How to Rescue Your Data Migration Project from Failure

Experian Pandora

Author : Johny Morris
Author of 'Practical Data Migration'
Director/Co-Founder - iergo Limited

Contents

1. Introduction

If we look at the statistics for Data Migration Project success rate then it's clear that many data migration projects either fail or fall far short of their intended goals.

Despite major improvements in success rate through the correct implementation of specialist technology and methodologies, data migration failures are still a common event.

However, it need not be this way.

In this guide I'm going to show you a simple workflow I typically adopt when asked to rescue data migration projects that have fallen into crisis mode.

There are typically 3 main phases I look to implement:

Stabalise   **Plan**   **Mop up**

2. Stabilisation

This phase rapidly prevents the current situation from getting worse.

We aim to address the ineffective working practices and panic-stricken firefighting by creating an environment where a more considered approach can be adopted.

As a consultant or manager tasked with turning around a failing data migration you will often need a great deal of tact and diplomacy to defuse internal politics and acrimonious disputes that may have evolved during the project.

What follows are some practical techniques for getting your data migration project back into a stable state so you can start to build a platform for delivery.

Assess the current status of the project

At iergo we have developed a comprehensive risk assessment that covers the following main areas:

- → Data Architecture
- → Business Engagement
- → Programme Governance
- → Policies
- → Migration Delivery
- → System Retirement policies
- → Key Data Stakeholder Analysis
- → Data Quality Rules

We have integrated this risk assessment into a rapid questionnaire which we can apply in hours if we are really pushed, but which we prefer to typically take no more than two weeks over.

If this is your programme that is slipping then you will be familiar with the issues but it is still worthwhile going through the formal check list to make sure that in the blinkered 'group-think' that develops as projects fail, you aren't mistaking the symptoms for the causes.

When projects are far behind their forecasted schedule and well into panic mode it can often be a challenge for the leadership to realise that they need to take stock of the situation but it's critical for this review to take place.

There are some caveats with this process:

- → The iergo list is both comprehensive and measured against industry statistics that allows us to produce an analysis of impact and potential overrun in terms of cost and time. If you are doing the same in-house then you will have to be pragmatic and use your common sense to work out the impacts

- → Although this list covers the same areas it is less detailed than the one we use

- → The list has to be applied with a sensitivity to circumstance. We weight the results depending on local Policies (see below) and the circumstances of the programme

- → I would use it as an aid memoir rather than a set in stone, fixed template

A properly run data migration must cover all these areas. The missing elements tend to be in the Business Engagement (BE) and Key Data Stakeholder Analysis (KDSA) areas but their impact is usually seen in Migration Delivery.

When we are called in to look at failing projects, our attention is nearly always directed to Migration Delivery but the problems typically have their genesis in BE or KDSA.

Within each category you will need to asess the following:

Data Architecture
Landscape Analysis - was it adequately performed? Were the results channelled into the DQR process (see below)? Were they reflected in the Design? The impulse to "just get on and do something" is often overpowering. The results of poor (or non-existent) Landscape Analysis is usually found in the vast number of data issues that flood out of the migration engine at run time and swamp the programme

Metadata Understanding - we use a selection of models: Migration Models, Legacy System Models, Target System Models, Conceptual Entity Models – to understand our migrations. Do you have a common understanding across the plethora of systems that make up your sources?

Master Data Consolidation - what are the key master data items your migration needs to manage? Is it customers or products or personnel? Often in the legacy data, patchy updating is overcome by operating procedures ("Always go to system x when you need the latest phone number not system x because it gets out of date" for instance). However when you try to put it together in a migration, mismatches occur all over the place

Business Engagement
Here we cover the more formal aspects of business engagement:

Communication Strategy - who is responsible for creating the BE strategy and the methods for getting messages out? How engaged are they in the programme? How aligned are your needs (especially important when a programme is failing) with the mechanisms for communicating? Urgent messages like failed updates or emergency workarounds may not be briefed and are not suitable for cascade briefings for instance

Data Transitional Rules - in any data migration of consequence there are one set of operating procedures prior to the migration and another set after the migration but there is often a third set specifically to cover special processing during the migration. The most common example is the treatment of transactions that start prior to the migration but end after it – the so-called "in-flights". How are you recording, developing policing and briefing your Data Transitional Rules? Are they being followed? In the current trend towards progressive migrations as opposed

to Big Bang, failure to create and follow appropriate Data Transitional Rules will lead to cumulative data errors in the target and source. Often these look like errors in the migration software but aren't

Training Plan - it should go without saying but are all your users trained in the new systems at an appropriate time? Train too early and we forget what we've learned. Train too late and we've already messed up the new system. The so called "Training Lag" is a real inhibitor on large migrations. Are you sure that all your people know what they are to expect?

Business Re-organisation - large organisations are in a constant state of flux. Often the migration is as a result of these changes (think of mergers or de-mergers). However the two programmes can get out of step with a delayed migration trying to be managed into a partially transformed business environment. And there can always be business re-organisations that have nothing to do with the data migration but which severely impact it. How much of your issues are related at base to simply not having the right organisation in place?

Project Governance
One would hope that after a dozen years of Prince etc, we would have a handle on what constitutes proper governance on a data migration programme. I wish that were so. This is not a comprehensive checklist but will give you a start:

→ Do you have a clear scope statement?

→ Do you have a Risks and Issues Log? Is it up to date with a process that works?

→ Is your Programme Management Office (PMO) functioning?

→ Do you have a change controlled, up to date plan?

→ Are your various project and programme boards in place? Do the

required linkages up and down the management project structure work properly?

→ Do you have visibility of the state of the programme and the current pressing problems?

Do you have the budget under control? Both actual and committed spend?

Policies

These are the high level drivers that shape and inform the scope but they can conflict. Going as fast as possible can conflict with maximising data quality for instance.

Has senior management been walked through the policies (some of which are often tacit)?

Is there a conflict resolution process?

At a high level, policy conflict resolution is part of the governance process. Day-to-day conflict resolution should be built into the migration programme low-level tasks.

Some common policies include:

Strategic Architectural Alignment - the drive to conform to a strategic architecture may inhibit quick fixes and workarounds that would get us to the solution

Master Data Management (MDM) - covered above but also consider the common situation where the strategic MDM solution (like the CRM solution for mastering Customer Data) is not the best available source of data. This leads to user resistance to the solution and frequent backtracking to get the "right" data

Hard Stop Flexibility - crucial to understanding what you can do in terms of lengthening the programme to get the optimal result

Regulatory Constraints – these can often be your friends in problematic migrations, but don't overplay the use of the Data

Protection Act or SOX compliance as a stalling mechanism as you try to get your migration back on track

Migration Delivery

This is the build, test and execution of migration.

In my experience although nearly always running late in delivery (and so therefore inadequately tested) it is rarely the nuts and bolts of the migration that causes failure. It just seems that way as all these other issues that should have been resolved elsewhere cause to it to fail. In any case there are enough books and articles out there on Software Engineering to render a short piece like this redundant but:

Appropriate Tool Selection - have you chosen an appropriate tool for your migration? Not that there's often a lot you can do about a poor selection that is backed by senior management. However it should influence your calculations of time to fix

Non-Functional Design - so often under-estimated, although by now with a migration failing it tends to be pretty obvious, but can you get the through put, end to end, that your migration needs? What are the bottle necks? Will the smart use of overtime or extended run times help? Can you design out the pinch points? Either technically or by the use of a Data Transitional Rule?

Fallout Management - do rejected records fall elegantly into a pre-designed process or do they fly out into a chaotic group of frantic technologists for non-planned and non-audited data hacking to fix? Don't however mistake a prettily designed fallout reporting tool for a proper fallout management solution. Is there any substance behind the facade? The quickest impact you can make to a failing project is to get a grip here but with the awareness that most

of the problems you see here will have their origin in failed activities elsewhere in this check list. Use your actions here not just to address the immediate but to start recovering these other failures. If you don't you will be in for an awful lot of firefighting

Fallback Policy - every well designed migration should have one. I guess if you are reading this in anger then you will be knee deep in yours and it's a bit late for me to ask how adequate you are finding it but on future projects it really does pay to design a fallback policy early in the project lifecycle

System Retirement Policies (SRP)

These are user facing documents that describe in business terms how the legacy systems are to be decommissioned and what degree of re-assurance the Key Data Stakeholders are going to get that their part of the business will continue to function post migration. In our experience these are rarely in evidence when we parachute into failing programmes. More commonly there are a series of guerilla engagements between the "techies" and the business each trying to browbeat the other with the overwhelming force of senior business sponsorship usually adhering to a totemic reference to ruling Policy. This responsibility gap, that I've commented on elsewhere, widens until the project plunges into it. It is symptomatic of a failure to resolve Policy conflicts and a failure to complete Key Data Stakeholder Analysis adequately. System Retirement Policies should include detailed sections on:

Audit - how does the business user know that all the essential items have been successfully moved?

History requirements (especially for data items that are NOT part of the new design but essential to business processes)

Business Migration Restrictions - here is where you record all the business side restrictions – maybe Key Performance Indicators that must be met or busy work periods – that might be compromised by the migration

Training requirements (touched on above)

Business continuity - what are the business-side restrictions (that must be fulfilled) that will constrain your fallback policy?

Reasons to say "No" - the most significant part of an SRP. What are the show-stopper issues that must be resolved before the migration can be signed off? At this point you will aware of some of them. The ones being presented to you right now. But are there others? Will you resolve one lot of restrictions only to have a second lot presented to you? Remember, at this point the user community is running scared. No one likes change and this change is going awfully badly. There will be layer upon layer of objections. You need to get them all out in the open, resolved or mitigated and signed off

Key Data Stakeholder Analysis (KDSA)

It is almost always the case, when we arrive at the site of the disaster that this step has been done badly. KDSA is not a RACI spreadsheet. If that's all you have – well it's a start but the passive activity of approving a design is a long way short of the active responsibility of specifying the SRP. It's really an invitation to accept the power to disapprove without the responsibility to resolve. A proper set of Key Data Stakeholders should include:

Data Owner - the Data Owner is NOT the person in the IT departments filing system with titular responsibility for the data stores you are migrating from or to. A Data Owner is any person within the organisation with the de facto authority to prevent a migration

from occurring. These are the people who must sign off the decommissioning certificate. They each should have completed an SRP

Business Domain Expert - this is not a technical role. This is a business person with day to day access to the system. They understand what it means in business terms. On very large systems you normally have a smaller set of Business Domain Experts but ones that are trusted by their colleagues and can reach out to the appropriate person in the business to answer any question

Technical System Expert - normally you are knee deep in these, at least for the target system but don't confuse them with the Business Domain Experts unless they also have day to day, hands on, experience of the systems you are migrating

Regulatory and other - there are a lot of potential Key Data Stakeholders but each one should have a clearly defined role

Data Quality Rules
These are both a set of artefacts and the process for managing all data related issues within a migration project. If you are following the method preached in my book Practical Data Migration then you will have set them up. If not you will still (probably) have some mechanism for investigating, prioritising and resolving data issues (even if the prioritisation is the usual mix of using issue management, technical feasibility, peer pressure and generally who shouts the loudest. Actually often when programmes are really under pressure to deliver something, prioritisation becomes an issue of what can be delivered first). Check your process for the following necessary features:

Method - is there one clearly defined method to handle all data related issues? One that isn't confused with the development/testing issues of the new stack? Are all your issues in one place or are they scattered around, some in the testing suite, some in fallout, some on the issues log etc?

Participation - does the data issues process fully include the Data Owner, Business Domain Experts and other Key Data Stakeholders in both prioritisation and delivery?

Scaling - are all of your known data issues fully quantified? Can you report precisely for each issue your percentage complete on delivery?

Prioritisation - is prioritisation driven be the active participation of the Business Domain Experts sanctioned by the Data Owners? Not as signoff's but as an informing part of the decision process? Does prioritisation include the option to do nothing and allow existing "bad" data to go over without improvement?

Delivery - are all available options considered? That is:

- → Fix in the source
- → Fix in the migration software
- → Allow to fallout and load manually
- → Migrate and fix on the target
- → Migrate and leave as is
- → Don't migrate?

Decision-Making - are the Data Stakeholders party to the delivery decision making? Or is it decided by a cabal of "techies" then presented as a done deal to the business? Where remedial actions are being considered are they controlled and reported on as part of a methodical approach that can be measured?

3. Planning

With your checklist in hand you should now have a feel for where the programme is failing and will not be confusing causes with symptoms. I recognise that buying time is easier for us in iergo because we are outsiders and are only brought in when management is prepared to acknowledge they have a programme that is failing.

We also have the benefit of having a tightly integrated set of criteria, the general outline of which is above, linked to industry measures so we can show degrees of risk and likely outcomes in terms of financial and time over-runs. We also don't own the problem but have the impenetrable sheen of disinterested professionals.

If you are reading this, however and you have been involved in the programme from the beginning then requesting extensions and sudden changes of tack might be suicidally counter-productive. I'm afraid I leave it to you to decide how far you share your insights with your senior management.

However, you now have a detailed analysis of the shortcomings. I don't suppose you really need this piece of advice but you have to stop chasing the programme and get back in charge.

The following are the classic steps that we find most failing programmes need to correct:

Data Architecture - get the list of Legacy Data Stores under change control. Does everyone on the programme know where to go to see a full list of the data sources? Is it easy to see which data sources in the plethora that are available to you contain which of your Conceptual Entities? Is it easy to find out which data sources will be expected to have matching Conceptual Entities for the migration (e.g. can you see which application within billing, has to have a matching customer reference to which application with sales)? Do you have an agreed Legacy Data Store Model against which each candidate legacy data store can be measured for consistency? Are you performing formal Landscape Analysis or are you going for the "Poke And Hope" approach (i.e. throw stuff at the target and see what fails)?

Data Quality Rules - get the list of data issues under change control and get them into your Data Quality Rules process. In panic mode the number of issues can escalate rapidly but what we need at this stage is focus and prioritisation. Most projects only track the really big issues on an issue register leaving hundreds of smaller issues to go unnoticed. These issues can rapidly eat up project hours. Get each data issue logged as a data quality rule so that the relevant Technical Systems Expert and Business Domain Expert are closely involved and working together for a solution.

Key Data Stakeholders - these people are essential and are often lacking on a failing migration. These people can help you prioritise and resolve the data issues and you must tactfully convince them to take responsibility for the data in their area. This is not easy if they have been used to being consulted but not having any responsibility to the programme for delivery.

Delivery - focus on consulting, not fixing. In panic mode there is often the temptation to fix every issue that arises. Stop. Accept that you have missed important data gaps and begin to educate sponsors and stakeholders that you will be delivering the project iteratively from this point on using a prioritised process.

Do the do-able - your analysis will tell you where you are going wrong but at this point you can't hope to fix everything. It is just not technically or politically feasible. Concentrate on iterative increments. Accept that knowing who the Key Data Stakeholders are and getting them to accept their responsibilities are two different things. Using the Data Quality Rules process will bring them into the project. You will not at this point in time be

allowed to perform a formal SRP creation process, but be alert to Reasons To Say No as they come out in conversation. Never let one slide by. Always ask what it would take to close it out. Record it. Build up trust.

Most data migration projects fail because they lack a cohesive data migration methodology that is converted into an effective plan. Create a checklist based on the pointers mentioned to see how close you are to industry best practice then amend your plan accordingly.

Working in a far more controlled manner after implementing our stabilisation phase we can start to plan releases in an iterative fashion that deliver measurable improvements to the current situation and give benefits back into the business.

Here are some final activities in this phase:

Prioritisation

Start prioritising faults and create realistic estimates of how long these will take to resolve. Data quality prioritisation is also critical, we always say "No enterprise wants, needs or will pay for perfect quality data" however in the rush to recover the project the temptation to load data will be great. If your data is sub-standard, how will it be corrected in a later iteration? How will it affect the load process? Have these risks been communicated to the Data Stakeholders? Your project may not necessarily need perfect quality data but you do need to understand and communicate the risks.

Change control

Tighten up change control and configuration management procedures. The fire-fighting mentality of failing projects has to be transformed into a stakeholder-driven release plan with clearly defined deliverables that the business can plan for.

4. Mop-up

In a well-managed migration there are a number of activities that would have been completed as a matter of course but when in failure mode can be forgotten.

Our mop-up phase ensures we focus on the critical migration tasks but are mindful of our post-migration house-keeping duties. At this stage there are probably a wealth of best-practices that have been omitted such as key data stakeholder management, data quality rules, legacy data store management and many more.

Here we introduce three other key project activities that are typically lacking but still need to be addressed:

System Retirement Policies

We cover these in great deal in the book but in most struggling projects they will have been omitted and this can be a real challenge. The best solution is to implement a system retirement policy from the outset and get this logged on the programme issue log so it has full attention. If you have no system retirement policies in place then you will certainly need to exercise some political sensitivity as system owners may be antagonistic if they've not been involved.

Key Business Data Areas

Chances are you've now got a lack of knowledge on exactly where your consistency problems exist between data stores. It's unlikely you will be given sufficient time to completely re-plan and do this correctly but certainly look to address consistency issues across your legacy data areas in subsequent iterations.

Reality Check

Again, it's unlikely you will have sufficient time to analyse the accuracy of your migrated data compared to business reality. However, that doesn't mean it should be forgotten. Use the legacy data stores and business experts you found in the stabilisation phase to help you corroborate the data. Ensure the data stakeholders are responsible for prioritising remedial work post implementation and start to transition your data quality rules framework (set up in the previous phases) over to the enterprise for ongoing data quality management.

5. In Summary

The lesson here is is a rather obvious one that by adopting data migration best-practices there is simply no need for your project to slip into failure mode. Use the checklists in this series to see how close your project is to best-practice even if you are yet to experience slippage.

By following the advice in this guide you should be able to assess whether your project has the correct approach and appropraite resources to be successful.

One of the main causes of failure is when the business takes a back-seat in the data migration.

Is your project IT-heavy? Are the right stakeholders, domain experts and business sponsors engaged and active in your project?

Integrating the enterprise into the project is probably the most important driver for migration success so my advice on a failing project is to at the very least focus on this critical area.

6. Do You Need Help?

If you have any questions about the topics raised or how to apply next generation data quality and data migration solutions on your project then please contact us:

E: **dataquality@experian.com**
T: **0800 197 7920**

Johny Morris
Director/Co-founder, Iergo Ltd

Johny Morris has over 25 years' experience in Information Technology spending the last 15 years consulting exclusively on Data Migration assignments for various industries such as financial services, utilities and telecommunications. His book "Practical Data Migration" defines the leading Data Migration methodology – PDM. Johny co-founded the specialist data migration consultancy, Iergo Ltd and is on the expert panel of Data Migration Pro.

> **About Experian Data Quality**
> Experian Data Quality has built up exceptional market coverage assisting customers with their unique data quality challenges. We provide a comprehensive toolkit for data quality projects combining our market leading software with a vast scope of reference data assets and services. Our mission is to put our customers in a position to make the right decisions from accurate and reliable data. The size and scope of data management projects varies considerably but the common factor in all ventures is unlocking operational efficiency and improving customer engagement. We see the potential of data. Whether it's in enabling ambulances to be sent to the exact location of an emergency or attributing charitable donations to the people who need it the most - data accuracy makes all the difference to service provision.

Experian Australia Pty Ltd
Level 6, 549 St Kilda Road
Melbourne, VIC 3004
T: (03) 8699 0100
info@experian.com

Experian™
Data Quality

Intelligent interactions.
Every time.