Data
Migration

CONTENT

DATA MIGRATION

From time-to-time various Business organizations implement new Software Application System to replace the functionality currently delivered by one or more legacy systems. Complications arise when there is an attempt to take the information currently maintained by the legacy system and transform it to fit into the new system. More often, the data structure of the legacy systems is different from the new application being implemented, and that difference is not just limited to the table names, field names or attributes or sizes. The types of databases are different and diverse, or the entity relationships definitions in the new system are not compatible with the older legacy application. To the business organizations all the data being held in the legacy system remains critical for their business functions and decision making.

To bring the legacy system data to the new application some Data Conversion must take place, where an initiative, separate or concurrent with the implementation of the new application, is undertaken to convert data from one structural form, used by the legacy application to the structural from required by the newer application.

Often in a Data Conversion process, one would tend to think that any two similar systems that maintain the same sort of data, as they are doing very similar functions should map from one to another without much trouble. But that is not really the case as -

- In Legacy systems, historically, data integrity checks were not strictly enforced, leaving orphan data
- Theoretical design differences exist between hierarchical and relational systems.
- Legacy data may require some data cleansing.

Other Factors contribute to complexity of such projects are

- Need for a well-defined target data structure model.
- In-depth understanding of the functionality of the source data structure.
- Constant changes on the target model design have a knock-on effect on conversion process design.
- Source data quality, if poor, needs to be cleansed to be successfully migrated.
- Degree of complexity of the target model in relation to the source data model.
- The differences in task definitions between the source and target data structures.
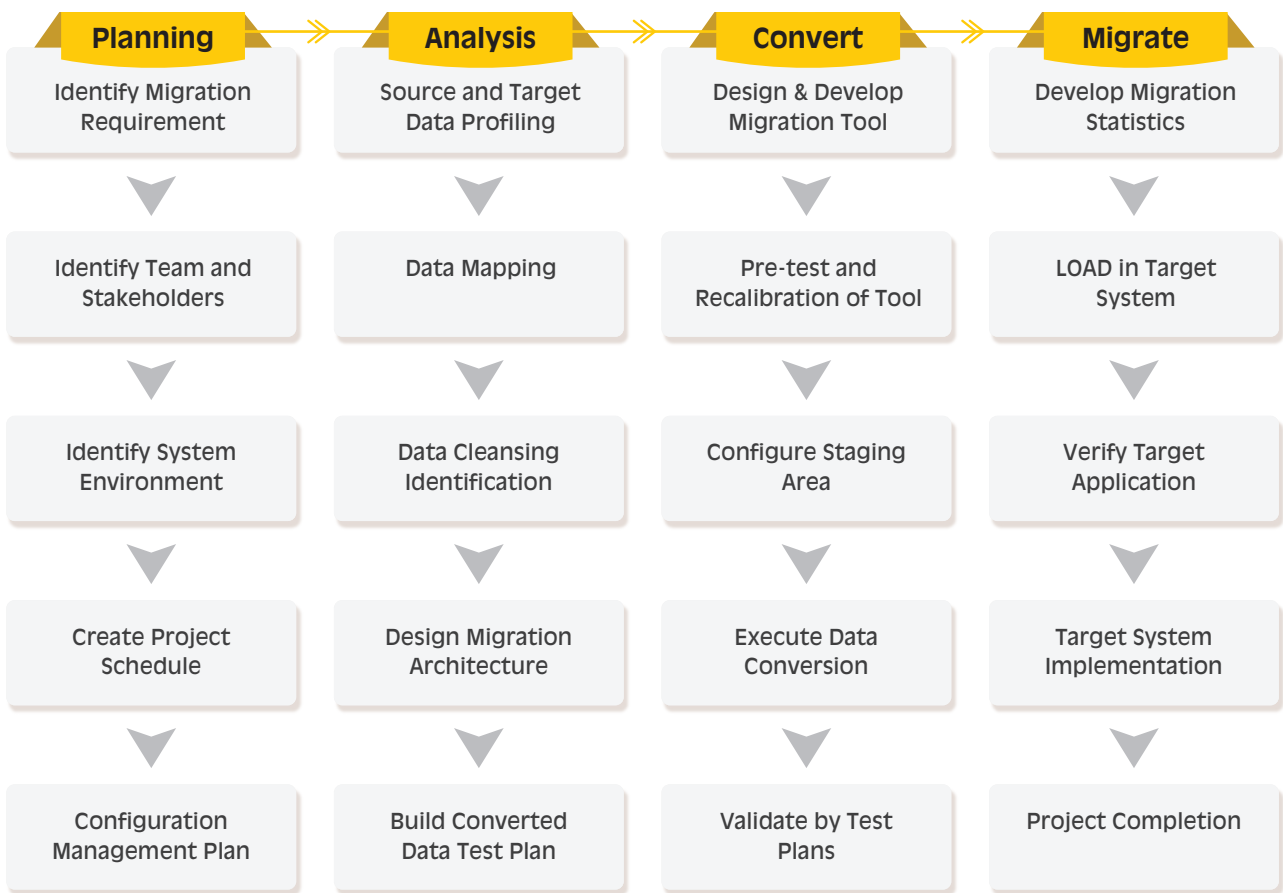
Therefore, it is important to have a sound, methodological approach by which organizations can undertake Data Conversion projects, which will help to confront unpleasant surprises on later stages and resolve those issues fast and effectively.

It is important to note that the methods and processes described within this document is generic in nature. Each project presents its own challenges and opportunities.

OVERVIEW

Conversion and Migration Process

Data Conversion project activities starts with Planning, leading to Analysis and Design, progressing to Conversion of Data, finishing in Migration - where converted data loaded in the target system database.

Planning	Analysis	Convert	Migrate
Identify Migration Requirement	Source and Target Data Profiling	Design & Develop Migration Tool	Develop Migration Statistics
Identify Team and Stakeholders	Data Mapping	Pre-test and Recalibration of Tool	LOAD in Target System
Identify System Environment	Data Cleansing Identification	Configure Staging Area	Verify Target Application
Create Project Schedule	Design Migration Architecture	Execute Data Conversion	Target System Implementation
Configuration Management Plan	Build Converted Data Test Plan	Validate by Test Plans	Project Completion

Issue Management, Risk Assessment, Change Control, Review of Implementation Plan

PLANNING

Conversion and Migration Process

To achieve the goal of a successful Legacy data migration through conversion, it is imperative that a lot of upfront planning happens prior to that move of data, irrespective of the complexity of the task.

The Migration Plan, which is the end result of the planning process, defines

Requirements i.e. what data is moved, where it is moved, how it is moved, when it is moved, and approximately how long the move will take

Team i.e. the Users, Business Analysts, System/Data Analysts, Testers, and Target Application Users

Environment under which the Legacy (Source) and the Target Application system operates, what will be the Data Conversion Stage Area

Schedules - and task distribution, staffing levels

Configuration Management Plan – to maintain control over changes needed due to various reasons

The other objectives of this planning process is to design and document resolutions for events like

» Application downtime,

» Performance degradation,

» Technical incompatibilities, and

» Data corruption/loss.

IDENTIFY MIGRATION REQUIREMENT

Gather information about the proposed conversion/migration task and define the Objectives and the Requirements for the project.

Following information to be collected as part of Requirement definition –

- Whether the project is part of the broader project of implementation of the target application or independent project?

- What data is to be converted and moved – like the nature and usage of the data – more likely by its current Application / Function system and its environment?

- How many source systems are involved?

- What is the size of the legacy system data?

- Where it is moved to – i.e. the target Application System?

- Which are the more critical Data elements with respect to the Target Application System?

- Whether any transformation or elimination will be part of this project?

- How it is moved – if the client has already identified any tool and/or designed, developed any conversion application system?

- When it is to be moved and how long the conversion/move could take?

- Whether the objective is to convert all the Legacy application data at once or the process will be staggered phases?

- Are there any production applications that may conflict with the migration?

- Can the data in legacy system be migrated into the target system without data cleansing?

- What is the level of data cleansing required to maximize the benefit of conversion and migration?

- Whether Historical Data to converted or not – as it could be used by the Target system in future? Historical data is any transactional data that has completed its business cycle and is generally needed for inquiry purposes (statistical analysis, trending, comparison generation and legal implications). Historical data is typically not converted from legacy systems.

- What are the major complications anticipated in the conversion process?

IDENTIFY TEAM AND STAKEHOLDERS

The table below defines the roles and responsibilities of various individuals – participating in Legacy System Data Conversion project.

Business Analysts	• Explains the Business Processes/Functions of the Legacy System and Target System • Identify Data elements / Groups related to various Business Process and Functions • Help to prepare / validate the Data Profile created by Business / Data Analysts • Validate Data Mapping document – developed by Business Analysts and Technical Leads for the Conversion and related reporting • Analyze impacts of the Data Conversion processes and related actions (including Freezing of operation) on the Business.
Technical Lead (Legacy System)	• Assist in team in learning/exploration of the Legacy System Environment • Identify Data elements / Groups related to various Business Process and Functions • Help in developing the Statistics (File/Table Counts, Record/Rows count) for the Legacy System • Assist in Identifying the Legacy System Entities (in terms of File / Table name, Record Layouts) • Resolve queries during Analysis phases • Identify and extract relevant Legacy System Data for Technical Team for Testing of Conversion Programs/ Migration Tool.
Technical Lead (Target System)	• Assist in team learn/explore the Target System Environment • Assist in Identifying the Target System Entities (in terms of File / Table name, Record Layouts) • Resolve queries during Analysis and Mapping phases • Makes technical decisions relating to the configuration of data exchange and transform features – after Data Mapping exercise is completed • Assist in determination Test results based on extracted relevant Legacy System Data for Technical Team for Testing of Conversion Programs/ Migration Tool.

IDENTIFY SYSTEM ENVIRONMENT

There are a few key environmental factors that influence the migration methods selected i.e. how the data is moved.

Technical Team Leads would gather information about the environment in which the Legacy system operates

» Operating System in which the Legacy System is hosted

» Database Type – whether it is RDBMS or Hierarchical Database or Flat File with Indexed files.

» Language used to develop the Legacy System

» How and where data is stored, backed up, and archived

Same details would be gathered for the Target System. Such information helps to determine whether the Legacy System data need to be converted to another format) which increases the complexity.

There other technology considerations, such as:

- How old is the operating system(s) under which data is to be migrated? Some migration tools do not support legacy operating systems.

- What staging area requirements are present, given current technologies and data migration requirements?

- Whether Client would need or want the option to recover quickly from the source disk, or to fall back to the original storage device as a fail-over? This is to determine/design both procedural and technological ways to accomplish that.

- Is a central console needed to manage data migrations across multiple servers?

- Is there a need to control the data migration from a local server – or a remote server? If remote, which protocols must be supported?

- Is there a requirement to throttle or control data flows between servers?

- Which storage tiers are involved?

- Whether any change in the Target System model is likely or not? Should that is possibility – that should be taken into account to analyze the consequences of the change in the eventual conversion process.

Based on the information gathered in response to the queries, listed above, team will enhance the Requirement definition. All these information will greatly influence in determining the Conversion Framework, method and ETL tools to be used or not.

CREATE PROJECT SCHEDULE

At this stage it would not be possible to prepare a detailed realistic schedule as there would be too many unknown parameters. Since this methodology uses a targeted search strategy, the amount of time required to cleanse data is directly proportional to the number and complexity of the mappings and transforms which must be applied to the source data.

An especially tricky part of doing any planning is that the target database is usually a moving target – it is being developed at the same time as the cleansing and conversion software, and is always subject to change. If the new software is not a custom application, but a commercially available package, then it makes the schedule creation process little easier even though that possibility still remains as most of the package system often heavily customized.

Schedule to be created using a project plan and pert chart, showing tasks involved, dependencies between the tasks, and the staff required to perform each task. The schedule will include time to be spent on task of familiarizing with the legacy systems' operations and its data elements and MIS staff with the data cleansing effort, and the tasks required from them.

Key milestones are

- The Migration Plan in place and agreed.
- Project team is formed.
- Completion of Source and Target System Data Profiling
- Completion of Data Mapping Operation and creation of Data Map specification
- Completion of task to identification of Data Cleansing requirement
- Decision on usage of Automated Conversion tool or Development of Customized Solution
- Development of Acceptance criteria
- Identification and procurement of Conversion Tool – if a Tool to be used
- Design, Development and System Testing of Customized Solution – if that is decided
- Test plan is written.
- Creation of the Staging area
- Test run of Conversion Process and validation
- Execution of the Conversion Process
- Reconciliation / data checking reports are run
- Acceptance criteria and performance metrics are evaluated
- A go / no-go decision takes place
- Data Load in the Target System
- Validation by the Target System
- Data issues may be resolved post-migration.

Project manager develops the schedules in consultation with the Client and the Project team member, taking their experience and issues into consideration.

Project schedules would be revised based on the actual time taken to resolve complex issues and procurement of necessary hardware, software resources and team resources.

CONFIGURATION MANAGEMENT PLAN

The Configuration Management (CM) Plan typically specifies how versions of the software will be managed.

Detailed procedure for controlling changes to be defined for the

» Mapping requirement and document
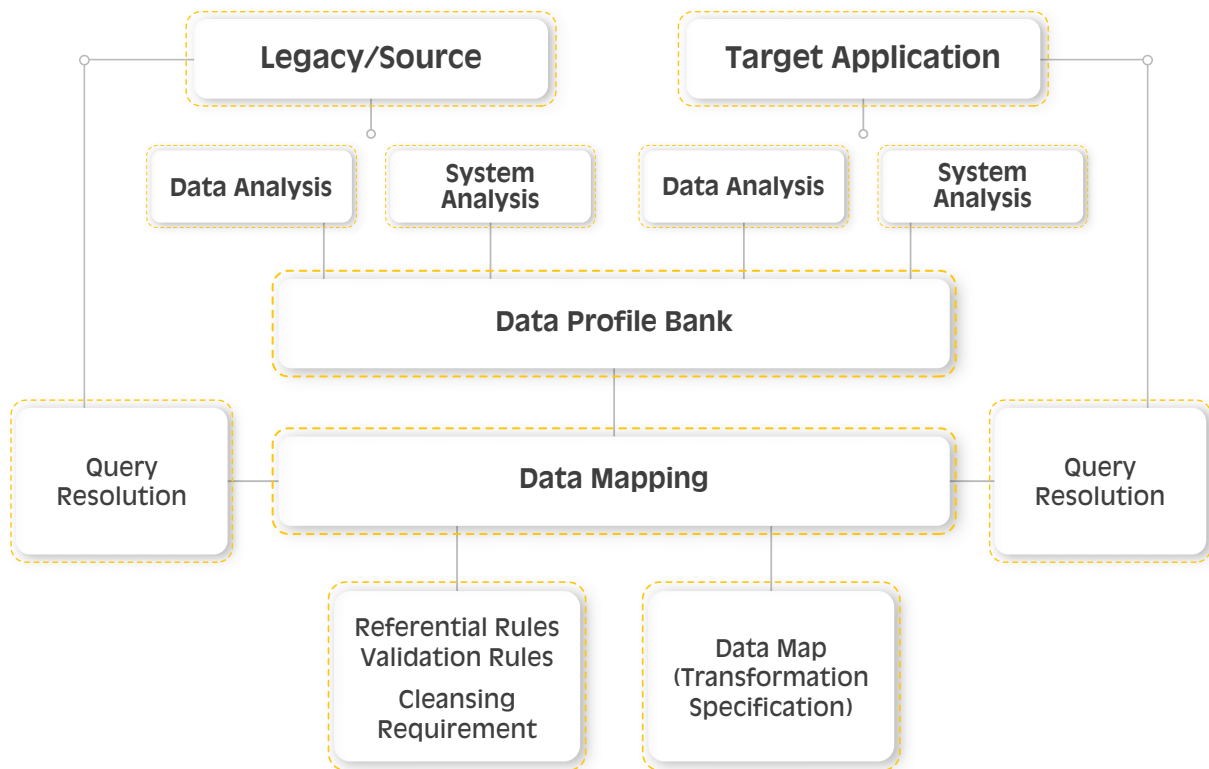» Software changes, in case customized solution to be deployed

A Change Control Board (CCB) could be constituted, consisting of Project manager and Client Project Owner, to review any proposed database changes and its impact. This would be critical to ensure communication between the members of the project teams – whether software development team, Data Analysts/ SMEs

ANALYSIS

In this methodology, the first stage of data conversion and migration is Data Classification, by creating Data Profile for the data elements used in Legacy System(s) and Target Application.

Analysis - Data Profile and Map Creation

- Legacy/Source
- Target Application
- Data Analysis
- System Analysis
- Data Analysis
- System Analysis
- Data Profile Bank
- Query Resolution
- Data Mapping
- Query Resolution
- Referential Rules Validation Rules Cleansing Requirement
- Data Map (Transformation Specification)

SOURCE AND TARGET DATA PROFILING

In this stage, Data Structures model analysis involves an in-depth study (qualitative and quantitative in the case of a legacy system) of both the Legacy and Target systems.

The analysis is undertaken by the Data Analysts/SMEs with the help of Client Business Analysts / Technical Leads, and documented in Data Profile repository.

This repository, which integrates source and target data in mutual format, provides a profile of the various entities - visibility into that data's usage, capacity, and growth patterns. Various interfaces and reports utilizing the to-be-migrated data are also considered. Also considered areas are

» Must data integrity be checked during (not just after) migration?

» How much data will be moving from point to point (server to server)?

» What is the estimation of the amount of transformation and cleansing needed?

» What are the data profiling and data validation rules/phases applicable to the data?

Data profiling process consist of **three** sequential steps with each step building on the information produced in the previous steps. Data sources are profiled in three dimensions: **down columns** (column profiling); **across rows** (dependency profiling); and **across tables** (redundancy profiling).

Column Profiling

Column profiling analyzes the values in each column or field of source data, inferring detailed characteristics for each column, including data type and size, range of values, frequency and distribution of values, cardinality and null and uniqueness characteristics. This step allows analysts to detect and analyze data content quality problems and evaluate discrepancies between the inferred, true Meta data and the documented Meta data.

Dependency Profiling

Dependency profiling analyzes data across rows comparing values in every column with values in every other column and infers all dependency relationships that exist between attributes within each table. Dependency profiling identifies primary keys and whether or not expected dependencies (e.g., those imposed by a new application) are supported by the data. It also identifies "gray-area dependencies" those that are true most of the time, but not all of the time, and are usually an indication of a data quality problem.

SOURCE AND TARGET DATA PROFILING

Redundancy Profiling.

Redundancy profiling compares data between tables of the same or different data sources, determining which columns contain overlapping or identical sets of values. It looks for repeating patterns among an organization's "islands of information". Redundancy profiling identifies attributes containing the same information but with different names and attributes that have the same name but different business meaning. It also helps determine which columns are redundant and can be eliminated and which are necessary to connect information between tables. Redundancy profiling eliminates processing overhead and reduces the probability of error in the target database.

Developing an accurate profile of existing data sources is the essential first step in any successful data migration project. The most significant problem associated with this phase could be if there are frequent changes to the Target Application System Data model. Any change in the target system model would have to be taken into account to analyze the consequences of the change in the eventual conversion process. This renders the whole process to be iterative until a point wherein there is a freeze on the Target System Data structure model or a complete understanding of the legacy system has been reached.

DATA MAPPING

Data Mapping is the process in which each source data elements are assigned to one or more target data element. After having done the analysis, Business Analysts, Data Analysts / SMEs and Technical Leads undertakes data mapping process of identifying and documenting the target field for each of the fields in legacy system.

Objective of this process is to produce a comprehensive mapping between Legacy and Target System. Every data field that is going to be migrated from the source system to the target system must be defined and examined to ensure compliance with field lengths, data types, domain values permitted, system rules, integrity checks and any other possible issues.

Data mapping process would yield the following results:
Gaps in the design of the target system:

»UN-MAPPED TARGET SYSTEM DATA ENTITY
this are the cases where data requirement is not satisfied by Legacy/Source system. Specification could be developed for Deriving/defaulting data values for such cases – specifically when that data is mandatory field in Target System.

»UN-MAPPED LEGACY SYSTEM DATA ENTITY
for which no Target Data element exists in the target system. This will lead to data loss, which might NOT be desirable in all cases.

Data mapping process aims at identification and resolution of such gaps and such issues can be addressed in different ways.

- Ignore the data in legacy system if it is not used.
- If it is business critical data then request for modification of the target system.
- In cases wherein target system cannot be modified and it is business critical data that is   absolutely essential to retain, consider some workaround to incorporate it as a part of some other field that can be easily understood and extracted with business consent.

Data Map (Transformation specification):

At the end of the data mapping process, a detailed document would be in place that would show the target field identified for each legacy field. Apart from identifying the target fields to which the legacy fields are mapped, mapping specifications define the rules to be applied in the conversion process. These rules are commonly known as Transformation rules.

- Referential Rules for Integrity check and domain values permitted
- Validation rules – Application specific Business rules

DATA MAPPING

≫ DATA MAPPING IS AN ITERATIVE PROCESS.

For any change in the design of the target system or change in rule for setting a value of a particular field there is a need to amend the mapping specification reflecting the changes in Transformation rules.

The resulting Data Map (Transformation specification) document would be used later

- In conjunction with third-party data migration tools to extract, scrub, transform and load the data from the old system to the new system or

- Develop a customized Application system to convert the Legacy / Source system data into Target System data model. This will provide essential information to the programmers creating conversion routines to move data from the source to the target database.

DATA CLEANSING

The Data Cleansing process results in relevant and accurate data being converted from the Legacy systems into the Target System.

Data Profile Repository prepared during Data profiling stage (by Data Analysis and System Analysis), to conduct the Data Cleansing – in which Legacy application data to verify that data are correct, complete, consistent and convertible, and it also includes process to identify:

- Unnecessary data records and specification for elimination of these records
- Inaccurate data and design specification / process to correct these data
- Duplicated data and design specification / method to merge those duplicates
- Data needs Reformatting and standardizing - so that it can be converted
- Mandatory Target System Fields without any related field in the Legacy/Source System and resolution / specification for each of those elements

Data Cleansing is critical to the success of any Data Conversion and Migration project. If not undertaken then business processes will not operate as designed. Data cleansing always takes more time and more resources than anyone anticipates. It is for this reason that data cleansing efforts will be launched as early as possible to make subsequent phase of the project easier and avoid delays.

Data cleansing can be accomplished in two different ways
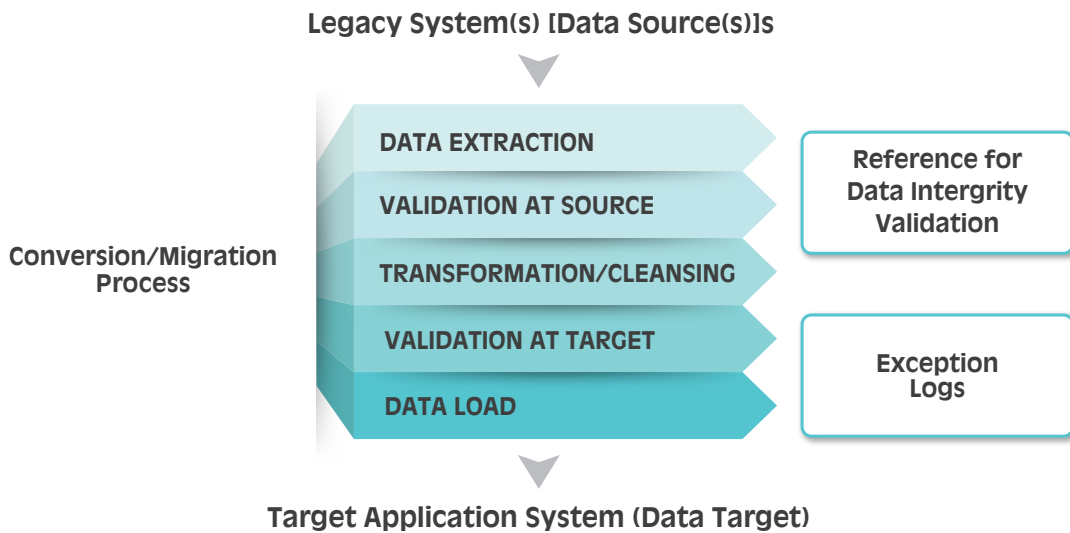
≫ CLEANSING AT THE SOURCE:

This involves the cleansing directly in the production of the existing legacy system or systems. The main advantage of this approach is the exponential reduction in complications. It makes conversion process very simple.

›› CLEANSING THROUGH EXTERNAL MEANS:

This type of cleansing is generally accomplished by spreadsheets. A report is sent to the persons responsible for data cleansing with data that is required to be cleansed. Care should be taken that the spreadsheet contains enough data to make the businessperson understand what they have to cleanse. It should also cater to the technical requirements so that it can be incorporated in the data conversion process with ease. Spreadsheet design is of utmost importance as it can be problematic, if at a later stage, a deficiency were to be identified with the basic design of the spreadsheet which results in an inability to accommodate it in the conversion process.

DESIGN MIGRATION ARCHITECTURE

Analysis - Data Profile and Map Creation

Legacy System(s) [Data Source(s)]s

Conversion/Migration Process

- DATA EXTRACTION
- VALIDATION AT SOURCE
- TRANSFORMATION/CLEANSING
- VALIDATION AT TARGET
- DATA LOAD

Reference for Data Intergrity Validation

Exception Logs

Target Application System (Data Target)

The generic framework for a Data Conversion process consists of following steps

» DATA EXTRACTION

Read and Gather data from source data store(s) into another storage, and if required converted to the data format of the Target System and loaded in Staging Area.

» VALIDATION AND CLEANSING

to confirm content and structure of extracted data in light of business rules and fulfills integration rules based on the referential rules of Target System. Data Cleansing is performed at this time based on requirements identified during Analysis phase.

» TRANSFORMATION

convert the extracted data from its previous form into the target form. Transformation occurs by using Transformation Rules defined in Data Map (Transformation specification) and lookup tables.

» VALIDATION

Target System - confirm content and structure of transformed data is valid for target.

» DATA LOAD

Write the data into the target database, either through script or copying data using system utilities.

BUILD CONVERTED DATA TEST PLAN

After determining the Framework, execution model – i.e. tool route or development of the tailored solution, Test team develops Test Process and Plan based on the Data Map (Transformation Rules).

Testing process in Data Conversion and Migration project could be categorized into two:

Physical errors
are the result of syntactical errors of the scripts / programs, which can be easily identified and resolved.

Logical errors
these are identified and resolved during Test phase. Such errors are result of the quality of the mapping effort.

Based on the on the Data Map (Transformation Rules), Team creates Test Plans – in which team identifies Legacy System data element and determines the Target system element and the expected results based on set of extract to be used for testing. This test plan is to be prepared for each of the data element being converted.

Response to the following queries would be gathered and verified by Testing team

- How many records were expected to be created by the scripts being tested?
- Did the correct number of records get created? If not, why?
- Has the data been loaded into the correct fields?
- Is the data load complete – or are certain fields missing?
- Has the data been formatted correctly?
- Are any post-migration clean-up tasks in order?

The goal of a successful data migration is to keep the length of the deploy phase(s) to a minimum.

CONVERT

» DESIGN & DEVELOP MIGRATION TOOL

The data conversion process can be accomplished by the following method

- Using a data conversion tool
- Scripts developed specifically for the purpose of conversion in the project
- Manual data conversion and migration

The choice of a right conversion tool for a given project is always debatable but the following significant factors form the basis of a rational decision

▶ Cost involved in procuring a tool and having trained personnel to run the tool. Is this cost less than the cost of employing developers to script the conversion process?

▶ Does the tool require any customization? If yes, then the cost and time scales of such a customization effort should be ascertained.

▶ The volume of data to be converted and migrated. If the volume of data is much less, manual data conversion and migration is the best option.

▶ Whether the Legacy System data and Target environment could be hosted on same environment as difference in operating system / data format would involve manual extraction and conversion to target environment.

» PRE-TEST AND RECALIBRATION OF TOOL

Mock Migration

- Conduct dress rehearsals for each planned release. Mock migrations may be partial or complete end-to-end cycles to verify migration procedures and benchmark the cycle times for each migration task.

Pilot Migration

- Complete end-to-end migration in the pilot environment. Coordinate with business users in doing data validation, verify and evaluate the control mechanism and metrics.

Test Extract and Load

- Once the Pilot Migration is completed, test the extract of the migrated data and test the load into the mock target environment.

CONVERT

KEY ACTIVITIES

- Create/verify data element mappings
- Run data extracts from current system(s)
- Create tables, scripts, jobs to automate the extraction
- Address additional data clean-up issues
- Execute application specific customizations
- Run mock migrations
- Load extracts into the new system using ETL tools or SQL loader with bulk loading functions
- Conduct internal data validation checks including business rules and referential integrity checks
- Report exceptions to client team
- Perform data validation

KEY PARTICIPATING GROUPS
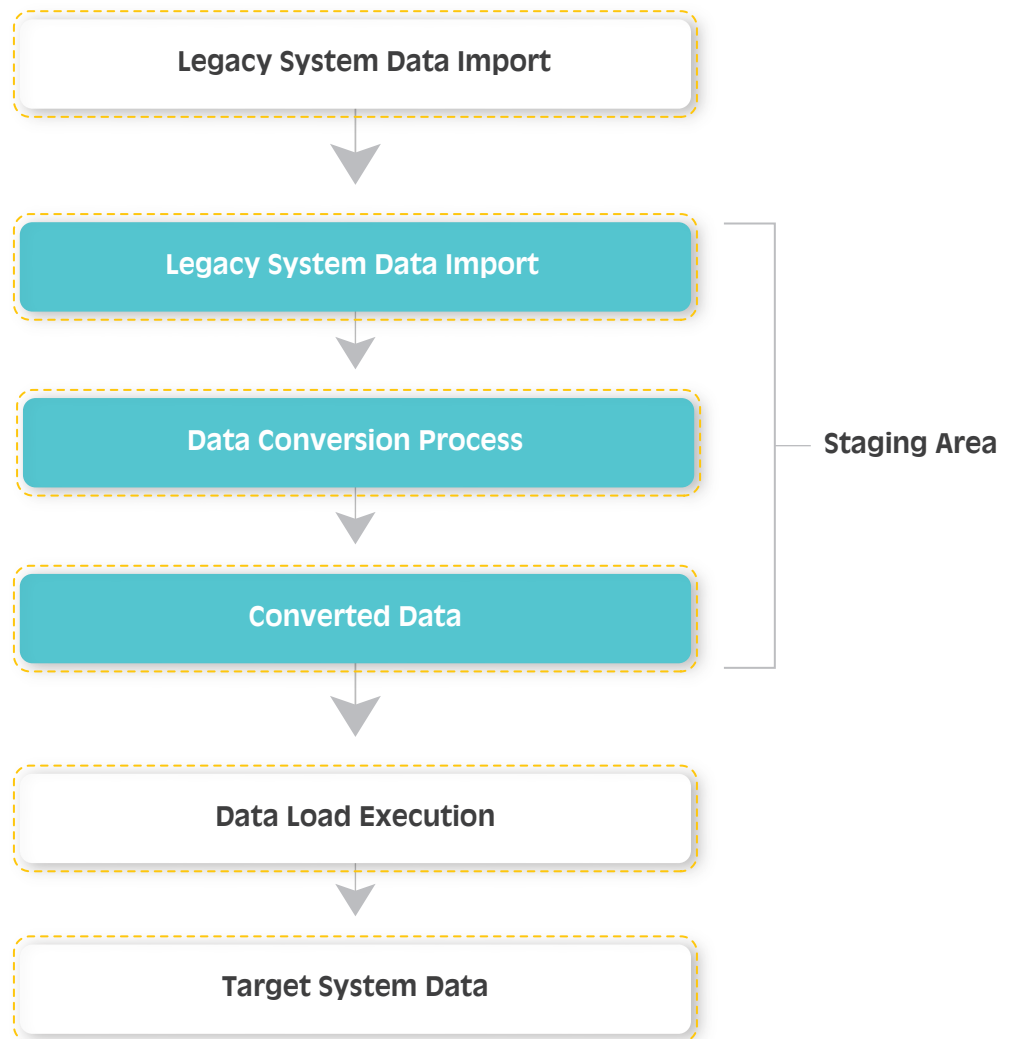
- Data migration team
- Client IS team
- DBA team

DELIVERABLES/OUTPUTS

- Extracts from source system
- Data migration modules, jobs, scripts
- Application loaded with converted data
- Exceptions, alerts and error handling control points.

Configure Staging Area

Data Conversion process is generally executed in a separate area known as the staging area. This area is created, preferably on a separate server, to have the converted data - without disturbing the existing data in the production server.

```
        ┌─────────────────────────────────┐
        │   Legacy System Data Import      │
        └─────────────────────────────────┘
                        ↓
        ┌─────────────────────────────────┐
        │   Legacy System Data Import      │ ┐
        └─────────────────────────────────┘ │
                        ↓                    │
        ┌─────────────────────────────────┐ │
        │    Data Conversion Process       │ ├── Staging Area
        └─────────────────────────────────┘ │
                        ↓                    │
        ┌─────────────────────────────────┐ │
        │        Converted Data            │ ┘
        └─────────────────────────────────┘
                        ↓
        ┌─────────────────────────────────┐
        │     Data Load Execution          │
        └─────────────────────────────────┘
                        ↓
        ┌─────────────────────────────────┐
        │      Target System Data          │
        └─────────────────────────────────┘
```

Execute Data Conversion

Conversion Process Design and Execution

This is the most important phase of data conversion process. Once the data mapping is complete, an overall data conversion process is designed to convert data into a structure required by the target system. Mapping specification serves as an input to this phase. All the rules detailed in mapping specification are incorporated in the conversion process design.

Configure Staging Area

Data Conversion process is generally executed in a separate area known as the staging area. This area is created, preferably on a separate server, to have the converted data - without disturbing the existing data in the production server.

Execute Data Conversion

Conversion Process Design and Execution
This is the most important phase of data conversion process. Once the data mapping is complete, an overall data conversion process is designed to convert data into a structure required by the target system. Mapping specification serves as an input to this phase. All the rules detailed in mapping specification are incorporated in the conversion process design.

Technical Mapping Specifications
Technical mapping specifications are translation of mapping specifications from a technical standpoint prepared by technical team, and these serve as a primary input to code development. Care should be taken to incorporate each and every mapping rule defined in the mapping specification.

Overall Conversion Process Design
Though each data conversion process design varies depending on the system to be migrated the philosophy remains the same, which is explained below.

Legacy System(s) Data Import
Data from the production database of the legacy system is imported to the staging area.

Data Conversion Process
The data conversion process is executed on the imported legacy system data in the staging area.

Converted Data
The data conversion process gives rise to the converted data that is in the required target data structure model.

Final Extract and Load
Load the converted data into the target environment.

➤➤ KEY ACTIVITIES

- Run final extracts from the current system(s)
- Execute specific customizations on target database and execute application specific customizations
- Run pilot migrations
- Load extracts into the new system using ETL tools or SQL loader with bulk loading functions
- Conduct internal data validation checks including business rules and referential integrity checks
- Report exceptions to client team
- Perform data validation

➤➤ KEY PARTICIPATING GROUPS

- Data migration team
- Client IS team
- DBA team

➤➤ DELIVERABLES/OUTPUTS

- Extracts from source system
- Data migration modules, jobs, scripts
- Application loaded with converted data
- Exceptions, alerts and error handling control points.

Validate by Test Plans

Testing the converted data is an important activity of data conversion. The converted data, a result of the conversion process, based on mapping specifications should be tested for the following:

➤➤ Is the entire set of mapping rules defined in mapping specifications document incorporated?

➤➤ Does the data converted follow the data structure model of the target system?

➤➤ Are all the fields mapped from the legacy system converted in the appropriate way in the target system?

➤➤ Are all the constraints of the target system fulfilled or is there is any risk of data load rejections?

➤➤ Did the correct number of records get created in the target system?

➤➤ Was the cleansing data incorporated correctly in the conversion process?

➤➤ Do the legacy and target system records reconcile?

MIGRATE

» DEVELOP MIGRATION STATISTICS

There are specific goals associated with implementing an effective data migration strategy. Primarily, data must be migrated from the source platform to the target platform completely and accurately, and according to company and regulatory policies on information controls and security. This means no dropped or incomplete records, and no data fields that fail validation or other quality controls in the target environment. Another goal of data migration is that the process be done quickly, with as short a downtime window as possible. Finally, the cost of data migration must be manageable, in terms of technology and staff requirements.

There are many metrics that can measure the effectiveness and efficiency of data migrations:

- Number of customizations required
- Percentage of migrated records
- Percentage of migrated tables
- Percentage of data with quality problems
- Number of migration errors
- Migration impact on database size
- Downtime due to migration
- Required staging storage / hardware
- Percentage of reconciliation errors
- Percentage of cleansed data

» DATA LOAD IN TARGET SYSTEM

Though the data in the staging area is present in the target data structure model, it is required to be loaded to target system. This is generally accomplished by data load scripts that pick data from the staging area and load it to the target system directly.
The data load scripts that load data into target system are tested after the loading is done to the target system.

Delta Data

Another important aspect of a data migration project is accounting for the data input to the production legacy system in between the last export of legacy system and the data conversion process run. This is commonly known as delta data. It has to be understood that the window of the conversion process is finite and the legacy system still operates during this window of conversion process execution. There are various methods to address this issue, notable of them are

MIGRATE

▶ If the volume of data input to production legacy system is very low, then a manual data conversion is a wise choice.

▶ If the data volume does not permit for a manual data conversion, then the conversion process is re-run on the delta data in a separate staging area and migrated to production.

Eliminating Delta Data Processing

Even though the delta data is migrated, there still exists another set of delta data unless the use of the legacy system is frozen in the window wherein the data conversion process is executed. A data freeze on production legacy system is not always a very good idea in all the areas of business, as some of them cannot afford a freeze on their systems for a single day. So delta data conversion can be eliminated but such a decision is dependent on the nature of the business in which it is carried out.

Deployment phase

The first step in the design stage is to put together a project plan and structure. As part of this process there should be close analysis of any dependencies in data migrations; where possible, such dependencies and complexities should be reduced to better manage deployment risk. During the deploy phase the following occurs:

- Physical data structures are frozen on source and target.
- Interfaces and processing on source and target are brought down where required.
- Data is staged from the source location.
- Quality reports are run and any data errors or inconsistencies identified.
- Data quality issues are fixed in the staging area.
- A preliminary reconciliation takes place in the staging area, and any reconciling items are investigated and resolved.
- Data is migrated to the target location.

Validation - Target Application

- Reconciliation reports are run.

- Acceptance criteria are checked; if reconciliation errors or other criteria are not met, the system is rolled back to the original data source. Otherwise, interfaces and processing from the source is discontinued and then activated on the target.

MIGRATE

Target System Implementation

Post-Migration Activities - Typical deliverables for the defined phases include:

- Data Migration Approach and Road Map
- Data Source Documents
- Infrastructure Planning and Metrics
- Technical Design Documents
- Failure Routines
- FMEA Document - Failure Mode Execution and Analysis
- Migration Status - Dashboards
- Data Migration Metrics and Control Charts

RISKS AND MITIGATION

Every Data Conversion initiative comes with certain risks and such risks should be planned and mitigation steps need to be prepared.

Some risks worth mentioning are

- Application downtime/business freeze,

- Performance degradation,

- Technical incompatibilities, and

- Data corruption/loss.

- Migration might be done as part of a larger chain of dependencies (operating system upgrades, application upgrades or implementations, database structural changes, etc.) – thereby increasing complexity.

- Data requirements are not clearly defined - data rules for integrity, controls, security, availability, and recoverability are often ill-defined. In the absence of such rules, data is migrated incorrectly.

- Migration acceptance criteria may not be defined.

- Data is often too distributed to be migrated easily.

- Budgets may limit technology options for performing migrations.

- Expertise in data migration and management may not be present.

- Management attention might be insufficient.

- There could be poor support from the vendor(s).

- Not treating the Data Migration initiative as a project unto itself.

Each phase explained in the data conversion and migration process is iterative. No conversion and migration process is accomplished in a single go. It has to be refined as and when a change is triggered at the design level, generally for the target systems and occasionally for the legacy system.

RISK MITIGATION

» ROLLBACK

When importing data into a target system, what happens if the data migration fails? Are we prepared to either utilize existing transaction rollback functionality or do we have capacity to design and build our own if none exists? How do we manage the client expectation in such cases? Do we have a mitigation plan in place? Have we discussed these with the client IS team and business users?

» REPLICATION

The issue being: what happens in case of disaster or irrecoverable system failure? Migrating data to a backup system at the same time as a new target system should be seriously considered to add one more layer of security and ensure that the disaster recovery plan is in place.

» DATA PROFILING

Gain a complete understanding of the content, structure, quality, and integrity of the data of the source system.

» DATA MAPPING

Develop an accurate set of data mapping specifications from the source system to the target system. Write the data into the target database, either through script or copying data using system utilities.

» MIGRATION APPROACH AND ARCHITECTURAL CONSIDERATIONS

Develop an accurate set of data mapping specifications from the source system to the target system. Write the data into the target database, either through script or copying data using system utilities.

» DEVELOPMENT

Selecting an ETL tool to automate the migration process and make it more scalable should be a high-priority item.

» QUALITY ASSURANCE

Conduct mock migrations, pilot migrations before the final migration run; this will ensure that the migration process is robust and trusted.

GOLDEN RULES FOR SUCCESSFUL DATA MIGRATION

▶ **Clearly define the scope of the project.**

▶ **Actively refine the scope of the project through targeted profiling and auditing.**

▶ **Minimize the amount of data to be migrated.**

▶ **Profile and audit all source data in the scope before writing mapping specifications.**

▶ **Define a realistic project budget and timeline, based on knowledge of data issues.**

▶ **Secure sign-off on each stage from a senior business representative.**

▶ **Prioritize with a top-down, target-driven approach.**

▶ **Aim to volume-test all data in the scope as early as possible at the unit level.**

▶ **Allow time for volume testing and issue resolution.**

▶ **Segment the project into manageable, incremental chunks.**