Data Migration through an Information Development Approach

An Executive Overview

Introducing MIKE2.0
An Open Source Methodology for Information Development

http://www.openmethodology.org
Data Migration through an Information Development Approach

Agenda

- Data Migration through Information Development
  - Executive Summary
  - Business Drivers for Better Data Migration
  - Guiding Principles for Better Data Migration
- MIKE2.0 Methodology
  - 5 phased-approach to Better Business Intelligence
  - Example Task Outputs from Strategy Activities
- Lessons Learned
Migrating from the legacy environment to a new system can be a straightforward activity or be a very complex initiative. Migration can come in many forms:

- A migration from a relatively simply system into another system
- Upgrading a system to a new version through an approach that requires changing the underlying data
- The convergence of multiple systems into a single composite system
- Complex migration from one system to a new system, which requires the migration to be rolled out over a period of time
- Multiple, con-current systems migrations and consolidation efforts. This is referred to as “IT Transformation”.

In most large organisations, migration of Enterprise Systems is very complex. To simplify this complexity, we first aim to understand the scope of the problem and then formulate some initial solution techniques.

The MIKE2.0 Solution for Data Migration provides techniques for measuring the complexity of the Data Migration initiative and determining the activities that are required. It also defines the strategic architectural capabilities as well as high-level solution architecture options for solving different data migration challenges. It then moves into the set of required Foundation Activities, Incremental Design, and Delivery steps. The Executive Summary presents some of the strategy activities.

Similar MIKE2 Solutions include:

- The MIKE2.0 Solution for IT Transformation provides a complementary Solution Approach for dealing with these issues on a very large scale.
- The MIKE2.0 Solution for Master Data Management provides an approach for on running multiple systems in an ongoing fashion that synchronise data sets such as customer, product, employee and locality.
Data Migration through Information Development

**Business Drivers for Better Data Migration**

### Achieve
- Better data quality into the new target systems
- A systematic approach to prioritizing functionality to be moved to the target
- Alignment of related migration initiatives
- A standards-based approach to large-scale systems implementation
- The ability to run co-existent applications to reduce deployment risk
- An ability to trace the flow of information across all systems in the architecture
- Building new analytical systems as part of the operational data migration

### Avoid
- High-risk implementations from a business perspective
- Very complex code that is difficult to manage and is only used “once off”
- Issues with reconciling common data across all systems
- Inefficient software development processes that increase cost and slow delivery
- Inflexible systems and lock-in to specific technologies
- Unnecessary duplication of technology spend

### Change Drivers
- Continuously Changing Business Environment
- Today’s Systems are More Data-Dependent
- Reduced Technical Complexity & Cost
- Architectures Moving to Vendor-Based Systems
Data Migration through Information Development

10 Guiding Principles for Better Data Migration

1. **Measure the complexity of your initiative** - understand your technology requirements based on the sophistication of the migration effort. Determine the full set of capabilities that are required.

2. **Don’t bite off too much at once** – establish an overall architectural blueprint for a complex programme and migrate system functionality a piece at a time. Complex systems can be progressively decommissioned through co-existent applications.

3. **Investigate & fix DQ problems early.** Data quality issues discovered at a late stage often result in programme failures or significant delays. Start with data profiling to identify high risk areas in the early stages of the project. As soon as possible, get your hands on real data.

4. **Use standards to reduce complexity.** Data Migration is simplified through the use of open and common standards related to data, integration and infrastructure.

5. **Build a metadata-driven solution.** A comprehensive approach to metadata management is the key to reducing complexity and promoting reusability across infrastructure. A metadata-driven approach makes it easier for users to understand the meaning of data and to understand the lineage of data across the environment.

6. **Take a diligent approach to testing.** Data Migrations are complex and user expectations will be high, considering the transition is typically from a working system. A systematic testing process should be followed, from initial functional testing to final user acceptance.

7. **Don’t provide an “infrastructure only” release.** Unless the delivery times are short or the infrastructure issues very significant, always aim to provide some form of new business capability in a release - business users often get frustrated with technology-only initiatives. New reporting functionality is often good complement to an infrastructure-focused release.

8. **Make sure the business case is sound.** If a system is going to be replaced, make sure there is a good business reason for it. Also make sure that the business appreciates that there will likely be an initial higher cost of systems in the early stages and that a properly constructed business case actually includes a replacement plan – even for the new system.

9. **Align projects into an overall programme.** If conducting multiple initiatives, there will be many commonalities across the different projects. Align these projects into an overall programme.

10. **Use a detailed, method-based approach.** The MIKE2.0 Methodology provides an open source approach for Information Development.
Data Migration Guiding Principles

*Investigate & Fix DQ Problems Early*

**Eliminate The 11th Hour Fire Fight**

The latter stages of testing is the most expensive and worst time to discover problems with the data. It is late in the process and there is little time to do the analysis and fix the problem. More times than not this has caused project delays. By starting with data profiling, we identify our high risk areas in the early stages of the project.

All problems need to be worked thru in the staging areas prior to further data movement. Therefore, we make as much of an effort as possible to fix the problems while the data is standing still. It costs time and resources to move data. Different types of problems are addressed in each staging area.
Data Migration Guiding Principles

Use Standards to Reduce Complexity

Current State Environments
- Inventory Source Tables
- Inventory Source Attributes
- Inventory Upstream Sources
- Inventory Downstream Targets
- Create as is Domain Model
- Create as is Entity Model

Future State Environments
- Enterprise Apps Data Models
- iODS Data Models

Creating Data Standards
- Rationalize Domains and Entities across Current State and Future State Environments
- Rationalize Attributes across Current State and Future State Environments
- Create Domain Model
- Create Entity Model
- Create Entity Relationship Model
- Create Entity Attribute Model

Attribute Mappings
- Finance
- iODS
- DW
- Customer

Initial Common Data Standards and creation of:
- Initial DQ Program
- Initial Data Ownership Model
- Initial Data Management Governance Processes

Map in all Application Environments to the Enterprise Standard.
What is MIKE2.0?

- MIKE stands for Method for an Integrated Knowledge Environment
- MIKE2.0 is our comprehensive methodology for Enterprise Information Management
- MIKE2.0 brings together important concepts around Open Source and Web 2.0
- The open source version of MIKE2.0 is available at: http://www.openmethodology.org

Key Constructs within MIKE2.0

- SAFE (Strategic Architecture for the Federated Enterprise) is the architecture framework for the MIKE2.0 Methodology
- Information Development is the key conceptual construct for MIKE2.0- develop your information just like applications

MIKE2.0 provides a Comprehensive, Modern Approach

- Scope covers Enterprise Information Management, but goes into detail in areas to be used for more tactical projects
- Architecturally-driven approach that starts at the strategic conceptual level, goes to solution architecture
- A comprehensive approach to Data Governance, Architecture and strategic Information Management

MIKE2.0 provides a Collaborative, Open Source Methodology for Information Development

- Balances adding new content with release stability through a method that is easier to navigate and understand
- Allows non-BearingPoint users to contribute
- Links into existing project assets on our internal knowledge management systems
- Unique approach, we would like to make this “the standard” in the new area of Information Development
The MIKE2.0 Methodology can be applied to solve different types of Data Migration problems.

- For simple migration activities, not all activities from the Overall Implementation Guide are required. The complete migration may take only a single release.
- For complex migration scenarios, most activities will be required and will be implemented over multiple increments. Complex migration scenarios often require very sophisticated architectural capabilities.
- Most migrations of Enterprise Applications are very complex processes.

The following pages go through some of the initial strategy activities in MIKE2.0 that:

- Help introduce the overall approach to Data Migration and how it is applied depending on the complexity of the problem
- Provide an example of one of the tasks in the initial current-state assessment
- Propose some high-level solution architecture options that can be applied to different migration scenarios
- Provide an approach for prioritizing complex migrations, based on business priorities and complexity of the implementation
MIKE2.0 Methodology: Phase Overview

The 5 Phases of MIKE2.0

Information Development through the 5 Phases of MIKE2.0

Strategic Programme Blueprint is done once

Phase 1  Business Assessment
Phase 2  Technology Assessment

Continuous Implementation Phases

Increment 1  Increment 2  Increment 3

Design  Development  Deploy  Operate

Begin Next Increment

Phase 3, 4, 5

Improved Governance and Operating Model
MIKE2.0 Methodology: Phase Overview

Typical Activities Conducted as part of the Strategy Phases

**Phase 1 – Business Assessment and Strategy Definition Blueprint**

1.1 Strategic Mobilisation
1.2 Enterprise Information Management Awareness
1.3 Overall Business Strategy for Information Development
1.4 Organisational QuickScan for Information Development
1.5 Future State Vision for Information Management
1.6 Data Governance Sponsorship and Scope
1.7 Initial Data Governance Organisation
1.8 Business Blueprint Completion
1.9 Programme Review

**Phase 2 - Technology Assessment and Selection Blueprint**

2.1 Strategic Requirements for BI Application Development
2.2 Strategic Requirements for Technology Backplane Development
2.3 Strategic Non-Functional Requirements
2.4 Current-State Logical Architecture
2.5 Future-State Logical Architecture and Gap Analysis
2.6 Future-State Physical Architecture and Vendor Selection
2.7 Data Governance Policies
2.8 Data Standards
2.9 Software Development Lifecycle Preparation
2.10 Metadata Driven Architecture
2.11 Technology Blueprint Completion
MIKE2.0 Methodology: Task Overview

Task 1.2.1 Develop and Initiate Information Management Orientation

<table>
<thead>
<tr>
<th>Activity 1.2 Enterprise Information Management Awareness</th>
<th>Responsible</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2.1 Assess Team's Understanding of Information Management Concepts</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task 1.2.2 Develop and Initiate Information Management Orientation</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Phase 1 – Business Assessment and Strategy Definition Blueprint

1.1 Strategic Mobilisation
1.2 Enterprise Information Management Awareness
1.3 Overall Business Strategy for Information Development
1.4 Organisational QuickScan for Information Development
1.5 Future State Vision for Information Management
1.6 Data Governance Sponsorship and Scope
1.7 Initial Data Governance Organisation
1.8 Business Blueprint Completion
1.9 Programme Review
End to End Tool Based Transformation Capabilities

<table>
<thead>
<tr>
<th>Acquisition Stage</th>
<th>Consolidation Stage</th>
<th>Move Stage</th>
<th>Post Move Stage</th>
</tr>
</thead>
<tbody>
<tr>
<td>The acquisition stage is focused on the sourcing of data from the producer. The data is placed in a staging area where the data is scanned and assessed. Judgments are made on the complexity of data quality issues and initially identified data quality problems are addressed.</td>
<td>The consolidation stage focuses on attribute rationalisation into an integrated data store that may be required to bring data together from multiple systems. Key transformations occur and further steps are required for re-engineering data. The data and processes are prepared for migration to the Move environment. Considerable collaboration is needed in those areas where decommissioning occurs.</td>
<td>The move stage focuses on moving the data and application capabilities that have been developed to the production environment. The move stage has a staging area that is as close to production as possible. Final steps around data quality improvement are done this environment.</td>
<td>The post move stage is focused on the data transformations and quality aspects that were best done after the move to production (but before the system goes live) such as environment specific data or reference data. Additional process changes or software upgrades may also be required. The skills and toolsets used are the same as the ones used in the prior phases. Attention is paid to the ongoing use of the interfaces created during the transition process.</td>
</tr>
</tbody>
</table>
MIKE2.0 Methodology: Task Overview

Task 1.2.1 Develop and Initiate Information Management Orientation

Introductory Concept: Depending on the level of complexity – different migration orientations are required. At an introductory level, MIKE2.0 classifies orientations as “lite”, “medium” and “heavy”.

Strategy

The migration effort can start at any one of the orientations. An enterprise transformation may have parts of the effort start concurrently at each of the orientations.

A migration effort may start at the Lite orientation and decide to move to the next orientation (medium) on the fly as the results of the data scans are examined.

Further some of the data rationalization and Data Quality work may be done in the target environment after the Move Phase.

End to End Tool Based Transformation Capabilities

<table>
<thead>
<tr>
<th>Acquisition Stage</th>
<th>Consolidation Stage</th>
<th>Move Stage</th>
<th>Post Move Stage</th>
</tr>
</thead>
</table>

Orientation - Migration Lite

A lite migration scenario is straightforward: it typically involves loading data from a single source into a single target. Few changes are required in terms of data quality improvement; mapping is relatively simple as is the application functionality to be enabled. Data integration may be on the back-end of systems and will likely be a once-off, “big bang”.

Orientation - Migration Medium

A medium migration scenario may involve loading data from a single source into a single target or to multiple systems. Data quality improvement will be performed through multiple iterations, transformation issues may be significant and integration into a common data model is typically complex.

Orientation - Migration Heavy

A heavy migration scenario typically involves providing a solution for application co-existence that allows multiple systems to be run in parallel. The integration framework is formulated so the current-state and future-state can work together. The model for a heavy migration scenario is representative of an organisation in IT Transformation.

As heavy migrations are long running and involve a significant data integration effort, it is useful to build a parallel analytical environment to attain a “vertical” view of information.
**Introductory Concept:** Different capabilities are required from the architecture depending on the level of sophistication required. Capabilities are first defined at the strategic component level in Activity 1.5.

<table>
<thead>
<tr>
<th>Capability / Skills</th>
<th>Orientation - Lite</th>
<th>Orientation - Medium</th>
<th>Orientation - Heavy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table and Attribute Assessment</td>
<td>Performed</td>
<td>Performed</td>
<td>Performed</td>
</tr>
<tr>
<td>Relationship Assessment</td>
<td>Direct copy of source systems</td>
<td>Key Integrity validated</td>
<td>Referential Integrity required</td>
</tr>
<tr>
<td>Data Replication</td>
<td>None</td>
<td>None</td>
<td>Multiple Targets</td>
</tr>
<tr>
<td>Data Transfer</td>
<td>To Target</td>
<td>To Target</td>
<td>Target / Downstream</td>
</tr>
<tr>
<td>Data Synchronization</td>
<td>None</td>
<td>None</td>
<td>For Interfaces</td>
</tr>
<tr>
<td>Data Transformation</td>
<td>Modest</td>
<td>Significant to similar structures</td>
<td>Major Activity</td>
</tr>
<tr>
<td>Data Mapping</td>
<td>Minimal</td>
<td>SME supported</td>
<td>Major Activity</td>
</tr>
<tr>
<td>Data Standardization</td>
<td>None</td>
<td>Key Attributes</td>
<td>All attributes</td>
</tr>
<tr>
<td>Pattern Analysis and Parsing</td>
<td>None</td>
<td>None</td>
<td>Yes</td>
</tr>
<tr>
<td>Record Matching</td>
<td>None</td>
<td>Based on similar IDs</td>
<td>IDs and pattern matching</td>
</tr>
<tr>
<td>Record De-Duping</td>
<td>None</td>
<td>None</td>
<td>Yes</td>
</tr>
<tr>
<td>Out of Box Business Rules</td>
<td>As appropriate</td>
<td>As appropriate</td>
<td>As appropriate</td>
</tr>
<tr>
<td>Configure Complex Rules</td>
<td>None</td>
<td>Application</td>
<td>Application / Infrastructure</td>
</tr>
<tr>
<td>Out of the Box Interfaces</td>
<td>As appropriate</td>
<td>As appropriate</td>
<td>As appropriate</td>
</tr>
<tr>
<td>Configure Custom Interfaces</td>
<td>None</td>
<td>Application</td>
<td>Application / Infrastructure</td>
</tr>
<tr>
<td>Data Governance Process Model</td>
<td>Documented in High Level Form</td>
<td>Key or Lynchpin Processes only</td>
<td>End to End Models</td>
</tr>
<tr>
<td>Database Independent Functions</td>
<td>As existed in Source</td>
<td>Few Custom APIs</td>
<td>Infrastructure Services</td>
</tr>
<tr>
<td>Reporting</td>
<td>Data Move Metrics only</td>
<td>DQ and DM metrics</td>
<td>Reporting as a Service</td>
</tr>
<tr>
<td>‘Active’ MetaData Repository</td>
<td>Specific and 'physical'</td>
<td>Multiple Passive dictionaries</td>
<td>Initial implementation</td>
</tr>
</tbody>
</table>
MIKE2.0 Methodology: Task Overview

Task 1.4.1 Assess Current-State Application Portfolio

Activity 1.4 Organisational QuickScan for Information Development | Responsible | Status
--- | --- | ---
Task 1.4.1 Assess Current-State Application Portfolio |  |  
Task 1.4.2 Assess Information Maturity |  |  
Task 1.4.3 Assess Economic Value of Information |  |  
Task 1.4.4 Assess Infrastructure Maturity |  |  
Task 1.4.5 Assess Key Current-State Information Processes |  |  
Task 1.4.6 Define Current-State Conceptual Architecture |  |  
Task 1.4.7 Assess Current-State People Skills |  |  
Task 1.4.8 Assess Current-State Organisational Structure |  |  
Task 1.4.9 Assemble Findings on People, Organisation and its Capabilities |  |  

Phase 1 – Business Assessment and Strategy Definition Blueprint

1.1 Strategic Mobilisation
1.2 Enterprise Information Management Awareness
1.3 Overall Business Strategy for Information Development
1.4 Organisational QuickScan for Information Development
1.5 Future State Vision for Information Management
1.6 Data Governance Sponsorship and Scope
1.7 Initial Data Governance Organisation
1.8 Business Blueprint Completion
1.9 Programme Review
The Application Portfolio documents major systems and their functionality:

- From an Information Development perspective, the deliverable should only be to get a quick overview of the system to understand major application functionality.

- The Application Portfolio also documents the system owner and any expected changes it is expected to undergo during the period of executing on the Blueprint vision.

- It is focused at a systems level, as opposed to infrastructure and information.

- Time for this task may vary greatly depending on the existing artefacts.

Ideally, this content is stored in a structured repository as opposed to an unstructured document form.

<table>
<thead>
<tr>
<th>System Name</th>
<th>Description</th>
<th>Platform</th>
<th>Level 1 &amp; 2 Functions</th>
<th>Application Complexity</th>
<th>Divisions</th>
<th>Inter-Divisional Complexity</th>
<th>Issues &amp; Limitations</th>
<th>Application Life Expectancy</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Brief statement that gives an overview of the system</td>
<td>Technologies the system consists of - application, database, operating system, programming language</td>
<td>This attribute maps the application to the end-to-end process model for both Level 1 (Functions) and Level 2 (Processes). For systems that are repeated across the business, this attribute is repeated for each instance.</td>
<td>Rating of the complexity of an application. The ratings used are as follows: - Low - These systems are relatively simple, use a simple database or small number of files and are reasonably well documented. Low complexity systems may also include &quot;black box&quot; systems where support and documentation is provided by a vendor, the product is stable (with infrequent new releases), and there is little to no customization. These black box systems would generally apply more to areas such as production than to customer-facing or financial systems. - Medium - These systems are generally more substantial in functionality than low complexity systems, are reasonably simple in functionality, but are poorly documented and/or use a number of related databases or files. Off-the-shelf vendor products may be classified as Medium complexity systems if they are generally well documented with some customisation. This classification may also represent a grouping of multiple low complexity applications. - High - Highly complex systems that use complex data structures, require trained staff to configure and support. It would also include those that are poorly documented and difficult to maintain, while containing a significant amount of business functionality. Major off-the-shelf vendor systems that have been heavily customized, have been tailored by the vendor to divisional requirements or for which upgrades would be difficult would likely be classified as High complexity.</td>
<td>Divisions/business clusters supported by the system</td>
<td>Rating of the inter-divisional complexity due to variations in the instances of systems supporting different divisions. Rating of the complexity of an application. The ratings used are as follows: - Low - the system is used in 1 division or in exactly the same way in 2 or more divisions. - Medium - there are multiple instances of the system but the differences between divisions are largely configuration changes. - High - there are multiple instances of the system and there are significant differences in code and configuration of these systems.</td>
<td>Key problems associated with the system</td>
<td>This attribute may contain the envisaged life expectancy for the system – e.g. whether it will be decommissioned or be part of the strategic architecture. Ratings may also be assigned to the system in terms of application life expectancy, using the following model: - Null - Unknown - Low - This system will be replaced in the near term (&lt; 2 years) - Medium - This system will be replaced in the long term (2 – 5 years) - High - Strategic, long-term application (&gt;5 years)</td>
<td>Additional comments related to the system – e.g. point-of-contact, open questions.</td>
</tr>
</tbody>
</table>
MIKE2.0 Methodology: Task Overview

Task 1.5.10 High Level Solution Architecture Options

<table>
<thead>
<tr>
<th>Activity 1.5 Future-State Vision for Information Management</th>
<th>Responsible</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.5.1 Introduce Leading Business Practices for Information Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5.2 Define Future-State Business Alternatives</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5.3 Define Information Management Guiding Principles</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5.4 Define Technology Architecture Guiding Principles</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5.5 Define IT Guiding Principles (Technology Backplane Delivery Principles)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5.6 Define Future-State Information Process Model</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5.7 Define Future-State Conceptual Data Model</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5.8 Define Future-State Conceptual Architecture</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5.9 Define Source-to-Target Matrix</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5.10 Define High-Level Recommendations for Solution Architecture</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
MIKE2.0 Methodology: Task Overview

Task 1.5.10 High Level Solution Architecture Options

A lite migration scenario is straightforward: it typically involves loading data from a single source into a single target. Few changes are required in terms of data quality improvement; mapping is relatively simple as is the application functionality to be enabled. Data integration may be on the back-end of systems and will likely be a once-off, “big bang”.

Below is a high level solution option for a lite migration scenario.
A medium migration scenario may involve loading data from a single source into a single target or to multiple systems. Data quality improvement will be performed through multiple iterations, transformation issues may be significant and integration into a common data model is typically complex.

Data migration may involve multiple iterations through a gradual roll-out of capabilities. Below is a high level solution option for a medium migration scenario.
A heavy migration scenario will require a comprehensive strategy that develops a vision for people, process, organisation and technology. A **heavy Application Co-Existence scenario** shows how multiple systems can be run in parallel so the current-state and future-state can work together. Below is a **high level solution option for a heavy migration scenario** and representative of an organisation in IT Transformation.
MIKE2.0 Methodology: Task Overview

Task 1.8.1 Prioritise Requirements and Identify Immediate Opps

Activity 1.8 Business Blueprint Completion

<table>
<thead>
<tr>
<th>Task</th>
<th>Responsible</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task 1.8.1 Prioritise Requirements and Identify Immediate Work Opportunities</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task 1.8.2 Define High-Level Programme Plan</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task 1.8.3 Develop High-Level Business Case</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task 1.8.4 Assemble Key Messages to Complete Business Blueprint</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
For Data Migration initiatives that involve replacement of a number of systems, a key part of prioritisation involves balancing the desire for new business capabilities with the complexity of their implementation.

High Level Project Estimating Factors Include:

- The complexity of the current-state environment
- The number of critical business functions to be enabled
- The level of technology sophistication that is required
- The number of systems to be migrated
- Amount of data within these systems to be migrated
- Level of documentation on the system
- Availability of Subject Matter Experts
- Complexity of system interfaces
- Quality of the data within the system

A key aspect of the MIKE2.0 approach is determining these Estimating Factors. The Estimating Model available as part of MIKE2.0 is described on the following pages.
There are many factors that will be used to estimate the time and resources required for delivering a Data Migration project. The model below can be used to make a quantitative estimate on the complexity of the project and to weigh business priorities. If multiple migrations are to take place across a large transformation programme, this model can be used to help prioritize the sequencing of the overall implementation.

These questions should be asked relative to an application or application cluster by senior staff. A large-scale transformation programme may have multiple applications or application clusters. A starter set of sample questions is listed below.

**Criteria for Assessing Difficulty**

1. Number and size of databases in Application
2. Number of Tables per database
3. Total Number of Attributes
4. # of attributes that have had multiple definitions over time
5. # of attributes in terms of synonyms and antonyms
6. Number of DB dependent processes
7. Number of one time Interfaces
8. Number of ongoing Interfaces
9. Number of Data Quality Problems and Issues to fix
10. Knowledge / Documentation of Data Quality issues
11. Ease of de-duping similar entities in the same DB
12. Ease of matching same entity records across multiple DBs
13. Completeness of the functional documentation
14. Availability of Subject Matter Experts (SMEs)
15. Maturity of the Enterprise in Managing their data

**Alignment with Business Enablers**

1. Degree to which system functions align with base capabilities
2. Degree to which system functions align with enhanced capabilities focused on the new business model
3. Degree to which the system addresses high priority customer segment growth
4. Degree to which the system addresses high priority customer segment retention
5. Degree to which the system addresses high priority areas of product growth
6. Degree to which the system addresses high priority areas of product stabilization
7. Degree to which the system is cost effective (i.e., cost takeout)
8. Degree to which the system is flexible to adding capabilities
### Difficulty Criteria

1. Number and size of databases in Application
2. Number of Tables per database
3. Total Number of Attributes
4. % of attributes that have had multiple definitions over time
5. % of attributes in terms of synonyms and antonyms
6. Number of DB dependent processes
7. Number of one time Interfaces
8. Number of ongoing Interfaces
9. Number of Data Quality problems and issues to fix
10. Knowledge / Documentation of Data Quality issues
11. Ease of de-duping similar entities in the same DB
12. Ease of matching same entity records across multiple DBs
13. Completeness of the functional documentation
14. Availability of Subject Matter Experts (SMEs)
15. Maturity of the Enterprise in Managing their data

<table>
<thead>
<tr>
<th></th>
<th>Small</th>
<th>2</th>
<th>Medium</th>
<th>4</th>
<th>Large</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Low</th>
<th>2</th>
<th>Average</th>
<th>4</th>
<th>High</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
MIKE2.0 Methodology: Task Overview

Task 1.8.1 Prioritise Requirements and Identify Immediate Opps

Alignment with Key Business Enablers

1. Degree to which system functions align with base capabilities
2. Degree to which system functions align with enhanced capabilities focused on the new business model
3. Degree to which the system addresses high priority customer segment growth
4. Degree to which the system addresses high priority customer segment retention
5. Degree to which the system addresses high priority areas of product growth
6. Degree to which the system addresses high priority areas of product stabilization
7. Degree to which the system is cost effective (i.e., cost takeout)
8. Degree to which the system is flexible to adding capabilities
### MIKE2.0 Methodology: Task Overview

**Task 1.8.1 Prioritise Requirements and Identify Immediate Opps**

#### Business Alignment

1. Degree to which system functions align with base capabilities
2. Degree to which system functions align with enhanced capabilities focused on the new business model
3. Degree to which the system addresses high priority customer segment growth
4. Degree to which the system addresses high priority customer segment retention
5. Degree to which the system addresses high priority areas of product growth
6. Degree to which the system addresses high priority areas of product stabilization
7. Degree to which the system is cost effective (i.e., cost takeout)
8. Degree to which the system is flexible to adding capabilities

#### Scoring Formulas

1. Number and size of Data Bases in Application
2. Number of Tables per Data Base
3. Total Number of Attributes
4. # of attributes that have had multiple definitions over time
5. # of attributes in terms of synonyms and antonyms
6. Number of DB dependent processes
7. Number of one time Interfaces
8. Number of ongoing Interfaces
9. Number of Data Quality Problems and Issues to fix
10. Knowledge / Documentation of Data Quality issues
11. Ease of de-duping similar entities in the same DB
12. Ease of matching same entity records across multiple DBs
13. Completeness of the functional documentation
14. Availability of Subject Matter Experts (SMEs)
15. Maturity of the Enterprise in Managing their data

#### Difficulty Index

1. Number and size of Data Bases in Application
2. Number of Tables per Data Base
3. Total Number of Attributes
4. # of attributes that have had multiple definitions over time
5. # of attributes in terms of synonyms and antonyms
6. Number of DB dependent processes
7. Number of one time Interfaces
8. Number of ongoing Interfaces
9. Number of Data Quality Problems and Issues to fix
10. Knowledge / Documentation of Data Quality issues
11. Ease of de-duping similar entities in the same DB
12. Ease of matching same entity records across multiple DBs
13. Completeness of the functional documentation
14. Availability of Subject Matter Experts (SMEs)
15. Maturity of the Enterprise in Managing their data

### MIKE2.0 Methodology: Task Overview

**Task 1.8.1 Prioritise Requirements and Identify Immediate Opps**

#### Business Alignment

1. Degree to which system functions align with base capabilities
2. Degree to which system functions align with enhanced capabilities focused on the new business model
3. Degree to which the system addresses high priority customer segment growth
4. Degree to which the system addresses high priority customer segment retention
5. Degree to which the system addresses high priority areas of product growth
6. Degree to which the system addresses high priority areas of product stabilization
7. Degree to which the system is cost effective (i.e., cost takeout)
8. Degree to which the system is flexible to adding capabilities

#### Scoring Formulas

1. Number and size of Data Bases in Application
2. Number of Tables per Data Base
3. Total Number of Attributes
4. # of attributes that have had multiple definitions over time
5. # of attributes in terms of synonyms and antonyms
6. Number of DB dependent processes
7. Number of one time Interfaces
8. Number of ongoing Interfaces
9. Number of Data Quality Problems and Issues to fix
10. Knowledge / Documentation of Data Quality issues
11. Ease of de-duping similar entities in the same DB
12. Ease of matching same entity records across multiple DBs
13. Completeness of the functional documentation
14. Availability of Subject Matter Experts (SMEs)
15. Maturity of the Enterprise in Managing their data

### MIKE2.0 Methodology: Task Overview

**Task 1.8.1 Prioritise Requirements and Identify Immediate Opps**

#### Business Alignment

1. Degree to which system functions align with base capabilities
2. Degree to which system functions align with enhanced capabilities focused on the new business model
3. Degree to which the system addresses high priority customer segment growth
4. Degree to which the system addresses high priority customer segment retention
5. Degree to which the system addresses high priority areas of product growth
6. Degree to which the system addresses high priority areas of product stabilization
7. Degree to which the system is cost effective (i.e., cost takeout)
8. Degree to which the system is flexible to adding capabilities

#### Scoring Formulas

1. Number and size of Data Bases in Application
2. Number of Tables per Data Base
3. Total Number of Attributes
4. # of attributes that have had multiple definitions over time
5. # of attributes in terms of synonyms and antonyms
6. Number of DB dependent processes
7. Number of one time Interfaces
8. Number of ongoing Interfaces
9. Number of Data Quality Problems and Issues to fix
10. Knowledge / Documentation of Data Quality issues
11. Ease of de-duping similar entities in the same DB
12. Ease of matching same entity records across multiple DBs
13. Completeness of the functional documentation
14. Availability of Subject Matter Experts (SMEs)
15. Maturity of the Enterprise in Managing their data

### MIKE2.0 Methodology: Task Overview

**Task 1.8.1 Prioritise Requirements and Identify Immediate Opps**

#### Business Alignment

1. Degree to which system functions align with base capabilities
2. Degree to which system functions align with enhanced capabilities focused on the new business model
3. Degree to which the system addresses high priority customer segment growth
4. Degree to which the system addresses high priority customer segment retention
5. Degree to which the system addresses high priority areas of product growth
6. Degree to which the system addresses high priority areas of product stabilization
7. Degree to which the system is cost effective (i.e., cost takeout)
8. Degree to which the system is flexible to adding capabilities

#### Scoring Formulas

1. Number and size of Data Bases in Application
2. Number of Tables per Data Base
3. Total Number of Attributes
4. # of attributes that have had multiple definitions over time
5. # of attributes in terms of synonyms and antonyms
6. Number of DB dependent processes
7. Number of one time Interfaces
8. Number of ongoing Interfaces
9. Number of Data Quality Problems and Issues to fix
10. Knowledge / Documentation of Data Quality issues
11. Ease of de-duping similar entities in the same DB
12. Ease of matching same entity records across multiple DBs
13. Completeness of the functional documentation
14. Availability of Subject Matter Experts (SMEs)
15. Maturity of the Enterprise in Managing their data

### MIKE2.0 Methodology: Task Overview

**Task 1.8.1 Prioritise Requirements and Identify Immediate Opps**

#### Business Alignment

1. Degree to which system functions align with base capabilities
2. Degree to which system functions align with enhanced capabilities focused on the new business model
3. Degree to which the system addresses high priority customer segment growth
4. Degree to which the system addresses high priority customer segment retention
5. Degree to which the system addresses high priority areas of product growth
6. Degree to which the system addresses high priority areas of product stabilization
7. Degree to which the system is cost effective (i.e., cost takeout)
8. Degree to which the system is flexible to adding capabilities

#### Scoring Formulas

1. Number and size of Data Bases in Application
2. Number of Tables per Data Base
3. Total Number of Attributes
4. # of attributes that have had multiple definitions over time
5. # of attributes in terms of synonyms and antonyms
6. Number of DB dependent processes
7. Number of one time Interfaces
8. Number of ongoing Interfaces
9. Number of Data Quality Problems and Issues to fix
10. Knowledge / Documentation of Data Quality issues
11. Ease of de-duping similar entities in the same DB
12. Ease of matching same entity records across multiple DBs
13. Completeness of the functional documentation
14. Availability of Subject Matter Experts (SMEs)
15. Maturity of the Enterprise in Managing their data

### MIKE2.0 Methodology: Task Overview

**Task 1.8.1 Prioritise Requirements and Identify Immediate Opps**

#### Business Alignment

1. Degree to which system functions align with base capabilities
2. Degree to which system functions align with enhanced capabilities focused on the new business model
3. Degree to which the system addresses high priority customer segment growth
4. Degree to which the system addresses high priority customer segment retention
5. Degree to which the system addresses high priority areas of product growth
6. Degree to which the system addresses high priority areas of product stabilization
7. Degree to which the system is cost effective (i.e., cost takeout)
8. Degree to which the system is flexible to adding capabilities

#### Scoring Formulas

1. Number and size of Data Bases in Application
2. Number of Tables per Data Base
3. Total Number of Attributes
4. # of attributes that have had multiple definitions over time
5. # of attributes in terms of synonyms and antonyms
6. Number of DB dependent processes
7. Number of one time Interfaces
8. Number of ongoing Interfaces
9. Number of Data Quality Problems and Issues to fix
10. Knowledge / Documentation of Data Quality issues
11. Ease of de-duping similar entities in the same DB
12. Ease of matching same entity records across multiple DBs
13. Completeness of the functional documentation
14. Availability of Subject Matter Experts (SMEs)
15. Maturity of the Enterprise in Managing their data
Metrics on Business Alignment and Difficulty help to formulate priorities for the overall implementation of a large-scale migration programme. This is done by starting with areas that are most important for the business and of the lowest complexity. Whilst a simple model, this helps to clearly illustrate to the business and technical community how priorities were driven for the project in an objective fashion.
Data Migration through Information Development

*Lessons Learned*

**Prioritise Planning**
- Define business priorities and start with **quick wins**
- **Don’t do everything at once** – deliver complex projects through an incremental programme
- “**Big bang**” if you can, know that often you can’t

**Focus on the Areas of High Complexity**
- Get the Technology Backplane capabilities **out in front**
- **Don’t wait until the 11th hour** to deal with Data Quality issues – fix them early
- **Follow the 80/20 rule for fixing data** – does this iteratively through multiple cycles
- Understand the **sophistication required for Application Co-Existence** and that in the short term your systems will get more complex

**Keep the Business Engaged**
- **Communicate continuously** on the planned approach defined in the strategy - the overall Blueprint is the communications document for the life of the programme
- **Try not to be completely infrastructure-focused for long-running releases** – always deliver some form of new business functionality
- **Align the migration programme with analytical initiatives** to give business users more access to data