How to Migrate Your Card Processing Platform with Minimal Disruption to Your Business?

— Payments Consulting Team, LTIMindtree

LTIMindtree

Large-scale, complex card migration projects must be designed to minimize risks, by keeping tabs on account requirements pertaining to different products, markets, and legacy systems. Thus, it is imperative to map out a migration plan that focuses on ample testing and a robust contingency strategy to ensure a smooth transition with minimal business impact and high customer satisfaction.
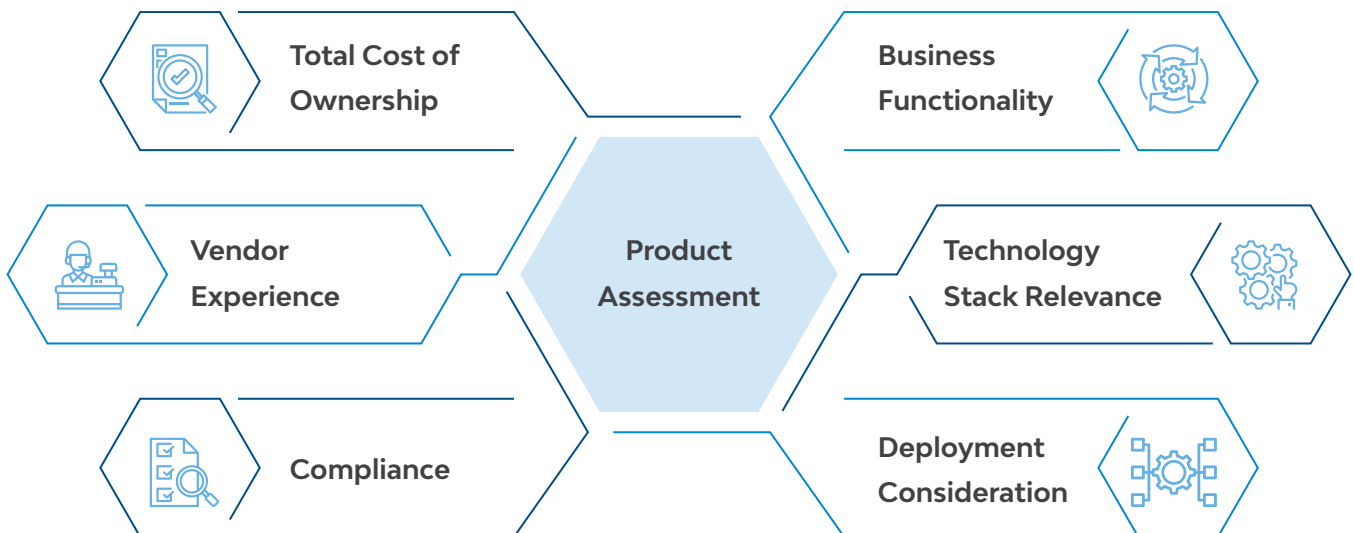
When it comes to platform migrations in the banking industry, these are normally complex programs requiring significant amount of time, effort, and money – especially the card platform migrations. Irrespective of the cause for the card migration which could be due to mergers and acquisitions, modernization of legacy applications or new regulatory requirements like data localization, there are a few key parameters, events, and activities that are a part of the Card Platform Migration journeys.

Below are the **11 essential steps** to consider for a successful card migration program.

01

Product Selection

The first step for a bank is to select the right card management product or platform that will be best-suited to cater to their current and future requirements. Banks typically shortlist a few products/vendors based on primary research and make the final selection before embarking on the migration journey. This is a critical pre-migration activity and there are several factors that needs to be evaluated for selecting a suitable product. Key factors for evaluation are:

- Total Cost of Ownership
- Vendor Experience
- Compliance
- Product Assessment
- Business Functionality
- Technology Stack Relevance
- Deployment Consideration

Product functions and features ensuring all key requirements are either catered to out-of-the-box or minimum customization needed.

Capital costs and Operating costs.

Technology stack relevance should align with technology roadmap defined by the bank.

Flexibility on deployment consideration (On-premises, Cloud, Third-party processor).

Licensing and ownership model offered by the vendor.

Prior experience of product vendor on implementation for banks of similar size.

Ease of use configurability of the product and ease of integration with the interfacing systems.

Some of the leading Card Management and Issuing Products in the market are:

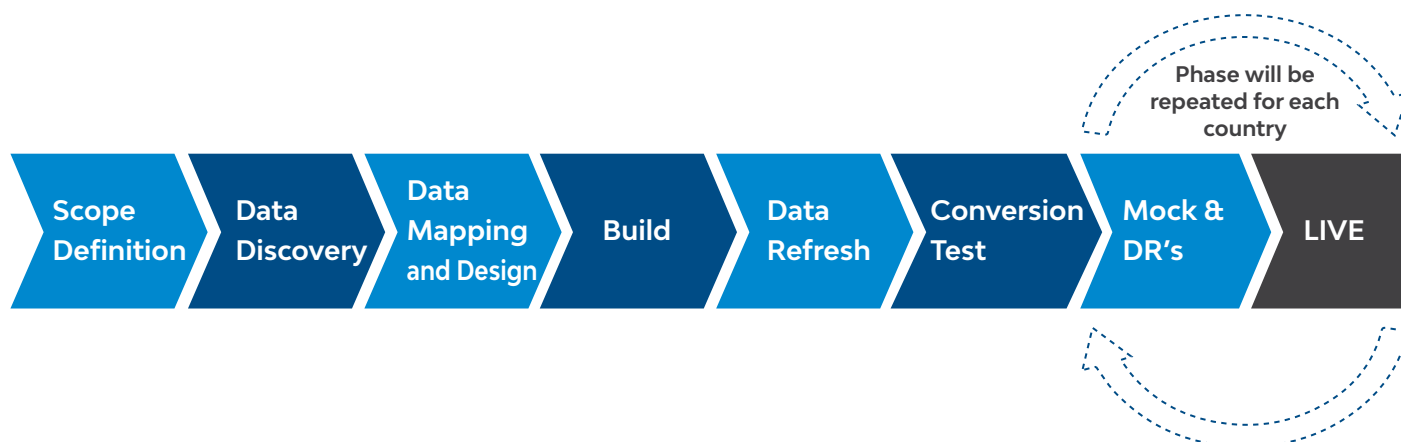| Product	FirstVision / VisionPLUS	PowerCARD	PRIME	WAY4
Vendor	fiserv.	HPS /	TSYS	openway

02

Requirement Definition and Functional Gap Analysis

The business requirements should be elaborately defined as a precursor to the Functional Gap Analysis. The gap analysis is normally done as a part of a workshop which will have representation from product SMEs, Business and Operations. The product vendor will demonstrate how each requirement will be fulfilled by their product. The outcome of the workshop will be a comprehensive list of functional gaps that are currently not supported on the target platform. The gap list will be further evaluated by the target platform SMEs focussing on size, complexity, and cost and the same will be presented to the Change Control Board for approval.

Note:

It is essential that the product vendor shares documentation on how the requirements, will be solutioned on the target product. The business should sign off all functional specification documents.

Phase will be repeated for each country

| Scope Definition | Data Discovery | Data Mapping and Design | Build | Data Refresh | Conversion Test | Mock & DR's | LIVE |

03

Interface Considerations

The card platform interfaces with several different applications like Digital Channels, ATMs, Collections, Core Banking, Fraud and Risk, Customer Onboarding, Notification Engine, Embossing, Datawarehouse, Schemes (Visa/Mastercard) etc. Data is exchanged between these applications either real-time or as a batch feed. Moving to a new platform will result in an integration impact as the interfaces will not be like-to-like, between source and target platforms. The target platform will have its own standard interfaces and the product vendors will normally push the bank to adopt their interfaces for batch and API integration.

Mapping each data element in the interface layouts (files and APIs) will be done to cover the interface requirements. This will result in interface gaps, which would have to be developed prior to start of integration testing.

Note:

It is also important to consider the impact of batch SLAs of the target platform as it can have an impact to surround interfaces. For example, a delay in the handoff of the collection files to the collections application may result in the collections batch process starting late and breaching its current defined SLA.
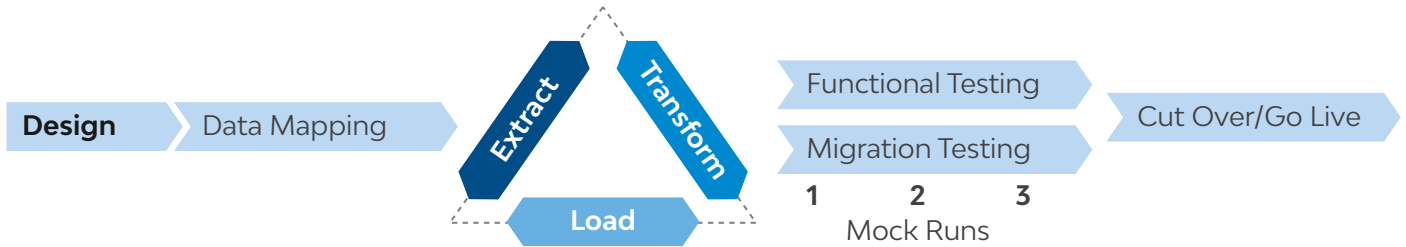
04

Data Mapping

This is a critical activity which is done as part of Requirement Analysis. This activity ensures a comprehensive mapping of data structures and data elements from source to target platform. All the mapping rules are incorporated as part of the data mapping specification and lays the foundation for data conversion.

![LTIMindtree]

Design → Data Mapping

Extract — Transform — Load

Functional Testing

Migration Testing

1   2   3
Mock Runs

Cut Over/Go Live

05
Data Cleansing

One of the challenges of legacy applications is the presence of incorrect, obsolete and redundant data. The target platform will have its own data integrity and security rules and may reject incorrect data on their systems. The data anomalies will get reported in the Data Conversion Reconciliation Reports as part of every mock run. Data cleansing should be owned by business and operations and may include manual or automated updates. This may be a sizable activity and should be planned with an intent of reducing the anomalies reported with each mock run.

06
Operating Model

The platform migration provides an excellent opportunity for operations to review, optimize and automate the current business process based on the target platform offerings. The result is an updated target operating model for business operations and overall platform maintenance

07

Testing Strategy

The testing strategy is critical and should include the following phases of testing:

Data Migration Testing – To test the conversion and reconciliation process.

End-to-End Functional Testing – To be carried out on new data as well as migrated data. Testing should include all interfaces as well.

User Acceptance Testing – To cover end-to-end functional testing to be executed by end users.

Scheme Certification Testing.

Regression testing and Production parallel testing.

The testing strategy should include automation and provisioning of a regression suite.

Note:

A new conversion region must be setup, where the agreed number of internal or external mocks runs and dress rehearsals can be executed. Infrastructure similar to live environment should be built by the responsible team to carry out all the necessary testing.

08

Migration Strategy

The Migration Strategy should be defined early in the planning phase where the bank must decide between a Big Bang vs Phased implementation.

There are various factors that may influence this decision ranging from data volumes, migration complexity and timeline, operational impact, integration complexity, in-flight projects and infrastructure impacts.

Big Bang   or   **Phased Rollout**
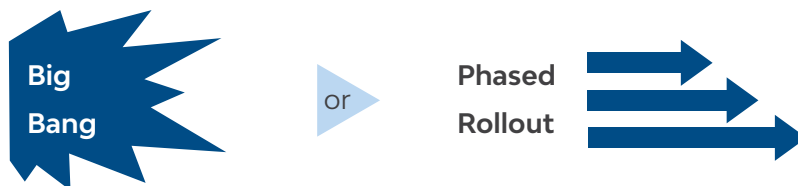
Big Bang	Phased Rollout
All data is migrated as a part of a single migration event. This is a simple approach and is recommended for low data volumes. This approach has a higher risk exposure to business, operations and call center when compared to a phased migration.	Data is migrated in a phased manner. The bank may choose to migrate certain products/BINs first, in order to limit the migration risk exposure. This approach is more complex and may be preferred for high data volumes. There is an overhead for dual platform support on technology teams, interfacing application, business, and operations, until all data is migrated. Although there is additional cost involved, this is the most preferred approach for most of the banks.

Along with the above approaches, historical data migration planning must be considered. Most common and recommended approach is to convert the historical data a week before actual migration, and further only delta portion can be merged on the day of actual migration. This approach is normally taken to avoid the conversion window getting extended.

09

Cutover Planning

This is the planning for final activity that details out all the sequence of events that will occur during actual migration. The cutover period is normally planned over a weekend and should be done in alignment with the schemes and key clients (most importantly corporate clients). The plan should ensure there is minimal impact to customers and business. The cutover plan is refined with every dress rehearsal activity.

Some of the critical activities that have to be planned are:

Operational freeze on the source application.	Deployment of all Interfaces.
Data extraction, transfer and load.	Data reconciliation and verification.
Setting up of new/existing processor control record for the schemes (Visa/MasterCard), which will enable the Issuer system to receive Authorization traffic from schemes.	Customer communication – to advise of system downtime (if any).
Decommissioning of source platform.	Live Proving.

The cutover plan should also include a detailed roll back plan, which should be invoked in the event of a critical migration issue. All impacted applications should be restored to their prior state. The rollback process should be tested prior to cutover date.

10

Pilot Go-Live Testing and Production Assurance

This is a recommended activity, that can boost confidence in the success of the migration, prior to go-live. Under this activity, the bank can confirm its integration with the scheme i.e., connectivity from the target platforms production box, as well as cryptographic key exchanges are tested prior to go-live as production keys are different from test keys. This activity is feasible if the bank has a spare/unused BIN, which can be migrated prior to go-live. A few cards can be setup under this BIN for bank staff users which can be used for pilot-testing.

It is also recommended to perform live production parallel testing for two to four weeks to ensure all necessary functionalities and features have been taken care of in the target platform and there are no customer impacts.

11

Additional Considerations

A **Managing Scope Creep**

In large complex migration programs involving legacy applications, there could be possibility of discovering new gaps (functional, technical, data or interface related) during System Integration Testing or User Acceptance Testing. These gaps can potentially delay the Go-Live date, putting both the project budget and schedule at risk. One way to largely mitigate this risk is to prioritize each gap based on business criticality, making the project team focus mainly on the key requirements gaps required for go-live and deferring the remaining requirements to after go-live. The timelines given **below** are indicative.

1. Critical and impacting large number of customers or key corporate clients → no operational workaround required for Go-Live.

2. High Value → operational workaround possible.
Not required for Go-live.
Solution could be delivered within four to six weeks after go-live.

3. Low Value → can be planned in the next release within three to six months.

B **Hypercare Support**

In large migration programs, the program team should plan for adequate support post go-live i.e., hyper care support for at least six to eight weeks, not only from the core product team, but also from all **surrounding** applications teams including infrastructure support. Depending on the application service criticality, adequate support (for e.g., 24*7 for all customer-facing applications) should be, in place in order to minimize any adverse impacts to customer and business. To be able to provide effective support, there is a need to retain historic data. Hence, banks should have a data retention strategy in their legacy system.