

PRACTICAL DATA MIGRATION

Second edition

Johnny Morris



Computer Weekly readers can get 20% discount on this book. Enter code **CWB3** at www.bcs.org/books/datamigration2e

PRACTICAL DATA MIGRATION

BCS, THE CHARTERED INSTITUTE FOR IT

Our mission as BCS, The Chartered Institute for IT, is to enable the information society. We promote wider social and economic progress through the advancement of information technology science and practice. We bring together industry, academics, practitioners and government to share knowledge, promote new thinking, inform the design of new curricula, shape public policy and inform the public.

Our vision is to be a world-class organisation for IT. Our 70,000 strong membership includes practitioners, businesses, academics and students in the UK and internationally. We deliver a range of professional development tools for practitioners and employees. A leading IT qualification body, we offer a range of widely recognised qualifications.

Further Information

BCS The Chartered Institute for IT,
First Floor, Block D,
North Star House, North Star Avenue,
Swindon, SN2 1FA, United Kingdom.
T +44 (0) 1793 417 424
F +44 (0) 1793 417 444
www.bcs.org/contact



PRACTICAL DATA MIGRATION

Second edition

Johny Morris



© 2012 Johnny Morris

First published in 2006

All rights reserved. Apart from any fair dealing for the purposes of research or private study, or criticism or review, as permitted by the Copyright Designs and Patents Act 1988, no part of this publication may be reproduced, stored or transmitted in any form or by any means, except with the prior permission in writing of the publisher, or in the case of reprographic reproduction, in accordance with the terms of the licences issued by the Copyright Licensing Agency. Enquiries for permission to reproduce material outside those terms should be directed to the publisher.

All trade marks, registered names etc. acknowledged in this publication are the property of their respective owners. BCS and the BCS logo are the registered trade marks of the British Computer Society charity number 292786 (BCS).

Published by BCS Learning & Development Ltd, a wholly owned subsidiary of BCS The Chartered Institute for IT, First Floor, Block D, North Star House, North Star Avenue, Swindon, SN2 1FA, UK.
www.bcs.org

ISBN: 978-1-906124-84-7

PDF ISBN: 978-1-780170-05-3

ePUB ISBN: 978-1-780170-41-1

Kindle ISBN: 978-1-780170-32-9

British Cataloguing in Publication Data.

A CIP catalogue record for this book is available at the British Library.

Disclaimer:

The views expressed in this book are the author(s) and do not necessarily reflect the views of the Institute or BCS Learning & Development Ltd except where explicitly stated as such. Although every care has been taken by the authors and BCS Learning & Development Ltd in the preparation of the publication, no warranty is given by the authors or BCS Learning & Development Ltd as publisher as to the accuracy or completeness of the information contained within it and neither the authors nor BCS Learning & Development Ltd shall be responsible or liable for any loss or damage whatsoever arising by virtue of such information or any instructions or advice contained within this publication or by any of the aforementioned.

Typeset by Lapid Digital Services, Chennai, India.

Printed at CPI Antony Rowe Ltd, Chippenham, UK.

To Josephine, with my love, for being there.

CONTENTS

Figures and tables	xi
Author	xii
Acknowledgements	xiii
Abbreviations	xiv
Glossary	xv
Useful websites	xviii
INTRODUCTION	1
What is the purpose of this book?	1
What types of data migration are covered?	1
What is not covered in this book?	1
Who is this book aimed at?	2
What is new to version two of <i>Practical Data Migration</i> ?	3
Paragraph styles	3
SECTION 1: EXECUTIVE OVERVIEW	5
1 DATA MIGRATION: WHAT'S ALL THE FUSS?	7
What is data migration?	7
First the bad news	8
So what usually goes wrong?	9
The responsibility gap	10
How to avoid the responsibility gap	15
2 GOLDEN RULES AND SUPER SMART TASKS	16
Golden Rules: the most important lessons	16
Golden Rule 1	17
Golden Rule 2	23
Golden Rule 3	24
Golden Rule 4	26
Super SMART Tasks and the mid-change cycle slump	28
3 PDMv2 OVERVIEW	35
Introducing <i>PDMv2</i>	35
Introducing the technology	40
Technology overview	41
The case for specialist technology	43
<i>PDMv2</i> provides all the answers	44

4	CREATING A MIGRATION STRATEGY	46
	Starting out on the right foot	46
	<i>PDMv2</i> data migration strategy	46
	What is in a <i>PDMv2</i> data migration strategy document?	47
	Data migration strategy checklist	74
	Getting the data migration strategy implemented	75
	SECTION 2: TOOLS AND TECHNIQUES	77
5	METADATA AND KEY BUSINESS DATA AREA DECOMPOSITION	79
	Metadata and <i>PDMv2</i>	79
	Uses of data models in <i>PDMv2</i>	81
	Instantiating data models	84
	Developing a conceptual entity model	85
	Project decomposition and key business data areas	86
6	DEMILITARISED ZONES AND KEY DATA STAKEHOLDER MANAGEMENT	89
	The impact of the demilitarised zone	89
	Introduction to key data stakeholders	91
7	LANDSCAPE ANALYSIS	105
	What is landscape analysis and why do you do it?	105
	Legacy data stores	106
	Landscape analysis as a Super SMART Task	108
	When to perform landscape analysis	109
	Documenting legacy data stores	109
	Profiling, data gaps and master data management	114
	Software tools to support landscape analysis	125
	Impact of the DMZ	125
8	SYSTEM RETIREMENT PLAN	127
	Why have a system retirement plan?	127
	What are system retirement plans?	127
	How and when to implement SRPs	128
	SRP minimum requirements	130
	SRPs and Super SMART Tasks	138
	Impact of the DMZ on SRPs	139
9	DATA QUALITY RULES	140
	Introducing data quality rules	140
	How do you generate DQRs?	141
	The DQR process	142
	How a DQR board works	147
	Software support	156
	Role of the DMZ	157
10	GAP ANALYSIS AND MAPPING	158
	Gap analysis	158
	Gap analysis as a service	159
	Data quality monitoring as a service	163

	Mapping	163
	Data lineage, data audit and the 'one-way street' problem	167
	Impact of the DMZ	170
	Software support	173
11	MIGRATION DESIGN AND EXECUTION	175
	Migration end-to-end design	175
	Extract, transform and load designs	181
	Detailed decommissioning design	195
	Migration build, test and execute	198
12	LEGACY DECOMMISSIONING	205
	Executing the decommissioning design	205
	SECTION 3: FAILING DATA MIGRATION PROJECTS	209
13	RESCUING FAILING DATA MIGRATION PROJECTS	211
	Introduction	211
	Stabilisation	212
	Planned activity	214
	Post-implementation mop-up	215
	Impact of the DMZ	216
	Conclusion	217
	APPENDICES	219
A1	CONFIGURABLE ITEMS	221
A2	IMPACT ANALYSIS FRAME	223
A3	DATA MIGRATION STRATEGY CHECKLIST	226
A4	SPECIMEN DQR FORM	229
A5	DATA MAPPING EXAMPLE	231
	Index	245

FIGURES AND TABLES

Figure 2.1	The three circles of activity	29
Figure 2.2	Standard reaction to change	30
Figure 3.1	A diagrammatic representation of <i>PDMv2</i>	35
Figure 4.1	Quality versus time versus budget	52
Figure 5.1	An example ERD	80
Figure 5.2	ERD relationship symbols	81
Figure 5.3	The involuted data relationship	81
Figure 5.4	Example conceptual entities	85
Figure 9.1	The DQR process	141
Table A1.1	Some <i>PDMv2</i> configurable items	221
Table A2.1	<i>PDMv2</i> impact analysis frame	223
Table A3.1	Data migration strategy checklist	226
Table A4.1	Specimen DQR form	229
Table A5.1	Target system equipment table	232
Table A5.2	Legacy Equipment Item Type table	233
Table A5.3	Target Equipment Type table	233
Table A5.4	Equipment type cross-reference table	234
Table A5.5	Equipment cross-reference table	235
Table A5.6	Location cross-reference table	236
Table A5.7	Partially completed example data mapping table	240

AUTHOR

Johnny Morris has over 25 years' experience in IT working as a programmer, analyst, project manager and data architect. He has worked as an independent contractor for the last 20 years and in that time has worked for some of the biggest names in IT consultancy including CSC, Logica CMG and Price Waterhouse Coopers (PwC). He specialises in data migration and integration and for the last 14 years has been involved with large-scale migrations at blue chip clients like Barclays Bank, National Grid Transco and British Telecom.

ACKNOWLEDGEMENTS

I would like to thank my publisher, Matthew Flynn, who showed great patience as I struggled with the text; Jutta Mackwell for her patient chiding that finally got me over the line; and Nina Turner with whom I started this data migration journey and whose insistence on only providing the best we can is a rule I have stuck with. Dylan Jones of Data Migration Pro deserves a mention for the many conversations we have had over the years where ideas have been batted around – he has been one of the most consistent voices demanding that this under-developed corner of IT deserves better recognition. Finally, my thanks go to my wife, Jo, without whose support this book would not have been completed.

Johny Morris

July 2012

ABBREVIATIONS

API	Application program interface
COTS	Commercial off-the-shelf
CRM	Customer relationship management
DBA	Database administrator
DMZ	Demilitarised zone
DQR	Data quality rule
ERD	Entity relationship diagram
ERP	Enterprise resource planning
ETL	Extract, transform and load
GAM	Gap analysis and mapping
KDSH	Key data stakeholder
KDSM	Key data stakeholder management
LA	Landscape analysis
LD	Legacy decommissioning
LDS	Legacy data store
MDE	Migration design and execution
MDM	Master data management
MIS	Management information system
MSG	Migration strategy and governance
PDM	Practical data migration
PID	Programme initiation document
PMO	Project management office
SI	Systems integrator
SME	System matter expert
SRP	System retirement plan
UAT	User acceptance testing

GLOSSARY

Check point A decision point at which it is agreed a new system is stable enough to go forward with or from which fallback occurs. (Also sometimes known as a 'go/no-go point'.)

Churn The relative frequency with which records of different types are added, amended or deleted from a data store.

Conceptual entity model A form of data model where atomic entities are grouped together to form higher level entities that are meaningful to the enterprise.

Control total Either the sum of some meaningful value within the data being transferred or a count of the number of units of migration being transferred.

Data architect The person responsible for the design of how the data required for an organisation, possibly held over multiple applications, is held.

Data audit The verifiable proof that all the units of migration in the legacy data stores are accounted for in the migration.

Data freeze The prevention of updates to records after they have been extracted for data migration and before they have loaded into the new system.

Data lineage The history of transformation that shows how an individual data item is transformed from one system to another.

Data mapping The rule(s) by which one or more items in the Legacy Data Store will have their values moved to one or more items in the new system.

Data migration The selection, preparation, extraction, transformation and permanent movement of appropriate data that is of the right quality to the right place at the right time and the decommissioning of legacy data stores.

Data owners All the people within or outside an organisation who have the legitimate power to stop a migration from happening.

Data quality rules A set of processes and deliverables that are used to measure the quality of the data within a data migration project and to resolve or mitigate data quality issues.

Data size The amount of data to be loaded.

Data stakeholder Any person within or outside an organisation who has a legitimate interest in the data migration outcomes.

Data transitional rules The temporary business operating procedures put in place to cope with the disturbance caused by data migration itself.

Demilitarised zone The interface between the technology provider and the wider programme.

Entity type The generic description of an entity.

Fallback The steps that will be taken to get an enterprise back into the position it was in prior to a data migration.

Fallback window The length of time between starting up a new system and taking the final check point that allows for the full decommissioning of legacy data stores according to the system retirement plans.

Homonym Two words that are spelled the same way, but have different meanings.

Instance A particular example of an entity type.

Key business data areas The segments into which a large data migration project is broken down for management and planning purposes.

Landscape analysis The systematic discovery, review and documenting of the legacy data stores, including their linkages, data quality and key data stakeholders.

Legacy data store A data repository of any type that holds data of interest to the new system.

Metadata Data about data. The data that technologists hold about the data in the business.

Migration form The technical style of a migration.

Navigation The links in the data that allow software to move from one data item to another. An example would be a foreign key that allows a program to get from a holding company record to all the operational company records beneath it.

‘One-way street’ problem Occurs when an algorithm transforms data in such a way that the original values cannot be identified.

Policies The explicit or tacit underlying drivers for a project and for the surrounding environment in which a project operates.

Project A one-off enterprise event with a beginning, middle and an end.

Semantic issue A disagreement about the definition of a business term or the use of fields in corporate systems.

Sequencing The ordering of update processes into a tenable progression.

Source data store Synonym of legacy data store.

Synchronisation Enabling changes to data items in legacy data stores to be reflected in the target (forward synchronisation) or changes in the target to be reflected in the legacy data stores (reverse synchronisation) or in both directions (bidirectional synchronisation).

Synonym Two words that are spelled differently, but mean the same thing.

System retirement plan The user-side requirements of a data migration that will allow a legacy data store to be decommissioned.

Target The final destination system or systems.

Topography The map of data store linkages.

Training lag The length of time it takes to train all the staff who need to be trained in the target system.

Transitional data store A temporary database created during the process of data migration.

Unit of migration The lowest level of data granularity of meaning to the business.

USEFUL WEBSITES

www.bcs.org Publisher of this book and home of ‘Johnny’s Data Migration Blog’ – the author’s regular commentary and insights into data migration.

www.dama.org Data Management International – some good data quality and modelling information, but light on data migration.

www.datamigrationmatters.org The Data Migration Matters event series with forthcoming events and past presentations.

www.datamigrationpro.com Probably the best community website devoted to data migration.

www.dataqualitypro.com Sister website to datamigrationpro with lots of product reviews, comment etc.

www.iaidq.org The International Association for Information and Data Quality – good for local events and discussions, especially via the LinkedIn® website.

www.iergo.com The author’s personal website.

SECTION 1: EXECUTIVE OVERVIEW

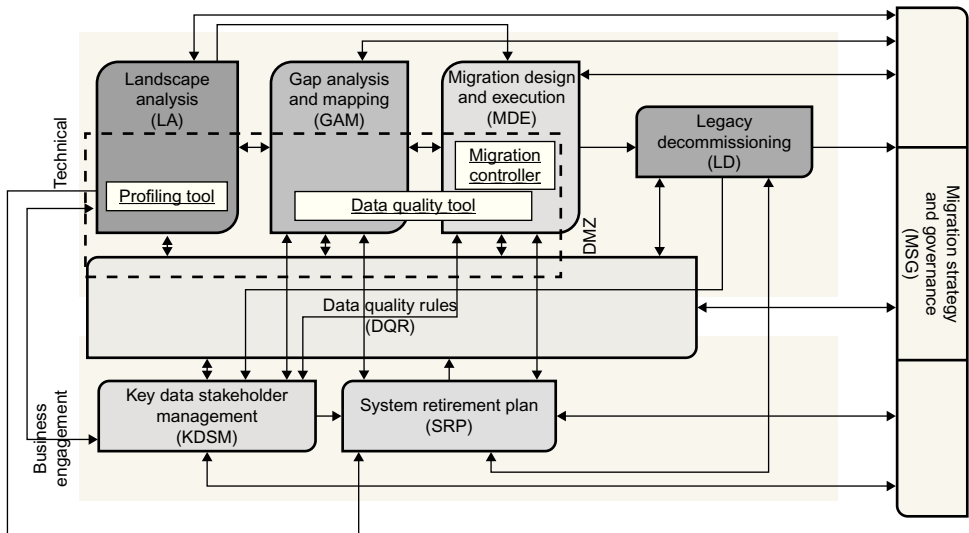
3 PDMv2 OVERVIEW

In this chapter I outline PDMv2 and show how it overcomes all the issues identified in Chapter 1 by using a set of integrated modules that cover the whole scope of a data migration from project start-up to legacy decommissioning and beyond. I give a brief overview of the types of software technology available to support data migration.

INTRODUCING PDMv2

PDMv2 is modularised with seven functional modules and one overarching governance and project control module. This modularisation helps, as you will see, in tailoring PDMv2 to other project delivery methods. Figure 3.1 illustrates PDMv2. The arrowed lines illustrate product flows between modules. It is clear from the diagram that it anticipates a degree of recursion in a data migration. For instance, at any point a new requirement for data might emerge from the larger programme of which a data migration forms a part. You might be concentrating at this point on testing your data migration solution, but you still have to handle the requirement. This is something over which you have no control within the data migration project, but it is something you can manage.

Figure 3.1 A diagrammatic representation of PDMv2



The functional modules within *PDMv2* are split over two work streams: business engagement and technical. Data quality rules (DQRs) span the two. It is important to note that even at this high level, *PDMv2* is designed from the ground up to integrate the business as well as the technical sides of the project. For *PDMv2*, business engagement is not a separate task, but is built into the way you go about your job.

Each of the modules is briefly explained below. More detailed explanations are given in Section 2, which provides sub-module, workflow-level descriptions.

Landscape analysis

The landscape analysis (LA) module uses various techniques to discover and catalogue legacy data stores (LDSs) and their relationship to one another. It is here that you look inside data stores to see how they work, what data they have and what data challenges they might contain. This is data profiling. It is performed by using both available software tools and manually. It is necessary to seek out consciously all the available LDSs, not just the official enterprise ones, welcoming all contributions. That way *PDMv2* turns a necessary technical activity into a set of Super SMART Tasks. Landscape analysis can commence prior to the design or even the selection of the target system.



HINT

If the decision on the target is still some way off, you can start on a small scale in an area that is likely to be rich in unofficial data stores. Start building the virtual team you will need to succeed. This also provides you with the opportunity to tailor the *PDMv2* deliverables to your programme management standards and to learn *PDMv2* by using it. Finally, data migration is a risky business. *PDMv2* helps de-risking by moving as much activity up the timeline as possible. Your mantra should be 'start small, start early'.

Do not expect that your selected systems integrator (SI) or software supplier can or will do much to help you in this area. Many of these LDSs are hidden away on departmental desktops, inaccessible to the reach of the SI. It is important to analyse and document how these data stores are linked (known as 'system topography') and include links across the scope boundary that receive or give data from and to stores that will be replaced. It is also a growing aspect of best practice that a fast iteration of LA is performed to quantify the scale of the data migration task you have in front of you prior to setting the budget for the remainder of the project.

Gap analysis and mapping

The gap analysis and mapping (GAM) module is where the data mapping takes place once the target system is available. Data mapping is the linking of fields in the LDSs to fields in the target, plus defining the transformation logic that is needed to split data up and merge fields. A classic example of this is reformatting name and addresses where perhaps a source database has the name in a single field, but the target holds the forename and surname separately and only holds the first line of the address, the rest being derived from a national postal file based on a postal or zip code.

HINT

It is perfectly normal for the target system to be delivered later than expected and in phases. Waiting for the target to be completely ready prior to starting data migration activities is a recipe for disaster. *PDMv2* gets around this by moving all discovery and profiling activity up the timeline to LA and the use of migration data models, which are explained in Section 2.



GAM is also where the mapping and gap analysis for legacy decommissioning design is performed so that an integrated target and archive design is developed. Again, it is absolutely standard for most SI or software suppliers not to be involved in designing a solution for those data items that the business might need (except rarely, for instance, where old transactions are preserved in case there is a tax inspection), but for which there is no place in the target. Therefore the design, build, test and execution of the archive solution is normally outside the DMZ, but driven by and dependent on the target migration design. After all, any item that is not moved to the target, but is needed by the business, has by definition to go into the archive, so these are activities that will need to be performed. *PDMv2* has them covered.

HINT

When reviewing proposals from suppliers, no matter how comprehensive they seem, comparing them against *PDMv2* will show you what is missing: in other words, what you will have to do yourself and therefore budget for. There are perfectly legitimate reasons, as you have seen from the discussion of the DMZ, for a supplier to limit their offering. Being aware of these missing elements, however, will allow you to plug the gaps.

**Migration design and execution**

The migration design and execution (MDE) module is where the physical design, test and execution of migration and archiving are carried out. Data migration is about more than just moving bits and bytes around. You have to be aware of business limitations, timings, audit requirements, data lineage, fallout, fallback, archiving requirements, reporting, management and control etc. MDE integrates all these elements within a single module solidly based on the business requirements expressed in the system retirement plans.

Legacy decommissioning

The legacy decommissioning (LD) module covers the physical or logical removal of legacy databases, hardware and software. It also covers the delivery of archived data storage for data items that have to be retained but which are not to be migrated to the target. There are also project close-down processes, including the handover of data quality issues (which it was not possible to fix within the project's time and budget constraints) to the in-life data quality teams (where they exist). Again it is absolutely standard for SI and software vendors to ignore the legacy decommissioning aspect of a data migration. You need to work with them through the *PDMv2* mechanisms to develop a single coherent design.



ANECDOTE

It is with some regret that I have to confess to not always finding willing recipients of those data quality issues that could not be fixed in the time and budget of a data migration project. However, at a corporate level, more organisations are making a conscious effort to manage data quality issues. As good corporate citizens we should endeavour to pass on what we can.

Data quality rules

The data quality rules (DQR) module is the centrepiece of what makes *PDMv2* unique. Sticking to the principle of Super SMART Tasks, this module manages all data quality and preparation-related activity on the programme. It integrates the legacy technical system experts, the target system experts and the business domain experts to prioritise, manage and complete all data issues, including the selection and exclusion of data sources. It is Super SMART because it builds the team by linking into the resources of the rest of the enterprise, creating a single virtual team; it builds the individual by empowering business colleagues and giving them the skills and opportunity to make a positive contribution to the project; and it completes the task by bringing much needed enterprise knowledge into a collaborative framework. DQRs are so important that Chapter 9 is dedicated to them alone.



HINT

To test how essential (and often missing) this element is in most data migration approaches, try the following. When analysing the bids of various suppliers, ask how they handle data quality issues. The common response is a long technical description of how they have sophisticated tools to trap errors. Your next question should be: 'And then what?' If you are lucky they will explain how they have an issues log that records the issues with appropriate dashboarding etc. Try another: 'And then what?' and you will be really pushing the boundaries of their method. The honest answer is: 'And then we wait while you, the project, come up with a fix.' They have no mechanism for getting that fix, only ones for uncovering errors and logging them. This illustrates the boundary of the DMZ. If you employ *PDMv2* and its DQR processes you will have one.

Key data stakeholder management

PDMv2 has its own specific role definitions for each key data stakeholder (KDSH). Key data stakeholder management (KDSM) manages the discovery, briefing and management of these individuals. *PDMv2* is very business-focused, so there are as many business as technical roles. *PDMv2* is very prescriptive when it comes to KDSHs. A full description of KDSHs is provided in Chapter 6, but from the business side the two most important stakeholders are data owners and business domain experts.

PDMv2 is quite clear about the definition of data owners.

DEFINITION

Data owners are all the people within or outside an organisation who have the legitimate power to stop a migration from happening.



This definition is not based on organisation charts of who has titular responsibility for a database. If a person can legitimately stop the migration occurring because their information is not adequately managed, then they are the data owner of that piece of information. This means that the often forgotten, but powerful, individuals like financial controllers, who can stop migrations if the results compromise their needs, are equal data owners with the people who work directly with a data store (but only for the data items that impact them). Each data owner is expected to take part in the system retirement planning process, all of which is explained in more detail in Chapter 8.

The system owners, however, are often senior executives who cannot be expected to answer every query coming out of the project personally and who, in any case, often have no direct, day-to-day, hands-on experience of using the systems in question. They will generally defer to nominated business domain experts who they empower to input to the various meetings and provide the detailed knowledge to the project.

This is all very well, you might say, but how do you get the commitment of these powerful few to your project? We all know how difficult it is to get real commitment from colleagues working in silos different from our own. In this case, you leverage the compelling event of the forthcoming system retirement to grab attention and make your needs real to people.

System retirement plans

PDMv2 does not begin its conversation with the business by asking about data mappings, data quality, data gaps, data lineage or any other esoteric technical feature. *PDMv2* begins with the ultimate goal of a data migration: turning off legacy data stores.

HINT

It might seem crudely brutal, but I have found in practice that the bold unvarnished question 'You know system X that you depend on to do your day job? Well, we are turning it off on (enter date). How will you be certain that you will be able to continue with your day job once that has happened?' works better than anything. I leave it to you to dress it up in language appropriate to your situation, but I find the unchallengeable simplicity of this statement works best.



Going back from this you uncover all the things that must be done so that data owners will be comfortable with signing off the decommissioning certificate. You seek and encourage objections to going forward. Getting these objections shows

you have moved the subject over the hump of denial, down the slopes beyond rejection, to at least the negotiation of reluctant acceptance, if not to positive acceptance. Starting the conversation this way makes real the compelling event of a data migration.

You provide reassurance by explaining that you will be proceeding through a controlled sequence of iterations (the data owners will have to sign off the system retirement plan (SRP) at least three times before you get to the decommissioning certificate) that allows them to be ready and confident in the migration and that they have made you aware of all the things you must do to satisfy them.

Under the guidance of the *PDMv2* model you ask a series of structured questions that elicit the business view of the migration, looking at necessary items like business migration audit requirements, data lineage requirements, data retention requirements, migration restrictions, user acceptance testing requirements, go-live restrictions, fallback requirements, units of migration definitions, migration resource requirements and data transitional processes (all of which are explained in detail in Section 2). All responses are expressed in business terms.

The SRP is your business colleagues' main view of the migration and is a key input to migration design and execution.

Migration strategy and governance

The migration strategy and governance (MSG) module covers all the standard programme management functions that are expected on a well-managed project, plus some unique activities that are mandated by *PDMv2*. Section 2 gives a full description of MSG; however, from an overview perspective there is one task that must be completed with the involvement of the senior management of the whole programme, that is the creation of a data migration strategy and that follows in Chapter 4.

Demilitarised zone

As you have seen, the demilitarised zone (DMZ) is the interface between the work of the technology supplier and the responsibilities of its clients. The DMZ is a key component of *PDMv2* that will be, to an extent, formally defined in the contract with the supplier. However, the DMZ is wider than the contract and its formal definition will help both sides understand and manage their reciprocal dependencies. Throughout the rest of this book, I will constantly refer to the DMZ and its impact on each of the *PDMv2* modules.

INTRODUCING THE TECHNOLOGY

So far I have looked at the softer issues around a data migration because these are typically where the project is likely to go wrong. I will now introduce the standard technology that underpins data migration projects. A more detailed description can be found in appropriate places in Section 2 where the use of the technology is covered.

TECHNOLOGY OVERVIEW

There has been an explosion in the technology available to assist with data migration over the last few years and it seems that each newcomer to the market has a different take on how to perform the necessary tasks. However, broadly speaking, there are three phases of activity where specific technology support is available: data profiling; data quality; and migration control. I will also discuss generic project-supporting software (hubs and workflow tools).

Data profiling tools

Data profiling tools allow you to analyse the legacy databases and discover unforeseen features at column and row level. Some tools on the market also allow cross data source analysis. This means that you can compare, say, customer names in a sales ledger with customer names in a customer relationship management (CRM) database, even though they are from different vendors. The majority of tools work at the database level, analysing and comparing fields, but some tools will even interrogate source code to find relationships that are in the code but not in the database management system. (This only works, of course, when you have access to the source code, but is very helpful in old, locally built legacy systems where validation rules have been lost in the mists of time.)

The use of profiling tools is a vital first step in producing data of the quality that will load into your new system. As you will now be aware, you must anticipate that there will be surprises in the LDSs. It is a common misconception that the use of tools can only start once the target system has been fully defined. This is not the case. As you have seen, semantic issues, the most difficult and time-consuming to resolve, will be issues whatever the new system is. Be prepared to profile data prior to the definition of the target system.

However, data profiling tools are almost too good these days and will generate large amounts of information. You will need a process to winnow the wheat from the chaff. Fortunately *PDMv2* comes complete with just such a process: DQRs. By just using data profiling tools without the DQR process you risk drowning in a sea of possible issues. Data profiling tools are also limited in what they can access. Although some can compare spreadsheet data, most are restricted to looking at corporate standard databases and none can look easily into hard copy sources like rolodex, notebooks etc.

Similarly, data profiling tools will not uncover, on their own, the hidden data sources that are not linked by database level exchanges of data. The clerk with the data stick, moving data from one machine to another before creating figures that are re-entered manually into an old application might be essential to regulatory processes, but will be invisible to any tools. To discover these you need to rely on the more pedestrian methods within the LA module and the creation of your virtual team.

However, having dwelt on the limitations of profiling tools, best practice now recommends the use of profiling tools for projects of any size prior to setting budgets and plans for data migration. For smaller migrations there are plenty of free-to-use, studio editions of some of the leading toolsets. They tend to have enough

functionality for small datasets, but are limited in their integration, and therefore application, to larger programmes. On the other hand, the studio editions are also a good way to get a feel for what is on the market even if eventually you need to scale up to enterprise level software.

Data quality tool

Once you have an idea of the target, on the one hand, and the constraints of the LDSs, on the other, data quality software allows the speedy (most are ‘point and click’) implementation of validation and cleaning rules. There is obviously some overlap in functionality between profiling and data quality tools. Some software can span both spheres, but most are stronger in one than the other. The difference is that profiling tools discover relationships and possible data quality issues, whereas data quality tools check for and enforce known data quality rules. It is for this reason that the leading vendors tend to have both in their software sets. Ideally the rules discovered in profiling should be passed seamlessly for implementation in the data quality software, which in turn is fully integrated with the migration controller.

Once again, there are studio editions of many of the leading software offerings. These are worth looking at to get a feel for what is on offer if you have no local expertise.

Migration controllers

Migration controllers are often known as the extract, transform and load (ETL) tools; however, they are expected to do more than merely perform these three functions. They are the essential ‘on the night’ software that delivers the migration. Migration controllers need to be capable of performing the following functions:

- Reading data from the LDSs (the ‘extract’ step).
- Validating the extracted data (preferably using the data quality software seamlessly embedded into the migration controller suite).
- Reformatting the data and blending data from multiple sources (the ‘transform’ step).
- Scheduling, starting and stopping the migration process.
- Writing the data to the target (the ‘load’ step).
- Managing fallout.
- Managing fallback.
- Reporting on execution.
- Reporting on fallout.
- Providing audits.

More sophisticated products are also capable of many other features including:

- Synchronisation (keeping the changes to source and target data in step after the data has been moved. This means that the source can continue to be used during the migration thus enabling zero downtime migrations).

- Data lineage (tracking individual units of migration to show how they were transformed, combined and written to the target. This is sometimes a necessity for regulatory reasons and is also helpful for other technical reasons covered in Section 2).

Modern, built-for-purpose migration controllers have so many complex features that it is unlikely that you would be successful in replicating them in locally produced software.

Hubs and workflow

It is commonplace these days on large projects to implement some form of hub. This is an area where documents can be shared and different software is used to facilitate collaboration. Good data migration projects are a hive of collaboration, with the disparate groups of workers sharing knowledge. Workflow engines allow the output of, say, records that have fallen out of the migration to be routed from the migration controller to the correct team for analysis. The use of these tools depends on the scale of the migration. Small, co-located teams, have less need for collaboration tools than large project teams spread over a number of time zones and continents, but the ubiquity of this software and its relative low cost compared with the benefits in productivity and knowledge sharing make it all but essential. At the time of writing there has been a surge in interest in using social networking type software for collaboration on projects. This really is an area where you need to discuss what is available for your project with your architectural resource.

THE CASE FOR SPECIALIST TECHNOLOGY

It is possible to complete a data migration using the code writing capabilities of an indigenous IT department, but there are risks. All code writing is inherently risky. Bugs work their way into the code and then need to be weeded out in testing. The more coding you do the more bugs, the more testing, the more time, the more risk. Using software designed for the job reduces the number of bugs dramatically because it is only the logic of the migration that is being tested not the logic as it was instantiated in someone's handcrafted code.

A second compelling reason is that off-the-shelf data migration software these days is incredibly sophisticated. There is no way that any company, other than a rival software vendor, could ever justify the investment to recreate even a tenth of the features available out of the box.

ANECDOTE

I do know of one large technology company that decided in a fit of hubris to build their own migration hub. I have been quietly tracking the project and after five years and considerable expense they have not succeeded in replicating even the features of one of the cheaper off-the-shelf products. I suspect that their business drivers must go beyond merely delivering a solution.



All that being said, it remains a business judgement as to just where the cost–benefit lies. Tools can be purchased separately or as part of a set. Each migration has its own set of issues. For instance, is it a true 24/7 environment (like production control systems or telephony) where you really can never shut down the systems? If so, investment in sophisticated migration controllers that allow zero downtime are called for. Or do you have a complex heterogeneous migration environment to migrate from where the use of leading edge data profiling tools capable of cross database analysis will be useful? Possibly you are in a heavily regulated environment where data lineage is vital. Maybe you have only a short window of opportunity to migrate the data, in which case tools with superior prototyping capability could be key. And so on.

Only you know your own migration challenges and drivers. On the one hand, you do not want to invest in heavyweight technology you do not need. On the other, you do not want to be hamstrung trying to manhandle a migration via an inappropriate vehicle. It really pays to take advice at this point from internal or independent experts who can refine your thinking and define your options.

***PDMv2* PROVIDES ALL THE ANSWERS**

A review of the issues identified in the preceding text that have historically damaged data migration projects shows that *PDMv2* provides a solution that covers all the bases but is flexible enough to be deployed in partnership with your chosen supplier’s own preferred approach.

Techno-centricity

PDMv2 sees data migration as a business-led joint IT–business activity, albeit with a clear view of where technology sits and how to make informed decisions about which technology is appropriate for your project. Through the DQR and SRP processes the business provides direction over data selection, data preparation, data quality, decommissioning etc., and takes ownership not just of the end point but of the process of getting there. All this is completed within an integrated set of linked activities.

Lack of specialist skills

PDMv2 provides the tools and techniques that are all you need to perform low-risk data migrations. You can use it either as a checklist against likely internal resources or seek training in specific skills or even the whole methodology.

For those partners, like implementation service suppliers, who will perform the final ‘lift and shift’ and who have their own preferred approach, *PDMv2* has the DMZ concept to insulate them from having to change their approach, which might well be optimised from their knowledge of their own technology and the target systems’ load requirements.

Underestimating

PDMv2 provides two ways of managing estimates. Firstly, there is the LA module that can be run separately on a fast pass through basis to generate the understanding on which estimates of scale can be arrived at prior to setting the budget for the

rest of the migration. Secondly, when the project is in-flight there is an assumption (Golden Rule 3) that more issues will be generated than can be solved, but there is the DQR process to manage the prioritisation needed to get the appropriate data of an appropriate level of quality to the right place at the right time. As I have shown, knowing there is an issue and solving an issue is not the same thing, especially if the issue is a business-side or semantic problem. Only *PDMv2* has built in the controls of these decision-making processes via DQR that reach beyond the boundaries of the project. Spotting the challenges early while working in a collegiate, virtual team, with the business taking a leading role, means that time, quality or budget can be flexed in a dynamic but controlled manner. Put together as a coherent whole you have a set of processes that allow sensible decisions to be made that will deliver the data you need at the time you want it within a budget that you can accept.

Uncontrolled recursion

Using *PDMv2* from the start of the project will build the single virtual team across business and technical stovepipes that will ensure that the responsibility gap never has a chance to develop. Via the DQR and SRP processes, you retain a tight control on your migrations using common metrics across multiple different data types, data sources, geographical locations etc.

Technology

Within *PDMv2* there is an understood place for technology and a clear decision point for deciding which technology is appropriate to your migration. However, technology is not divorced from the rest of the migration processes. Technology on its own, in any walk of life, rarely solves any problems without being wrapped in some form of meaningful best practice. Technology is embedded in the modules that make up *PDMv2* and is therefore still directed by business towards optimally aiding your data migration. If necessary, and to get the maximum benefit from your software investment, encapsulation in the DMZ means that you do not expect to have to destabilise your chosen implementation partner's favoured approach, which will optimise the use of their toolset's salient features.

To achieve all of this, however, you need to make sure that you have set up your project in the right way in the first place: in other words that you are setting off from the right starting point. For that, you need to get your data migration strategy right in the first place. This is the subject of Chapter 4.

CHAPTER REVIEW

In this chapter I have introduced you to the various modules of *PDMv2* and showed their interrelated nature. I also looked at the use of technology and you saw how it overlays the activities within the modules. Finally, you saw how the use of *PDMv2* mitigates all the risks, both technical and non-technical, that I identified in previous chapters.

INDEX

- abstraction 79
 - accreditation in *PDMv2* 61
 - always up migration 54
 - amnesties, legacy data stores 66, 100, 107
 - application program interfaces (API) 182, 191
 - architectural diagram 178
 - archived data storage
 - delivery of 37
 - design 37, 195–197
 - implementation 205
 - testing 131
 - assessments, data quality 110, 154–155
 - audit *see also* data audit
 - audit experts
 - on DQR board 149
 - identifying 101–102
 - automated fixes of data 145–146
 - awareness training 130
 - benefits, post-project 206–207
 - big bang migration 54
 - 'black boxing' 90
 - budgeting 68–69
 - budgets
 - for project 49
 - for software 57–58
 - see also* quality versus time
 - versus budget
 - build, migration 198–199
 - business colleagues
 - gaining support of 32–34
 - losing trust of 31–32
 - see also* key data stakeholders (KDSH)
 - business domain experts
 - on DQR board 148
 - identifying 95–96, 137
 - and testing 131
 - business engagement
 - communications strategies 65–67
 - need for 9, 17–23
 - overcoming resistance 28–34
 - ownership 91–92
 - see also* key data stakeholders (KDSH)
 - business engagement work stream
 - Fig. 3.1, 36
 - business knowledge, importance of 17, 23–24
 - business readiness 74
 - business readiness teams 67, 110, 129
 - business-class plane flight
 - metaphor 13–14
 - change, emotional reactions to
 - Fig. 2.2, 29–31
 - change control 64–65
 - in failing projects 212–213
 - of legacy data store list 110, 212
 - change management 70–71
 - check points 186–187
 - churn 183–184
 - close down, project 206
 - commercial off-the-shelf (COTS) packages 43–44, 118, 182
 - communications strategies 65–67
 - communications teams 67, 110, 129
 - complex datasets 111
 - concatenating data 168–169
 - conceptual entity model
 - definition 82
 - and migration models 118, 161
 - use in project decomposition 87
 - use of 83
 - worked example Fig. 5.4, 85
 - configurable items 70, 71
 - list of 221–222
 - configuration management 69–72
 - in failing projects 214, 217
 - conflicts of interest 89
 - connectors 59
 - continue with caution fallback 187–188
 - control totals, definition 133
 - corporate data architects *see* data architects
 - costs
 - of systems integrators 60
 - see also* budgets
 - customer experience, in system retirement plans 135
 - cutover plans 194, 195, 204
 - cutover timelines 178
 - dashboards 74, 142
 - data architects
 - definition 99
 - on DQR board 149
 - identifying 99–101
 - data audit 167, 170
 - definition 133
 - and demilitarised zone 172
 - design 194, 195
 - in end-to-end design 177, 179
 - fallback 189
 - manual 159
 - testing 200, 201
- data customers 102
 - data dictionaries 161
 - data gaps 114–120, 158–159
 - internal 115–116, 160
 - migration model 116–120, 121–122, 160
 - target model 114, 145, 158, 160–161
 - topographical 159
 - see also* reality gaps and checks
 - data governance 101
 - data lineage 133–134, 167–168, 170
 - and demilitarised zone 172
 - in end-to-end design 177, 179
 - fallback 189
 - software feature 43
 - testing 200
 - data mapping 36–37, 163–170
 - and demilitarised zone 171–172
 - 'one-way street' problem 169–170
 - parsing and concatenation 168–169
 - post-project benefits 207
 - rules 145, 165–167, 171–172
 - software for 173–174
 - worked example 231–244
- data migration
 - common problems 9–10
 - definition 7–8
 - where to start 75
 - data migration analysts
 - chairs DQR board 149
 - identifying 97–98
 - data migration strategy 46–75
 - checklist 226–228
 - communications strategies 65–67
 - contents of 47–48
 - implementation 75
 - initial migration plan 55–56
 - migration form 54–55
 - policies 50–53
 - project office functions 64–72
 - project scope 48–49
 - software selection 56–60
 - systems integrator selection 60–63

- data models
 - data store models 83
 - entity relationship diagrams
 - Fig. 5.1, Fig. 5.2, Fig. 5.3, 80–81
 - instantiating 84–85, 161–162
 - target data models 83, 117
 - types of 80
 - uses in *PDMv2* 81–83
 - see also* conceptual entity model; migration models
- data owners
 - definition 38–39
 - identifying 94–95
 - and system retirement plans 128–130
- data profiling
 - in landscape analysis 36, 114–120
 - services 125–126
 - tools 41–42, 125–126
- data quality
 - assessments 110, 154–155
 - compromise 24–26
 - enhancing 24, 26
 - monitoring 163, 171
 - removing duplicates 120–121, 124
 - responsibility of the business 10–12, 17–23
 - see also* quality versus time versus budget
- data quality rules (DQR) 38, 140–157
 - control of in failing projects 213
 - definition 140–141
 - and demilitarised zone 157
 - documentation 153–156, 229–230
 - escalation path 68
 - fix methods 143–144
 - fix types 144–146, 159
 - initial list of 72
 - monitoring 163, 171
 - post-project benefits 206–207
 - priorities 142, 154
 - process Fig. 9.1, 142–147, 151–152, 207
 - reporting on 73, 146–147
 - software for 156–157
 - specimen form 229–230
 - see also* DQR board
- data quality tools 42, 125, 146
 - and migration models 161–162
- data retention 131–132, 179
- data sizing 181–182
- data stakeholders *see* key data stakeholders (KDSH)
- data standards in legacy data stores 100
- data store models 83
- data transitional rules 143, 183–184, 235
- decommissioning *see* legacy decommissioning (LD)
- demilitarised zone (DMZ) 20–23, 40, 89–91
 - and data quality rules 157
 - definition 22
 - and end-to-end design 176–177
 - establishing 61–62
 - and extract, transform and load 190, 191, 192, 193, 195
 - and failing projects 216–217
 - and gap analysis and mapping 170–173
 - and in-house resources 23
 - and landscape analysis 125–126
 - and legacy decommissioning 177
 - logical 90
 - physical 90
 - and programme experts 98
 - and system retirement plans 139
 - and transitional business processes 193
- design *see* extract, transform and load (ETL) design; migration design and execution (MDE)
- DMZ *see* demilitarised zone (DMZ)
- documentation 69
 - data quality rules 153–156, 229–230
 - legacy data stores 109–114
- downsizing policies 53
- DQR *see* data quality rules (DQR)
- DQR board
 - composition of 148–149
 - frequency of meetings 152
 - running meetings 150–151
 - size 150
 - tasks 142–143
- emotional reactions to change
 - Fig. 2.2, 29–31
- end-to-end (E2E) design 175–181
- engagement *see* business engagement
- enterprise application migration *see* data migration
- entity model *see* conceptual entity model
- entity relationship diagrams (ERD)
 - Fig. 5.1, Fig. 5.2, Fig. 5.3, 80–81
- escalation path, data quality rules 68
- estimating *see* underestimating
- exclusion rules 145, 166, 171
- execution
 - legacy decommissioning 205–208
 - migration 203–204
- extract, transform and load (ETL)
 - design 181–195
 - and demilitarised zone 190, 191, 192, 193, 195
 - extract design 190
 - fallback design 185–189
 - load design 191–192
 - non-functional requirements 181–184
 - outputs 193–195
 - transformation design 190–191
 - transitional business processes design 192–193
 - transitional interface design 192
- extract, transform and load (ETL) tools 42–43
- extract design 190
- extraction rules 165, 166, 171
- failing projects, rescuing 211–218
- fall forward plan 136, 188
- fallback
 - contents of plan 186–189
 - definition 185
 - design of 185–189
 - forms of 187–188
 - reasons for 185
 - in strategy 66
 - in system retirement plans 135–136
 - testing 199
- fallback window 189
- fix methods 143–144
- fix types 144–146, 159
- framework purchasing agreements 62
- full fallback 187
- gap analysis and mapping (GAM)
 - 36–37, 158–174
 - data quality monitoring 163, 171
 - and demilitarised zone 170–173
 - gap analysis 159–162, 171
 - software for 173–174
 - see also* data mapping
- gaps *see* data gaps
- Golden Rules 16–27
 - and data quality rules 140, 152
 - importance of 16
 - rule 1 17–23
 - rule 2 23–24
 - rule 3 24–26
 - rule 4 26–27
- go-live restrictions 134–135, 179
- governance
 - data 101
 - migration strategy and governance module 40
 - project 49–50
- hardware considerations 184
- hubs 43
- impact analysis 70, 71
 - frame 223–225
- in-flight transactions, handling 138, 178, 192–193
- initial lists 72
- instance, definition 120
- integration testing 199
- internal gaps 115–116, 160
- issue management 67–68
- key business data areas 86–87, 177, 216
- key data stakeholder management (KDSM) 38–39, 102–103
- key data stakeholders (KDSH) 91–103
 - definition 93
 - documenting 110
 - in failing projects 213–214
 - identifying 92–102
 - initial list of 72
 - managing 38–39, 102–103
 - and ownership 91–92
- landscape analysis (LA) 36, 105–126
 - data profiling 36, 114–120
 - and data quality tools 161–162
 - definition 106
 - and demilitarised zone 125–126
 - documenting legacy data stores 109–114
 - identifying legacy data stores 36, 107–108, 110
 - master data management 120–124
 - software for 125
 - as Super SMART Task 108–109

- timing of 109
- see also* master data management (MDM)
- LD *see* legacy decommissioning (LD)
- legacy data stores (LDS)
 - amnesties 66, 100, 107
 - control of in failing projects 212
 - definition 106
 - documenting 109–114
 - identifying 36, 107–108, 110
 - initial list of 72
 - modelling 83
 - technical details of 111–112
 - technical system experts 148
 - topography of 112–113
- legacy decommissioning (LD)
 - and demilitarised zone 177
 - design 195–198
 - execution 205–208
 - introduction 37
 - reporting on 74
 - signing off 130, 138, 203
 - testing 199
- lessons learned 207
- library services 69, 73
- load design 191–192
 - see also* extract, transform and load (ETL) design
- load testing 199
 - see also* mock load testing
- loading rules 145, 167, 172
- logical migration models 161
- 'long tail' problem 206
- manual fixes of data 145
- mapping *see* data mapping
- master data management (MDM)
 - 101, 120–124
 - objective of 122
 - policies 52–53
 - post-project benefits 207
- MDE *see* migration design and execution (MDE)
- meetings, DQR board 150–151, 152
- metadata
 - definition 79
 - see also* data models
- metaphors for data migration 13–14
- method statements 155–156
- methodologies
 - project 49, 51
 - supplier-side 23, 61, 90
 - see also* PDMv2 methodology
- methods, fix 143–144
- metrics and data quality rules 156
- mid-change cycle slump Fig. 2.2, 29–31
- migration build 198–199
- migration controllers 42–43
- migration design and execution (MDE)
 - decommissioning design 195–198
 - end-to-end design 175–181
 - introduction 37
 - migration build 198–199
 - migration execution 203–204
 - migration testing 199–203
 - see also* extract, transform and load (ETL) design
- migration form 54–55
 - definition 54
 - impact of training lag on 130
 - and key business data areas 87
 - and 'one-way street' problem 170
 - and software selection 58
 - and systems integrator selection 60
 - see also* phased migration
- migration model gaps 116–120, 121–122, 160
- migration models 83, 116–120
 - instantiating 119, 161–162
 - logical 161
 - physical 162
- migration plans, initial 55–56
- migration readiness 66, 74, 147
- migration strategy and governance (MSG)
 - introduction 40
 - see also* data migration strategy; governance
- mock load testing 201–202
 - see also* trial migrations
- models *see* data models; migration models
- MSG *see* migration strategy and governance (MSG)
- navigation, definition 159
- navigation rules 166
- network considerations 184
- non-functional requirements of design 181–184
- off-the-shelf software 43–44, 118, 182
- 'one-way street' problem 169–170, 189, 196
- open source software 41–42, 57–58, 60
- open transactions, handling 138, 178, 192–193
- orchestration 177
- ownership 91–92
 - see also* data owners
- parallel migration 54
 - and 'one-way street' problem 170
- parallel processing 58
- parsing data 168–169
- partial fallback 187
- PDMv2 methodology
 - accreditation 61
 - overview Fig. 3.1, 35–40
 - solutions to common problems 44–45
- people management 28–29
- phased migration 54, 87
 - due to training lag 130
 - and 'one-way street' problem 170
 - switching to 188
 - and transitional interfaces 192
- physical decommissioning design 197
- physical migration models 162
- plane flight metaphor 13–14
- planning, project 67
 - initial migration plan 55–56
 - updating after end-to-end design 180–181
- plans
 - cutover 194, 195, 204
 - initial migration 55–56
 - see also* system retirement plans (SRP)
- policies
 - in data migration strategy 50–53
 - definition 51
 - framework purchasing agreements 62
 - risk aversion 51–52, 57, 147
 - software 58
- post-implementation mop-up in failing projects 215–216
- priorities, data quality rules 142, 154
- procurement processes 21, 89, 97
- profiling *see* data profiling
- programme experts, identifying 98–99
- programme initiation document (PID) 48
- project close down 206
- project decomposition 54, 86–88
 - in failing projects 216
- project governance 49–50
- project office functions 64–72
- qualitative assessments of data 154–155
- quality *see* data quality; data quality rules (DQR)
- quality versus time versus budget Fig. 4.1, 51–52, 57, 147
- quantitative assessments of data 154–155
- quest metaphor 13–14
- readiness
 - business 74
 - migration 66, 74, 147
 - see also* business readiness teams
- reality gaps and checks 115, 119–120, 159–160
 - in failing projects 216
 - in master data management 121
- recursion, uncontrolled
 - PDMv2 solution 45
 - problem with 9
- regulatory experts
 - on DQR board 149
 - identifying 101–102
- regulatory requirements, policies 53
- release management 71
 - in failing projects 214
- reporting
 - on data quality rules 73, 146–147
 - on legacy decommissioning 74
 - during migration 191
 - requirements 177
- rescuing projects 211–218
- resources, in system retirement plans 136–137
- responsibility gap 10–15
- retention, data 131–132, 179
- risk aversion policies 51–52
 - effect on software selection 57
 - revisiting 147
- risk management 67–68
- rollback *see* fallback
- rules
 - data mapping 145, 165–167, 171–172
 - data transitional 143, 183–184, 235
 - see also* data quality rules (DQR); Golden Rules
- run times 182–183

- scope
 - data migration project 48–49
 - management of 64–65
 - programme 48
- semantic issues 11–12
- sequencing 183–184
 - rules 167, 172
- SI *see* systems integrators (SI)
- sign-off
 - check points 187
 - decommissioning 130, 138, 203
 - system retirement plans 74, 129–130, 194, 203
- sizing, data 181–182
- SMART tasks 67
 - see also* Super SMART Tasks
- social networking 43, 59
- software
 - budget for 57–58
 - data profiling tools 41–42, 125–126
 - data quality tools 42, 125, 146, 161–162
 - in end-to-end design 175–176, 178
 - for migration build 198
 - migration controllers 42–43
 - off-the-shelf 43–44, 118, 182
 - open source 41–42, 57–58, 60
 - researching the market 59–60
 - selection 56–60
 - support for data quality rules 156–157
 - support for gap analysis and mapping 173–174
 - support for landscape analysis 125
 - synchronising 42, 54, 58, 184, 188–189
 - and systems integrator selection 60
- specialist skills, lack of
 - PDMv2* solution 44
 - problem with 9
- spreadsheets, interlinked 105
- SRP *see* system retirement plans (SRP)
- stabilisation of failing projects 212–214
- staging databases 162, 173
- stakeholders *see* key data stakeholders (KDSH)
- strategy *see* data migration strategy
- studio editions 41–42, 57–58, 60
- Super SMART Tasks 28–29, 32–34
 - data migration strategy as 47
 - data quality rules as 38
 - landscape analysis as 108–109
 - system retirement plans as 127, 138–139
- supplier/client responsibilities *see* demilitarised zone (DMZ)
- suppliers *see* systems integrators (SI)
- switch to phased delivery fallback 188
- synchronising software 42, 54, 58, 184, 188–189
- syncretistic option 237
- system retirement plans (SRP)
 - contents 130–138
 - definition 127–128
 - and demilitarised zone 139
 - and end-to-end design 179–180
 - in failing projects 215
 - implementing 128–130
 - introduction 39–40
 - signing off 74, 129–130, 194, 203
 - as Super SMART Tasks 127, 138–139
- systems administrators 96
- systems integrators (SI)
 - costs of 60
 - selection 60–63
 - tasks outside scope of 36, 37
- target data models 83, 117
- target model gaps 114, 145, 158, 160–161
- team building 28–29
- teamwork, on data quality rules 147–148
- technical system experts
 - on DQR board 148–149
 - identifying 96–97
- technical work stream Fig. 3.1
- techno-centricity
 - PDMv2* solution 44
 - problem with 9
- technology
 - case for using off-the shelf 43–44
 - overview 41–43, 178
 - in *PDMv2* 45
 - see also* software
- telephone conferencing 157
- testing
 - data migration 199–203
 - exception handling 202–203
 - mock load 201–202
 - responsibility for 176–177
 - in system retirement plans 131, 179
 - user acceptance 130, 131, 200
- timing
 - of landscape analysis 109
 - of migration 134–135, 178
 - of mock load testing 201
 - run times 182–183
 - windows of opportunity 134–135, 179, 182, 183
 - see also* quality versus time versus budget
- tracking *see* data lineage
- training 130, 179
- training lag 130, 179
- transactions, handling in-flight 138, 178, 192–193
- transformation
 - and demilitarised zone 176
 - design 190–191
 - see also* extract, transform and load (ETL) design
- transformation rules 145, 167, 172
- transitional business processes 65–66, 124
 - closing down 206
 - design of 192–193
 - in system retirement plans 138, 194
- transitional data stores 160
- transitional interface design 192
- transitional rules 143, 183–184, 235
- trial migrations 55–56
 - see also* mock load testing
- type, definition 120
- UAT *see* user acceptance testing (UAT)
- uncontrolled recursion *see* recursion, uncontrolled
- underestimating
 - PDMv2* solution 44–45
 - problem with 9
- unit testing 199
- units of migration
 - definition 27
 - in system retirement plans 137–138
 - use in reporting 74, 146–147
- user acceptance testing (UAT) 130, 131, 200
- validation rules 145, 166, 172, 190
- widgets 59, 60
- windows of opportunity 134–135, 179, 182, 183
- workflow 43

PRACTICAL DATA MIGRATION

Second edition

Johny Morris

Nearly 40% of data migration projects are over time, over budget or fail entirely. Studies have shown that using a proven methodology greatly increases the chances of on time, on budget and zero defect migrations. Enhance your career prospects by reading this book and making a success of your migration projects.

The author uses a series of steps developed in real-life situations that will deliver a populated, clean, working system backed by the user population.

The first edition of this book provided the first non-proprietary methodology for data migration and quickly became the primary text on the subject. This new edition is updated to take account of changes in technology and the maturing of the market for data migration services.

- **New updated edition of bestselling text**
- **Get the dirty old data out of your legacy systems and transform it into clean new data**
- **Enriched with anecdotes, hints, golden rules and definitions**

ABOUT THE AUTHOR

Johny Morris has over 25 years' experience in IT working as a programmer, analyst, project manager and system designer. He has worked as a data migration consultant for some of the biggest names in IT consultancy (CSC, Logica CMG and others) and has been involved in data migrations large and small at blue chip clients like Barclays Bank, BT, Network Rail and Jaguar Land Rover.

You might also be interested in:

PRINCIPLES OF DATA MANAGEMENT:

Facilitating Information Sharing

Keith Gordon

BUSINESS ANALYSIS Second Edition

Debra Paul, Donald Yeates and James Cadle (Editors)

For any practitioner faced with the challenge of delivering a successful data migration, this book is an absolute necessity.

**Dylan Jones, Founder
Data Migration Pro/Data Quality Pro**

Management;
Information technology



ISBN 9781906124847

