



Legacy Apps and Data Migration

Move your Apps and Data to Cloud

Contents

- 01** Abstract
- 02** Legacy Platform Challenges
- 03** Top 3 Goals for Legacy Modernization
Per Gartner Survey
- 04** Migration Journey – Steps to Migrate to Cloud
 - Step 1:**
Understand the Current Environment and
Analyze Current Usage
 - Step 2:**
Design an Integration Plan
 - Step 3:**
Setup Deployment and Data Pipelines,
Cloud Data Stores, Native App, and Container
Environments
 - Step 4:**
Migrate Data Definition Language (DDL) and
Enable Data and App Governance
 - Step 5:**
Load Legacy Data (One Time) and App to Cloud
 - Step 6:**
Update Applications to Consume Data from
Cloud Data Stores
 - Step 7:**
Validation – Run QA with Parallel Stacks
 - Step 8:**
Decommission Legacy Apps and Data
- 05** Conclusion

Abstract

As per industry estimates, most of the business-critical applications and data not only of midsize companies but also of many Fortune 500 corporations run on legacy systems, due to the system's acceptable performance and stability. Key challenges faced by CIOs and IT managers handling these applications are:

- High cost of maintenance and enhancements
- Scaling up is difficult due to legacy technologies
- Unmitigated risk associated with lack of skill availability in the resource market
- Tech debt and unsupported platforms
- Higher time to market due to large, monolithic, complex, and less productive systems
- Rigid Integration with newer systems due to incompatibility with new and different technologies

Legacy Platform Challenges

In this fast-paced world of today, many enterprises and organizations find themselves at a juncture where they are faced with a difficult choice:

- To keep running age old systems that have been tuned and customized to the functional requirements over the years, supporting business critical operations

OR

- To answer to increasing business pressure to become agile, flexible and more responsive by upgrading technologies that struggle to keep up with modern times

As enterprises struggle with this dilemma, they are also faced with the following challenges that hamper an organization's digital transformation:

- High risk of migration
- Extended timelines lead to stale ROI
- Numerous technical hurdles, such as:
 - > Rewriting current apps, scripts, and jobs
 - > Lack of domain knowledge
 - > Manual verification and audit of the apps and data
 - > Managing a large amount of data, spreading to multiple TBs and PBs
 - > The need to have technical knowledge of the app and data
 - > Multiple versions of apps
 - > Unsupported apps

These challenges drive organizations to not proactively planning its modernization. And, in this digital era, this tendency might just lead to disaster.

With evolution of new-gen systems, armed with modern data analytics, native apps, APIs and microservices on Cloud capabilities, separation is needed between the computations from storage, thereby enabling enterprises to scale their infrastructure automatically and instantly. Many enterprises have begun to evaluate using them in their modern applications.

Here are five key things to look out for during your migration journey to the Cloud:

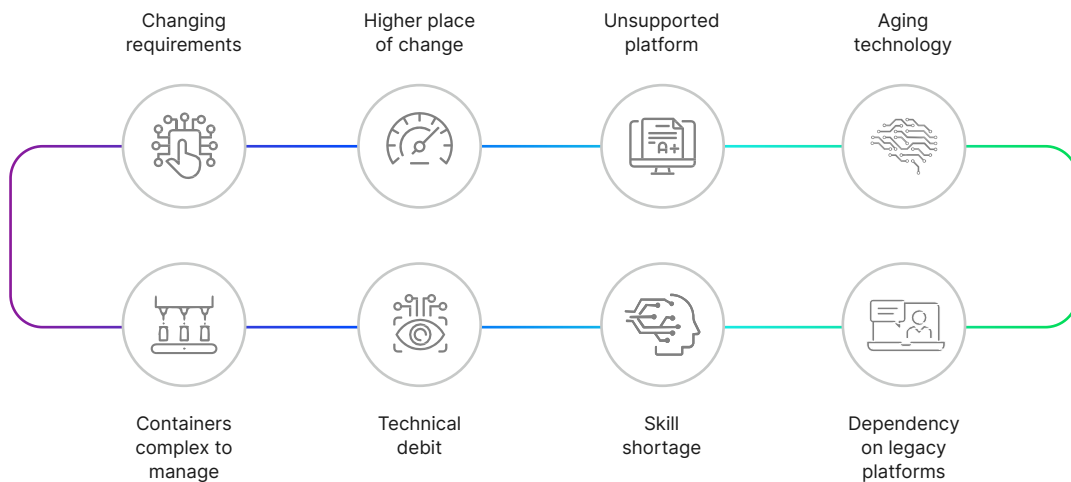
- **Legacy Data migration** to Cloud requires looking into traditional/current data pipelines for quality, and then moving them to multi-Cloud pipeline environments such as Spark or Cloud specific ones like AWS Glue, EMR, Azure ADF, and Synapse Pipeline
- **Migration of Legacy Mainframe applications to Open Stack applications** compatible on multi-Cloud environment
- **Migration of Legacy Windows applications running on Windows 2008 and Windows 2012**
- **Migration of Legacy APIs into Cloud based APIs and Microservices**, Service Mesh using industry standards like BIAN, IFX, FDX, LIMRAA, etc.
- **Migration of Legacy Enterprise Apps** to container, and the establishment of CI/CD to be ready for deployment to multi-Cloud environment

In addition, organizations need to also be ready for **native development, security, CIAM, SIEM, and holistic Cyber Security, all in one place.**

Taking this new-age approach while adopting and migrating to the Cloud will quicken the migration pace and generate savings. Savings thus realized can be used to build innovative and native solutions on the Cloud to accelerate the Digital Transformation.

For such seamless migration, modern enterprises require a platform that is versatile, cost-efficient, scalable, and performant while providing solutions to cater to the aforementioned five key things. Such a platform will enable seamless migration to Cloud, thereby enabling infrastructure convergence, simplicity, and support for exponential capabilities in a cost-effective and highly efficient manner.

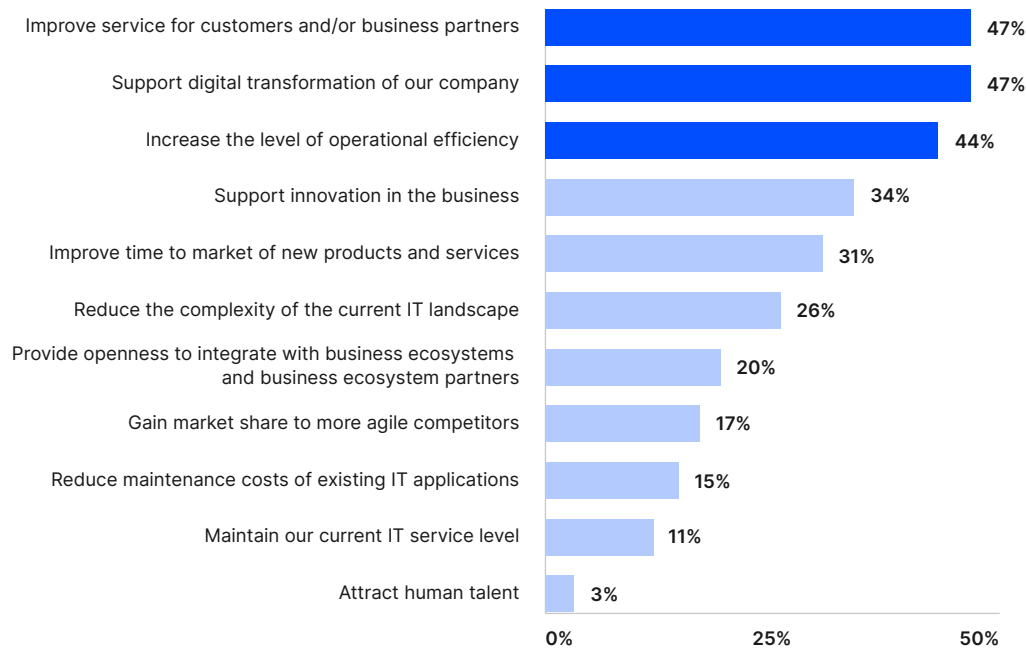
Problem Statements that Drives Modernization



Business Motive to Modernize the Applications:

- Reduce infrastructure costs - consolidation
- People costs/skill strategy
- Business agility/regulatory
- Reduce software maintenance costs
- Sourcing strategy - packaged software

Top 3 Goals for Legacy Modernization per Gartner Survey



Migration Journey – Steps to Migrate to Cloud

Migrations are usually complicated and can get messy if not planned and executed properly. We recommend a carefully thought-out step-by-step approach to migrate your apps and data.



Step 1: Understand the Current Environment and Analyze Current Usage

Start with analyzing your current implementation to get a full 360-degree view of your ecosystem. This process involves finding answers to the following questions:

- What are the priority applications/LOBs?
- Which users are using these apps?
- Which are the data stores/databases and tables that need to be migrated?
- How is data in these tables/objects getting used — which scripts and applications pull data from these tables/objects?
- Which users, roles, and applications can access these apps, databases, and tables/objects?
- How is data getting loaded into these tables/objects — which scripts and applications are responsible for data loading?



Step 2: Design an Integration Plan

Now that you are done with understanding and documenting your current usage, let us lay out an execution plan for the migration.

Instead of an all-in-one-go approach to migration, we recommend a phased approach as follows:

- Do a quick Discovery to finalize the scope.
- Plan to Migrate either a set of apps or a data store, get some success under the belt, and then move on to the complex migration tasks.

No matter which approach you take, by the end of this step, you will have a full plan of action for the migration in place.

- Take the output from the previous Analysis step, information, databases, and migrations into logical phases
- Phase 1 will repeat based on priority with a focus on a particular application, data store, or low-impact table cluster
- It is best to plan a full vertical slice migration — i.e., end-to-end ingestion, migration, and consumption together. This helps isolate issues at every stage early in the process
- Identify tools that can help speed up the migration process. Hand-coding is generally not recommended
- Try out Automated containerization tools like CloudHedge, Windows Migration tools like CloudHouse, Data Pipeline Automation tools like DataSwitch, Self-Service Ingestion, intelligent Data Management (iDM), Automated Data Validation, and Audit Control for executing the migration. They can reduce time to market substantially by automating a significant portion of the migration and re-tooling. This becomes especially beneficial for repeatable steps in the phased migration approach

Please note that we recommend a focus on outcomes versus a focus on tools. It is generally recommended not to hand-code but hand-correct to stop mess-in and mess-out. Tools used here are mostly one-time use during the migration.

- Once migrated, all scripts are enabled in the native Cloud environment and its tooling environment to take out any other tool dependency



Step 3: Setup Deployment and Data Pipelines, Cloud Data Stores, Native App, and Container Environments

Now that we have a clear plan of action, let's get started on executing the migration. First up, you want to set up your Cloud environment and sizing to meet your needs.

Configure the following:

- Create Multi-cloud deployment Pipelines for app deployment
- Create Data Pipelines using good quality code available to move data to Cloud utilizing the Cloud-native data tools

- Create a Cloud Data Lake on Cloud data stores such as Azure Data Lake or AWS S3
- Create structured databases and warehouses as needed
- Create users and accounts
- Create native app build environments and deployment pipeline
- Create deployment orchestration using native container and workflows tools that can work in multi-cloud environments



Step 4: **Migrate Data Definition Language (DDL) and Enable Data and App Governance**

Now that you have primed these Cloud data stores with accounts and databases, you can start creating the database structure in stores like Azure Synapse, Redshift, and Snowflake. This typically involves using DDL exports from Legacy stores, converting them to Cloud datastore compatible DDL, and executing it.

However, an intelligent Data Pipeline platform like DataSwitch can handle this automatically, saving you a tedious and potentially error-prone process. With this, all you need to do is select the tables and data fields that need to be migrated to Cloud data stores.



Step 5: **Load Legacy Data (One Time) and App to Cloud**

After Step 4, your Cloud Data Store should be ready to handle data from Legacy data stores.

If you are running a legacy data store on-premises, it is advisable to push data first to a Cloud storage solution like AWS S3 or Azure DataLake before loading it into Snowflake, Synapse, or Redshift.

This step is another reason for considering an intelligent Data Pipeline platform like DataSwitch. DataSwitch can create scripts that can be run in Cloud-native environments using their tools to extract data from legacy data environments automatically, load it into an intermediate Cloud storage if needed, and subsequently load the data into Cloud data stores.

Use an app scan tool like CloudHedge or CloudHouse Alchemy to scan the environment for the app and its deployments, and understand integration points and configurations.



Step 6: **Update Applications to Consume Data from Cloud Data Stores**

Once you have ensured that all your data is now being written into Cloud data stores, start tuning all your applications that were dependent on Legacy data. We recommend running two parallel versions of each application during the migration — one that continues consuming data from Legacy data and another that is based on data from Cloud data stores.

Customizing scripts and applications can be complicated, and it might require significant re-engineering. This is where an Automated Pipeline platform like DataSwitch or CloudHedge comes in handy. The platform creates scripts to be run using Cloud-native tools in their environments using native tools, thereby reusing the existing pipelines!



Step 7: **Validation – Run QA with Parallel Stacks**

Once you have configured parallel end-to-end stacks, a comprehensive QA cycle should be run to compare functionality across two systems.

In addition to validating the applications, you should compare row counts, sums of numeric fields, and hashes of string fields to identify discrepancies. This step can be easily automated with our automated data validation tools.



Step 8: **Decommission Legacy Apps and Data**

Finally, we are done with migration! With each one of the steps completed and all the applications running on the Cloud, inform all your on-prem users about their new accounts and other changes. Turn off all load processes and revoke access to legacy apps and data in the traditional data center on-premise.

Conclusion

This whitepaper introduced you to a comprehensive step-by-step plan for migrating Legacy apps and data from on-premises to Cloud and is designed to avoid any pre or post-migration hiccups.

As we saw, a big chunk of the migration process can be achieved more quickly and reliably with intelligent, modern Data Pipeline Automation tools like DataSwitch, Self-Service Ingestion, intelligent Data Management (iDM), Automated Data Validation, and Audit Control, Automated Containerization tools like CloudHedge, CloudHouse, Asysco AMT tools.

While legacy systems have been around for quite a few years, new-gen systems are slowly replacing them. Their scalability and flexibility in usage are something enterprises look forward to. Therefore, migration processes that ease the transition from on-premises systems to Cloud systems are the need of the hour.

About Us

SLK is a global technology services provider focused on bringing AI, intelligent automation, and analytics together to create leading-edge technology solutions for our customers through a culture of partnership, led by an evolutionary mindset. For over 20 years, we've helped organizations across diverse industries - insurance providers, financial service organizations, investment management companies, and manufacturers - reimagine their business and solve their present and future needs. Being A Great Place To Work Certified, we encourage an approach of constructively challenging the status quo in all that we do to enable peak business performance for our customers and for ourselves, through disruptive technologies, applied innovation, and purposeful automation. Find out how we help leading organizations reimagine their business at <https://www.slksoftware.com/>

© SLK Software Pvt. Ltd. 2022