Data Migration & Transformation Solutions Services Overview
Data migration is necessary when an organization decides to use a new computing system or database management system that is incompatible with the current system.

- Architecture Re-engineering
  - Legacy Application
  - Database
  - Platform

- Systems Consolidation

- Data Acquisition

- Data Integration

- Database Technology
Methodology - Introduction

A methodical and disciplined approach is required to ensure a successful data migration.

Migration Project Planning

- Determine data scope
- Create dedicated migration project plan
- Set milestones and deliverables
- Allocate committed resources
- Coordinate with development plan
- Test Planning

Discovery, analysis & requirements

- Validate data sources
- Determine field usage and values
- Analyze data & business rules
- Populate metadata repository
- Assess data quality and integrity
- Develop data mapping requirements

Design, develop, test & implement

- Develop migration & load strategy
- Code and test migration programs
- Conduct system level testing
- Support user acceptance testing
- Perform test migrations
- Develop production execution plan
Methodology

Migration and Transformation Services is based on our six-step data migration methodology.

**Discovery, Analysis & Requirements**

- Gather system and business rules by domain
- Gather data-metrics by source
- Populate metadata repository

- Data mapping
- Integrity analysis
- Develop detailed extract paths
- Develop detailed transformations
- Determine validation rules
- Create business requirements document

**Design, Develop, Test & Implement**

- Develop migration strategy
- Identify software configuration
- Develop testing strategies
- Develop quality assurance strategy
- Develop certification strategy

- Create detailed migration blueprint
- Publish programming standards
- Develop program specifications
- Performance planning and benchmarks
- Design base functions: exception handling, logging, etc.

**Implementation Planning**

- Build base functions
- Build migration engine
- Unit test

- Execute test plans: system, E2E, UAT, volume
- Execute simulated migrations
- Enhance transformation rules

**Phase 1 Analyze**

**Phase 2 Map**

**Phase 3 Design (High Level)**

**Phase 4 Design (Detail)**

**Phase 5 Construct**

**Phase 6 Test & Deploy**
# Methodology (cont’d)

## Issue mgmt., risk assessment, change control, QA, implementation planning

<table>
<thead>
<tr>
<th>Discovery, Analysis, Requirements</th>
<th>Design, Develop, Test, Implement</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1 Analyze</strong></td>
<td><strong>Step 2 Map</strong></td>
</tr>
<tr>
<td>➔ Gather system and business rules by domain</td>
<td>➔ Data mapping</td>
</tr>
<tr>
<td>➔ Gather data-metrics by source</td>
<td>➔ Integrity analysis</td>
</tr>
<tr>
<td>➔ Populate metadata repository</td>
<td>➔ Develop detailed extract paths</td>
</tr>
<tr>
<td>➔ Develop detailed transformations</td>
<td>➔ Develop detailed transformations</td>
</tr>
<tr>
<td>➔ Determine validation rules</td>
<td>➔ Develop migration strategy</td>
</tr>
<tr>
<td>➔ Create business requirements document</td>
<td>➔ Identify software configuration</td>
</tr>
<tr>
<td>➔ Develop migration strategy</td>
<td>➔ Develop testing strategies</td>
</tr>
<tr>
<td>➔ Design base assurance strategy</td>
<td>➔ Develop quality assurance strategy</td>
</tr>
<tr>
<td>➔ Design transformation strategy</td>
<td>➔ Design base functions: exception, logging, etc.</td>
</tr>
<tr>
<td>➔ Create business requirements document</td>
<td>➔ Create detailed migration blueprint</td>
</tr>
<tr>
<td>➔ Publish programming standards</td>
<td>➔ Build base functions</td>
</tr>
<tr>
<td>➔ Develop testing strategies</td>
<td>➔ Build migration engine</td>
</tr>
<tr>
<td>➔ Performace planning and benchmarks</td>
<td>➔ Unit test</td>
</tr>
<tr>
<td>➔ Develop quality assurance strategy</td>
<td>➔ Enhance transformation rules</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Metadata Repository</th>
<th>Requirements Document</th>
<th>Strategic Plans</th>
<th>Specifications</th>
<th>Migration Software</th>
<th>Certified Migration</th>
</tr>
</thead>
</table>

Tasks create deliverables

**Step 3 Design (Strategic)**

- Develop migration strategy
- Identify software configuration
- Develop testing strategies
- Develop quality assurance strategy
- Develop certification strategy
- Create detailed migration blueprint
- Develop program specifications
- Performance planning and benchmarks
- Design base functions: exception, logging, etc.
- Publish programming standards
- Build base functions
- Build migration engine
- Unit test
- Execute test plans: system, E2E, UAT, volume
- Execute simulated migrations
- Enhance transformation rules
Methodology - Project Planning

Project planning for data migrations is often overlooked or deferred, while all focus and energy is directed toward development.

➔ Start data migration planning early.

➔ Create a separate project plan for migration.

➔ Assume the effort will take longer than you expect.

➔ Assume the effort will be more complex than anticipated.

➔ Commit dedicated resources.

➔ Commit dedicated environments.
A “divide and conquer” strategy is employed to develop expertise in the source and target data stores and create manageable units of work.

- Develop source and target expertise by deploying independent analysis teams.

- Create manageable units for analysis

- Designate discreet logical data domains within the target and further sub-divide both teams by target data domains.

- Target data domains can be used during construction and testing.
Define, from the target perspective, the scope of data and the logical classifications of data (domains). This will help to estimate and plan the project.

- Identify target data stores.
- Identify entire set of candidate data sources.
- Classify data by domain.
- Identify in-scope data sources.
- Identify out-of-scope data sources.
- Publish clear and concise scope document.
- Scope may evolve with further analysis.
Typically, data migration is performed by a set of customized programs or scripts that automatically transfer the data.
The process steps of a typical data migration are:

- Extract – read and gather data from source data store(s).
- Source validation – confirm content and structure of extracted data.
- Transformation and Integration - convert the extracted data from its previous form into the target form. Transformation occurs by using rules or lookup tables or by combining the data with other data.
- Target validation – confirm content and structure of transformed data is valid for target.
- Load - write the data into the target database.
Analysis & Mapping

Our methodology ensures a thorough and complete analysis of the source and target data stores which is key to achieving migration success.

- Data analysts identify data characteristics, properties and volumetrics.
- Data quality is measured and assessed to requirements.
- System and business analysts study integration of data and business rules.
- Metadata repository integrates source and target data in mutual format.
- Mapping data from source to target is a three step process
  - Map to the target – driven by the target data requirements, in the context of the target business rules, attempt to satisfy requirements with source data.
  - Un-mapped source – validate requirements for data not included in the target.
  - Un-mapped target – determine course of action when data requirement is not satisfied by source – consider deriving/defaulting data values.
- Business Requirements Document
  - Detailed description of the extract and transformation rules
  - Input to detail design and test plan creation.

Detailed Requirements

Data Mapping

MetaData Repository

Source

Data Analysis System Analysis

Source

Data Analysis System Analysis

Issue Resolution

- Extract methods
- Validation rules
- Transformation rules
- Referential rules
- Load dependencies
- Cleansing requirements
The data quality assessment is performed to gain a complete and thorough understanding of both source and target data. Understanding data will avoid unpredictable transformation results.

- Systematic approach to gaining knowledge about data.
- Identify data anomalies.
- Materially improve the quality of data content.
Improving data quality is ongoing for the duration of the project. One by one, remedies for data anomalies are developed.

» Quality cycle

» Baseline assessment

» Identify data anomalies

» Develop recommendations
  • Alter source data
  • Alter extracted data prior to transformation
  • Fix included in transformation
  • Circumvent, migrate and fix after migration

» Implement data remedies

» Simulated full volume migration (to transformation)

» Improvement monitoring and anomaly tracking
Strategy 7 follows the Department of Defense Total Quality Management Methodology, a hierarchical approach to assessing and improving data quality.

- Total Quality Management Methodology

  - Level 0 – Domain Assessment
  - Level 1 – Completeness and Validity Assessment
  - Level 2 – Structural Integrity Analysis
  - Level 3 – Business Rule Compliance
  - Level 4 – Transformation Rule Compliance
Migration Software Tool

Strategy 7 recommends the use of the DataStage XE software tool to assist, simplify and facilitate the analysis, design and development.

- Simplifies data extraction, transformation, cleansing and loading of migrated customer data
- Both tunable and scalable to accommodate the conversion volumes dictated by the project
- Runs on multiple platforms and communicates with host data bases and files
- Robust and battle tested
- AT&T FA Approved