Experian™
Data Quality

Data Migration Method

Experian Pandora

Experian Technical

1. Introduction

Most data migration projects fail to meet the objectives and constraints set at the start of the project, and most of these failures are due to surprises in the data being moved.

The combination of the Experian Pandora software product and the associated approach detailed in this document maximises the chances of success. This document should be taken as a guide by migration practitioners and project leaders, but common-sense should always prevail and you may find it appropriate to modify these suggested approaches for your particular project.

There are two viewpoints related to all data migration tasks, the source and the target, and both should be addressed by every project:

✓ 1. The need to ensure that the data being delivered to the target is fit for purpose, not only with respect to the technical requirements of the target database or application, but also with respect to its behaviour when used within the target system.

✓ 2. The need to ensure that all relevant data has been extracted and delivered with no breakages (lost, badly transformed or duplicated), and we must be able to justify why any of the available source records has not been delivered to the target.

Projects usually address point #1; if they didn't the data probably wouldn't load into the target system. Also, the people destined to use the target system will "try out" samples of data that have been loaded during what is usually known as "user acceptance testing", so the behaviour isn't untested.

Point #2 is rarely well covered, because it requires a rigorous approach to the exploitation of relationships across all the source data (filters, joins, aggregations, dependencies).

Experian Pandora allows project teams to adopt the best approach, both rigorous and objective, delivering as rapidly as possible on requirements, and providing real data to the target systems at the earliest possible stage.

Experian Pandora allows teams to objectively evaluate and manipulate all the source data (point#2) so that it corresponds to the requirements of the target (point #1).

Starting with knowledge of the objective (target) and a Scoping Phase to properly identify the extent of the task, the Analysis continues down to Subject Areas, then Entities (Tables), and finally Attributes (Columns) in a top-down way.

The final deliverables are:

1. Validated data transformation processes
2. A complete Audit Trail of development and execution
3. Data present and validated in the target application
4. Re-useable "Content" – functions, patterns, domain files
5. Business Glossary information, associating terms with their actual data

> "An error detected in the project testing phase can cost up to 100 times more to correct than the same error found during the design phase"
>
> **The Quality Assurance Institute.**

2. Project tasks overview

1. Investigation and scoping

i. Determine the project objectives, priorities, scope, constraints, timelines and sponsors/decision-makers

ii. Determine sources & any overlap/ consolidation/expected consistency

iii. Estimate complexity

iv. Compile Business Glossary

2. Subject area mapping

i. Categorise & prioritise subject areas

ii. Determine which source systems/data are required for each target area

iii. Initial project estimates, risks, constraints, resources etc.

3. Entity prototyping

Prototype and evaluate all source relationships for each target

i. Browse/load critical files (any table which supplies information and/or which can modify the number of records to be outputted)

ii. Validate/discover keys. May need to prepare common format join keys.

iii. Determine how to associate (join) data sources. Derive the data model, starting from the tables for which keys have been validated.

iv. Examine relationships and analyse results (is the join key as expected?). Do they influence the number of records produced, or provide complimentary information?

v. Filter out unwanted records from each source file. Technical activity based on the source structures.

vi. Evaluate feasibility & refine join rules

4. Detailed profiling

i. Identify and perform initial analysis of source attributes

ii. Note initial findings then move on

iii. Prioritise & Assign detailed analysis based on importance of target fields

iv. Carry out detailed investigations - values, frequencies, presence, data types, validity, format, consistency, integrity

v. Determine corrections (if necessary)

5. Detailed prototyping

i. Apply transformation/correction rules to complete the target tables

ii. Build reference & translation tables

iii. Carry out (prototype) Data de-duplication/Consolidation

iv. Manually refine de-duplication results

v. Consider exporting early versions (with some columns using default values), in order to feed the target application and validate the behaviour of the data and the application

6. Prepare Validation Rules

i. Place validation rules for source vs. target load files (counts)

ii. Place rules to validate validity, integrity & consistency of target load files (keys, values, joins, formats)

iii. Place rules to validate load files vs. target (counts)

7. Prepare Execution

i. Decide on trigger mechanism

ii. Identify workflow

iii. Determine dependent processing

iv. Design incremental loading

8. Execute & validate

i. Execute validation & reconciliation rules. Study results.

4. Some common-sense reminders

4.1 Focus

The more detailed analysis should be limited to entities and attributes that are actually going to be used in the target system. Sometimes, as little as 20% of source system data is moved to a new target. It is all too common to see Analysis Paralysis as people spend their time analysing the entire source system, or analysing interesting but low priority issues.

The Subject Area Analysis and Entity Mapping stages must focus the analysis effort.

4.2 Get information from source

Where possible, use all the original source data for analysis, and analyse each source table of file individually. Sampling, or the use of previously developed extractions which were designed and built with other requirements in mind, often combining different source files and tables is always a source of error.

4.3 Step by step

Stick to the documented sequence of tasks. Don't work on the detailed analysis of interesting data fields before having correctly understood the fundamental data relationships.

4.3.1 Involve the right people

IT and business, together.

The most effective way of working is to meet round a large screen and look at the actual data in a joint workshop. Of course an IT person can perform preliminary investigations to give data workshops some initial direction and questions to resolve, but a large number of issues are uncovered simply by interactively browsing through the data. Do not underestimate the enormous synergy generated by working together in this way.

4.3.2 Data cleansing

In reality, businesses usually have to compromise on quality improvement goals in order to get a system which actually works within the costs and timescales of the currently defined project. If there are specific quality objectives for the project they should be clearly identified and their potential benefit quantified at the start of the project.

Data Quality issues should be documented, categorised as either technical or business and fed to a separate Data Cleansing team so that the business can

1. Evaluate the cost of repair vs. the benefit to the business.

2. Prioritise the issues and assign an appropriate amount of resource.

The effort spent performing complex cleansing operations such as de-duplication should be related to their technical necessity and their benefit to the business.

Being able to load data that functions in the target system means that the data is technically clean, however this does not necessarily mean that it is in the state that the business people want.

For example, when moving Account-orientated data to a Customer-orientated system the business sponsors invariably ask for the identification and de-duplication of customers. But, having multiple customers in the target system will not stop it working... So, how much time should be spent de-duplicating...? It's up to you to decide, but you must time-box all such activities.

5. Initial Investigation and Scoping

Since this task is defined in most data migration methodologies this document is restricted to a summary description.

The work is carried out in accordance with the relative priorities of the Target entities (tables) and the objectives of this task are to:

1. Identify the source systems that will be integrated and migrated, thus defining the boundaries of the project.

2. Evaluate the complexity of the migration effort, e.g. is there duplicate or overlapping data in a system, or across different systems.

5.1 Set the ground rules

Determine the project objectives, priorities, scope, constraints, timelines and sponsors/decision-makers

5.2 Inputs

This phase is based on workshops and interviews, as well as researching existing documentation. Both application experts and business people will be involved as we want to be sure we don't miss any potential sources. It's easier to exclude a source based on an investigation than it is to integrate a new source later on!

5.3 Deliverables

- A scoping document, with initial findings
- Pandora notes with attached examples/ evidence to support the findings of the document (don't lose any of the valuable information you have found already).
- First-cut Business Glossary

5.5 Steps

5.5.1 Determine data sources

Perform workshops with subject matter experts, profiling, viewing and manipulating source and target (if available) data to ensure you are talking about the same things and to get an early view of the data which the project actually has to deal with as opposed to what people think it has to deal with…

The task consists of:

- Examining the target system and performing a rapid analysis of the likely candidates of source data for the target
- If it appears that only a subset of some data is required, this should be documented now, and the criteria for selecting the appropriate records drafted.
- If existing data is to be enriched using external data sources (including manual data entry), this should be identified.
- If there is more than one potential source for some data, they should be identified and documented in the Business Glossary.
- Perform high-level gap analysis to ensure no source areas which look relevant have been forgotten.

5.5.2 Identify systems consolidation & consistency

If data needs to be consolidated/de-duplicated, potential "matching" information should be identified. If systems should be consistent, document this fact so that this can be verified later.

Ask simple but probing questions:

- How do you know these records are for the same product/ person/ company etc.?

- Why is this particular product/person/company etc. more important than the others?
- What is the risk/cost if this type of information is not de-duplicated/consolidated?
- Separate "technical de-duplication" from "fuzzy-matching".

The former involves making allowances for technical differences such as upper/lower case, or data definition differences, e.g. a bank account number could be stored as a "string" or as a number in a system, making them different strictly speaking, even if they are the same in business meaning and content.

5.5.3 Compile/Update Business Glossary

Add to or refine the Business Glossary. Define terms and associate the columns and fields which contain the actual data allowing future work to be consistent across different implementations of data.

5.5.4 Estimate complexity

An initial evaluation should be made of the level of migration required, (cleansing, re-engineering and repurposing), as well as the complexity of the source systems and the need and complexity for combining and relating different source systems to satisfy the new target.

6. Subject area mapping

This task allows the project team to highlight and then focus on the data areas which present the greatest risk to the success of the project, giving them the opportunity to start working on these at an early point in the project.

Subject Area Mapping is extending/completing the Investigation stage and its objectives are:

1. To provide a high-level specification of Source-to-Target data at the system, application or database level; the full set of target entities, and a list of candidate source databases/entities that could supply data to those targets.

2. To prioritise the data movements, taking into account complexity, data volumes, business and technical priority. (Execution time is usually a significant and immovable constraint and needs to be addressed early on as it can affect the implementation approach and even feasibility).

3. Identify and explain out-of-scope source data.

6.1 Inputs

Much of this activity is performed using existing documentation, Data Profiling and workshops. Assumptions and issues requiring further investigation are documented.

6.2 Deliverables

Simple, high level diagrams (e.g. Powerpoint) & descriptions of which systems and subject-areas are involved, how they should relate, and their relative priorities.

Include decisions taken so far, e.g. to exclude certain subjects, systems.

6.3 Steps

6.3.1 Subject Area Categorisation

The Subject Areas themselves can be categorised as follows:-

A. High Priority, Complex

These are always going to be included into the main project scope, as without them there is little chance of success in building the target system. They are, however, going to be complicated to migrate and integrate and the project must allocate sufficient time to deal with the discovery of the unexpected data issues that are likely to exist in the source data and to overcome the technical complexity of data extraction and/or data combining.

B. High Priority, Simple

This data is also critical to the successful delivery of the project but is simple in nature and of reasonable quality and as a result there should not be any unexpected surprises that will affect project timescales.

C. Low Priority, Complex

It is not essential to have this data for the target system to be delivered successfully, and its inclusion will result in a lot of additional complexity and project risk.

D. Low Priority, Simple

Again, this data is not essential to delivery of the target, but it is at least simple in nature and thus if it were included would present a predictably small additional risk to overall project delivery.

Concentrating on category A reduces project risks over time as more unknowns are eliminated through detailed analysis. Effort should not be wasted on low priority data areas that may end up being dropped from the scope.

Having categorised and prioritised the Subject Areas, they can subsequently be analysed in greater detail. The project plan can be augmented to incorporate the schedule of Entity Analysis for each of the Subject Areas.

Further analysis of the category C and D Subject Areas should only be carried out once the analysis (at both Entity and Attribute level) has been addressed for category A and B Subject Areas. The C and D categories are often dropped entirely in the first delivery in a bid to deliver business benefit early. Otherwise, resources (and hence costs) are increased to deliver more within the same time period.

6.3.2 Determine which source systems/data are required

For each target area, refine the mapping of source systems to target subject area.

6.3.3 Improve Business Glossary

Use the results of any further analysis to annotate the data for future steps but also to improve the content of the Business Glossary for this and subsequent projects.

6.3.4 Initial Project estimations

Identifying the effort and priority early helps to mitigate the risk of the project overrunning by empowering the project manager to trade cost against time and to take decisions in the Analysis stage.

The exposure of these risks can also be used to provide early warning as to whether correctly moving the data to the target system is actually feasible!

7. Entity prototyping

The most critical task when populating a target table is managing to get the correct number of records with the appropriate key information on them.

This task allows you to do just that, thus validating the feasibility of the data movement. The details can be worked out later.

Entity prototyping analyses how different source entities relate together to build a target entity and whether any filtering or transformation is required at the record level (e.g. aggregation, joining, normalisation, de-normalisation, de-duplication etc.).

Don't forget that ensuring duplicates are not created by loading data into an existing system means that the target itself may be a source!

The majority of data migration complexity is identified during Entity Prototyping, and it is essential to ensure that this is being addressed before any detailed design (prototyping) occurs.

Often there are multiple sources of data (different entities) and the assumption that a particular entity is suitable for a particular mapping may be disproved at this stage for a number of reasons.

Expending effort designing mappings of attributes from an entity that may not be used is a waste of analysis effort. Worse still, if the Entity Mapping is not done at all, the unsuitability of the source entity may not be identified until build/test time, resulting in an even larger project cost and impact.

7.1 Inputs

1. For each target table, the exhaustive list of source tables and files which might be used to populate it.

2. An idea of how those sources should be related (join keys, join cardinality) to build the target.

7.2 Deliverables

The output of Entity Prototyping is the generated first version of a Data Mapping specification for the target entity in question. This can then be completed in the Attribute Mapping stage.

7.3 Steps

Focus on a single target entity at a time.

Start by analysing the source data relationships that are necessary to populate the most critical and complex target tables, working through the target entities in priority order.

Add notes to each entity to document issues, questions, progress, etc. If you notice something interesting about the data, do not spend time investigating in detail right now, create a note about it and move on; focus.

7.3.1 Identify Possible Source Entities

Refer to the Subject Area Mapping and the source systems schema information to find most likely Source Entities to build data for the target. There may be additional sources required such as Master Data reference tables etc. These also need to be identified. Any attributes required for the filtering of source entity records will also need to be identified at this stage (for example Record Type fields).

7.3.2 Identify the driver

Which source(s) will determine the number of records being prepared for the target. E.g "I want as many records as there are customer telephone numbers".

7.3.3 Identify relationships between sources

Load the identified sources into Pandora, and all relationships will be discovered automatically based on common data values.

The relationship between two sources may be evaluated in Pandora at any one time to ensure that the relationship is exactly as expected.

If you are not sure about the relationships, e.g. there are unknown "link" tables between the main data tables, start from what appears to be the main source table and:

- Discover (find) the unique key(s) for that table
- Find all relationships with other tables which rely on the key field(s). Those with the best relationships are related!
- Continue out from those tables, analysing relationships to discover the model

For multi-column keys, join on the most unique of the key columns, and use the drilldown manager (filter) to add the other columns to the join condition.

If a relationship is being used simply to fetch complementary values form another table, as opposed to altering or helping determine the number of target records produced, then it is not critical from an entity mapping point of view. Of course in this case you should still verify that the relationship you have in mind is linking to a single record.

e.g. if the requirement is to fetch the customer telephone number from the telephone number table and it turns out that some customers have two telephone number records, do you need both numbers, or do you ignore one of them…?

If you want to use a relationship which is not based on single unique values you may need to create a derived join key, and re-analyse the result, i.e. create a new column using the Pandora expression editor, and save (materialise) as a derived table.

With the current version of Pandora (v3) if you need to join four source tables, you'll need to materialise the results of the first join, then drill to the new relationship between the materialised table and the third source table. Then materialise the resulting table and drill to the new relationship with the fourth table. Though this may appear fastidious, is ensures a 100% correct understanding of all aspects of every relationship; very record is accounted for, which is one of our initial requirements for all data migration.

Relationships can be optional or mandatory. Furthermore, they will be one-to-one, one-to many, or many-to-many. You have to decide whether you need to keep unmatched records (orphans, or the "outer join"). All combinations can be browed in the relationships made available by Pandora.

Be wary of relationships which are based on a set-based transformation of a source table, e.g. join to the most recent delivery record, or join to a total, by product of the sales records. These may well require you to save some kind of aggregated/transformed table to avoid ending up with duplicated records.

7.3.4 Remove unwanted subsets of data

A table may contain unwanted records (within the context of the current target) which you want to filter out, e.g. history records, inactive records, unwanted record types or even duplicated records. Filters can be applied to the data at any time using the Pandora interface. If you don't need to create new join keys, but you do want to filter the data, it is usually best to browse the results of a relationship (join) and filter the results (rather than filtering source tables, materialising the results and browsing the resulting join).

7.3.5 Improve Business Glossary

Use the results of any further analysis to annotate the data for future steps but also to improve the content of the Business Glossary for this and subsequent projects.

7.4 Performance prototype

The entity prototype specification can also be provided to the implementation team so that they can build the necessary data extractions and joins in order to give an early evaluation of the execution times with the chosen ETL/Data Integration tool if one is being used. There is no point in finishing the details of the analysis and design if the resulting process takes so long to execute that it is infeasible. At the very least you can find out about processing hot spots and decide how to handle them when it's not too late; decide how the implementation can be tuned, or simply choose another approach to get the same result.

7.5 Technical de-duplication & Consolidation

During the entity prototyping task you should investigate the relationships required to de-duplicate data across systems.

The objective at this point is to de-duplicate and consolidate records which are clearly the same/ identical. Do not attempt "fuzzy matching" and de-duplication, which is an open-ended activity which should be a time-limited and carried out later, off the project critical path.

E.g. yes, we can consolidate customer lists from two systems based on "company tax identifier".

It is possible at this stage to rely on some simple transformations such as changing case, removing spaces and punctuation, changing data-type and padding/formatting values, however you should not be tempted to modify/standardise/cleanse the values beyond these simple technical operations.

Use interactive Grouping and aggregation (count) on the values to see which are duplicated.

7.6 Building reference tables

Start to build reference data tables which will be used by the implementation team during Build and Execution, or you can simply specify and validate the rules required to build the table if the data is volatile.

Example approach:

a) Group-by (combinations of) unique values

b) Apply a count to ensure they are unique

c) Filter out unwanted values

d) Fill in missing values from other columns or even other sources (using expressions and joins)

8. Detailed profiling

Detailed profiling activity began with the key information during entity prototyping, and can continue once that task is completed for each table. It supports the detailed prototyping task by utilising real data and automated analysis to dispel as many assumptions about the data as possible.

Profiling is about fully understanding the data in order to know if/how it is to be used to build records for the target. It is imperative that Data Profiling is performed against real data and that the data is as close to the real sources as possibly – and ideally is extracted from the source systems in exactly the same way that it would be by the implementation. Using data samples or generated data may either obscure data problems, or mislead analysts into thinking there are problems there that do not actually exist.

Try to avoid using data provided by extraction programs which filter, join, aggregate, transform or cleanse source data otherwise you won't be getting a true picture of the source data.

Only ever use real application data – never old backups or artificial test data.

Study the derived information to evaluate its suitability for the target

- Values and their frequency
- Empty fields
- Types of data
- Formats, and their frequency

Pandora lets you sort, filter, group and count to assist this process.

Add notes to each attribute to document issues, conclusions, progress etc.

8.1 Inputs

- The source data tables and files.
- Target data (if it exists)

This allows you to understand the content, format and structure of the target data. You can also evaluate the quality of what is there. This can always be useful later if there are claims that your project has broken something in the target.

8.2 Deliverables

Knowledge about the source data in the form of notes.

8.3 Steps

Load the data into Pandora and it will derive the basic information required.

8.3.1 Identify source attributes

Finding appropriate attributes can be time consuming. By allowing you to search for values and value formats across all fields of all source files and tables, Pandora allows you to discover where the relevant data is situated (on top of the obvious places uncovered by studying documentation and interviewing subject area experts).

This unique capability allows you to ensure you don't miss any relevant data sources. It also allows you to find relevant data which has been stored in the wrong fields.

There are various useful techniques to use when looking for source attributes:-

1. Look for attributes that have similar names. Pandora allows you to view, sort and filter all field names.

2. Attribute names may be similar but not identical, but have similar data. For example Product Id and Product Code. So look for values with the same format using the right-click options on drilldowns.

3. Look for columns which have common values – use the relationship analysis.

4. Search through the list of values – throughout the entire Pandora database or for a table. Search using values, patterns, sounds. Search for entire values and/or embedded values.

e.g. use a pattern (regular expression) to find bank account numbers or credit card numbers embedded in free-text values.

8.3.2 Note and move on

Tell-tale signs of issues are:

- Conflicting Datatypes
- Value frequency almost 100%
- Values low/high
- Rare Value formats

Try to address the different fields/columns in priority order:

- Don't analyse anything that is not required by your data movement process
- Don't spend a lot of time analysing attributes in detail. No more than 5 minutes!

Use the Columns summary drilldowns and the Outliers report to start of this process by easily identify unusual data.

Begin by finding and "noting" potential issues.

If you have looked at some data and concluded it is not an issue, then create a note to say so. This will avoid duplicating the same analysis effort later.

8.3.3 Improve Business Glossary

Use the results of any further analysis to annotate the data for future steps but also to improve the content of the Business Glossary for this and subsequent projects.

8.3.4 Prioritise and Assign

In consultation with business colleagues and target system experts:

- Decide which of the noted "issues" requires further investigation
- Determine priorities
- Assign the investigation to the most appropriate people

8.3.5 Investigate

Use Pandora to interactively investigate the data. The most commonly used sources of information are the unique-values and the formats.

Use functions, filters, metadata drilldowns (values, formats) to view interesting records and use Pandora to:

- Quantify the issues (use the instant, right-click profile)
- Measure the potential impact/importance (is there a monetary value, risk category etc. that can be associated with the data).
- Determine root causes for the bad data. E.g. it is always for a particular product or originates in a particular office department. (instant profile again)

Develop and (re-)use validation rules for your specific requirements based on:

- Pandora Functions
- Formats (define these as "business constants" to make them reusable by colleagues
- Domains defining valid values

Use the “drilldown summary” to create quick reporting to communicate with colleagues.

Create and save validation drilldowns for tables. New versions of the source data will inherit these validation rules when they are “loaded”, allowing you to regression test and measure the quality progress of the source data.

8.3.6 Determine correction

Based on your analysis, decide how to address the issue.

8.3.7

There are many possibilities:

- If there is a potential fix which requires more investigation, specification and development, then that will be tackled during “prototyping”.
- Can the data be corrected at source? Even manual fixes are feasible if the volumes are small.
- Can you build a corrected version of the data using Pandora and/or manual editing of a file? If so, this corrected version could be applied to the source by the people in charge of that application, or used as a “lookup” table by the data movement process – for this old value, replace it with this new value.
- Can you specify a correction and/or validation to be applied during the data movement process? (preferred solution)

Determine whether you are likely to see more bad data being produced in the future, and whether some action is required in the source system to avoid this:

- Restrictions in the application
- Staff training
- Changes to business process

9. Detailed prototyping

This is effectively logical design for each target table and uses an agile approach to development.

Based on the findings from entity prototyping and detailed profiling, use the Pandora interface to interactively transform the source data to suit the target; filter, join, aggregate the records, transform, create and calculate individual values for the fields of the target records. Translate and standardise using domain tables.

All the operations required to make the source data suitable for the target are memorised by Pandora. Save your drilldown as a View and right-click on it in order to access the generation of the full logical design document which is effectively the audit trail of the transformations applied.

A target focus is maintained at all times to ensure only relevant source attributes are analysed and profiled, and that the context of the target (rules, ranges, semantics etc.) are considered and tested.

A target can be built up incrementally as the rules are refined. Deliver (export) the data to the target as soon as possible to allow early validation of both fitness-for-purpose and load-execution times.

Interactive Data Profiling and Prototyping should be used at all stages to validate transformation rules, data ranges – basically to enforce the rules of the target attribute on the source attributes using real data.

9.1 Inputs

1. Detailed profiling information, including analyst notes

2. Technical information about the target – table layouts, definitions of the fields, expected relationships between target tables.

3. Source data!

9.2 Deliverables

For each target table:

1. A validated data transformation process

2. A generated logical design/specification, which has been proved to be correct

3. A file of target data (which can also be used to validate en ETL implementation/execution if necessary)

4. Categorised analyst notes - Data quality issues, questions and answers, issues, conclusions etc.

5. Validation rules on the

9.3 Priorities

Use the previously establish priorities for the target fields and focus on the most critical attributes first.

9.3.1 System Mandatory Attributes

These are the minimum required to do basic processing in the target, e.g. insertion and identification of unique records. These are all mandatory, in that they must exist, and will never be defaulted or left blank. They will support all primary and foreign key relationships, or navigational hierarchy relationships, and the ability to select records. They provide an operational shell to provide basic transactions for the target, but not necessarily all the required business meaning!

Imagine a target entity that was a customer table. The need to uniquely identify customers and relate them to an address is a basic system requirement. Hence, populating the Customer table with a unique customer id, and a related address id is all that is needed to insert and delete customers. Everything else could be defaulted. This is quite a critical concept. If you achieve the correct mappings for the System Mandatory attributes, and the correct number of target records are created/updated etc., that is around 80% of the hard effort required to actually populate that target, and it means the bulk of the risks have been removed, from a technical perspective.

9.3.2 Business Mandatory Attributes

These attributes provide the lowest level of business intelligence in the target application. In the example above, with the Customer table, it may be that the business must at all times know the Gender, Date Of Birth and Email Address of the Customer, but everything else might not be important to the business, just nice-to-know. For example, occupation.

None of these attributes will have any effect on the structural nature of the target entity (e.g. these are non-key attributes). Typically, this data is Master Data or Reference Data.

9.3.3 Mapping Non-Essential Attributes

These attributes have the lowest importance. They are not required for structural purposes (e.g. creation of target records) and are not critical to the business operation or business value of the target system. They could be considered as nice-to-have. These are the attributes which should be worked on last as they could always be delayed for a later phase of migration if their business value does not justify delaying the whole project. They can also be worked on as a background task to fill-in periods of natural delay (common in migration projects). It is likely that there are many attributes in this group, so they may be optionally subdivided into smaller groups and prioritised to manage the work effort. It is these simple attributes that most project teams are drawn to early in a project as they feel, wrongly, that dealing with this 80% of attributes first gets them 80% down the road to success.

9.4 Potential operations

9.4.1 Datatype Translation

There is often the need for datatype conversion, e.g. from alphanumeric to numeric. This should be examined carefully using the type information in the Pandora profile. If there is any issue discovered, a transformation rule may be required, or perhaps some data needs to go to a different target, or the database definition needs to be modified.

9.4.2 Length Mismatch

A target attribute may truncate the source data if it is too long to be stored. A more complex rule may be needed to correctly process the source data. In this case, either the target attribute must be modified to have a longer length (not always possible) or the source data further processed or encoded in a different way. For attributes that are longer than the supplying source, it may be necessary to pad data, or format it in a particular way (e.g. leading zeros for numeric fields) to satisfy the length requirements of the target attribute.

9.4.3 Format Translation

Once the format of the target field is understood, the source field can be profiled to see how well the format matches, and appropriate transformation rules to rectify the format can then be built.

9.4.4 Semantic Translation

Data could potentially be of the right physical type but semantically incorrect for the target system, requiring translation. A common example of this would be coded fields that are represented differently in the target system. A source attribute Gender field uses ‘M’ and ‘F’ to denote male and female, whereas the target may use a coded system of ‘1’ for Male, ‘2’ for Female and ‘3’ for unknown. This may be a foreign key to a Gender table. A transformation rule to translate ‘1’ to ‘M’, and ‘F’ to ‘2’ would be used to correct this.

9.4.5 Missing Value Resolution

Missing values in the source are a common problem and can be quite complex to deal with. If the source attribute is populating an attribute that must always have a value, then analysis will need to be done to determine how best to populate the target attribute when the source has a missing value. This may be a default value; it may be a case of choosing a value from another source using some kind of precedence logic. The expression editor allows all such solutions.

9.4.6 Date Processing

Date reformatting is very common, but many people overlook just how complex data processing can be. The expression editor has a list of functions allowing sophisticated date manipulation.

9.4.7 Master Data Translation/Standardisation

The value required for the target attribute may need a translation table to convert it to standard Master Data values. Translation tables are frequently used. These can be built using Pandora Domains.

Example approach:

- Group-by (combinations of) unique values
- Apply a count to ensure they are unique
- Filter out unwanted values
- Fill in missing values from other columns or even other sources (using expressions and joins)

9.4.8 Resolving Multiple Source Attributes

Whenever multiple attributes are mapped to a target there is a need for a transformation rule to determine the value to be used.

It could be a simple concatenation rule, or some of the source attributes could be used with If-Then-Else type logic to determine which of the other source attributes are used, and in what way.

It could also be a mathematical calculation. An example would be the source attributes Department, Salary, Bonus and Commission mapped to the target attribute Compensation. The rule might be that sales people get a compensation of Salary + Commission where as everyone else gets Salary + Bonus. The Department field would determine which mathematical calculation was used, and hence which attributes are used. Use the expression editor for this.

Make sure all the source fields are actually available (have been joined or retrieved using a lookup).

9.4.9 Data de-duplication/Consolidation

Data movement processes are often seen as the opportunity to "clean up" data, removing duplicates.

e.g. de-duplicate customers based on fuzzy matching of name and bank-account number, etc.

Don't let this this opportunity side-track you and the project. Determine how critical it is to the success of the project? It's usually important but it should not be allowed to impact the project end-date. This type of activity is very open-ended and difficult to estimate. It should be tackled as a parallel, time-boxed work stream outside the main project critical path.

Get a feel for how effective this consolidation will be, by trying it out using functionality from the Pandora expression editor. For example, standardise data by removing all blanks and punctuation, changing all to upper case, then compare records based on this prepared "matching key", or perform an interactive Grouping and Count aggregation on the derived values to see which are duplicated.

The Pandora parsing, pattern matching, string manipulation and domain functions are extremely powerful and ideally suited to this activity. Develop domains of values, patterns (using regular expressions) and use them to recognise, split out and standardise values. Comparison of records is much more effective with these standardised values.

A seasoned Pandora user can create an entire set of standardisation and cleansing rules and use them for matching purposes within a couple of days, including preparation of translation and reference domains and validation formats.

Most companies can re-use existing reference files, and some Experian partners propose packaged solutions containing domains, formats, cleansing, standardisation and matching rules.

For most data migration projects, this one-off "prototype" is the end result in itself; just export and use the cleansed data, with no other software required. If Pandora is used to provide a list of "candidate" duplicates, the list can be manually refined and used as-is or as a "lookup" in the data movement process.

9.5 Steps

9.5.1 Define transformation rule

The need for a transformation arises when the data in the source system does not match the formats or processing requirements of the target, which is fairly often! Transformation will also be required to resolve data format and standardisation issues.

Build rules incrementally, validating the results at each step using the actual data on the screen. You will be doing the equivalent of "writing a specification" and "validating it".

Source attributes will be used in many different ways:

1. Single source attribute to populate the target, verbatim.
2. Single source attribute to populate with transformation via a rule.

   e.g. if the source was 'M', set to 'Male', if it was 'F', set to 'Female'.
3. Multiple source attributes to populate with a transformation (e.g. concatenate all source fields with spaces).
4. Multiple source attributes to provide data and decision criteria within a transformation rule, e.g. a TYPE field could be used to decide on the formula to be applied to a SUM field for different transaction types.
5. Transform discount code to a percentage taken from another (reference) table.

A target attribute may not have matching source attributes for the following reasons:-

1. t is a new concept that has no existing source data.
2. It is going to have a default value set.
3. The value will be programmatically generated (e.g. an auto-incrementing sequence number)

9.5.2 Improve Business Glossary

Use the results of any further analysis to annotate the data for future steps but also to improve the content of the Business Glossary for this and subsequent projects.

9.5.3 Create extra targets

You may also create "targets" which are actually cross-reference tables to be used during execution of the data movement process. In this case the result of the "prototype" will actually be used by the end solution. Hybrid cross-reference tables created initially using Pandora and refined by business users are a pragmatic solution which can avoid trying to invent tortuous rules for a few rare cases.

9.5.4 Create Domain and translation tables

Another type of reference table often created is "alias" tables. These can be used both by Pandora and by other data movement processes because they are contained in simple delimited files. Domain tables can be created and maintained by business people with no access to the Pandora software.

10. Prepare validation rules

The objective of this work is to prepare the rules that will allow us to validate the migration process and prove that we have neither lost nor created/duplicated data.

10.1 Inputs

- Data to be compared is loaded into Pandora

10.2 Deliverables

- Validation rules positioned on Tables
- Statistics provided for all tables (migration day statistics should be "close" to these)

10.3 Steps

10.3.1 Validate the data ready to load into the target

The objective is to validate that all the prepared data is as required for the target

- Validate that prepared keys are indeed unique before attempting to insert them
- Validate the content and format of the data fields
- Validate the relationships between tables
- Apply more sophisticated rules to evaluate, count, etc.
- Ensure the values to be inserted exist in the appropriate reference list/tables
- Ensure that the required data integrity (e.g. detail records have header records) exists in the data which is ready to load.
- Validate consistency and validity of data with respect to other fields using business rules.

10.3.2 Account for each source record

The objective is to ensure records are not lost or duplicated, and is carried out by comparing keys/identifiers.

- The reconciliation of source(s) to Pandora outputs is carried out to ensure that the data ready to load corresponds to what it should be given the sources we have used.
- The reconciliation of load files to "loaded" data is carried out to ensure that the load processes of the target application have performed as expected.

These rules should be prepared during the development of the migration processes – Detailed Prototyping. If the structures have radically changed, validating record counts may require some aggregation or splitting of statistics, but it is necessary in order to prove the correctness of the data movement.

- For each source record, lookup to ensure that its key exists in the target load files
- For each record in the target load files, lookup to ensure that it exists in the loaded target application.

11. Prepare Pandora Execution

This chapter describes how to provide data quality measurements with relative weightings, and how to aggregate or summarise them in various ways.

Once the data on-screen in Pandora matches the requirements for the target system there are two choices.

1. For most migration projects, the processes can be automated and executed. The data files produced by Pandora can be loaded into the target application.

2. Some (very large) migration projects may decide to use a Data Integration (ETL) tool for the actual data movement. In this case Pandora generates the specifications which are implemented in the ETL software. ETL processes can be validated by comparing their output data with that of Pandora.

11.1 Inputs

The individual processes that constitute the Migration have been built and validated in Pandora. Each dataset to be loaded into the target will have a "drilldown" definition.

11.2 Deliverables

For each source file or database:

- Trigger file
- Automated-import definitions

For each source file/table:

- Novation options set on saved drilldowns
- Automatic-export filenames defined on saved drilldowns

11.3 Steps

11.3.1 Define data import automation

Use trigger files to signify to Pandora that the data is ready to be read and transformed. For databases you will need an XML file that names the tables to be exported, and this can be generated by Pandora.

Review and set the Novation options on each drilldown.

11.3.2 Define drilldown export automation

Specify output file names in the appropriate export folder on the server.

Output to the "import" folder if processes are to be chained together automatically.

11.3.3 Identify dependent processing

If processes require the output from other processes this needs to be identified and built into the implementation workflow. This can have an impact on the end-to-end execution time (critical path), as well as the re-use of certain processing

11.3.4 Build an execution workflow

On modest projects (30-40 tables) this can be carried out manually based on written procedures/ checklist.

Ideally this should be automated using an enterprise scheduler, but some simple scripting and the Windows scheduler are often sufficient.

12. Prepare ETL/DI Execution

Only necessary if Pandora is not being used for the execution.

12.1 Inputs

Logical Design Specifications have been generated they must be analysed collectively in order that an optimal Physical Design can be produced for the Data Migration. The Logical-To-Physical Analysis identifies commonality of processing across Logical Mapping Specifications and consolidates this functionality to optimise performance and minimise development in the Build Phase.

12.2 Deliverables

Design deliverables include:

- Number Of Individual Data Movement Processes
- The need for a Staging Platform?
- Opportunities for Parallelism
- Audit/Reconciliation processing
- Restart/Recovery processing
- Transactional Integrity processing

12.3 Steps

12.3.1 Identify common/shared processing

In many cases the physical design will be identical to the Logical Mapping Specification, but some situations do merit a separate physical design. For example, if more than one Data Movement extracts similar data from the same source, it may make sense for that data to be extracted once into a Staging Area instead of repeatedly querying the source system (i.e. to maximise performance).

Equally, pre-processing source data to standardise formatting to facilitate joining of data is another example of a typical pre-processing requirement across multiple Logical Mapping Specifications that could be performed once instead of many times.

Of course this kind of re-use may have been identified earlier and built in to the logical design, but if there are several people working on overlapping data it is not always the case.

12.3.2 Identify dependent processing

If processes require the output from other processes this needs to be identified and built into the implementation workflow. This can have an impact on the end-to-end execution time (critical path), as well as the re-use of certain processing.

12.3.3 Incremental Loading

An immediate observation with any repetitive data movements is that each phase will load additional data to the target system. This may be to new target entities, or it may be (and probably will be) additional data to target entities populated on previous phases, including potentially updating or extending the information previously loaded.

Change Data Capture techniques exist for the identification of changes to source data, such as using timestamps, or comparing current keys with previous keys (which may need to be accounted for in the logical design). Mature change capture technologies exist too. The changed data thus supplied needs to be profiled and prototyped too.

Incrementally loading the target can be catered for by exploiting the relationship between the keys which have been prepared for loading, and the existing keys on the target, but usually, the load technology deals with this using any one of a number of techniques (caching the existing keys, performing lookups, performing INSERT ESLE UPDATE commands, etc.)

In cases where changes cannot be reliably identified at all, or where data volumes are small (up to a few million records) it is often more efficient to continually bulk load and refresh.

13. Execute & validate

13.1 Inputs

- Trigger files
- Transformation processes (drilldowns)

13.2 Deliverables

Data in the target application

Validation results checked

- Do the target record counts and values match the prepared load files
- Was any data rejected at load
- Does the target data match the prepared load files

13.3 Steps

13.3.1 Execute the Pandora load processes

Either manually or using triggers, load data into Pandora in the correct order. Since the previous steps have already prepared this, there is not much to say about this.

13.3.2 Load data into Pandora for validation

- Loaded the data ready to load to the target system back into Pandora
- Re-novate the source tables to ensure that they are comparing the values with these versions of the load files.
- Load the data that was loaded into the target system back into Pandora
- Re-novate the tables in Pandora that correspond to the files of data loaded to the target application to ensure that their validation rules are comparing the values from the target.

13.3.3 Compare source data with data ready to load

These validation rules were set up previously and will show up any issues. Compare migration-day statistics with previous tests to ensure the results are similar or the same.

13.3.4 Compare the data ready to load with data actually loaded

This allows you to validate that the load process has worked (nothing has been corrupted or lost) and that the existing relationships in the target are being respected by the new data. These validation rules were set up previously.

14. Different Types of Data

Source application use of data differs widely and falls into three main categories. The way that these are handled and processed within Data Migration is based largely on the typical usage in source and target systems, each coming with their own needs.

14.1 Historical Data

Data which is static and will not change during the life of the project.

The volume of Historical Data is often very large indeed, depending on how much historical information has actually been kept. Storage is often an issue, and much of the data may be archived in optical media, or worse, summarised. It is unlikely that the summarisation will have destroyed any statutory needs for the data, but it may reduce the capabilities of the target system if the need for detailed history over and above what has been summarised is required.

This type of data can be moved before the main migration execution, or not moved at all.

14.2 Master/Reference Data

Master Data is essential reference data used throughout the organisation. In most instances it tends to be replicated, duplicated and present in variable levels of quality and completeness across various source systems. Master data has to be in place before transactional data is moved. This is where the data cleansing and de-duplication effort will be required.

Master Data volumes are usually quite small, though in certain cases, for organisations with millions of customers, it can be pretty large in its own right! It does tend not to grow quickly, unlike transactional data.

Non-volatile reference data can even be prepared for loading using Pandora.

14.3 Transactional Data

Transactions are typically high volume data related to the reference data. Records are short and the data is highly repetitive. This is the data which is most likely to be on the critical path for the execution time of any data movement.

15. Experian Pandora post-migration ROI

The infrastructure and processes established during the migration are reusable, providing a framework for enterprise-wide Data Quality Management or Data Governance, ad-hoc reporting, or investigations into fraud and error. Even after the migration project has completed, Pandora can continue to provide long-term value to organisations in multiple ways.

- Data Quality/Data Governance
- Data cleansing & enrichment
- Ad-hoc reporting
- Creation of Master Data
- Fraud and error detection

16. Experian Software

Experian Software is a leading Data Management software vendor. The Experian Pandora software product combines business-minded usability, unrivalled performance, and the richest selection of capabilities to accelerate organisations towards Data Management maturity.

Experian Pandora is used by organisations in a diverse range of industries across the world to carry out activities such as Data Migration and Data Quality Management.

17. For more

For more information on how Experian Pandora could save your organisation time and money, please contact Experian Software on 0800 197 7920 or send an email to datamigration@Experian.com.

Get a flavour for some of the Pandora functionality by downloading the Experian Pandora Free Data Profiler from http://free.x88.com (registration required).

Experian Australia Pty Ltd
Level 6, 549 St Kilda Road
Melbourne, VIC 3004, Australia

T (61) 3 8699 0100 | F (61) 3 9923 6280
E info@au.experian.com | W edq.com/au

Experian™
Data Quality

Intelligent interactions.
Every time.