

Reading Sample

This sample chapter describes how to plan and organize your data migration project with discussion of the different types of migration projects and the project phases necessary for each. It also gives in-depth coverage of the data migration work packages and the tools you can use to establish which business objects to migration and to determine the duration of your migration project.



"Plan and Organize Your Data Migration Projects"



Contents



Index



The Authors

Frank Densborn, Frank Finkbohner, Johann Gradl, Michael Roth, and Michael Willinger

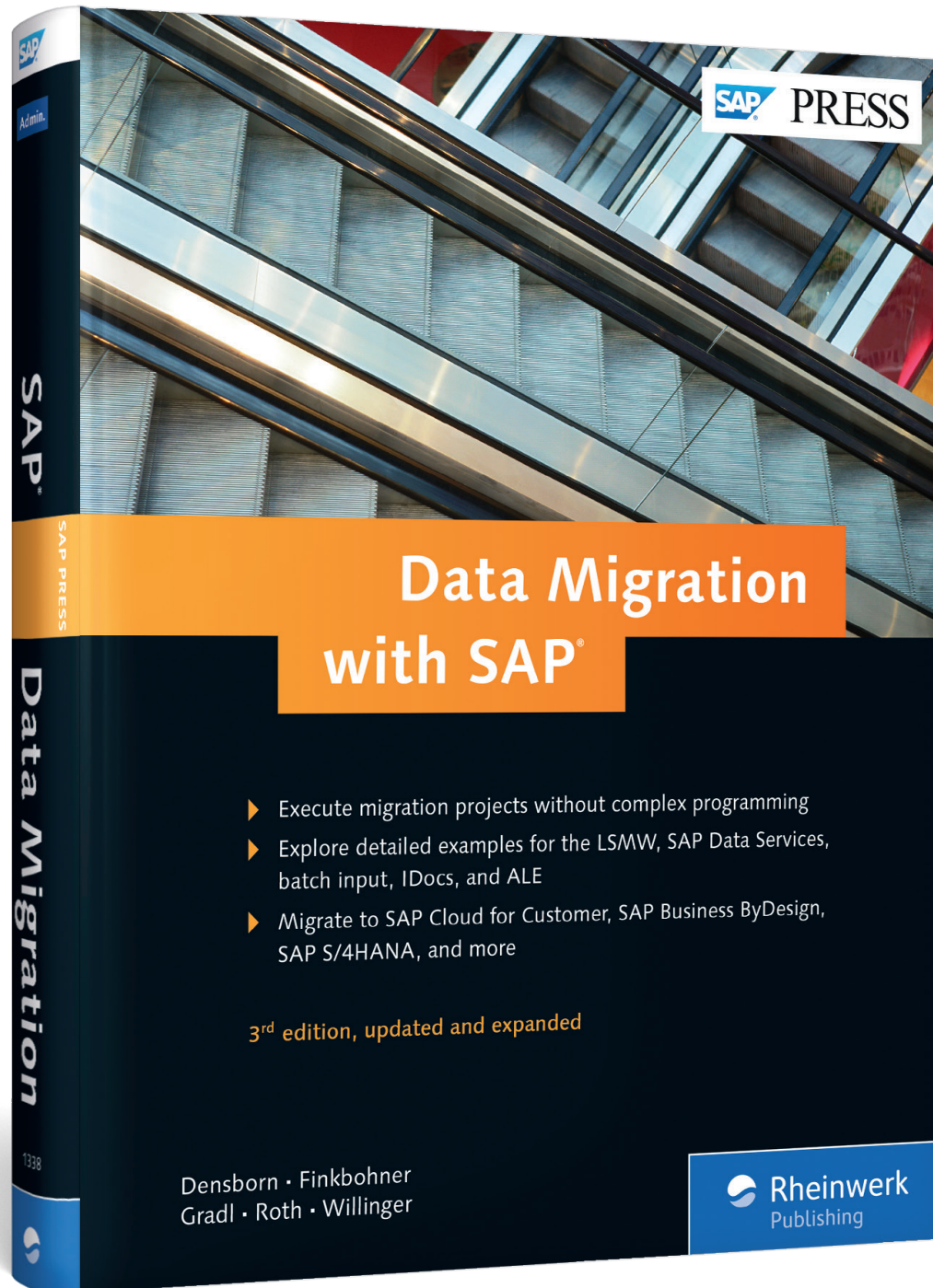
Data Migration with SAP

563 Pages, 2016, \$79.95

ISBN 978-1-4932-1338-2



www.sap-press.com/4019



Structured and well-organized data migration projects are more successful than unstructured projects. Your level of success depends on defining and planning the necessary work packages and recognizing the required effort and resources from the beginning.

3 Plan and Organize Your Data Migration Projects

In Chapter 1, we introduced the business basics of data migration projects. This chapter explains how to plan and organize your data migration projects. In the first section, we briefly define the different types of data migration projects. Subsequently, we describe the different phases of a data migration project and how they are integrated into the phases of an implementation project. In the third section, you'll find out which tasks will appear in each data migration phase and how to schedule these tasks into work packages. Finally, we give you some advice how to determine data migration objects and calculate the required effort.

3.1 Types of Data Migration Projects

All data migration projects deal with the question of how to get data from one system to another system, but we can differentiate the different types of data migration projects based on a few points, as described in the following subsections.

3.1.1 Initial Load (Green Field)

In *initial load* projects, also known as *green field* projects, you migrate the data into a newly implemented SAP system. The data for such projects comes from different source systems, which is normal for a data migration project.

3.1.2 Upgrading an Existing System (Brown Field)

By upgrading existing systems, called *brown field* projects, an already productive SAP system will be extended with new data for existing or new business processes. There can also be different source systems to load the data from.

3.1.3 Phased Roll-Out

Modest risk,
long period

A *phased roll-out* project is a combination of an initial load and upgrading an existing system project. This kind of data migration project is often used if an IT landscape is restructured. This is typically for mergers and acquisitions, system centralizing, or roll-out of an SAP system to local or international subsidiaries. Such projects are projected for longer periods, often years. Upgrading existing SAP systems demands well-organized project management to plan the duration and quality of the data migration. A phased roll-out allows for cost-cutting and an increase in quality that build on one another over the years. However, you also need to build additional interfaces to connect your still existing systems to the new SAP system landscape.

3.1.4 Big Bang

High risk,
short period

A *big bang* implementation is a phased roll-out in one single phase. This will normally be used if the numbers of subsidiaries is small or they are using the same business processes with no or minimal changes to the parent company's processes. The risk to fail of such big bang implementations is higher than with a phased roll-out. You therefore need to plan and test the go-live of the SAP system more accurately. With the big bang method, the whole company is changing its system landscape and processes overnight. If the data quality or something else fails, the whole company's production system will fail.

3.1.5 System Optimization

Mergers and
acquisitions

If processes or organizational structures of a SAP production system must be moved completely to another SAP system, you'll have to use special legal conform techniques and services. These services and techniques

are called *system optimizations* or *system landscape optimizations*. The most common use cases for a system optimization is selling certain subsidiaries or parts of your company to another company or acquiring other companies. In the first case, you want to carve out processes and organizational structures and their data off your SAP production system. In the second case, you want to include additional processes and organizational structures into your SAP production system. Because both the source and target systems are SAP systems, the data migration can be done on a database table level most of the time. This method is very fast and allows historical data to be transferred. However, these kinds of system optimizations are often subject to accountability based on financial audits and the countries' generally accepted accounting principles. You should therefore make use of services offered by certificated special service providers such as SAP's System Landscape Optimization (SLO).

3.1.6 Cloud Migration

Cloud migration is the process of migrating all or certain parts of your system into a public cloud provided as *software-as-a-service (SaaS)* and not private cloud or hosted cloud models. A data migration into such an SAP cloud system such as SAP Cloud for Customer, SAP SuccessFactors, or SAP S/4HANA cloud edition differs from a data migration to an on-premise SAP Business Suite system. For cloud solutions, SAP offers tools and prebuilt content to migrate your data. You can generate Microsoft Excel/XML files with these tools, fill them with your data, and load them into your cloud system.

For some cloud solutions, it's possible to directly connect to your SAP source system using these tools to extract your data. You don't have access to the cloud backend system. Well-known tools and interfaces such as the Legacy System Migration Workbench (LSMW) or batch input can't be used anymore. Due to the limited customizing effort and prebuilt content, the effort to implement the data load procedures will be lower. You have to provide the files in the correct format and maintain the value mapping.

Cloud migration
differs from
on-premise

»

Cloud Systems: Further Reading

You'll find more about SAP SuccessFactors in Chapter 8, SAP Business ByDesign and SAP Cloud for Customer in Chapter 9, and SAP S/4HANA, cloud edition in Chapter 12.

Note that the data migration phases of these different kinds of data migration projects will be generally the same, as discussed next.

3.2 Phases of Data Migration Projects

ASAP To plan and organize SAP implementation projects, you can use different implementation methods. One of the most commonly used implementation methods is *ASAP*. *ASAP* is used to implement SAP products faster, based on a standardized method. Data migration project phases are different from the general implementation project phases. In this section, we'll explain the different data migration project phases and how these phases will be included into the standard project phases of *ASAP*.

The *ASAP* implementation method consists of five phases:

- 1. Project preparation
- 2. Business blueprint
- 3. Realization (including tests)
- 4. Final preparation
- 5. Go-live and support

Seven data migration phases As described in Chapter 1, the best way to include the data migration tasks in an implementation project is to set up the data migration as a subproject. There are seven data migration phases, which differ from the standard *ASAP* implementation phases:

- 1. Data analysis
- 2. Mapping
- 3. Implementation
- 4. Data migration tests

- 5. Data validation
- 6. Data cleansing
- 7. Productive load and support

Figure 3.1 illustrates the different data migration phases.

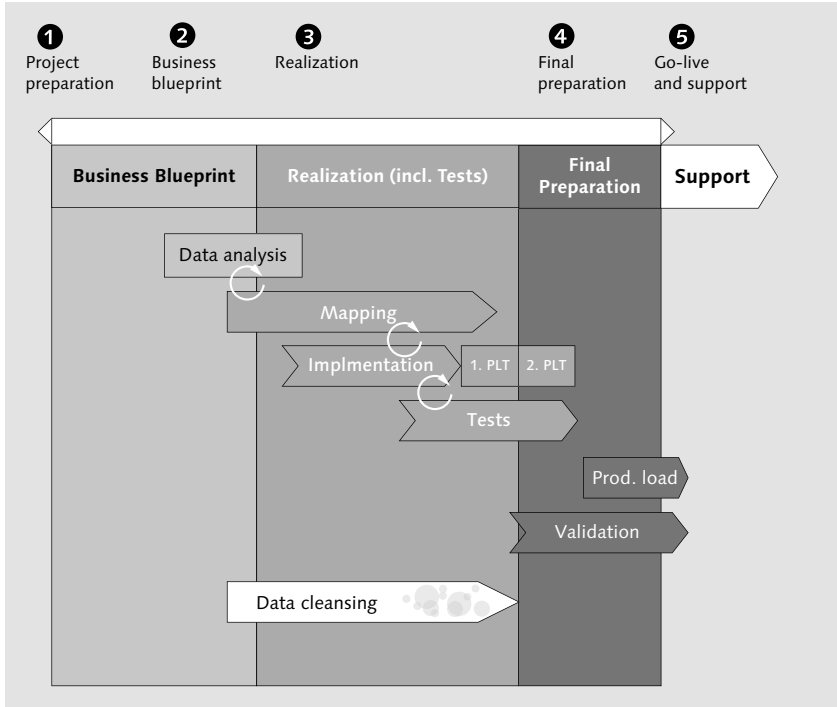


Figure 3.1 Migration Phases within the ASAP Implementation Method

In contrast to classical *ASAP* phases, data migration phases aren't strictly sequential. They are repetitive, and can overlap with *ASAP* and other data migration phases. Results of subsequent phases influence previous phases. This is necessary, as it allows you to react to errors and flaws in a flexible way. You'll often detect such flaws in the business blueprint when you load real data into the customized test systems. The business blueprint, data analysis, and mapping must be adopted to correct such flaws. The next sections go into more detail of the different data migration phases.

Phases must overlap

3.2.1 Data Analysis

The *data analysis* phase is normally started in parallel with the *business blueprint (BBP)* phase; however, you don't have to start the data analysis phase with the BBP. You can start with analyzing the data at the halfway point of the BBP. The advantage of doing so is that most business processes to be implemented are known and designed at this stage, so it's easier to identify the data migration objects and the leading source systems from which to extract the migration objects.

[>>]

Business Object vs. Migration Object

In this chapter, we're talking about *business objects* and *migration objects*, as described here:

- ▶ A business object represents a single object to be used to model a business process. It's a semantic entity to represent the smallest logical unit to take part in a business scenario, for example, material, customer, purchase order, invoice, and so on.
- ▶ A migration object represents an entity of a business object that is used to migrate the data of a business object, for example, basic material data from system A, material sales data from system B, and so on.

The relationship between business object and migration object is 1:N. There can be many migration objects for one business object but only one business object per migration object.

3.2.2 Mapping

After identifying the data migration objects, you can start with the *mapping* phase. The mapping phase overlaps with the data analyzing phase. You should start the mapping as soon as possible—don't wait until the analysis is completely finished. A good starting point is when the first master data objects are identified and the BBP for the business processes to maintain these objects are mostly finished.

Conversion rules

The BBP documents include rules and notes to build conversion rules. The mapping phase can persist into the *data migration test* phase. The results of the first data migration test often have a deep impact on the mapping and the conversion rules.

3.2.3 Implementation

You can also start the *implementation* phase and first functional tests early, during the *realization* phase of the project. The best starting point for the implementation phase is when the bigger part of your conversion rules have been defined. You then can start extracting the data and testing your conversion rules by loading that data into your target test system.

3.2.4 Data Migration Tests

Tests are the most important thing in each data migration. You must start with the first functional tests as soon as possible. Testing is essential. The more complex a data migration is and the more target systems, source systems, data migration objects, and conversion rules that are involved, the more crucial tests will be.

Tests can't be replaced by something else, besides more tests! You have to test the whole data migration process and the go-live itself by doing a dress rehearsal at least once before your go-live. Experience has shown that one test isn't sufficient. Having at least two productive data load tests into an SAP system, which is technically the same as the target system, under go-live conditions, will have a positive impact on your data quality and your ability to adhere to the timetable.

Tests are the alpha and omega

Tests Reduce Costs

The higher the number of tests, the higher the data quality, the less the rework that must be done in the target systems shortly before or after go-live, and the less the total costs will be.

[+]

3.2.5 Data Validation

Validating your migrated data is indispensable for the quality of your business process in your target system. Master data and transaction data of superior quality is the basis for perfect working business processes. In some industries, such as pharmacy, validations are mandatory and must be precisely documented. For example, without appropriate validation

of all migrated data in your target system and a well-documented data migration process, you'll never pass the FDA audit. Without a Certificate of FDA Registration, you're not allowed to sell pharmaceutical products or medical devices on the US market. More about U.S. Food and Drug Administration and Regulations and the Assessments of SAP Products can be found here: <http://scn.sap.com/docs/DOC-7854>. Don't disregard the *data validation* phase.

You should start defining your validation rules when you start defining your conversion rules. Rapid Data Migration with SAP Data Services includes predefined validation rules, which help you validate your data. An automatic validation enables you to cleanse your source data by examining the results of the validation and determining erroneous data records and conversion rules.

More information can be found in Chapter 8, Section 8.3.3 and Section 8.3.6. You should plan your data validation phase during the integration tests and after the productive load phase at an early stage.

3.2.6 Data Cleansing

Data cleansing is one of the most underestimated tasks within a data migration project. If possible, data cleansing should be done in the source system. Under certain circumstances, it may also be possible to do this within your data migration tool. For example, SAP Data Services has a number of built-in data quality procedures and transforms to do deduplication, address cleansing, gender code detection by name, and other useful procedures. Methodic errors in source data records can be converted into clean transfer data by using tailor-made conversion rules.

"Garbage in, garbage out"

You should also use the results of automatic validation to cleanse your data in the source system. The cleaner the data is in the source system, the cleaner your data will be in the target system and the better your processes will run.

Start your data cleansing tasks as early as possible. Ideally, there are already defined data cleansing processes running in the source system to permanently check and cleanse the data records. You should include these processes in your data migration project and should work closely

with the responsible teams. Existing cleansing rules must be identified and compared to the defined conversion rules. The data quality results of the first data migration tests are important for further data cleansing activities.

The data cleansing phase must be completed with the last productive load test to assure that the results of the latest tests are taken into consideration for the go-live one by one.

3.2.7 Productive Load and Support

The *productive load* is the last and most critical phase within your data migration project. As mentioned before, the productive load must be planned and tested accurately. In addition to your normal migration tests, you should plan at least one, and we recommend two, dress rehearsals to test the complete migration process. The more you test, the better your results will be. There are a lot of things which must be taken into consideration that you normally don't think about. Here are some examples:

- ▶ If you plan to work overnight or on the weekend, you have to align with your workers' council, the related trade union, and your local authorities to follow the rules. For example, are warehousemen allowed to take stock at these special times?
- ▶ If you'll work cross-country or intercontinentally, you have to check the local bank holidays, vacation periods, and other local rules up front. For example, you're doing a data migration in a 24/7 mode, and you need your data at 10 am CET. The database you're extracting the data from is located in a San Francisco data center. You then have to verify with your data center that someone is working at 1 am local time to extract your data.
- ▶ Many more items must be taken into consideration, such as the time your productive load will take, during which your IT system might be unavailable, or only partially available. Therefore, you have to plan your production and supply chains carefully, so you don't have hundreds of trucks standing in line at your loading zone waiting to load or unload their supplies.

Follow local labor laws and regulations

Support period ▶ After the productive load, you also have to plan a support period during which your data migration staff is available after the go-live. This is to fix errors, bugs, and flaws due to false or not completely implemented conversion or extraction rules. Sometimes, restrictive rules may prevent data records from being loaded, which *must* be loaded. Don't let the data migration staff go too early after the productive load. You might need them again.

3.3 Work Packages in Data Migration Projects

In this section, we'll describe the work packages within a data migration project. We'll also illustrate the sequence and dependencies of work packages.

In the data migration projects, you ask the same questions again and again:

- ▶ What are the necessary tasks?
- ▶ When will these tasks performed?
- ▶ Which phases are involved?
- ▶ What are the tasks' sequences?
- ▶ Who can perform these tasks?
- ▶ How much is it?

When implementing a new software, the project tasks to perform are always the same and can therefore be included in an implementation method such as ASAP. Regarding data migration projects, you'll notice recurrent tasks that follow a certain structure and can be organized and grouped into work packages.

SEAMAP We refer to this method of organizing and grouping work packages as *SEAMAP*. SEAMAP is a German acronym for a phrase that can be approximately translated to *structure how to effectively group migration projects into work packages*. SEAMAP has been modeled over the years in customer projects and used to plan and estimate data migration projects. It's a template for planning and organizing that can be individually adopted to your project.

However, as a basic rule, each data migration project can be grouped into around 20 work packages. Work package 14 (define and create test data load procedures) isn't a classical data migration work package. It's rarely used in long-lasting data migration projects to provide test data for business process tests by using data migration procedures, which is why we talk about 19 core packages plus 1 additional. You can read more about this in the section on work package 9 in Section 3.3.4.

An overview of all 20 work packages and their relationships is illustrated in Figure 3.2. Due to the size of that overview picture, you can also download it from the book's web page. In the following subsections we have grouped work packages by function (corresponding to the phases of a data migration project), and as such, they will not appear in strict numerical order.

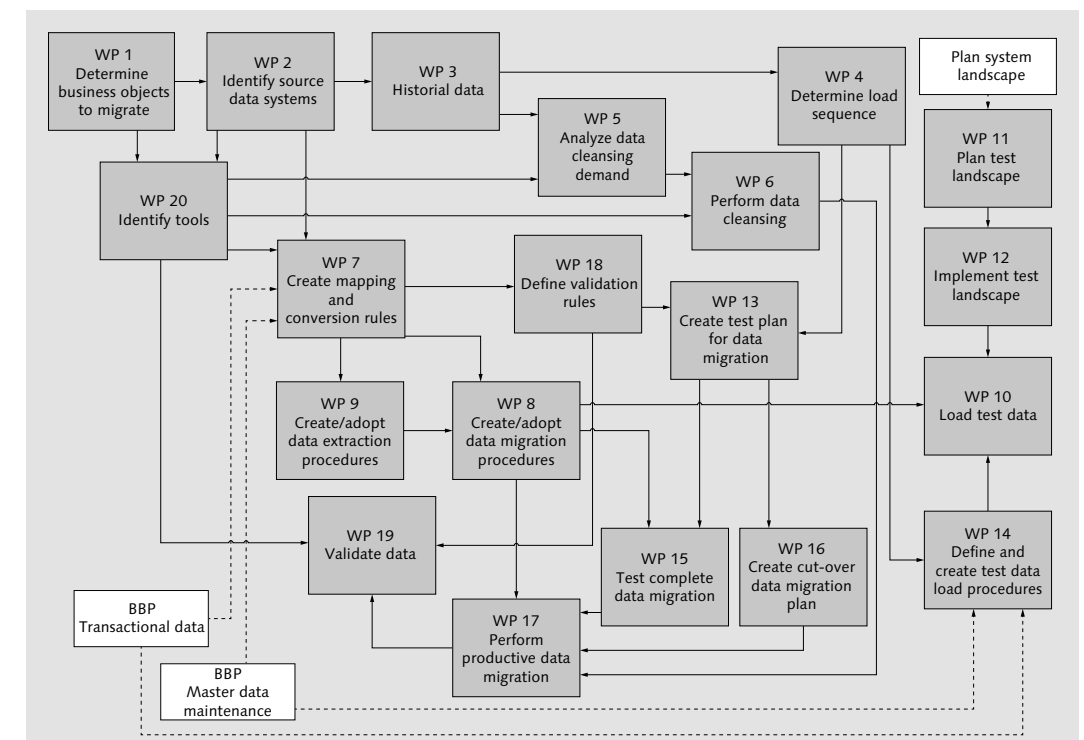


Figure 3.2 Overview of All 20 Data Migration Projects' Work Packages

3.3.1 Data Analysis

In this section, we describe the four data analysis packages as well as work package 20 for identifying tools, which happens during the data analysis phase, and which is dependent on the results of the data analysis packages. Figure 3.3 illustrates an overview of the data analysis work packages (1-4).

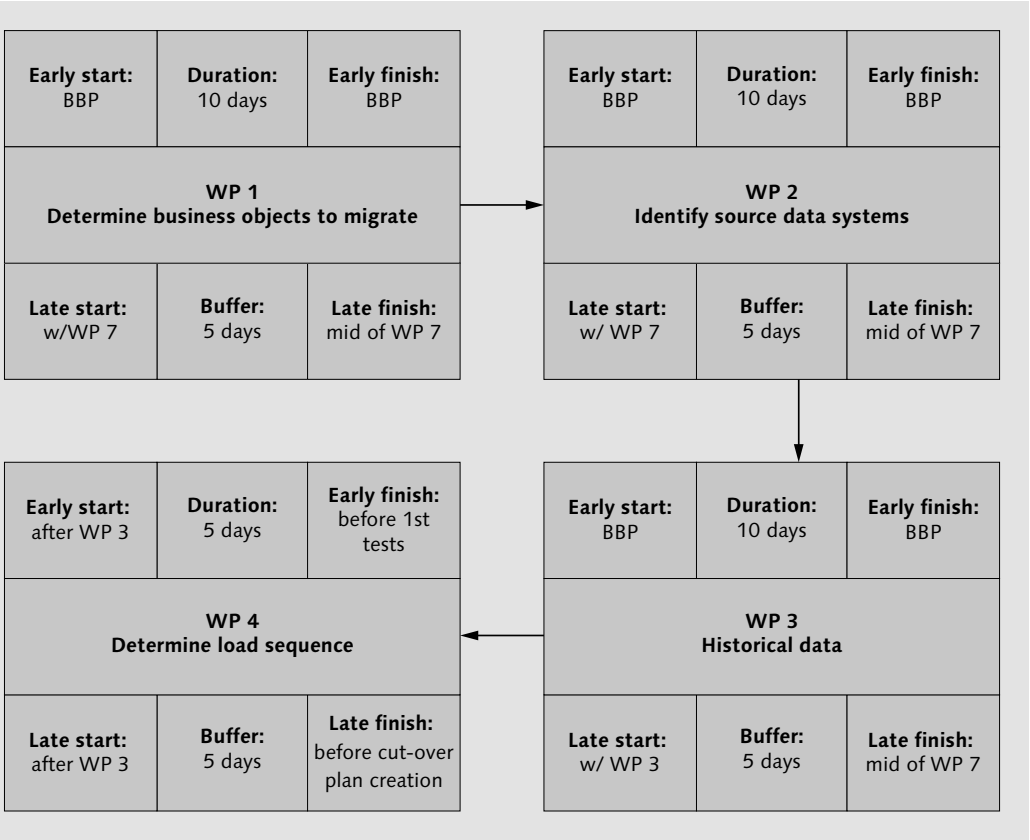


Figure 3.3 Work Packages of the Data Analysis Phase

Work Package 1: Determine Business Objects to Migrate

Identify your business objects to migrate

The first work package of a data migration project is to determine the required business objects to migrate. As describe earlier, we distinguish between business objects and migration objects.

Customer Master Data

[Ex]

From a business process perspective, the customer master is one single business object. From a data migration perspective, this business object can persists of multiple migration objects if it's technically not possible to load its data as one single object.

The earliest possible starting point of this work package is to start at the same time as the BBP. However, it's often better to start by determining the business objects at the second half of the BBP. The blueprint of the business processes must be completed to a certain extent before the relevant business objects will become clear. It makes sense to maintain a list of migration objects in parallel to the BBP with less effort. This will also simplify the following steps in your migration project. The migration objects can be identified and described based on this list. The basic information to collect in this list is described in Section 3.4.1.

The determination of migration objects is normally not finished at the end of the BBP. The data migration often identifies gaps in the BBP during the mapping and realization phases. These gaps may result in new business processes and also new business or migration objects, which no one kept in mind before.

Alternative Units of Measure for Material

[Ex]

An alternative unit of measure (UoM) is a UoM defined in the SAP system in addition to the base UoM of a material. Stocks are posted in the base UoM, for example, whereas the purchase orders might be in the alternative UoM. Consider the following example: In a bigger data migration project, the BBP didn't intend to have alternative UoMs for material master, neither implemented in, nor migrated to the SAP system. During the examination of the international subsidiaries' system data, it turned out that hundreds of material master records in the several subsidiaries' systems had completely different base UoMs stored for the same material. For example, material A was stored with the UoM meters in the subsidiary for Europe, the Middle East, and Africa (EMEA); in inches in the United States; and in rolls in Asia. If there are no alternative UoMs implemented in the system, and the material master of EMEA will be migrated first with a base UoM of meter, the US stock with 400 inches will be stored as 400 meters, and the 20 rolls in Asia will be stored as 20 meters. This would have resulted in

massive problems and costs if it wasn't detected in time by the data migration team. As a consequence, the alternative UoM for material was added into the BBP, and the missing conversion factors from the base UoM units for these hundreds of materials were maintained in a separate Excel spreadsheet. This spreadsheet was then loaded as a separate migration object in addition to the main material master migration object.

Time frame In general, the relevant migration object should be determined within the first half of the mapping phase. It shouldn't take more than 10 days to identify all migration objects.

Work Package 2: Identify Source Data Systems

Identify your source and target systems If the business processes and the business objects to migrate are determined, you can start to identify your source systems. The earliest date to start this work package is along with the start of the BBP, and the work package should be finished in the first half of the mapping phase. You must be clear on the systems to transfer your data from at an early stage. At first sight, this might look easy. However, in large companies with distributed systems and interfaces to exchange their data, it's easier said than done. The zoo of systems, collected over years and decades, must often be structured and arranged first.

Identify the leading system After that, you can start to determine source systems with the most reliable, the most current, and the highest quality data to use for the data migration. Determining the best source data system can be very complex on closer inspection. In such a case, you must be clear on the data stream in your network to identify the leading systems for your migration objects with the best quality to migrate. The leading system is often the system that is at the top node of an interface structure. However, it could also be that some specific data (e.g., address data) is more accurate or current in one of your subsystems.

Don't forget to look at your target systems. Generally, the easiest case is to migrate into one system. If you have to migrate to multiple systems, for example, SAP ERP, SAP Customer Relationship Management (SAP CRM), and SAP Supply Chain Management (SAP SCM) systems, you should be aware of this fact as early as possible.

Data from Various SAP Systems

[Ex]

If you load data into an integrated network of SAP systems, such as *Service Parts Management (SPM)*, you must know where to load the master and transaction data first. The data within such an integrated network is distributed to the other systems via internal interfaces such as the *Core Interface (CIF)* between SAP ERP and a connected SAP SCM or the *CRM Middleware* between SAP ERP and SAP CRM. Customer and vendor master data is migrated to the SAP CRM system as business partner data and then transferred to SAP ERP via the CRM Middleware. The company code data and other financial data must then be added to the transferred customer and vendor data in the SAP ERP system. You have to take care of the order to load the different migration objects, and you have to maintain the availability of the interfaces between the systems to load the data consistently. It often makes sense to first inactivate the internal interfaces, validate the data, and then activate the interfaces again to prevent distributing garbage to the connected systems.

Normally, it doesn't take longer than around 10 days to identify the source systems. In very complex networks, especially mergers and acquisitions, it might be difficult to identify or define one leading system with the best data quality. In such cases, it might take longer than 10 days.

Work Package 3: Historical Data

As described in Chapter 1, Section 1.2.3, it's urgent to clarify whether the historical data needs to be migrated and the way it can be migrated to the target systems. We distinguish between the following historical data definitions:

- ▶ **Out of use/obsolete master data**
This includes, for example, data from old vendors or customers, routings for products that aren't produced anymore, assets that have been sold, organizational data of HR departments that are outsourced, and so on. It's useless to migrate such master data into a target system.
- ▶ **Closed transaction data**
This includes completed and invoiced orders, invoices, account movements, and so on. Although many want to migrate these kinds of data

to the target system, this type of migration is difficult, risky, and often forbidden by local law.

Obsolete master data Obsolete master data should only be migrated to a new system if it's essential. The term *obsolete* indicates that this data is no longer used and often not up to date. This obsolete data uses memory unnecessarily and often causes problems during migration, for example, address data with antiquated or missing postal/ZIP codes. You can find more about this in Chapter 1, Section 1.2.1 and Section 1.2.2.

[Ex]

Migrate Organizational Master Data Using SAP Business Warehouse

In a data migration project to migrate human resource (HR) master data from a legacy system to SAP ERP Human Capital Management (SAP ERP HCM), the company wanted to migrate the complete organizational structure and its assigned HR master records for the preceding 10 years. This type of migration project requires a lot of effort. When asked why this was necessary, the client said it was needed for a quarterly statistical report to the managing board that lists the number of employees per organizational level per quarter. Because the client also used *SAP Business Warehouse (SAP BW)*, we came to the agreement to migrate all existent statistical reports to that SAP BW system and to only migrate the current organizational structure to the new SAP ERP HCM system. All changes in the current organizational structure were then transferred to the SAP BW system after go-live. That way, the client could make sure that the board had access to old and new statistical reports. This procedure simplified the data migration, saved costs, and achieved the same result in the end.

Generally accepted accounting principles Closed transaction data must always be considered critical. If you follow the *generally accepted accounting principles (GAAP)* that may differ from country to country, it's forbidden to migrate closed accounting documents into another system. It's only possible if no changes appear on the migrated documents and if they aren't running through the newly implemented business processes, which only occurs if you migrate from one SAP system to another SAP system directly on the database level. To do this, there are only a few tools available, such as *System Landscape Transformation* or *System Landscape Optimization (SLO)*, provided by a handful of companies, for example, SAP's SLO consulting. We recommend that you always ask your auditor before trying to migrate closed transaction data.

This is also why SAP doesn't offer data migration programs to migrate closed transactions. Before you insist on migrating historic data, you must familiarize yourself with the risks and the alternative solutions.

The earliest starting point and the latest finish are the same as described in the previous work packages. Due to the many constraints, it's possible to finish this work package within 10 days. **Time frame**

Work Package 4: Determine Load Sequence

If all migration objects and source systems are identified, and the decision to migrate historical data is made, you can start to plan the load sequence of your migration objects. Because some migration objects are dependent on other migration objects, the load sequence should be finished as far as possible before the first functional tests. Without material master data and customer master data, you can't migrate sales orders, for example.

The *migration test plan* and the *cut-over plan* are based on the load sequence. The latest finish of this work package should therefore be before you start to create the cut-over data migration plan.

Work Package 20: Identify Tools

After determining the migration objects and source systems, you can start to select and identify the tools for your data migration process. It's very important to choose the correct data migration tools. If you choose the wrong tool at the beginning of your migration project, it may lead to high additional costs for training, maintenance, or licenses. Figure 3.4 illustrates an overview of the work packages and their dependencies. **Identify your tools**

As a basic rule, the following types of tools are used for data migration:

- Data cleansing tools
- Data load tools
- Data validation tools

There are various tools on the market. Some of them, called *extraction, transformation, and loading (ETL)* tools, cover all aspects of a data migration process. **ETL tools**

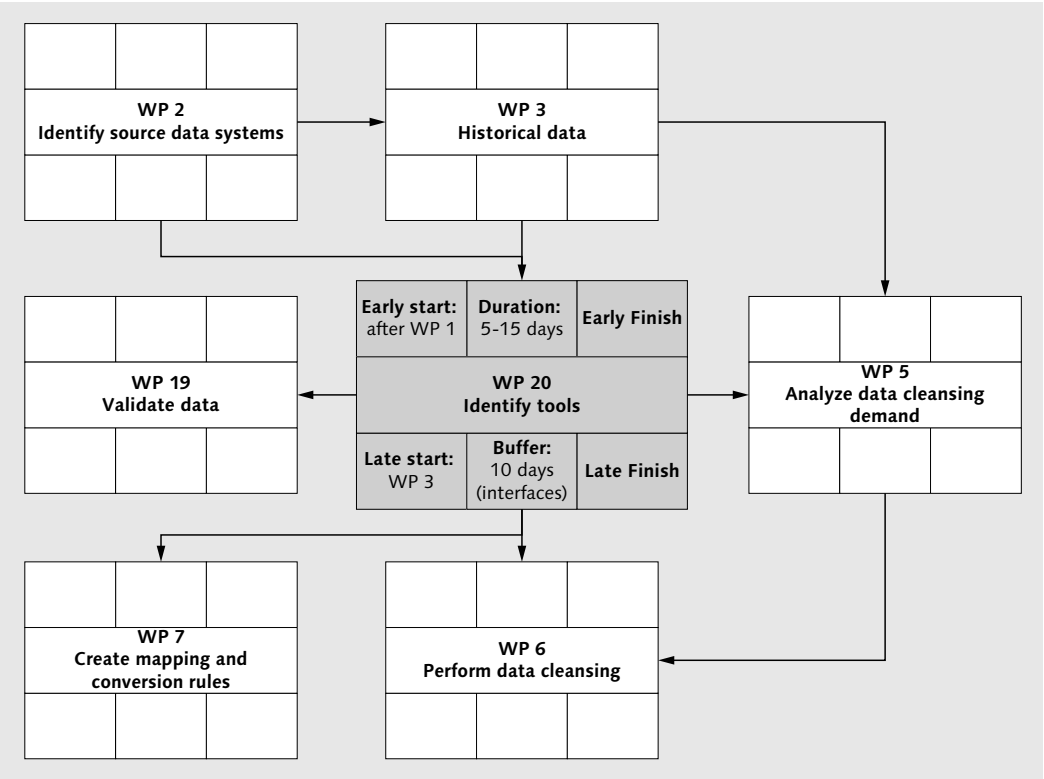


Figure 3.4 Work Packages Related to Work Package 20

Some of the standard built-in SAP products, for example, the LSMW (Chapter 6) or the Data Transfer Workbench (Chapter 5, Section 5.2), only cover a small part of the data migration requirements. Other tools, such as the ETL tool SAP Data Services (Chapter 8) cover a broader spectrum of data migration requirements. When choosing the tools, it's important to check the compatibility with your source and target systems and to verify the availability of consulting experts who know how to use these tools.

[Ex]

Choosing the Wrong Tool

In a public sector's SAP implementation project (planned to take more than five years, with hundreds of SAP consultants on board), the purchase process of the data migration tool called for offers across Europe. After allowing a very long deadline for offers, a small number of tools were shortlisted. Finally, the

company decided on a specific tool and paid a lot of money for the tool's licenses. The company also staffed additional consultants to train some internal staff. However, this tool wasn't widespread on the market. The company couldn't find any data migration consultant or specialists who were familiar with this specific tool, and additional funds to train external consultants weren't available. The company finally used this tool only for a small aspect of the data migration process after wasting a lot of money.

You should request enough comparable proposals for different tools. Ask your vendor for reference customers and reference projects. Check the freelance portals and IT market job exchanges for specialist for specific tools that are available on your country's market.

Best practice

3.3.2 Data Cleansing

Data cleansing is a major component of a data migration project. The work packages involved and their dependencies are displayed in Figure 3.5.

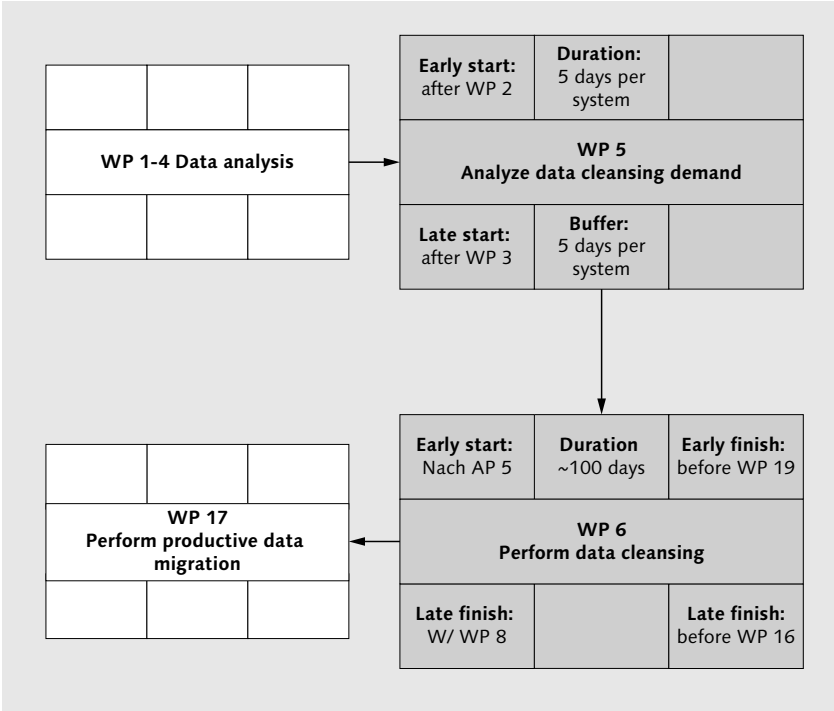


Figure 3.5 Work Packages of the Data Cleansing Phase

Work Package 5: Analyze Data Cleansing Demand

If the bigger part of migration objects and source systems are determined, you can start to analyze the data cleansing demand. The effort should not exceed 5 to 10 days per system.

Work Package 6: Perform Data Cleansing

Performing a data cleansing can last days or months. The data cleansing itself must be finished last before the productive data migration into the target systems. The best point in time to finish is before you start the latest productive data migration test. The cleansed data can then be used for the data migration dress rehearsal.

Archiving If you migrate your data into a productive system already in operation, it might be essential to remove already existent data in your target system. It's not easy to delete data in SAP systems. In such cases, you should familiarize yourself with how SAP data records are archived.

[Ex]

Load Material Master Records into a Productive System

In a phased roll-out project, material master records had to be migrated into a productive system already in operation. A huge number of the existent material master records were loaded in previous data migration projects using conversion rules that didn't match the current data migration project's conversion rules. The easiest way to get rid of these records was to delete them. Material master records used in transaction documents or in stock can't be removed. After intensive investigations, it was decided to use the SAP archiving programs to remove them. Running the archiving programs in test mode allows the client to identify material master records used in transaction documents. By adding a small piece of program code into the related archiving program's *Business Add-In (BAI)* BADI_MM_MATNR/IF_EX_BADI_MM_MATNR~ARCHIVE_MARA, it was possible to store the material numbers not used in any transaction documents and without any relationships to other objects into a customer's Z table by running Program MMREO050N in test mode. In the second step, this Z table was used to archive the material master records stored in this table by calling Program MMREO050N in production mode and deleting these materials from the database tables.

Archiving might also be helpful if you want to remove erroneously loaded data records during the data migration tests.

3.3.3 Mapping

The mapping phase is the best point in time to determine the correct data migration technique for each migration object. The mapping and conversion rules should follow the migration object's data structure provided by the selected data migration technique.

If you have a lot of organizational structures to map and migrate, an intermediate document (IDoc) is your best choice (see Chapter 5) to migrate your data. However, you shouldn't exclude batch input as a data migration technique because some data migration scenarios are better suited to batch input. You can learn more about this in Chapter 11. Figure 3.6 provides an overview of the mapping phase packages and their dependencies.

IDoc versus batch input

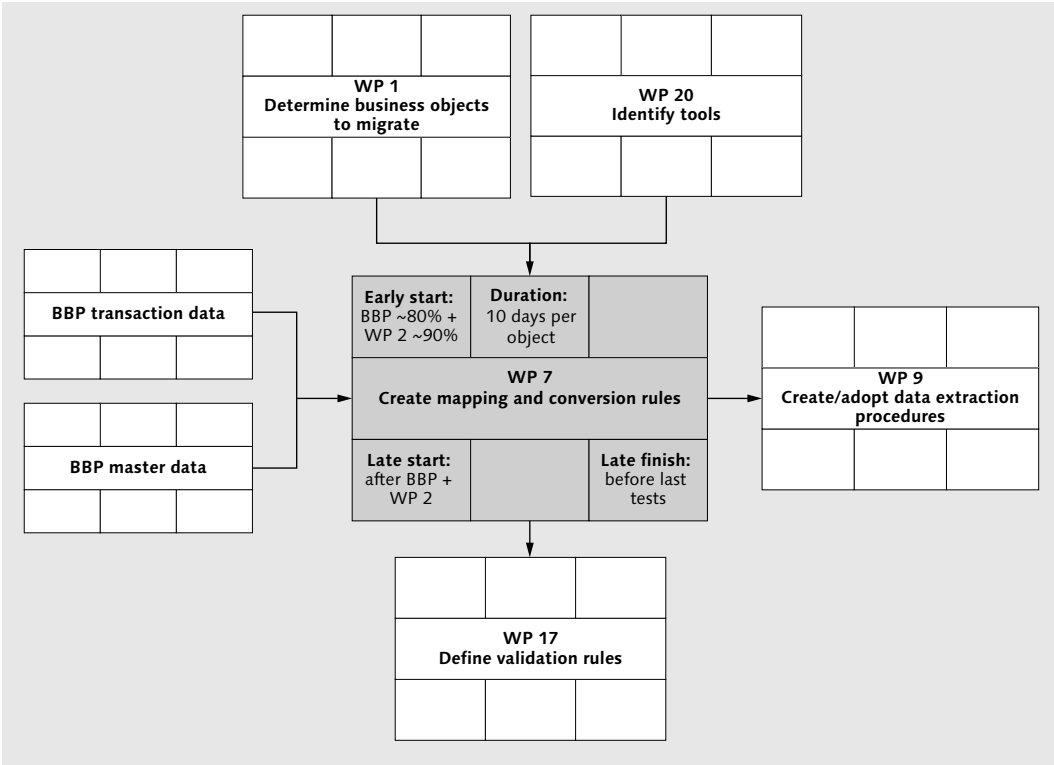


Figure 3.6 Work Packages of the Mapping Phase

[Ex]

Choosing the Best Technique for a Specific Purpose

The best way to map the material master is to use the IDoc type MATMAS06, the BAPI BAPI_MATERIAL_SAVEREPLICA, or the standard direct input Program RMDATIND. Using one of these solutions, it's possible to load multiple organizational levels (company code, plant, sales organization, storage locations, etc.) with one single shot. All three solutions call the function module MATERIAL_MAINTAIN_DARK to post the data. By using the IDoc or BAPI, it's also possible to parallelize the execution into several work processes and to load more than one material at the same time. Using batch input, you have to create, maintain, and test recordings per material type and view, and you can't load multiple organizational level structures (e.g. three plants and two sales organization structures) at the same time. Not using batch input will boost enormously the load time and the cost to implement and test. New custom views and fields are also easy to map with an IDoc or BAPI.

Customer master data and vendor master data can also be migrated smoothly using an IDoc if there are new customer views to integrate, and there are fewer than 5,000 master data records. You can also record a transaction instead of implementing a very complex extension of the DEBMAS or CREMAS IDoc type, especially if this extension can't be reused in interfaces.

Work Package 7: Create Mapping and Conversion Rules

The earliest possible time to create the conversion rules is at the 80% finished point in the BBP. At this time, most conversion rules are known. You must start creating your rules at the latest when all migration objects are determined, and you must be finished before your last tests.

Paper based mapping

If possible, you can use existing mapping templates to map certain migration objects. The so-called paper based mapping can be made using a Microsoft Excel spreadsheet. Discuss the source structure and field mapping together with the person in charge at the customer and the source system's IT staff, and note this down in the migration template. The Rapid Data Migration with SAP Data Services delivers such templates. Check Chapter 8, Section 8.3 (especially Section 8.3.1 and 8.3.4), for more on the topic of mapping templates.

[+]

Use the LSMW's Object Overview

You can use the LSMW to create a simple template for the paper mapping. Just follow the steps described in Chapter 6, Section 6.2.18. Click on the OVERVIEW IN TABLE FORMAT button to place the chosen target structure of your

migration object and existing field mapping in table format. You can easily export this to a local PC file by right-clicking on the application toolbar and choosing SAVE TO PC FILE or by pressing [Shift] + [F8]. You then choose TEXT WITH TABS to store it in the comma-separated values (CSV) file format.

It's essential to bring together both the staff with the technical know-how on the source systems data structures and the staff who best know the business processes. Avoid inviting too many people to such meetings, and try to only talk about one or two migration objects within the same meeting to avoid spinning your wheels and wasting time.

If you're mapping in parallel with the BBP, don't forget that changes in the BBP may have an impact on the mapping and the conversion rules. You therefore should closely and permanently stay in touch with the other subproject teams that are working on the BBP and implementing its processes. This is the only way to discover rule changes at an early stage. You should ask further questions if rules aren't delivered in time. Sometimes it may take weeks or months to define certain conversion rules.

Don't Wait Too Long

In a certain data migration project, it took more than 10 months for the operating department to agree on how to define and assign purchasing groups to certain material master records and to define the related conversion rule. The rules and the customizing were delivered shortly before the last complete data migration test.

Take care to involve all important operating departments that need to deliver conversion rules into the data migration process.

The process of creating mapping and conversion rules is a very communicative process, which can't be outsourced to any offshore party. To avoid mistakes and to promote understanding, the parties involved in this process must speak one business process language and, if possible, the same mother tongue.

Sometimes after the mapping and the first test, some problems appear that result in new data analysis with changed mapping and conversion rules. Conversion rules that seem definitively defined during the data

Involved parties

[Ex]

Review conversion rules

analysis phase might run into serious errors when mapped against real data and tested in the first function tests.

[Ex]

Quality of Address Data

Address data is often maintained without following existing maintenance instructions. Older legacy systems often provide too few or no fields on their entry screen to store address data such as mobile phone numbers, email addresses, web pages, and so on. Data entry employees therefore often store such data in unused fields. Many of these systems allow this because of missing field context checks.

If you then perform the first data extractions, you'll discover that one employee entered the email address into the third address line, the second employee entered it in a comment field, and the third entered it in wherever there was enough space. You'll also discover that no rules were followed regarding how to enter telephone numbers. Some entered them with the international dialing code, others entered it without, some entered with hyphens, and others used slashes between the regional dialing code and the number. You'll discover a ton of incomparable address data that makes it hard to define automated rules to map the different source data fields to their target fields. Sometimes, you'll also discover employees using their own databases and Microsoft Excel sheets to store their contacts address data. In the end, this will lead to new migration objects, mappings, and data cleansing activities.

Validation rules The best time to start thinking about and writing down the validation rules is when you define the conversion rules. Rapid Data Migration with SAP Data Services (presented in Chapter 8, especially Section 8.3.6) contains prebuilt validation rules and procedures. With these rules, it's possible to compare the extracted source data with the migrated data in the target system very easily.

Validation rules always have to be aligned and defined by or together with the client. The client is the only one responsible for validating the data in the target system. Only the client can decide if a data record is migrated correctly or incorrectly.

3.3.4 Implementation

After you've defined the first bunch of conversion rules, you can start to implement the data extraction and data migration procedures and

programs. The final definition of 90% of a migration object's conversion rules is therefore a good reference point. On average, you can calculate around 10 days to implement a migration object. In a cloud environment with predefined data migration procedures, it might be less. To implement simple objects, it may take only 1 day, but to implement very complex objects, it will take longer than 10 days. On average, however, you can calculate the time at 10 days per migration object. You should ascertain that all procedures and programs are finished before the last productive load tests. Figure 3.7 illustrates the two work packages of the implementation phase and their relationship to other work packages.

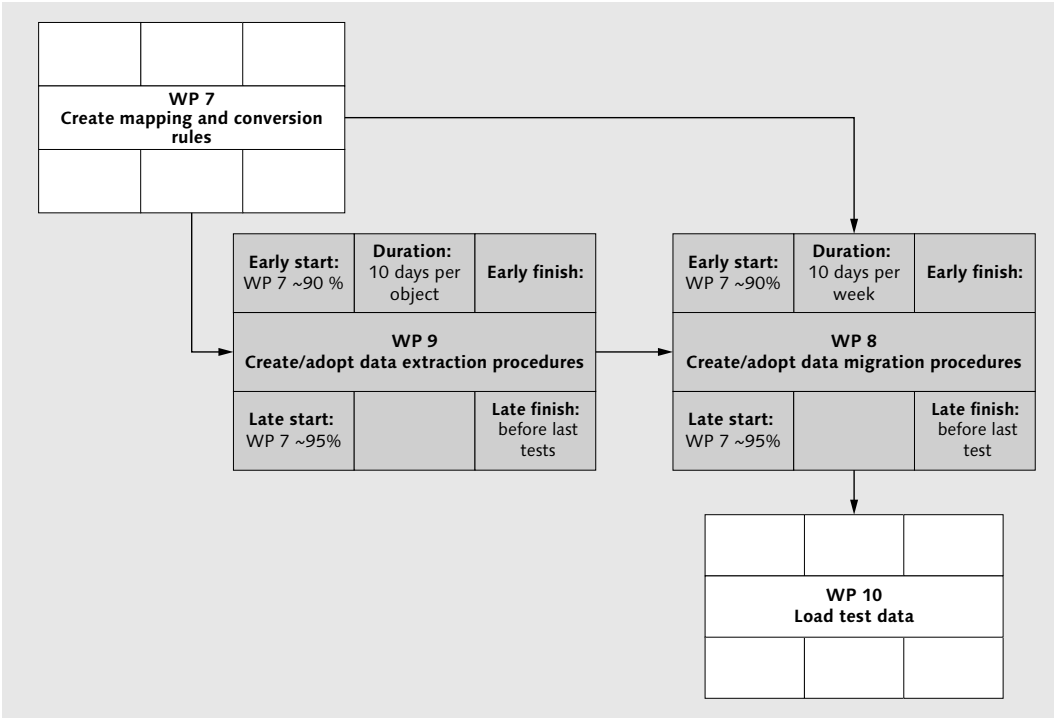


Figure 3.7 Work Packages of the Implementation Phase

In addition to the basic customizing discussed in Chapter 1, Section 1.3.1, it's important to mention that depending on the data migration technique, the business process customizing must be finished to the greatest possible extent. Especially if the data migration procedure is

Customizing impacts on implementation

using a recorded transaction or batch input, this will be a mandatory precondition. If the business process customizing isn't finished, particularly with regard to a finished screen control, it will result in errors during the first function tests. For a recorded transaction, the recordings or related generated programs must be adopted continuously to the changing screen control. IDocs are normally resistant to a continuously changing screen control; they use internal function modules to verify and post the data to the database. But there are also IDoc types that are using batch input techniques to post the data, for example, DEBMAS and CREMAS. If you use such IDoc types, the screen control customizing must also be finished before you can start with the first tests. Compared to recorded transactions, you can start with the implementation using IDocs at any time.

Single-use programs Typically, data extraction and data migration programs are single-use programs. There are only used once to migrate the data and then never used again. Only when these programs are used for *phased roll-outs* are they possibly adopted for a longer period.

Manual migration In certain circumstances, it might be easier and cheaper to manually migrate data:

- ▶ The number of data records to migrate is small and can be handled manually by a small team.
- ▶ The effort and costs to manually create the data in the target system(s) is smaller than the effort and the costs to create and test the data extraction and migrations programs. You must also take the number of tests where this data is required into consideration.

[+]

Use Employee Trainings to Manually Migrate Your Data

If there is a relatively small amount of un-complex data to migrate, you can think about using an employee training to let training participants manually create the data in the target system. Please note that these employees must also be trained in the SAP's data entry processes. If you are running out of time and the effort to implement a migration program is higher than the effort to manually create the data, this may be an option. An assistant's wage rate to manually migrate the data is often much lower than those of a developer to create a program. Don't forget to validate the manually created data!

You must critically assess the need to use *continuous interfaces* to load the data. A continuous interface can be used if the target system isn't the leading system but is instead a subordinated system within an integrated network, and the system is continuously loaded with data via that interface. During an initial load, a huge number of data records will be transferred via this interface into the target system. The huge amount of data records of an initial load will be many times higher than the regular daily load via this interface. This will generally result in an interface capacity overload and delayed schedules. If you plan to initially load a target system via a continuous interface, you'll have to take all resultant risks into consideration that endanger your go-live. Sometimes it's better to use an extra data migration program to initially load your data. You can probably reuse the programs you're already using to load test data into your systems to be used for the business process function tests (see the section on work package 14 in Section 3.3.5).

Migration via continuous interfaces

Work Package 8: Create/Adopt Data Migration Procedures

Depending on the data migration technique and the chosen data migration tool, this work package is the most time-consuming package. Start as early as possible to implement the conversion rules. You should use existing integrated validation and check rules of your data migration tool to improve your data quality and to detect erroneous date records and conversion rules.

You must assume that at the beginning of your data migration project, your source data won't have the quality to run your mapping and conversion rules without any errors. Verify and check your source data during the data conversion and make proactive checks to determine errors. By continuously verifying and adopting the mapping and conversion rules, you'll gradually increase the data quality of your migrated data into the target system.

Running changes

Use Function Modules to Check

If you're familiar with ABAP and you use the LSMW, you can also insert existing internal check function modules into your field mapping. Each SAP system provides a lot of internally usable function modules to verify the correctness of

[+]

postal/ZIP codes, bank accounts, bank keys, and so on. You can use these function modules within your field mapping to determine records with wrong values. Print these records and the related error messages into a log, and forward this to the respective operating department or your data cleansing team. They can verify and correct these incorrect records in the source system.

Work Package 9: Create/Adopt Data Extraction Procedures

Close collaboration To create the extraction programs, it's very important to work closely together with the system administrators of the source data system. These are often the sole persons who know these systems best and can extract the required data. If the source systems are based on nonstandard, self-developed software, it will be better to outsource these work packages to the related system administrators and developers of the source system.

3.3.5 Data Migration Tests

The work packages of the test phase, in addition to the data validation work packages, are the most valuable packages in a data migration project to ensure an acceptable data quality. Figure 3.8 displays the work packages of the test phase and their dependencies.

Coincidental tests and mapping The results of the first function tests will be used to improve the mapping and conversion rules. Sometimes, the analysis of a data migration object can only be closed after the first tests of the mapped source data and their related conversion rules.

[Ex] How to Handle Bad Legacy Data

In an international HR master data migration project, the result of the first function test was that the conversion rules to migrate the employees' private addresses were useless. The source data was partially outdated and in very bad condition; for example, postal codes were missing or in the wrong format, country codes and streets were missing, and telephone numbers were either missing or outdated. The data quality was so bad that it could only be solved with a major effort to manually correct the addresses after the data load. The client decided not to automatically migrate the private addresses.

The employees were told to maintain their private addresses again in the new system via an Employee Self-Service (ESS) after go-live, which severely reduced costs.

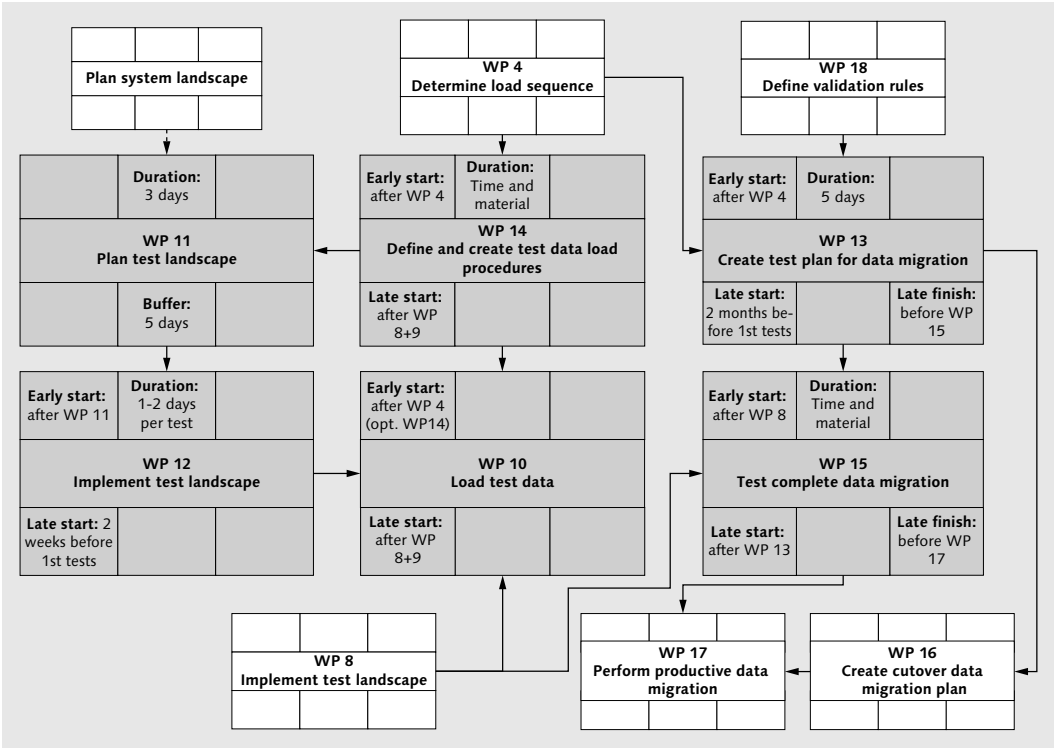


Figure 3.8 Work Packages of the Test Phase

Work Package 10: Load Test Data

Work package 10 includes all activities to provide test data for all functional and implementation tests based on the data migration programs to test the business processes. This work package is distinct from the work package 15 (test complete data migration) in that the data is loaded only to test specific business processes by the BBP teams.

This is also the best time to perform time measurements and to record the duration of the different data migrations steps per migration object:

- 1. How long does it take to extract the data from the source system?
For example, if the server is located in China and you have to access it from the US the time to extract the data takes longer than sitting directly in front of the server.
- 2. How long does it take to transfer the data files to the target application server or a specific network folder?
For example, it may take some minutes to send files greater than 500MB over a slow Internet connection to a far server.
- 3. How long does it take to read the data into the data migration tool?
For example: it makes a difference if a very huge file is read from an application server, from an internal network folder, or from your frontend.
- 4. How long does it take to convert the data with the data migration tool?
For example, converting 1.5 million consumption data records takes more time than converting 1000 customers, depending on the mapping and the validation rules.
- 5. How long does it take to make some prechecks of the converted data?
For example, to check a list of 100 vendors takes less time than checking 5000 material records.
- 6. How long does it take to load/post the data into the target system?
For example, depending on the migration technique, there could be substeps to create batch input sessions or IDocs, for example, before the real load process can be performed.
- 7. How long does it take to validate the data after loading?
For example, it takes more time to validate 5000 customer data records manually than defining validation rules, implementing them in a tool and let the tool validate the data.

The earlier you get an overall picture of the data migration time, the better you're able to plan your tests and the cut-over.

Only recording the load/post time of a migration object is often not sufficient. As a rule of thumb, the sole load/post time is around 15% of the entire object's migration time, including data validation. You should calculate 75% of your migration time to validate the data after loading.

Work Package 11: Plan Test Landscape

It's essential to have a sole data migration system or at least your own system client in the development and quality assurance system to create and test your data migration programs and procedures.

Sole data migration system

Rededicate a Migration System as an Integration Test System

[Ex]

In a big international logistics project, they set up a separate data migration system that was only used to test the data migration. It was connected to the development system. The customizing and workbench requests to get the latest changes, were queued and only imported on demand. It was also used to build up the system for the business process integration tests. After the complete data migration was tested, the system was rededicated to be the new integration test system. The existent integration test system was reinitialized and used for the next migration tests that took place in parallel with the current integration test. During the integration test, the loaded data was tested intensely by the implemented business processes. Detected errors, mapping, and conversion rule changes were immediately forwarded to the data migration team to correct and to be used within the new data migration tests. This procedure ensured that a fresh system was available for the next integration test based on the latest findings and corrections.

You should also make use of *database snapshots* or *system backups* during your data migration tests. After loading bigger migration objects, especially master data objects, you should create a backup of your system or a snapshot of your database. If you then discover massive errors after loading the object, just reset your system by restoring the backup or the snapshot. Database snapshots consume a lot of disk space. But in comparison to a normal backup, it saves a lot of time to restore a snapshot rather than restore a backup file. Taking and restoring a snapshot may take 30 minutes whereas a backup and restore may take hours.

Database snapshot

Work Package 12: Implement Test Landscape

Thinking you can save money by implementing a low-cost test landscape is based on a false economy. You must make sure that your test system environment is very close to the later SAP production system environment. The best practice is to run the tests on an identical system environment. The more identical your test environment is to the productive

environment, the more exact your time measurement and cut-over plan will be. Even a slight difference in the environment can lead to very different results.

[Ex]

Modification Effects

Even one additional application sever in an integrated system network can cause performance problems in the complete system network if there are errors in the operating system or if it isn't set up correctly. And even the size of the internal memory has a much more significant impact on the load time than the number of CPUs. You'll only figure this out, however, if you've planned enough time for testing.

Work Package 13: Create a Test Plan for Data Migration

If you determined the load sequence of your migration object and finished your first data migration programs, it's time to create the test plans for the complete data migration process. To create a test plan, you also need to know the time measurement results of Work Package 10 (load test data). The load sequence and the time measurements are the basis of the test plan. The results of this test plan will seamlessly be integrated into the cut-over migration plan.

Work Package 14: Define and Create Test Data Load Procedures (Optional)

Provide test data

In big implementation projects that are planned over a longer period, it's necessary to provide data for specific business process tests. The technical expertise for extracting and loading programs/procedures to provide the test data can be used as a basis for the real data migration programs. This work package is optional, and you'll be rarely confronted with it in bigger implementation projects.

[Ex]

Interfaces

If the target system is a subordinated system, that is, not the leading system, continuous interfaces are often used to load data into that system. These interfaces might also be used to initially load the data into the target system. Developing such interfaces is often very extensive and time-consuming. To

test the business processes, additional data load programs are developed to load test data into the test systems. If you're using the same load technique for the additional programs as for the continuous interfaces, the findings and results can be incorporated into the development of the continuous interfaces.

If you encounter capacity problems using a continuous interface, it might be advisable to use these test data load programs to perform the initial load into the target systems.

Work Package 15: Test Complete Data Migration

To have high-quality data and a smooth cut-over, it's very important to start with testing your complete data migration process as early as possible. The best time is when all data migration programs (extract and load) and the data migration test plan are finished. It makes the most sense to test the complete data migration in advance of a business process integration test. We described a helpful example in the section on work package 11 in Section 3.3.5.

The special testing situation is called a *productive load test*, which often happens in the course of the *user-acceptance test*.

Productive load tests

As mentioned previously, you should test the complete productive data migration process at minimum twice on a test system environment that is very close to the later SAP production system environment. This is the only way to determine and solve time crunches, technical issues, staffing shortages, or any other kind of problems.

Once is good; twice is better

3.3.6 Productive Load and Support

The productive load and support phase is the project's most critical phase and is affected by fixed deadlines and narrow time frames. At the end of the data migration, it will be clear whether the members of the project team have done their homework. The result will be mercilessly rated by the data validation scoring judges. You can see an overview of the related work packages in Figure 3.9.

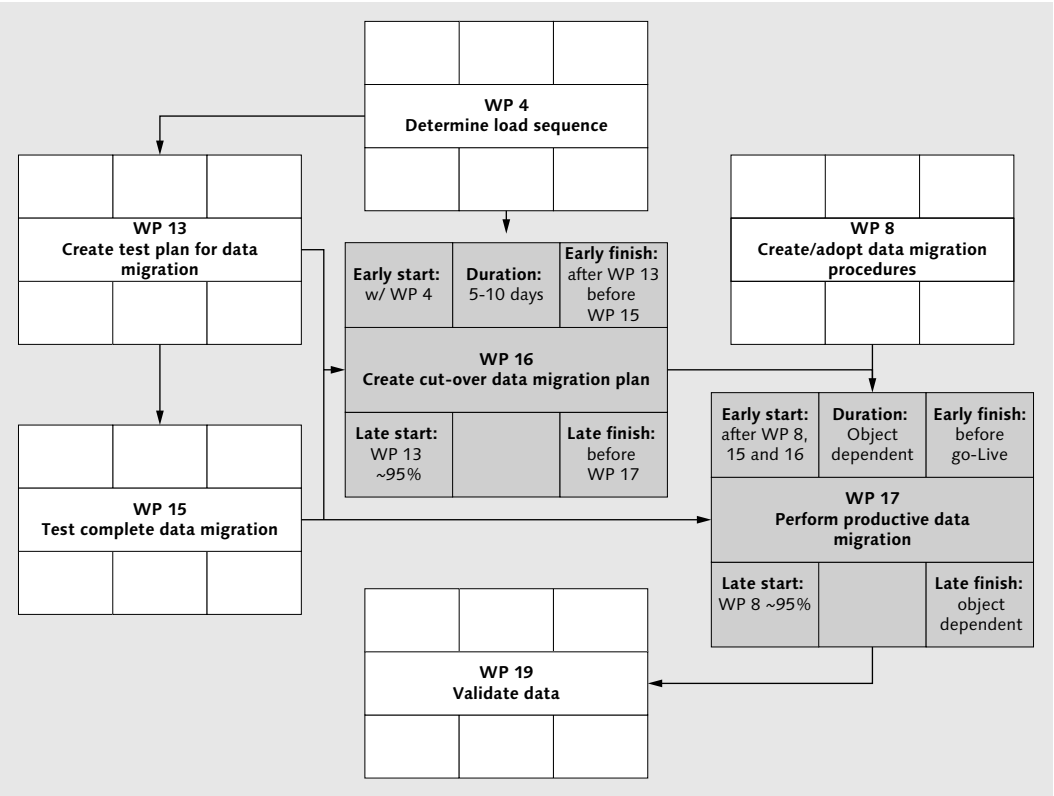


Figure 3.9 Work Packages of the Productive Load and Support Phase

Work Package 16: Create Cut-Over Data Migration Plan

Before you start migrating your data, you have to create a *cut-over migration plan*, which will be later integrated into the big cut-over project plan. The best time to start creating the cut-over migration plan is when you determine the load sequence. However, you shouldn't start until you've created your data migration test plans. The test plans are the basis for the cut-over migration plan. As a general rule, the migration plan is finished after the last tests are passed. The effort to create such a plan shouldn't exceed more than five days. In phased roll-out projects, however, this could take longer due to the number of planned go-lives.

Narrow time frame If you have a narrow time frame to perform the initial data migration, you should categorize your migration object according to go-live importance.

Some business objects are very stable and are rarely changed or extended for a longer period. You can therefore migrate these objects days or even weeks before your go-live into the target system. As a prerequisite, the needed customizing must already be imported into the target system. You then have to double maintain your data in the source and the target system and document every change. Migrating in a system already in operation includes a certain amount of risk, which must be evaluated beforehand.

Some migration objects don't have to be available in the target system at go-live. You can migrate such migration objects after go-live but note that this will lead to a higher system load at migration time and that you load data into a system that is already in operation.

Work Package 17: Perform Productive Data Migration

This is one of the final work packages of a data migration project. The prerequisites are as follows:

- ▶ All data migration programs/procedures (extract and load) are finished.
- ▶ At minimum, one successful complete data migration test is passed.
- ▶ The cut-over plan, including the data migration load plan, is finished.

Generally, the data migration load plan rules the cut-over project plan; that is, the cut-over plan follows the lead of the migration load plan.

3.3.7 Data Validation

An overview of the work packages and their dependencies in the *data validation* phase is shown in Figure 3.10.

There are two points in time in a data migration project to validate your migration object's data:

- ▶ **Before the data migration**
You validate your source data integrity and correctness before you load—sometimes even before you convert your data.

► **After the data migration**

You validate your data integrity and correctness after you load with the data following the defined validation rules.

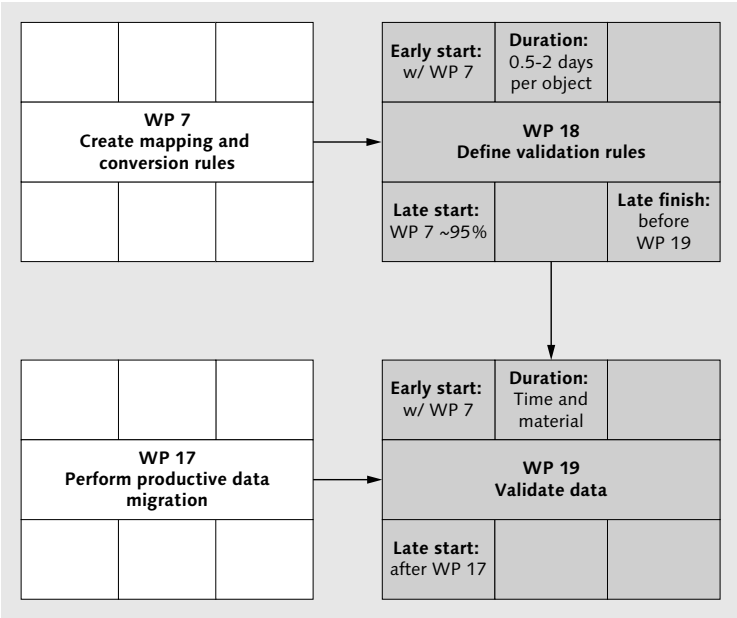


Figure 3.10 Work Packages of the Data Validation Phase

Both variants have their respective merits. If you validate before you load the data, you can prevent loading garbage into your system. If you validate after you load the data, you can check the migrated data after passing the conversion and the program logic of the program used to migrate the data. The big advantage of validating after loading is that you can use test tools such as the *extended Computer Aided Test Tool (eCATT)* to automatically run through complete business processes and check the correctness and data integrity of the loaded business object. You can also create *SAP Queries* based on the validation rules up front and transport them to the target system. This is an easy way to validate your data as fast as possible on formal aspects. The advantage of using validation tools and queries is that you can reuse them after go-live, for example, to test your processes installing service patches, SAP Notes, upgrades, and so on. You may also be able to use them for data cleansing

processes in your new system. You only have to build them once and then can use them many times.

In principle, it often makes more sense to validate each business object after the load. You should proceed with loading dependent objects only after the validated business objects have been released. This process is very time-consuming, but it guarantees that your migrated data is compliant with your quality standards. Choosing to validate the target data is especially wise if there are very complex conversion rules used to convert the source data.

Validation Scheduling in a Customer Project

[Ex]

In a big pharmaceutical industry project, the data validation must be done following FDA rules. Due to the limited time frame to migrate the data, the company invented a 24/7 project plan to migrate and validate the data in several teams around-the-clock. After a migration object was loaded, the related staff members to validate the data were called to get to the company. They ran a three-shift operation to check and update the project plan and to control the complete project process. The three project plan teams worked in overlapping nine-hour shifts and controlled the schedule, corrected times, and called the staff to migrate or validate the data at least one hour before they had to perform their activities. It was therefore essential to perform a lot of tests to be able to document the estimated duration of the several data migration steps as accurately as possible. Without testing and documenting the times, scheduling different teams and staff, as well as remaining compliant with the standard operating procedures and the legal regulations, would have been impossible. They were rewarded for their intensive validating using this approach by a very smooth go-live and an extremely small number of erroneous data records that needed to be corrected after the migration; for example, only 20 of 32,000 loaded material master records had to be corrected manually after the validation.

Work Package 18: Define Validation Rules

Validation rules must always be created and defined together with the client's staff. The mapping and conversion rules must be created first and can also be used as a base for the validation rules. Some tools and procedures to load data into systems, such as Rapid Data Migration with SAP Data Services (see Chapter 8), have predefined or built-in functions to validate migrated data.

Syntactical checks

[Ex]

Syntactical Checks

Tools often provide built-in functions to syntactically check loaded data for correctness in the following areas:

- ▶ Correct field length
- ▶ Correct and valid date and time formats and values
- ▶ Mandatory field checks
- ▶ Format check for number fields, and so on

You can find more about this Chapter 8, Section 8.3.6.

Semantical checks In addition to the syntactical tests, you should also perform semantical tests.

[Ex]

Semantical Checks

There is no way to prebuild semantical checks up front to check the correctness of the data contents. The checks are dependent on the context of the data and vary from project to project and object to object. Valid checks might include the following:

- ▶ Each customer of a specific sales district must be assigned to a certain sales group.
- ▶ Materials/products of a certain product group must be assigned to a specific purchasing group.
- ▶ Materials that overrun specific values (volume, weight, or dimensions) must be assigned to special storage bins.

Such checks must be individually defined per migration project. They can be automated by defining SQL queries (e.g., SAP Query) to a certain extent. If you intend to use such queries and checks after go-live to continuously check and improve your data quality, you should take tools such as *SAP Information Steward* into consideration. With the aid of such tools, you can define rules and thresholds to check and classify your data.

Work Package 19: Validate Data

As mentioned earlier, you can perform the data validation before or after the data load. If it's done after the data load of an object, you should consider a successfully passed validation check as a prerequisite to loading other dependent objects. This improves your data quality and

prevents issues when loading such dependent objects. It's essential that the validation is done by the client itself and that all results are documented.

3.4 Planning and Effort Estimation

Now that you know the data migration work packages and their sequence and relationships, let's discuss how to determine the business objects to migrate and how to estimate the duration of the overall data migration process. This section also provides some formulas and numbers given by some internal evaluations of former data migration projects.

3.4.1 Determine Business Objects to Migrate

To determine the business objects to migrate, you should record the following information about the business objects:

- ▶ Target SAP system (product, version, system ID).
- ▶ Name of the business object and a short description. To determine a meaningful object description, you can use the Transaction SAPTERM.
- ▶ Object type (master data, transaction data, customizing, other data). For some objects (sales or purchasing contracts), it won't always be clear whether it's considered transaction data or master data. In these cases, check the frequency of change, the number of newly created objects in a period, and the number of relation to other objects. An object that is created once without many changes within a period and related to many other objects is probably a master data object. In contrast, there are many transaction data objects created in a specific period, and they may have many changes within this period.
- ▶ Estimated overall complexity of the business object (dependency on other business objects, complexity of the object's data structure, number of fields to migrate, number of conversion rules needed, complexity to implement the conversion rules, etc.) from 1 = less complexity to 10 = very complex.
- ▶ Dependency on other business objects to migrate.
- ▶ Number of records to load (volume).

- ▶ Source system (name, product, version, location).
- ▶ Major contact persons for the following:
 - ▶ Business object on customer/client site and in the implementation project.
 - ▶ The legacy or source system to extract the data.
 - ▶ Implementing the data migration in the target system.
- ▶ Name and product version of the tool(s) to do the following:
 - ▶ Extract the data.
 - ▶ Cleanse the data.
 - ▶ Convert the data.
 - ▶ Load the data.
 - ▶ Validate the data.
- ▶ Name and location of the related BBP or business process documents.

Store this information in a central table or spreadsheet (e.g., Microsoft Excel), and complete this with additional required object or status information. An example for such an object list is shown in Figure 3.11.

														Source systems				Region 1				Region 2	Region 3	Object
No.	Area	Object number	Object name	Object Sub number	Description	Comments	MD= master data CH=Customizing TC=Transaction SH=Shell	Data migration needed Y or N or M= manually	Normal Complexity 1 - low 10 - very high	Implementation Effort 1 - less 10 - very high	Dependencies No. (not A) which must be completed before this could run	Linked Release number (if records)	System 1	System 2	System 3	System 4	System 5	System 6	Business lead (SAP) e.g. Mapping, Validation	Business lead (Customer) e.g. Mapping, Validation			SAP Technical responsible e.g. Upload	
+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	
1	SD	001	Customer Master	001	Load Additional Customers (Create new)	02/27/07 Mr. SAP1: ask Ms Gamma for further informations	MD	Y	3				X		X				Mr. SAP1	Mr Beta	Ms Gamma	n/a		
2	SD	001	Customer Master	002	Additions to existing Customers (Change existing)	02/27/08 eMail by Mr Alpha Shipping conditions "E"	MD	Y	3		1 001-001		X		X				Ms SAP2	Mr Alpha Ms Fox	n/a	n/a		
3	MM	002	Vendor Master	001	Local Purchase Vendors	03/27/09 Mr Beta: no migration except for some bolt-on's. Mr Fox to cleanse and further informations Spreadsheets exists for 0002 for Ms SAP2 and Mr SAP1	MD	Y	2						X				Mr. SAP1	Mr Beta	n/a	n/a		
4	MM / SD	003	Material Master	001	New materials to create & existing to extend (org struct)	03/27/09 Mr. Beta: * Only new materials, no field changes in for existing materials (to verify) .	MD	Y	5				X						Ms SAP2 Mr. SAP1	Mr Alpha Ms Fox	n/a	n/a		
5	MM / WM / SD / PP	003	Material Master	002	Plant extensions of existing materials MRP + Work scheduling + Storage location	03/27/09 Mr Beta: MRP1 to MRP4 (MRP1 no dispo areas MRP4 also MRP indicator MARD-DISK2) * Work scheduling (Serial number may be needed)	MD	N	6				X	X					Mr. SAP1	Mr Alpha Mr Beta	n/a	n/a		
6	SD	003	Material Master	003	Sales view Extended Tax classifications to all existing Sales views		MD	Y	2										Mr. SAP1	Ms Gamma				
7	WM	003	Material Master	004	Shelf life data (Plant storage-1)		MD	N	4				X		X									

Figure 3.11 Migration Object List Example Using a Microsoft Excel Template

3.4.2 Estimate the Effort

If data migration projects are well organized and planned, the data migration's consulting effort to migrate should average out at 7 – 10% of the whole project's consulting effort. If data migration projects are ill-prepared or poorly planned, the data migration's consulting effort might exceed more than 30% of the overall consulting effort. The following numbers are based on internal evaluations of former data migration projects. They can serve as a rule but can vary from project to project.

Effort per project phase

- ▶ **Business blueprint (BBP)**
During the BBP, the effort for the data migration project will be less. You should project between 2% and 5% of the overall data migration work packages' effort for this phase.
- ▶ **Realization**
The main effort for the data migration occurs during the realization phase, and is between 50% and 70% of the total data migration effort. Just for testing and test support, you should count on 30 – 40%.
- ▶ **Final preparation**
After the first tests at final preparation to test the complete data migration process at go-live, the effort will decrease to approximately 10% of the total data migration effort.
- ▶ **Go-live and support**
To perform the productive data migration and to support the project after the go-live, you should project between 5% and a maximum of 10% of the total of the data migration effort. It could be less, however, depending on the go-live preparation and the data validation outcome.

After determining the required work packages, you can start to estimate the project's effort using an Excel spreadsheet. For some work packages, the effort can be calculated as a flat effort, for example, work package 16 (create cut-over data migration plan), work package 1 (determine business objects to migrate), work package 2 (identify source data systems), and so on. For some phases, especially the data analysis phase, you can roughly calculate the effort using a simple formula. The formulas in the following boxes have been used and verified in various data migration projects for years. While they vary from project to project, you can use them as a rough rule of thumb.

[+]

Data analysis effort

Formula to Estimate the Effort for the Data Analysis Phase

Estimate the complexity of the business object to migrate on a scale from 1 (less complexity) to 10 (very high complexity), and divide this number by 2. You can use the following example criteria to evaluate the complexity:

- ▶ Numbers of data sources (systems)
- ▶ Number of fields to map
- ▶ Number of conversion rules to implement
- ▶ Simple one-to-one conversions or additional program code requirements
- ▶ Number of records
- ▶ Quality of the source data and related conversion rules to correct wrong entries automatically

The calculated number is the effort in person days that will be used approximately for the data analysis phase work package 1 (determine business objects to migrate).

[+]

Data mapping effort

Formula to Estimate the Effort for the Data Mapping

You can use the data analysis rule for the data mapping work package 7 (create mapping and conversion rules), too. Because of different data sources or complex data conversion rules, there may be more technical data migration objects for one business object. This may reduce the effort per migration object to determine the conversion rules and map the fields and structures because of the synergy effects and the learning curve for this object. If, for example, the customer master data is loaded via IDoc from different source systems, the effort for the mapping and creating the conversion rules can be calculated for the first migration object using the formula. The effort for the further customer master data migration objects can be estimated up to a half or a quarter compared to the first object's effort.

Implementation effort

For implementing the data extraction and data migration programs/procedures, there will be no such formula. The number of fields and structures to map and the number of conversion rules to implement are the crucial factors. Even the experience of the person implementing the rules and procedures might not be underestimated. While a very experienced data migration specialist needs, for example, one day to implement and test a specific data migration object, a less experienced person may need several days to do the same.

The effort to test and load the data into the target system depends on the time to load the data. It can be calculated more accurately after the first performed data load tests. To preestimate such numbers, you can resort to empirical values of other projects.

The effort to validate the data depends on the volume of records to validate and the complexity of the validation rules to perform. It can't be calculated using a formula. You can decrease the effort if you use a tool to implement the validation rules to let them run automatically after the data load. Implementing is only done once, and then you just have to consider the run time of the automatic data validation procedure.

Test and data load effort

Data validation effort

[+]

Data Migration to SAP S/4HANA, Cloud Edition

For data migration to SAP S/4HANA, cloud edition, some work packages can be excluded because of the given landscape, for example, work package 20 (identify tools) and work package 12 (implement test landscape). Some other work packages may be calculated with less effort than on usual SAP implementing projects because of the predefined data migration content, for example, work package 7 (create mapping and conversion rules) and work package 8 (create/adopt data migration procedures).

Figure 3.12 shows an example effort estimation using a Microsoft Excel spreadsheet.

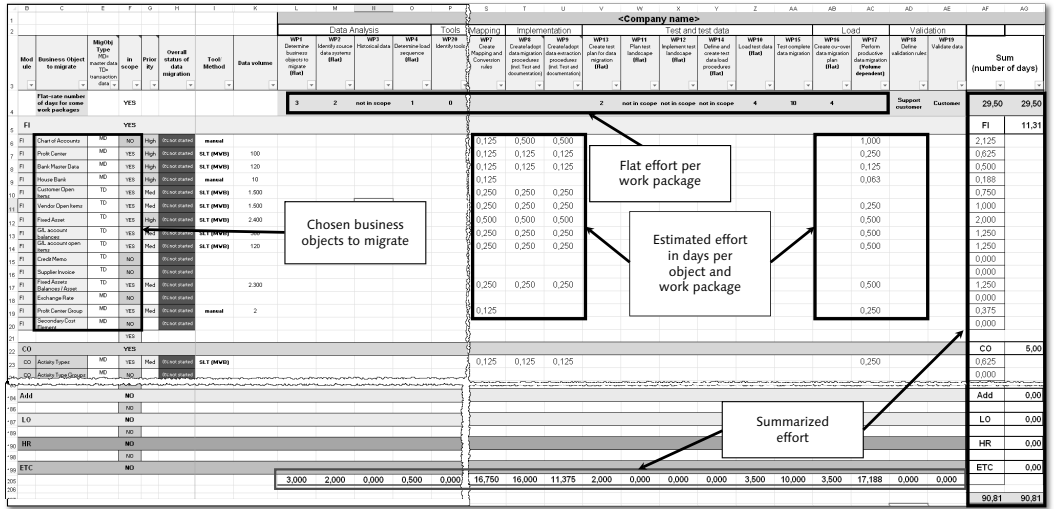


Figure 3.12 Effort Estimation Example Using a Microsoft Excel Template



Note

You can download this spreadsheet to determine your objects and estimate your effort from the book's web page. It can be used in your project and adopted to your needs. We also don't warrant or represent that the spreadsheet is valid and without any errors.

3.5 Summary

You've now learned about the data migration phases and how they are integrated into the phases of a classical SAP implementation project. We've shown you how to organize your data migration projects by dividing the tasks into work packages and their dependencies. Using these work packages, you can see the big picture of complex data migration projects to plan and organize them, as well as better calculate the effort required. The procedures and formulas to calculate the effort, presented in the final section, are rough rules of thumb. We don't warrant or represent that the given numbers and formulas are also valid for your projects.

Contents

Introduction 15

1 Business Basics for Migrating Data to SAP 23

1.1 Data Migration as a Subproject 23

1.2 Preliminary Considerations 25

1.2.1 Defining the Dataset for Migration 25

1.2.2 Identifying Dormant Data 26

1.2.3 Measures for Reducing the Data Volume 28

1.2.4 Preparatory Measures for Extracting the Legacy Data 28

1.2.5 Addendum: Accounting Considerations 29

1.3 The Data Migration Process from the Project Perspective 35

1.3.1 Basic Customizing 35

1.3.2 System Presentations in SAP 36

1.3.3 Business Reengineering 36

1.3.4 Simulating the Data Migration 37

1.3.5 Mapping (Field Matching) 37

1.3.6 Data Extraction from the Legacy System 39

1.3.7 Manual Postprocessing of the Extracted Data 39

1.3.8 Selecting a Data Migration Technique 39

1.3.9 Uploading the Data in SAP 40

1.3.10 Testing the Business Processes in SAP 41

1.4 Summary 42

2 Technical Basics for Migrating Data to SAP 43

2.1 Basic Terminology 43

2.2 The Data Migration Process from a Technical Perspective 45

2.2.1 Exporting the Data 45

2.2.2 Reading the Data 45

2.2.3 Converting the Data 46

2.2.4 Importing the Data 48

2.2.5 Verifying the Data 50

- 2.3 Overview of Technical Procedures for Data Migration 50
 - 2.3.1 Batch Input 50
 - 2.3.2 Legacy System Migration Workbench 51
 - 2.3.3 SAP Data Services 51
- 2.4 Summary 52

3 Plan and Organize Your Data Migration Projects 53

- 3.1 Types of Data Migration Projects 53
 - 3.1.1 Initial Load (Green Field) 53
 - 3.1.2 Upgrading an Existing System (Brown Field) 54
 - 3.1.3 Phased Roll-Out 54
 - 3.1.4 Big Bang 54
 - 3.1.5 System Optimization 54
 - 3.1.6 Cloud Migration 55
- 3.2 Phases of Data Migration Projects 56
 - 3.2.1 Data Analysis 58
 - 3.2.2 Mapping 58
 - 3.2.3 Implementation 59
 - 3.2.4 Data Migration Tests 59
 - 3.2.5 Data Validation 59
 - 3.2.6 Data Cleansing 60
 - 3.2.7 Productive Load and Support 61
- 3.3 Work Packages in Data Migration Projects 62
 - 3.3.1 Data Analysis 64
 - 3.3.2 Data Cleansing 71
 - 3.3.3 Mapping 73
 - 3.3.4 Implementation 76
 - 3.3.5 Data Migration Tests 80
 - 3.3.6 Productive Load and Support 85
 - 3.3.7 Data Validation 87
- 3.4 Planning and Effort Estimation 91
 - 3.4.1 Determine Business Objects to Migrate 91
 - 3.4.2 Estimate the Effort 93
- 3.5 Summary 96

4 Batch Input 97

- 4.1 How Batch Input Works 98
 - 4.1.1 A Batch Input Session 98
 - 4.1.2 Processing a Batch Input Session 101
 - 4.1.3 Creating a Batch Input Session 104
- 4.2 Standard Batch Input Programs 105
 - 4.2.1 Program RFBIDE00: Customer Master 106
 - 4.2.2 Program RFBIKR00: Vendor Master 106
 - 4.2.3 Program RFBISA00: General Ledger Account Master 107
 - 4.2.4 Program RFBIBL00: Financial Documents 107
 - 4.2.5 Program RCSBI010, RCSBI020, RCSBI030, RCSBI040: Bill of Materials 107
 - 4.2.6 Program RM06BBIO: Purchase Requisitions 108
 - 4.2.7 Program RM07MMBL: Material Documents 108
- 4.3 Batch Input Recording: General Approach 109
 - 4.3.1 Creating the Batch Input Recording 109
 - 4.3.2 Generating an ABAP Program from a Batch Input Recording 114
 - 4.3.3 Modifying the Generated ABAP Program 118
 - 4.3.4 Creating and Processing the Batch Input Session 126
 - 4.3.5 Call Transaction and Batch Input Session in Case of Error 128
- 4.4 Summary 130

5 IDoc and ALE Distribution 131

- 5.1 IDoc Basics 131
 - 5.1.1 IDoc as an Interface 132
 - 5.1.2 Using IDocs 132
 - 5.1.3 IDoc Customizing in SAP 136
 - 5.1.4 Optimizing IDoc Performance 138
- 5.2 Data Transfer Workbench 142
 - 5.2.1 Basics 142
 - 5.2.2 Data Migration Using BAPIs 144
 - 5.2.3 Combining the Data Transfer Workbench and the LSMW 145
 - 5.2.4 Data Transfer Workbench Tools 146
- 5.3 Summary 147

6	Legacy System Migration Workbench	149
6.1	Overview of the LSMW	149
6.2	Data Migration with the LSMW	152
6.2.1	Getting Started with the LSMW	153
6.2.2	User Guide: The Main Steps of Data Migration	158
6.2.3	Maintaining Object Attributes	159
6.2.4	Maintaining Source Structures	163
6.2.5	Maintaining Source Fields	164
6.2.6	Maintaining Structure Relations	170
6.2.7	Maintaining Field Mapping and Conversion Rules	173
6.2.8	Maintaining Fixed Values, Translations, and User-Defined Routines	185
6.2.9	Addendum: Files	193
6.2.10	Specifying Files	195
6.2.11	Using Wildcards in File Names	201
6.2.12	Assigning Files	202
6.2.13	Reading Data	203
6.2.14	Displaying Read Data	205
6.2.15	Converting Data	207
6.2.16	Displaying Converted Data	209
6.2.17	Importing Data	210
6.2.18	Object Overview	213
6.3	Recordings	215
6.3.1	Creating and Editing Recordings	217
6.3.2	Using a Recording	223
6.4	Long Texts	226
6.4.1	Long Texts in the SAP System	226
6.4.2	Target Structures and Field Mapping	227
6.4.3	Importing Long Texts	231
6.5	Transporting Projects	232
6.5.1	Creating a Change Request	232
6.5.2	Exporting Projects	233
6.5.3	Importing Projects	234
6.6	Preparations for Using IDoc Inbound Processing	234
6.7	Summary	237

7	Legacy System Migration Workbench for Experts	239
7.1	Advanced Functions	239
7.1.1	Layout and Processing Times	240
7.1.2	Global Variables	242
7.1.3	Global Functions	243
7.1.4	Reusable Rules: Naming Conventions	243
7.2	Advanced Applications	245
7.2.1	Determining the Transaction Code at Runtime	245
7.2.2	Skipping a Record	246
7.2.3	Skipping All Records of a Transaction	246
7.2.4	Duplicating a Record	246
7.2.5	Assigning Multiple Source Structures to a Target Structure	248
7.2.6	Assign a Random Number of Records to One Transaction	249
7.2.7	Error Messages in the Conversion Log	254
7.3	The LSMW Conversion Program	255
7.3.1	Activate Additional Process Steps	255
7.3.2	The Conversion Program's Structure	256
7.4	User-Defined Routines and Advanced Tips	259
7.4.1	Initialize IDoc Structures with a Nodata Sign	259
7.4.2	Directly Modify Table Records Using the LSMW	262
7.4.3	Adding Selection Parameters to the LSMW Conversion Program	264
7.4.4	Assign Multiple Source Structures to One Target Structure with Substructures	267
7.4.5	Extend LSMW with an Advanced Conversion Log	274
7.4.6	Using Conversion Exits and Routines	281
7.5	Summary	287
8	Rapid Data Migration with SAP Data Services	289
8.1	SAP Data Services Overview	290
8.2	Rapid Data Migration Architecture	292
8.3	Rapid Data Migration Process	298
8.3.1	Data Migration Content	299
8.3.2	Connecting Source Systems	308

- 8.3.3 Data Profiling 311
 - 8.3.4 Field Mapping 316
 - 8.3.5 Value Mapping 325
 - 8.3.6 Data Validation 333
 - 8.3.7 Data Load 337
 - 8.3.8 Monitoring 340
 - 8.4 Lookups and Validations 343
 - 8.4.1 SAP Lookup Tables 344
 - 8.4.2 Translation Tables 345
 - 8.5 SAP Data Services for Experts 346
 - 8.5.1 Scripting Language 346
 - 8.5.2 System Functions 346
 - 8.5.3 Custom Functions 348
 - 8.5.4 Validation Enhancements 350
 - 8.5.5 Data Quality 352
 - 8.6 Data Migration to SAP Cloud Solutions 353
 - 8.6.1 Rapid Data Migration to the Cloud 354
 - 8.6.2 Interfaces for SAP Cloud Solutions 356
 - 8.7 Summary 358

9 Data Migration in SAP Business ByDesign and SAP Cloud for Customer 359

- 9.1 Migration Templates 360
 - 9.1.1 Structure and Elements of Migration Templates 360
 - 9.1.2 User Tips 363
 - 9.1.3 Field Formats 365
 - 9.1.4 Export Data from the Source System 368
 - 9.1.5 Filling Data into Migration Templates 369
 - 9.1.6 Alternative Options to Populate Migration Templates 370
 - 9.1.7 Data Cleansing 372
- 9.2 Migration Tool 373
 - 9.2.1 Navigation in SAP Business ByDesign 374
 - 9.2.2 Navigation in SAP Cloud for Customer 375
 - 9.2.3 Migration Tasks in the Implementation Project 376
 - 9.2.4 Prepare for Data Migration 377
 - 9.2.5 Migration Objects 380
 - 9.2.6 Migration Cockpit 383

- 9.2.7 Upload CSV Files 388
 - 9.2.8 Migration Cockpit: More Features 392
 - 9.2.9 Integrated Data Editor 394
 - 9.3 Migration Process Steps 399
 - 9.3.1 Validate File 400
 - 9.3.2 Convert Values (Value Mapping) 407
 - 9.3.3 Simulate Import 415
 - 9.3.4 Import Data 424
 - 9.4 Migration of Accounting Transactional Data 428
 - 9.4.1 Preparation in SAP Business ByDesign 429
 - 9.4.2 Preparation in the Source System 431
 - 9.4.3 Migration Unit 434
 - 9.4.4 Migration Templates in Accounting 439
 - 9.4.5 Accounting Migration: Main Steps 442
 - 9.4.6 Simulation of Single Files 443
 - 9.4.7 Posting Preview 444
 - 9.4.8 Cross-Check of Balances 446
 - 9.4.9 Acceptance of Imported Data 450
 - 9.4.10 Postprocessing 451
 - 9.5 Change Migration 452
 - 9.5.1 Add New Records 453
 - 9.5.2 Enhance Existing Records 456
 - 9.5.3 Modify Existing Records 460
 - 9.5.4 Replace Existing Records 462
 - 9.5.5 Mass Data Maintenance 464
 - 9.6 Migration as a Subproject 470
 - 9.6.1 Project Management 470
 - 9.6.2 Migration Tasks in Implementation Projects 472
 - 9.6.3 Test Strategy 474
 - 9.7 Tips and Tricks 476
 - 9.7.1 Volume Performance by Parallelization 477
 - 9.7.2 Reconciliation of Migration Results 479
 - 9.8 Summary 482

10 Techniques for Avoiding Programming 483

- 10.1 Problem Area: Data Conversion 483
- 10.2 Techniques for Converting Data 484
 - 10.2.1 Modifying Structures 484
 - 10.2.2 Modifying Field Contents 489

10.2.3	Using Microsoft Excel VLOOKUP Instead of Microsoft Access Database Join	499
10.3	Summary	503
11	Assessment of Data Migration Techniques	505
11.1	Use Cases	505
11.2	Advantages and Disadvantages of the Procedures	506
11.2.1	Batch Input	506
11.2.2	Legacy System Migration Workbench	509
11.2.3	SAP Data Services	510
11.3	Reasons for Favoring a Certain Procedure	511
11.3.1	Complexity of the Migration Task	511
11.3.2	Quality of the Legacy Data	512
11.3.3	Data Volume	512
11.3.4	The Importance of Data Security	513
11.3.5	Reusability	513
11.3.6	Restrictions	514
11.3.7	User-Friendliness	515
11.4	Summary	515
12	Advanced Topics and SAP S/4HANA	517
12.1	Data Migration between SAP Systems or within an SAP System	517
12.1.1	SAP Landscape Transformation	518
12.1.2	Using the Data Browser to Extract Records	523
12.2	Database Migration	527
12.2.1	Basics	528
12.2.2	Rapid Database Migration	528
12.3	Migration Paths to SAP S/4HANA	533
12.3.1	Migrating Data to SAP S/4HANA	533
12.3.2	SAP Activate	537
12.3.3	SAP S/4HANA Cloud Editions	540
12.3.4	SAP S/4HANA, On-Premise Edition	543
12.3.5	Integrating Your Data with SAP S/4HANA, Cloud Marketing Edition	545
12.4	Summary	548
The Authors	549
Index	553

Index

.TXT Format, 195
21 CFR Part 11, 503
24/7 project plan, 89

A

ABAP, 246, 508
ABAP commands, 266
ABAP conversion program, 249, 257
ABAP Dictionary, 98
ABAP Editor, 184, 192, 255
ABAP includes, 266
ABAP program, 118
ABAP statement, 246
ABAP Workbench, 510
Absolute amount, 494
Account determination group assignments, 382
Account type, 34, 38
Accounting data, 371
 consistency, 440
Accounting objects, 436
Accounting transactional data, 394
 migration, 428
Administration, 157
Advanced conversion log, 275, 286
Alternative units of measure for material, 65
Analyze data cleansing demand, 72
Application Link Enabling, 131–132, 295
 communication, 136
Application Programming Interface (API), 355–356
 OData, 357
Application server, 151
Archiving flag, 263
ASAP, 56, 62
 business blueprint, 56, 93
 final preparation, 56, 93
 go-live and support, 56, 93
 project preparation, 56
 realization, 56, 93

Asset value, 31
Authorization concept, 157
Authorization in production systems, 264
Authorization profile, 157
Auto field mapping, 179

B

Background processing, 138
Backup, 83
Backup and restore, 40
BAdI, 72
Balance sheet account, 30
Balance sheet item, 30
Balances, 26
Balances preview, 446
BAPI, 97, 144–145, 150, 291, 489, 491, 509, 512–513, 520
BAPI_MATERIAL_SAVEREPLICA, 74
Batch input, 18, 50, 73, 78, 145, 150, 161, 506
 background processing, 127
 include bdcrcx1, 117
 interface, 506
 PERFORM bdc_dynpro, 116
 PERFORM bdc_field, 116
 PERFORM bdc_transaction, 117
 PERFORM close_group, 117
 PERFORM open_group, 116
 processing, 97
 program, 508
 recording, 104, 109–110, 112, 509, 514
Batch input session, 98, 104, 127, 506, 509, 514
 blocked, 103
 blocking and releasing, 104
 deleting, 104
 display errors only, 101, 127
 error, 103
 in background, 103
 in process, 103

Batch input session (Cont.)
 in the background, 101
 lock date, 126
 new, 103
 nodata indicator, 126
 process in foreground, 101, 127
 processed, 103
 processing automatically, 103
 retained, 102
 user, 126
Batch mode, 281
BDCDATA, 98
Best practices, 295
Big bang, 54
Bills of material (BOM), 40, 107
Business blueprint (BBP), 58, 65
Business data object, 44
Business document (BDoc), 134
Business object, 44, 46, 58, 359
Business objects to migrate, 91
Business partner relationships, 382
Business process, 41
Business process integration test, 83
Business reengineering, 25, 37, 39
Business reporting, 298

C

Call transaction, 117, 128
Carve out, 519
Central Management Console (CMC), 292
Change history, 519
Change migration, 452
 change records, 460
 new records, 453
Change request, 232
Charts of account, 27
Check tables, 345, 511
Checkbox structure, 185
Cleansing rule, 61
Cloud, 55
Collective view, 404
Comment, 280
Company code, 137, 244
Compliance, 503

Concatenation, 184
Configure Your Solution, 539
Continuous interface, 79
Conversion, 46, 514
Conversion exit, 282, 284
Conversion log, 254, 274
Conversion log message, 278
Conversion routine, 284
Conversion rule, 51, 153, 173, 509
Conversion status, 409
Conversion tasks, 51
Convert text to columns wizard, 493
Converted data, 274
Converting data, 150
Converting legacy data, 483
Converting numbers, 494
Core Interface, 67
Country code, 334, 407
Create cut-over data migration plan, 69, 86
Create test plan for data migration, 84
Create/adopt data extraction procedures, 80
Create/adopt data migration procedures, 79
Credit posting, 38
CRM Middleware, 67
Cross-check of balances, 446
Cross-system company code, 136
CSV file, 388, 467
Currency field, 367
Currency format, 493
Cursor position, 123
Custom function, 348
Customer hierarchies, 455
Customer master, 106, 246
Customer-specific extension fields, 379
Customizing, 24, 35
Cut-over plan, 69
Cut-over project plan, 86

D

Data
 analysis, 58, 64, 93
 cleansing, 60–61, 71, 291, 298, 372

Data (Cont.)
 cleansing tool, 69
 conversion, 46, 151, 209, 483, 509
 converting, 46
 correction, 452
 editor, 386–387, 394, 404, 467–468
 errors, 420
 exporting, 45
 extracting, 45
 format, 313
 importing, 48
 load tool, 69, 337
 loading, 48
 loss, 462
 mapping, 290
 migration demo, 355
 migration object, 44, 154
 migration technique, 79
 migration test, 58–59
 migration test plan, 85
 migration tests, 80
 migration tool, 79
 normalization, 291
 object, 44, 150, 243, 489, 509
 profiling, 291, 297
 quality, 60–61, 80, 297, 352, 512
 reading, 45
 records, 337
 security, 39, 513
 transfer, 44
 transformation, 46
 transforming, 46
 type, 310
 unloading, 45
 validation, 59, 80, 87, 291, 355, 400
 validation tool, 69
 volume, 28, 512
Data Browser, 518
Data Dictionary, 282
Data Migration Server (DMIS), 542
Data Services → SAP Data Services
Data Transfer Library, 146
Data Transfer Workbench, 70, 142, 146, 152, 513
 error handling, 144
 run, 143
Data type and format validation, 333

Data validation, 400
 semantical check, 90
 syntactical check, 89
Data variety, 474
Database
 table, 259, 321
Database migration, 527
 package, 529
 rapid database migration, 528
Database Migration Option of SUM
 DMO, 531–532
Database server, 506
Datastore, 308
Date format, 367, 489
Date value, 489
DDMMYY, 47
Debit posting, 38
Decimal marker, 367
Decimal point, 491
Default values, 47
Define and create test data load procedures, 84
Define validation rules, 89
Defining the dataset, 25
Delta file, 444
Determine business objects to migrate, 64
Determine load sequence, 69, 86
Dialog processing, 110, 506
Direct function call, 134
Direct input, 145, 150, 161, 509, 513
Direct input session, 211
Document number changes, 252
Document type, 34
Dormant data, 25–26
Drag and drop, 52
Dress rehearsal, 476
Duplicating a record, 246
Dynpros, 97, 112

E

EDIFACT standard, 132
Effort estimation, 93
Electronic data interchange (EDI), 132, 339

Employee data, 382
Endpoint URL, 357
Enhance existing records, 456
Enhanced mode migration, 457
Enhancement package, 530–531
Erroneous data records, 263, 313
Error, 128
Error analysis, 405, 419
Error file, 144
Error handling, 515
Error log, 402, 406
Error message, 254, 418
 unclear cause, 421
Error session, 128
Errors, 387, 396, 449
Estimate the effort
 data analysis, 94
 data analysis formula, 94
 data mapping, 94
 data mapping formula, 94
 data validation, 95
 implementation, 94
 test and load, 95
ETL, 47, 69, 289–290, 323
Export, 414
 value conversion, 392
Export data, 368
Exporting legacy data, 483
Extension field, 481
Extension fields, 393
Extract system, 290
Extraction program, 80

F

Fact sheet, 381
FDA, 89
FDA/GMP, 503
Field assignment, 323
Field contents, 114
Field documentation, 181
Field format, 365
Field mapping, 46, 173, 240, 255, 296, 305, 321, 371, 389–390
 layout, 240
Field names, 119

File, 44, 193
 amount field, 204
 assignment, 202
 character set, 195
 code page, 195
 date value, 204
 end-of-record indicator, 193
 for converted data, 200
 for read data, 200
 format, 514
 read authorization, 199
 sequential, 194
 table, 194
 wildcards, 201
 write authorization, 199
File upload, 385
 errors, 386
Filling data, 369
Financial Accounting, 38
Financial documents, 107
Fixed values, 243
Form routines, 117
Frontend, 44, 197
FTP server, 338
Function module, 121
 MATERIAL_MAINTAIN_DARK, 74

G

General Ledger, 29
 account, 30, 33
 account master, 107
 account postings, 35
 accounting, 31
Generally accepted accounting principles
 GAAP, 68
Generated code, 272
GL account line item, 123
Global data, 266
Global variable
 g_nodata, 260
Global variables, 240
Go-live, 61, 79, 88
Governance, 296
Guided configuration, 539

H

Historical data, 67, 519
Hit probability, 180
Hosted cloud, 55
HTTP
 messages, 358

I

Identifier fields, 411
Identify source data systems, 66
Identify tools, 69
Identify your source systems, 66
IDoc, 49, 73, 78, 97, 131, 134, 144, 150, 161, 234, 259, 291, 299, 301, 340, 489, 491, 509, 512
 administrator, 136
 creating an IDoc overview, 212
 Customizing, 136
 extensions, 133
 format, 144
 inbound, 340
 inbound processing, 144, 234, 236
 message types, 133
 performance, 138
 segment, 341
 segments, 302
 starting the IDoc creation, 212
 starting the IDoc processing, 212
 structure, 316
 target structure, 306
 type, 49
 work item, 212
IDoc type MATMAS06, 74, 260
Implement test landscape, 83
Implementation, 59, 76
Implementation method, 56
Implementation project, 56, 474
Import data, 424
Import run, 450
Import simulation, 415
 best practices, 418
Import technique, 161
Import value conversion, 392
Importing data, 150

Inbound processing, 49
Income statement account, 26
Inconsistent values, 403, 405
Industry-Specific Migration Workbench (ISMW), 299
Information governance, 300
InfoTypes, 355
Initial load, 53, 79
Integration test, 83
Interface, 84
Internal table, 120
ISO codes, 46, 329
Iteration, 41

J

Job run, 326
Jobs, 299, 304
Join, 486
 technique, 487

K

Key fields, 366
Key user authorization, 450

L

Landscape transformation, 536
Layout, 240
Layout structure, 252
Leading zeros, 495
Legacy data, 28, 142, 263, 297, 354, 543
Legacy data structure, 251
Legacy system, 149, 314, 354, 369, 497
Legacy System Migration Workbench → LSMW
Legacy values, 330
legal regulations, 89
Line functions, 397
Line items, 26, 30
List box, 265
Load system, 290
Load test data, 81

Loading, 48
Local Object Library, 303
Local template files, 370
Logical file name, 201
Logical path, 201
Logical system, 136–137
Logical Unit of Work, 140, 144
Long text, 226, 418
 flag, 226
Long texts, 262
Lookup value, 307
Loop, 122
LSMW, 47, 55, 70, 106, 145, 149, 173, 239, 487, 491, 493, 509, 512–513, 520
 action log, 159
 ALE-EDI Customizing, 212
 assigning files, 202
 attributes, 159
 authorization concept, 157
 authorization profiles, 157
 conversion log extension, 255
 conversion program, 255–257
 conversion rule, 150, 225, 229
 converting data, 207
 default setting, 180
 display variant, 180, 230
 equal rank, 163
 exporting projects, 233
 field length, 166
 field mapping, 225, 229, 267
 field type, 166
 fixed value, 183, 185–187
 global data definitions, 241
 global functions, 243
 global variable, 242
 identifying field value, 166
 IDoc inbound processing, 234
 import, 150
 import technique, 159, 223, 228
 importing data, 210
 importing projects, 234
 labels, 230
 main steps, 158
 my objects, 162
 object, 228, 262
 object attributes, 159, 228
 object overview, 74, 159, 213

LSMW (Cont.)
 object type, 159, 223, 228
 owner, 162
 periodic data transfer, 162
 processing times, 230, 241
 profile, 157
 project, 154, 509
 project documentation, 156
 reading, 150
 reading data, 203
 recording, 215
 recording function, 150
 required target structures, 170
 reusable rules, 243
 selection parameters, 203
 skip records, 246
 source field, 164, 173, 228
 source structure, 163, 202, 228, 485
 specifying files, 195
 structure relations, 229
 subordinate, 163
 subproject, 154, 509
 system-dependent file name, 163
 target field, 173
 target structure, 485
 technical field, 180
 translation, 183, 186–187
 translations, 185
 user menu, 159
 user-defined routine, 184–186

M

Manage Your Cloud Solution, 539
Mandatory fields, 318
Mandatory fields validation, 333
Manual maintenance, 168
Mapping, 37, 46, 58, 65, 73, 355
Mapping phase, 66, 73
Mapping sheets, 307
Mapping template, 74
Mapping templates, 316
Mass change, 398, 452
Mass data maintenance, 464
Mass data upload, 138
Mass uploads, 141

Master data, 16, 25, 378
 example, 26
 obsolete, 67–68
Master data migration, 118
Material documents, 108
Material master, 264
 descriptions, 272
Material valuation data, 453
Merge SAP systems, 519
Merging, 391
Message type, 162
Microsoft Access, 486
Migrate Your Data, 540
Migration, 43
 work package, 62
Migration account, 29, 33
Migration cockpit, 383, 392, 396, 427, 443
Migration object, 58, 66, 250, 380, 456, 542
Migration posting date, 429, 431
Migration process steps, 399
Migration project, 542
Migration project plan, 472
Migration reconciliation account, 447
Migration services, 297, 325
Migration step
 convert values, 407
Migration tasks, 376
Migration template, 360, 363
 accounting, 439
 copy data to, 370
 error, 364
 saving, 364
 structure, 360
Migration test plan, 69
Migration unit, 434, 436
Migration Workbench, 518–519
Missing target values, 412
Modification
 field contents, 484
Modify existing records, 460
Modify table records, 262
Modifying structures, 484
Multiple source structures, 248
MWB → Migration Workbench

N

Naming convention, 384
Naming procedure, 244
Navigation screen, 207
Nodata character, 182
Nodata sign, 271
Nonfatal errors, 405
Number format, 491

O

Object, 44
Object level, 432
OData, 355, 357, 546
OData call, 357
ODBC protocol, 297
Offsetting posting, 111
Offshore, 75
On-premise, 55
On-premise application, 354
Open Database Connectivity (ODBC) protocol, 290
Open purchase orders, 381
Opening balance sheet, 30
Opening balance sheet account, 30
Optional entry field, 38
Organizational structures, 455
Organizational unit, 521

P

P&L account, 31
Package size, 144
Paper-based mapping, 74
Partner, 162
Partner agreement, 162
Partner profile, 137
Perform data cleansing, 72
Perform productive data migration, 87
Phased roll-out, 54, 72, 78, 86, 264
Plan and organize data migration projects, 53
Plan test landscape, 83

Port
 file port, 234
 tRFC port, 235
Postal code, 312, 346
Posting documents, 26
Posting key, 34, 38, 111, 494
Posting logic, 445
Posting preview, 444
Postprocessing, 451
Preparation, 25
Price list, 454
Private cloud, 55
Process code, 137
Productive load and support, 61, 85
Productive load test, 77, 85
Profit center, 523
Program
 RBDAPP01, 139
 RCSBI010, 107
 RCSBI020, 107
 RCSBI030, 107
 RCSBI040, 107
 RFBIBL00, 107
 RFBIDE00, 106
 RFBIKR00, 106
 RFBISA00, 107, 216
 RM06BBIO, 108
 RM07MMBL, 108
 RMDATIND, 74
Program RM06EV47, 263
Project scope, 23
Public cloud, 55
Purchase order (PO), 250
Purchase requisitions, 108

Q

Quality checks, 538

R

Random number, 249
Rapid Data Migration, 89, 293, 296, 305
Rapid Data Migration with SAP Data Ser-
vices, 76

Rapid Database Migration, 528
Rapid Deployment Solutions, 529
Read data, 205
Reading data, 150
Realization, 65
Reconciliation, 298
Reconciliation account, 30
Reconciliation of migration results, 479
Record type, 163
Recorded transaction, 78
Recording, 104, 150, 223, 508, 511
Recording overview screen, 114
Regional options, 491
Relationship profile request, 314
Remote Function Call (RFC), 132, 339
Repeat validation, 403
Replace existing records, 462
Required entry fields, 36
Resetting the mapping, 325
REST, 357
Retained earnings account, 31
Reusable rules, 191
RFC server group, 140
Roll-out project, 452
Runtime, 161

S

SAKOMIG, 448
SAP Activate, 537
 methodology, 538
SAP Adaptive Server Enterprise (ASE),
 529
SAP application server, 198
SAP Basis Release, 152
SAP Best Practices, 320, 343, 354, 535
SAP Business ByDesign, 359
 navigation, 374
SAP Business Suite, 517, 541
SAP Business Warehouse, 68, 529
SAP BusinessObjects BI Launchpad, 340
SAP BusinessObjects Business Intelli-
gence (SAP BusinessObjects BI), 290,
 510
SAP BusinessObjects Web Intelligence,
 296, 300, 340

SAP BusinessObjects Web Intelligence
reports, 511
SAP Change and Transport Management,
 232
SAP change request, 232
SAP Cloud for Customer, 359
 navigation, 375
SAP Customer Relationship Management
 (SAP CRM), 134, 291, 294, 505
SAP Data Services, 46, 51, 60, 70, 89,
 291, 298–301, 304, 326, 510, 543
SAP Data Services Designer, 308, 316
SAP Data Services profiler, 311
SAP Enterprise Information Management
 (EIM), 289
SAP ERP, 142, 294, 535
SAP ERP 6.0, 152, 546
SAP ERP Human Capital Management
 (SAP ERP HCM), 297, 353
SAP Fiori, 533
SAP HANA, 528–529
SAP HANA Cloud Integration for process
 integration, 546
SAP home directory, 200
SAP Hybris, 545
SAP Information Steward, 90, 290
SAP Landscape Transformation, 518–519
 company code transfer, 519
 merge systems, 519
 Replication Server, 518
 Work Center, 518
SAP List Viewer (ALV) Grid Control, 274
SAP lookup tables, 344
SAP LT → SAP Landscape Transformation
SAP Master Data Governance, 518
SAP NetWeaver Application Server, 44
SAP Query, 88, 90
SAP R/2-SAP R/3 Migration Workbench,
 149
SAP Rapid Data Migration, 543
SAP Rapid Deployment solutions, 543
SAP S/4HANA, 533, 535, 538, 540
SAP S/4HANA cloud edition, 95
SAP S/4HANA, cloud marketing edition,
 545–546
SAP Service Marketplace, 152, 358
SAP SLO consulting, 68

SAP SuccessFactors adapter, 356
SAP SuccessFactors Employee Central,
 353, 356
SAP transport system, 232
SAP Utilities industry, 299
SAPinst, 531
Scripting language, 346
SEAMAP, 62
Search routine, 285
Segments, 132
Select query, 487
Selection button, 158
Selection parameter, 264
Selection screen parameter, 265
Sequence number, 175
Sequential file, 28, 44
Service Parts Management (SPM), 67
SFAPI, 357
Short field help text, 362
Simple file transfer protocol, 546
Simple Object Access Protocol (SOAP),
 357
Simulating, 37
Simulation log, 425
Single record view, 404
Single source structure, 199
Snapshot, 83
Software Provisioning Manager, 531
Software Update Manager, 531
 DMO, 531
Software-as-a-Service (SaaS), 55
Source data, 412, 542
Source data field, 252, 285
Source fields, 166–167, 181, 223
Source structure, 163, 223, 268, 485
Source structure definition, 272
Source system, 295, 308, 355, 368
 preparation, 431
SOX, 503
Staffing bottlenecks, 23
Standard batch input, 509
Standard batch input program, 50, 104–
 105, 507, 512
Standard interface, 49, 151
Standard Operating Procedures, 89
Storage data, 274
Structure relations, 170, 251, 272, 487

Structured variable, 260
Subledger, 31
Subledger accounting, 30
Subordinate source structure, 249
Subroutine technique, 120
Suffix, 183
SUM → Software Update Manager
SWPM → Software Provisioning Manager
System conversion, 535
System Landscape Optimization (SLO), 55, 68
System Landscape Transformation, 68
System optimization, 54

T

Table CEPC, 523
Table flag, 262
Table-like file, 44
Target data field, 252
Target environment, 315
Target field, 322
Target field name, 390
Target structure, 170
Target structure field, 260
TDMS Manager, 521
TDMS → Test Data Migration Server
Template download list, 379
Temporary user accounts, 473
Test complete data migration, 85
Test data, 143
Test Data Migration Server, 518, 521
 data scrambling, 522
Test environments, 475
Test migration, 387, 430, 474
Test phase, 80
Test plan for data migration, 86
Test run, 291
Test system, 423
Test Your Processes, 540
Testing, 41
Text file, 39, 44
Text key, 227
Thousand separator, 491

Throughput, 144
Transaction, 110, 135
 AL11, 200
 BD87, 342
 EMIGALL, 299
 FB01, 109, 118
 FK01, 99
 FS01, 216
 SM35, 211
 WE02, 342
 WE05, 342
 WE19, 147
 XD01, 246, 282
 XD02, 246
Transaction code, 245
Transaction data, 16, 26, 29, 41, 378
 closed, 67
 example, 27
Transaction Recorder, 113
Transaction recorder, 97, 104, 109, 215
Transaction RFC (tRFC), 132
Transactional data, 411
Transfer (MOVE), 182
Transformation, 38, 46
Translation, 46–47, 497–498
Translation tables, 345
Translation type, 189
Translations, 243
Transport system, 232
Type definition, 277

U

Unit of measure (UoM), 458
Unused source values, 412
Upgrading existing systems, 54
Upload, 48
User acceptance tests, 475
User administration, 472
User Menu button, 255
User profile, 489
User Training and Onboarding, 540
User-acceptance test, 85
User-defined routines, 243, 259, 275
User-friendliness, 515

V

Validate data, 90
Validation, 76
 after load, 88
 against lookup tables, 333
 before load, 87
 data errors, 405
 enhancements, 350
 log, 401
 screen, 468
Value conversion, 410, 481
 export and import, 392
 import, 415
Value help, 266
Value mapping, 325, 331, 369, 412–414
Variables, 122
Vendor master, 106
View Solution Scope, 539
Visualization, 296
VLOOKUP, 501
Volume performance by parallelization, 477
Volume test, 476

W

Web Services, 291
Work center, 374
Work center views, 364
Work item, 212
Work package, 62
Workflow customizing, 136
Worklist, 408, 450

X

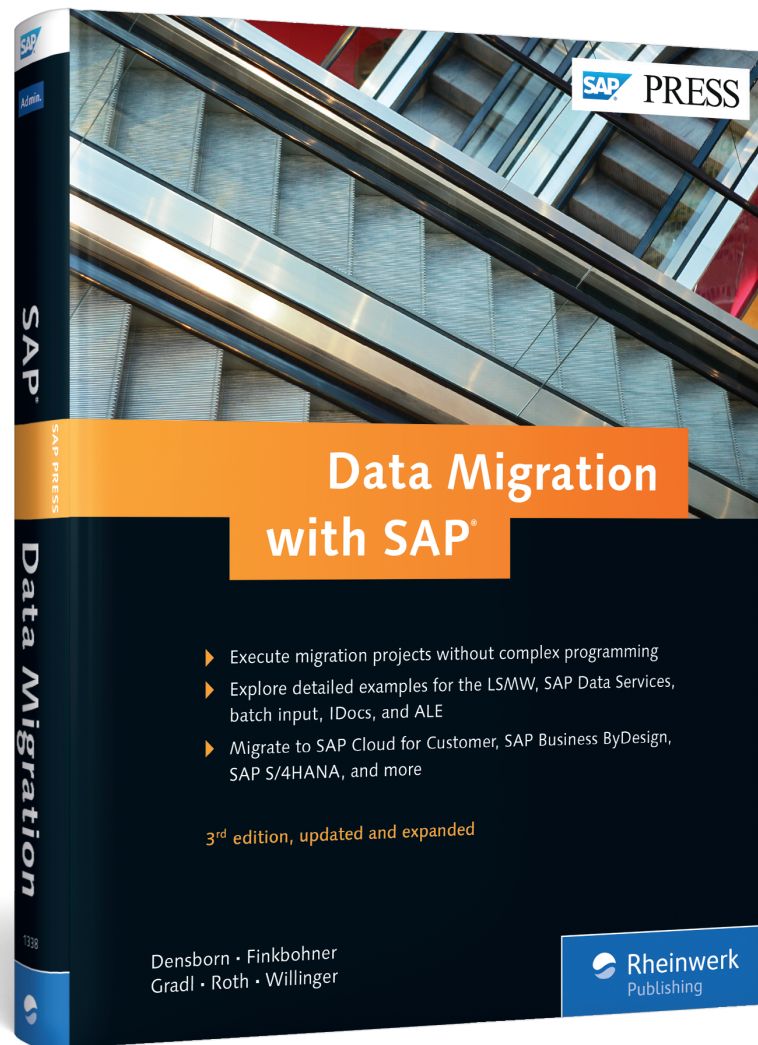
X FIELD, 185
XIF interface, 134

Y

YYYYMMDD, 47, 166

Z

ZIP files, 385



Frank Densborn, Frank Finkbohner, Johann Gradl, Michael Roth, and Michael Willinger

Data Migration with SAP

563 Pages, 2016, \$79.95

ISBN 978-1-4932-1338-2

 www.sap-press.com/4019



Frank Densborn is a product manager for SAP Rapid Deployment solutions in the technology area, focusing on data migration and cloud integration. He joined SAP in 2004 and has held various roles in development, education, support, and product management.



Frank Finkbohner joined SAP in 1999 as a development consultant, with a focus on planning and implementing legacy data migration. He is now the project manager for building rapid-deployment solutions in the areas of enterprise information management, data migration, and data loads.



Johann Gradl joined SAP in 1993 and was a driving force behind the development of the Legacy System Migration Workbench (LSMW), and subsequently became responsible for maintenance of SAP CRM.



Michael Roth joined SAP in 1994, where he worked in various positions in service and development. Today, he is a member of the Platinum Engineering Team for SAP Business ByDesign.



Michael Willinger works at SAP SE in the area of system landscape optimization, where he focuses on migration projects that go beyond the standard SAP ERP system.

We hope you have enjoyed this reading sample. You may recommend or pass it on to others, but only in its entirety, including all pages. This reading sample and all its parts are protected by copyright law. All usage and exploitation rights are reserved by the author and the publisher.