Data Migration Phases for an SAP S/4HANA Project

ou can use different process models and methods for SAP S/4HANA implementation projects. The current process model or framework for implementing SAP software is SAP Activate, which contains roadmaps developed by SAP for the implementation of your products and is constantly adapted and further developed.

In this blog post, we explain which project phases exist in a data migration project and how they are integrated into the standard project phases of an SAP S/4HANA implementation project.

Roadmap Viewer in SAP Activate: For more information about SAP Activate, you should visit the Roadmap Viewer web page at *https://go.support.sap.com/roadmapviewer*.

SAP Activate is divided into six phases:

- 1. Discover
- 2. Prepare
- 3. Explore
- 4. Realize
- 5. Deploy
- 6. Run

The best way to set up a data migration project is to run it as a subproject. The tasks of a data migration project differ from those of a product launch using SAP Activate. These tasks are distributed across seven data migration phases:

- 1. Data analysis
- 2. Data cleansing
- 3. Mapping
- 4. Implementation
- 5. Tests
 - A. Functional tests
 - B. Productive load test (PLT)
- 6. Validation
- 7. Productive load

This figure shows a representation of the individual phases.



Unlike SAP Activate phases, data migration phases are not strictly sequential but instead are iterative and overlapping. Thus, findings from a downstream phase intentionally affect the preceding phases. This is necessary so you react flexibly if errors arise. In the following sections, we'll go into more detail about the individual data migration phases.

Data Analysis

The analysis phase is usually started within the Prepare phase, but it does not necessarily have to start with this phase. You can just as well start the analysis halfway through the Prepare phase because, by then, the processes to be implemented are known, and some of them have often already been fully described. This approach makes it easier for the data migration analysis to identify the business objects, data migration objects, and source systems.

Business Objects versus Data Migration Objects: A *business object* is an individual data object that is needed to model a business process, such as a material, a customer, or a purchase order. A *data migration object* (or simply a *migration object*) is usually a business object or a part of one. A migration object is used to migrate a business object. Business objects sometimes must be divided into smaller units for technical reasons, perhaps having different data sources or data migration object. One example is the business object product, which is divided into multiple migration objects in SAP S/4HANA.

Master Data versus Transaction Data: *Master data* is data that is stable and used as the basis for business objects. Master data changes only slightly or not at all over a specific period. Known master data objects include product, customer, supplier, bank, and bill of materials (BOM). *Transaction data* is data that is volatile and subject to continuous change. Common transaction data objects include stock levels, account data, all types of orders, and documents.

We recommend that you start the data analysis process as early as possible as it lays the foundation for all other activities.

Data Cleansing

Data cleansing is one of the most underestimated activities within a data migration project. If possible, data cleansing should always be performed in the source system. Under certain circumstances, however, it can also be done in the respective data transfer tool. Systematic errors in your source data can sometimes be converted to clean load data using conversion rules.

You should also use insights from machine validations to clean up your data in the source system. The cleaner the data in your source systems, the cleaner the data in your target systems.

Data cleansing should start as early as possible in your migration project. Ideally, you already have defined processes in your source systems that permanently check the data for errors and clean them up. You should closely involve these processes, or the teams tasked with them, in the data migration project. Likewise, you must identify existing data cleansing rules and match them with the conversion rules. The findings on the quality of the source data from the initial data transfer tests are important for further data cleansing processes.

The data cleansing phase should be completed, at the latest, with the last productive load test. This is the only way to ensure that the findings of the last test can be adopted one-to-one at go-live.

Mapping

Once the data migration objects have been identified, the mapping phase can start. In the mapping phase, the structures and fields of your source system data to be migrated are mapped to the structures and fields of the data migration interfaces and programs of the target system. Since this task is usually done on paper, this step is often referred to as *paper mapping*. The mapping phase often overlaps with the analysis phase. However, you should not wait for the analysis to complete; you should start mapping early. A good time for mapping is when the first master data objects have been identified and the configuration for maintaining these objects is largely complete.

Configuration documents (formerly also referred to as *business blueprints*) usually already contain hints for conversion rules. The mapping phase can extend into the testing phase. Often, insights emerge from loading the initial test data that have a strong influence on the mapping and conversion rules. We therefore advise you start migration activities as early as possible so that sufficient time can be scheduled for testing the data migration and the post-processing resulting from these tests.

Implementation

Implementation, including your first small functional tests of the data extraction and data migration programs, can be started quite early in the realize phase of the project. The best time to download the first test data is when most of your conversion rules have been established so that the initial load can be pulled from the source systems.

Testing the Data Migration

Testing is the be-all and end-all of any data migration, and you should start the first functional tests as early as possible. You can't run enough tests. The more complex a data transfer is and the more target systems, source systems, data migration objects, and conversion rules involved, the more important performing tests becomes.

There is no substitute for testing—except more testing! Test the entire data migration process and cutover at least once before go-live. However, our experience shows that one productive load test alone is often insufficient. We recommend at least two productive load tests (load tests under go-live conditions and in systems that technically correspond to the productive system) for a positive effect on data quality and adherence to scheduling.

Testing Reduces Costs: We can reduce this discussion into a simple formula: The *higher* the *number* of tests and the *higher* the *quality* of the loaded data, the *lower* the *rework* in the target system shortly before or after go-live and the *lower* the *overall* costs.

For testing, especially for multiple tests of a data migration, an early planned data migration testing strategy is required. You must carefully plan the systems, timing, and sequence of testing and clean up or recover your test systems. Define an appropriate backup-restore concept for your test systems and determine the duration required for a backup and restore. Taking what are called *database snapshots* is often a matter of choice. In this context, you can create snapshots of a database at specified times and can quickly reset it.

Validation

Validating the loaded data is essential for the quality of your processes in the target system. Only high-quality master data and transaction data guarantee the clean operation of your processes. In some industries, such as the pharmaceutical industry, such validations are even mandatory and must be documented. Without an appropriate validation of all data in the target system, for example, you cannot receive an *Food and Drug Administration (FDA)* certificate, and without this certification, you won't be allowed to sell goods on the US market. Thus, don't neglect the validation of your data.

The best time to start defining your validation rules is when determining the conversion rules. A machine-based validation process allows you, for example, to use the results and any identified erroneous source data for the data cleansing process.

You should consider early on how you can validate your data later, during integration testing and go-live.

Two approaches are available for validating your data during a data migration:

- Validation before the data migration: In this case, the source data is checked for syntactic correctness (e.g., length and data type) and semantic correctness (e.g., meaningful values with reference to the business process configuration). Conversions that are made subsequently will be ignored.
- Validation after the data migration: In this context, you check the number and completeness of data records in the target system by asking yourself a few questions: Have all values been converted correctly? Have the corresponding fields and structures been filled correctly? Can the business processes, based on the loaded data, be executed without errors?

A combination of both approaches is frequently used.

Productive Load

The productive load phase is the most critical one and forms the basis for the later go-live decision. If you have done everything correctly in the previous phases, tested sufficiently, and planned your cutover cleanly and above all in detail, you can start this phase with a clear conscience. However, often small, inconspicuous things can mess up your cutover.

Let's consider a few examples from our previous work as data migration consultants:

- Works councils, trade unions, working time legislation, and authorities: Works councils, trade unions or the trade supervisory office should be involved in planning your implementation project. Night work, weekend work and overtime must be arranged with or registered with the appropriate groups and offices. These processes require lead times and cannot be rushed. Otherwise, you may find that none or only a few employees are available for the final inventory at the go-live weekend.
- **Public holidays, time zones, and calendar weeks:** Every country has specific periods where you can expect reduced productivity or the absence of project members, as well as of their subcontractors and other companies. In Germany, you can often write off May and June as "useless" months for your project. Depending on the state in Germany, the months of May and June have the highest density of public holidays, bridging days and vacation days (e.g., May Day, Ascension Day, Corpus Christi, and Pentecost). In France, the whole summer is a wash, as is the US between Thanksgiving and Christmas or China around Chinese New Year and

Golden Week. Thus, if possible, you should not schedule any elaborate work, tests, or go-lives within or shortly after those periods.

If you plan a data migration across time zones with multiple teams, you should make sure that the different time zones are also included in your planning. Starting a data migration, for example, at 10 a.m. CET, assumes that a team member in the San Francisco data center has also been scheduled at the dead of night and is available to provide the data. Pay attention to the different handling of daylight-saving time and standard time and their different changeover dates in different countries.

Another curiosity is the various ways to calculate the beginning of the first calendar week of a year. In Europe, calculation according to International Organization for Standardization (ISO) 8601 is common. Here, the first calendar week is the week that contains at least 4 days in the new year. Other countries, such as the US, calculate the first calendar week as the week which includes January 1. As a result, there may even be a calendar week 54 in the US, which is not even possible with DIN 1355/ISO 8601. Depending on the calculation method, the calendar weeks within a year may therefore differ from country to country. Confirming deadlines based on calendar weeks can be rather dangerous if you need to adhere to your schedule. For example, 2022 was a year in which calendar weeks differed between the US and ISO-8601 countries.

- **Communication of downtime:** Communicate downtime with your suppliers and customers so that trucks aren't parked in front of your facility during that downtime, waiting to be loaded or unloaded.
- Check and adjust automatic system processes: Interfaces to and from existing source systems or workflows in the target system should be disabled during the data migration process. In this way, you avoid falsified data, a system breakdown due to bulk generation of workflows, or the flooding of inboxes of the people in charge of workflow entries. Another problem that is frequently underestimated involves unplanned automatic backups. A backup process suddenly starting in a source system during the extraction of data its or in the target system during the loading of data can delay your schedule by more than 12 hours. In other words, if you notice only the next morning that the overnight data extraction or data transfer was too slow or was canceled, you must start all over again.
- **Upgrades and hotfixes in cloud environments:** In cloud environments, such as SAP S/4HANA Cloud Public Edition, upgrades and hotfixes are deployed to the respective systems at specified time slots (in what is called a *blue-green deployment*). Your data migration activities such as extracting data, loading data, etc. should be scheduled so that they do not conflict with downtime. Although these dates and time slots should be known to every customer, unfortunately a common problem is that customers schedule their activities in parallel with these time slots. In SAP S/4HANA Cloud Public Edition, you should definitely observe the following SAP Notes:
 - SAP Note 3024158: Blue-Green Deployment for Hotfix Collection for RISE with SAP S/4HANA Cloud (for production systems)

 SAP Note 3042314: Hotfix Collection Deployment for RISE with SAP S/4HANA Cloud (for non-production systems)

This post should have provided you with a better understanding of the phases of an SAP S/4HANA implementation project. Wondering if you should utilize SAP Activate when working on an SAP S/4HANA project? This post can help shed more light.