. Generally, any new procurements of hardware assets/software(s)/applications will cost a fortune as they are used at Enterprise Levels to support massive workforce of users across the organization. Thus, such transformation projects are executed under the supervision of Senior Leadership teams or Business leaders such as (such as the CIO/CTO).

A brief look back

Transformation was not always welcomed in this way or was not even looked with a sign of respect. In fact, transformations in the past were always associated with some huge and lingering problems. Infrastructure technology and transformation were only and only associated with IT-based companies only. Transformations were always associated with a **No Other Option** choice with a popular notion that the firm does not have any other option but to go for a radical change and transform itself. It was a sign or an indication that a firm has got complex, Broad, and deep-rooted systematic problems. It was a general belief that a company undergoing transformation:

Is not able to cope up with current market scenarios

Unable to meet the challenges

Unable to live up to customer expectations

Not able to meet its financial targets

Unable to generate any new value

Unable to explore new markets

Unable to expand client base

Is struggling with poor or old Infrastructure

Old or outdated skills possession

Overall, transformation for any organization was related to overall disruption in the organization and its functions:

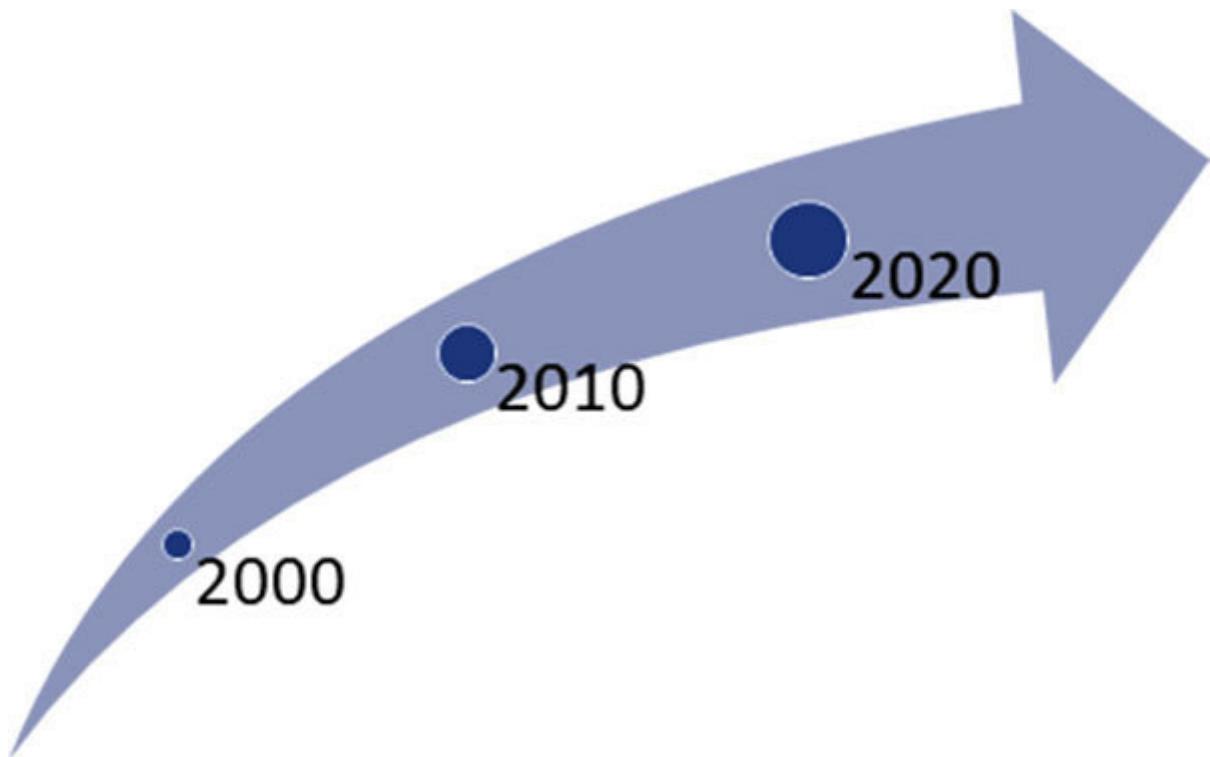


Figure 3.4: Transformation acceptance

However, at the end of the last century, IT started coming of age. This was the time when information technology as a whole started making Leaps and Frogs. It was not just limited to IT companies, but it started exploring other fields such as Pharmaceuticals,

Media, Entertainment, Research, Space, and even defense worlds. New and emerging technologies like Internet of Things, Artificial Intelligence, Machine Learning, Robotics, Digital evolution, 3D Printing, and so on have really helped the cause. Now, information technology is at forefront of everything. And that is how transformation found its turnaround.

As per a study, major organizations across the globe have gone through a transformation in the last decade to upgrade, to accommodate, to adapt, and to be on top of things in terms of information technology. This has helped the transforming organization to reach a position of strength. Companies are enabling transformation programs to completely overhaul their existing landscape and add a great value. Every single component of information technology, operating in an organization's premises is being assessed and transformed like a well-oiled machine that gears into production at just the onset of a button. And this is what showcased to all potential clients as well which is being displayed as a Strength, adapted modernization, and readiness.

Now, transformation is being viewed and pursued as an imperative factor in today's business context and even for the brands who have global presence and are at the top of their game. Every firm on this planet is now trying to operate out of a self-imposed transformed state and the rest are following the suite and trying to catch up. Transformation is now being seen as an answer to such volatile market scenarios with the forces sometimes beyond control. It is being seen as a tool with the emergence of such technologies to meet highly complex business

requirements. It is being adopted as a synonym to research and reinvent.

Transformation is being accepted as a newer technology that can lend a helping hand to cure a patient lying in an Operation Theatre. It is being accepted as a technology that helps and improves daily life of every global customer. Lastly, it is being trusted as a medium to thrive.

In a nutshell, the transformation has been embraced with both arms wide open. It has become not just a tool to reinvent your clients but also to allow resurgence of your own business as well.

It has become a need that all the companies should focus on transformation as an *always-on* process. A process of evolution to maintain a definite competitive advantage as market conditions change.

Strategy and vision

A transformational strategy and its approach are as important in the field of IT as a transformation itself. In normal terms, a strategy is nothing but a plan of action. However, it is not just any plan of action. It is a logical plan of action that is being construed and aimed at defining as to how the company would function or operate for a specific project/client/region/technology or a user base.

A strategic plan of action also focuses or strategizes on as to what, how, when, and which technology will be adopted by the organization.

Strategy is normally defined for a long-term perspective unless there are some emergency measures are to be for a specific client of business unit:



Figure 3.5: Core components

Strategy along with a vision is very important in an IT transformation project. Strategy forms the basis of any and every IT-based transformation project. A strategic approach is embedded with a clear vision and associated roadmap to execute the project and move it forward in a specific direction. Every IT transformation project needs an effective strategy to meet the defined vision:

It starts with developing a clear, coherent vision of the future state of IT as to what exactly is required.

Transformation project comes with a business case for transforming desired pieces of IT.

Transformation vision must be developed with the input and sponsorship of senior management.

Business and IT leaders also must define the foundational needs, scope, and scale of the IT transformation project, including having a clear project timeline:

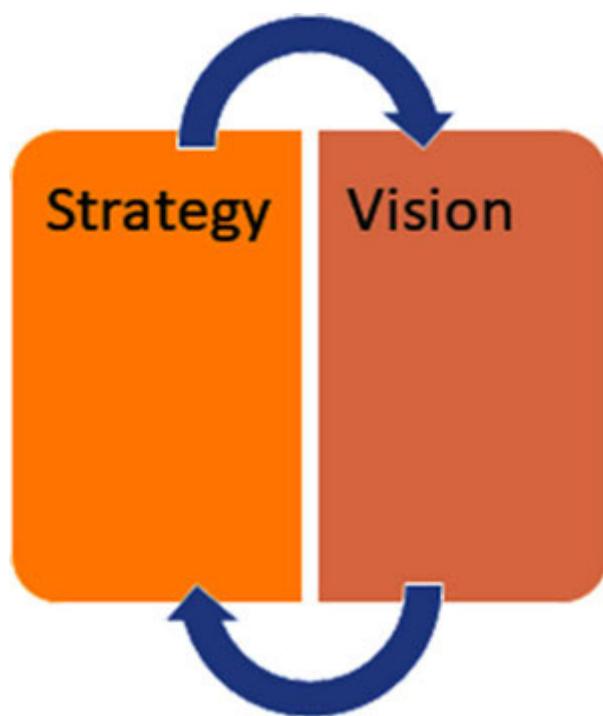


Figure 3.6: Transformation strategy and vision

Strategy and vision are defined at the very start of the project itself. In fact, strategic guidelines are laid out much before the project reaches its operating stage. Strategies start coming into formation as soon as a client issues a RFP and the project is up for grabs. Specific strategies are formed depending upon the state of the project which can be at RFP stage or can be at the execution stage. Strategic guidelines are created for all the different stages of the project and as per client requirements. These strategy guidelines are formed with a vision of the future.

Strategic guidelines are important guidelines, however, they do not contain any design and thus are not the blueprint. They are generally treated as the operating guidelines along with all the designs that are part of blueprints.

For the users, blueprints are the guide, design, pattern, or the guidelines of something which is to be developed or build. A blueprint is now taken over by designs and patterns in today's world. A strategy plan or guidelines is called as a Blueprint as it contains sections such as plan, section, action, and elevation. It formulates a strategy so important which sets up the context of the entire project or the organization and makes it moves forward.

A strategy once formulated is being implemented across the length and breadth of the organization through Human Resource mechanism. Each and every component of the project or the organization is made to follow the strategy and the vision with which the strategy has been embedded. Generally, a strategy plan or guidelines document will normally consist of the following:

Precise roles and responsibilities

Rules

Guidelines

Funding strategy

Milestones

Stakeholders

Future mode of operations

Escalation matrix

Strategy guidelines or strategy plan as some may call it is normally prepared by keeping the ground realities in mind. A strategic plan will provide greater flexibility to all the project entities who are executing the project and generally would project them as decision-makers as well in some cases. Such plans are created keeping the below thing in mind:

Funds/incurring costs/expenditures tracking

Market shares and gains

Achieve profitability

Turn any loss into profits

Assess any potential risks and mitigate them

Higher sales/Upsale

Strategic plans have wider reach and they normally stay under constant watch from the critics, public, and all the competitors as well. This is basically as a strategy normally depicts as to how an organization would function in the future and what will be its mode of operandi. A well-defined strategy passes on a message to everyone with-in the organization and to the entire external world about the passion and engagement of the top leadership which generally generates a positive outlook about a company or its associated projects. This normally depicts as to how *Things are Handled* in terms of overall operations.

A positive strategy with a promising and clearly defined vision may end up changing fortunes of a company or a project across the latitude and longitude across this planet. It should be well noted that along with users, general public, clients, and stakeholders, all the competitors duly take notice of the corporate strategies of everyone else in the same domain. Almost all of them, make an effort to precisely understand and study the strategic roadmap. It is a very well-known practice to minutely understand, predict and learn the practices adopted by others.

At the end of the day, it is the strategy embedded with a well-graphed vision of an organization that lays out the path to progress or profitability. It is being seen as a medium as to how a corporate entity would eke out profits and will manage and increase its sales.

Purpose

Every transformation project is always associated with an effective and well-defined purpose. Reason being, an effective transformation project means pursuing, meeting, and achieving the fundamental purpose of an organization. It puts a lot of weight in the core values of the firm and it helps to efficiently fulfils and matches the current goals and needs of the targets that are being pursued. A purpose well laid and put behind a transformation project helps to gain the momentum and confidence of all related stakeholders.

An IT transformation aims and seeks to change an organization's entire IT Landscape including all existing environments and upcoming ones from being a reactive, inflexible organization to being a proactive, flexible part of the business.

A transformation project or approach enables an organization to adapt and respond. The purpose behind a transformative initiative is to:

Redesign and realign IT landscape

Adapt as per new opportunities

Unilateral response to extremes

Change as per new trends

High-enable IT delivery

Reimage all existing operating models

Enhanced and effective collaborations

Lasting impressions within the enterprise and beyond its traditional boundaries

As we learned earlier, transformation is to completely overhaul the line of business and the basic purpose behind is to introduce new products. At times, the real purpose is to target a specific set of customers, or a region, or a specific product that is aimed at a specific generation group. Also, it is mandatory that organizations must take steps to keep pace with an increasingly digital and competitive market by not only optimizing existing systems, but also creating and acquiring new applications and services that generate deeper insights into their business, industry, and customers.

With every transformation, the ultimate purpose is to achieve:

Increase efficiency

March towards new growth horizons

Usher into new opportunities

It is always the purpose that stays with the project until it is being achieved. A purpose can be defined with other synonyms such as aim or an objective. It must be clear that a project when initiated, is marked by a definite purpose and then objectives are set to meet that purpose.

In simple words, the purpose is and always has been the heart of a transformation project along with its strategy, its vision and its goals:



Figure 3.7: Transformation purpose

The objectives of a transformation project are always defined based on the following parameters:

What is the core function of the organization?

What needs to be adapted?

How to optimize?

How to stay on with its core values?

How to achieve?

How to realign the project or the organization?

How to adapt a changed external environment?

Objectives are in fact set and defined for each and every stage of the project. For every such stage, milestones are set for a project throughout its journey. At each milestone, project is being thoroughly reviewed. These reviews are done on the basis of the objectives which have been set for that particular milestone.

Thus, percentage of objectives that have been met actually state the current state of the project. It gives an idea or clue as to where the project is heading. That is when the leadership team heading the project reviews and takes decisions based on the current situation.

These objectives highlight if the project is under stress, deviating from its expected path, going overboard with funds, going as usual, stuck due to internal or external reasons, needs expert resources or a revamp.

It is very important to note that a project's ratings are being done based on the objectives it has met during its lifecycle. Everyone associated with a transformation project work towards achieving or meeting the objectives of the project. It the sheer purpose of a project which drives it forward towards its goals.

Purpose also envisions to efficiently and optimally meet and reconcile the current needs of the target market and of key stakeholders. It surely provides assistance to inch towards to achieve and fulfil the vision of:

A step into the future and upcoming markets

Executing future projections and targets

Overall Improvements in performance

Growth in terms of more revenue

New business opportunities

Benefits of a definite It is an outright advantage for an organization to have a well-defined purpose(s) right at the onset. It is also important in a way for:

Furnish and dispense accountability for all internal stakeholders of the organization and all external stakeholders that include

client/consumers and even Vendors at times. It is of paramount importance to allocate exact roles, expectations, and associated outcomes beforehand to all such parties involved. Purpose of the transformation is deemed to be prime in all engagements and mutual collaboration is sought via collective efforts.

Assisting in securing the critical commitments levels from both internal and external stakeholders. Securing the commitment means securing the active engagement of all these stakeholders. This active engagement comes very handy when the project is going through shaky waters or is facing critical risk situation.

Certifying that all the transformation project steps, tasks, efforts, and initiatives are aimed at retaining and achieving the fundamental and core purpose of the organization. It ensures that the organization stays well within the scope of its core values. It is imperative for an organization to not deviate from those core values and principles with which it was formed. Any such deviations result in loss of customers and clients which means loss of revenue.

Envisioning and defining the required or desired end state of the IT infrastructure. All project plans, resourcing, budgeting are done only when the final target state is envisioned. Every plan or task starts taking shape and comes into development once desired target state is ready, finalized and approved by all the internal and external stakeholders.

Establishing, refining, and providing the ways and methods of communication. It is predominantly important to setup an effective

communication channel between all the stakeholders to ensure success. Regular communication eases any sort of differences which is the basis and sign of triumph.

Engaging the workforce that executes all these strategies and meets objectives on the ground level, the employees. A transformation approach and initiative make it indispensable to engage all the employees that are part of the organization or the associated project. All employees must have a clear idea as to what is desired and how it is going to be achieved and via which ways and mechanisms. It is utterly critical to keep all the employees or the people engaged and informed which are associated with the Transformation Initiative or project. Employees' engagement always underpins the Transformation approach and makes it reach its goal it aspires.

Forecasting and earmarking the monetary part where in devoting the appropriate funds to run the show. Not only allocating the budget, but also a treasure in the form of a Project Manager along with a PMO team (Project Management Office) are allocated to track, trace, allocate, assign, spend, save and report everything about the budget.

Crafting and designing a highly agile, methodical, structured, cost-effective, and coherent solution that is future centric. A solution which is able to scale up and down as per requirements and is able to accommodate critical parameters like automation and innovation as well. A solution that is clear of all sort of hurdles, reduces risk factor to its lowest value and promptly accelerates the overall deployment of IT Infrastructure.

Challenges

Transformations are as tough as they can be in the entire IT landscape. Transformations at times are very difficult to execute and build a momentum. There is a numerous challenge for a firm going for transformation. Even for a client who is looking to outsource their business and looking to transform entirely, it becomes very difficult at times for the outsourcing firm to execute the project within specified timelines or cost for that matter. Let us have a deeper look at all these challenges.

Transformations are always easier said than done. Transformations bring about fundamentals level change for any organization. Transformations are entitled to make grass root level changes and thus they are deemed as supremely risky affairs.

As per a study, as many as 75% of transformation projects fail or get stuck midway:

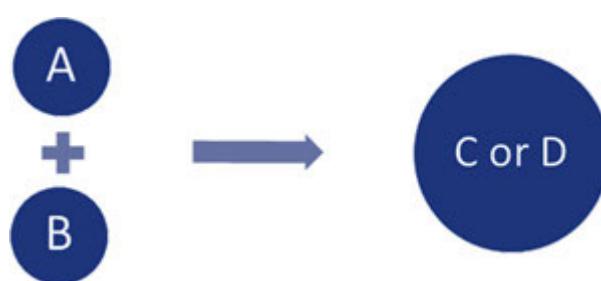


Figure 3.8: Transformation calculations

If you look back, a lot of organizations were founded during pre-IT era. This means, that the organization grew up on its feet with an entirely non-IT approach and support. This also means that whatever IT Infrastructure is running in the compounds of these organizations, it is either very old or very obsolete to say the least.

As per a research, about 68% firms that started their business in pre-computer era or before the computers came into picture, are facing issues such as:

Legacy applications

Very old system hardware

No redundancy

No mechanism of incorporating modern applications or technologies

Old strategies and business models

Budget constraints

Unskilled staff

No definite strategy, vision, and purpose

Lack of willingness to move out of comfort zone

Hostility from old guard

A transformation initiative, project, or an approach will most likely make an impact on the:

Organization and its existing hierarchy

Current strategy and work order

Corporate culture

One of the leading research firm Gartner recently conducted a research on the same. As per their research, it is the corporate culture that in actual is the top barrier for a successful transformation. This particular challenge is as stiff as the cost factor itself and certainly needs to be climbed down to nurture an organization on to the path of transformation.

A positive corporate culture embedded with a vision to embrace the new and modern technologies is what is required from the top leadership of a firm. This is essential to achieve and crave for a subsequent future state. Along with corporate culture, challenges such as enabling communication channels, getting resources skilled, enabling and establishing new processes and training existing staff with new process and technologies are as daunting as they can be:



Figure 3.9: Challenges

Here are some of the top listed challenges which are faced by an organization/project while going for transformation:

Lack of Transformations are generally perceived by organizations that have faced and executed a number of transformation projects and possess ample experience in their kitty. A transformation requires a lot of planning, discovery, collaborations, assessment, analysis, and so on. And all this is followed up by execution. Each and every of these steps are dependent on each other and thus prior experience is absolutely critical while working on a transformation project. At times, organizations employs new and less experienced professionals in transformations which at times lead a project to a divergent direction that is other than what is desired.

Capability Resources are Gold as they say!! Professionals with prior and rich transformation experience are what you seek when executing a transformation project. A transformation initiative or project requires team to do multitasking at times and take ownership while working on their part of the responsibilities. Thus, things may become extreme for a new executive to keep things at bay and strike the right balance with required pace and precision. And, any delays in any of the assigned tasks will result in project going off-track and will eventually result in higher costs.

Financial It is always the financial ingredient that is and always has been the main and deciding component. The very facet of cost allocation lets the project go off the blocks. And the limitation of the same cost, lets a project go back into hiding. Let us be honest, transformation projects are exorbitant in nature and thus require the executing organization to make a large spend first up. At times, organizations are not able to control or track the costs which tend to shoot up and lets a project get an abrupt end.

Stakeholders A transformation program would have a number of stakeholders from almost all the parties which are involved. Stakeholders are normally the owners, creditors, or belong to the senior leadership team. It is a requisite to bring every such collaborator on board and on the same page. It is often observed that many or key stakeholders are not in full agreement either with the strategy, purpose, vision, solution, approach, or the way things are being managed by executing team. Lip Service also plays a very important role here.

Conflicting Opaque, fuzzy, and conflicting objectives add to the twist of a transformation project. Last thing a project manager will need is various teams competing to execute the same set of tasks leading to sheer wastage of time and money and addition to chaos. This brings a bad reputation in front of the end client as well as everyone tries to hog the limelight and steal the credit. This ultimately spells disaster for a transformation project.

Scope Addition of scope means additional work with renewed planning along with change in scheme and every possible dynamic of the entire program. It means to re-spell the plan of action. Addition of scope may not go down well with the project as it will also entail additional costs and starting few of the things from the scratch and may impact tasks that are in execution phase as well. Bigger constituents may cause havoc in the program.

Overall A transformation project seeks active and well-managed administration for each performing resource and component to be crafted. Lack of willingness or a substandard mechanism of managing daily transactions will lead to things falling apart. Every dot has to be well connected with each other and internal integration needs to be seamless. A unified version of scope, plan, execution and summary is required and can be achieved via potent and constructive administration.

Key steps

The equation is very simple either transform yourself or perish. It is going to be extremely challenging for an organization to survive in today's fiercely competitive world without transforming itself. It is of paramount importance for an organization lay down a strategy, define their approach and milk out the way implementations will be done. In fact, executions are the most gruelling and burdensome task in transformations. It really needs someone to devise and compose a step-by-step plan. This plan needs to incorporate all the key steps, for each and every phase and produce documentation with pivotal information.

This documentation is known as build plan in transformation project life cycle. This contains all the critical activities that are to be carried out from all the participants of the program ranging across all the sections or pieces of the program including the implementation phase that is the most difficult phase. For example: it should include key steps for:

Assessment reports

Analysis done on assessments

End-to-end implementation methods covering all dimensions

Validated architectural diagrams

Applications stacks – dependencies

Network Information such as IPs change/vLans
extension/hostname change/firewall ports/load balancer dependency

Processes such as processes like change request

Communication mechanism to all stakeholders

Timings for implementation

Finalized templates for emails/reports/artefacts presentation

Type and frequency of reporting/status updates

Bridge call details for daily and weekly calls

SPOC details from each stakeholder

Resourcing/teams details

Escalation matrices

RACI matrix

Training details

Risk registers

SIPs/SOPs to execute tasks in desired manner

Vendor details

Testing mechanisms

New technologies/tools technical charts

Proof-of-concept exercises and so on...

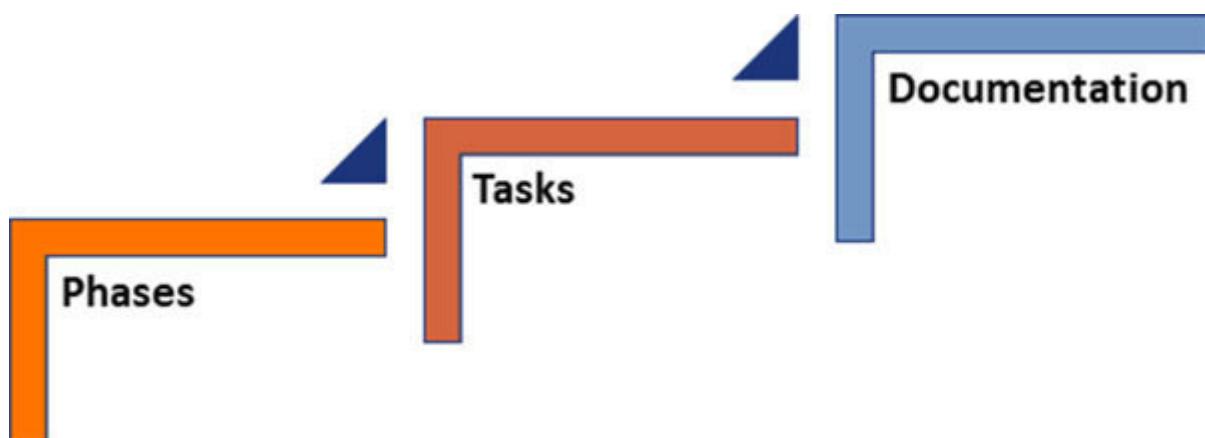


Figure 3.10: Key steps

A transformation project or program cannot be executed without creating a solid foundation that lies on its documentation. Every single step is treated as a key step and is treated with utmost importance. Documentation of key steps displays experience, capability and maturity of an organization to handle such

mammoth projects. It is in fact applicable to every other industry as well and not just to the IT industry.

However, an IT transformation strategy must include below prime and foremost important steps:

Offload management overhead.

Advanced and modern target state for existing and critical IT infrastructure

Improve efficiency and agility in terms of performance

Reduce operational expense

Use technologies like automation to carve out human errors and risk reduction

Fasten up IT processes to quicken up the provisioning and deployments

Stimulate and spur up delivery and consumption of IT services

Realign manpower and associated roles

Conclusion

This marks the conclusion of this chapter on Transformation – Lets Usher or The Beginning. We have gone through the logics behind a transformation project as to what it means and what is the purpose behind such a mammoth project. We have gone in detail as to what is the associated strategy and vision and what are the challenges faced by project teams. Important is to remember the key steps which are taken here.

In the next chapter, we will look at project lifecycle of a transformation project along with a full-fledged process flow-chart. We will also go through the phases of project life-cycle and will learn as to how things flow.

Points to remember

A transformation project is carried out for making changes in IT
changes to adaptor to evolve

A transformation project needs a definite strategy and vision

A transformation project has strategy, purpose and goals in its
core

Challenges are mitigated with the help of leadership team and
Project Management Team

CHAPTER 4

The Transformations Project – Lifecycle and Process Flow Chart

Introduction

In this chapter, we will read and understand about a project lifecycle and its associated phases. Who creates these phases and on what basis? We will cover the types of lifecycles and then we will go through process flowchart and its importance. Thereafter, we will have a look at a complete and full-fledged process flow chart for a transformation project followed by interpretation and explanations for each and every phase.

Happy reading dear readers!!!

Structure

In this chapter, we will discuss the following topics:

Lifecycle

Its phases

Process flowcharts and as to why they are being used

What are the benefits of using a flowchart?

Detailed process flowchart for a complete transformation project

Detailed elaboration for each phase

Objectives

Our main objective in this chapter is to learn about a Transformation Project's:

Lifecycle

Its phases

Process flowcharts and why they are used

What are the benefits of using a flowchart?

Detailed process flowchart for a complete transformation project

Detailed elaboration for each phase

Lifecycle of a project

A project undergoes a dynamic and varied lifecycle in its span. This lifecycle is in fact a complete sequence of various phases or stages right from its initiation or from when it was conceived first till the point it goes for a complete closure. This lifecycle is a well-defined and conceptualized framework, which comprises a set of or rather sets of definite, distinct, and recognizable high-level junctures to plan, organize and transform a design/scheme/idea/project/proposal or proposition into actuality. In addition, let's not forget, all this is to be done in a very efficient, methodical, and well-organized manner.

A project lifecycle represents the way a project has been taken up and conducted by an organization and all assigned project team members. An effective and systematic life cycle with a structured approach and clear roles and responsibilities ensures the success rate of a project.

A lifecycle in actual presents an opportunity to undertake project tasks in a standardized and systemized way. A lifecycle is known to strengthen and support the main cause of the project and makes things easy and smooth when it comes to final executions or deployments. As always, final executions or deployments as per current trends, are the most difficult phase or stage of a transformation project. It is also the last stage of the project. Thus, if results are satisfactory from the previous phases, then deployments are not expected to go seamless and might have a

huge impact. Thus, it makes it all the more important to disengage a transformation project into different phases as part of its lifecycle.

Now, the question which arises here is that who decides about a project and its lifecycle? Well, it is always the leadership team who gets first hand on the project. They are the ones who decide about the project management team. It is this management team that decides about various stages of a project lifecycle and the order in which they appear or follow each other. However, there are a quite a few constituents to check before a project is assigned a lifecycle and is segregated into various phases, such as:

Project scope

Organization strategy

Project domain

Area of application

Client requirements

Project geography

Number of resources required

Resources availability

Associated budgeting

Potential and visible risks

Perhaps, one of the biggest challenges which not only derive a project but also impacts its outcome is the lack of clarity of objectives and scope.

Management or the leadership team while considering and structuring project team, will keep project's objectives and scope right at the top of all the factors.

Not to mention, project lifecycle is entirely reliant on successfully achieving its objectives and its scope:

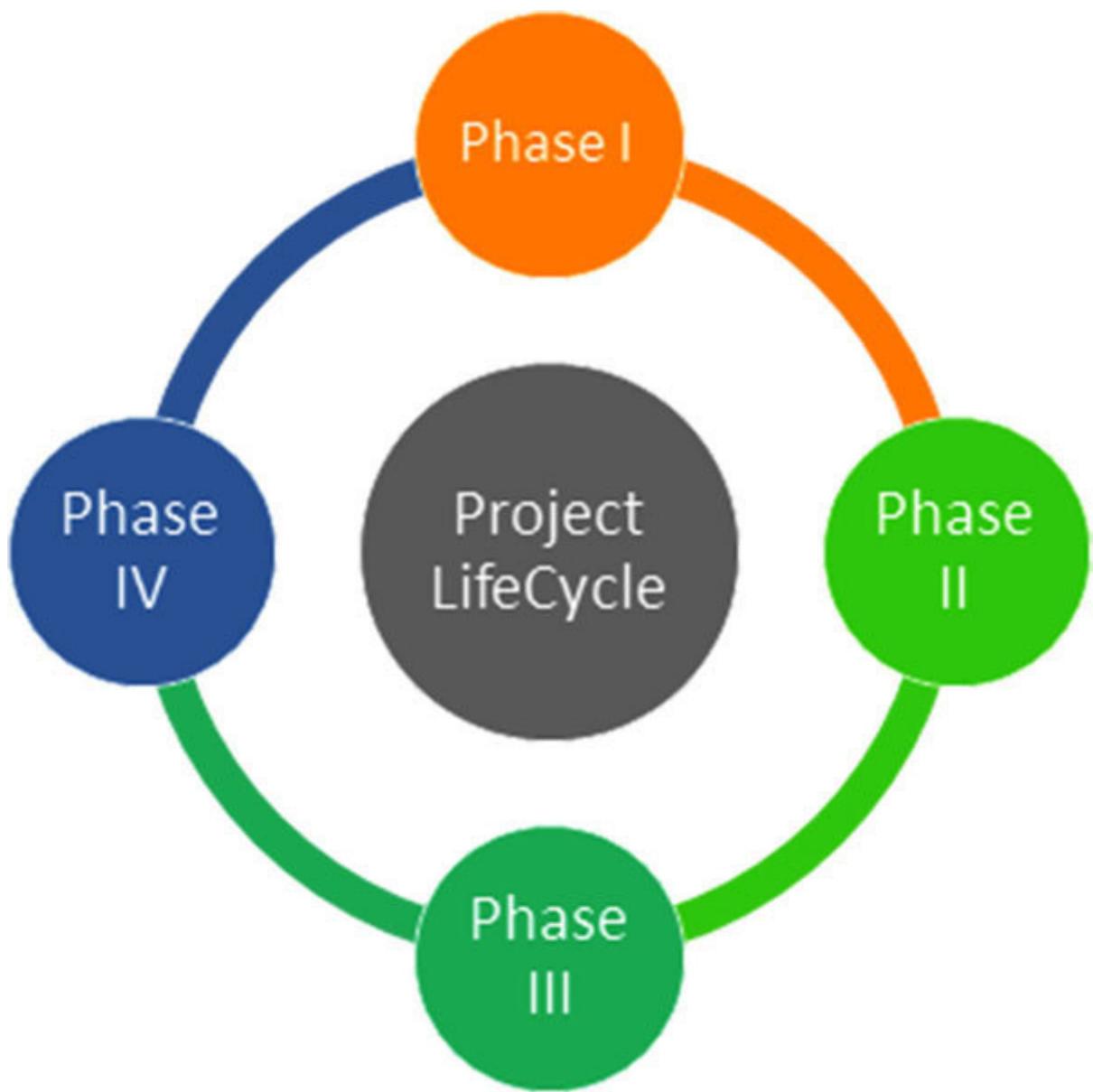


Figure 4.1: Project lifecycle

It is worth mentioning that project lifecycle once factored is not changed very often unless of course there is some contingency. All the decisions are taken right at the start of the project regarding lifecycle stay intact. Too many or persistent changes in project management, resources, or its lifecycle can easily get a project to go south. However, everything can change if client changes the scope of the project and its deliverables.

Each phase of a project lifecycle is assigned with explicit dates to start and to end the project along with a control monitor or point. Every phase is associated with timelines and constraints. Each phase, timelines, and controls are driven by the requirements, strategy, scope, and objectives. Even though each phase may have a similar or common goal, the duration may vary for each and every one of them. Duration for each phase depends on the type of activities that are to be performed by the project team or by the transformation team. Duration of a project is guided via objectives, deliverables, and activities which often tend to vary widely. However, it is basic principles of lifecycle and its essence that keeps a project intact and integrated and lays a basic and a very solid foundation for things to materialize through each phase and eventually amalgamated and contribute into the final product. Here are some of the listed benefits:

It brings clarity

It assigns clear guidelines as to what needs to be performed when and by whom

It helps to track the project against measured timelines and desired results

It segregates projects into smaller chunks and makes it easy to manage things

It guides and assists teams to be more focused towards their set of responsibilities

It brings the best out of each professional team

It makes the communications flow over and across easy

Types of lifecycle

A project's lifecycle is dependent on the nature of the project as well. Almost all the times, project specifications are discussed and finalized right at the start of the project. It is due to the essence of the project and also due to the classification. In such a case, all the major decisions are taken at a very initial stage only and then the project lifecycle is developed with all the required ingredients.

Once project lifecycle is developed and finalized, it is not subjected to any sort of alterations. Any changes if required, they are first addressed to the senior leadership team along with:

Root cause analysis

Impact

New expected timelines

Additional cost

More resources

Probably a new solution as well

All these specifications are communicated and are addressed very meticulously as they were not expected at the start of the project. It is then on the discretion of the management to decide whether to adopt and accommodate these changes. This type of project lifecycle is classified as static lifecycle.

It is also called as plan-driven approach as all the phases along with all the resources and tasks are identified and finalized right at the start of the project itself. Everything in such a project works as part of a plan that is used to formulate a strategy and purpose along with setting of definite goals.

Deviations are not expected in a plan-driven approach. In fact, expected results are marked with the end of each phase along with framed timelines. Thus, any change or variations in terms of expected results in any of the life-cycle phases is termed as part of the risk.

Such risks are documented along with the project lifecycle and are tagged with Importance severity. They are addressed and resolved through available mechanisms or with the interference of the leadership team. It may involve the client team as well to resolve some of the risks which are threatening to take the project off-target:

Project LifeCycle



Static / Plan-Driven

Dynamic / Adaptive

Figure 4.2: Types of project lifecycle

However, not all the projects are static or plan-driven. At least, they do not go by the rules of static or predictive lifecycle. A few projects go by the rule of evolution where in project lifecycle keeps on evolving itself as the project goes on.

As mentioned earlier, the project lifecycle is derived based on the nature of the project which entirely depends on the classification of the end product or service(s) which is to be delivered. In some cases, the end product needs a lot of iterations or reiteration before it is considered ready for the next phase or if the results are good enough to move onto the next phase.

This reiteration simply means that a few phases of the project lifecycle will undergo repetition or repetitions for that matter.

Repetition of the phase will signify the repetition of everything that has been part of the phase. This will include all the tasks and responsibilities associated with that phase of the lifecycle will be repeated until desired results are achieved. It also involves doing a lot of research as well to assess and validate as to which solution fits the best.

Not to mention that this includes all the labour work, technology usage, engagement from all required stakeholders including vendors and sometimes clients as well. These iterations may get repeated a certain number of times. Thus, it becomes very dynamic in nature and the next phases of the project life cycle are not even started until the desired version of the product is prepared. As the project continues, the scope of the project is always redefined as per the iteration which is being repeated.

That is why the lifecycle for such projects is never predictive but it is always dynamic and sometimes goes on for a long time due to reiterations of different phases over and over again. These lifecycles are dynamic in nature and hence are called so. Sometimes they are also referenced as adaptive life cycles as they continue to evolve.

It is worth mentioning that dynamic life cycles are specially crafted with a specific intent of designing and building a not an ordinary product. Such a project keeps all the stakeholders engrossed and every activity is very keenly monitored. Such projects are generally

less expensive as a very limited amount of resources and a bit of team is required to carry out the daily tasks.

Process flowchart and its importance

A process flow and associated flowchart is a mechanism to depict the flow of things that are to be done in an orderly way. This process flowchart basically manifests the sequence and succession of procedures/tasks/processes involved in the accomplishment of the project.

A process flowchart would furnish details about processes that are to be followed across the length and breadth of the project. It normally covers the entire project lifecycle of the project which means covering every single process through all the phases of the project.

Flowcharts are also designed for specific tasks or for a process that is different than the others. At times, flowcharts are drafted to specify a specific process. However, with each and every flowchart, the common aim is to let the project run smoothly and ease out all the processes.

Process flowcharts are deemed very important in the lifecycle of a project as completing a project is not an isolated task. It is an outright team effort and is the epitome of professionalism, collaboration, skill display, and ownership. But it is required that each and every member of the team would need to be thoroughly acquainted with all the different procedures and preparations.

In most of the cases, project teams have to get fairly acquainted with client-based processes as well. In fact, at times, client-based processes are embedded into project processes. Thus, it makes it all the more important to be familiarized with every process and their flow.

A process flowchart would ensure that there is zero or minimum of errors being made by the teams. A process chart will clearly depict the point where in the teams have to communicate with each other and this helps to smoothen up things all across.

Here are some of the benefits offered by fabricating process flowcharts:

Brings transparency and clarity

Lists down all the processes associated with the project/organization

Segregates processes for various phases of the project lifecycle

Streamlines all the prevalent processes and also marks the addition of new ones

Ensures that project is not slowed down due to processes

Displays every step associated with the process

Makes it easy to follow

Meets industry standards

Risk identification and mitigation

Let us have a look at a sample process flowchart with a project lifecycle comprising of different phases:

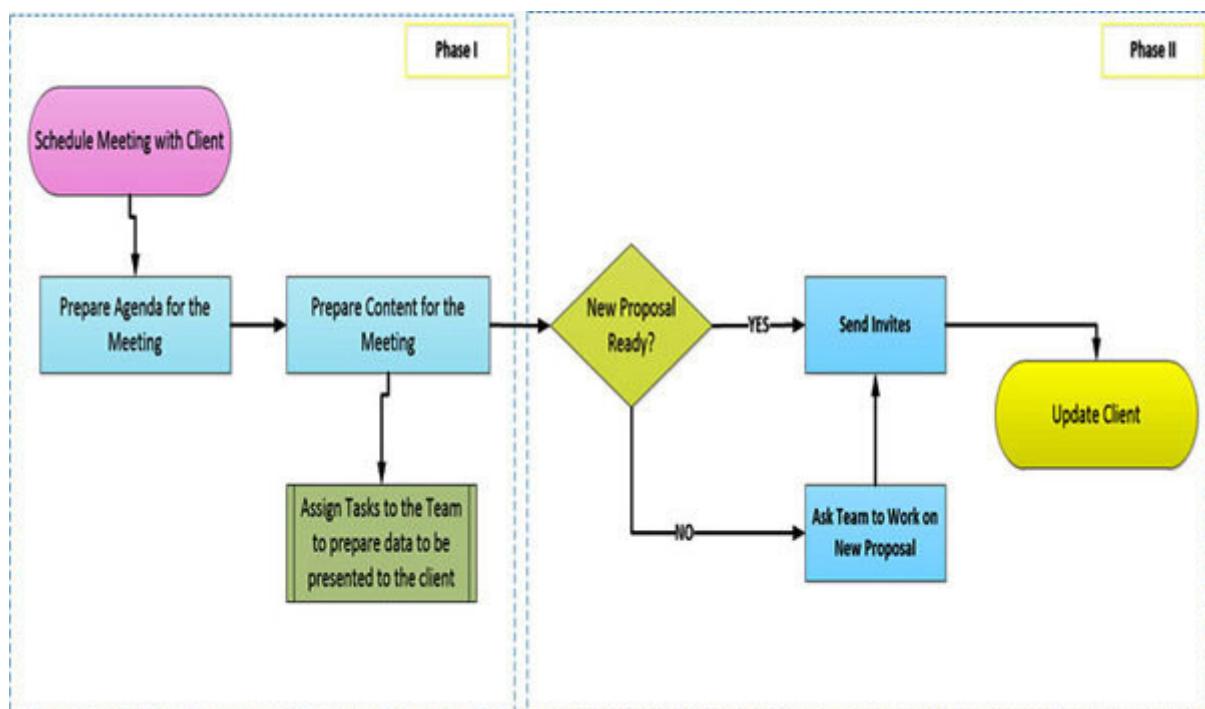


Figure 4.3: Process flowchart

As mentioned earlier, a transformation project is segregated into various phases which all constitute and form a project lifecycle. The preceding flowchart illustrates as to how a transformation project is bisected into various phases and executed.

Process flowchart depicted above displays the processes that are required to run in parallel to each other and also which need are independent of others. It clearly marks the sequence between all the processes over the course of the entire project lifecycle.

Flow charts are used by Project Managers, PMOs, and even by the technical teams. Even if there is a technical task that is to be performed, it still can be divided into various stages and teams. And to make things crystal clear and bring every member of the team on the same page, technical leads normally carve out the entire process into different stages and present it into a flowchart.

A flowchart not only helps to establish a logical order of how to do things but it will also help to trace the current situation of the assigned tasks to the concerned team. It will provide a clear status report as to where things are at this point of time and what exactly is happening and by whom.

This makes the performing team well aware of the consequences and they tend to ensure that there are no lapses from their end. It can also be seen in a way where in process flowcharts helps to gain the maximum of all the performing team. No doubt, process flow charts are an industry standard and are used across the domain and industry to improve project efficiency.

A process flowchart also helps to build a strong rapport with the clients as a client as it sets things straight. It ensures that there are no slippages. A flowchart would guarantee that the agreed roadmap is absolutely clear to the performing team(s) and they have been thoroughly guided as to how to efficiently complete the

task or the project while adhering and abiding by all the laid-out rules, compliance factors, and processes. It shows and reflects the rigor of the performing team and most certainly ensures to provide a very satisfactory and long-lasting experience to the end client.

Process flowchart for a transformation project

Here is a detailed process flowchart for a transformation project:

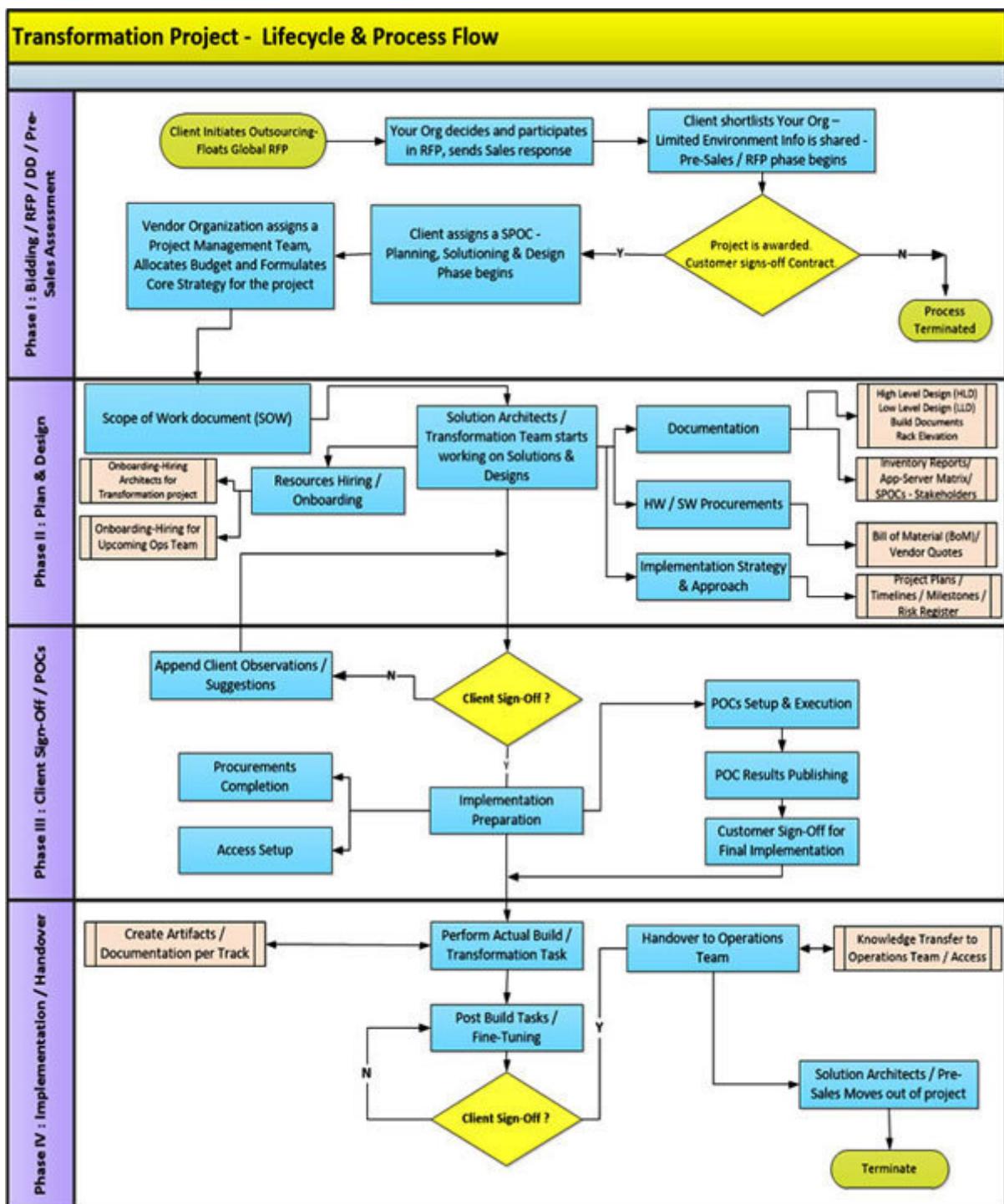


Figure 4.4: Transformation project process flowchart

Above depicted is a pictorial view of a full-fledged transformation project that has been segregated into four different phases. Each phase has been developed with its own set of responsibilities.

Interpretation – Phase I

Phase I of a transformation is usually the slowest of them all. This is the stage where a client in some part of the world realizes or feels the need that his IT environment needs transformation. These decisions are taken on the basis of:

Current market trends

Orders received

Eyeing new markets

Upgrades required for specific applications

Client requirements

Portfolio upgrade

Loss of revenue

Outdated and poor infrastructure

High maintenance costs

Higher internal expenditure

Once a client decides that he is going to outsource his business/a part of business/to transform his IT environment, he will initiate the process. This process is called as RFP or request for proposal. It is also called as **Expression of Interest**

A **request for proposal** is a standard business charter that declares and bestow details about a project, as well as solicits bids from contractors who will contest and help complete the project.

Almost all the industry players prefer using RFPs along with many governments as well. RFP is the standard process and an Industry established process used by clients worldwide to initiate a process for:

Business outsourcing

IT transformation

Environment migrations

IT consulting

Inviting new bids for any small or large projects

Initiating an RFP process would mean that client is inclined and prepared to source his business a vendor. These RFPs are very open in nature and are opened with a specific set of guidelines

and instructions. It is always on the cards that the organization opening the RFP, will be requesting for specific:

Skill sets

Domain

Industry type

Level of expertise

Geography

Prior experience

Organization size

Annual turnover

Global reach and presence

Certifications with respect to organization level and resource level

Once the RFP is initiated and floated globally, all potential service providers or vendors respond to the RFP. An RFP is always floated with a timeframe and any service provider interested in applying for the response has to abide by the timelines which are mentioned in the RFP:

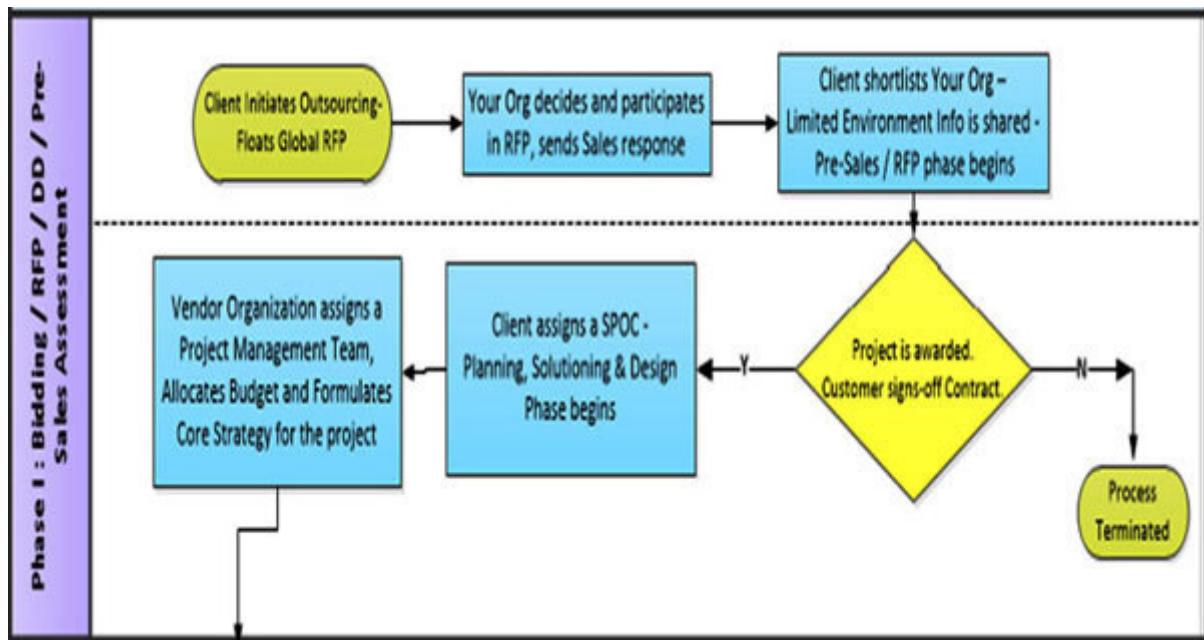


Figure 4.5: Transformation project phase I

RFPs are worked upon by the sales and solution teams of an organization. Generally, each organization possesses a sales team that is solely responsible for working on RFPs. The sales team of an organization has a presence in every part of the globe. The regional sales team of a service provider operating in the region in which RFP has been floated, will enquire and acknowledge the RFP if it suits the profile of their own organization. Every Interested service provider is expected to prepare and submit their respective response within stipulated timelines.

Post submission, all the submissions are thoroughly reviewed and assessed. This review process may take a few weeks or a few months and results are announced as per mentioned dates. During this review, the client evaluates and shortlists a few potential vendors or bidders. Criteria for selection are again:

Experience

Exposure

Vendor organization size

Estimated cost

Domain skills

Financial status

Shortlisting potential candidates marks the beginning of pre-sales/RFP process. This is where sourcing clients or organizations would normally provide a little glimpse or details of their environment w.r.t. its current stage and status and what they expect to be done with it.

It is this piece of information and glimpse of environment overview that is taken as a reference and response to be prepared by all the selected participants to prepare potential solution for client pain areas.

Sometimes, vendors participating in the RFP may request for a bit more information. Once again, there are timelines to submit proposed solutions and every vendor bidding for RFP has to meet the dates.

All participating vendors are expected to submit their respective solution documents that are a PowerPoint presentation at this point of time before the final submission date. Thereafter, client once again reviews every solution document and awards the contract and the project to a suitable vendor whom he feels is the right candidate to perform the tasks.

This review process of solution documents is lengthy in itself and sometimes takes about 2–3 months, however, it varies from client to client. This review process involves deep technical reviews and financial evaluations (also known as T1/L1 process) within client fraternity and includes technical inputs from various client organization technology heads and administrators as well.

Once the contract is awarded to a service provider, a joint (kind of) management panel is setup. It is important to note here that this is the phase where vendor organization assigns a project management team for the project, allocates budget, and formulates a core strategy. Whereas the client assigns a **Single Point of Contact**

This SPOC interacts with vendor organization and helps them with:

Required access of environment

Required information

Mitigate any risks

Identify any loopholes

Ensures a gap-free delivery within designated timelines

Track and trace spending and funding

Get required approvals

This SPOC acts as mediator between client and vendor organizations and he helps them project to kick off.

Interpretation – Phase II

The second phase of a transformation project is for plan and design. This is one of the most crucial phases of a transformation project lifecycle. This is the phase where in the project gets off the hook and starts taking a shape.

This phase starts once the contract between client and vendor organizations is finalized and signed with all the terms and conditions agreed mutually and officially. As mentioned, the vendor organization assigns a project team and client organization a SPOC to keep things tidy.

Post contract signing, **Scope of Work** is the next important document. SOW document is written by specific authors upon request either by client or by vendor (generally by client) but is mutually agreed and signed-off by both the participants. However, before signing off SOW, each and every fraction of the document is reviewed, assessed, discussed, and then agreed by both clients and vendor(s) organizations.

SOW is referred as Scope of Work or Statement of Work. A SOW document is an agreement or a working agreement rather between both client and vendor organizations and covers all the aspects of the project. This document is used to define the entire scope of the work that is to be carried out by the vendor organization. This document working deliverables from each and aspect which

basically means domains in terms of infrastructure technology along with specified timelines and a RACI Matrix. RACI matrix is a mechanism to clear out roles and responsibilities for each and every task. RACI means the following:

Responsible

Accountable

Consulted

Informed

It is important to mention here that SOW is a formal document and is being referred all along the lifecycle of transformation project. For any task, if there is any clarity which is sought, SOW is being referred by both the parties which is actually a routine exercise as everyone, from both client and vendor organizations, abide by SOW and everything which is written in it. Once SOW is agreed and signed -off, it is circulated to all the members of the project team on both the sides so as to everyone on the same page:

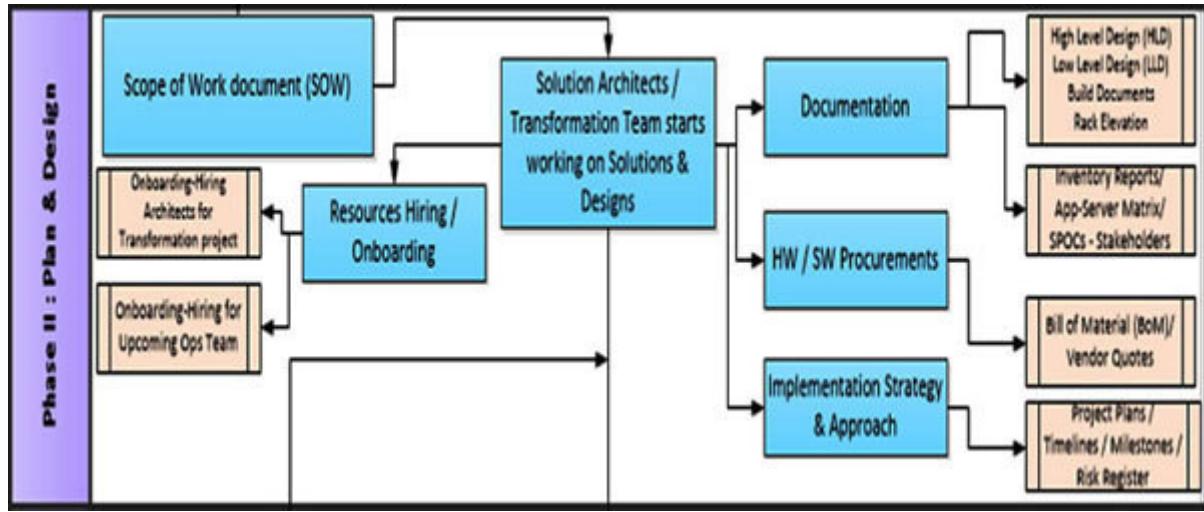


Figure 4.6: Transformation project phase II

Along with SOW, there are a lot of activities that are done in parallel in this phase such as:

Creating core strategy for transformation

Defining approach

Resources hiring and on-boarding

Preparing documentation

Bill of material (BoM)

Initiate procurements with vendor quotes for hardware/software purchases

All these activities are done in parallel and are carried out by various teams from different tracks or domains as per their requirements and as agreed with in SOW.

Creating strategy and approach is generally done by the senior leadership team. However, once the project is signed and comes into effect officially and the project team is assigned and onboarded, then project leads and managers are also involved in drafting a specific approach and construct an efficient strategy. This is where the entire project lifecycle and all associated phases are factored and conceptualized, based on the strategy which has been decided to be pursued for the project. Timelines, expected results, budget spending per phase, resources allocation, communication channel are also discussed, agreed, and stamped with each phase of the project lifecycle.

Documentation plays a very vital role in this phase. There is a list of documents that are prepared in this phase such as:

Assessment reports

As-Is architecture details

Inventory details

Application - server mappings

Dependency details

High-level design (HLD)

Low-level design (LLD)

Rack elevation diagrams and sheets

Build documents, that is, SOPs/SIPs/MOPs

Security – policies

Network details

Checklists

All these documents are critical to the theme of the overall transformation approach. Without proper design documents, it would be impossible to build and transform anything. But the documentation part always starts with doing a discovery and fetching inventory details that are then followed by preparing an assessment report.

Once the complete assessment of the environment is done, an application-server matrix along with dependencies is being created. It is only then design documents start coming to life.

Design documents are flag-bearers of the implementations. Design documents once created are sent to client for review and for any required modifications. All the aspects of a design including all

the technical aspects are discussed very thoroughly indeed. Success of the project depends on how designs of the project have been prepared. There are at times many presentations, workshops, meetings, or discussions between client Team and vendor teams for design discussions and reviews before actual sign-off takes place.

There is one more very crucial activity that takes place in this phase and that is the identification of all stakeholders which is of paramount importance. Most of the stakeholders are actually project influencers/business owners/delivery heads/senior management folks and thus it is always beneficial to know the guys who are actually calling the shots from behind. Identification of stakeholders is also important as they are normally required for:

Approvals

Reporting

Escalations

Budget discussions

Risk discussions

Interventions

Expertise

Emergencies

Decision making

Regular interactions and communications with key stakeholders are absolutely necessary for the project management team or leads. Keeping all the stakeholders up to-date with the latest progress updates and with a sign of things to come and to be done in near future will help the project tremendously. It not only brings a very positive vibe in the project but also motivates and rejuvenates everyone in the team. Most importantly, bringing all the stakeholders on the same page will help to resolve any conflicts or issues. Knowing client stakeholders helps in building a strong rapport which eventually helps to bring in required approvals and information. It makes things easy to coordinate with various business units within the client fraternity and any hurdles/hostility/no response issue can be resolved within a very short span of time. It certainly helps during procurements and design discussions which are key decisions.

Interpretation – Phase III

The third phase of a transformation project earmarks a knock on the door of final implementations and executions. In this phase as well, a lot of activities are carried out by the project team and that too in parallel such as:

Design reviews by client

Client sign-offs

Hardware/software procurements completion

Test or POC environments setup

Project team access on the client environment

Vendor(s) engagements

Final testing

Publishing results

Final approvals for implementations

This phase begins with reviews of the designs by the client and vendors (if applicable) which were prepared by vendor project team in the second phase. For each track for which designs are being prepared and transformation is pursued, the client assigns a SPOC for each of those tracks. Associated project team related to specific track from vendor organization, interacts with their respective SPOC from client organization for anything related to information, guidance, design, access, and so on. These client SPOCs are responsible for their respective track to get through with respect to design and solutions. They are entitled and aligned with the vendor project team from their specific track to help them with whatever data or inputs or directions for that matter as is required.

Design reviews are very crucial to the success of the overall project. Designs are prepared by Solution Architects/Design Team from Vendor Organization and they would use Project SOW as the baseline. Solutions are prepared with an aim to bolster the strategic roadmap that has been defined for the project for the next 2, 3, 5, or may be 10 years down the line. A solution in the proposal has to meet everything which the client is rooting for to modernize his IT environment. Every client or organization seek returns when they go for such expensive and mammoth transformations and thus the solution has to be:

Technically advanced

Modernized

Robust

Resilient

Resourceful

Compatible

Secure

Easily accessible

24x7 available

Cost effective

Meets latest industry lines

Basically, the solution should comprise of everything the client needs and something that will help thrive business of the client organization.

Once these designs or solutions are prepared, they are generally first reviewed by project leadership team or in some cases even by Leadership Team of the vendor organization. Most of the solutions carry a lot of price tag on them and the stakes are generally very higher which are associated with them. This compels senior leadership teams to go through the solutions or the designs which have been prepared for the high brand clients

or for the projects that have very high cost. Everyone on-board the team to review the solution will ensure that the solutions are matched precisely with the requirements mentioned in the SOW document and each and every jotted detail is checked, rechecked, and validated before they are sent to the client for their review and subsequent sign-off:

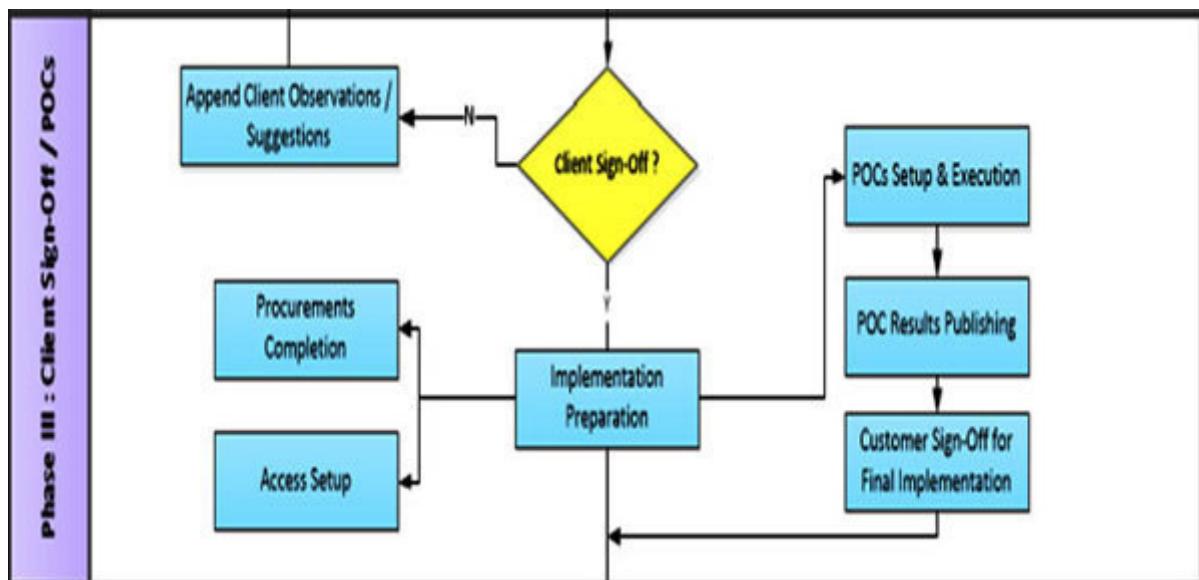


Figure 4.7: Transformation phase III

Client review is a very important activity in this phase of Transformation project and they are always planned well in advance. Once the designs are prepared by solution architects/design teams, they setup and schedule meetings with assigned client SPOCs with whom they are either mapped or aligned. At times, it takes more than a single meeting or workshop to get the reviews completed.

If the client has any suggestions or observations regarding the solutions, the project team has to go back and rework on the solutions and designs. In that scenario, the entire exercise is

repeated again and again until the desired sign-off is achieved which is a very big or rather huge milestone in a transformation project lifecycle.

Once the client signs off all the designs-solutions, project team begins with completion of procurements process. All the vendor quotes for all hardware and software assets are materialized post design sign-off and cumulative orders are placed with all shortlisted vendors. All the orders are placed with timelines. Hardware orders normally take about 28 days for delivery where-as software orders are provided digitally within a day or two. Along with procurements, project team has to coordinate with hardware vendors for Installation and setup as applicable. However, if that is the case, then they will also have to arrange for data center or client premises visit as well for the hardware vendor. Before any installations, project teams share signed-off architectural diagrams with their hardware or software vendors to bring them on to the same page and have seamless installations.

Along with procurements, post-sign-off, the vendor project team starts making requests to gain access on client environment(s). The level of this access is defined in the contract/SOW document. This access is used by project team to:

Validate the information received so far

Gather missing information

Familiarize with client environment(s)

Identify any unknown issues

Prepare documents for client processes

Mitigate any risks or issues related to access

In addition to their own access, vendor project team can also request access on client IT environment for their own respective hardware or software vendor if there is a requirement to engage a specific vendor. All such access-related requests are entertained and catered by specific client SPOCs assigned for each respective track or domain. Access is normally provided to all the performing members of the project team:



Figure 4.8: Test environments

Once hardware and software deliveries are completed, project teams start working on building test environments. Building test environments or **POC** environments **of** is of utmost importance for

everyone. With test labs or test environments, project team will build a prototype of overall environment and test out all the functionalities which have been proposed or are expected to be used. These test environments and subsequent testings are pivotal and decisive for the project as they would establish the credibility of the solution in practical sense and would strengthen client's trust.

Success criteria for these testing depend on as to how the nominated application, its various modules, database models, and overall transactions with respect to application accessibility and underlying infrastructure performance. Each test environment and subsequent testing/POC has identified parameters in terms of functional requirements performance. These parameters are required to be met in terms of performance for the testing to be termed as successful.

These functional and performance parameters are usually defined by Application Admins, Database Admins, and Infrastructure Admins.

All the testing is considered on top most priority by both client and vendor organizations. On occasions, even Business Unit or Delivery Unit heads from client-side join testing.

Even if a single testing fails, entire transformation project and solution for the respective track are jeopardized. All these testings that are carried out for each respective track by project teams or vendors, are very keenly observed by all the key stakeholders. This is because entire project hangs on a thread during these testing.

For each testing that is carried out, project team has to prepare a detailed report wherein they have to publish:

Exact end results

Duration

Method of testing

Tool being used

Functionalities impacted

Downtime (if any)

Resources involved

Any new risks identified

Issues faced

Once testing is completed, detailed analysis and results are shared with all the stakeholders including prominent client leadership. If the testing results are positive, then client would provide sign-off to proceed for final implementations or executions of designs and solutions and that would mark the end of this phase III. Positive results will surely designate the solution, and the features–functionalities as the ultimate solution. It would signify that the

correct solution was prepared and will help the client in the long term and to achieve desired returns.

However, if the results are negative, then the project team may have to go back to drawing board again and find out as to what is the root cause for the failure. They will have to analyse and assess the entire solution or the part which is related to their specific track. In the root cause, they will have to specify:

Exact nature of the problem

Component or the section which failed

Impact of the failure on the entire solution

New or already identified risk

Severity of the risk

Plan of action to mitigate the risk

Duration required for risk mitigation

Extra costing impact if any

Once assessment and root cause of the failure is identified, project team will then redraw or modify the solution documents and arrange for another set of testing and this process is repeated unless issues are sorted out. Once testing is deemed

complete and successful, with client acknowledgment, transformation project moved into its final phase which is for final implementations or executions.

Interpretation – Phase IV

The final phase of a transformation project lifecycle comprises of final implementations. This is the phase where all the planning, strategies and goals meet the final purpose. It is the implementation of all the final designs and solutions that have been proposed and tested through various phases and are executed to transform the IT environment of the client as per the signed contract and SOW document.

In this phase, final implementation takes place wherein a data centre or a cloud stack is being built or a service is being revamped or introduced:

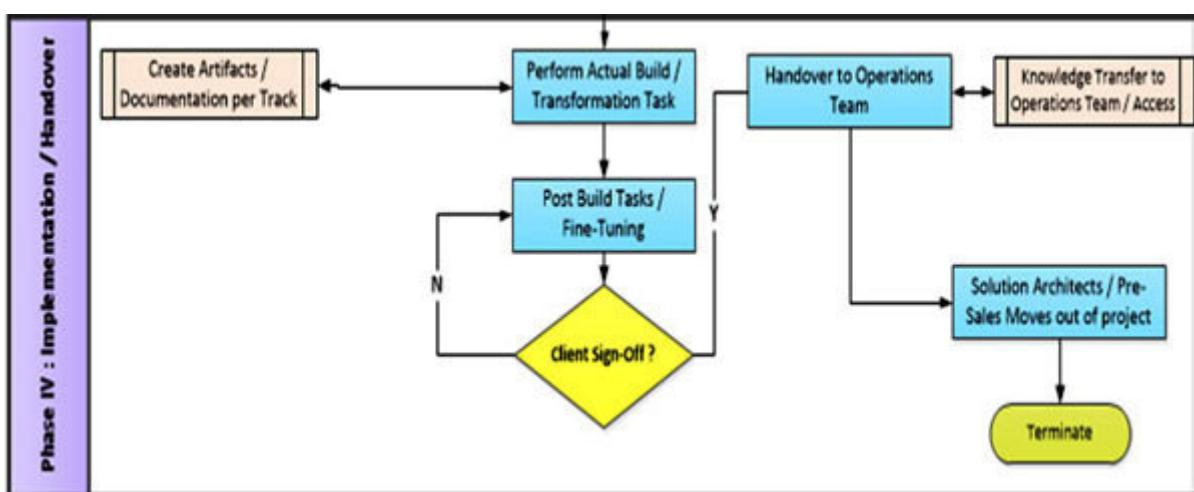


Figure 4.9: Transformation phase IV

During implementation and while building the solution, project team ensures that every step of solution implementation is being

captured and is thoroughly documented. Pictures, snapshots, videos are created as artefacts that are used as references. All these artefacts and documentation hold a great value. These artefacts not only serve as evidence and reference for validating solution build and implementations, same artefacts are used during compliance audits, both internal and external.

Audits are validation of an organization's adherence of industry standard guidelines and they have to be religiously followed. Thus, it is needless to say that these pieces of documentation and artefacts become glittering gold during audits.

Additionally, and again importantly, this documentation prepared by project team during final implementations is used by operations team to run the show. They would need this documentation initially to understand the environment which has been built and they would use this documentation to train all their current team members and anyone joining in the future as well. However, this documentation is shared with operations team during Handover only which happens post build and post client sign-off.

Getting back to final implementations, project team would ensure to build every aspect of the proposed solution is build and completed and is documented. Post solution build, entire environment will go for industry standard fine-tunings. These fine-tunings might just be suggestions from client as well who may want to tweak a few settings here and there to support a particular application.

Once Builds are finished, Acceptance Testing (UAT) are carried out to ensure that Applications are able to function as per the design and desired performance is achieved. UAT can be very thorough in this phase as it depends on the overall modules of the application and location of the user base.

Post solution build, fine tunings and UATs, newly built environment will be reviewed by customer and will be substantiated for all the tracks. As mentioned earlier, there are assigned client SPOCs for every track and they are responsible to make sure that environment has been built as per approved designs and solutions and is matching every bit in terms of services which have been defined in the SOW document. Once validated, they are required to provide sign-off to project teams for their respective tracks. These sign-offs are generally provided over the emails or through a PMO specified document.

Build sign-off would mark the end of transformation project for project team and they will start handing over to operations team. Operations team or Ops team as they are fondly called, will take over handover of:

Newly built IT environment

Documentation

Access mechanism

Artefacts

Information about all stakeholders

Processes information

Vendors details

Procurement details

Warranty and support structure with vendors

Post-handover and transition, operations team will start providing daily support services to the client which is a routine task as part of operations. Project team will support operations team for a few initial days or may be a few weeks so as to help them understand the environment and also to train them on latest technologies being used in building the environment. Slowly, project team will phase out of the project and that will be the end of transformation project and entire process.

Conclusion

This marks the conclusion of this chapter on Transformation – lifecycle and process flowchart. We have understood about project lifecycle and we have seen various phases as well. We have gone through each and every phase in details and we have walked across a complete transformation project lifecycle. We have also seen and learned different types of project lifecycle.

In the next chapter, we will look at various tools being used during a transformation project along with deep know how on client connect.

Points to remember

A transformation project lifecycle is a sequence of various phases

A project lifecycle can be a static or a dynamic one

A process flow chart is a mechanism to depict flow of tasks and phases in a project lifecycle

Lifecycle phases can overlap each other and also depend on each other

CHAPTER 5

The Transformations Project – Tools and Executions

Introduction

In the last chapter, we deep dived into the project lifecycle and its various phases. We have gone through each and every phase in details of a complete transformation project life cycle. We have also seen and learned different types of the project lifecycle.

Considering the transformation lifecycle, we need to identify a finalize transformation tools to actually carry out the transformation from source estate to target estate. Post identifying a transformation tool, it needs to be setup and configured in order to actually transform the source environment.

In this chapter, we will go through various transformation tools and they are used for creating solutions. We will also learn as to how to designs and solutions are prepared and what are the guidelines that are followed. We will see as to deeply the client impacts a solution.

Structure

In this chapter, we shall cover the following topics:

Transformation planning and best practices

Different types of discovery tools

Different types of transformation tools

Happy reading dear readers!!!

Objectives

Our main objective in this chapter is to learn about a Transformation Project's:

Design preparation and planning

Discovery tools

Transformation tools

Creating projects

Transformation planning and best practices

Generally, large-scale IT transformation projects occur over an extended period in a production environment that might span multiple locations. It is tough to judge future network activities as details are unknown at that stage. Complex projects with higher quantity of workloads may take more than a year(s) to complete. Therefore, transformation project planning is a must for successful transformation within agreed timelines post the workload discovery.

Finalizing the stakeholders and point of contact

The following stakeholders are of high importance in transformations:

Project managers (both from customer and supplier).

List of application vendors and owners for workloads on which applications are hosted.

Transformation managers/architects for creating waves and batches along with project managers in a project plan. They will also finalize the application and workload dependencies. They will also finalize the transformation tools, licensing, and so on.

Transformation specialists for actually carrying out the transformations and participating with application owners and vendors for finalizing the transformations and their cutovers. Below are some other responsibilities of the architects:

Creating standard operating procedures for the migration of the servers.

Creating standard operating procedures for the migration of the databases.

Raising appropriate change requests to abide by customer change management process and compliance parameters for transformations and cutovers.

Involved in setting up POC, staging, and cutover environments in the cloud and on-premises connectors.

Testing team to test the transformed applications in the target environment.

Reporting: As a part of project management, team members will monitor the waves and then update the status to the leadership about the progress on a daily/weekly basis. They work alongside Project managers and under their guidance.

Once stakeholders are finalized, planning and grouping of **servers** **into** Here plan the waves in terms of the following factors:

Application criticality (least critical applications to be grouped in first waves or even in Proof of concepts)

Environment basis (Dev/Test/UAT/Prod)

Or some other factor depending upon customer's business requirements

The project plan details each and everything about each application along with dates and timelines for the project

completion along with waves and groups.

Transforming servers and applications

Transforming workloads from their current operational mode to a future operational mode is the fundamental management goal for your transformation project. A transformation plan includes the following information:

Application dependencies and workloads dependencies between tired application infrastructure.

Approach of transformation in the current mode of operation and future mode of operation. that is, As-Is move, re-host, and data move or some other.

Transformation task order and sequence.

Dates and timelines

Project dates are decided as per business and network requirements. Business factors include criticality of the applications, downtime if they can survive, task dependencies, and the availability of resources. Network dependencies are based on available bandwidth, link speeds, and the amount of data being transformed.

Transformation tools

Discovery and assessment tools

The first step in transforming the existing IT state is to discover it first and do its assessment in order to decide if it can be transformed to the new estate whether in another data center or in any cloud environment.

There are variety of tools in the market that needs to be identified and selected based on the organization's requirements and needs.

Following are the popular tools used for discovery and assessment of the environment:

Microfocus Plate Spin Transformation Manager

AWS Application Discovery

Azure Migrate Server Assessment tool

Red Hat Migration Toolkits

The above tools help you plan the transformation and movement to Cloud provider by collecting data of the workloads present in on-premise data centres.

Generally, all the discovery tools work with the following approaches:

Agentless discovery

In agentless discovery, a connector virtual appliance from a particular provider is deployed in the on-premise data center. It can be in any of the virtualization environment such as VMWare environment managed via VCenter or in Microsoft SCVMM or Hyper-V environment or may be Citrix Xen environment. It collects configuration information such as server names, RAM, CPU usage, disk utilization for each VM, and physical servers present in the network range allowed for scanning by the data center administrators.

Agent-based discovery

In this approach, discovery agent is deployed on each VM and the physical servers to capture more granular information such as network interfacing with other applications, processes that are running. This helps in identifying the dependency mapping and then making a decision to transform the environment as a set of servers as tiered layers for the application being consumed by users.

Generic connector appliance architecture

There are multiple ways to deploy appliances via OVA file provided by Microsoft suitable for vSphere and for Hyper-V. The deployment can be manual or scripted.

Below is the generic architecture of any discovery appliance architecture:

Appliance starts discovery on any hypervisor management tool (vSphere VCenter or SCVMM or Hyper-V) on appropriate ports (443 or 5985 or some other).

Appliance then gathers metadata and performance data.

The data is then sent to cloud provider portal such as AWS migration hub or Azure migrate portal integrated with public cloud provider credentials. This data is then getting replicated to the cloud portal and inventory can be seen in the cloud migration portal.

Migrations can be planned and initiated as per agreement on the basis of assessment:

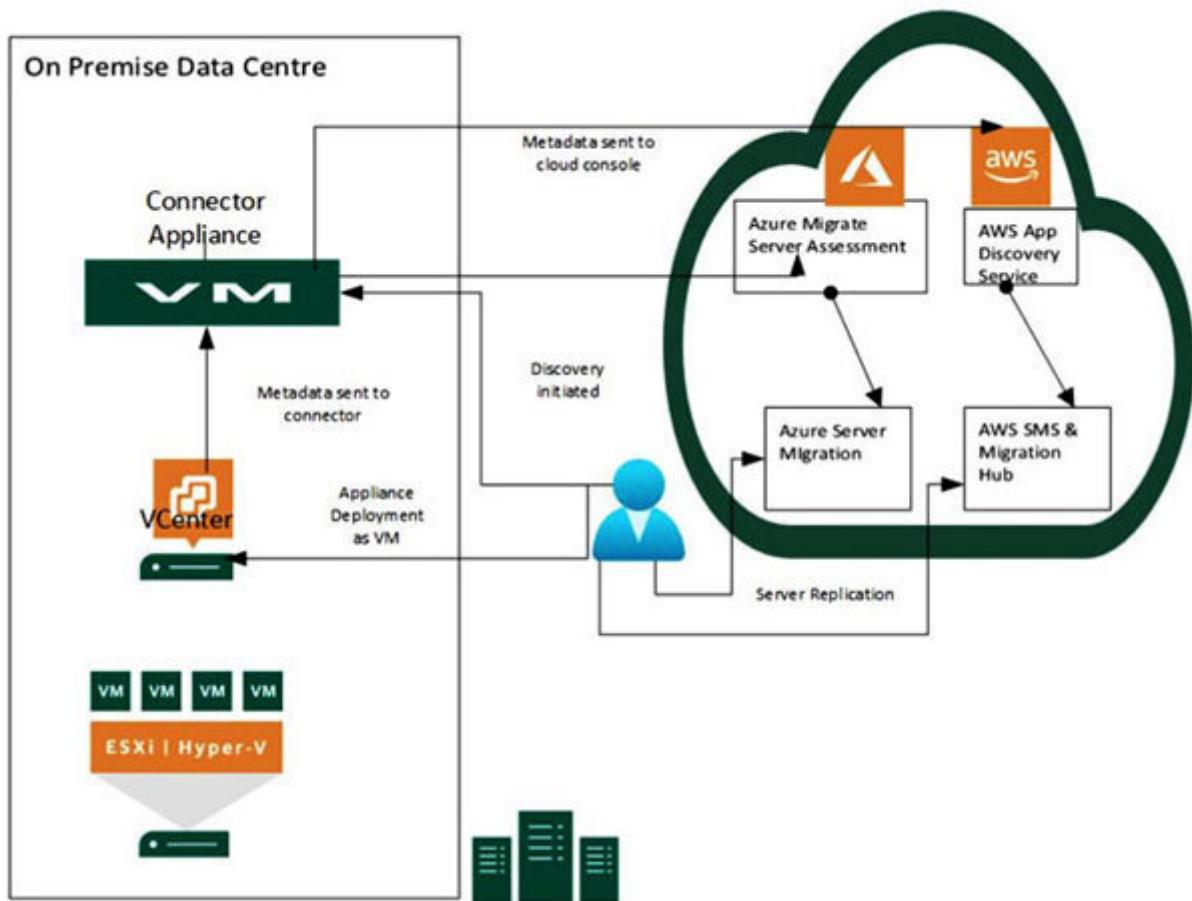


Figure 5.1: Generic discovery appliance architecture

In the On-Prem data center virtualization environment (VMWare or Hyper-V) Customer download the connector collector appliance and configures it.

Then the discovery is started. Metadata of the inventory is sent to the connector.

Connector appliance then sends it to cloud discovery service depending upon the cloud provider.

Replication can then be initiated for the migration in the cloud console.

Let us go through each of the tool and understand how these works at a higher level.

Microfocus Platespin transformation manager

Platespin transformation manager is a great tool for large scale transformation project and has the following features:

Management and keeping track of each workload for all phases, that is, from import to complete.

Transformation manager appliance also does automation of migration and also track manual migrations across multiple sites in combination with multiple migrate connector servers.

Access enablement via role-based access and based on projects in transformation manager.

Platespin transformation manager works with Platespin migrate to do the automated discovery of the workloads and hosts.

The automated discovery of workloads and hosts via import methods can be in form of spreadsheets, IP range provided, or by providing the details of the virtualization hosts.

It supports transformation methods such as

Physical to virtual

Physical to physical

Virtual to virtual

Physical to cloud

Virtual to cloud

Cloud to cloud

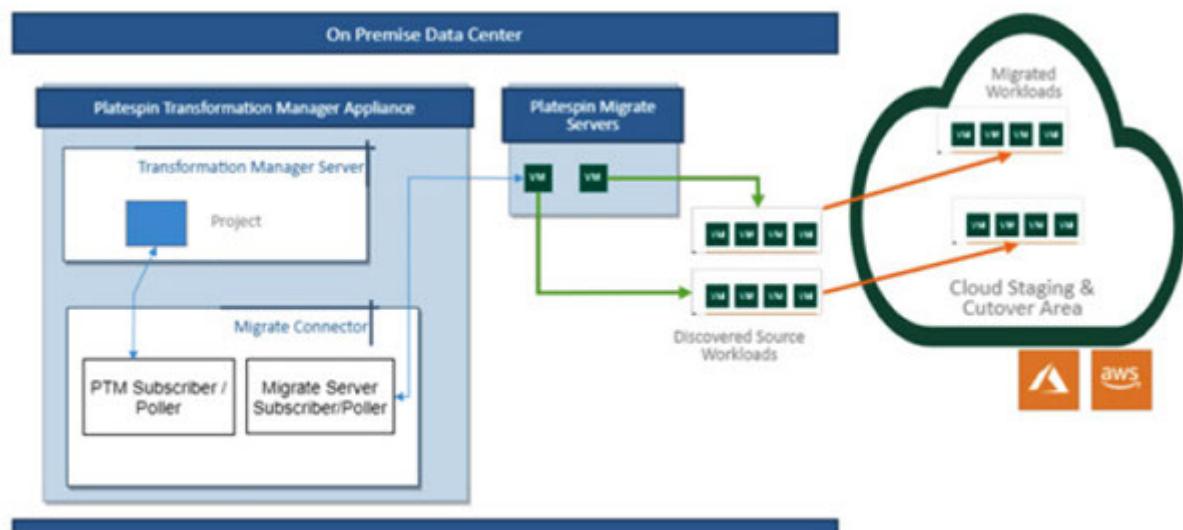
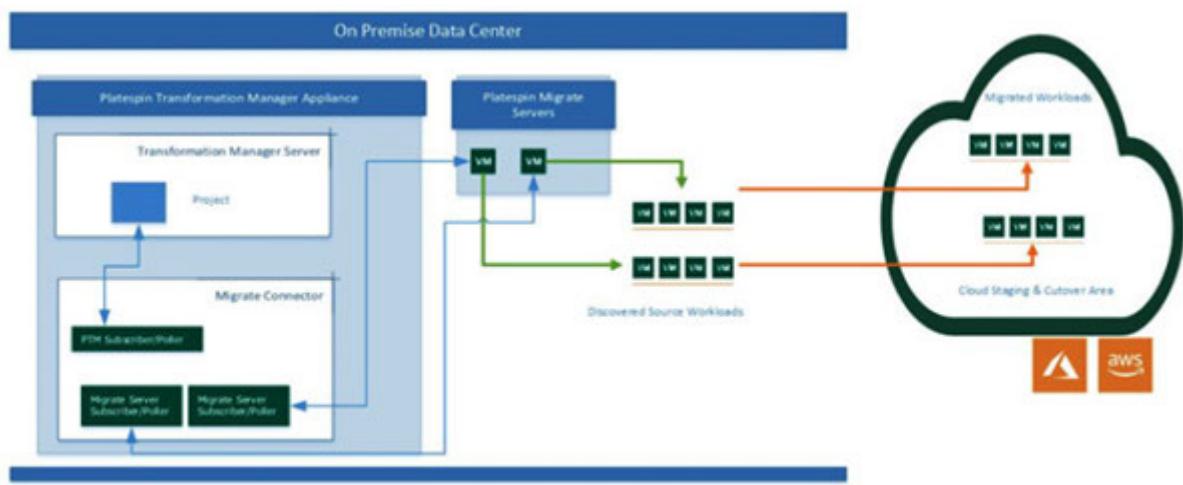


Figure 5.2: Platespin transformation manager architecture

Platespin transformation manager appliance consists of transformation manager server component and migrates connector component deployed on PTM server itself. There can be more migrate connector instances for different networks.

Project in transformation manager is a logical project created for workload discovery and migration. This is generally applicable for multi-tenants.

PTM subscriber fetches information from the PTM server and gives it to the PTM poller.

Migrate server subscriber and poller works in tandem and collects events and do the polling respectively to the migrate servers which interfaces with discovered workloads and then do the actual migration to the cloud or data center environment. Here, the cloud environment has been shown as a targeted estate.

AWS application discovery

Deploy and configure AWS discovery connector for agentless discovery. Following are the step by step process to deploy and configure AWS discovery connector:

Download the OVA connector from AWS specified link

<https://s3.us-west-2.amazonaws.com/aws.agentless.discovery.connector.bundle/latest/AWSDiscoveryConnector.ova>

Browse to <https://IPaddress> of the connector.

On the discover the connector setup page, add the following asked information

Enter VCenter Hostname or IP address of the VCenter appliance

Add VCenter username and password

Bypass SSL certificate

On the Configure AWS credentials page, add the following asked information

Enter IAM credentials username and password as bind with AWS IAM policy.

Move to the next page

Publish the data to the appropriate location i.e. local file or to AWS endpoint.

Initial and start data collection from collector's page

Data starts appearing approximately 15 min after starting the data collection. Data can be exported in .csv format using AWS CLI for agentless discovery. For agent-based discovery, data can be exported from Console as well as AWS CLI.

Discovered servers can be searched using filters.

AWS Application Discovery Service is integrated with AWS Migration Hub. Data can be managed via data collectors, servers, and applications within Migration Hub.

Azure migrate server assessment tool

The above tool assesses on-premise Microsoft Hyper-V virtual machines, VMWare virtual machines, and physical servers for transformation to Azure cloud platform or to **Azure VMWare Solution**

Azure migrate appliance is deployed in on-premise estate which collects real-time data of the servers (virtual or physical) with a performance history of one week.

Azure migrate appliance assessment provides below as information generally in form of CSV file:

Assesses if the workload is suitable for movement to Azure cloud or not.

It gives the sizing estimate of the VM, storage, and network.

Provides a monthly cost of the resource post-migration to Azure cloud.

Server assessment includes information such as:

Target location

Target storage disk

Sizing criteria

Performance history

Percentile utilization

Offer based on usage and uptime

Confidence rating of the VM assessed

Transformation tools

Below are some of the tools used along with assessment tools for transformation of the IT infrastructure from data center to cloud or the data center to data center or from cloud to cloud. In this section, the focus will be on data center to cloud transformation and accordingly all the architectures have been shown for the functioning of the tools and processes.

Following transformation tools shall be covered under this section

Platespin migrate tool

Google migrates services (earlier known as Velostrata)

Cloud endure

Azure migrate – server migration

AWS migrate – server migration services

Carbonite migrate

Red Hat infrastructure migration services

Platespin migrate tool

Platespin migrate tool is a Microfocus tool used for the following transformations.

Physical server consolidations, that is, conversions from physical to virtual machines

Migration from old hardware environment to new hardware environment.

Migrate of on-premise to cloud environment

The following diagram is the generic setup and architecture for the Platespin migrate transformation tool:

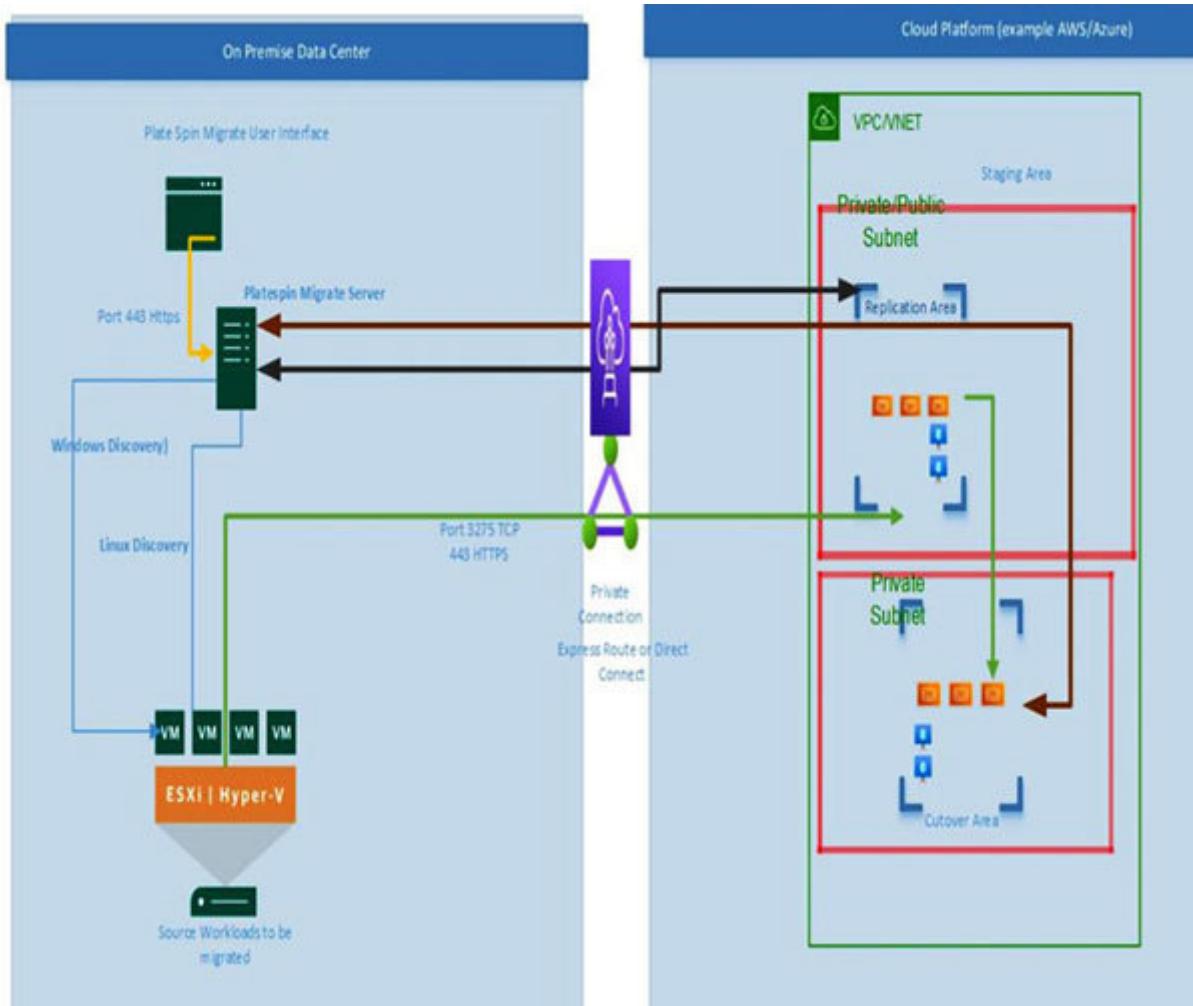


Figure 5.3: PlateSpin migration factory architecture

Also, PlateSpin migrate works along with PlateSpin transformation manager for automated migrations. Above architecture setup involves the following major components:

A user account with appropriate permissions which is needed to be connected from cloud platform.

A user account in Cloud (AWS/Azure for example) with designated roles for PlateSpin migrate which shall be used in setting up and configuring the PlateSpin migrate appliance.

Platespin migrate server (an appliance that needs to be downloaded as OVA appliance from Platespin website) and then provisioned in on-premise data center as a virtual appliance in VMWare Vcenter.

Once it is setup, it needs to be configured and connected to the cloud platform environment using IAM credentials depending upon the cloud provider (AWS or Microsoft Azure or some other provider such as VMWare Cloud).

The following example shows the setup and architecture when Platespin migrate is hosted in cloud environment:

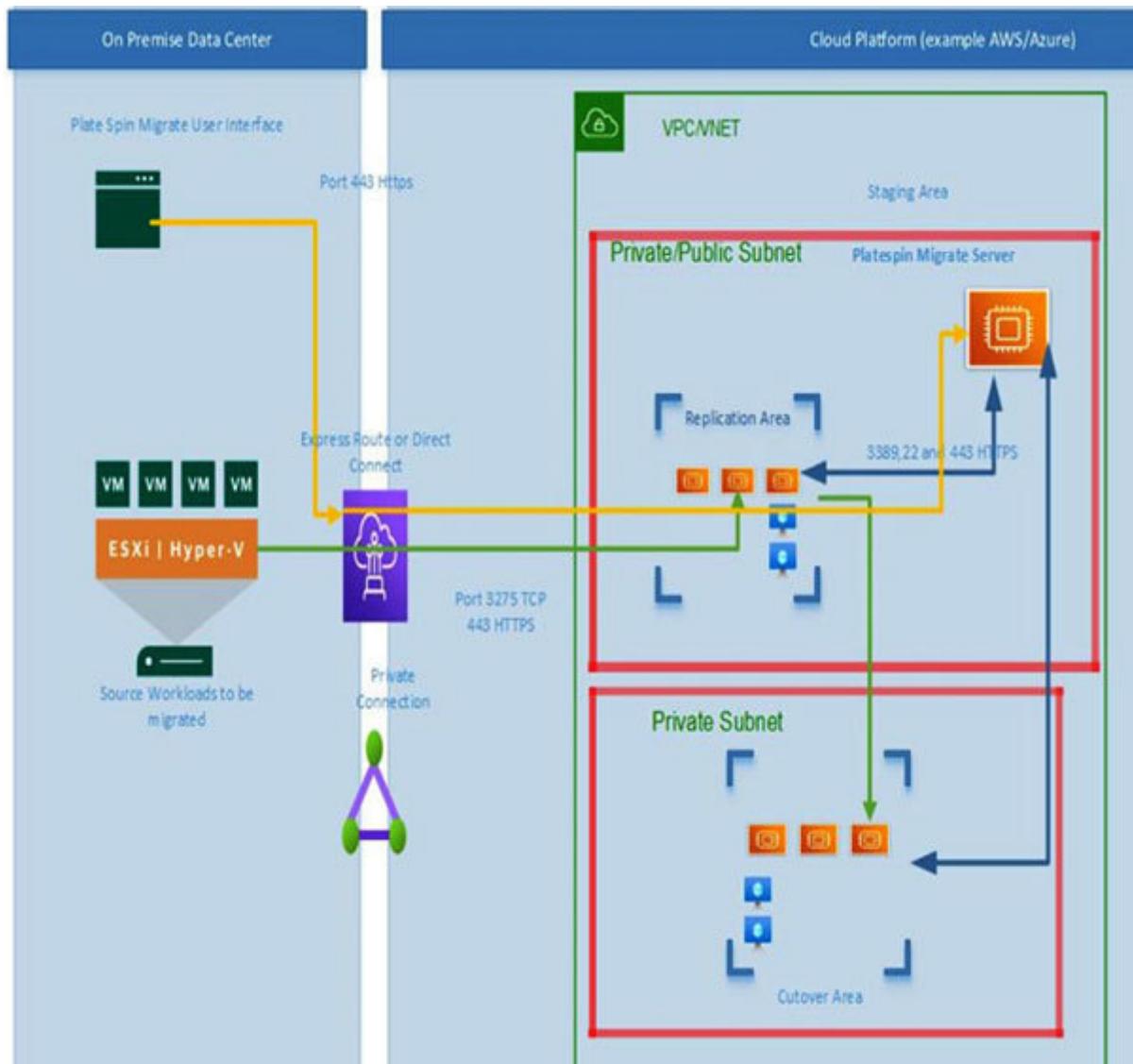


Figure 5.4: Platespin migration factory on architecture

Velostrata/Google migration services

The following architecture diagram for Google migrate service which is a tool to transform on-premise estate to Google cloud:

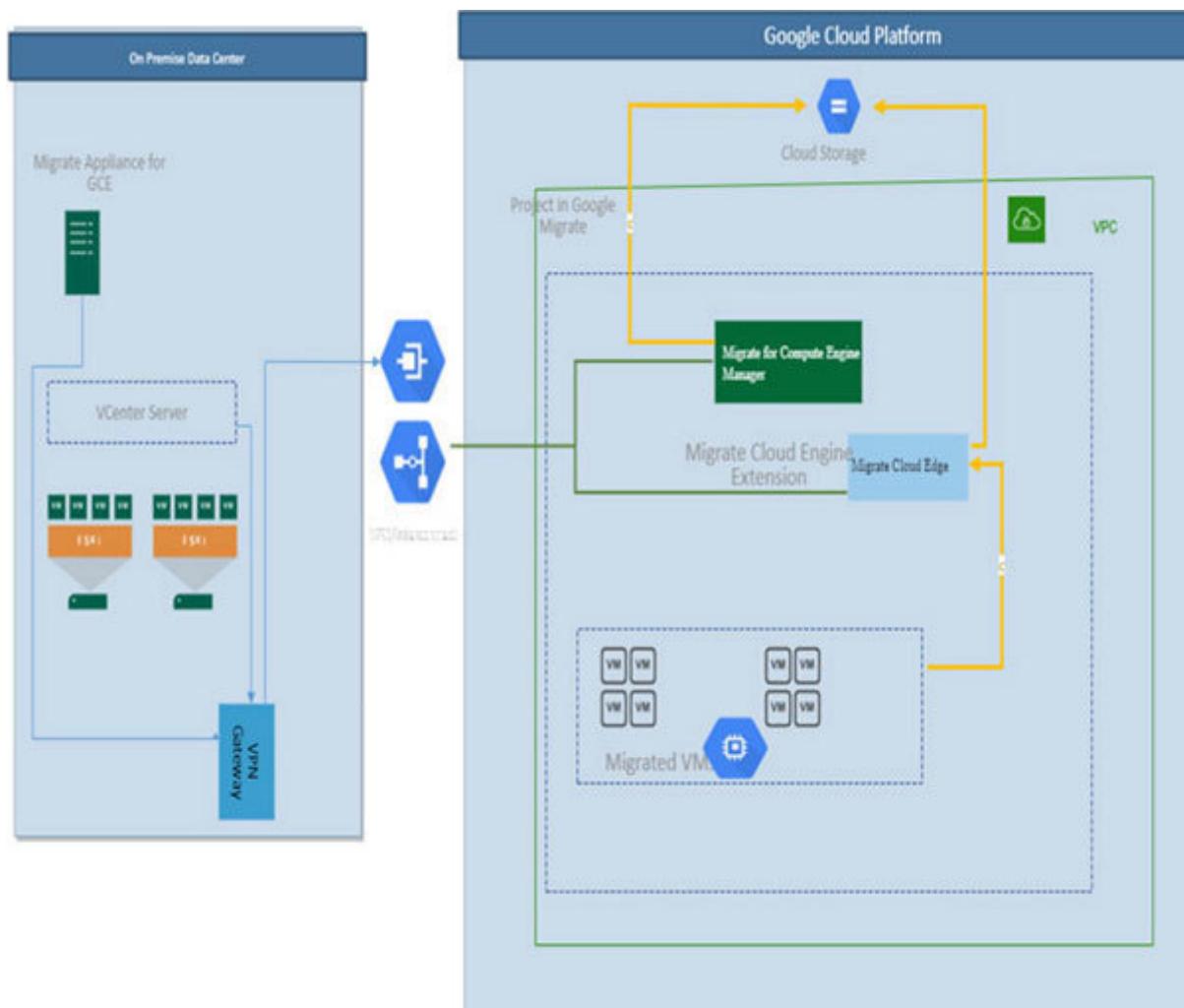


Figure 5.5: Google migrate service architecture

The following components and their role are mentioned in the preceding architecture diagram:

Migrate for Compute Engine manages all components and orchestrates migrations. It also serves the Migrate for Compute Engine UI.

Migrate Cloud Engine Handle storage migrations and serve data to migrated workloads during migration. Generally, a Cloud Extension is a pair of Cloud Edge nodes.

The Migrate for Compute Engine On-Premises virtual appliance/Cloud virtual appliance serves data from VMware or cloud storage to the cloud extension.

Connectivity between two entities, that is, data center and Google cloud are via VPN or via google cloud-based interconnect service.

Cloud endure

The following is high-level architecture diagram for cloud endure and how the transformation happens to the cloud platform, in this case, the target estate is AWS and source can be either data center or other cloud provider other than AWS such as Azure:

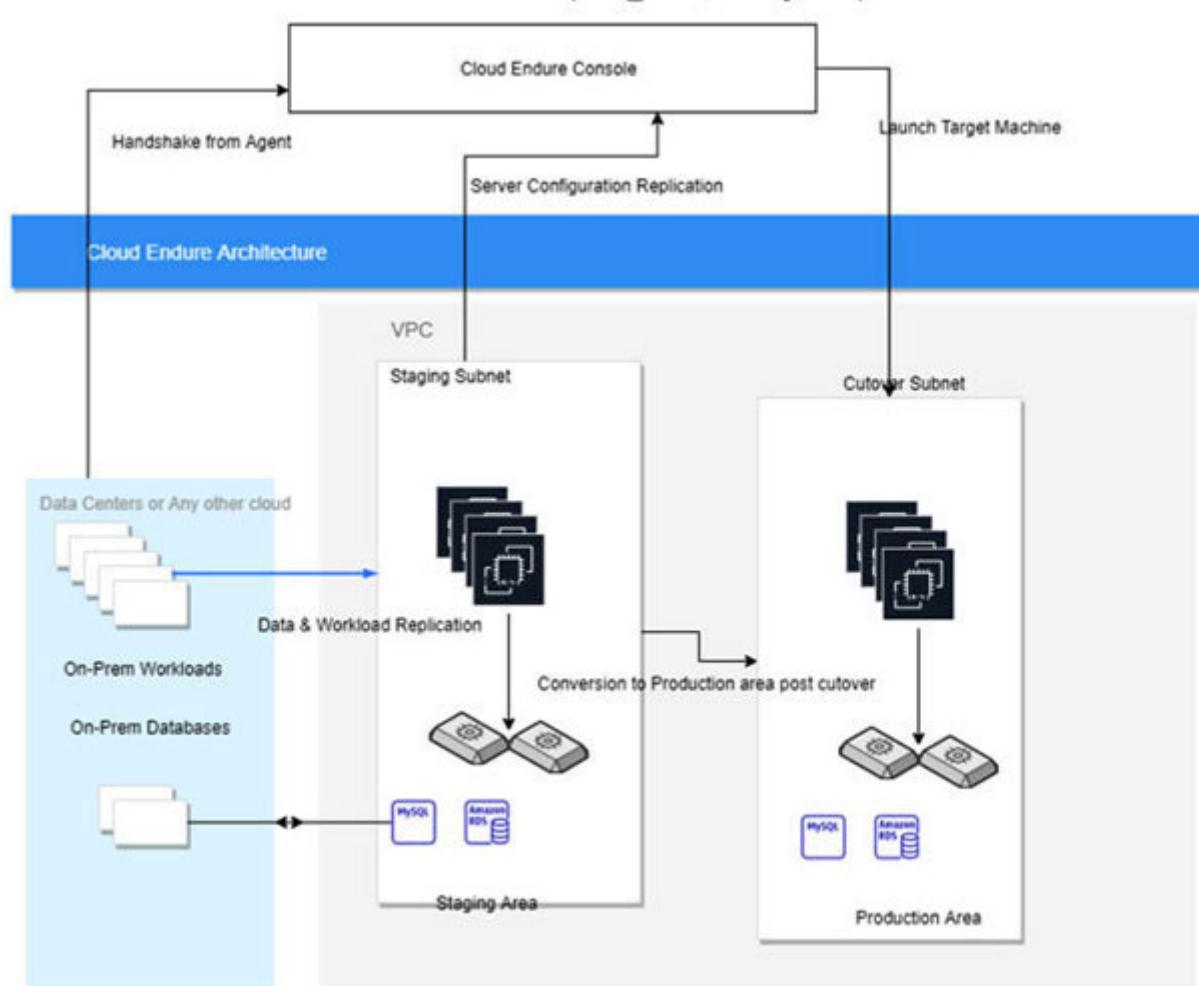


Figure 5.6: Cloud endure architecture

Cloud endure has the following flow for transformation:

Identify which workloads to be transformed based on the assessment done.

Post identification, cloud endure agents to be installed on the source workloads in on-premise data centers or any other cloud provider where workloads are hosted.

Create waves for the source workloads for grouping them in various stages of the transformation. By grouping, you make sure which workloads will be moved together as they will be having dependency on each other for example in the case of tiered application infrastructure.

As per agreed waves schedule, initiate replication of source workloads to the staging area in AWS cloud via cloud endure portal.

Provide UAT users access rights to test the application in staging area. Address any issues in the case reported by users from an application and infrastructure perspective.

Once users have tested the application functionality in the staging area, raise a change for the final cutover, which will move the application from staging to the cutover area that is, final production subnet in a **virtual private cloud**

Shutdown the source workloads.

Remove source workloads from the cloud endure portal.

Licensing of the cloud endure is workload based with an expiration date of migration licenses is 90 days. You need to install the agents on your source machines during this time frame and migrate your workload to the cloud.

AWS Server Migration Service

AWS Server Migration Service is an agentless service used to migrate on-premises workloads to AWS. You can automate, schedule, and track incremental replications of live server volumes in order to carry out large scale migrations.

Instead of SMS, cloud endure is now generally used in AWS for migrations of server workloads.

Difference between cloud endure and SMS is mentioned below:

Cloud Endure Migration is a block-level replication tool that simplifies the process of migrating applications from physical, virtual, and cloud-based servers to AWS.

AWS SMS is an agentless migration service to migrate on-premises virtual machines to AWS using virtual appliance:

appliance: appliance:

appliance: appliance: appliance:

appliance: appliance: appliance: appliance: appliance:
appliance: appliance: appliance: appliance:

appliance:

appliance:	appliance:	appliance:	appliance:	appliance:
appliance:	appliance:	appliance:	appliance:	appliance:
appliance:	appliance:	appliance:	appliance:	appliance:

Table 5.1: Differences between Cloud Endure and AWS SMS

Process to setup AWS SMS in VMWare Vcenter and replicating to the AWS environment:

An IAM user account for Server migration connector with **ServerMigrationServiceConsoleFullAccess** to ensure that IAM users have the permissions required to use AWS SMS.

Install the SMS connector to migrate VMs from On-premise data center to AWS as EC2 instances. Download the appropriate OVA file or VHD file from the AWS website and then deploy in a VMWare Vcenter or Microsoft Hyper-V/SCVMM respectively.

Setup the Vcenter service account with appropriate permissions. Vcenter Administrator has all the permissions/roles to setup the SMS connector along with AWS IAM user account.

Open the SMS connector VM console and login with the Service account and setup the static IP address on the connector.

Once configured, access the URL of the connector from browser using **http://IPaddress of the**

User Vcenter and AWS credentials created initially and bind the connector to AWS account.

The connector then shows up in the Connectors page.

Use AWS SMS console to replicate VMs from Vcenter to AWS using SMS connector hosted in Vcenter environment in On-premise data center.

Open the AWS SMS console at

Import server catalogue to reflect servers added in your VMware environment. This process can take up to a minute.

Select a server to replicate and choose **Create replication** Configure a replication job and then create and then start the replication to AWS.

Once replication is complete, delete the replication job.

For tiered applications, you can group the servers and migrate them together by creating an application entity in the AWS SMS console and then adding groups in the application.

The following diagram explains the architecture and flow of the AWS SMS functionality:

AWS Server Migration Service Architecture

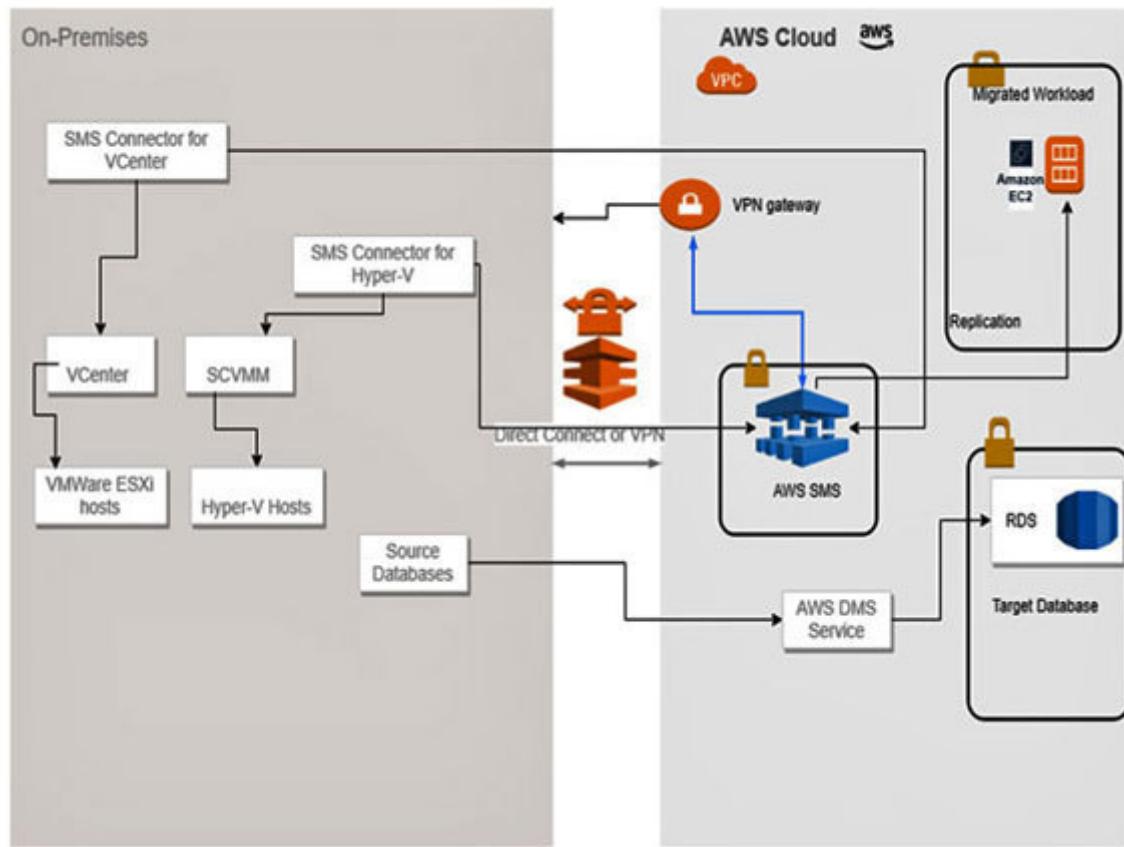


Figure 5.7: AWS SMS architecture

AWS Database Migration Service

AWS Database Migration Service is an AWS cloud service to migrate relational databases, data warehouses, NoSQL databases, and other types of data stores. You can use AWS DMS to migrate your on-premise data or cloud data into the AWS Cloud.

DMS is used with Schema Conversion tool in case the database type has to be converted from one DB to another say from SQL to Oracle or so on. Schema conversion tool helps you in providing a report which tells what needs to be changed from application code perspective post migration of the database (SQL) to cloud in the desired form (say Oracle). Post migration those actions need to be taken so that the database at target can be used by the users and the database is functional:

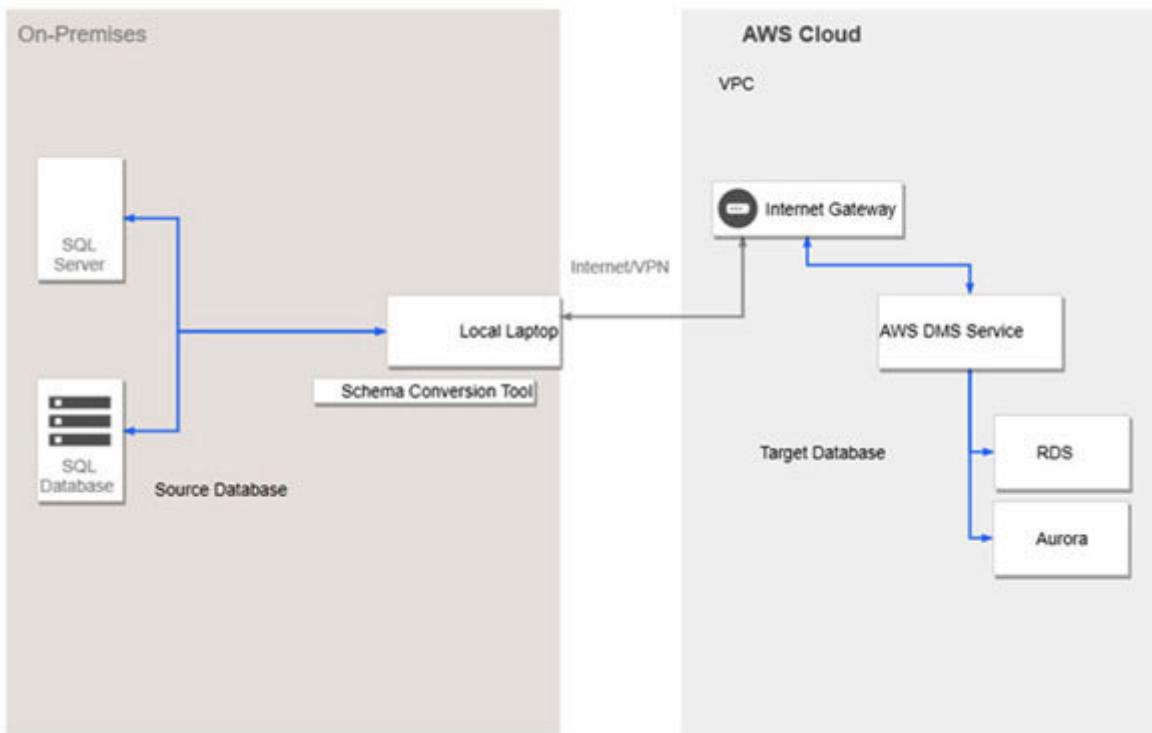


Figure 5.8: AWS DMS architecture

If the data is very high, then AWS products and services can be used:

AWS Snowball

AWS Snowball Edge

AWS Snow Mobile

AWS provides these hardware devices on which the organization can place their data in encrypted form in a secured way and give it to AWS that then import the same in customer's AWS S3

bucket or some EBS volume. This saves network data cost by the cloud provider and also it is faster.

Snowball Edge support capacity of 100 TB capacity as compared to Snowball which has storage capacity of 80 TB.

AWS Snowmobile is an Exabyte-scale data transfer service used to move extremely large amounts of data up-to 100 PB to AWS. Snowmobile is actually a 45-foot long container, pulled by a semi-trailer truck.

Azure Migrate Server Migration Service and Database Migration Service

Azure migrate Server Migration tool along with Azure DMS is another tool offering from Microsoft Azure which have similar architecture (not same) as other tools like AWS SMS or some other migration tool. There are few differences from other tools.

We will not go more into it as it will look repetitive.

Each tool discussed above is simple and specific, results-oriented, with the potential to revolutionize your work and for specific cloud providers. They are designed to help migration consultants, transformation engineers, and designers to successfully plan and transform the environment from source estate to target estate.

Conclusion

In this chapter, we covered the discovery and assessment tools for getting the inventory of the workloads running in source estate. Along with that, we also got a view of how application dependency happens with agent-based discovery. Once the assessment is completed, transformation tools were discussed such as Platespin, Cloud Endure, AWS SMS, and so on, which are utilized for transforming the source estate (such as On-premise) to target estate (public cloud).

In the next chapter, we will look at various benefits and associated documentation of a transformation project. Various risks, challenges along with the cost factor.

CHAPTER 6

The Transformations Project – Add-Ons

Introduction

In the last chapter, we deep dived into the tools and execution procedures to carry out transformation projects. We have gone through each and every tool for every cloud provider which are being extensively used Transformation project life cycle.

Considering the nature of transformation projects, there are a few add-ons that are deemed critical. We need to understand these add-ons that include a strategic roadmap, basis of the solution offering, associated benefits, and cost reduction mechanism.

Structure

In this chapter, we shall cover the following topics:

Engagement roadmap

Solution considerations

Benefits

Add-on-offerings

Cost-reduction – On the ops

Happy reading dear readers!!!

Objectives

Our main objective in this chapter is to learn about the add-ons of a transformation project's:

Engagement roadmap

Solution considerations

Benefits

Add-on-offerings

Cost-reduction – On the ops

Engagement roadmap

An engagement roadmap is a pictorial presentation of the strategic roadmap which depicts all the critical milestones for a transformation project. The main objective behind using an engagement roadmap is to provide clear strategy across the project along with its timelines and milestones.

An engagement roadmap is used to depict the:

Entire lifecycle

Various phases

Milestones

Timelines

Project modes such as transition/transformation or operations

Duration it takes to complete a single mode

Any parallel or overlapping modes or phases:

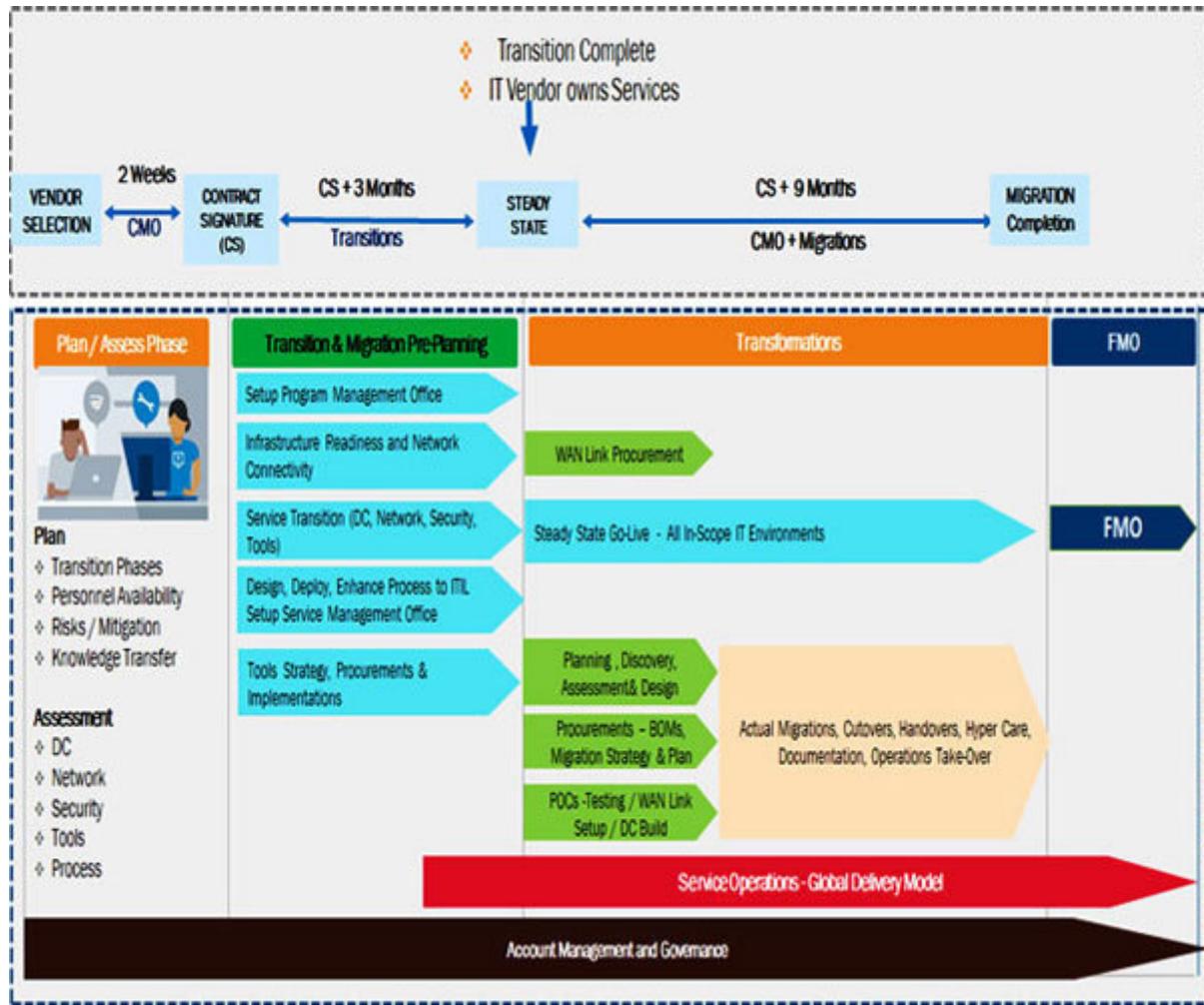


Figure 6.1: Engagement roadmap

Additionally, an engagement roadmap is also used to depict high-level activities list that is critical for the overall program such as contract signature, current mode of operations (CMO), and so on:

Contract Sign Post contract signing, selected IT vendor will commence the service transition across client landscape as the very first step:

Service provider would establish and transition governance office to drive the service transition

Analysis of gaps in existing client service delivery tools and process and initiate alignment of the same to industry standards

Transition of As-Is services for all the DCs/IT environments/business units/applications environments

Understanding the current delivery model and operating methodology with client IT organization

Process standardization and tools rationalization

Operability index: To measure the continuous improvements

Post completion of the overall transition, IT vendor would take over support and administration of all IT environments/existing DCs/Application support activities/vendor engagement:

Govern and provide support for all Infrastructure operations

Engage with any vendors

Identify and assess Infrastructure for transformation scope

Maintain agreed SLAs (Service Level Agreement)

Assess scope for continuous improvements across all the environments

Post contract signing, selected IT vendor will initiate transformation program. This will be a parallel activity to overall Transition program and will be executed by a different set of teams.

Capture information/artefacts/documents with respect to client information which are required from transformation's perspective

Initiate a complete discovery of all the IT environments/servers/services across the geographies

Install, configure and setup tools required to capture information and create relationship charts

Hold sessions/workshops/meetings with client SPOCs to get further clarity on the captured information

Hold sessions/workshops/meetings with the transition team to validate and review captured information

Prepare designs, BoM, build, test and implementation plans

Engage with the client leadership team, SPOCs, business owners, and any vendors

Future mode of operations As part of FMO operations, an IT vendor will perform the following tasks:

Identify elements of automation to be added

Innovation practices to be added

Simplification of tasks and activities

AI-based operations

Solution considerations

Every vendor vouching for client transformation project projects itself and proposes to be a very reliable strategic and collaborative partner to the client in its journey for growth. They would normally envisage a solution document with a future vision embedded into it.

However, every bit or component of the proposed solution is bound with some of the solution considerations which act as the base of the overall solution approach.

A client would normally provide RFP objectives and partial information about its IT environment. Based on the RFP objectives and deep insights of the environment data provided by the client, every vendor prepares deep analysis of the client environment. They will normally come up with the graphical representation of the client IT environment.

IT vendor will construct a solution based on few specific considerations which in turn are mapped with each of the service objectives as are defined by the client in the RFP document.

As an example, following table summarizes the key service objectives, implications for an IT vendor as a partner and the key solution tenets that form the basis for the solution and delivery model:

Business Priorities	Technology Priorities	IT Vendor Solution Considerations
Thought Leadership	Modernization: <ul style="list-style-type: none"> Scalable and repeatable architecture Address significant levels of technical debt Legacy Infrastructure modernization 	Modernization: <ul style="list-style-type: none"> Consolidation of nominated DCs [M1] Build & Migrate Workloads – Agile, high performance & scalable Infrastructure [M2] Customized migration plan for each of Hosted environments [M3] Nominated landscape migrations [M4]
Higher expectations on IT	Service Management and Improvement & Automation: <ul style="list-style-type: none"> Reduce frequency and severity incidents Reduce lead times for provisioning of new infrastructure to enable quicker delivery Increase visibility and control 	Service Improvement & Automation: <ul style="list-style-type: none"> Integrated managed services enabling client to focus on business [S1] SLA driven and ITIL aligned services for better customer experience [S2] Integrated command centre ensuring high service availability [S3] AI based automation solution [S4] Predictive, Perfactive, prescriptive and preventive analytics [S5] Reliable & Secured Operations [S6] Customer Satisfaction [S7]
Asset Reuse	Reduce Risk and optimization: <ul style="list-style-type: none"> Minimal disruption with current Client SLA's Secure and compliant management of solutions and data Risk mitigation plan 	Risk Reduction: <ul style="list-style-type: none"> Mature program management approach [R1] Risk mitigated transition plan [R2] Client Solution based and geo-based migration strategy [R3] Event Co-relation Engine & End-To-End Automation Engine to be enabled for risk reduction and Highly Available Environments [R4] Focused approach on reducing technical debt [R5] Seamless integration of Migrations and operations team [R6]
Buying Power	Cost Reduction: <ul style="list-style-type: none"> End-IT ownership Technical Debt reduction and efficiency improvements to reduce overall cost base Reduce costs of support by removing the 'encumbrance' of legacy systems 	Commercial Innovation: <ul style="list-style-type: none"> Build an agile and efficiency based Infrastructure thereby reducing overall consumption cost [C1] Reduction in Physical Footprint [C2] Automation and Innovation approach & projects to further enable cost reductions [C3] Unit based pricing model [C4] Workloads Optimization – Subscriptions Cost Reductions [C5]
		M - Modernization S – Service Improvement R – Risk Reduction C – Commercial Innovation

Figure 6.2: Solution considerations mapping

Benefits

An IT transformation project is a game changer in every possible way for an organization. An Infrastructure Technology Transformation project would ensure complete assessment or rather reassessment of the entire galaxy of IT operations running in any particular organization. Anything and everything which is part of IT Infrastructure is dependent on it and running with the help of Information Technology would be impacted by an IT transformation project.

IT transformation projects are termed as monumental due to their sheer size, no. of assets they cover, duration which take for completion and the amount of years for which they are forecasted to keep the environment steady and functional. Such a mammoth and humongous project would ensure that all the systems pertaining to infrastructure technology go through a complete and much needed overhaul. Normally a transformation project would impact and entail changes to all the core sectors of the IT world which would comprise of:

Hardware in use pertaining to all the tracks

Reference architecture(s)

Accessibility and authorization mechanism

Data storage and archival

Clients/vendors/user base

Software distribution

Licensing and volumetrics

Service level agreements

Support services:

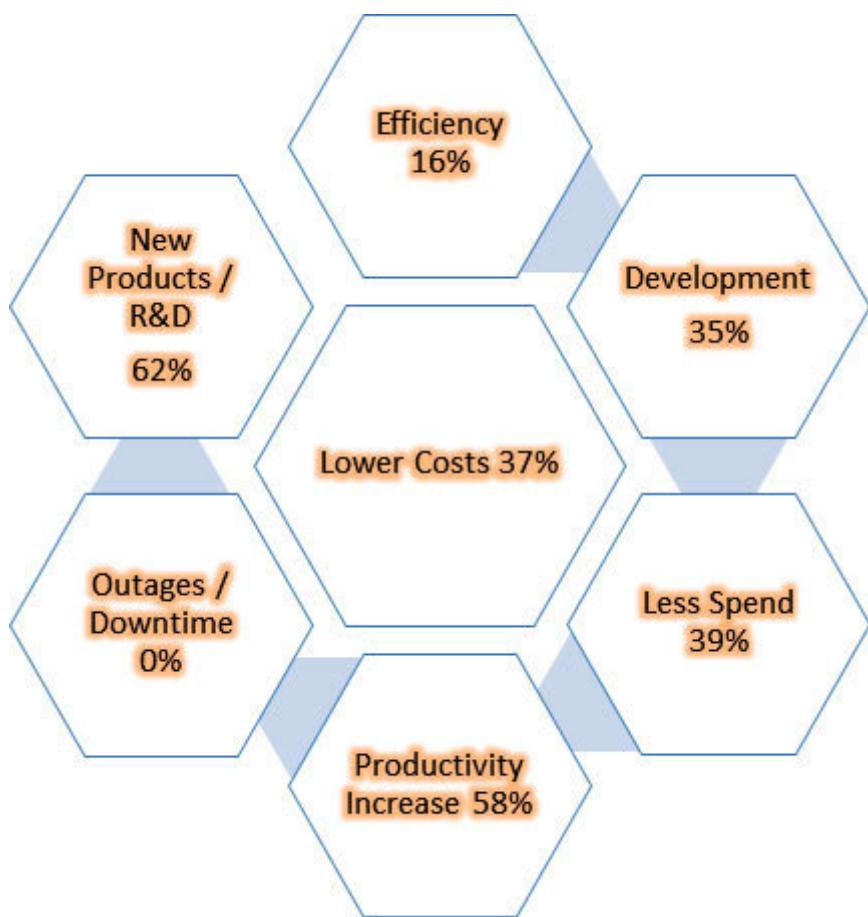


Figure 6.3: Transformation benefits

Transformation project as we know would simply target for the modernization of all core services to say the least. That is the reason, an IT transformation is sometimes referred to as a "Rip and Replace" project as well.

Let us go through some of the benefits of an IT transformation project:

Business – IT One of the most sought out aspect and benefits of an IT transformation project is better business and IT alignment. Whenever an IT transformation project is pursued, current and future business requirements are captured with respect to all the existing applications and the ones which are expected to be hosted in the near future.

Once these requirements are captured, thereafter, IT platforms are designed and hosted in such a way that they meet and match all the specified requirements. Not only this, they would also contain a lot of buffer content (in terms of servers/storage/licenses) for any potential up-scaling of any of the environments.

This way, business houses are assured of underlying IT platform during any emergency spikes of load or to cater any additional requirements. This sort of alignment is very critical in terms of stabilizing the overall business outcome.

One of the latest trends in the IT industry is to bring in innovations. Today's IT world is changing at a dynamic pace and terms like innovation or automation have become a common term with the IT platforms. It is always the question of the hour as to how the innovations will come, what will be quotient and in what form and quantity.

However, innovations do not come or happen into old or lethargic DCs. To accommodate Innovation projects, you need a datacentre that has enough capacity and capability to support all the running existing operations and then also accommodate innovation or automation or research or development projects through the spare or additional capacity. Such capabilities are not found in an old datacenter that is still running old technologies with old hardware and having frequent issues with respect to outages and capacity.

A transformation project gives much needed breather and infuses a new lease of life into the hosted IT environments. It gives all the reprieve, resilience and adds stacks of new repositories which are provisioned with a vision through a strategic roadmap of at least next five years. This paves the way for specific projects like innovation and automation which are generally hosted on separate environments (other than operations hosting environments).

Reduced risk/risk-free An old datacenter running any environment is like a car being driven with no one at the steering wheel. It can go to any direction at any point of time and there is nothing which anyone can do. An outdated DC is a bag full of risks as it would contain:

Loose and unstructured cabling

Old and outdated hardware pieces

Legacy pieces of infrastructure and application components and versions

Non-redundant hardware elements

Partially or unsecured environments – vulnerable to hacks

Improper cooling/UPS arrangements

Unidentified/no owners specified type of Cls

Uncontrolled access

Improper monitoring

No vendor support/no AMCs

Non-compliant workloads

Repetition of tasks/unresolved critical incidents-alerts

No provision/capability of growth

Any many more. Due to the presence of many of these, an old DC always presents a very high percentage of risks that may arise at any point of time during the functioning.

However, post-transformation, an upgraded DC or a new DC would be able to mitigate most or almost all of these risk factors and would be able to cope up with any unexpected challenges. With the amalgamation of new and up to-date devices for every track, risk percentage takes a dive and becomes negligible.

Minimal An upgraded and transformed datacenter will ensure that any sort of outages related to underlying Infrastructure is completely ruled out or are at their minimum. In fact, reduced risk and minimal outages are one of the major factor which drives and fuels a transformation project.

Outages means business down that means the loss in the revenue or overall earnings. Any outage has the potential to cost a company a huge fortune that may have a long-lasting and a highly adverse impact on the overall outlook as well. Such an outage also brings a lot of bad reputation for the organization and it only makes it more difficult to negotiate and bring new business opportunities to its shores.

Thus, organizations with strong IT presence tend to go for overall IT transformation to stay abreast with new technologies and avoid any sort of business outages.

Reduced operational It is no secret that a transformed datacenter is highly cost-effective than an old one. While it may cost an organization a good fortune to upgrade and transform its datacenters but it is not at all a costly affair to manage a transformed one. One of the major benefits of a transformation is have reduced cost of operation.

An old DC normally requires a lot of effort in terms of management and maintenance with respect to managing hardware devices/DC components/Warranties – AMCs. However, once a DC is transformed and is running up to current industry levels, its management and maintenance costs swindle down to the bottom.

Today's IT devices are embedded with energy efficient features and they are easily able to scale themselves up during requirements and then shut themselves down when the requirements or the peak load goes down. Overall, it is estimated that a transformed DC normally is more than 40% more cost-effective than an old one.

Agility to host versatile A transformed DC will provide you with an option to host complex and versatile solutions and environments which is generally not the case with an old DC. A new and highly efficient hardware will be able to hold and cope up with any new, complex, and resource hungry environments with just a few slight modifications or configurations tweaks.

With the advent of virtualization, cloud platforms and new emerging DevOps technologies, any sort of massive, huge, or mega environments with requirements reaching up to the limits of

sky are very much in reach and are easily configurable and manageable.

With IT becoming the backbone of almost every other sector, large and very complex environments with very highly loaded application stacks are also conceived on IT platforms. Environments such as high-rate media streaming across the globe or any highly popular mobile-based application are an example of how agile an IT environment can be once it is transformed and loaded with newer set of devices and software(s).

Application delivery speed/development Development work pertains to the application delivery speed and x development which involves or makes use of databases, tools, and underlying infrastructure. Development work is mainly a research work wherein developers would get a couple of servers, load them with specified application modules, tools, databases, and other prerequisites that are to be met.

They would normally demand the amount of resources that are required to be loaded on each of the workloads which they are going to use. Sometimes, these resources are very high in demand, for example, a SAP application testing module may require up to 64 GB RAM per server with about 8 GBPS IOPS as an output capacity from the underlying storage box.

And even this is not enough, a developer may end up crashing any workload any point of time and may come running to Infrastructure teams for another server. With an old DC, such liberties are simply not available. An old DC will not be able cater

such resource requirements or will not be able to even provide additional workloads in the first place. Lastly, they may not be always readily available to provision a new server immediately.

However, post-transformation, a DC will have enough resources to cater any kind of development work. Also, the process of provisioning a new server can be automated using various infrastructure or DevOps tools such as Terraform, Ansible, Kubernetes, or vCAC. This will ensure that no work is hampered and interrupted and normal routine operations will continue to flow.

Operational efficiency/productivity A newer or a refreshed system will not only dispense exceedingly service but will also ensure that it fetches the same from the users as well. It is a scenario wherein users do not have to wait for the systems to be up and running or be available for the work to commence. There is simply no wastage of time as there is no waiting time as the overall operational efficiency is enhanced considerably.

Due to newer and transformed systems in place, hosted environments, all the transactions and transmissions are processed with in a million of micro-second in the backend. This basically appears as if things are working in a flash which eventually gets the maximum out of the users as well.

Highly enabled Post-transformation, as the operational efficiency is enhanced, all the systems become much more efficient with ultra-rapid processing of transactions, it puts the overall delivery of the project deliverables into high gear. Prior to transformation, tasks

that used to consume a long time, are now processed within seconds post-transformation. Here are a few examples:

Application modules

Infrastructure platforms

Databases

Defect detections

Remediation's

All transactions

Network transmissions

Data feeding

Data fetching processes

Data upload

Data read rate

Data access rate

The overall process of working on a system for all the associates and executives becomes much faster and it helps with the deliverables. Due to system readiness and rapid response, project is able to execute, process and deliver all the required deliverables within the specified timeframes.

Add-on offerings

There are lots of security and computing risks to be considered during IT transformation from one estate to another, say from data center to cloud. Security risks such as data breaches, compromised credentials, authentication issues, account hijacking are the biggest threats. These are challenges and risks which need to be mitigated for overall transformation and migration. Cloud providers do offer the below services to curb these challenges:

Key management service

Secrets management/vaults

Repositories

Encryption techniques, data encryption, and safety

Identity management and access control with multi-factor authentication

Privileged access management solutions

Firewalls and web application firewalls

Certificate authority service

Micro services models and tools

HIPPA and other PCI DSS certification models

Log auditing

Bastion services and host-based services.

Dedicated compute models instead of shared physical host models

Vulnerability scanners

Thread detection solutions

The main purpose behind these offerings is to mitigate any type of risk and overcome all the challenges.

It is important to understand that each region of the planet comes with a different set of rules, regulations and federal laws for compliance.

For example, entire European countries follow **General Data Protection Regulation** compliance which is actually a regulation in European Law on Data protection and privacy in European Union and the **European Economic Area**. Same GDPR regulations also address the transfer of personal data outside the European and EEA areas.

Thus, if a specific migration or a transformation project has the requirement to move the data in or out of any European country, then it has to be first subjected to the GDPR rules and regulations or it will be a breach of law and will levy very heavy penalties for the processing organization.

Cost-reduction – on the ops

It is a no secret that one of the purposes of transformation from one estate to other estate is Cost reduction and high **return on investment**. Both these terms go hand in hand with **total cost of ownership**. Organization try and save cost through various means and mechanisms.

It does not have to be a very extensive plan and mechanism to save cost. A substantial cost can be saved using smaller chunks of efforts and strategies.

Following are the best practices to reduce your overall operations end estate bill:

Unused Every client will probably have resources that are being lying idle and are billed for nothing. As a service provider and administrator of the overall IT environment, it is imperative to find the resources which are not in use or are not consumed in the operations cycle. Such resources when found, are first checked for historical information as to from how long they are lying idle and is there going to be any future that may arise for them. Once confirmed, such resources are either deleted/decommissioned or are archived.

A great example for the same would be the servers running in the environment but not being used or it can be the unused

storage or data, and so on.

Optimum At times, large and heavy instances are demanded by applications administrators or by the developers. However, most of the times, such heavily loaded VMs stay underutilized. However, they continue to occupy large set of resources which eventually makes a loading on the costing bill.

It is mandatory for the vendor managing the IT environment, to keep such instances under round the clock monitoring and change the specifications/size of the VM or the basic type itself of the VM if it is underutilized. Same approach should be applied for all such underutilized VMs.

Any VM should be given higher set of resources only if it is able to justify the demand.

There are a lot of discounted offers that are offered by various cloud platform vendors and at times, these offers are in abundance. These offers contain VM instances such as reserved instances and spot instance.

These type of instances are less expensive and should be utilized for any non-production or non-critical type of activities or environments where-in the utilization is not that much and the requirement is temporary.

Auto-scaling: Auto-scaling consists of two different types of scaling. One of them is vertical scaling which is resource addition or

trimming. And the second type of scaling is horizontal scaling which is basically the instance addition/removal.

Almost all the cloud platforms offers auto-scaling feature which is basically the capability of the cloud platform to scale-down or scale-up the Instances or resources as per the requirements and demand.

This scaling up (resources addition) and scale-down (resource trimming) happens automatically and requires no manual intervention. This feature can be used to add/remove additional instances according to usage which helps to save a lot of cost during off-peak seasons.

Appropriate An experienced administrator will always choose the most appropriate service as per the requirements. For example, if any application only needs a very basic and standard kind of IOPS or throughput requirements, then the administrator will not choose any premium tier or the hardware (compute/storage, and so on). Any standard set of options like compute or a standard storage disk options available in cloud will suffice here.

Another example would be to use a cold storage or an archival solution if the levels of the data are very inconsistent.

Appropriate There are a lot of tools available and are provided by cloud platform as embedded or native tools which help to make precise recommendations. Tools such as cloud advisor tool help to prepare and identify recommendations.

Choosing the right add-on Appropriate tools: It is worth mentioning that it is imperative to choose the right add-on service. As is evident, all **Cloud Service Providers** would offer *Pay-As-You-Go* pricing model and every component or service is chargeable. Thus, it becomes very important to select and choose the right add-on service for your environment. Unnecessary addition of services will surely result in the increase of cost and charges from the providers.

Conclusion

This marks the conclusion of this chapter on transformation – add-ons. We have had a nice look at the overall engagement roadmap and understood as to what it means and why it is used. Then we had a look at solution considerations and as to how they are mapped with client objectives. What followed this was the benefits, add-on offerings and a few cost-reduction mechanisms.

This marks the end of all the topics on transformations and we will proceed with migrations topic from the next chapter.

CHAPTER 7

The Migrations – Introduction and Lets Usher

Introduction

In this chapter, we will read and understand migrations in general and what are IT Migrations. We will also cover the impact of migrations on an organization and what it means. We will also understand the basic principles and sole purpose of a migrations project. We will then read about migration strategies and associated parameters and their benefits.

Happy reading dear readers!!!

Structure

In this chapter, we will cover the following topics:

Generic migrations

IT migrations

Migrations impact

Migrations purpose

Migrations principles

Migrations drivers

Migrations strategy

Strategic parameters

Benefits

Key terms

Conclusion

Objectives

Our main objective in this chapter is to learn about:

Information technology migrations

Basic principles of IT migrations

Associated purpose

Strategic approach

Strategic parameters

Benefits of a migrations project

Generic world

Migrations means movement. It is a very dynamic term in itself. The movement of a living being in the form of a human or animal from one place to another is known as migrations. It may also be defined in a way that it is a gesture of moving away from someone's current location or situation to a better or a new place in order to achieve something new or change base overall. The literal meaning of **Migration** is *shifting of people or an individual or group of individuals from one cultural area to another, which may be permanent or*

It is an act or an instance of moving from one country, region to settle in another. In simple words, migration means moving from one place to another with intention of settling permanently or temporary:



Figure 7.1: Migrations

Migrations have always been around as civilizations have evolved on our planet over the course of thousands of years. Herds, communities, people, groups, and even animals have migrated from one place to another. Well, there may be a lot of reasons for a migration but the most common one is desire for a better life. A hope which promises a better life and well-being and at times to end the current misery, need for relocation arrives. Historical evidence across the centuries have also highlighted the fact that masses have moved in an aspiration to be able to better themselves. At times, ambitions of finding a new nest or a breeding ground have also motivated a migration.

Migration promises that alluring green pasture for which everyone has a craving. It promises that yearning that drives the hunger within a soul to move.

It is that sheer nature of migrations that still keeps it intact and is still pursued across all kinds of life forms. And not just across generic terms, migrations are adopted with in IT world as well due to its traits and dynamic nature. We will discuss and understand this in details in upcoming topics as a part of this chapter.

Information technology migrations

Migrations are right at the heart of Information Technology world. They are termed as huge and humongous tasks. In IT world, Migrations are always associated as a synonym with the word

In information technology, migrations is the manoeuvre that is undertaken to process the movement of an IT landscape into a new environment. It is the wholesale movement of an operating environment towards a different operating environment. This movement is perceived as the new operating environment is always believed to be a better one for overall functioning and is likely to pay better returns and dividends along with saving existing incurring costs.

Migrations consists of full-scale project life cycle and comprises of many phases such as:

Assessment

Planning

Testing

Execution

There might be a few sub-phases as well depending on the characteristics of the overall migrations project. Each of these phases is dependent on each other. There may be partial start to a new phase but it cannot get into full steam until preceding phase completes successfully. Thus, it makes very critical for executing project teams to pitch in focused and dedicated approach to the migrations project they are working on.

During migrations project, all the sections of the environment go through a lot of steps/phases such as:

Assess

Plan

Design

Test

Validate

Publish

Approve

Document

Report

Migrate

Manage

Operate

Migrations are generally carried out via a tool that is being used to actually move the components from the current habitat to the new territory. This tool is decided upon by both client and vendor teams and is deemed fit only proper testing and subsequent positive results. Migrations generally involve following mentioned rules when being planned:

Ripping apart

Retain

Replace

Re-define

Rectify

Re-arrange

Re-architecture and off course

Move:

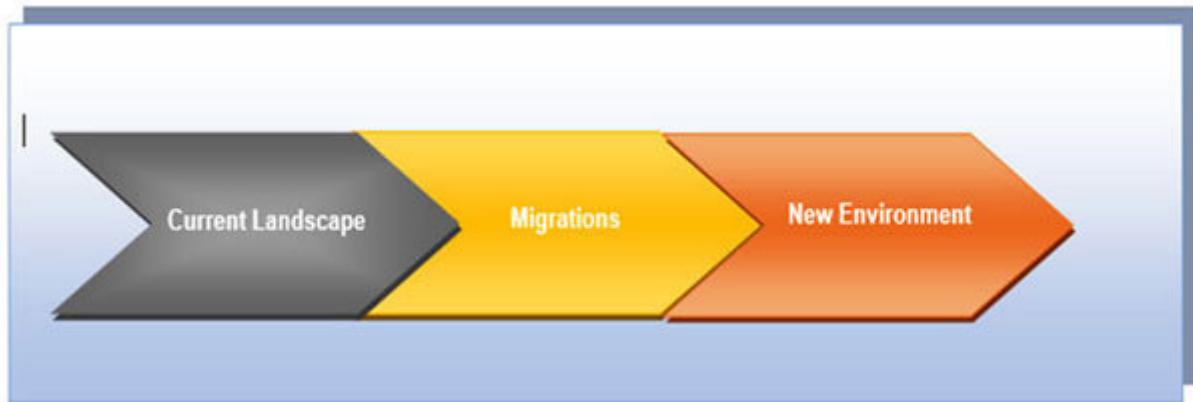


Figure 7.2: Migrations landscape

Migrations are difficult and can be very complex as well. They involve moving or upgrading the entire or the selected IT environment to a new one which may involve moving onto a new hardware platform all together or upgradation in terms of software or the application. The migration process ensures that the movement takes place seamlessly from the old environment to the new one. It makes use of all the features and the assets which can be used from the old environment and all such items are at the disposal of migrating team. However, all the migrations are pursued to gain better values.

Migrations can be small and very long as well and at times, they can prove to be an extremely lengthy and exhaustive processes. It may even take a few years for some specific migrations to finish. Reason being, migrations impacts almost all the components of an existing or current IT environment. It ensures that everything which is in use in the current environment is being assessed and planned. Migrations of any magnitude demands proper planning

and assessment. These are the two main pillars that form the base of actual migrations. Post assessment, an organization comes face to face with the:

Actual challenges

Depth of the environment

Real stakeholders to deal with

Dimension of the earmarked applications

Overall client portfolio

Realization in terms of hardware and software

Actual cost

Approximate duration

Effort estimation

Migrations impact

Migrations leave a long-lasting impact on everything they are applied on and everyone who gets to work on such a project.

When migrations are planned, each and every component of the current environment gets on to the impact list. This means that every single entity associated with the current environment gets ready to go through a very thorough assessment process that implies that whole environment is going to be impacted.

The project team working on the migrations project has to work and perform a very careful and detailed examination of all residing constituents or modules of the current environment one by one. Their motive is to find out about the:

Role of performing component

Its current age

It impacts in the overall cycle

Its current version

Compatibility chart for migration

Candidate for upgrade/retain/replace/remove

Support cycle availability

Inter-dependencies with other platforms

Local/global accessibility

Adaptability with proposed solution/platform

This detailed examination is known as the assessment phase and it is done right at the start of the migrations project. Assessments are must for every migrations project prior to making or planning a movement of any sort. Assessments output provides a lot of data that goes through a lot of computations and arithmetic to finalize inputs for the next phase.

It is important to note the impact which is made by all the sections and components of the current environment. All the associated ingredients associate with each other in some which is known as dependencies. Thus, even if a single component is being missed, overlooked, or ignored during assessments, it has the capacity to derail the entire migrations project. For a migrations project to be successful, it is imperative that all the pieces and elements of the current environment are in line with the proposed platform. It is also mandatory to perform test migrations with the selected tool or methodology before actual migrations are planned or are scheduled.

Migrations always have huge cost factor associated with them. Each and every phase of the entire project life cycle is associated with a cost tag and which generally includes a variation of up to 10%. This provides a complete cost forecast readily available with the project leads or managers and they are very well aware that they have to conclude the project phase within designated timelines else there will be a huge cost impact. An interesting fact about cost escalation is that cost escalations are minimal initially but they really sum up later on. For example, it might just be 5% or 10% cost escalation in year 1, but it may shoot up to 25% or so in Year 5 which is a very significant number in itself and a major impact on the overall project:



Figure 7.3: Migrations cost graph

Cost also covers two more main aspects that are:

Expenditures

Procurements

Expenditures cover tasks related to spending for project resources (human) such as salaries or allowances for project team/logistics like laptops, and so on. It will also cover travel costs as well. Project leads/managers will have to bear all the costs related to the project team as per organization policies. Also, the project will have to arrange for laptops/desktops, and so on, for the project team to be functional. Additionally, if it is required for a few project team members to travel to client team, all this cost is also accommodated by the project team only.

Additionally, all the procurements, that is, hardware or software, are also part of the project cost management plan and are initially handled by the client organization.

With respect to all the cost and associated spending, any magnitude of deviations in any form will make a startling impact on the project and the budgets. For every migration phase, task, activity, or procurement that does not go as per plan/not successful/delayed, impact is realized and calculated in terms of duration, effort and cost:

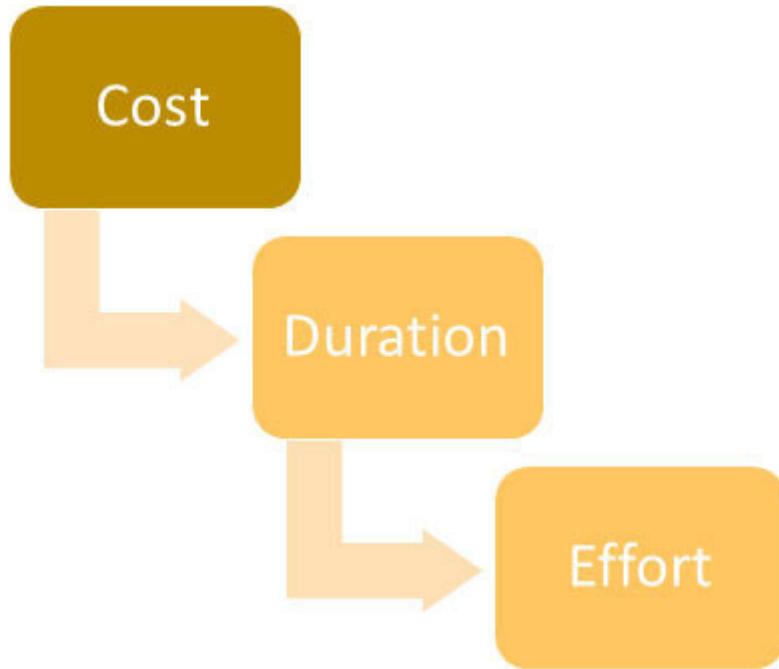


Figure 7.4: Migrations cost impact

Such deviations from the scheduled cost spending plan will send all the leadership into a frenzy and will impose a daunting impact on the overall budgets of the project.

Purpose of migration

Migrations is the process of moving current IT environments to a newer infrastructure or platform with the purpose of upgrading the current Infrastructure and yielding better business returns and value. Migrations are visualized and then realized with concrete aims and goals which are actually the laying foundation of any migration project. Organizations come to terms with migrations and exercise the process with pre-planned goals and definite purpose to achieve. There are a number of reasons behind a migration project. Let us have a look at all of them in detail:



Figure 7.5: Migrations purpose

With migration, organizations target high performance. This means that making systems, resources, and platforms more and more efficient and fetching the maximum out of them. It also means that moving onto better or newer platforms to run the business. Newer platforms generally imply transferring applications on to new set of systems which are termed as the best in the industry. With newer platforms, it is widely expected that all error-prone issues and sluggishness will walk away and performing applications will have a new room of available resources. It is also expected that newer platforms will provide newer versions of Infrastructure of support thereby maximizing the overall performance bar by a few notches.

Security has become a very main aspect in today's infrastructure technology world. It is one of the main ingredient of a platform. Organizations target environments with more and more security layers and protocols due to the possibility of:

External attacks

Hackers

Human errors

Phishing sites

Sensitive data leaks

Organizations now a days opt for systems which are:

Encryption enabled

Multiple layers of security protocols

Restricted access

Authentication mechanism

Latest technology enabled

Up to date devices and tools

Vendor supported

24x7 monitoring enabled

Daily status updates

Reports and dashboards

Moving on to such a system will also help in terms of getting approvals from clients as well as everyone tends to look for a safe and secure system in place.

Scalability is the ability of a system or platform to scale itself UP when the demand or usage is high and then scale itself down

when the usage is very thin. Scalability is a feature that has come into play very recently specially with the introduction of cloud platforms. This feature provides flexibility to underlying Infrastructure platform to stretch itself with the addition of more hardware/software servers when the application usage hits its peak. And then the same platform has to the ability to shut off some of its components when there is a very lean demand. Scalability is directly related to the production and efficiency terms and an increase in scalability means increased production.

This feature specially helps to save a lot of costs since the systems shut themselves off when they are not being used.

Today's IT world is a world of hosting multiple applications on the same platform. This marks an indispensable need for platforms that can be easily tweaked to let various business applications function through. An unstable system will not be able to go through much of change and is likely to go offline at some point of time. This is where the need arises for robust systems.

A robust system is able to handle and host cluster of applications with ease. The base Infrastructure is efficient enough and is tweaked in a way that it is able to grasp numerous transactions at any given point of time and still provides the required stability. Thus robustness is one of the major goal during a migrations project.

System agility is a major purpose and it means that system has the liveliness to go through required changes and is ready and prompt at the same time. Agility takes support from both

scalability and robustness as it may be required to accommodate changes through its base. Both scalability and robustness contribute to the cause of agility but it's the scalability that actually helps to achieve agility. Timely increase in the number of support systems and resources helps to cater various businesses at the same time and is easily able to gauge changes through the core systems to help achieve required productivity.

Of all the reasons, cost is perhaps one of the biggest purposes for executing a migrations project. Higher costs are always a driving force in an away drift from the current standings in terms of infrastructure. Whenever a migration project is planned, costs are calculated upfront. Costs related to expenditure on the migrations along with Return on Investment (ROI) values are measured along with TCO that is Total Cost of Ownership. ROI values help to determine the likely cost returns as against the overall expenditure on the migrations project whereas TCO helps to establish the direct and indirect cost of a system or a product that also includes the operating costs. This helps to prepare an analysis as to how much cost reduction is likely to happen post migrations project that is then compared against the existing operating costs.

Managing costs is an important objective of a migrations project life cycle and is pursued very aggressively by designated stakeholders. Even a slight negative variation in terms of costs is thoroughly analyzed.

Migration principles

Like every other process, IT migrations also works on few pre-defined but ever-evolving principles or guidelines. All the principles associated with the process of migrations help to strengthen, solidify and modernize the environment. Along with this, all these guidelines drive the business in a revenue generation mode that is always the prime objective of the Organization. the following chart mentioned are the principles associated with migrations project:

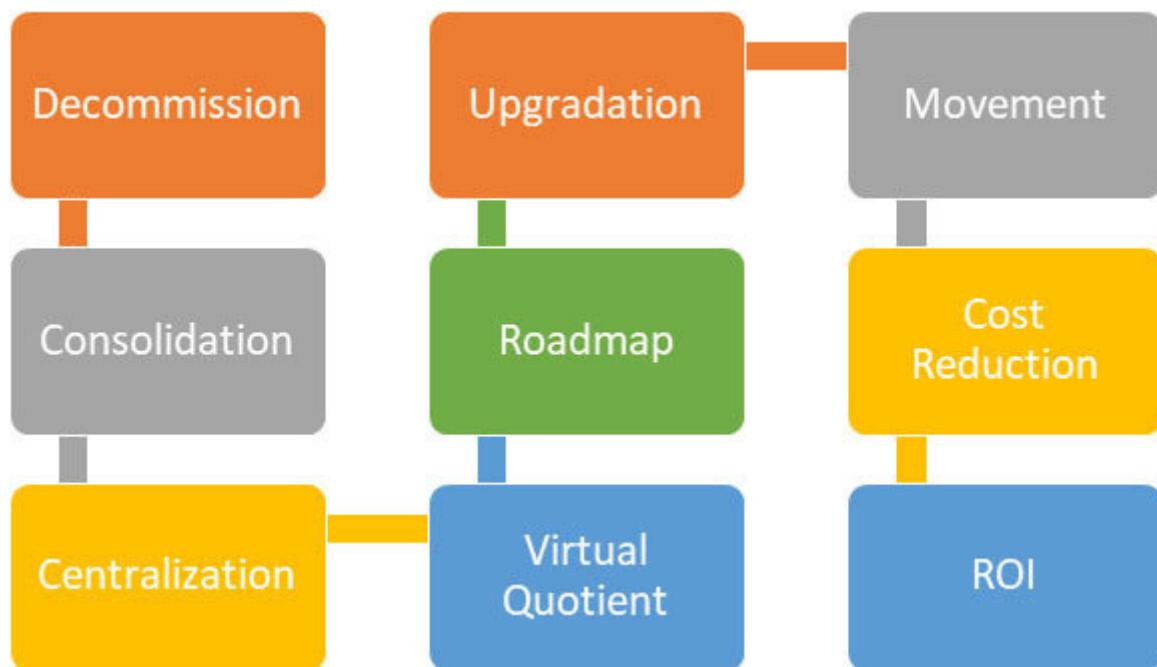


Figure 7.6: Migrations principles

Migrations aims to scrap out the components which are old and are not compatible enough to move to the next generation platform or hardware. It can be a piece of hardware or even an

application. All these components, generally called as legacy components, are identified during the assessment phase. They are assessed on the parameters of:

Future requirements

Still Billable

Compatible with latest platforms or systems

Newer/upgraded versions available

Vendor support availability

Application criticality

User base

Resource(s) requirements

Once all such pieces of the environments are identified, they are subjected to client review and subsequent approval to proceed with decommission or retire.

Consolidation of the environment is one of the major principles of a Migration project. Migrations aims to consolidate the environment that is common in nature but is scattered across. This can be easily elaborated in a way that if an application has

10 app servers and five of them are below 5% utilized for more than 8 months a year, then workloads of underutilized servers can be consolidated onto two servers from five.

Same way, a database server can be consolidated and served as a single entity for smaller applications that are only using a single instance of the DB server. This helps to structure the environment and makes it easily manageable in terms of ease. At the same time, it helps to free up resources in the environment that can be used to cater critical services or host new requirements.

Consolidation is termed as a key principle that is being applied during a migrations project. Principle of consolidations helps in tremendous cost reduction.

Centralized management of the environment helps in better administration and upkeep. All the infrastructure platforms that are an amalgamation of hardware, virtualization layers, and embedded software are used through a single console or a single mechanism making it more centralized. This option is particularly used by administrators or technical leads.

The basic idea is to reduce many ways of logging in to core infrastructure and the entire platform(s) is being managed through a safe and secure terminal that is under constant monitoring and has restricted access. This helps to minimize manual or human errors and ensure that only authorized and designated experts have complete access.

Additionally, centralized management platforms also provide daily reports of all tasks being carried out inside the consoles and also tracks the environment as a whole. This way, the entire environment is administered with ease while adhering to all the security protocols.

During migration discovery and assessments, a lot of components are identified which are actually running at a lower version or an old version of their own software application. This includes not only business critical applications but also infrastructure related and support applications as well. Upon discovery, it is then assessed if any such application or piece of environment can be upgraded to its latest available version. There are a lot of parameters that are to be assessed prior to making a decision about upgrade of an application:

Latest version available

Application compatibility

Criticality

Approximately cost for upgradation

Additional resources count

Available support

Client recommendations

Once all these parameters and volumetric data are collected, it is presented for client review as he is the one who will be taking a final decision on the application, whether to upgrade it or sunset it. The same process is followed for all the applications except for business applications where in Business Owner or DU Heads for the respective application takes a call on the application future. He would normally consult with his own team of administrators and leads and will review Application billability status and client indication for future use of the application. If the end product is already delivered to his client, business owner may decide to park the application and decide against any upgrades.

Migrations is the Ultimate principle that is being followed in a migrations project. Principle of migration is applied once the assessment phase is over for all the pieces and elements of the environment. On an average, about 98% of the components go through migrations where-in they are migrated to a new platform altogether.

Some components may be moved in an “As-Is” state but they will have a new dynamic host environment to run business applications. As-Is means similar state for the running workload in terms of operating system and resources but the advantage of migrations is there will be no error prone host environment. All the applications post-migration will be subjected to a fault-tolerant and highly efficient environment which has the capacity to auto-scale itself.

Migrations are carried out using a thorough and detailed plan that is known as migration plan. Migration plan contains end-to-end details about all the components that are to be migrated. It also contains details about migration tool that is going to be used during all migrations. We will be covering migrations plan and all associated documentation in detail in the next chapter.

Future Future roadmap or the strategic roadmap is the duration for which the infrastructure is prepared to be supported. For example: if the future roadmap of any particular application is 2 years, then application infrastructure will be designed in such a way that the application will continue to run seamlessly for the mentioned duration. All aspects which the application may need to function are planned and factored in advanced. This principle holds true for the entire Infrastructure platform that has been designed and earmarked for all the applications post migrations. The future roadmap will include planning for:

Resources

Extra resources as buffer

Disaster recovery scenario

High availability

Auto-scaling enablement due to demand peaks

Load-balancing application infrastructure

Secure access to remove manual errors and any malfunctions

Future roadmaps are driven by the duration mentioned in the contract document for the project.

Virtual The term virtual quotient refers to virtualization ratio of the environment. It means the ratio of virtual servers hosted on top of physical servers which are also called as host servers. It is a responsibility of the assessment team to find out the virtualization ratio of the client environment.

They dig out and create an analysis as to what is the average number of virtual servers or virtual machines as they are fondly called are hosted on a single physical server. They would assess on the basis of:

Total number of physical processors

CPU capacity (in GHz)

Total number of CPU cores

CPU utilization analysis

Total physical memory

Memory utilization analysis

As per industry norm, overall environment virtualization ratio between 85 and 90% is considered as healthy and in good zone. If the existing environment has virtualization ratio below 85%, then the vendor project team would target to get it past 90% post migrations.

It is those returns that are estimated on the overall investment on a migrations project. Every migration project is mammoth in size and requires a lot of funding to execute and process work orders. Every phase of migrations project life cycle requires cost expenditures to be on resources and procurements. Thus, on top of all the investments, returns on investment are always high on the agenda for the client organization.

Along with ROI, client organization also looks to tap the ballooning costs through the migrations project. It is always the high maintenance costs of the old, faulty, and error prone systems which prompts an organization to go for a migration to a new, robust, fault-tolerant, and cost efficient platform that can serve the requirements for a long time.

Also, since new technologies and platform promise a lot of new trends such as automation and innovation, thus, a lot of tasks from older platforms are automated and does not require any manual or human intervention. This implies less workforce on ground. This also helps to achieve cost reduction in a huge way.

Migration drivers

Every migration is first perceived based upon a few substantial parameters. Such extensive and expensive migration processes are always driven by some certain drivers or levers which fuel the fire of moving on to new, advanced, and better platforms. Companies seek migrations as they strive to stay in the business and these drivers are in-effect the real reasons behind such decisions.

Every organization need to showcase their capability and adaptability in terms of accommodating and running new and popular technologies, and they go through one of these reasons to make a push for a newline of their environment.

The following mentioned are some very high octane drivers beneath a migration project:

High costs

Frequent issues/outages

New potential business opportunity

New area/domain of business

The company is taking a change in direction

Promising client(s)

Below par performance of the existing or current system

The old system becomes deprecated or outdated

Compatibility issues

Vendor support for the existing system is not available anymore

Advent of technology in your realm of business

New version of application version is available

Fear of loss of revenue/clients

Fear of competitors swaying ahead

Evolving with technology or being banished from horizon

A new technology that drives processes faster becomes available

Migration strategy

Acute strategy and associated parameters are the baseline for any successful migration project. Strategy is as important as the project itself because if the right strategy is not formulated, then the entire project may go into shambles. It is the strategy that works as the basis of a project and drives it forward successfully. A strategy approach would normally involve:

Service(s) to be delivered

Strategy roadmap

Approach to be used in every phase of the project

Implementation approach

Vision of the project

Mission of the project

Aspirations

Long term objectives

Short term objectives

Tactics to apply

Performance audits

Gaps analysis

Contingency plans

It is always the top leadership team that is the think tank behind the strategy formation. They would also form a core group and lay the roadmap for a strategic roadmap. This would setup a base and serve as the main strategy approach for the project. It is then the responsibility of project management team to stick to laid out strategy and drive project ahead.

However, it is important to know that the senior leadership team will only formulate the core strategic approach which works as a whole throughout the life cycle of the project. Anytime project management team needs guidance with respect to the strategy, they would refer to core strategic approach. But the granular level strategies that relate to how a specific task or tasks are done, such low-level strategy formation is always done by the project leads and project managers.

Strategic approach formation is very critical for project folks who are actually on the ground. They need to know exactly as to what route they have to take once they start working on the project.

They would have to be absolutely sure of the mechanism to be adopted and the way forward in terms of the approach or else errors will start creeping in.

Once created, strategic approach is carried forward and adhered to throughout the project. Project leads and managers hold various sessions to elaborate strategic approach towards to all the respective team leads/track leads/PMO teams/Quality teams/compliance teams/team members, and so on. They ensure that every member who has been on boarded for the project is thoroughly acquainted with the strategic approach and everyone is on the same page as to what needs to be done and how it needs to be done. Strategic adherence is very strictly followed and tracked by compliance teams/quality teams or by PMO teams in some cases.

As part of the strategic approach, gaps analysis, contingency plans, and internal audits are planned. All these analyses and plans are required to track the individual performance and do the remediation's if required. It is required to keep a check on any manual errors or deviations that may result in major issues later on.

Strategic parameters

Strategies are formulated based on certain parameters. Here are a few critical strategy parameters which are must for a migration project:

Workloads inventory

App-Server mapping

Dependency charts

Application stack details

Planning/waves

End to end migration plan

Strategic approach definition

Documentation

Resources

Trainings

Risks register

Rollback plan/strategies

Backups-restore

Client processes – CRs

Communication channels

Bridge calls

IP changes

vLAN changes

Host name changes

Firewall ports

Load balancer requirements

All these strategy parameters are absolutely inevitable with respect to a migration project:

Date and timings

Reporting/templates

Vendors

Stakeholders

Approvals

Monitoring

POCs and testing

Pilot migrations

Results

Roadmap

Milestones

Up to date backups

Network bandwidth

Buffer time addition

Migration tool finalization

Final implementation

Benefits

Migrations projects are always pursued to get better returns. Such projects are always executed to tweak:

Productivity

Competence

Efficiency

Cost

Business as a whole

At the same time, an organization would target to nurture benefits such as:

Cost-effectiveness

Updated business processes

Billable applications only

Removal from legacy environments

Rise in potential of the organization

Relevant data flow

Overcoming of all weaknesses

Modernized platforms

Improved security

Greater efficiency

Tailor made industry standard platform

Flexibility to host multiple business critical applications

Versatility

Varied options and flavours

Conclusion

This marks the IT conclusion of this chapter on Migrations – Introduction. We have understood the meaning of migrations as a whole and what are IT migrations. We went through the impact of migrations and what is the purpose of a migrations project. Then we learned about Principles of migrations and the drivers behind every migration project along with benefits.

In the next chapter, we will look at Migrations – Life cycle and process flowchart. We will learn as to how a migrations process flow during a complete life cycle.

Points to remember

Migrations means movements as a whole

IT migrations means movement of IT environments

Migrations has its own set of principles and a specific purpose

It is always the strategy that drives a migration project ahead

CHAPTER 8

Migrations Types – Application Stacks – Move Groups

Introduction

In this chapter, we will read and master various types of migrations that are prevalent in IT world today. We will start with migration methodology and then will learn about migrations ranging from physical to virtual (P2V) and to cloud environments. Post that, we will jump into application stacks and will grasp as to what is an application stack. We will then move onto application move groups and will learn as to what a move group is and what is its significance.

Happy reading dear readers!!!

Structure

In this chapter we will cover the following topics:

Migration methodology

Types of migration

What are application stacks

What are application move groups?

Objectives

The main objective of this chapter is to learn about migration projects and some of their main components such as migration methodology which depicts as to how migrations are being carried out. Migration methodology is followed by topic types of migrations which lets us know as to how many types of migrations are there. Thereafter, we will do a detailed elaboration of application stacks and what they mean. Lastly, we will deep dive into various application move groups.

Migration methodology.

IT migrations varies from platform to platform. Infrastructure technology is a very wide landscape in itself and accommodates a huge range of devices, software(s), components, platforms, and so on. Elements of IT can range from a Nano device to huge mainframes and from a remote site location to a cloud location that has evolved over the last century. During IT migrations, each and every piece plays its part. But which all pieces will get to be engaged in an IT migration, it all depends on the type of IT migration you are carrying out.

IT migrations involve below major elements that are:

Source location

Migrations process

Migration tool

Data consistency

Target location

In laymen terms, a migrations happen from source location, which is the existing home of the IT environment that has been

earmarked for movement. This existing home or the source location can be any:

Datacenter

Remote or branch office

Client office

3rd party vendor location

Any small site office

Virtual platform

Cloud platform

Migrations may occur to a similar platform in a target location that can also be any of the ones mentioned as the source location. A target location is the location where IT objects are destined to reach post movements and function thereafter. This movement from source location to the target location is migrations and the entire process is called as **migrations**

It is worth mentioning that migrations may be of the smallest magnitude sometimes like moving a few users from their current desk to another within a same office. And sometimes, migrations can be of humongous proportions like moving an entire

datacenter to another locations that contain thousands of physical and virtual workloads accessed by hundreds and thousands of global users.

Migrations may range from a physical platform to another physical platform or from a DC to a cloud platform. The movement of an application setup from one hardware base to another within the same datacenter is also a type of migration. Migrations are normally planned well in advance and involve a lot of instructions and guidelines. But all the preparation which is done prior to finalizing a migration depends on the:

Target location

Application architecture

Strategic roadmap

A setup is moved with a motive of saving cost and making an application run with new efficiency and potential on a new set of hardware. Thus, the target locations play a huge role in determining the nature and specification of an IT migrations project.

Even if the target locations can vary from a wide range of platforms, basic concepts and principles of an IT migrations remain the same, both technically and non-technically:

Infrastructure Technology

Source Location

Migrations Process

Target Location

Types of migrations

There are various types of migrations with different categories and flavours. Let us explore various types of migrations that are prevalent in today's infrastructure technology world:

P2P – Physical to Physical

P2V – Physical to Virtual

V2V – Virtual to Virtual

C2C – Cloud to Cloud

All of these are main categories of all kinds of migrations and are split further:

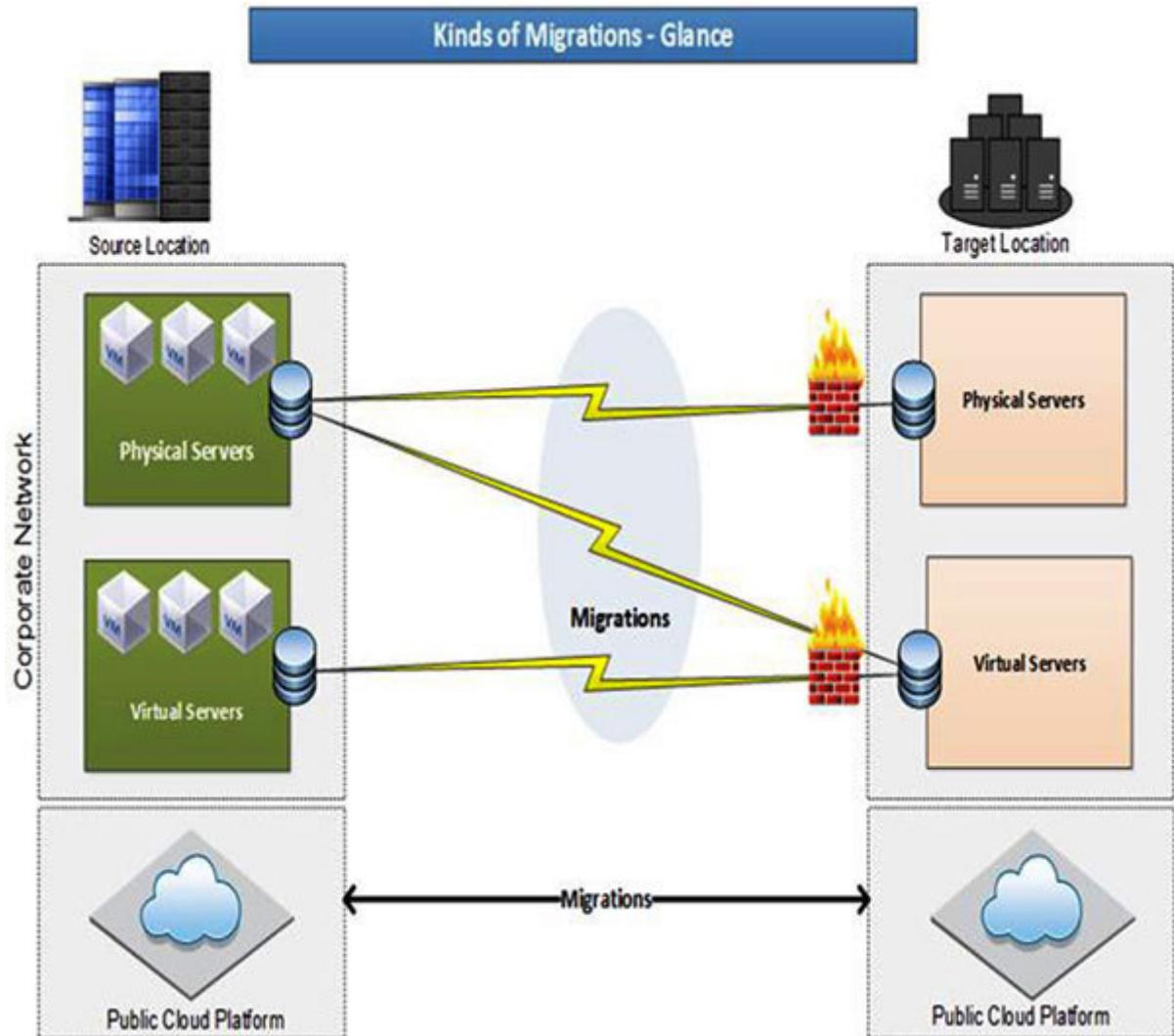


Figure 8.2: Types of migrations

This stands for **Physical to Physical** that means migration from one physical platform to another. It is not necessary that both the source and target physical bases are in the same datacenters. In this case, both source and target locations can be in a single DC or in different DCs as well. Significant point is the physical movement of the environment that is being migrated. This physical movement can be done via two ways which we will go through both of them.

Fork lift In a forklift migration, application hardware is physically packed up and moved from one data center to another. Forklift is a P2P type of migration which implies that the same hardware is being used in the target location as well. Such a migration is normally carried out due to specific nature of the application where it requires a specific set of hardware only to run and function. Thus, it becomes absolutely necessary to move the exact piece of hardware to the new location to keep the application functional. Such migrations are carried out via the specialists which are third-party vendors and have the expertise and prior experience to carefully pack and move a hardware from one location to another. They ensure that under any circumstances, there is no damage to the hardware and it is being transported with maximum care.

Even though forklift migrations are always part of overall migrations plan, they are however budgeted separately and are paid by the client upfront to the vendor who will actually move the hardware items. All the coordination work is done by the project team along with client SPOCs which includes:

Shut down the application and DBs

Ensuring full application-DB backups

Communication to all users and stakeholders

Discussions with appropriate vendors to facilitate migrations

Arranging vendor quotes and price negotiations

Client approvals

Arranging physical access to both source and target locations

Follow-ups

Post migration checklists and validations

Once the hardware is moved to the new location, it is simply powered on and connected with the network to make it operational again.

Swing kit Swing kit migrations are popularly known as swing moves and are a P2P kind of migration, albeit a little different one. In this type of migration, hardware devices stay positioned and are not moved anywhere. In a swing kit migration, it is required to duplicate the application hardware in the new data center and then migrate the application and associated data to the new environment. Once again, it is always application requirements to run on a specific set of hardware or configuration to be fully functional.

Thus, new pieces of hardware are setup exactly in the same way and with the same configuration as is available in the existing source location. Once the hardware is setup in the new DC, a POC (proof of concept) is carried out on the newly configured hardware. POC test finds out:

Application functionality testing

Its features

IOPS

Compatibility issues

Peak load handling capability of hardware

Application – database connections

Transactions status

Accessibility

Once POC test results are established and are found to be good enough to host the full version of the application, then the application is switched off from the source location and is enabled on the target location or at the new DC. Before turning off the application and its DB at source DC, every bit of data is backed-up and multiple copies of backup are maintained.

Once the application is enabled at the new DC, applications SPOCs are required to perform validation testing before a Go-Live signal is given which marks the completion of a swing kit migration.

Emulation This is a very rare kind of migration and is not just limited to P2P migrations. This kind of migration can happen from P2P and P2V as well. This can be done with in a DC, with in two different DCs and also from a DC to a cloud platform.

Emulation migrations are generally applied to legacy servers/applications that reside on an old piece of hardware and run an outdated version of the application. However, despite the fact that they are legacy, they are still termed as business critical and are deemed important. A few examples would be SunSPARC, VAX, PDP, Alpha Servers, OpenVMS, and HP-UX servers.

Such applications are generally not compatible with the latest hardware sets and they neither have the latest software versions as well. However, due to operational requirements and contract commitments, if it is required to keep such applications alive, then they are emulated to the new platforms. Emulation here means that creating an exact same replica of the application environment but without any set of hardware.

The emulation work is carried out via a third-party software that is specially designed to create an emulation such as Stromasys. This emulation can be created on a new physical platform, a virtual platform, or on a cloud platform as well:

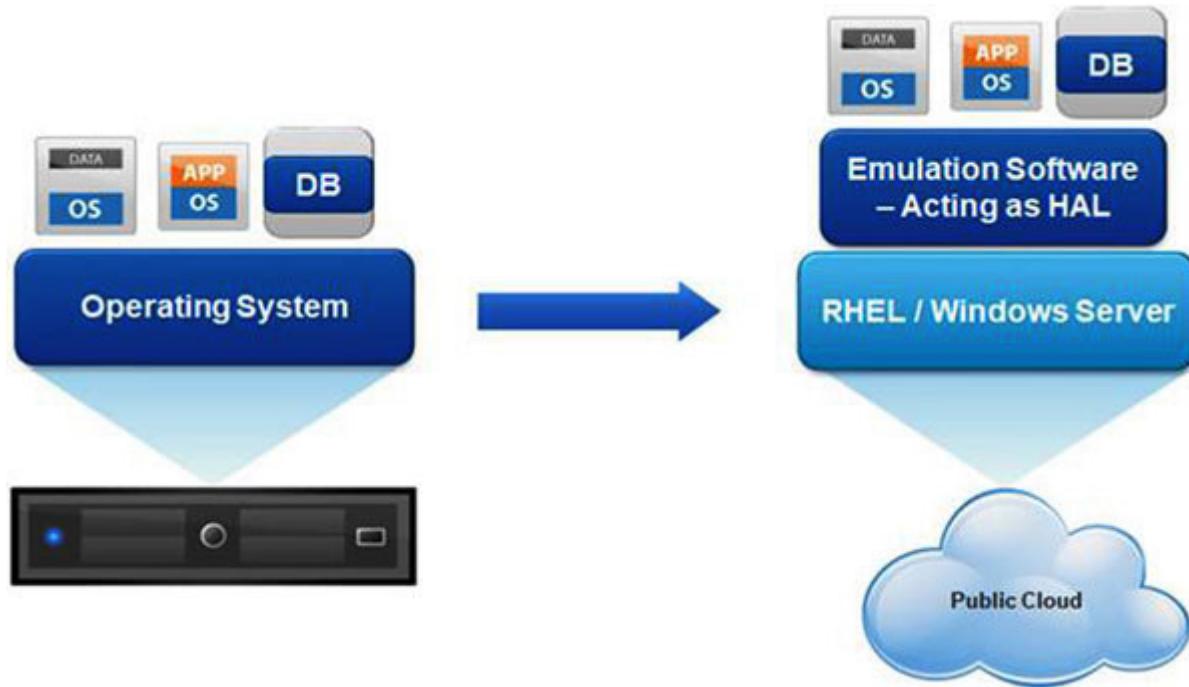


Figure 8.3: Emulation software

The emulation of the environment ensures that applications gets to live on in a new environment and also helps the organization to get rid of old and legacy hardware. However, these types of migrations are rare as they tend to be very expensive and are used only in the case of old and legacy environments where there is no other option available to migrate.

This stands for **Physical to Virtual** and it means that migration from a physical platform to a virtual platform. P2V is not just migration but it is also a conversion of a server or an environment from a Physical base to a brand new virtual platform. This type of migration may take place:

Inside a data center (both source and target locations reside in a single DC)

From one DC to another DC

From a DC to a cloud platform

During the assessment phase, all the servers and applications are assessed and the ones which are compatible with virtualization technologies, are migrated onto new virtual platforms. Virtual platforms provide much needed ease of management and it is always less expensive than a physical platform as there are minimal maintenance costs. All the virtual servers are managed through a single management console and access is restricted and authorized:

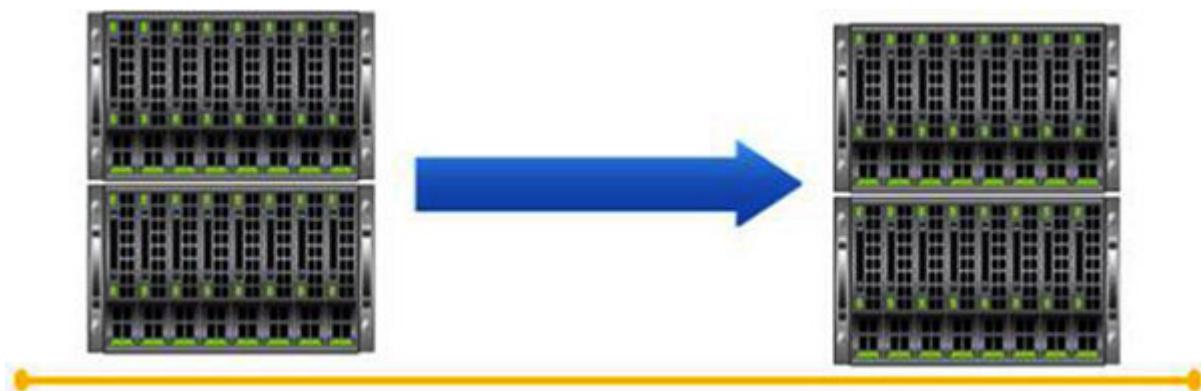


Figure 8.4: Physical to virtual migration

P2V migrations have a lot of significance as this is the migration that is being undertaken during DC consolidation/upgradation or migration projects. In a physical server setup, a piece of hardware is more often than not used to host a single server. However, in a virtual server setup, the same physical server can be used to

host multiple virtual servers. This can be done in three ways and lets go through them one by one.

P2V using virtualization In P2V migrations, virtual setups are created using specialized virtualization software(s) or technologies that are deployed on top of the hardware servers so as to enable them to host virtual servers.

This virtualization software acts as a piece of operating system (OS) in itself which are called as A hypervisor is a thin layer of software that interacts with the underlying resources of a physical server (host hardware server) and allocates those resources to virtual server OS. The OS of a virtual servers requests resources from the hypervisor.

Hypervisor, when deployed, amalgamates the resources of the hardware servers such as memory, disk, and CPU, and so on, and then divides them to create multiple virtual servers:



Figure 8.5: P2V migrations

Each virtual machine runs its own OS and behaves like an independent server, even though it is running on a portion of the actual underlying server hardware. The virtualization software ensures that the host hardware server never runs out of any resources itself by providing hardware efficiency and performance reports.

Virtualization technologies are capable of forming clusters of various hardware servers. This means that various physical servers are combined together to make a *Cluster* and they create a pool

of resources (processor, memory, disk, and so on). This cluster as a whole is then used to host many and multiple Virtual servers and all the resources to these virtual servers are assigned from this pool only. This cluster ensures that even with failure of any hardware from the cluster, other hardware boxes quickly share the load of the failed server, thereby ensuring that the application continues to run and there are no outages.

There are many Virtualization technologies that are prevalent in the market today such as VMware, Microsoft Hyper-V, and so on.

P2C stands for **Physical to Cloud**. This is a kind of migration that migrates and converts a physical server into a virtual server that is actually hosted on a cloud platform. A cloud platform is actually a datacenter that is hosted by a cloud provider and contains hundreds and thousands of hardware servers clustered together. The cloud provider allows you to use these hardware servers and their associated clusters so that virtual servers can be carved out:

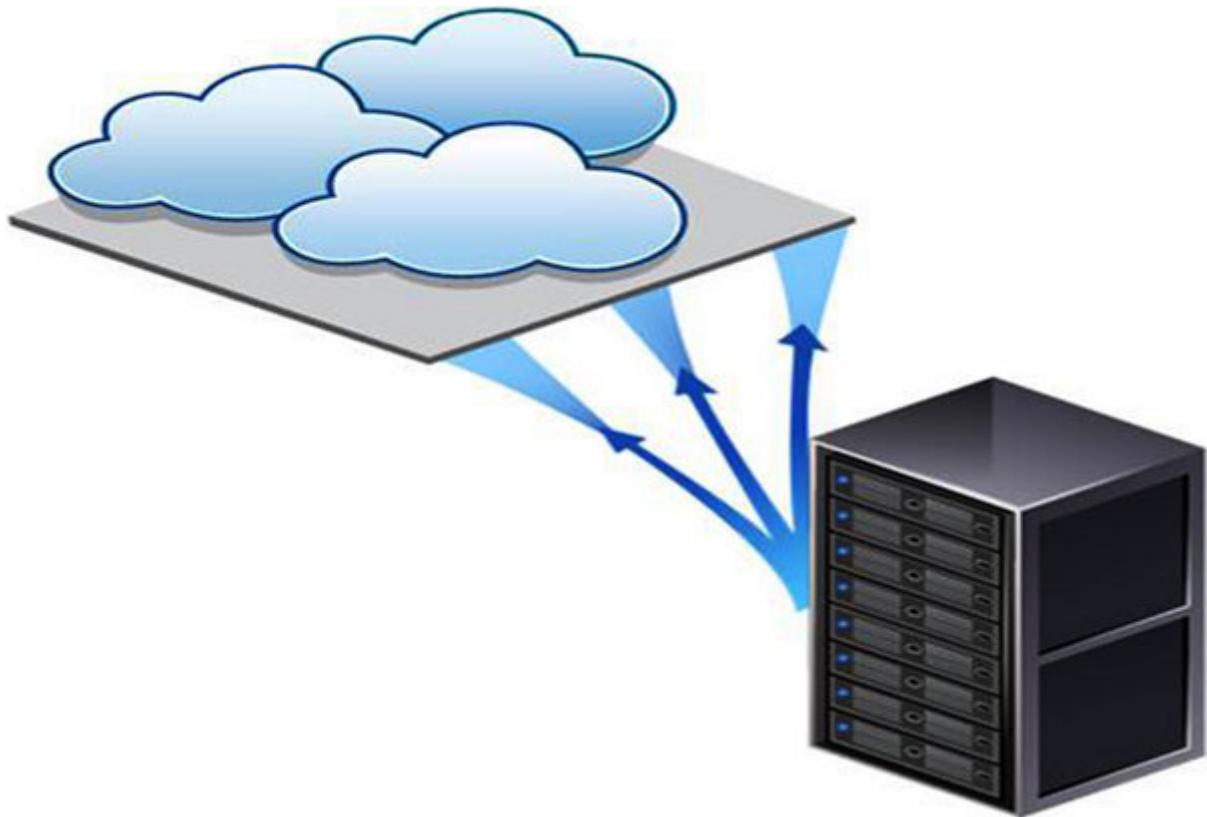


Figure 8.6: Physical to cloud

When a server is migrated onto cloud, it loses all its characteristics that are associated with a physical server such as dedicated resources and isolation. On a cloud platform, server is assigned dedicated resources again but they can be scaled UP or down as per the requirements or load. Cloud migrations are very significant in today's IT world as the general notion is that cloud saves cost.

Cloud migrations are just like any other migrations and come with a set of protocols and guidelines as is usually the case with any other migration. A cloud migration would require the migrating application to be compatible with the cloud platform that is basically the first step in the migrations assessment and is

followed by all the other set of requirements such as establishing network connectivity, access controls, and so on.

Cloud platform come with a variety of features/services and are capable of hosting any domain-specific environments such as:

Pharmaceuticals

Banking

Logistics

IT companies

Telecom sector

Federal Government

Media and streaming

This makes an easy decision for an organization to move its environments onto cloud.

Also, cloud providers keep base and underlying hardware well maintained and up-to-date, this helps in lesser number of outages. One major advantage with cloud platforms is that you do not have to worry about hardware maintenance, cost, age, replacement, and so on, as all this is taken care by cloud

provider. An organization just has to take care and pay for the workloads they are running on the cloud.

This stands for **Virtual to Virtual** and points to a migration which happens between a single virtual platform to another virtual platform. These kinds of migrations may happen between a single DC, between two different DCs, between a DC and a cloud platform and between two different cloud platforms as well.

V2V migrations do not happen very often as in most of the cases virtual platforms are the final destination. However, there are a few cases where in virtual migrations takes place such as:

Platform This refers to virtualization platform upgrade process wherein the platform is updated from an old version to possibly the latest version and in the process, the entire infrastructure on top of the platform is also migrated along-with. This type of migration can happen with in a DC or with in different DCs.

An example of this would a VMware virtualization platform getting upgraded to version 6 from the current five which would require all the host hardware servers to be upgraded or migrated on to the new version. Another example would be upgrading a Windows 2012 or 2016 server to Windows 2019 Server OS version:

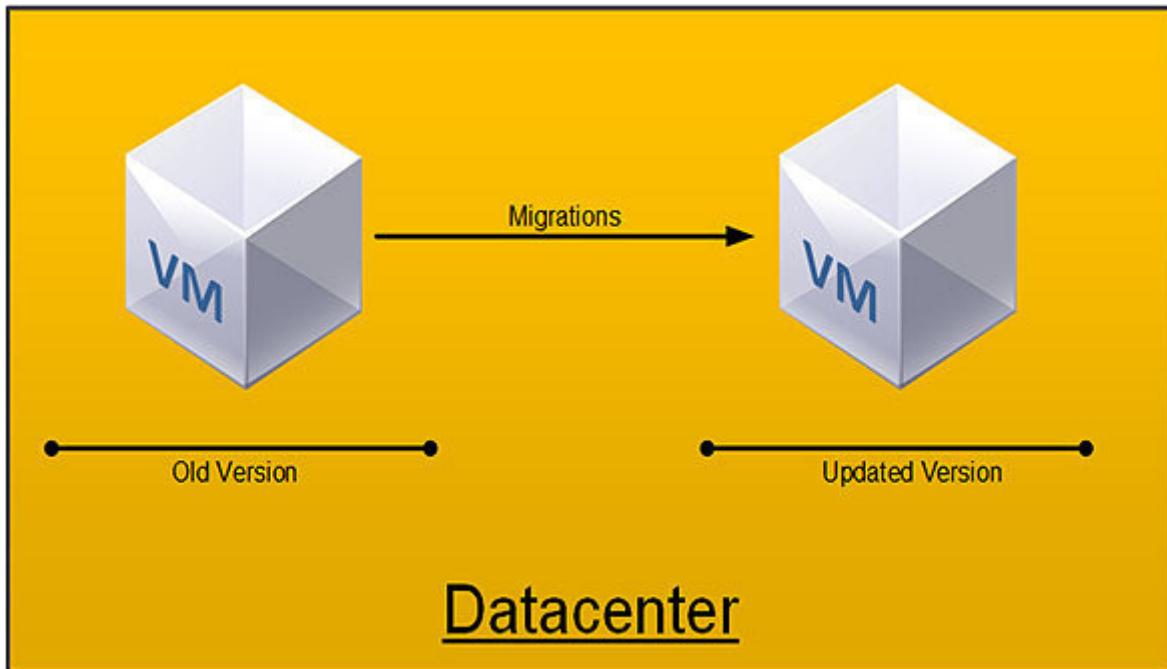


Figure 8.7: Platform upgrade migration

During the platform upgrade process, all the running virtual servers will also need to be updated with the latest platform specifications. This is required as a VM with the specification for old environment will not be able to function on the newly updated environment. An example for this would be VMware VM tools. Post platform upgrade, all the instances are updated so as to make them compatible with the new environment and are able to use all available features and services.

These types of migrations are not very common and happen rarely. These migrations are actually well-planned and at times done on vendor inputs as well. It is a way of keeping the infrastructure environment and platforms up-to-date by feeding them with the latest versions available in the market. These kinds of migrations helps an organization to run the application on the latest version of the host environments. This also helps in vendor

support as the support is always available for the latest versions of the platform products.

Switch Switch migrations are also known as platform change migrations, which means the change in underlying virtualization host platform. These migrations may occur within a DC or between different DCs. Switch migrations simply means that the organization has decided to switch the virtualization software on which all the virtual workloads are running to a new one. This also means that all the virtual servers will also be switched on to the new virtualization platform that is a platform change migration.

Switch migrations does not necessarily mean any change in the hardware as the same hardware can be used for the new virtualization technology as well. Hardware replacements are huge in magnitude in terms of cost and thus, prior to finalizing underlying virtualization platform change, the organization would normally take a stock of all the hardware in its datacenters. Before zeroing on a new virtualization platform, they would assess the hardware for their:

Age

Capacity

Current usage

Available capacity

Load on top of the clusters (hardware)

Upcoming application requirements

Along with hardware details, they would also assess if the proposed virtualization platform has any specific requirements and is it compatible with the available sets of hardware servers. Along with, it is critical to find out as to how the migrations from existing to a new platform would take place. Organizations would not want to make things complex and risk the business and thus may go for POCs. It is absolutely important to do a complete testing and identify any possible risks prior to migrating the environment to a new platform.

These migrations are done in phases and environments are migrated one by one. Hardware servers are picked up cluster by cluster and are migrated/upgraded with new virtualization platform. Post that, all the VMs are then converted into files supported by new platform. This conversion or migration of VMs to new platform is also carried out via a specialized tool such as virtual machine converter.

Once the VMs are converted, they are powered back on the set of hardware that has been already migrated and is ready with new virtual platform software. Post powering on, VMs would pick all the required specifications and would be operational:

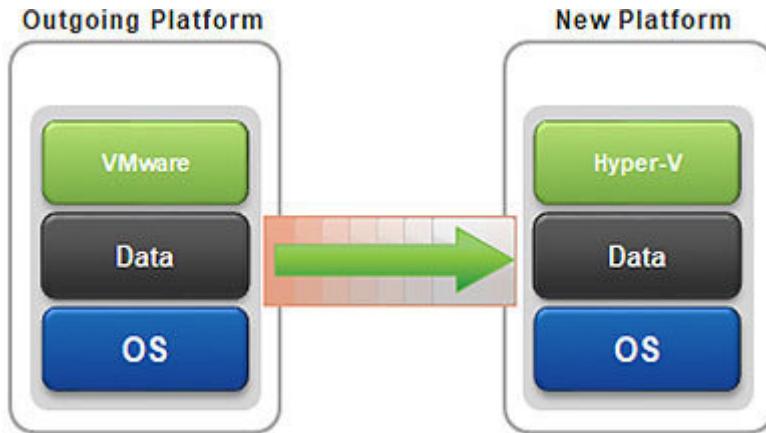


Figure 8.8: Upgrade

An example for this can be a switch from VMware virtualization platform to Microsoft Hyper-V platform. And the reasons behind the switch can be both technical as well as non-technical, but they certainly include cost as a prime factor responsible for the switch.

It is very much likely that the new virtualization platform may be equally capable and may offer almost the same set of services or features, but making a switch from one platform to another is actually strategic decisions and is generally based on strategic roadmaps and reducing costs. Such decisions are taken by organization's senior leadership team (likes of CEO/CTO, and so on).

This stands for **Virtual to Cloud** and represents a migration from an On-prem virtualized environment to a public cloud platform. This kind of migration is purely virtual in nature and is carried between a datacenter/remote site to any of the public cloud platforms.

Virtual to cloud migration basically means that the parent organization is looking to make use of cloud services. This implies that organization has decided to pack one or all of its operating environments and decided to move them all on-to a cloud platform that deems more desirable to their needs in terms of efficiency and cost:

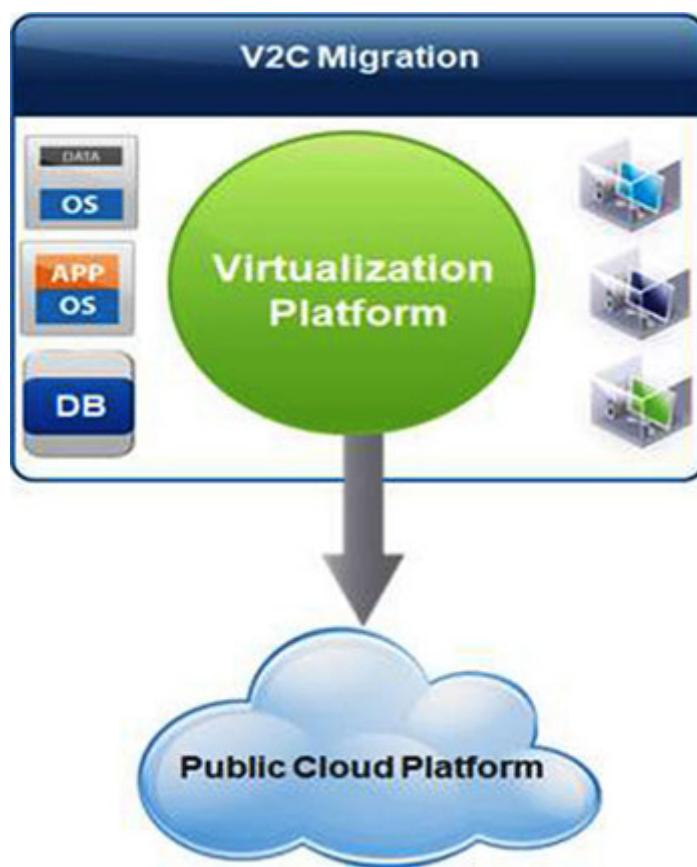


Figure 8.9: Virtual to cloud

There can be a number of reasons as to why such migrations are done. First and foremost, the reason that comes to mind is the cost factor (for obvious reasons) along with performance and efficiency. A very interesting point here to note would be that such migrations are decided well ahead renewal/replacement/upgrade cycle for:

Hardware

AMCs

Warranties

Software(s)/Applications

Databases

Organization would make an assessment and setup a baseline estimate as to how much funding is required to replace an old and aging hardware or renew the maintenance/support from vendors. These particular costs normally run into huge chunks of money and organizations are often compelled to make these purchases in order to keep up the hardware up, up to-date, running, fault-tolerant, and efficient.

However, with the advent of cloud technologies, organizations now have an option to not make these purchases and save costs that are spent on hardware replacements and renewals of any support contracts. With cloud, they are only required to do pay for the servers/Apps they are running on cloud with no provisions for any support contracts /warranties/AMCs/hardware replacements, and so on.

This is not their headache anymore and organizations do not have to shell out extra cash piles. They can instead use the same amount of money to grow the business. Also, in terms of performance, cloud provides numerous options and features to host and run basically any type of environment, it does not matter as to how so ever resource demanding the application or the database is.

This spells all answers for any organization looking for an optimized performance and highly efficient environment to run its workloads at minimum cost. This provides enough fuel to pave the way for a V2C migration.

This stands for **Cloud to Cloud** and represents a migration between two different public cloud platforms. These kinds of migrations does not happen in any of the datacenter or any remote site, it happens over the Internet and on a vendor-hosted cloud platform.

C2C migrations are new and one of their own kind and to speak of statistically, they are very rare. There is a very little chance (like 1 out 100) that an organization would move their operating environments from one cloud platform to a different one, unless Of course, there is a huge price difference or it is a specific requirement from their own clients:

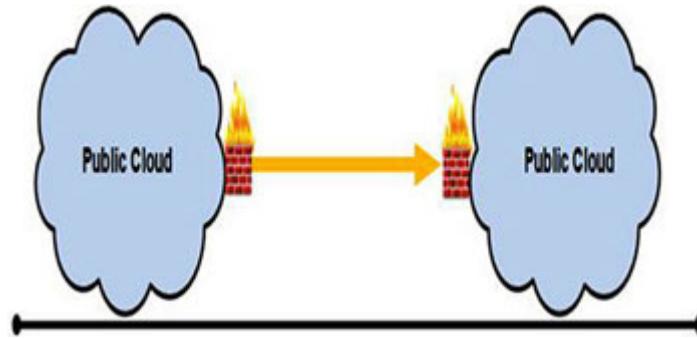


Figure 8.10: Cloud to cloud

For a C2C migration to materialize, recommendation from a client to an organization for a specific Cloud Vendor and platform is the prime reason. It has been noted that around 99% C2C migrations have happened due to client requests to an organization.

Let us take an example: An organization X is having a tie-up with a public cloud provider AM. X decides to outsource, floats a RFP, gets responses, completes the assessment, and then finally selects a suitable partner in Y. However, Y is running its workloads on a public Cloud AZ. But when X selects Y to run and supports its business, it asks Y to run everything on Cloud AM and not AZ. This will prompt Y to migrate all the concerned workloads and environments (which are required to support the environment of X) from AM to AZ.

Along with client recommendations and cost factor, another reason which may trigger a C2C migration is the strategic roadmap. Let us take an example to understand this. If an organization has to setup and run an application environment and a certain cloud provider is actually the manufacturer of that application, then in that case a scenario can develop for a C2C migration. For

example: if an organization wants to host an application manufactured by Microsoft, then Microsoft may offer them special discounts and deals to host their IT environment on top of their cloud platform that is Microsoft Azure. Similarly, if a cloud platform hosts specific features for developers and coding, then an organization may migrate to avail those features.

Generally, on the comparison, all the cloud platforms are almost at par with each other in terms of capability, reach, and features. Almost all of the clouds are accessible through the main business and prominent geographies of the world. However, yet there are a few instances where organizations have made a move from one cloud platform to another. However, as per statistics, about 97% of C2C Migrations are due to strategic reasons only or due to client recommendations.

Application stacks

An application is the face of the business. It is a world of hundreds of components in itself. It contains a lot of components, which amalgamate together and form a bundle that helps to run an application. All these components are part of an application stack.

An application stack is a set of individual components which are packed together to run collectively as a function in the form of an application.

All the components of an application stack are also called as a **Suite** that is again a bundle of similar and compatible constituents to be executed to operationalize an application. It is understandable that all these constituents are very closely linked with each other and are also functionally dependent on each other to force and factor the desired output.

An application stacks is responsible for generating an environment feasible enough for an application to function. All the bundled components are required to ease out the processing workflows and to also manage all related tasks. An example would be the Microsoft Office application, which is a bundle of not only many components but also applications. All these applications as part of Microsoft office are used individually but they all use the same stack files for installation and processing. They even use the same

executable which interacts with the underlying OS for resources request, consumption and processing transactions.

An application cannot function even if a single component is missing or not functioning. At times, a single stack is used by various applications which are of similar nature (like web based, agent based, from the same manufacturer). This is known as the **Dependencies** in terms of application universe where in one application uses the components of a different application to function. This is where an application stack is very important from a migration's point of view. It makes it very important to find out as to which applications are functioning as a single application and which ones are dependent on other applications:

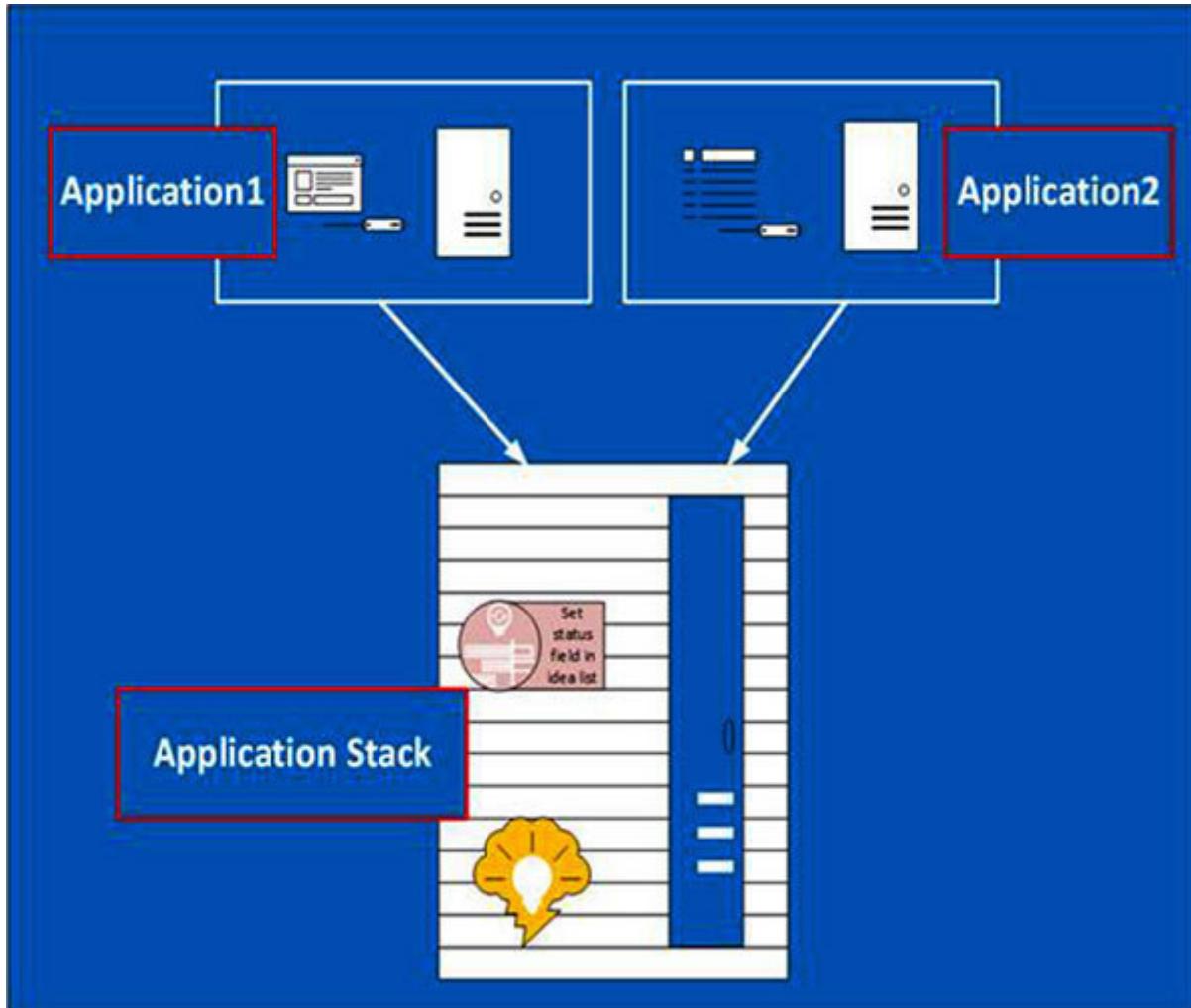


Figure 8.11: Application stacks

These application stack details and associated dependency details are discovered and analyzed during **Assessment** phase of a migration. An application and all its server(s) cannot be migrated unless dependency details are established and application stacks are known.

Thus, during discovery phase, applications and server details are discovered first. Then, few specific tools are used to establish application-server mapping and application dependency details that are actually driven from the application stacks only. As for tools,

there are a few industry specific ones that help to establish a clear mapping between Infrastructure components such as servers, storage or network identities and between application elements and modules.

Once application stacks are established along with applications-server mapping and associated dependency details, it marks the end of an important phase and set of activities in migrations phase. This is one of the most important set of activities where in application details are discovered, then they are mapped with Infrastructure components which is followed by forming dependency details. This paves the way for structuring and carving out move groups for actual migrations, which is our next topic.

Application move groups

An application move group is a group of applications that are tethered together due to a similar feature or a dependency. This group of applications is formed with a target of migrating them all together, in a single migration execution movement:

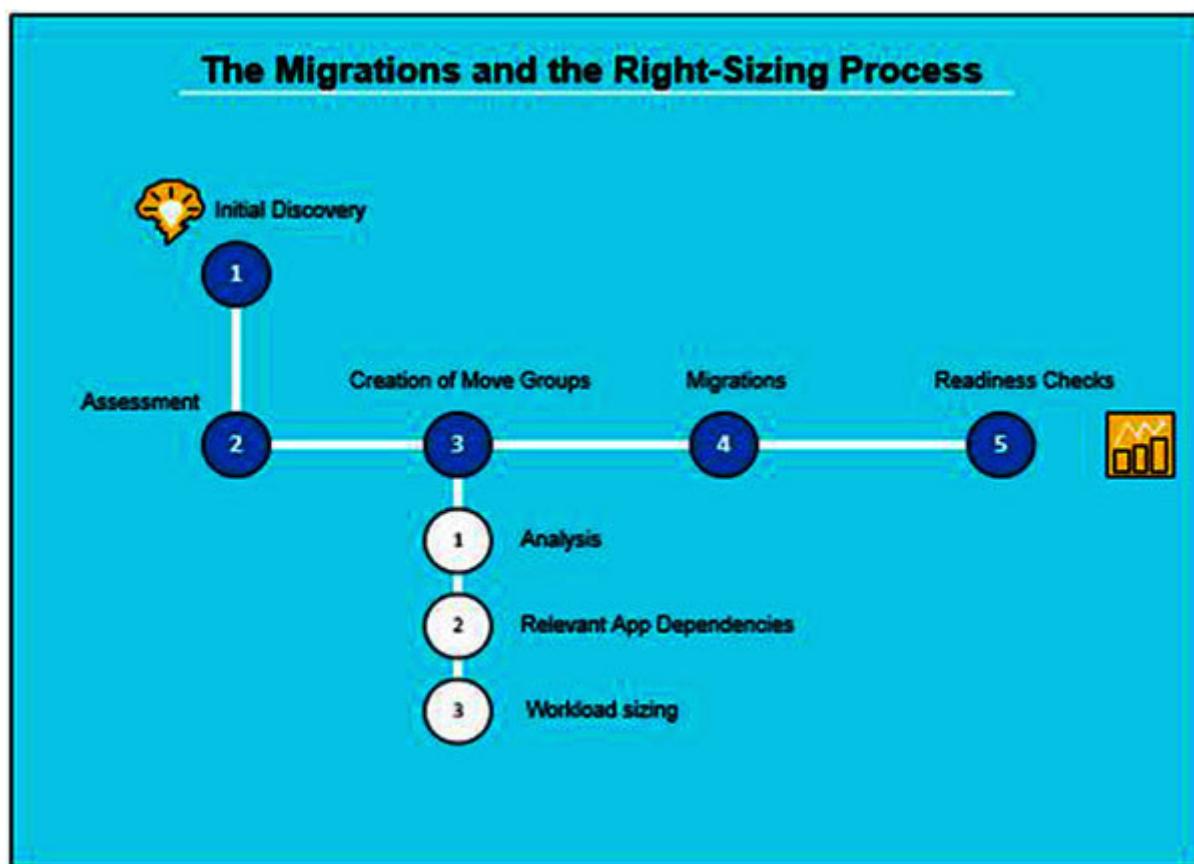


Figure 8.12: Application move groups sizing

Application move groups are created during **Assessment** phase of a migration project. Formation of application move groups is done so as to split the entire application-infrastructure landscape which

is due for migrations. Based on the discovery reports, applications-server's mappings are done and dependency analysis is done. This helps to establish the applications which are similar in nature or are interdependent on each other. The objective of forming an application move group is to migrate the entire group together in a single go.

The breaking down of entire IT landscape into various small groups helps to manage them together. However, it is vital to understand that all the move groups are created after a very detailed and thorough study of the applications and their relationships. This also includes a comprehensive analysis of all the dependencies which helps to link the applications together.

An application move group certainly helps to smoothen up a migration project as it splits entire process into smaller chunks which are far easier than migrating huge cache of servers. Move group's helps to form a reliable set of applications and their servers and their infrastructure components which gives a sense of comfort to the migrating team in a sense, if the migration fails, then it is easier to revert a smaller number of servers.

Application move groups are generally formed with the help of application administrators, SPOCs or owners. At times, it can be a third-party vendor as well who can help with inputs to form a reliable move group.

A move group will generally contain all the servers of an application module or the entire application itself depending upon how large the landscape is. An application landscape may contain:

One or more application servers that contains the actual application software

Web servers

Database servers

Specific network vLANs and IP addresses

Specific firewall rules

Exemption policies from anti-virus scans

Specific ports access to vulnerability scans

Global clients access provision

Specific groups in ADS

Specific routes in the routing table to filter and route the traffic

VPN Access to application modules

Significance of application move groups

Here are some of the benefits of forming move groups:

Move groups are assigned names or numbers which makes it easy to identify as per moving application's landscape

Number of servers migrating in a move group can be easily adjusted as the requirements of the business unit

Different dates are assigned to different move groups which makes it easy to track, trace and plan

Move groups are planned ahead and are assigned resources as per the number of moving servers and other requirements in advance

Setting up move groups enables to move through the discovery and migration phase of project with accelerated ease

Conclusion

This marks the conclusion of this chapter on *Migration Types – Application Stacks – Application Move*. We have feasted thoroughly on all the varieties of migrations that are available in the market today. We have also gone deeply into physical, virtual and cloud-based migrations. We then learned about application stacks and their basic concepts followed by application move groups and the significance they bring into a migration project.

In the next chapter, we will look at various migration components and a full-stack migration life cycle.

Points to remember

Migrations range from P2V to cloud platforms

Application stacks are components of the application

Move groups contain related applications

Move groups make it easy to migrate

CHAPTER 9

The Migrations Project – Process Flow-Chart

Introduction

In this chapter, we will read and understand about different Migration components that are associated with a Migrations project. We will then move onto understanding a full-fledged migration project life cycle, wherein we will understand everything phase by phase through a process workflow.

Happy reading dear readers!!!

Structure

In this chapter, we will cover the following topics:

Migration components

Migration life cycle

Migration process flow

Migration phases

Objectives

Our main objective in this chapter is to learn about a Migrations Project's:

Life cycle

Its phases

Process flowcharts and why they are used

What are the benefits of using a flowchart?

Detailed elaboration for each phase

Migration components

Infrastructure technology migration projects consist of various components. These components form the core of a Migration project and are actually the components that are to be migrated. Main Migration components range from:

Infrastructure components

Database components

Application components

Testing components

Tools components

All these components basically sum up the entire migration project life cycle and the IT environment as a whole. However, all these main components are further segregated into various other sections and their respective components. Let us have a look at all the components which are assessed during a Migration project life cycle and are considered for migrations:

Infrastructure Infrastructure contains various core domains and thus Infrastructure components form the biggest huddle of

components. These include:

Network

Security

Compute

Storage

Backup-archive

Compute domain along with Network, Security, and Storage domains form the backbone of any Infrastructure environment. For any migration project, all these domains are considered to be an automatic choice for migrations. Compute would contain workloads pertaining to various application stacks and also for their respective and associated databases instances.

Security controls the incoming and outgoing traffic for the application environment along with controlling the access. Network forms the layer on which application transactions take place and all environment variables are connected.

Storage domains provide end-to-end storage services and this implies safely storing all the critical data for the application, its processes and related databases into storage devices.

Lastly, it is the backup and archival services that are used as core Infrastructure services. All components of the environment are required to be backed up to be retrieved during any sort of emergencies. Archives are used to storing historical or old data which is not accessed frequently. However, such data is stored as it contains the value and is sometimes used as references.

It is near improbable for any business unit to function without using these infrastructure components. And thus, these infra components make automatically to the list of migration candidates:

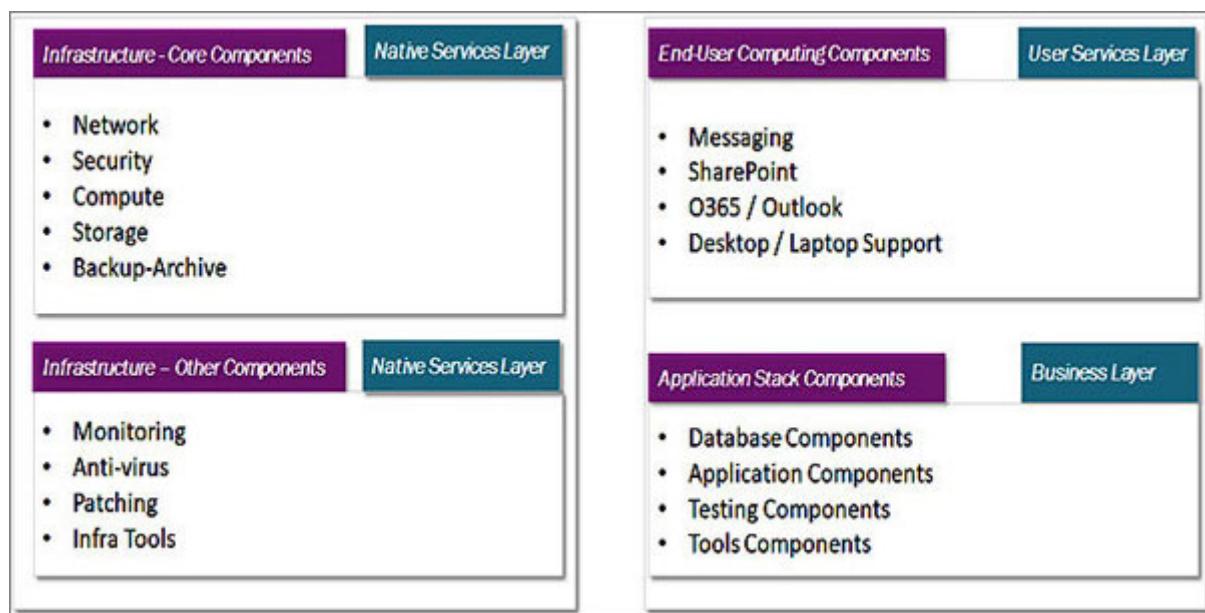


Figure 9.1: Migration components

Apart from core infrastructure components, end-user computing components also make the cut for infra components, albeit not the core ones. End-user computing components of the environments deal with end-user services such as:

Messaging

SharePoint

O365/Outlook

Desktop/laptop support

End-user computing components are not automatically considered as a part of migration components as they are not considered to be essential components for a business to run. End user components are services by an organization to its user base for effective communication and operational ability.

This includes Messaging which contains organization's official messenger tool such as Microsoft Teams. Messaging tools are used for quick and effective communications and are also used for hosting internal and external official meetings.

Along with Messaging, organizations usually setup a SharePoint portal where all the critical business documents are uploaded and shared with required stakeholders. SharePoint portal gives flexibility to each business units to setup their own dedicated space and to upload their business documents. Documents are shared via SharePoint links with external parties like clients and vendors if a need arises during an engagement.

Outlook or the latest O365 platform is the Exchange Tool that is being provided to all the users of the organization. This is the platform where in the electronic mails are being exchanged and is generally the most critical component under end-user computing. Emails are the most common medium of communication across any organization and between an organization and its client as well. Thus, specific care is taken and all requirements are met while setting exchange environments.

End-user computing also contains desktop and laptop support for all the users. This would normally include Imaging of the devices, catering software requests and installations, handling any possible hardware issues, coordination with vendors and providing daily support and remediation for any issues.

Infrastructure components also contain a few native components which are equally critical from a migration's perspective. These components are not always migrated in terms of servers but will have to present in the Target location or DC. All the native components are mentioned as follows:

Monitoring

Anti-virus

Patching

Infra tools

Native infrastructure components include monitoring tools which includes Hardware and software monitoring tools for all the Datacenter based equipment's, devices, and servers. Monitoring tools are setup to observe, monitor, and report 24x7 activities for all the DC components. Apart from the DC components, monitoring tools are also used to monitor activities being carried out by the users on their respective desktops and laptops as well to track and trace any violations or policy breach.

Patching is another native service that has to be either part of migrations or it has to be newly built and setup in the target DC location. Patching provides the latest set of patches and hotfixes which are meant to remove any sort of anomalies from the operating environment which includes both DC components and user-based devices. Latest patches deployment on all deployment strengthens the overall computing environment and saves them from all any deficiencies.

Anti-virus is a very critical component considering the vulnerabilities present in today's IT world. It is absolutely imperative to load anti-virus software on all the eligible servers and devices in the DC along with all the user-based devices of the organization. Up-to-date virus definition on all the functional devices in the environment is like a top priority and is a mandatory requirement for any IT environment. Anti-Virus service is a must for any and all operating environments.

Last but not the least, apart from Infrastructure components, there is another set of components that are directly part of a migrations project. These components are the actual core of the

application stack and are directly responsible for an application to function and run on top of the underlying Infrastructure components and platform. These components are also called as the business components of the environment. These components include:

Application components

Database components

Testing components

Tools components

Applications are business and applications generate revenue. Application components contain the components which form an application stack. These components form the basis of an application which is generally the basis and backbone of any business environment. Application components contain elements from all the modules of the application and they also contain various direct and indirect constituents like application dependencies, application nature, and type. All these integral details of an application are termed as invaluable as far as a migration project is concerned. With any of these details missing, application migration will go off limits and that will derail the entire migrations project including Infrastructure migration as well.

Database components are very tightly coupled with application components and are equally responsible for application

transactions. Database components are setup as per application requirements and are integrated within the application stack. Every activity being done within the application modules and every transaction is being recorded with in databases instance tables. It is the database tables that allows hundreds of application users from both internal LAN and from Internet to transact at the same time with the application and still manages to maintain application and data integrity. An application stack or an environment cannot function without a database and its associated components. A database is a must for an application to function.

Along with database, it is the testing components that are very tightly coupled with the application stack. Post application migrations, it is the testing that is being carried out on the application stacks and overall environment to find out if application environments have developed any defects. It is termed as critical to check out the migrating application and its entire environment for any defects. Any sort of anomalies being developed in the application environment will cause the application to throw errors and will impact overall business landscape. Testing environments are always considered automatically for migrations along with application migrations. In some cases, testing environments are build afresh at the target location.

Application tools are used for making application transactions and activities easy. Most of the application tools come as a part of the application package and their main job is to simplify complex application activities during migrations, application tools, and their associated servers are considered automatically for migrations as they are considered an integral part of the application stack.

Migration life cycle

A migrations project has a full-scale life cycle. The migrations project life cycle is meant to contain and possess set of activities that are carried throughout the overall journey. This migrations life cycle journey comprises of various facets such as:

Phases

Resources

Planning

Discovery

Documentation

Tools

Applications

Databases

Testing

Assessments

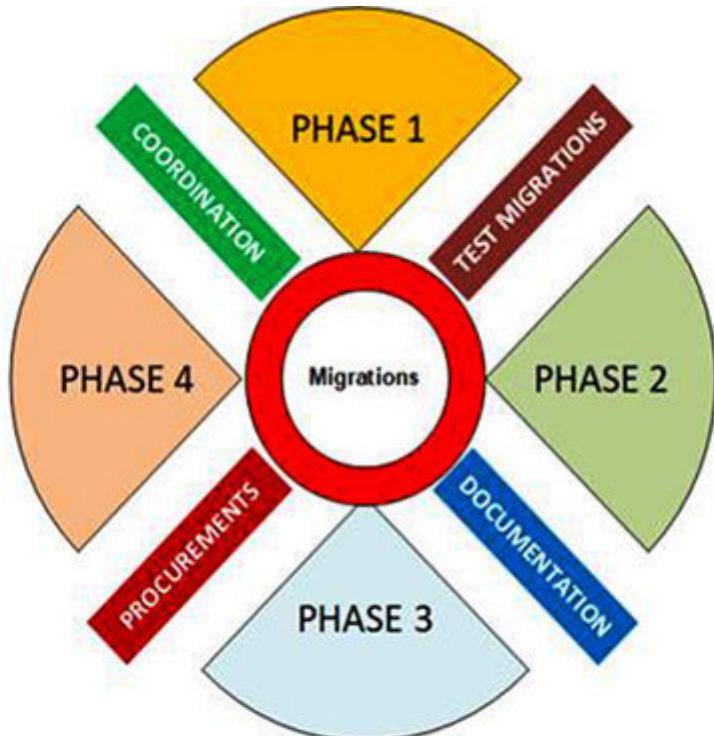
Approach

Strategy

Handover

Vendors

A migrations project life cycle is full of complexities throughout all of its phases. It starts with a planning and discovery phase and ends with a phase wherein Handover is given to the operations team. In between, this life cycle consists of many other phases that includes not only assessing the entire environment but also working on a strategy, coordinating with vendors, doing documentation, and even preparing test environments. Not to mention that is not all, then rather most of these tasks are interdependent on each other and overlap during different phases:



Migrations Life-Cycle

Figure 9.2: Migrations life cycle sample

Application and testing also come into the picture for a migration life cycle as infrastructure technology migrations are fuelled and driven by business motives. As stated earlier, every migration is basically an application migration and the infrastructure associated with that application(s). Application migrations makes a baseline for migration life cycle and paves the way for a solid foundation which is attributed with lots of planning, discovery, assessment, tooling, and testing. All the components are analyzed against the baseline and are added or removed in the theme of migrations life cycle.

The entire migrations life cycle revolves around a document named as migration plan which is like the bible of a migration

life cycle. It contains:

Each and every step and associated activity

Responsibility and roles

Technicality of the things

Processes to be adhered to

What to procure

What and how to build

What, when, and how to migrate

What quantity to migrate

Which vendor

Which tool

Which location

What and how to Rollback

Who to approve

Whom to speak

Who is the owner

How and what to report

What tests to be done

What to support

What documents to create

Whom to handover

And much more. Migration life cycle and its strategy along with its approach and framework are fabricated inside a migration plan document. This document serves as the epitome of a migration life cycle:

Migrations life cycles are long and lengthy and contain questions like:

If an environment is going to be just migrated

If an environment is going to be upgraded and then migrated

If an environment is going to re-platform and then migrated

If a new environment will be laid out and then apps will be migrated

All these particulars set the base for a migrations project and its associated life cycle that is further segregated into various phases for the ease of implementation and achieving required goals.

Migration process flow

Migrations project life cycle contains a complete process work flow. Information and the set of activities that are executed and carried out during a migration life cycle are depicted through a process flow-chart. This process flow chart portrays:

Complete set of activities

Migration phases

The order in which tasks and phases appear

How one activity impacts the other one

A process flow chart is an Industry standard practice of representing the entire work flow during a full-fledged life cycle. A migrations project involves a lot of hands and teams. There are a lot of vendors as well who are involved in the migrations planning and execution process. Then there are client SPOCs as well to coordinate and discuss things with.

With all these stakeholders, it is at times near impossible to memorize things and execute in the desired order. And a miss of even a single activity across any phase will jeopardize the entire migrations project.

Thus, process flowcharts are used in this situation to make things simple. A process flow-chart is intended to make it easy to understand and adapt to the upcoming requirements. It also makes it clear for every engaged stakeholder as to by when they need to perform their respective set of activities and with whom they have to coordinate and with whom they have to hand it over post completion.

Process flowcharts also illustrate the process maturity of the vendor organization in front of the client. It showcases the capability of the vendor organization in handling complex projects and associated environments. Process flowcharts also reflect the prowess of a vendor to fabricate and standardize a process:

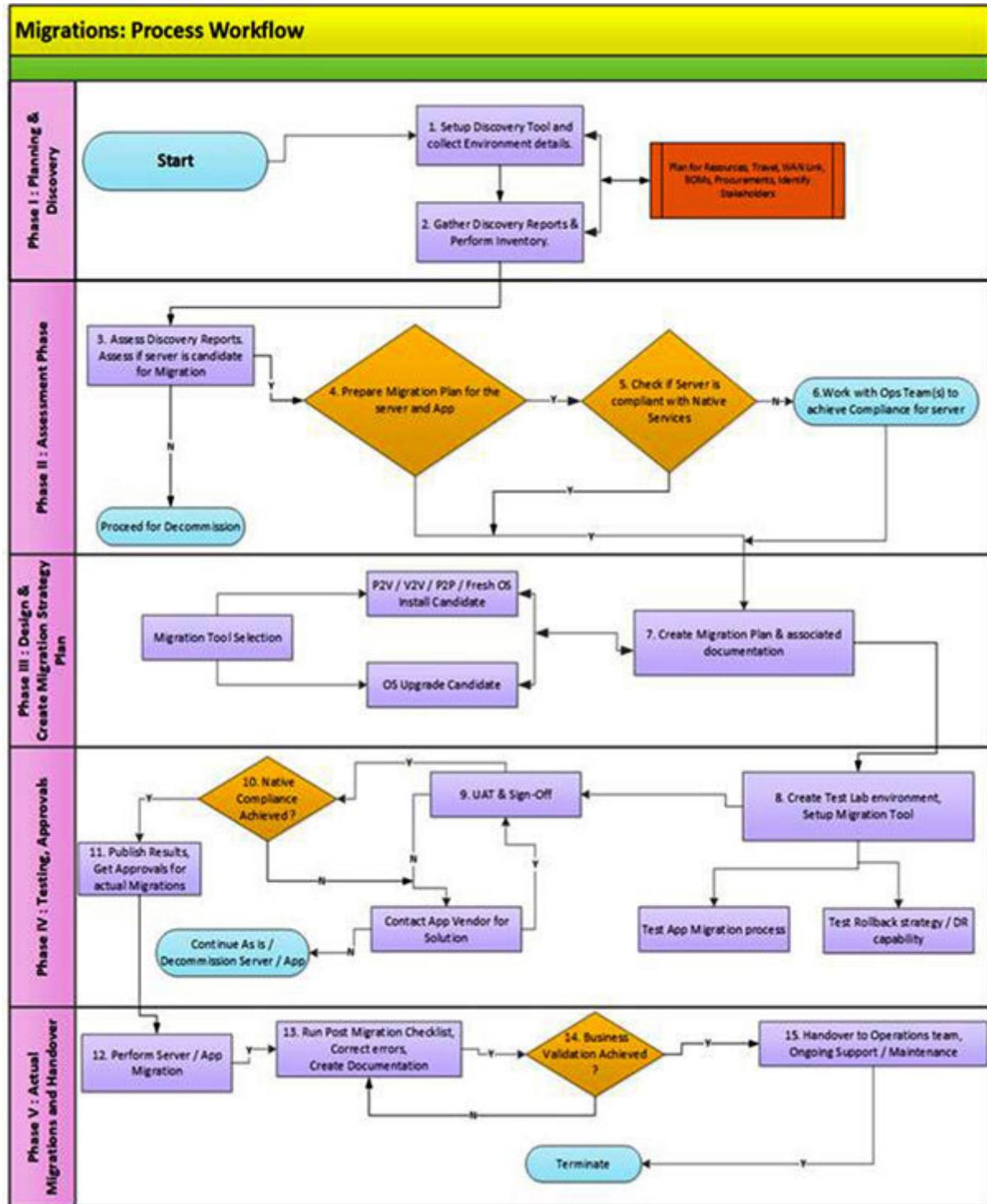


Figure 9.3: Migrations process flowchart

A process flowchart also helps the teams which are working on the ground in many ways. Such a flowchart helps to develop an understanding amongst all the teams and their respective

members. In a single glance, a clear picture of the entire process is visualized and grasped by the teams. This visualization helps to keep the memoirs alive till the project is going on.

A process flowchart also brings everyone on the same page which is a must from a long migration project's perspective. It removes any sort of ambiguities with respect to the:

Overall process

Number of phases

Timelines

Roles

Responsibilities

Performing teams

Stakeholders

Decisions and decision-makers

Start and end of a phase

Inputs and outputs for each activity

Since a process flowchart develops a clear understanding amongst all stakeholders, it provides ample time for everyone to think, plan and prepare as per their role which helps to get the best efficiency out of every performing member.

At the same time, every team member working on the project is aware of all the other teams working on the project and their associated roles and activities. Thus, a process flowchart also helps to bring out the best of the coordination amongst all the teams.

This coordination helps to resolve any pending or incurring issues as well. Due to the overall clarity about the process and roles and responsibilities, it helps to craft faultless and precise documentation which is vital for the overall project.

Let us go through all the phases depicted in Figure. We will be listing detailed list of activities that are carried out in all of these phases (at a generic level). Let us explore:

Migration Phase 1

The first phase of a migrations project life cycle is the kick-off phase. This is the phase of planning and discovery which are done in parallel to each other.

In planning, vendor organization lists out a few key lists of activities.

Here are some of the prominent planning phase activities:

Planning phase:

SOW finalization and signoff

Resource identification

Tools identification

Project kickoff

Requirements templates

Questionnaires

Travels

Request for access to existing documentation on the HLD, LLD, Architecture

Asset list – Inventory, Model and version number, device configuration.

Access for configuration review (Read-Only)

Facilitate for remote VPN as required

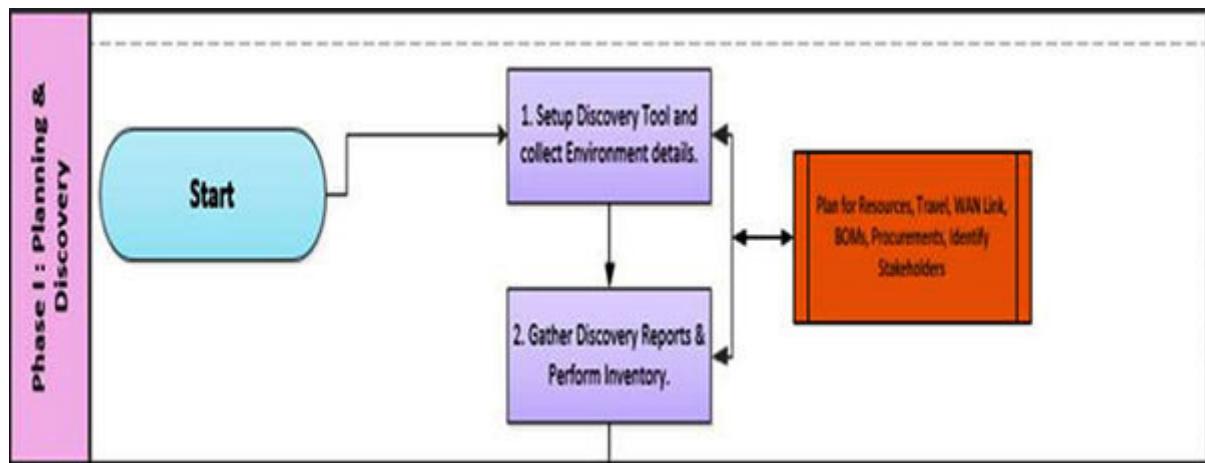


Figure 9.4: Migrations Phase I

At the same time, vendor organizations start with the Discovery process. Here are some of the activities of this phase.

Discovery

Discovery tool setup

Coordination with client IT Team for creating servers

Change request in process

Providing servers specifications

Creating SOP for tools setup

Coordinating network, storage, and firewall requirements

Coordinating access provision

Validating all specifications including access

Installation of discovery tool on the servers

Configuration of the discovery tool

Validating tool access on the environment

Making dummy server discoveries

Making dummy subnet discoveries

Publishing test reports

Initiate final discovery for the entire environment

Monitoring the discovery

Error checking

Checking the report in generation

Validating if inputs for Infra, App, DB, and other assets are getting populated

Infrastructure - assets mapping initiation

Discovery reports generation check

Validate data accuracy and perform cleansing

Identification - client processes

Change requests

Communication to all stakeholders

Off business hours

Finalize email template

Status report templates

Bridge call details

POC details

Vendor details

Customer SPOCS identification

Key stakeholders identification

Resources

Profiles collection

Profiles scanning

Interviews

Team selection

List submission for access

CAB process

Identification of CAB members - client

Identification of CAB members –vendor organization

Identification of CAB frequency

CAB duration and timings

Additions into CAB from vendor organization

WAN Link procurement

Prepare BoM for P2P Link procurement

Send BoM for client review and approval

Send BoM to vendor post client approval

Applications discovery

App groups

App inventory

App owners

App business units identification

BU heads identification

Migration Phase 2

The phase which follows Planning and Discovery Phase is the Assessment Phase. This phase focusses on the assessment of the discovery reports collected in the first phase:

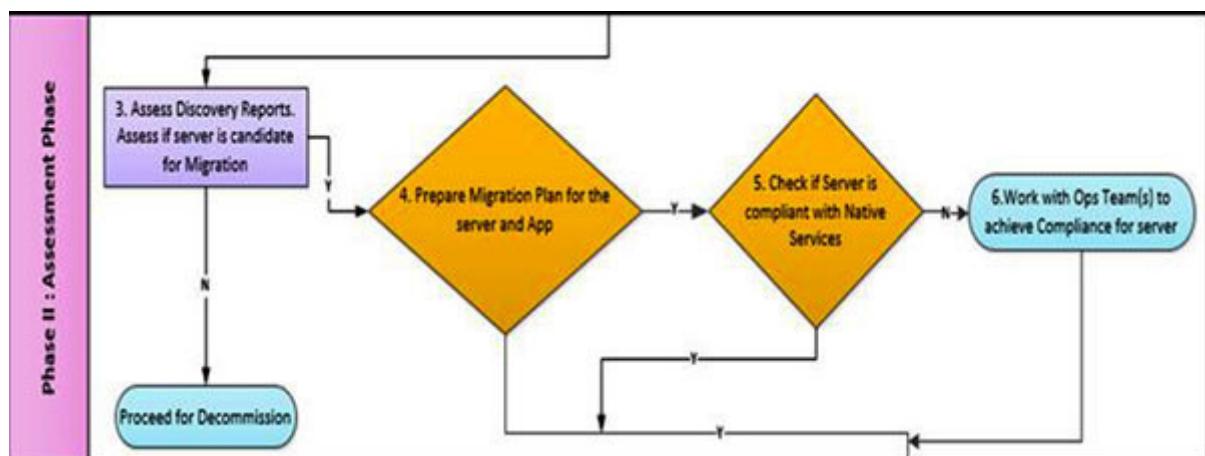


Figure 9.5: Migrations Phase II

Here are some of the activities which are being done in this phase:

Application

Application owner/BU identification

Application support identification

Application landscape identification

Application sequencing/prioritization

Creation of application Stacks

Defining application dependencies

App details

Identify App feasibility for migrations

Identify App roadmap

Identify App resource requirements

Identify App future roadmap

Identify App vendor support

Application type

Application modules

Infra

Discovery reports assessment

Infrastructure - application mapping

Identify App infrastructure requirements

Identify App upgrade path and challenges

Operational load at backend/front end load

Number of concurrent users/requests

Number of concurrent requests

Base load and peak load week/month

Number of processes and instances of processes

Log size

Archival requirements

Persistence requirements

Base recommendations from vendor

Installation requirements

Test results and extrapolation

Interface architecture and performance tuning

Architecture resilience

Load factor loss

Legacy interfacing loss/overheads

Complexity of events and mapping

Factor of safety

App transactional load

App newer versions check

App version to be upgraded

Concurrent sessions and processes

App environments required to be build

Attachments to be uploaded and their size

Internal and internet users

IOPS requirement

Storage requirement

Processing requirements

I/O operations

Infra requirements

OS details

IP ranges

vLAN requirements

Natted IPs

Host name

Firewall ports (incoming/outgoing)

Load balancer dependencies

Identify hardware age

Network bandwidth and latency

Network/transmission losses

Database server requirements

Network performance and bandwidth

Migration from On-Prem required

Any servers to be decommissioned

App compatibility with virtual environments

App compatibility with cloud environments

Specific network requirements

Amount of data to be uploaded onto cloud

ADS Integrated

Any legacy environment or modules within app

Backup requirements

Checks

Current storage capacity available

Storage in use/reserved

Storage data classification (File/DB)

No. of racks available

No. of "U" availability in available racks

No. of LAN/WAN ports availability

Current network bandwidth

Structured/labelled cabling

Rack elevation charts availability

DC architectures availability

Available documentation

DB Versions

In-Flight projects details

Datacenters

Datacenter/DC room understanding

Identify the racks/cage/cabinets location

Identify the underlying servers/storage for migration

Understanding of racks/cage/cabinets to update

Identifying the equipment's in physical servers/virtual/storages

Validating the equipment's with quantity compute as per the CMDB report

Cross verify the labels location of compute (servers, storages), if any mismatch updating document.

Identifying and validating storage/SAN switches connected.

Marking the position in documents and updating reports.

HW equipment's

Identify existing hardware for servers, storage, network switch (related to storages), and so on

Validations:

Validating the position of the equipment in Racks/Cabinet as per CMDB

Validating Serial no's/Hostnames/IP Address/Model No

Verifying Serial no's/Hostnames/IP Address / Model No

Marking the Serial no's/Hostnames/IP Address/Model No/Make (OEM)

Updating documents serial no's/Hostnames/IP Address/Model No/Make/Location for report and architectural diagram

Validating the server/storage/backup/network, and so on, which are under scope

Storage and backup only up to total capacity validation

Server validation only for physical servers

Virtual servers count allocated per physical servers, configuration

Operating system details

Current patch status

Identifying the database servers/databases.

Identifying any databases clusters in use.

Infra mapping to application/databases

Identifying the application for migration

Identifying the application databases

Discussion with stakeholders from client organization for infra details for mapping

Validating infra identify and mapping with applications

Verifying jointly with team mapping landscape

Verifying existing documents for analysis

Existing network infra assessment (subnets, VLANs, Zoning)

IP schema

AD landscape

Common infra services - NTP, SMTP and DNS, and so on

Application assessments

Assessment of overall test strategy for application

Identify scenarios for As-Is and post-migration testing

Assessment for test automation improvement

Test design and development (Test cases for As-Is testing and post-migration testing)

Network

Existing DC network - switching and routing

Existing devices configuration

Existing connectivity to WAN and Internet

Existing VPN connections

Rack, space, and cooling availability in the existing DCs

Connectivity to the communication racks

Existing DCI link and configurations

Existing subnets, VLANs, Firewall Zones

Migration Phase 3

The third phase of a Migrations project is designing and creating migrations strategy plan. This is a lengthy and complex phase:

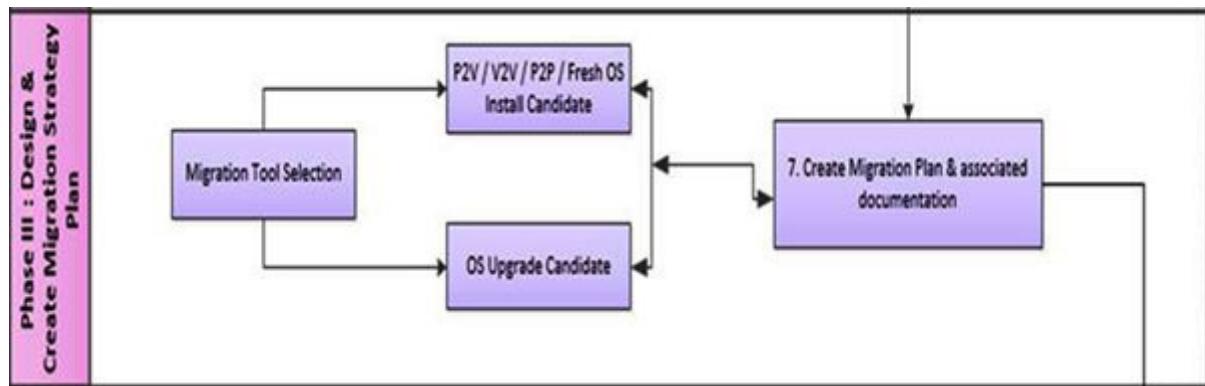


Figure 9.6: Migrations Phase III

Here is the list of all the activities carried out in this phase:

Move groups

Strategy

Conduct workshops with App Owners/DBs

Discuss migration timelines/path

Identify identical application

Identify application stacks for migrations

Divide application stacks in different groups

Develop app groups and waves

Collaborate with client teams

Receive inputs from all stakeholders

Discussions with key stakeholders

Finalize move groups for each application

Prepare high-level App move schedule

Prepare move group schedule

Application sequencing

Application prioritization

Review move groups with clients

Review move groups with all stakeholders

Review move groups with all vendors

Receive client approval for move groups

Publish move groups to all the teams

Mapping for application groups to New DC Infrastructure

Identify the stakeholders/teams required for application migration

Application components/architecture reconciliation/review

Identify timelines for move group

Discuss timelines feasibility internally with all the performing teams

Discuss timelines feasibility with vendors (If Any)

Discuss timelines with Business Unit Heads/Admins

Take business consent for migration timelines

Develop the high-level design deliverables/Cook books

Develop detailed build and deployment scripts for all applications

Develop check lists and cook books and communication plans

Submit the documentations for review and approvals and base line the plan

Sign off of the migration strategy and detailed move group plans

Migration tool finalization:

Migration tool

Migration tool identification

Migration tool feasibility

Migration tool compatibility with existing environments

Migration tool discussion

Vendor presentation for tool

Analyze number of licenses required

Analyze approximate cost required

Receive client feedback and inputs on migration tool

Finalize migration tool

Rack elevations

Rack positions

Analyze space available in current racks

Validate rack power redundancy

Validate rack connectivity to L2 switches

Availability of labelled cabling

Rack locks working

BOMs and documentation:

Prepare bill of material for hardware procurement for compute
(servers/racks)

Prepare bill of material for software procurement for compute

Prepare bill of material for hardware procurement for storage
(devices)

Prepare bill of material for software procurement for storage

Prepare bill of material for hardware procurement for the network
(devices/racks)

Prepare bill of material for software procurement for the network

Documentation for server installation

Documentation for server build and configuration

Documentation for network devices installation

Documentation for network devices build and configuration

Documentation for storage devices installation

Documentation for storage devices build and configuration

Prepare end to end migration plan

Prepare test environment build and execution plan

Landing zone design

App migration designs

Solution design for compute tower

Solution design for network tower

Solution design for storage tower

Industry best practices and recommendations

Client inputs incorporation

Vendor inputs incorporation

Preparation of high-level design documents

Preparation of low-level design documents

Existing DC network and new DC network integration design

Network devices POST tests

Basic connectivity testing between the equipment and servers and storage

Verify the integration with the existing DC network

Verify the reachability of the network equipment over the WAN and the Internet

Verify various firewall zones and policies

Verify LB policies

Test resiliency of core switches

Test resiliency of firewalls

Test resiliency of the load balancers

Test migration link connectivity

Test migration link throughput

Test the resiliency of migration links

Conduct workshops with customer's network and security teams

Understand the constraints and dependencies

Understand the security requirements

Understand the IP addressing requirements

Understand the VLANs/firewall zoning requirements

Understand the security and network-related applications hosted in the USA and India

Strategize the migration of these applications

Service management activities

Migration Phase 4

The fourth phase of a migrations life cycle is for testing and subsequent approvals:

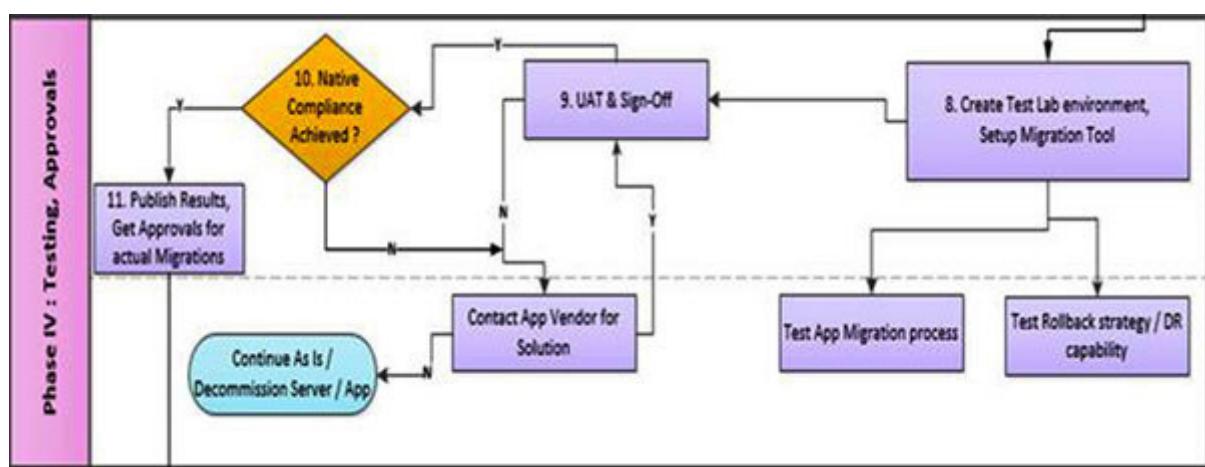


Figure 9.7: Migrations Phase IV

Here is the list of activities being performed in this phase:

Test environment setup:

Test environment preparation

Identify timelines for POCs execution

Identify number of servers to be setup

Identify storage requirements

Identify IP ranges and vLANs

Identify firewall ports

Identify host names

Identify ADS users to be created

Identify level of access required for the execution

Identify test data to be placed on test servers

Identify if trial version of migration tool is available

Identify if trial version of application is available

Identify source location or environment

Identify target location or environment

Get network bandwidth allocation

Identify load balancer to be used

Identify client stakeholders/vendors

Identify performing Teams like:

Compute Team

Storage Team

Network Team

Firewall Team

Monitoring Team

Application Team

Database Team

Tools Team

Client SPOCs

Vendors (as applicable)

Test plan/documentation

Create test plan documentation

Create/use pre-migration checklist

Create/use post-migration checklist

List out detailed technical steps as per the migrating environment platform and Tool to be used for migrations

Review test plan internally and modify as required

Share test plan with client stakeholders

Receive client sign-off on test plan

Prepare rollback strategy

Add rollback strategy to test plan

List out detailed steps

Change requests

Raise change requests with appropriate details

Represent the change in the CAB Meeting

Get change approved

Inform all Stakeholders about CR Approval and proceeding with test migrations

Testing environment creation

Setup minimum two versions of a single OS

Use client images/processes for server build

Run server build checklist

Move test data onto test servers

Setup migration tool on test servers

Configure migration tool - as per vendor recommendation

Install test version of application

Finish configurations of test app

Validate test servers with DNS

Validate test server reachability and availability

ADS users creation

Authorized access Storage allocation

Test migrations execution:

Test migrations initiation

Start 30 Mts prior to the test migrations

Run pre-migration checklist

Find and resolve any defects/errors

Inform all performing teams to be ready

Send out communication about commencement of test migration

Initiate test migrations using the migration tool

Use test plan documentation

Capture network packets and drops if any

Get monitoring alerts - status updates

Monitor entire test migration

Capture every sequence and phase

Create artefacts

Complete test migrations

Shut down test servers at source location

Validate if test servers are able to start at target location

Analyze test migration results

Validate data movement

Run post-migration checklist

Verify storage transactions

Verify IOPS during migrations

Verify network transmission and any losses

Verify any monitoring errors

Verify OS-based errors

Verify any errors/logs of migration tool

Reports

Prepare detailed analysis report in appropriate template

Add test volumetrics and parameters

Publish test migration results to all Stakeholders

Furnish test migration evidence

Updated change request with test migration results

Close change request

Request for client approvals to initiate actual cutover

Receive client sign-off on test migration results approvals

Migration Phase 5

The final phase of migration project life cycle is for actual migrations and then handover to operations teams. Here is the list of activities that are being carried out in this mammoth phase:

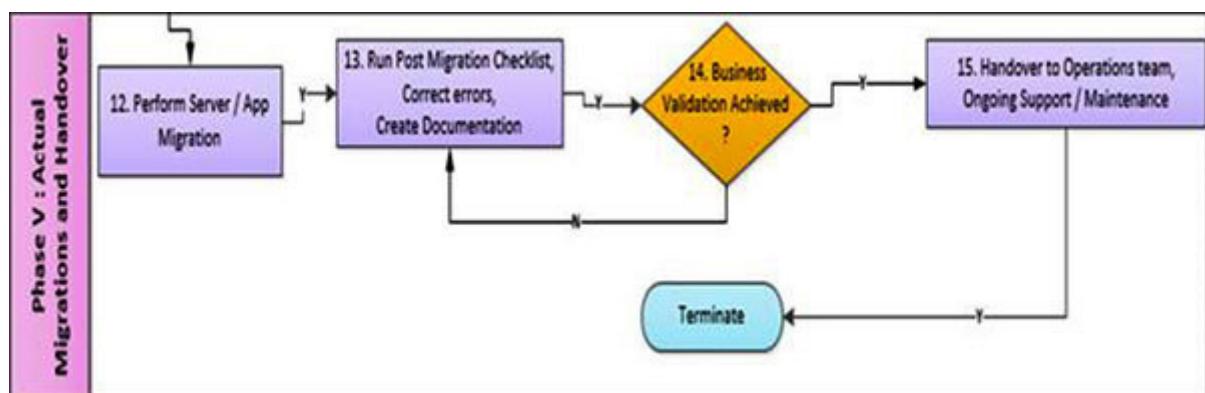


Figure 9.8: Migrations Phase V

Actual migration preparation:

Pre-migration activities

Identify application landscape

Identify the total number of App servers

Identify total no. of DB servers

Identify total no. of web servers

Identify servers hosted in DMZ zone

Identify load balancers

Identify application internet URL

Identify application intranet URL

Identify application DR requirements

Check last DR test reports

Identify application and DB YOY growth rate

Identify application critical modules

Identify application critical users

Identify ADS groups for application

Identify static DNS entries for application

Check specific host file entries

Verify ADS users for application

Verify any zone entries in DNS for application

Identifications:

Identify total storage

Identify IP Range and vLANs assigned

Identify Natted IP/Public IPs

Identify number of internal users

Identify number of concurrent users

Identify number of external users

Identify SPOC from app team

Identify SPOC from client team

Identify SPOC from vendor team

Identify mechanism to establish a communication channel

Identify templates for reports

Identify templates for status updates

Identify frequency for reports/updates

Identify SM team for sending updates

Identify PMO team for running bridge calls

Identify timings for running bridge calls

Identify email templates for bridge call invites

Identify email template for sending communication

Identify stakeholders to be part of emails

Review total set of servers

Identify dependencies

Identify dependency matrix of all servers

Identify and split servers into various move groups

Validate if all move groups can function independently

Discuss and review with client on move groups data

Discuss and review with vendor (if any) on move groups data

Find feasible dates for application migration

Review internally on migration dates

Check with client on migration dates

Receive client sign-off on migration dates

Check for vendor availability for migration dates (If any)

Inform all internal/external stakeholders about upcoming migrations

Inform all users about upcoming migrations

Discuss migration dates with infra/migration team

Inform monitoring team about migrations

Inform network team about migrations

Inform storage team about migrations

Inform security team about migrations

Inform DB team about migrations

Provide inputs about move groups

Provide inputs about migration dates

Coordinate with infra teams for migrations

Ensure all the servers are completely backed up

Ensure all the servers are up to with new patches

Ensure all the servers are up to date with Anti-Virus

Sprint planning

Sprint Review/Retrospective

Restart any server in a maintenance window if it is sluggish

Validate if target platform is compatible with application

Check for feasibility of application upgrade (if required)

Document all validations

Initiate processes required to execute migrations

Discuss with infra team on the total time for actual migration

Addition of 1 Hr. of Buffer time total projected duration for migrations

Prepare rollback strategy

Discuss with infra team and select a rollback mechanism

Prepare the rollback strategy in case of NO-GO

Document all the steps to be executed

Initiate a change request process

Provide all details in the change request

Represent the change request in the CAB meeting

Receive CR approval from the CAB

Inform all the stakeholders about change approval/status

Inform and confirm timings and dates to all Infra Teams

Inform and confirm timings and dates to vendor (if required)

Coordinate with infra team for setting up landing zone

Provide required specifications for app environment to infra teams

Coordinate with infra teams to work on the specifications on target location

Validate each Infra component post completion (network, security, storage. servers)

Validate that each member of app team has required access

Arrange access for vendor access if required

Raise a CR for vendor access if required

Ensure deployment tools are installed

Initiate a test migration and capture results

Conduct POCs of sample app components on new DC

Build and deployment activities as part of application move groups

Co-ordinate the application defects remediation with client product teams

Build and deployment activities as part of application move groups after remediation

Build and deployment activities as part of application move groups after remediation

Feasibility of parallel runs and delta migrations (data updates, user data base, and so on)

Regulatory and compliance sign Off of new Environment and Applications

Prepare documentation for migrations

Prepare SOPs/MOPs

Prepare migration checklists (Pre and Post)

Prepare migration plan document

Share and review documentation with client

Revise the documentation to include the POC results

Receive sign-off from client on the target location preparation

Communications:

Inform all the stakeholders about Landing Zone preparation

Inform all the Vendors about Landing Zone preparation

Inform all the participants and teams about Landing Zone preparation

Inform all about proceeding with Migrations

Update change request

Update change requests with appropriate details for Production
GO LIVE

Initiate Audio Calls with all the teams to explain migrations deeply

Explain R&R clearly to all the team members

Receive Sign-Off

Send email confirmation to all the team members

Complete Readiness Checks

Actual Cutover initiation

Initiate proceedings 45 Mts prior to actual migration

Send out email reminder to all users about Migrations and cutover

Send out email reminder to all teams about Migrations and cutover

Send out email reminder to client about Migrations and cutover

Send out email reminder to Vendors (if any) about Migrations and cutover

Ask all required stakeholders/Teams to join bridge calls

Ensure that all participating teams are available on the bridge

Ensure that all migrating servers are completely backed up

Receive confirmation that servers are up to with Backup

Ensure that there are min. 2 copies of backup available

Ensure that servers were updated with latest patches

Ensure that servers are not in hang state

Initiate stopping of Application Modules one by one 10 Mts prior to Migration

Run Application based pre-migration checklist on all the migrating servers

Confirm Infra Teams about Application Shut down

Database team to initiate shutting down of DBs

Database team to confirm shutting down of DBs to Infra & App teams

Infra Teams to validate server availability and app shut down

Infra teams to run Infra Based Pre-Migration checklists

Infra Team to initiate Migrations for nominated servers in batches

Infra team to send communication about commencement of migrations

Infra Team to monitor the migrations

Network Teams to monitor Network Transmission for migrations

Monitoring teams to assess any errors

Infra teams to capture the migrations

Infra Teams to send regular updates about Ongoing Migrations

Actual Cutover initiation

Test Data Management and Test Environment Readiness

Test Executions

Test Planning

Assessment for Automation testing enhancement

User Acceptance Testing of the upgraded Application

Post-migration:

Post-migration validations

Infra team to validate migrations post completion

Infra team to validate servers post completion

Infra team to validate storage and data migration post completion

Infra team to run post-migration checklist on the migrating servers

Infra team to send validation

Infra team to send migration completion confirmation to app team and database team

Infra team to inform on the bridge call about migration completion

Infra Team to inform on the bridge call about infra validations completion

Database team to restart all DB instances on the DB servers one by one

Database team to confirm DB start to Infra and App teams

App team SPOCs to validate the migrating servers

App team to validate access on the server

App team to validate network connectivity on the server

App team to validate storage and data integrity on the server

App team to initiate application modules one by one on the server

App team to access each application module/page one by one on the server

App team to validate each application module page and feature on the server

App team to validate if application is communicating on all the ports

App team to validate host names

App team to validate DNS entries

App team to validate ADS users

App team to validate access for all the users

App team to validate DB connectivity with application

App team to validate IP Address and vLANs

Infra and App Team to validate Load Balancer functionality

App team to confirm if the application is working fine post-migration

App team to send out an email communication about status of application migration

App team to send out an email to all the stakeholders about Application migration completion

App team to send out an email to all the Users about Application migration completion

App team to check with users about app accessibility

App team/infra team to handle any user issues

App team to confirm for App GO-LIVE

Handover to operational teams

Conclusion

This marks the conclusion of this chapter on migrations components, life cycle, its phases and process flowchart. We have understood about project life cycle and we have seen various phases as well. We have gone through each and every phase in details and we have walked across a complete migrations project life cycle. We have also seen detailed list of activities related to each migration phase.

In the next chapter, we will look at migrations milestones, its journey and list of documents.

Points to remember

Migration components consists of both infrastructure and application components

A migrations process flow chart depicts entire project life cycle

Each migrations phase consists of number of activities in a project life cycle

Life cycle phases can overlap each other and also depend on each other

Key terms

Life cycle – Migration Project Life Cycle

Phases – Migration Project Phases

Process – Migration Processes

Flow Chart – Process Flow-Chart

Stakeholders – Relevant People

Client – Customer

CHAPTER 10

The Migrations Project –Topology, Milestone, Legacy OS-Hardware and Documents

Introduction

In this chapter, we will read and understand about migration topology and its journey along with various milestones that are achieved during the entire life cycle. Post that, we will have a detailed look at overall migration approach where-in we will learn as to how approach a migration, especially for legacy hardware and legacy operating systems. Lastly, we will look at different migration documents which are used at different stages of a migration projects.

Happy reading dear readers!!!

Structure

In this chapter, we will cover the following topics:

Objectives

Migration topology

Migration milestones and journey

Migration overall approach

Migration approach for legacy hardware

Migration approach for legacy operating system

Migration documents

Conclusion

Points to remember

Key terms

Objectives

Our main objective in this chapter is to learn and understand about a migration project's:

Topology and its complete flow - throughout its phases

Various milestones and peek at overall journey

Overall approach and specific approach for legacy components

Detailed view of all the documentation

Migration topology

As we have learned that every migration life cycle is divided into various phases. All these phases form the overall topology of the overall project. Every vendor organization follows the same approach of splitting the entire project life cycle into various phases which is also an Industry standard for IT migrations. And all this is being done so as to implement an effective strategy and approach and to manage control.

Here is a look at the chronological order of migration phases and overall topology:

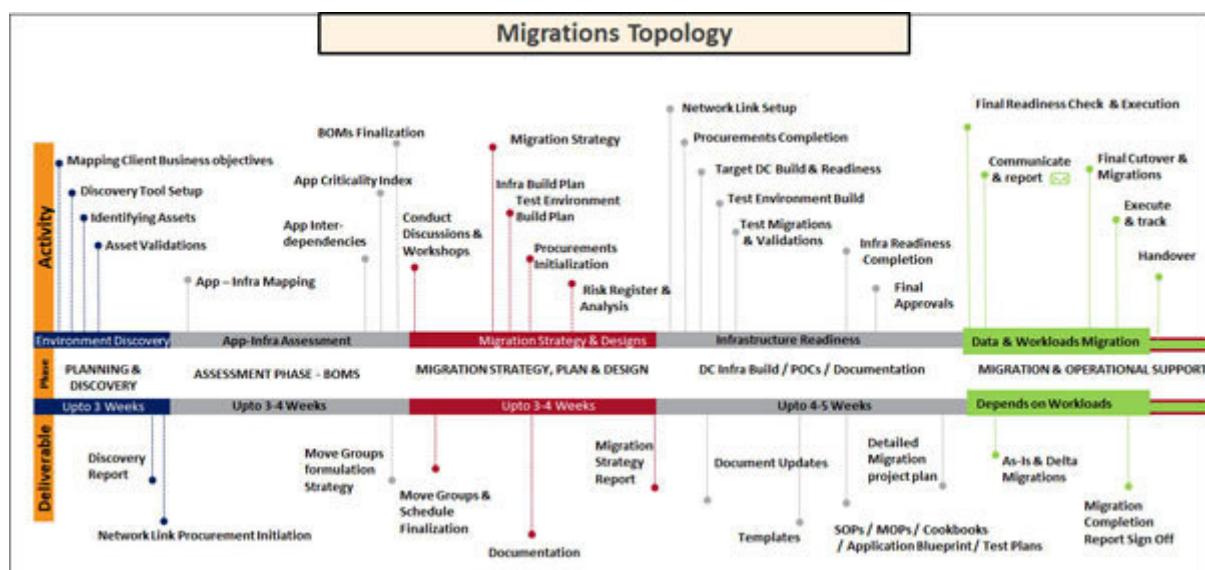


Figure 10.1: Migrations topology

Planning and discovery phase

Migrations project will commence with the planning and discovery phase. First things first, **Scope of Work** guidelines are drawn by both Vendor and client organization and things commence.

What follows is the initiation of network link connectivity between target and source DC locations which will be used for migrating servers over the network. A network link procurement usually takes about 12–16 weeks for procurement once ordered (minus any delays) and thus this particular step is initiated right at the start of the project.

This phase also focuses on the vendor organization's preparation for the upcoming migrations that will commence with the setting up of a discovery tool. Discovery tool will help to collect all the infrastructure-based details and all the application environment details along with databases information. The discovery tools are a mutually selected tool by both client and vendor organizations and go through a lot of feasibility, compatibility, and cost analysis before being selected.

With respect to planning, this phase gives about 2–3 weeks to the vendor organization to plan for the upcoming phases and tasks. Resources planning, client processes identification, collecting information about important client stakeholders such as BU heads or owners, app-DB admins, templates, and so on.

Assessment phase

A detailed assessment is the second phase of the migrations life cycle project. This phase will give a complete synopsis of the environment including application mappings and details. This phase begins with analysis of the discovery report collated in the discovery phase which results in the creation of an overall environment inventory report.

This assessment helps to identify critical applications and their associated workloads. This further helps to identify the critical inter-application dependencies and application stack formation. The same process is being repeated for all the applications sets and stacks. All these activities are done in coordination client assigned application and DB administrators cum owners. This paves the way for formulating a strategy as to how create move groups for migrations.

Another critical activity that is done in the phase is the finalization of **Bill of Material** for the procurement of both software and hardware items. BOMs usually take place post-infrastructure-application mapping and assessment which provides a holistic view of the overall requirements. Assessment of environment helps to assess the requirements for new equipment's/devices/software licenses and media and accordingly BOMs for respective towers are placed.

Migration strategy, planning, and design phase

The migration strategy, planning, and designing phase is the third and a very critical phase in the migration project life cycle. This phase begins with finalizing move groups for the actual cutovers via various discussions and workshops with application environment owners and admins. Along with move groups, schedule finalization of actual cutovers is also done in this very phase.

Parallelly, design, and build plan for infrastructure build at target locations (if it is part of the scope and is required) is being carried out. Build plan is also prepared for test environments that follow the infrastructure build. All the designs and build plans are prepared and sent for client review and sign-off.

Service management processes are also kicked-off in this very phase itself where in all the processes, such as change management, are being pursued in the client environment are followed by the vendor organization prior to performing any activities. All the relevant and compliance-based guidelines (ITIL, GDPR) are also adhered by vendor organization while preparing design and build plans.

Documentation is another critical activity that gets started in this very phase. Post all the workshops and discussions with Application/DB/Infra admins and owners, vendor organization

SMEs kick start documentation for their respective tracks. This includes preparing SOPs/MOPs/application cookbooks, application blueprints, test environment plans, configuration documents, build plans, Rollback plans, and the very critical migration plan document.

Infra build/test migrations - POCs phase

This phase emphasizes on building new infrastructure components in the target DCs. Infrastructure build activities are done so as to consolidate and accommodate incoming Servers during actual migrations. If there is more than a single DC to be build, then the same set of build activities are performed in each of them.

All the build work is done on the basis of build plans and SOPs created during the strategy phase.

Following the infrastructure build activities, test environment setup and executing test migrations activities are carried out. Test migrations are vital to actual migrations as they assess the platform readiness.

Once test migrations are conducted, their results are prepared and shared with all the stakeholders. Final decisions for the actual migrations are taken on the basis of how test migrations have fared. If there are any issues or errors with test migrations (as shown by results), they are sorted out with the highest priority. Test migrations are carried out again and again until desired results and infrastructure readiness is achieved.

Once desired results are achieved, test migration results are sent for final sign-off and approvals from client for a go-ahead for actual cutovers.

This phase also sees the completion of all the documents which includes migration plan document. This document acts as the bible of the entire migrations project and contains of very detailed steps for overall migrations including all technical steps/procedural steps/stakeholders/schedules and timings for each move groups/migration tool and reporting mechanisms. Migration plan document also contains Rollback strategy and plan in case things go south during actual cutovers. All these plans and documents are reviewed and signed-off by the client.

[*Actual cutovers/migrations phase*](#)

This is the phase wherein actual migrations will take place. In this phase, plug is pulled from the application environment at the source location and migration commence as per the schedule and timings of the move groups. Depending on the results achieved during test migrations, platform readiness will be qualified accordingly.

All the migrations are done in the form of move groups only which were created and finalized during the strategy phase. All the move groups, as per their designated timelines will be migrated to their respective target location one by one. Post migrations, the migrations team will provide Hypercare support to the migrated servers to eke out any possible issues with the servers or with the user base. Servers are usually kept in Hypercare and are observed by the migration team.

Post migrations, the migration team will prepare and update the documentation for the just completed migrations. All the details related to the migrations such as timings, durations, application and environment details, platform details, server name, IP, and role and access details are thoroughly documented for future reference.

Handover phase

This is the final phase of a migration project life cycle for a workload. This is the phase of handover where in a successfully migrated server or servers are handed over to operations team. All the servers that are migrated are required to be supported on a daily basis and this type of support is provided by managed services team or operations team.

Operations teams provide daily routine operations support on all the servers which are migrated and are operational through the target DC location.

During the handover, the migration team will help operations team with all the documentation which was created during migrations process.

Handovers are done with the proper exchange of documentation related to the server(s) and application environment including stakeholders and access levels and details. Migration team has to ensure that operations team has required access to the environment prior to handover.

All the handover to operations teams are signed-off by the respective SMEs that mark the completion of the process and also marks the end of migration life cycle for a workload.

Migrations milestones and journey.

Each migration phase will have associated milestones throughout its journey. Milestones are actually the outcome or the output of migration phase. They are a mechanism to validate the quantum of completion of the project phase. If the desired output or milestone is not achieved, then project phase is deemed as failures.

It is important to mention here that each migration phase is associated with pre-defined timelines. Migration teams adhere to these timelines very religiously. Achieving the milestones within the stipulated timelines is crucial to get the project moving in the right direction.

If the milestones are achieved post the deadlines or the timelines, it is still deemed as the failure of the phase. Also, if the milestone is not achieved in the desired format, again the phase is tagged with failure or question marks.

Let us understand this with an example. Discovery report is the milestone of the discovery and planning phase which also happens to be the first phase of the entire life cycle. The discovery tool is setup and run by the vendor organization's migration team to fetch and collect client environment's details. The discovery tool is supposed to collect and then collate the collected information for servers/devices/equipment's, and so on, in a spreadsheet or any

other recognized format. However, it is very much possible that the discovery tool is unable to run and collect information.

There can be a number of reasons for this, such as:

Access issues

Wrong IP addresses and vLANs information

Configuration issues

Incomplete information feeded in tool

Network issues

There might be a lot many more reasons for a task to fail which will eventually mark the failure of the phase as the desired output which is **Discovery Report** is not achieved in this phase. This milestone quantifies the migration phase if it's a success or it needs to be revisited again.

Same way, each migration phase is measured with the outcome that is a milestone for that very phase. Here are the Milestones attached with each migration phase:

phase:
phase: phase: phase: phase: phase: phase:

phase: phase: phase: phase: phase: phase: phase:
phase: phase: phase: phase: phase: phase: phase:

phase: phase: phase: phase: phase: phase: phase:
phase: phase: phase: phase:

phase: phase: phase: phase: phase: phase: phase:
phase: phase: phase: phase: phase: phase: phase:

phase: phase: phase: phase: phase: phase:

phase: phase:

Table 10.1: Milestones

Here is a pictorial view of all the milestones tagged with each migration phase:

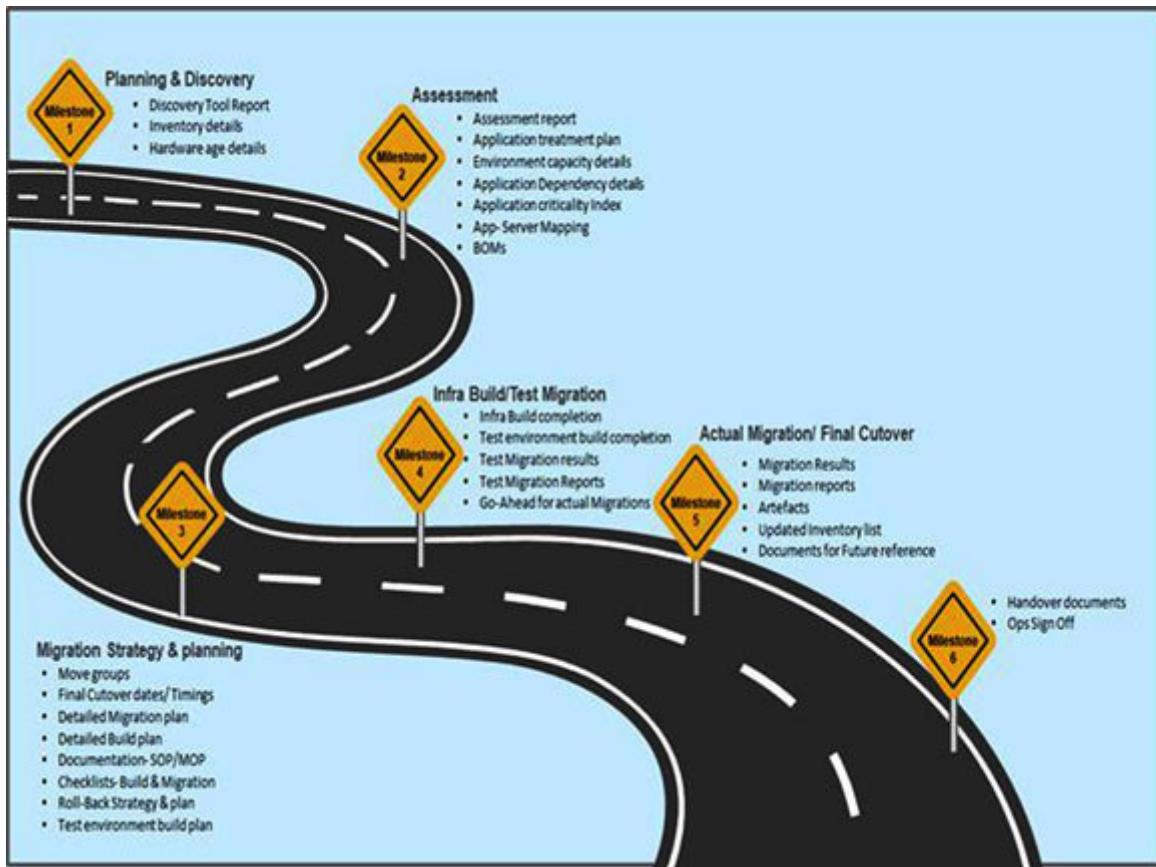


Figure 10.2: Milestones

Migration overall approach

Once move groups are created and finalized in the strategy phase, the vendor organization's migration team will start formulating an overall migration approach for all the workloads and environments. The migration team will work on some standard migration principles to devise an approach to execute effective migrations. These standard principles will include migration approach such as:

REHOST

REPLATFORM

REARCHITECT

RETAIN

RETIRE

Each and every workload will be assessed based on these principles and the final approach will be decided post-assessment and discussions with application admins and owners.

While deciding on the final migration approach for a server, the migration team has to consider the role of the server and the application environment of which it is part of. It is very much

possible that server's operating system is eligible for an upgrade but it is also possible that the recommended OS (post upgradation) may not be compatible with the overall application environment.

It is also possible that an application may require all of its servers to be upgraded together which may prove to be rather tedious or herculean task in some sense if it is a huge application environment. Exactly same principles are applied on the database section or workloads of the application environment. Again, OS of the database server might be eligible for an upgrade but the upgraded OS may not be compatible with the hosted DB version. Even the DB version may be ok for upgrade, but if it going to be compatible with running application post upgrade, that remains a dwelling question.

Amid all these scenarios and situations, the migration team will connect with the vendors of the application if it is a third-party application or with app and DB admins from the client organization if it is an in-house application. During this connect and discussion, they found the future plans of the applications and the possibility to upgrade the environments. Thereafter, they conduct a feasibility study of the application with all the mentioned stakeholders and find out as to which component of the application environment can be upgraded without making an impact on the environment.

They will make final decisions based on the feasibility study of all the components of the application environment and also after

enquiring about future plans for the applications. Some of the decisions are likely to be like this:

this:

this: this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this: this:

this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:

this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:

this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:

this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this:

this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this:

this: this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this: this:

this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this:

this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:
this: this: this: this: this: this: this: this: this: this:

Table 10.2: Approach

Migration approach for legacy hardware

Legacy hardware is one of the main reasons behind a migrations project. Most of the client environments contain a lot of legacy hardware. This is that kind of hardware which is been running for years and is out of support and considered obsolete. A hardware device generally comes with a warranty of 5 years and stays in the support cycle of the vendor for upto 10 years. After 10 years, it is considered as old and obsolete. It is recommended to replace an old hardware device as it is likely to fail at any given point of time and may cause downtime.

In some of the cases, the client is even found to run with hardware that is decades old and with no support mechanism for the running hardware or its hosted operating system. One good example of the same would be the case of SUNSPARCS systems. There is still a thick percentage of SUNSPARCS operating in different parts of the world, mainly in America and European regions, even though they went obsolete during the 1990s.

For a vendor organization, it becomes a huge challenge as to how to tackle the legacy hardware, especially when the client is insisting on retaining the hosted applications.

Legacy hardware servers are hard to imitate, replicate or even emulate. They run an operating system that is obsolete in every

possible sense and is not compatible with today's modern technologies such as virtualization technologies, cloud platform or DevOps containerization for that matter. Even the hardware is at its peak in terms of vulnerability that if you turn it down to move from one rack to another, there is no guarantee that it will power back up. And remember, there is no support available from any possible vendor and there are no spare parts also available in the market for such decade old hardware boxes.

In the nutshell, it becomes a tedious task to even formulate a strategy and a migration approach for these servers.

Here are some of the approaches which are adopted by migration teams to answer the question of legacy hardware.

Emulation software

Emulation software is used to migrate legacy environments running business-critical applications. Emulation software support PDP, VAX, Alpha, OpenVMS, Tru64, HP 3000, PA-RISC, and SunSPARC systems and they can work on very old legacy hardware devices as well.

Emulation software techniques are compatible and work very well with virtual platforms like VMware and Hyper-V and Cloud Platforms such AWS or Azure as well.

Emulation software acts like a hypervisor. It is installed after the server is setup with OS. Once emulation software is setup on the server, it then emulates the old legacy hardware server and creates an exact same copy of the server onto the new server. Thereafter, it continues to act as an intermediary between the OS and the emulated environment running on top of the environment:

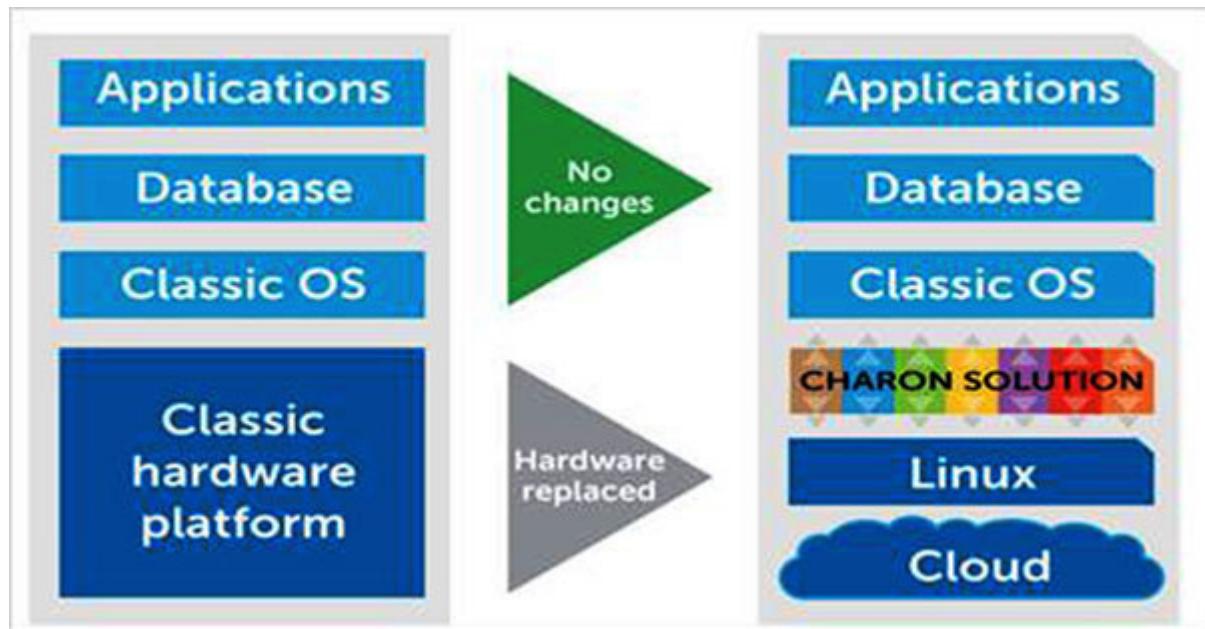


Figure 10.3: Emulation Migration

Here is how the architecture of emulation software will look like:

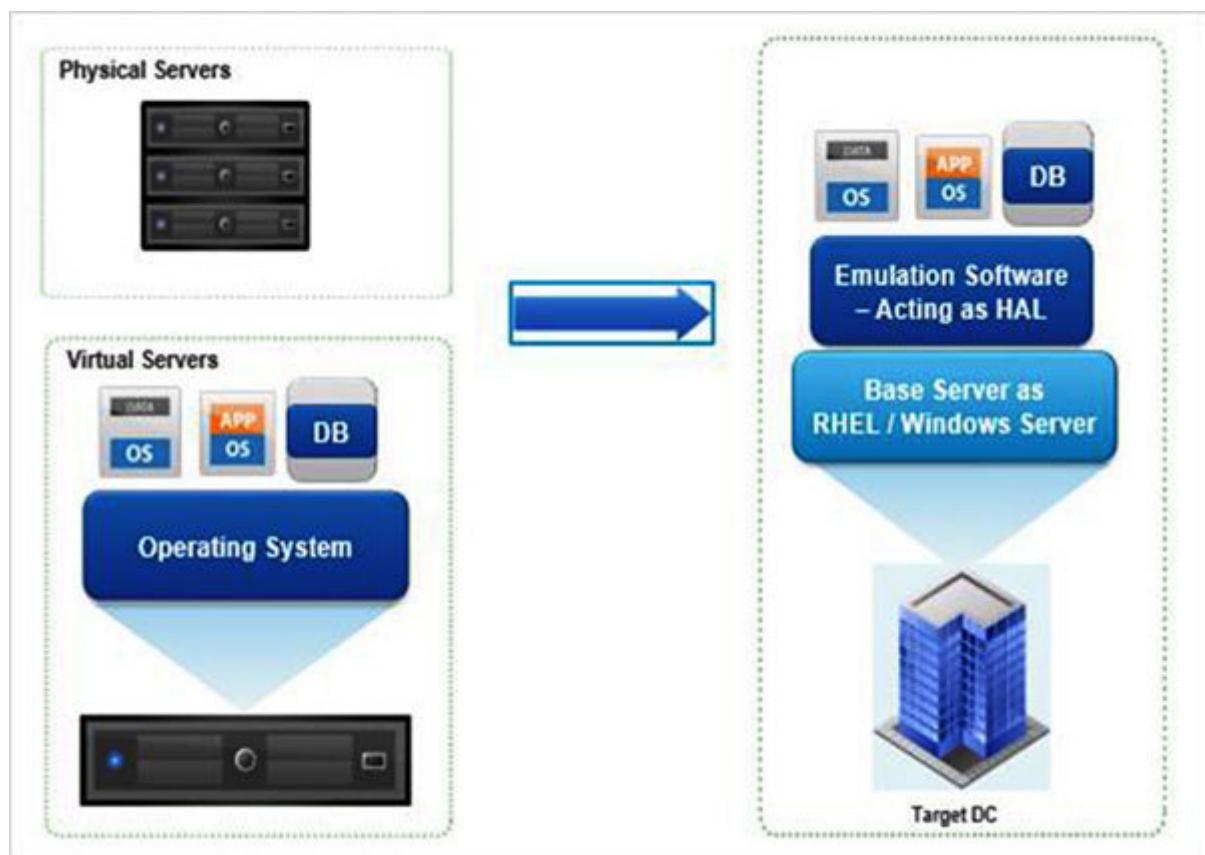


Figure 10.4: Emulation Architecture

Here are some of the tasks that will be carried out while executing and creating an emulated software migration:

migration:

migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration:

migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration:

migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:
migration: migration:

migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:

migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:

migration: migration: migration: migration: migration: migration:
migration: migration: migration: migration: migration: migration:
migration: migration: migration:

Table 10.3: Emulation software approach

Here are some of the benefits of using emulation migration approach:

approach:
approach: approach: approach: approach: approach: approach:
approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach:
approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach:
approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach:
approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach:
approach: approach: approach: approach: approach: approach: approach: approach:
approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach: approach:

Table 10.4: Emulation software benefits

Swing-kit migrations

Swing-kit migrations are similar to emulation software migrations, but they are carried out without any emulation software to create a replica of the environment. In this type of migration approach, data from a hardware device is copied onto another new device. This new device is then transferred to the target DC location and is installed there. All the data from the transferred device is copied on to the devices of the target DC devices.

Swing-kit migrations are rare as they generally involve downtime of the application or the overall environment. Also, they are expensive as they involve a third device kit that means extra cost even if the third kit being used for transferring the data is rental. Also, there is always the risk of losing the data on the way.

Here are the steps involved with swing-kit migration approach:

approach:

approach: approach: approach: approach: approach: approach:
approach: approach: approach: approach: approach: approach:
approach:

approach: approach: approach: approach: approach: approach:
approach: approach: approach: approach: approach: approach:

approach: approach: approach: approach: approach: approach:
approach: approach:

approach: approach: approach: approach: approach: approach:
approach: approach: approach:

approach: approach: approach: approach: approach: approach:
approach: approach:
approach: approach:

approach: approach: approach: approach: approach: approach:
approach: approach:

approach: approach: approach: approach: approach: approach:
approach: approach: approach: approach: approach:

approach: approach: approach: approach: approach: approach:
approach: approach: approach: approach: approach: approach:
approach: approach:

approach: approach: approach: approach: approach:

approach: approach: approach: approach: approach: approach:
approach: approach:

approach: approach: approach: approach: approach: approach:
approach: approach: approach:

approach: approach: approach: approach: approach: approach:

approach:	approach:	approach:	approach:	approach:	approach:
approach:	approach:	approach:	approach:	approach:	approach:

Table 10.5: Swing-kit approach

Fork-lift migrations

Fork-lift migration approach is considered only in extraordinary circumstances. In this migration approach, hardware devices and servers are put down and packed. They are then sent to the target DC wherein they are installed in the racks and are then powered on. This is how fork-lift migrations are done wherein hardware devices are physically moved from one location to another. There are both pros and cons of this particular migrations approach. Let us have a look at them:

them:

them: them: them:

them: them: them:

them: them: them:

them: them: them:

them: them:

them: them: them: them: them: them:

them: them: them:

Table 10.6: Fork-lift approach

approach

approach approach approach approach
approach approach
approach approach approach
approach approach approach approach
approach approach approach

approach approach approach approach approach approach
approach approach approach
approach approach approach approach approach approach approach
approach approach approach

Table 10.7: Fork-lift pros and cons

Migration approach for legacy OS

Along with legacy hardware, vendor organization and its migration team will also formulate an effective migration approach to solve the mystery of legacy operating systems. They will have to put in efforts and thoughts to come up with a strategy as to how to modernize the legacy operating systems. Generally, a vendor organization will use below mentioned principles to approach legacy operating systems:

Upgrade

Re-platform

Reuse:

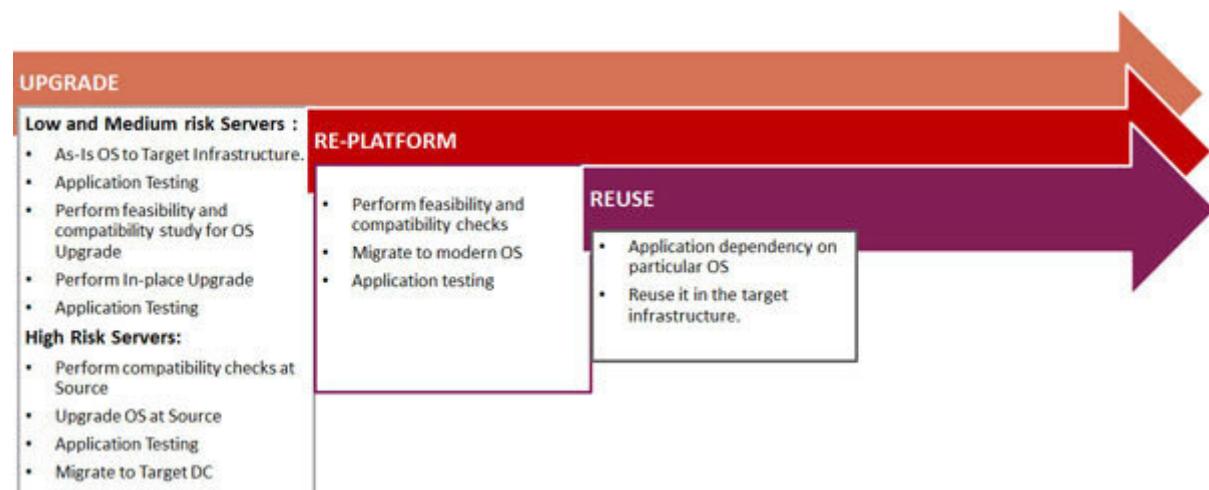


Figure 10.5: Legacy OS approach

Operating system upgrade

For operating system upgrades, vendor organization will assess the complete list of all operational servers during the assessment phase. All the servers running with legacy or obsolete operating systems will be earmarked in that phase itself. Thereafter, as mentioned earlier, migration team will hold discussions with all the application owners, database owners, third-party vendors to assess the overall application environment and overall feasibility of upgrading the operating system of the servers as per the application permit limit.

Any upgrades will have to be permitted by the application and database teams or administrators. Operating system that is going to be used post upgrade should be able to support the overall application environment and should be compatible with all the application stack components.

Prior to any upgrades, migration team will analyze and will find out if a server can be upgraded with new and latest operating system and the hosted application is compatible with the latest OS. Relative inputs from all relevant stakeholders will be considered during such assessments. The final assessment will be sent for client review for a final decision to be taken.

Along with the final review, vendor organization will also help to present risks and mitigations involved along with expected cost

projections. End decision will be taken by the client team and vendor organization will act accordingly as per client's directives.

In case of no compatibility, the server will be sent for customer review to proceed with decommission. In case of non-obsolete OS, the server will be reused for further operations.

Re-platform

In this approach, the migration team will look to re-platform the application environment and upgrade all the legacy operating systems among all its workloads if there is a feasibility. However, re-platform decisions are risky and expensive and thus are taken in accordance with senior management and on their directions.

Any possible upgrades with respect to operating system of any server of the environment are once again subject to the clearance from application and database owners and administrators. They will have to analyze if the recommended OS is going to work with other components of the environment.

There might be a few testings carried out as well by the application teams. However, once a green signal is received from application team and from the client, application servers are upgraded one by one on a new platform.

Reuse

In this approach, the same operating system and server are used at the target location as well. This approach points to a scenario where in the application has a dependency on a particular operating system and cannot be upgraded or changed.

Thus, in such a scenario, similar server along with similar operating system are migrated to the target DC and are reused in the application environment.

Migration documents

To execute an effective migration life cycle, vendor organization is required to craft potent documents that will be used during different phases of the entire migrations project. This will include.

[*Discovery tool report*](#)

This report is created at the end of the discovery report. This report is the output of the discovery tool that is setup to fetch and collect client environment details. This report is the main output of the discovery phase.

Inventory details

Inventory details are created when discovery reports are analyzed. This is being done in the assessment phase of the migrations project. It is being termed as absolutely critical to prepare an Inventory report for all the Assets running in the environment.

Assessment report findings and observations

Assessment report and findings are done in the phase that follows the discovery phase. This is the phase where in all the information gathered for the client environment is analysed and assessed. All the assessment findings and observations are then documented and are used to formulate migration move groups and waves for the actual cutovers.

Workloads health and age report

This report is related to the age of all the hardware devices being functional in the client environment. It is imperative to know the age of the hardware devices prior to making detailed migration plan. Aged, old, and legacy hardware devices are always kept out of the migration plan of things and thus, special emphasis is being paid on finding hardware age and devices. This report is being analysed in the assessment phase.

Databases versions and instances reports

Databases and their number of Instances form a very critical part of an application environment. Collecting information related to all the databases and their respective versions is as critical as moving the application itself. Any mismatches with respect to database will end up spoiling the entire application stack.

Application – server mappings

Application to server mapping is being done in the assessment phase. This is an activity where in an application is identified along with all of its modules along with the identification of all the workloads they are using for the entire environment. Post identification, each server is mapped with the overall application stack contains all the servers roles such as app server, DB server, web server, or an infra server.

Application criticality index

Criticality index of an application refers to how an application is being treated as a critical application. Such an index highlights as to how various tasks or associated activities of an application stay within the critical parameters. The criticality index of an application is important to analyse to determine as to where do the application stands in terms of criticality and business importance. Such an analysis is done in the assessment phase:

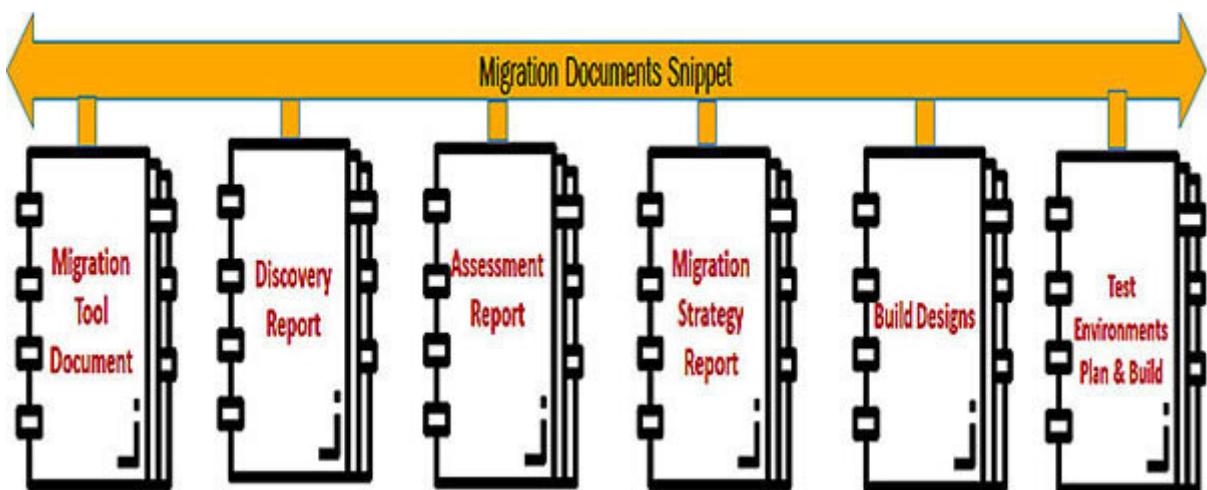


Figure 10.6: Migration documents

Application dependency charts

Application dependency charts refer to a scenario where more than a single application is dependent on a certain variant or service to function. This can be referred in a way that 2–3 applications use a particular OS Service to execute their functions such as WMI service. This way, applications make dependencies. In such a scenario, it is difficult to move a single application and normally the complete set of applications with similar dependencies are migrated together. This analysis is done during the assessment phase.

Migration tool assessment report

The migration tool is the mainstay of a migrations project. During the assessment phase, it is being assessed that which migration tool should be used to migrate the workloads. Any migration tool is being assessed on the parameters such as:

Type of workloads

Data size

No. of disks

Presence of RDMs (Raw Disks)

Database type, model, and version

No. of workloads to be migrated

Number of licenses required

Overall cost

Tool's prior history and performance

Compatibility analysis

Prior to migrations, it is required to check the compatibility of migrating servers with respect to:

Target DC environment

Target DC virtual platform

Migration tool

Migrations will fail if this analysis is not done during the assessment phase.

Physical servers movement

It is very critical to assess and prepare a strategic report as to how physical servers will be migrated. Moving physical servers is a time and costly affair, however, newer hardware servers are always expected to be included in the migration scheme of things. Migrating teams always adopt the strategy of converting physical servers into virtual servers however, new hardware servers are not disposed-off. The physical server's movement analysis is done in the assessment phase.

Reporting templates

These are those templates that are being used to report or inform any status updates/daily updates/activity updates/dashboard updates, and so on. These templates are Excel or Word document-based documents. These templates are prepared in conjunction with client SPOCs and are aligned as per ITIL compliance parameters. All these templates are approved and signed-off by the client before being put into use. These templates come into existence during the strategy and design phase by PMO service management teams.

Email templates

Email templates are the templates that are used to report and update status updates during a specific activity over the email. Specific content and order of content in which it is presented are pre-decided for such an email along with the list of recipients and stakeholders. Additionally, the frequency at which such updates are to be sent is also pre-decided. Such email templates are designed and finalized during the strategy and design phase.

Escalation hierarchy

Escalation hierarchy is basically the organization structure of the vendor organization. It represents various levels and individuals working in the project and their subsequent managers in the organization that is termed as **Escalation**. In case of no response from a certain team or individual, the client is likely to report the matter to the manager of the concerned team as an escalation. An escalation matrix is also created during the strategy and design phase.

Hardware BOMs

BOM for hardware devices (servers/storage/network/security/tools, and so on) is finalized during the end of the assessment phase. Once discovery tool reports are fetched and assessed, that is when migration team comes to know as to what exactly is running in the environment and what is the exact age of the running hardware. They come to know as to what exactly is to be retained and what is to be decommissioned. They also have a clear view of the strategy roadmap and thus accordingly, hardware devices are ordered. For BOMs, vendors for specific track are asked to provide quotes (for example, Dell/HP/IBM for compute servers and Cisco/Juniper for Network). These quotes are received and are sent to the customer for review and subsequent approval. Thereafter invoices are generated and sent to the vendor and orders are placed.

Software BOMs

BOM for software licenses is also prepared during the assessment phase. It is only after completing assessments, migration team determines as to which Software (for example, OS licenses, DB licenses, tools licenses) and their exact quantity are required. Accordingly, vendor quotes from respective vendors are received and are sent to the customer for review, approval, and generating invoices.

Move groups formulation strategy report

Move groups are the core of any migrations project and the migration team starts formulating them from the assessment phase itself. In the assessment phase, when application and infrastructure data are being analysed and associated mappings are done, it starts giving birth to move groups based on the analysis of application stacks. The migration team would normally create a strategy as to how to create move groups.

The finalized version of migration move groups

Move groups are finalized only in the strategy and design phase of the migrations project. A lot of workshops and discussions are done between app owners/DB owners/infra owners and vendor organizations to complete app-server mapping and create application stacks. Based on the application stacks, injunction with application environment owners and administrators, move groups are given a final shape.

Detailed build plan

In most of the migration projects, vendor organizations are required to either build a DC or extend the infrastructure of an existing DC to accommodate the incoming servers which are to be migrated. Thus, it is important to fabricate a DC build plan which is being done in the strategy and design phase.

SOPs – DC build

During data center build or Infrastructure extension, standing procedures are required to be laid out and followed for an effective and seamless build. During the strategy and design phase, all standard operating procedures are created by the migration team:

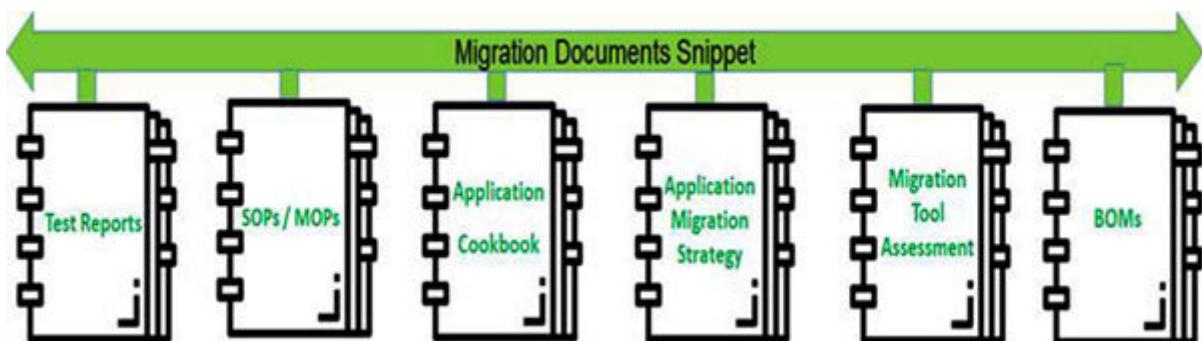


Figure 10.7: Migration documents

MOPs – DC build

For effective configuration and detailed technical processes, **Method of Procedure** are crafted by the migration team during strategy and design phases. MOPs are applicable for each track participating in the build of the DCs prior to migrations.

Application cookbooks

Application cookbooks are played to demonstrate, determine and capture the nature and overall application environment. This is very critical with respect to defining the overall application migration approach. Application cookbooks are prepared in the strategy and design phase.

Application blueprint

The application blueprint contains the overall architecture of the application and its structure. This basically covers the development component of the application as well and contains overall architecture specifications. This is required to understand the application topology prior to the migrations. This blueprint is prepared in coordination with application developers, owners, and vendors if any. This blueprint proves to be an immense help in understanding the application and its operating nature along with all supporting components that are of vital importance during migrations. Application blueprints are prepared in the strategy and design phase.

Risk register

The risk register contains all the potential risks that are foreseen by the migration team. Right from the start of the migration project and from its very first phase, the migration team will start identifying and documenting the risks which may have a direct or indirect impact on the project. Some of the major risks have the capacity to halt or ruin the overall migrations project such as:

Approval delays from client

Influential external stakeholders

Budget constraints

Hostile local environment

Technical limitations

Vendor delays

Resource(s) unavailability

Procurement delays

All such risk items are documented by the migration team.

Risk mitigation plan

Mitigation plan for all the identified risks is documented in the risk register itself. Usually, there are weekly meetings between client and vendor organizations and all the risk items are discussed. Post discussions, owners for each risk item are identified and updated in the risk register. Also, potential mitigation plan along with tentative timelines and dates to close to risk are also updated. Thereafter, in each risk meeting, respective owners are required to update about the status of the risk along with challenges they are facing and expected timelines it will take to close the risk. Risk register and mitigation plan document are a live document and they keep on getting updated throughout the entire migrations life cycle.

Pre-build checklist

For each build that is to be done prior to the migrations, a pre-build checklist is created by the associated team. This checklist ensures that all the pre-requisites are met and through with respect to upcoming DC build and there is no last minute surprise in terms of resources, tasks, and approvals. This checklist comes into force during the strategy and design phase and is played during build phase.

Post-build checklist

For each build that is done during migrations, post build checklists are run to ensure the Infrastructure readiness and there are no missing points. Post migration checklists are the validation that infrastructure is ready as per build designs and there are no deviations. Post build checklists reports are shared with the client post completion to showcase depict infrastructure readiness testing and validation. Post migration checklists are created during the strategy and design phase and are used during the build phase.

Test environment build plan

Test environment build plan is for building a small and short environment to carry out test migrations post infrastructure build. This environment once build, lasts only for a few days and contains just a handful of servers hosting trial versions of applications, migration tools, and other software licenses. This build plan is created during strategy and design phase.

Test migrations plan and steps

This plan contains detailed list of steps to be taken during test migrations to carry out the overall test migrations process by each team/members. This plan contains step-by-step details of how each and every team member will perform his respective activity during test migrations. Test migrations plan is created during the strategy and design phase.

Test migrations results and reports

Test migrations results and reports hold great value as they are the door to actual migrations. Approvals and sign-off for actual cutovers and migrations are granted on the basis of how test migrations have fared. Each and every component of test migrations are tracked that includes:

Network bandwidth utilization

Network packet drops

Latency issues

Transmission rate

Total duration of migration

Errors and logs

Accessibility issues

Migration tool reachability to both target and source environments

Size of the data

Data integrity checks post migration

UAT results

DB checks

ADS and DNS checks

Migration plan document

Migration plan document is treated as the bible of the entire migration project and its life cycle. Migration plan document is the baseline plan based on which entire migrations are carried out. Migration plan document contains:

Move groups

Migration schedules

Migration timings

Stakeholders

SPOCs

Vendors

Reference to SOPs/MOPs

Reference to test environments

Reference to all POCs

Data size

Data replication mechanism

Application names, cookbooks, and blueprints

Rollback strategy and plan

Rollback strategy and a rollback plan is an essential piece of a migrations project. Without proper rollback planning, migrations planning is never termed as complete. Rollbacks refer to moving the workloads back to their original location in case of a migration not going in the desired direction. There may be a number of reasons as to why any migration may fail and in that scenario, the migration team has to ensure that workloads and application environment is back to functional at the source location. The entire process to plan a migration rollback and associated steps are documented thoroughly.

Detailed steps for move groups migration execution

This is one of the most critical document that is created during a migrations project life cycle. This particular step documents as to how a move group will actually be migrated and this involves all the technical steps. This document also contains the list of all the teams and parties who will be participating in the move group migration along with list of the detailed list of activities and tasks associated for each of them. This also documents the sequence and process in which the entire move group migration activity is to be performed. This normally involves the participation of the following parties:

Overall migrations team

Infrastructure team (compute/storage/network/security/DC team)

Monitoring team (infrastructure and application)

Application team

Database team

Vendors

Client SPOCs

PMO team

Handover templates

Handover templates are used post completion of migrations and Hypercare. Once the workloads are migrated to the target location, migration team observes the servers for a few days and checks for any possible issues with:

Server's operating system

Hosted application

Database issues

Network connectivity issues

Data integrity issues

User issues

Migration team will work on any issues that may occur post migrations and may engage other teams or vendors if required. Post migration, server stays under observation with migration team for about 7 days to 2 weeks and then it is passed onto operations team for daily routine support. During handover, migration will help operations team with:

Access

Documentation

Information in hand

Business units

Stakeholders

Artifacts created during migrations

Historical information about server

Vendors information

Platform information

Conclusion

This marks the conclusion of this chapter. We have understood about migration project Topology and we have seen various milestones as well. We have gone through overall migration approach and specific approach and mechanisms for legacy components of an IT environment as well. We have also deep dived into various migration documents and have walked through all possible documents which are fabricated through a migration life cycle.

In the next chapter, we will learn about test environment, types of testing being done in a migrations project, migrations deliverables of all kinds and how do they look like.

CHAPTER 11

The Migrations Project – Deliverables, Testing, and Rollbacks

Introduction

In this chapter, we will read and understand about different migration deliverables that are used for facilitating deliverables associated with a migrations project. We will then move onto understanding a test environment and testing process wherein we will understand everything which is being done in a testing phase. Lastly, we will read about migration rollbacks.

Happy reading dear readers!!!

Structure

In this chapter, we will cover the following topics:

Migration deliverables

Test environments

Infrastructure tests

Application tests

Migration rollbacks

Objectives

Our main objective in this chapter is to learn about a Migration Project's:

Deliverables

Test beds and testing environments

Infrastructure testing

Applications testing

What are migration rollbacks?

Importance of migration rollbacks

Migration deliverables

As we have read through, every migration project has many deliverables. Each migration project has many phases and every phase has different deliverables, some of which are template-based deliverables. These deliverable are delivered in a specific format and with a specific TOC as is specified by the client. A few examples would be:

Project plan

Document

Presentation

Report

Analysis or assessment

Visio diagram or a design

All these deliverables represent ongoing migrations and the phase or the phases that are running live. These deliverables showcase the readiness, understanding, and planning of the vendor organization. Most of these deliverables are pretty common such as a document or a presentation deck or even an assessment

report which is basically a spreadsheet. We are absolutely sure that anyone reading this book is aware about a Word document, a PowerPoint presentation, and definitely with an Excel Spreadsheet. Thus, we are taking the liberty to skip these very common deliverables for the sake of saving time and maintaining the curiosity levels.

Coming back, two of the deliverables which are ‘not common’ and are different here are the project plan which is delivered through Microsoft project and has the MPP extension and a vision diagram or a design which is created in Microsoft Visio and has the VSD File extension.

We will go through both of these deliverables one by one to highlight their importance and contribution to a migration document.

Project plan

A project plan is something that is used and created in the very first phase of a migration project. A project plan will depict end-to-end plan of a migration project along with specified timelines. This plan will contain:

All the migration phases

Goals

Objectives

Associated budgets guidelines

Priorities

Start and end dates

All the domains/tracks participating in the project

High-level activities

Stakeholders list

Milestones associated with each phase:

Task Mode	Task Name	Duration	Start	Finish
	- Datacenter Migrations Project	195 days	Fri 1/1/21	Thu 9/30/21
	+ Planning Phase	10 days	Mon 1/4/21	Fri 1/15/21
	+ Discovery - Data Gathering	15 days	Mon 1/4/21	Fri 1/22/21
	+ Assessment Phase	20 days	Mon 1/11/21	Fri 2/5/21
	+ Migration Strategy & Plan	30 days	Mon 1/25/21	Fri 3/5/21
	+ Infrastructure Build Activities	50 days	Mon 2/1/21	Fri 4/9/21
	+ Testing	10 days	Mon 4/5/21	Fri 4/16/21
	+ Actual Migrations	105 days	Mon 4/19/21	Fri 9/10/21
	+ Handover	14 days	Mon 5/17/21	Thu 6/3/21
	+ Milestones	194 days	Mon 1/4/21	Thu 9/30/21
	+ Documentation	195 days	Fri 1/1/21	Thu 9/30/21

Figure 11.1: Project plan

A project plan will also depict all the activities which are likely to be carried out during the entire migration project life cycle, albeit at a very high level only. Project plans are generally created **Project Management Office** representatives. However, they coordinate with each participating team and individuals to create this mammoth document. Every team being engaged in a migration project will define and come up with their respective list of activities that are to be performed in each of the phases of the overall project. They will have to mention every activity with respect to:

Planning

Assessment

Analysis

Coordination

Designing

Procurements

POCs and testing

Builds

Migrations

Validations

Reporting

Documentation

Handover

Travels

Logistics:

Task Mode	Task Name	Duration	Start	Finish
+	+ Discovery - Data Gathering	15 days	Mon 1/4/21	Fri 1/22/21
+	- Assessment Phase	20 days	Mon 1/11/21	Fri 2/5/21
+	+ Discovery Reports Assessment	20 days	Mon 1/11/21	Fri 2/5/21
+	+ App Details	20 days	Mon 1/11/21	Fri 2/5/21
+	+ Infra Requirements	20 days	Mon 1/11/21	Fri 2/5/21
+	+ Checks	20 days	Mon 1/11/21	Fri 2/5/21
+	+ Datacenters	20 days	Mon 1/11/21	Fri 2/5/21
+	+ HW Equipments	20 days	Mon 1/11/21	Fri 2/5/21
+	+ Application Assessments	20 days	Mon 1/11/21	Fri 2/5/21
+	+ Network	20 days	Mon 1/11/21	Fri 2/5/21
+	- Migration Strategy & Plan	30 days	Mon 1/25/21	Fri 3/5/21
+	+ Strategy	30 days	Mon 1/25/21	Fri 3/5/21
+	+ Migration Tool	30 days	Mon 1/25/21	Fri 3/5/21
+	+ Rack Elevations	30 days	Mon 1/25/21	Fri 3/5/21
+	+ BOM	30 days	Mon 1/25/21	Fri 3/5/21
+	+ Documentation	30 days	Mon 1/25/21	Fri 3/5/21
+	+ Designs	30 days	Mon 1/25/21	Fri 3/5/21
+	+ Network	30 days	Mon 1/25/21	Fri 3/5/21
+	+ Service Management	30 days	Mon 1/25/21	Fri 3/5/21

Figure 11.2: Project plan phases

All these activities are mapped with a start date and an end date in the project plan which gives an indication as to by when those activities will be performed. This makes it clear about as to how quantify and quality the overall success ratio of the project and how to trace and track activities.

It is worth mentioning that these dates may vary during the actual execution of the activities. As per industry standards, up to 20% of variations falls into the normal and accepted perimeter.

Another critical component that is depicted via a project plan is the milestones. We have already learned that each migration phase

is associated with a milestone or milestones. All these milestones are aligned with their respective phase and are mapped with the timelines and dates.

A project plan will clearly mention the duration of any particular migration phase or any major or critical activity and will specify an associated milestones along with exact dates that will depict as to by when the phase is likely to start for execution and in exactly how many days it is likely to reach its completion as is shown in the following chart:

Task Mode	Task Name	Duration	Start	Finish
	≡ Milestones	194 days	Mon 1/4/21	Thu 9/30/21
	Discovery Phase Completion	11 days	Mon 1/4/21	Mon 1/18/21
	Assessment Phase Completion	16 days	Mon 1/18/21	Mon 2/8/21
	Landing Zone Designs	36 days	Mon 2/8/21	Mon 3/29/21
	Move Groups Completion	36 days	Mon 2/8/21	Mon 3/29/21
	Migration Tool Finalization	36 days	Mon 2/8/21	Mon 3/29/21
	BOMs Finalization	16 days	Mon 2/1/21	Mon 2/22/21
	DCs Build Completion	16 days	Mon 3/15/21	Mon 4/5/21
	WAN Link Setup Completion	51 days	Mon 1/4/21	Mon 3/15/21
	POCs Completion	11 days	Mon 4/5/21	Mon 4/19/21
	Migration Plan Completion	21 days	Mon 3/1/21	Mon 3/29/21
	Actual Cutover1	11 days	Mon 5/3/21	Mon 5/17/21
	Actual Cutover2	11 days	Mon 5/31/21	Mon 6/14/21
	Actual Cutover3	11 days	Mon 6/28/21	Mon 7/12/21
	Actual Cutover4	11 days	Mon 7/19/21	Mon 8/2/21
	Actual Cutover5	16 days	Mon 8/30/21	Mon 9/20/21
	Handover Completion	6 days	Mon 9/20/21	Mon 9/27/21

Figure 11.3: Milestones

Project plans are not used to depict only the timelines, phases, or list of activities but they can be used to depict other critical

components of the migration project. Let us take an example of documents.

Normally, in a complete migration cycle, more than 35 documents are created and a project plan can be used to showcase a list to the customer even before even the project is started. Such an advanced depiction of list will signify the expertise and capability of handling large migration projects to the client. It also showcases the readiness and preparation of the vendor organization:

Task Mode	Task Name	Duration	Start	Finish
?	Documentation	195 days	Fri 1/1/21	Thu 9/30/21
?	Discovery Tool Report			
?	Assessment report Findings and Observations			
?	Inventory Details			
?	Workloads Health & Age report			
?	Databases Versions and Instances reports			
?	Application – Servers Mappings			
?	Application Criticality Index			
?	Application dependency Charts			
?	Migration Tool Assessment Report			
?	Compatibility Analysis of Virtual Servers with Target DC platform			
?	Detailed analysis of Physical Servers movement criteria			
?	Reporting Templates			
?	Email Templates			
?	Escalation Hierarchy			
?	BOMs – Hardware for each respective track (as applicable)			
?	BOMs – Software for each respective track (as applicable)			
?	Move Groups formulation strategy Report			
?	Finalized version of Migration Move Groups			
?	DC Build Plan - Detailed			

Figure 11.4: Documentation

Vision diagram or a design

A visio-based diagram is used to create images or designs that are used to portray the future state of environments or the pictorial design of a particular track or component of the migration project.

Visio is basically a diagrammatic software from Microsoft that helps to graphically present the diagrams or interfaces. This software is normally used by all the solution or enterprise-level architects while designing the environment of a client.

A graphical representation helps to present all the interfaces of an environment in a single picture and also depicts the tight coupling between all the components and all their associated services.

Such a diagram is not a direct deliverable but is part of another deliverable such as a high-level design or a low-level design document. It might just be used to create an Image or a design to be presented through a PowerPoint presentation.

A visio-based diagram can be used to:

Illustrate a Datacenter based design

A particular platform such a VMware

A particular track or domain representation such as compute

To outline a particular service or a task

A particular environment (Prod/Non-Prod)

The following [Figure 11.5](#) represents an environment that is to be built by **Data Center Services Tower**. It showcases all the servers that are to be built across all the domains or tracks as part of the overall design and which are going to be part of the Production environment.

It also portrays various services which are actually associated with DCS tower and all the servers that are to be created and setup such as server administration, proxy services, and so on.

Apart from servers and services, the following figure also showcases connectivity to the internet for all the servers and to the cloud. It also depicts the network bandwidth of the **Internet Leased Line**

This represents a complete design, in a single picture to showcase as to how the future environment of the client will appear once it is built and is ready for operations:

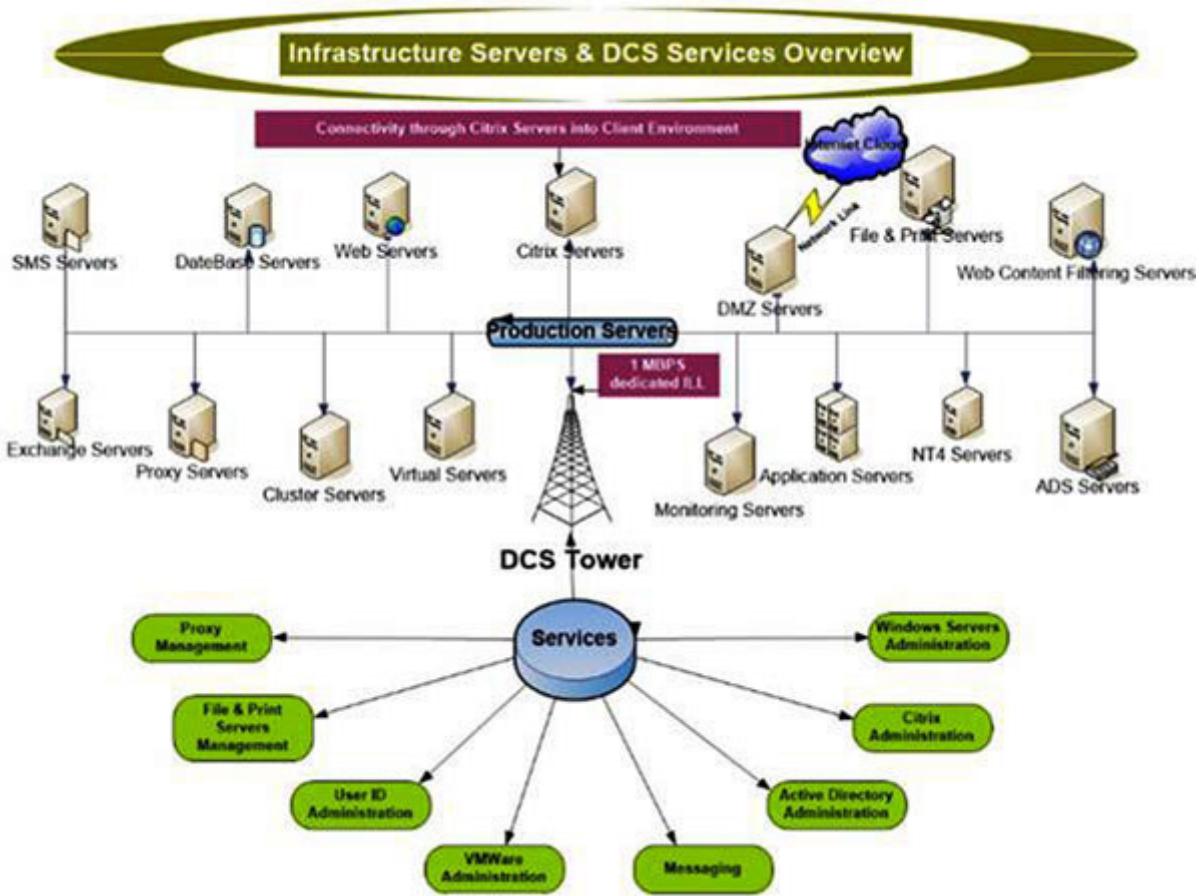


Figure 11.5: Visio diagram

A Visio diagram helps the cause as it presents a pictorial view that contains all the components which are actually going to be the part of the environment and highlights the connectivity and relationships between them.

The following diagram represents pictorial view of a **Primary DC** and its **DR DC**. It basically represents the application environment that is going to be hosted on the hardware platform inside the DC:

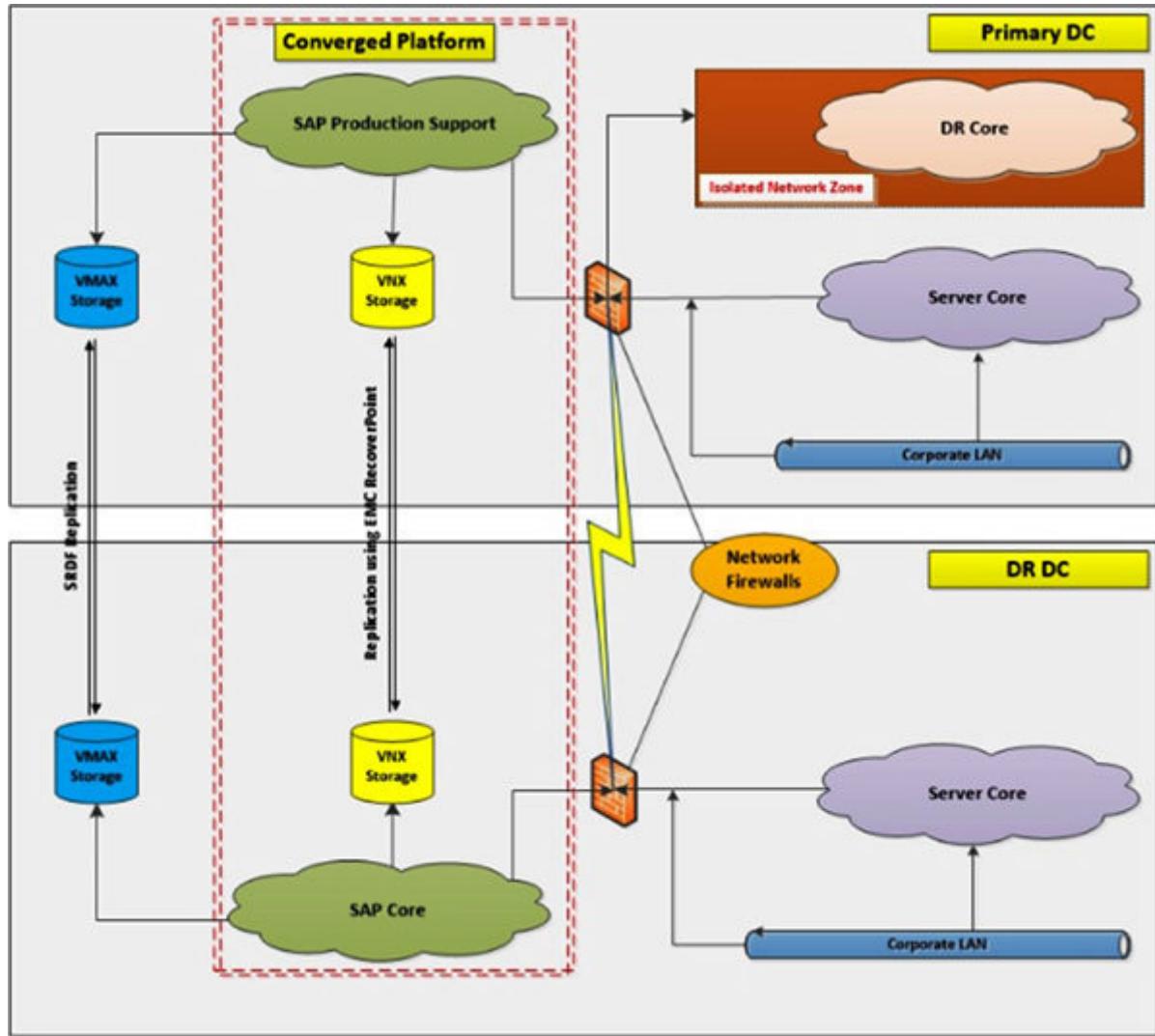


Figure 11.6: Visio depiction of a design

A visio diagram is also used to represent flowcharts, as mentioned below, for any specific process or for any program life cycle. It helps to put up a complete flow of the process with clearly assigned roles and stakeholders along with all the phases. It also showcases different approval/review and test stages of a particular process. It also helps you to identify as to when to initiate the process and exactly when to finish or end it. It also points out precisely as to which activity will be carried out at what juncture of the overall process or the project and by whom:

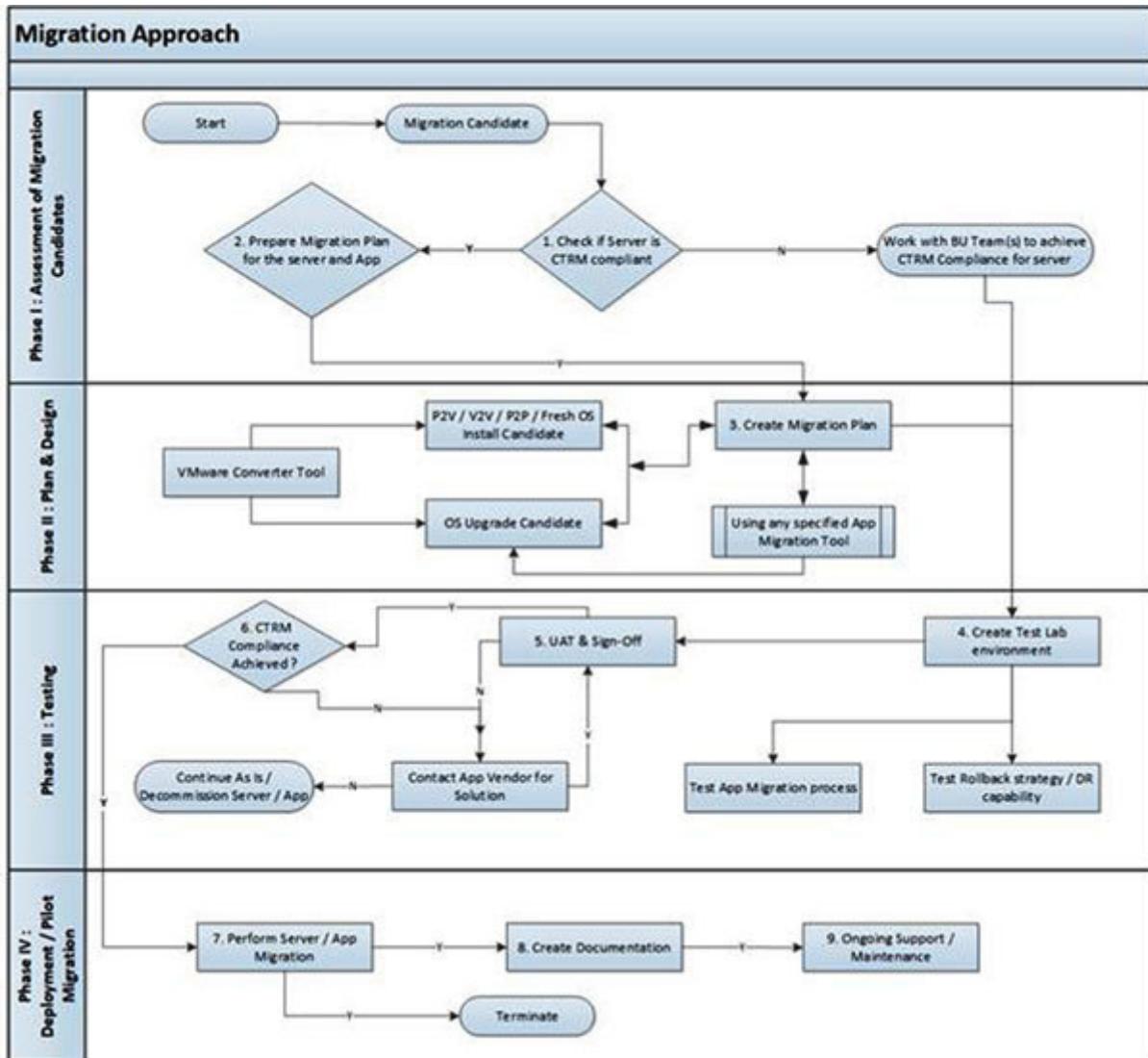


Figure 11.7: Flowchart

Test environments

Test environments or test beds or testing are a critical component of a migrations project life cycle. Test environments are created to check or to test the readiness of the overall infrastructure platform.

This particular phase and set of activities are carried out right before the phase of actual migrations, for both application components and for the corresponding Infrastructure set of pieces. The main objective behind setting up a test environment is to check if the laid out infrastructure platforms are ready for the upcoming migrations.

To elaborate further, environment or infrastructure testing is actually a mechanism to find out any errors, gaps, shortcomings or any component failures in the target environment that is normally a newly built or newly upgraded entity. Environment testings will present a way to check and find any sort of anomalies in the behaviour and performance of overall infrastructure environment and all the specified tools being used or to be used in the target environment.

But the major objective of setting up a test environment is to test and analyze the strength of the network pipes or tunnels or the assigned network bandwidth to the migration link. Any sort of migrations demand dedicated network bandwidth over dedicated

tunnels. It is a bare minimum requirement that is to be fulfilled prior to commencement of the migrations.

All migrations from the source location to the target location(s) are carried over this dedicated network tunnel and through assigned network bandwidth. Infrastructure environment test will help to find out as to how much loads can be taken up by the network tunnel without dropping a single network packet and experiencing any sort of network latency.

A network packet drop and latency issues means trouble in migrations in the form of data loss or overall failure in the migration of servers. Just to simplify as to what it means, this means as to how many servers can be migrated over in a single go or wave without having to face any sort of network issues or choking the network tunnel.

Test environments are planned and designed during design phase in terms of:

Number of servers

Operating system versions

Site/location

Dates

Time and duration

Application environment

Reason being, after doing discovery and assessment of the client IT environment, vendor migration team comes to know about them:

Number of workloads

Operating system flavors and versions

Applications and their stacks

Dependencies

Pre-requisites

Functionality per App/stack

All these information pieces help them to assess as to how many workloads would exactly be required to be built to during test phase to execute test migrations.

As it is very evident, test environments are much smaller in overall quantum as they are only used for testing a few selected servers and other earmarked infrastructure components. They usually consist of up to 4–6 servers only with trial or evaluation

version of operating systems and software(s) loaded on top of them. Even the application versions are evaluation versions that are hosted on top of these servers.

Test servers are generally setup at the source location or the source DC. Test environment setups are done on the basis of test plan and designs. These designs are created during the design phase of a migration life cycle. Test plans usually contain the number of workloads to be hosted and their exact quantity which is decided as per suggestions from all the relevant stakeholders or technology tracks. They would (test plans) also include all the specifications and configurations with respect to the test environments which is to be setup.

Test environments would normally contain:

Up to 4–6 servers

Evaluation version of OS

Trial version of application

Trial version of migration tool

ADS integrated access

Network and monitoring tools

Dedicated network tunnel for test migrations

In general, test environments are setup to conduct two types of tests prior to an actual migration. They are as follows.

Infrastructure tests

Infrastructure is always the base and underlying platform for any environment. And as migrations are carried out from one base to another, thus, it makes it very important to carry out the infrastructure tests to test the overall the resiliency and operational readiness of the target infrastructure platform. Also, it is very critical to test the migration methodology and mechanism to ensure fool proof migrations.

As part of Infrastructure tests, following tests are done:

Migration tool functionality

Frontend load capacity of the migration tool

RDMs or direct attached disks migration capability

Databases instances and logs migration capability

Migration network link functionality

Assigned bandwidth checks

Network transmission errors

As-Is migration tests

Data transmission speed

As mentioned earlier, post testing completion, testing teams are required to prepare a detailed report comprising art of all the Infrastructure tests containing real-time artefacts along with final results. Real-time artefacts will include parameters like:

Start time

End time

Total duration

Number of servers migrated

Network speed

Network bandwidth consumption report

Memory leak errors

Network errors

Logs analysis

Data consistency and integrity analysis

Overall test reports are presented to the management team along with all the stakeholders. Stress is always on the overall results. If the results are good, then it is always a go-ahead for actual migrations. If the results are negative, then stress is being paid to resolve the errors and issues which caused the tests migrations to fail.

It is worth mentioning that once all the errors and issues are resolved, test migrations are carried out once again to validate and achieve the desired outcome:

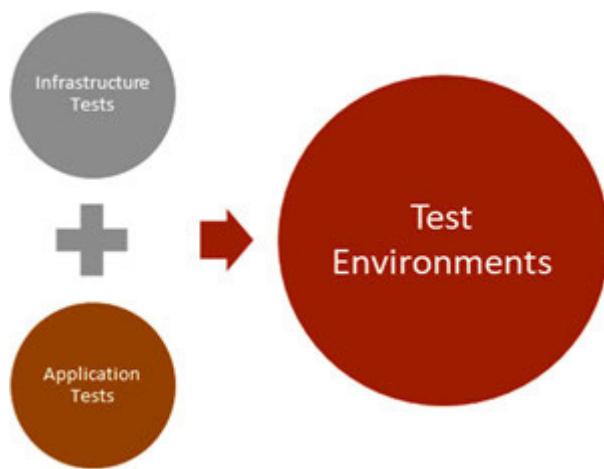


Figure 11.8: Test environments

Application tests

Application tests contain the business end of the migrations and are associated with the tests which are related to the applications or their associated components

As is evident, it is absolutely critical for an organization to have the application environment up and running, especially post migrations. Thus, application tests are also carried out post completion of infrastructure tests during overall testing phase.

Application test will normally include:

Application as-is migration test

Application upgrade and test

OS upgrade – application test

Defects test

Defects remediation's

Application migration tool functionality

Migration tool capability to migrate all the features testing

Error logs

Data transmission speed

Application tests are comprised of **User Acceptance Testing** which is the final test as part of application testing. UAT is done by nominated users or SPOCs from the application team. They are required to check and validate if application and all its modules are working as desired post completion of all the migrations.

This particular step as a part of the overall testing is the most important step as it labels the migrations as a failure or success.

If the applications are not working as expected, then organization may have to revert back to the source environment to continue with overall operations. Migrations will be termed successful only when the complete applications environment gets a go-ahead to function from the target location post-UAT.

Migration rollback

No migrations or migrations plans are complete without the flavor of rollbacks. IT migrations rollbacks is something that nobody wants to see or experience but the fact of the matter is, rollbacks are as important as the actual migrations itself.

Rollbacks is the process of reverting to the source or the previous DC or the environment of the application. Rollbacks are applied only if a migration is stuck or failed. Let us understand this with a scenario:

An application comprising of five different modules has a stack of 10 servers. Stack contains of five application servers (one for each module), two database servers and three web servers. Two database servers are hosting shared database instances for all the modules and are redundant to each other. Same is the case for web servers as well.

When it comes to migrations, entire application is most likely to be moved in a single go due to shared resources. All the servers of the application environment will be migrated together along with application and DB Instances. As per the process, post migrations, all infrastructure and application components will undergo validation testing and user acceptance testing. During UAT testing, even if a single module of the application is not working as per expectations, then it is very much possible that

entire application stack will be reverted back to the source location since they are all using the shared database and web Instances.

In order to attain full operational readiness, organization is very much likely to revert back to original location of the servers until issues at target location are not sorted and tested with 100% positive validation.

This is a scenario where in rollbacks are done as the applications are not fully functional at the target location. However, there can be many more scenarios wherein rollbacks are done.

In order to ensure a successful rollback, migrating teams would normally keep a copy of the entire data of the original servers and application environment at the source location. They would ensure that all the servers to be migrated are completely backed-up prior to the migrations.

They would ensure that any delta changes are also copied as part of the backups. Additionally, during migrations, source servers are kept in powered off state. This gives an option to power ON the original servers in case anything goes wrong in the migrations.

Ultimate intent behind this exercise is to ensure that business is up and running from either of the locations while issues with the migrations are being sorted out:

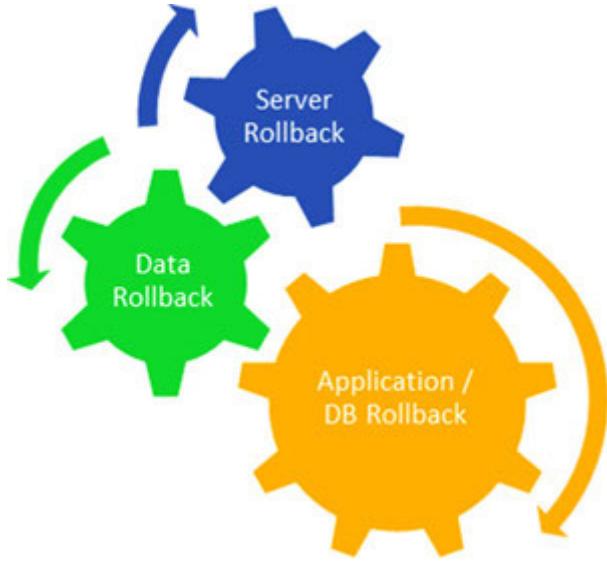


Figure 11.9: Migration rollbacks

Rollbacks are planned during the design phase and post-assessment phase. Just like any other components, rollback strategy and planning is also done along with rollback documentation. During testing of the environment, rollbacks are also tested out. The reason being, migration teams have to be on ground terms in case they have to facilitate rollbacks and they cannot afford to have any surprises.

For rollbacks, documentation with detailed technical steps is prepared. RACI matrix depicting clear roles and responsibilities for all the participating teams, vendors and client members is created and established to bring everyone on the same page. All the documentation associated with rollbacks is part of overall migration plan document and is being reviewed and approved by client prior to implementations.

Conclusion

This marks the conclusion of this chapter on *Migrations – Deliverables, Test Environments and Migration*. We have dived very deeply into various mechanisms being used to deliver various deliverables of a migration project. Post that, we have dug very deep into test beds and environments where in we have also covered various types of tests being carried out. Migration rollbacks and their importance is the last topic which we have covered in this chapter.

This marks the end of all the topics on migrations and brings us to our final chapter of this book which will be an overall summary.

CHAPTER 12

The Epilogue

Introduction

With the commencement of this chapter, we have now started moving towards the end of our wonderful journey. In the last chapter, we will look back at each of the previous chapters and will refresh ourselves with some insights and some memoirs which are likely to last long deep down in us.

In this chapter, we shall cover all the previous chapters as topics and we will list out main excerpts from each one of them.

Objectives

Our main objective in this chapter is to look back at each of the previous chapters and refresh ourselves with what we have learned across this book.

Chapter 1: Solution Architecture - The Astonishing Quotient

Solution architecture deals with the Solution piece of IT infrastructure.

It is the process of designing an application (or applications) or to design the underlying Infrastructure which hosts the designated application.

It is the process that is responsible for the design of one or more applications or services within an organization or for an external client:

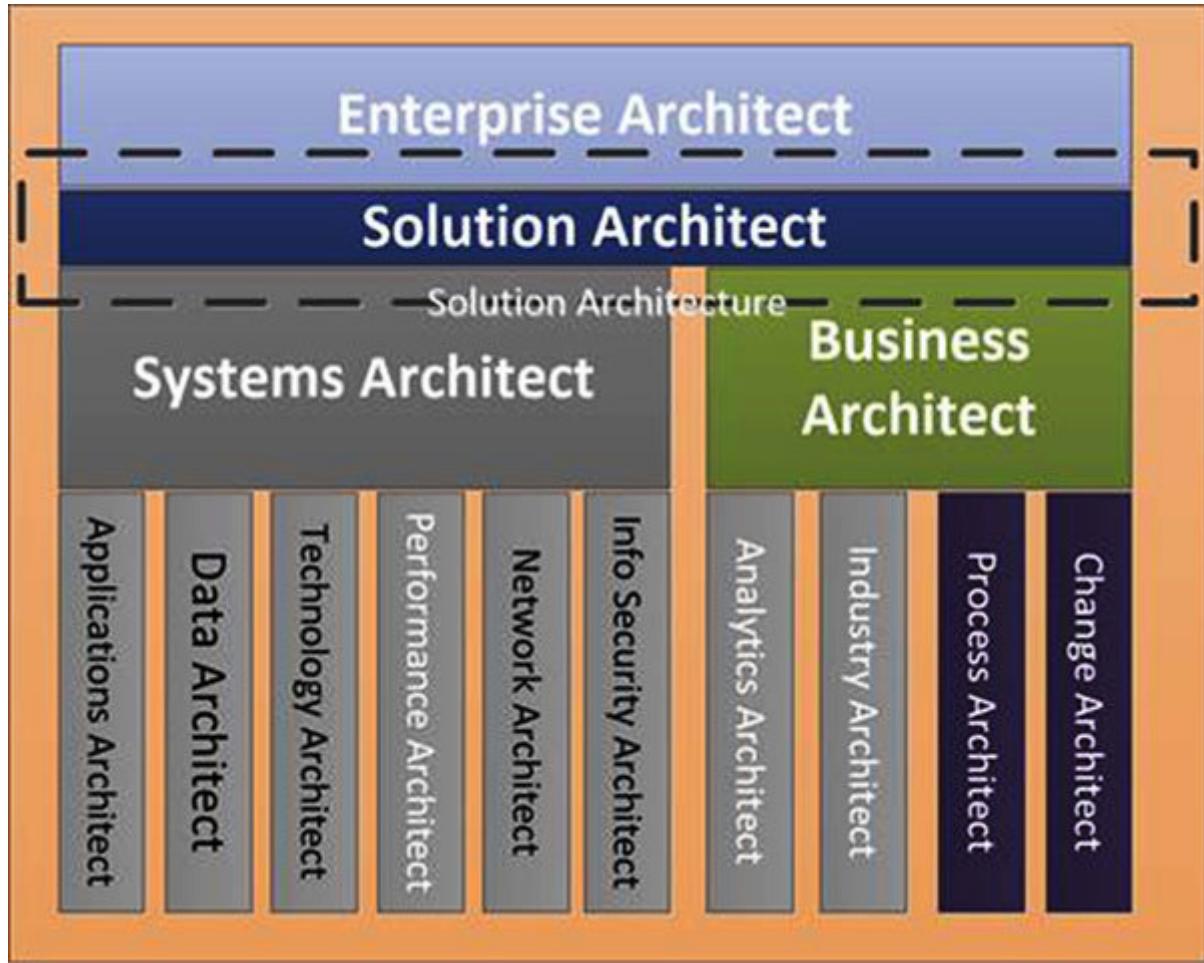


Figure 12.1: Solution architecture

The base model of the architecture is the key to overall solution.

Focus is on shaping and defining the overall solution and all its associated structures, all integrated components, their dependencies and responsibilities and precisely (and minutely) making core decisions related to impending design and all its associated aspects.

Different pieces of IT: IT World is segregated into various sections as mentioned as follows:

Segregation in terms of roles and

In terms of roles and entity, infrastructure technology has been segregated mainly into below mentioned various fields:

Management

Sales/pre-sales

Solution development

PMO

IT operations

Support Functions (Finance – HR – Administration - Internal IT Support - Learning)

Segregation in terms of

IT has always been segregated into two main categories since its introduction, namely: hardware and software. Ever since it was introduced, it has essentially segregated into below mentioned components:

Software

Databases

Human resources and procedures

Telecommunications

Hardware

Chapter 2: Solution Architecture - The Logics Behind

Architecture EA principles are guidelines and rules created for achieving the enterprise objective and goals. Principles are developed based on the respective domains such as business, data, application, and technology as mentioned below:

Business principles

Data principles

Application principles

Technology principles

Solution architecture Solution architecture helps enterprise architecture to deliver organizational values. Solution architects come into play during solution design and implementation.

Prior to solution design and build, solution architects understand solution's business context. They then define the solution vision and its requirements.

Solution architecture Solution architecture is of three types:

Conceptual

Logical

Physical

Solution architecture

Infrastructure modernization which can cover migrations, networking, security and compliance, hybrid environment, and so on.

Data management which can cover database, storage, SAP, healthcare, and so on.

Application development which can cover DevOps, Serverless, and so on.

Analytics such as AI, ML, Big Data, and so on.

Solution architecture A solution architect wears different hats covering the following areas:

Business

Technical

Functional

Data

Build and implementation

Operational management

Solution architecture major

IT Transformations

IT Migrations

Chapter 3: The Transformations Project: Lets Usher

Generic Transformations means change as they bring a change to about everything they are applied to. And it is not just any change, they change a human, a life, a product, a process, a material, and an area altogether.

Transformations when applied to a business, they bring about a change in terms of:

Processes and how they are applied

The way an organization function

All the human resources

Approach and strategic changes and

Off course how the funds are spent

Overall product design, its nature, and outlook

Information Technology Information Technology Transformations can be termed as a tool or a set of tools that are being put in use by an organization to achieve the desired milestones. It can be termed as a mechanism or a process that is being applied to:

Create

Renew

Rebuild

Refurbish

Information technology is also a capability service in itself. This service is being normally applied on an existing environment for an IT Client or a customer or for an internal upgrade for a new application or software.

Strategy and Strategy along with a vision is very important in an IT Transformation project. Any Transformation project (or any other IT project for that matter), which lacks a well-defined strategy and a vision that enables the future version of the IT Landscape, is very likely to fail. Every IT transformation project needs an effective strategy.

It starts with developing a clear, coherent vision of the future state of IT as to what exactly is required.

Transformation project comes with a business case for transforming desired pieces of IT.

Transformation vision must be developed with the input and sponsorship of senior management.

Business and IT leaders also must define the foundational needs, scope, and scale of the IT transformation project, including having a clear project timeline.

An IT transformation aims and seeks to change an organization's entire IT landscape including all existing environments and upcoming ones from being a reactive, inflexible organization to being a proactive, flexible part of the business.

A transformation project or approach enables an organization to adapt and respond. The purpose behind a Transformative initiative is to:

Redesign and realign IT landscape

Adapt as per new opportunities

Unilateral response to extremes

Change as per new trends

High-enable IT delivery

Reimage all existing operating models

Enhanced and effective collaborations

Lasting impressions within the enterprise and beyond its traditional boundaries

Some of the top listed challenges which are faced by an organization/project while going for transformation:

Lack of experience

Capability

Financial implications

Stakeholders support

Conflicting objectives:

Key steps: Some of the key steps are:

Assessment reports

Analysis done on assessments

End-to-end implementation methods covering all dimensions

Validated architectural diagrams

Applications stacks – dependencies

Network information such as IPs change/vLANs extension/Hostname change/Firewall Ports/Load Balancer dependency

Processes such as processes like change Request

Communication mechanism to all stakeholders

Timings for implementation

Finalized templates for emails/reports/artifacts presentation

Type and frequency of reporting/status updates

Bridge call details for daily and weekly calls

SPOC details from each stakeholder

Resourcing/teams details

Escalation matrices

RACI matrix

Training details

Risk registers

SIPs/SOPs to execute tasks in desired manner

Vendor details, and many more

Testing mechanisms

New technologies /tools technical charts

Proof-Of-Concept exercises

Chapter 4: The Transformations Project: Lifecycle and Process Flow Chart

A Project accomplishes a veritable lifecycle in its span. This lifecycle is in-effect a complete sequence of various phases or stages right from its initiation or from when it was conceived first till the point it goes for a complete closure.

This lifecycle is a well-defined and conceptualized framework, which comprises a set of or rather sets of definite, distinct, and recognizable high-level junctures to plan, organize and transform a design/scheme/idea/project/proposal or proposition into actuality. And let us not forget, all this is to be done in a very efficient, methodical, and well-organized manner.

A project lifecycle represents the way a project has been taken up and conducted by an organization and all assigned project team members:

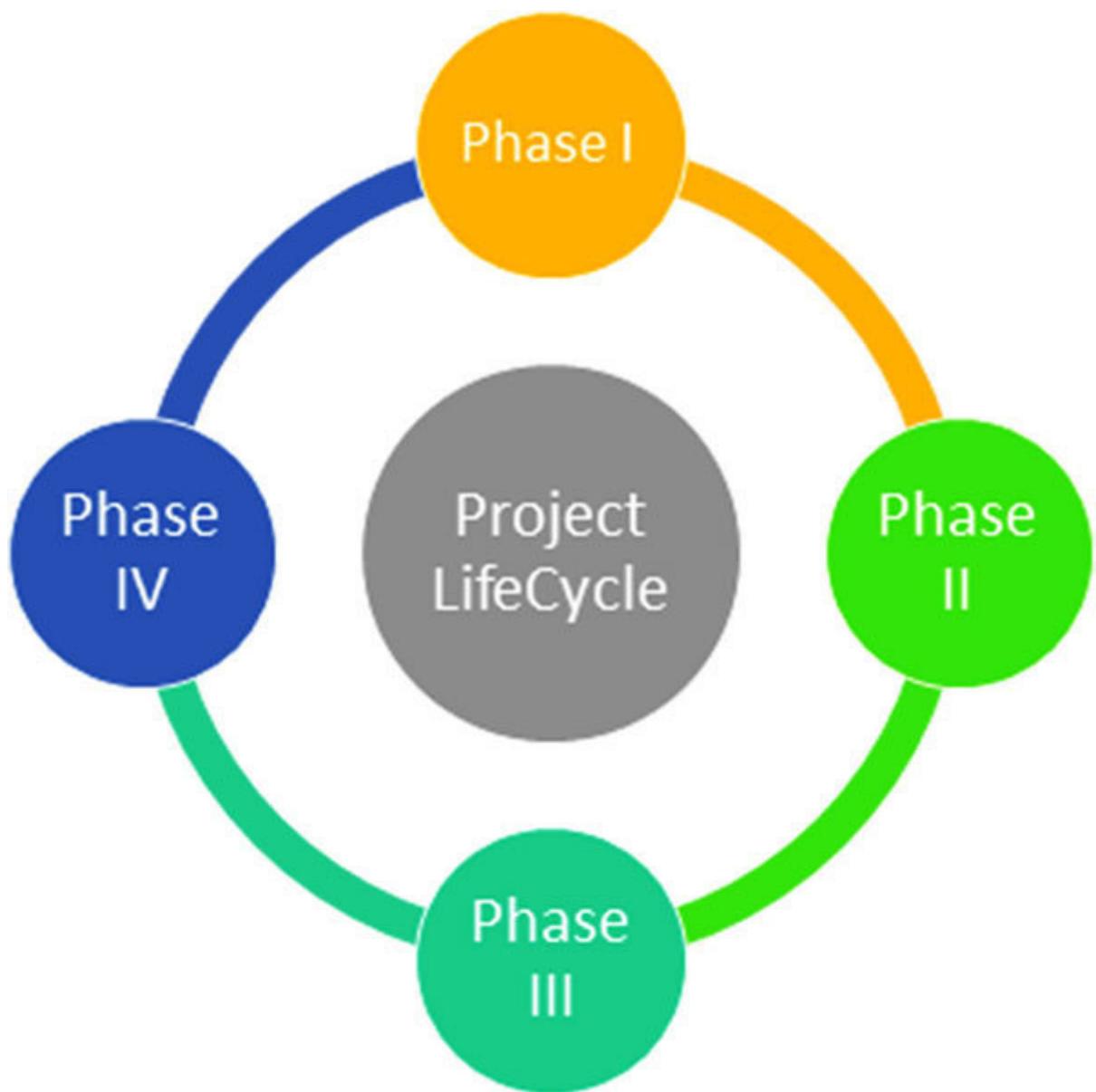


Figure 12.2: Project Lifecycle

Here are some of the listed benefits:

It brings clarity

It assigns clear guidelines as to what needs to be performed when and by whom

It helps to track the project against measured timelines and desired results

It segregates projects into smaller chunks and makes it easy to manage things

It guides and assists teams to be more focused towards their set of responsibilities

It brings the best out of each professional team

It makes the communications flow over and across easy

Lifecycle There are two different types:

Static/plan driven lifecycle

Dynamic/adaptive lifecycle

Process flowchart and its

A process flow and associated flowchart is the mechanism to depict the flow of things that are to be done in an exact way.

A project flowchart would furnish details about processes that are to be followed across the length and breadth of the project.

Process flowcharts are deemed very important in the lifecycle of a project as completing a project is not an isolated task.

A process flowchart would ensure that there is zero or minimum of errors being made by the teams.

Chapter 5: The Transformations Project: Tools and Executions

Transformation planning and best

Finalising the stakeholders and point of contact

Transforming servers and applications

Dates and timelines

Transformation

Discovery and assessment tools

Generic connector appliance architecture:

Micro focus Platespin Transformation Manager

AWS Application Discovery

Azure Migrate Server Assessment tool

Platespin Migrate Tool

Velostrata/Google Migration Services

Cloud Endure

AWS Migration Services

AWS Server Migration Service

AWS Database Migration Service

Azure Migrate Server Migration Service and Database Migration Service

Chapter 6: The Transformations Project: Add-Ons

Engagement An Engagement Roadmap is a pictorial presentation of the strategic roadmap which depicts all the critical milestones for a transformation project. The main objective behind using an engagement roadmap is to provide clear strategy across the project along with its timelines and milestones.

An engagement roadmap is used to depict the:

Entire lifecycle

Various phases

Milestones

Timelines

Project mode such as transition/transformation or operations

Duration it takes to complete a single mode

Any parallel or overlapping modes or phases

Solution Every bit or component of the proposed solution is binded with some of the Solution considerations which act as the base of overall solution approach:

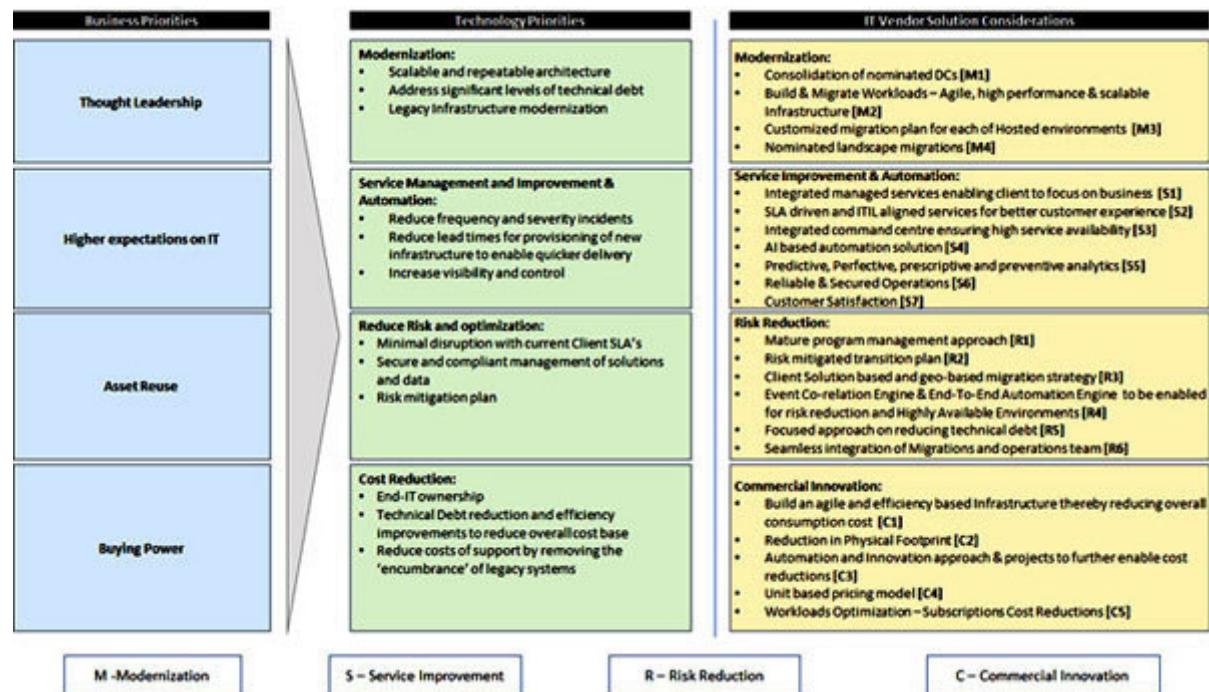


Figure 12.3: Solution Considerations

Business – IT alignment

Innovation/automations

Risk-free environment:

Minimal outages

Cost-effectiveness

Agility to host versatile solutions

Development work

Efficiency/productivity increase

Deliverables

Add-On

Key management service

Secrets management/vaults

Repositories

Encryption techniques, data encryption, and safety

Identity management and access control with multi-factor authentication

Privileged access management solutions

Firewalls and web application firewalls

Certificate authority service

Micro services models and tools

HIPPA and other PCI DSS certification models

Log auditing

Bastion services and host-based services

Dedicated compute models instead of shared physical host models

Vulnerability scanners

Thread detection solutions

Chapter 7: The Migrations: Introduction and Lets Usher

Information Technology In information technology, Migrations is the manoeuvre that is undertaken to process the movement of an IT Landscape into a new environment. It is the wholesale movement of an Operating environment towards a different operating environment.

This movement is perceived as the new operating environment is always believed to be a better one for overall functioning and is likely to pay better returns and dividends along with saving existing incurring costs.

Migrations consist of full-scale project life-cycle and comprises of many phases such as:

Assessment

Planning

Testing

Execution

During migrations project, all the sections of the environment go through a lot of steps/phases such as:

Assess

Plan

Design

Test

Validate

Publish

Approve

Document

Report

Migrate

Manage

Operate

Migrations

Performance

Security

Scalability

Robustness

Agility

Cost-Reduction

Migrations principles:

Decommissions

Consolidations

Centralization

Upgradations

Migrations

Future Roadmap:

Virtual Quotient

ROI

Cost-Reduction

Migrations A migration strategy approach would normally involve:

Service(s) to be delivered

Strategy roadmap

Approach to be used in every phase of the project

Implementation approach

Vision of the project

Mission of the project

Aspirations

Long term objectives

Short term objectives

Tactics to apply

Performance audits

Gaps analysis

Contingency plans

Chapter 8: Migration Types – Application Stacks – Move Groups

Migrations

IT migrations involve below major elements which are:

Source location

Migrations process

Migration tool

Data consistency

Target location

In laymen terms, migrations happen from the source location, which is the existing home of the IT environment that has been earmarked for movement. This existing home or the source location can be any:

Datacenter

Remote or branch office

Client office

Third-party vendor location

Any small site office

Virtual platform

Cloud platform

Migration Primarily, all the migrations are classified into the following category:

P2P – Physical to Physical

P2V – Physical to Virtual

V2V – Virtual to Virtual

C2C – Cloud to Cloud

Application An application stack is a set of individual components that are packed together to run collectively as a function in the form of an application.

All the components of an application stack are also called as a ‘Suite’ which is again a bundle of similar and compatible

constituents to be executed to operationalize an application.

Application move

An Application Move Group is a group of applications which are tethered together due to a similar feature or a dependency. This group of applications is formed with a target of migrating them all together, in a single migration execution movement.

The objective of forming an application move group is to migrate the entire group together in a single go.

[*Chapter 9: The Migrations Project: Process Flow-Chart*](#)

Migrations Main migration components range from:

Infrastructure components

Database components

Application components

Testing components

Tools components

Migrations project A migrations project has a full-scale lifecycle. The migrations project lifecycle is meant to contain and possess set of activities that are carried throughout the overall journey. This migrations lifecycle journey comprises of various facets such as:

Phases

Resources

Planning

Discovery

Documentation

Tools

Applications

Databases

Testing

Assessments

Approach

Strategy

Handover

Vendors

Migrations process A process flow chart is an industry standard practice of representing the entire work flow during a full-fledged life cycle. Migrations project lifecycle contains a complete process work flow. Information and the set of activities that are executed

and carried out during a migration life-cycle are depicted through a process flow-chart. This process flow chart portrays:

Complete set of activities

Migration phases

The order in which tasks and phases appear

How one activity impacts the other one

Chapter 10: The Migrations Project: Topology, Milestone, Legacy, OS-Hardware and Documents

Migrations

Planning and discovery phase

Assessment phase

Migration strategy, planning, and design phase

Infra build/test migrations - POCs phase

Actual cutovers/migrations phase

Handover phase

Migrations overall

REHOST

REPLATFORM

REARCHITECT

RETAIN

RETIRE

Migrations approach for legacy

Emulation software

Swing-kit migrations

Fork-lift migrations

Migrations approach for legacy

Upgrade

Re-platform

Reuse

Chapter 11: The Migrations Project: Deliverables, Testing and Rollback

Migrations deliverables:

Project plan

Document

Presentation

Report

Analysis or assessment

Visio diagram or a design

Test Test Environments or Test Beds or Testing are a critical component of a Migrations project life-cycle. Test environments are created to check or to Test the readiness of the overall Infrastructure platform.

Test environments are planned and designed during Design phase in terms of:

Number of servers

Operating system versions

Site/location

Dates

Time and duration

Application environment

Migration Migration rollbacks is the process of reverting to the source or the previous DC or the environment of the application. Rollbacks are applied only if a migration is stuck or failed.

Conclusion

Presenting and authoring this book has been such an endeavouring and thrilling experience. We have tried to present the facts before you from the practical grounds of a Datacentre going through a build process or being transformed or being migrated to some other place. We have tried to embed Industry best practices and standards to enlighten each and every reader of this book. We hope that we have been able to make a mark and left an impression on the minds of our readers.

Thank you everyone...for your continuous attention and affection.

We will be back very soon with another edition of a book with a new topic altogether which is most likely going to be related to Micro-Services...do watch out for the same...

This marks the final end to this wonderful journey...

Namaste and Stay Safe!!!

A

application move group
about [243](#)
benefits [150](#)
application stack [243](#)
assessment tools [82](#)
AWS application discovery [86](#)
AWS Database Migration Service (AWS DMS) [96](#)
AWS Database Migration Service (SMS) [94](#)
AWS Server Migration Service (SMS) [92](#)
Azure Database Migration Service [96](#)
Azure migrate server assessment tool [87](#)
Azure Migrate Server Migration Service [96](#)
Azure VMWare Solution (AVS) [86](#)

B

Bill of Material (BOMs) [187](#)
business components [155](#)

C

cloud endure
versus Server Migration Service (SMS) [92](#)
Cloud platform

domain-specific environments [140](#)

Cloud to Cloud (C2C) [146](#)

cost reduction

about [108](#)

best practices

D

Data Center Services Tower (DCS) [221](#)

definite purpose

benefits [48](#)

dependencies [147](#)

design and creation

tests and subsequent approvals [173](#)

Disaster Recovery (DR) [223](#)

discovery tools

about [82](#)

agent-based discovery [82](#)

agentless discovery [82](#)

E

emulation migration [137](#)

emulation migration approach

benefits [200](#)

emulation software

End of Life (EOL) [194](#)

engagement roadmap

about [239](#)

CMO [100](#)

contract sign (CS) [99](#)
future mode of operations (FMO) [100](#)
transformations/migrations [100](#)
enterprise architecture [18](#)
Escalation Points [207](#)
European Economic Area (EEA) [108](#)
Expression of Interest (EOI) [64](#)

F

forklift migration [201](#)

G

General Data Protection Regulation (GDPR) [108](#)
generic connector appliance architecture [84](#)
generic transformation
about [235](#)
challenges
history [41](#)
information technology
IT fundamental changes [39](#)
key steps
purpose
strategy and vision
Google migration services
about [90](#)
components and role [90](#)

H

host servers [124](#)

hypervisor [138](#)

I

information technology migrations

infrastructure architecture [18](#)

Infrastructure as a Code (IAAC) [24](#)

infrastructure components

infrastructure tests [227](#)

Internet Leased Line (ILL) [221](#)

IT migrations

about [29](#)

components [30](#)

license-based tools [32](#)

migration tools [32](#)

plan [30](#)

principles [29](#)

process flowchart and lifecycle [33](#)

target [31](#)

IT roles and entity

about [6](#)

IT operations [9](#)

management [7](#)

Organization Support Functions [9](#)

Project Management Office (PMO) [9](#)

Sales [8](#)

Solution and Software Development [8](#)

IT technology
about [10](#)
databases [11](#)
hardware [10](#)
people and procedures [11](#)
software [10](#)
telecommunications [11](#)

IT transformation
about [234](#)
benefits [27](#)
challenges [236](#)
purpose [235](#)
strategy and vision [235](#)

IT transformation, phases
client, initiating [27](#)
client sign-offs [28](#)
implementation [28](#)
plan and design [28](#)

IT transformation project
benefits

L

Local Area Network (LAN) [11](#)

M

major processes
about [25](#)
IT transformation [26](#)

Method of Procedure (MOPs) [209](#)

Microfocus Platespin transformation manager

about [85](#)

features [84](#)

migration

about [241](#)

benefits [129](#)

full-scale project life cycle [241](#)

impact

phases [241](#)

migration approach [245](#)

migration approach, for legacy hardware

about [245](#)

emulation software

fork-lift migrations [201](#)

swing-kit migrations [201](#)

migration approach, for legacy OS

about [202](#)

operating system upgrade [203](#)

re-platform [204](#)

reuse [204](#)

migration components

about [244](#)

infrastructure components

migration deliverables

about [216](#)

project plan

vision design [224](#)

vision diagram [224](#)

migration documents

about [204](#)

application blueprint [209](#)
application criticality index [205](#)
application dependency charts [206](#)
application cookbook [209](#)
assessment report findings [204](#)
assessment report observations [204](#)
compatibility analysis [206](#)
databases versions [205](#)
detailed build plan [208](#)
discovery tool report [204](#)
email templates [207](#)
escalation hierarchy [207](#)
handover templates [213](#)

hardware BOMs [207](#)
health and age report [205](#)
instance reports [205](#)
inventory details [204](#)
Method of Procedure (MOPs) [209](#)
migration move groups version [208](#)
migration tool assessment report [206](#)
move groups formulation strategy report [208](#)
move groups migration, executing [213](#)
physical servers movement [206](#)
planning [212](#)
post-build checklist [210](#)
pre-build checklist [210](#)
reporting templates [207](#)
risk mitigation plan [210](#)
risk register [210](#)
rollback strategy and plan [212](#)
server mappings [205](#)

software BOMs [208](#)

SOPs [208](#)

test environment build plan [211](#)

test migration plan and steps [211](#)

test migration results and reports [211](#)

migration drivers [125](#)

migration life cycle

migration methodology

about [133](#)

elements [243](#)

migration phases

assessment

design and creation

migration and handover

planning and discovery

tests and subsequent approvals

migration principles

about [242](#)

centralization [122](#)

consolidations [122](#)

cost-reduction [125](#)

decommissions [121](#)

future roadmap [124](#)

migrations [123](#)

ROI [124](#)

upgradations [123](#)

virtual quotient [124](#)

migration process flow [244](#)

migration project lifecycle [244](#)

migration purpose

about [118](#)

agility [120](#)
cost-reduction [120](#)
performance [119](#)
robustness [120](#)
scalability [120](#)
security [119](#)
migration rollback [246](#)
migrations milestones
migrations process [133](#)

migration strategy
about [242](#)
AS-IS migrations [31](#)
decommissions [31](#)
rearchitect [31](#)
systems [31](#)
migration tools
functions [32](#)
migration topology
about [245](#)
actual cutovers phase [189](#)
assessment phase [187](#)
design phase [188](#)
handover phase [189](#)
migrations phase [189](#)
planning [188](#)
planning and discovery phase [187](#)
POCs phase [189](#)
strategy [188](#)
migration types
about [243](#)
Cloud to Cloud (C2C) [146](#)

- emulation migration [137](#)
- forklift migration [135](#)
- P2V, with virtualization technologies [138](#)
- Physical to Cloud (P2C) [140](#)
- Physical to Physical (P2P) [135](#)
- Physical to Virtual (P2V) [137](#)
- swing kit migration [136](#)

- Virtual to Cloud (V2C) [145](#)
- Virtual to Virtual (V2V) [141](#)

- o

- operating systems [11](#)

- p

- Physical to Cloud (P2C) [140](#)
- Physical to Physical (P2P) [135](#)
- Physical to Virtual (P2V)
 - about [138](#)
 - with virtualization technologies [139](#)
- Platespin migrate tool
 - process flowchart
 - about [60](#)
 - benefits
 - for transformation project [63](#)
 - importance [61](#)
 - process flowchart, phases
 - client transformation
 - design and solutions

implementation
plan and design
project lifecycle
about
benefits [58](#)
types
Project Management Office (PMO) [217](#)
Proof of Concept (POC) [73](#)

R

real-time artefacts
parameters [227](#)
request for proposal (RFP) [64](#)
return on investment (ROI) [108](#)

S

Scope of Work (SOW) [187](#)
security risks [108](#)
Senior Management or Leadership Team [7](#)
Server Migration Service (SMS)
versus cloud endure [92](#)
Service Level Agreements (SLAs) [19](#)
Single Point of Contact (SPOC) [66](#)
solution architect role [22](#)
solution architecture
about [232](#)
base model
evolution

hats [234](#)

IT [6](#)

IT roles and entity [6](#)

IT technology [10](#)

levers [233](#)

major processes [234](#)

overview

purpose [233](#)

results [14](#)

types [233](#)

solution architecture levers

solution architecture principles

about [233](#)

application principles [20](#)

business principles [19](#)

data principles [20](#)

technology principles [21](#)

solution considerations [239](#)

strategy parameters [128](#)

Suite [146](#)

swing kit migration [201](#)

T

test environments

about [246](#)

application tests [228](#)

infrastructure tests [227](#)

total cost of ownership (TCO) [108](#)

transformation best practices [80](#)

transformation planning
about [80](#)
dates and timelines [81](#)
point of contact, finalizing [81](#)
servers and applications, transforming [81](#)
stakeholders, finalizing [81](#)
transformation project
add-ons [240](#)
benefits [237](#)
lifecycle and process flow chart [237](#)
lifecycle, types [238](#)
process flow chart, importance [238](#)
tools and executions [238](#)

transformation tools
about [87](#)
AWS application discovery [86](#)
AWS Database Migration Service (AWS DMS) [96](#)
AWS Database Migration Service (SMS) [94](#)
AWS Server Migration Service (SMS) [93](#)
Azure Database Migration Service [96](#)
Azure migrate server assessment tool [87](#)
Azure Migrate Server Migration Service [96](#)
cloud endure [92](#)
discovery and assessment tools [82](#)
generic connector appliance architecture [84](#)
Google migration services [90](#)
Microfocus Platespin transformation manager [85](#)
Platespin migrate tool
Velostrata [90](#)
types, solution architecture
conceptual solution architecture [18](#)

logical solution architecture [19](#).
physical solution architecture [19](#).

U

User Acceptance Testing (UAT) [228](#)

V

V2V migration
platform upgrades [142](#)
switch migrations [143](#)
Velostrata [90](#)
virtual private cloud (VPC) [92](#)

Virtual to Cloud (V2C) [145](#)
Virtual to Virtual (V2V) [141](#)

W

Wide Area Network (WAN) [11](#)