CORE®

MANAGING A SUCCESSFUL GO-LIVE

Risk-minimizing Rollout Management in the Transformation of Core Banking Systems

October 2014
White Paper

Risk-Minimizing Rollout Management in the Transformation of Core Banking Systems

Large IT projects are accompanied by higher levels of risk and core banking transformations increase the complexity further

The realization of IT projects is beset by high risks, due, on the one hand, to the low maturity level and the accompanying deficits in planning and management in the data processing industry, and on the other hand, to the short half-lives of applied technologies and the know-how of the employees involved. Large IT programs display a decidedly higher level of risk, as disproportionately high risks build up through vague, changing demands from the business departments and the comparatively low organizational quality within the field of IT engineering.

Core banking transformations are large, highly complex IT programs of long duration that are essential to a company. Risk-minimizing measures are therefore highly relevant, not only to ensure that a company remains in charge of the schedule, carries out planned functions, and remains within the project budget, but also for a company to recognize possibly dangerous effects early on and to deal preventatively with them.

Profound specialized knowledge and experience in dealing with core banking transformations are available only to a limited extent on today's market, and the number of successful national and international projects of this kind is also limited. Very few core banking transformations have been carried out so far in German-speaking areas, and of these only a limited number have been implemented successfully within the defined parameters.

The consistent embedding of projects and sub-projects in the overall responsibility of the program is one of the factors for success of a large transformation

This White Paper provides an approach for carrying out large and successful IT core banking transformations based on the experience gathered by CORE in previous projects. The essential factors for successfully overcoming the hurdles posed by *time, budget and functions* are explained using examples from the project management methods applied in these projects.

Regarding core banking transformations, this White Paper is useful to decision-makers and program managers by

- serving as a guideline and basis for decision-making
- providing examples for effective reporting, and
- helping companies to achieve their goals within the predefined framework conditions.

With the help of a concrete example of a core banking transformation project, this document describes the essential points of the most important phases and provides support methods.

The visualization of all linkages is an essential management element

The *big picture* of a core banking transformation plays an extremely important role: The embedding of the projects and sub-projects in the overall responsibility of the program underlines the dependencies created by the schedule and content. An essential insight derived from large transformation projects is that the necessary facts relevant to decision-making must be provided to the program managers. It must be made clear to the (sub-) projects and (sub-) project managers which facts are relevant and why, and why this information must be collected for reporting, as explained in the following section.

Using the example of a work breakdown structure and the master plan that derives from it, we will describe those planning instruments that are essential tools for the structuring and monitoring of the program and that illustrate the most important decision-making points.

A tight interlinking of business and IT areas enables a structured Go-live release

Large-scale transformation projects are often marked by a high level of emotional engagement on the part of the employees involved in the project. This is due to the fact that a new start often means leaving behind familiar territory. In cases where several companies are represented in one corporation, sometimes in decentralized organizational structures, it is possible to foresee that conflicts may reach all the way up to the senior management level. A dispersed, decentralized organizational structure can be flexible in its reaction to local demands, but that decentralization also makes the management of the whole more complex. At first glance, these are not new insights, but as is often observed during large projects, it is not uncommon for decision-makers in an organization to lack security in their management style. Using the example of a test procedure in a decentralized test organization, we will present the most important points and show how this challenge can be dealt with in a constructive manner.

The organization of the necessary releases of the attained results is the last hurdle before the Go-live. Without this explicit acceptance of the responsibility for the end product by all stakeholders, a Go-live decision in IT is vulnerable. It becomes clear by this essential step in the process, if not earlier, whether the program was able to turn those affected by the program into participants and to achieve the pre-defined goals. The progress control of the release process all the way to the final physical acceptance signature is exemplified using a dashboard.

We have observed that the business departments and the IT department use very different criteria for measuring the success of a core banking transformation. Oftentimes the IT departments are organized in a more structured way for IT-based changes, as release changes and software updates belong to an IT department's regular business and as success in this field is measured based on technical data such as system downtimes and the number of flawed migrating data sets. Other questions are relevant for the business departments, namely ones that significantly influence daily business operations and correspondingly cause additional work and costs. Foreseeing and planning for these criteria is critical in the first weeks after the Go-live. The success of a large transformation will be clearly illustrated using the example of the development of the occurrence of customer complaints and production deviations.

Initial Situation and Goal

CORE supports banks in the introduction of new core banking systems. Using a sample project, we will present the experience we have gathered in the field of these large, complex transformations.

The sample project consists of the core banking transformation of a bank that has decided to switch to a SAP core banking platform.

The Initial situation is a core banking transformation with the goal of a transfer to an SAP platform

The initial situation of the bank was mainly shaped by three main points that created large amounts of pressure within the bank to act regarding payment transactions and current accounts:

Main points of the initial situation:

- Consequences of a fusion in the provider environment
- Functional and qualitative deficits in foreign payment transactions
- High costs in the areas of accounts and payment transactions in daily operations

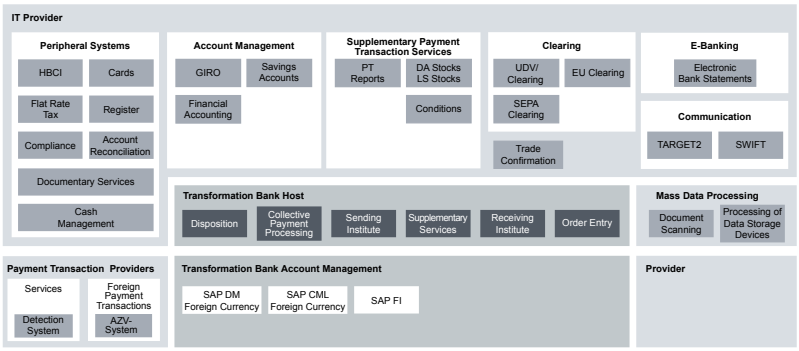

Image 1 Functional application view prior to the transformation.

The IT landscape shows a complex provider and application structure with high operative costs and functional deficits

The account management systems including the necessary peripheral systems and the relevant supplementary services for payment transactions were partially in the hands of different providers. In part, they were distributed over several units within the organization outside of the contractor's department, units that had been created through earlier acquisitions. In a first step, the management of foreign payment transactions had been given to a new provider while the processing of bank data (such as collective orders) continued to run on an internal cost-intensive host system. The foreign currency shares of the accounts management section had by this point been transferred to the end-goal systems SAP DM and SAP CML and therefore the daily operations in this section ran in a more stable fashion than prior to the changeover.

The following factors were essential to the transition of the core banking system landscape to the SAP goal scenario:

Reasons for Migration:

- The end-of-lifecycle of the current system of the core banking provider forced a changeover to a different core banking system.
- The outdated technology and business concepts created high operative costs.

- A heterogeneous provider structure existed in different product portfolios for the same or similar products.
- Critical business processes were dependent on external know-how to an extremely high degree.

Compared to other competitors in the industry, all of the points above contributed to a high Cost-Income-Ratio (CIR) for the bank in the sample project.

Therefore the goal of the core banking transformation program was to move to a more stable application landscape in several migration steps. Furthermore, consolidation was to lead to fewer providers, such as in the area of cards, and this together with the medium-term dissolution of inherited waste in the system was to lower operative costs and regain control over accounts management and core data, two areas of critical importance to the corporation (see Image 2: High-Level Migration Plan).

The target scenario is an application landscape equipped for the future with reduced operative costs and fewer providers as well as the insourcing of critical know-how

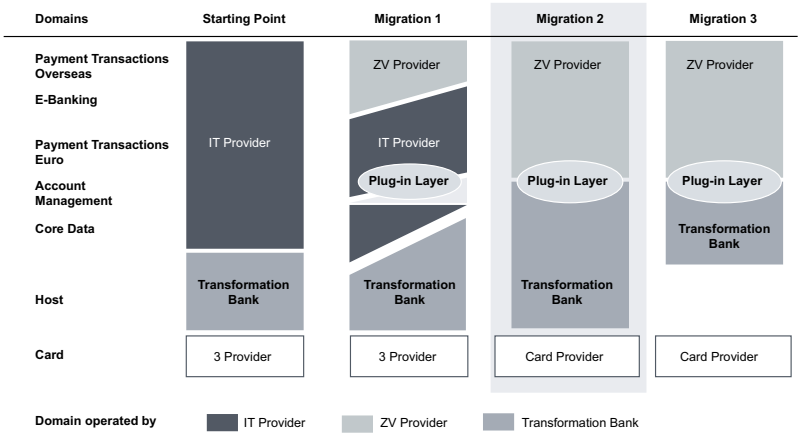

Image 2: High-Level Migration Plan

The targeted end goal (see Image 3: Targeted End Goal) that was to be reached in three migration steps not only had to do justice to the external factors previously discussed, but also fulfill the following requirements:

Requirements of the End Goal:

- Standardization of the IT architecture through introduction of Packaged Software Standards – Standards based on SAP DM as well as introduction of a Plug-in Layer to connect the core bank with the peripheral systems and providers
- Introduction of standardized products (debit and credit cards) and reduction of provider diversity
- Cost reduction through switch to new service providers, infrastructure innovation and use of scale effects through packaged software
- Establishing core know-how in-house
- Meeting the start date established in contracts

The project duration of four years and the allocation of 75% of the annual IT project budget reflect the significance of the project

With its approximately four-year duration, this transformation was the largest IT modernization program ever implemented in the bank and consumed up to 75 percent of the entire annual IT project budget of the bank. The entire program was characterized by a high level of complexity. During some periods, over 400 employees were occupied with the essential implementation phases and project teams were at work in different company locations in different cities, while reorganization processes were being carried out in the bank during the duration of the project.

External framework conditions define the date of migration

The framework conditions, some of which were defined by external actors, as well as the established migration date formed the fixed requirements that were never called into question throughout the duration of the program. On the one hand, this exacerbated the conflict between scope, quality, and the deadline, while, on the other hand, these fixed requirements provided the (sub-) project leaders with a high degree of security in their planning.

During the intensive *Migration 2* phase, the essential program committees met on a weekly basis so that decision-making timeframes were extremely reduced.

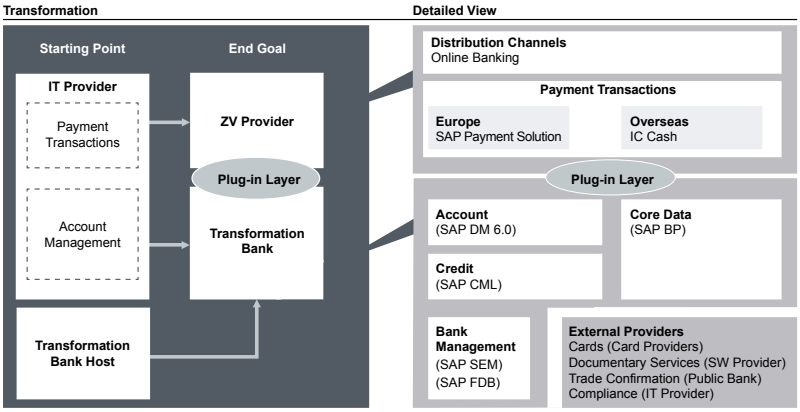

Image 3: End Goal

Procedure Model

The procedure model used in the program was based on the CORE Transformation Framework. This framework encompasses the entire lifecycle of a transformation. In the project discussed here, CORE support began once the project was already running (Migration 2), whereby an emphasis was placed on review and relaunch actions during the beginning phase.

The CORE Transformation Framework divides a transformation into three main phases

The following three phases are of varying duration as they are defined by the program schedule (see Image 4: Overview of Essential Management Elements):

Phase 1 – Modeling

- Analysis of the program situation
- Examination of feasibility of the desired end goal (function/quality, time and budget)
- Design of a new implementation model, relaunch of the program and commitment by all stakeholders to new implementation model

Phase 2 – Preparation

- Consolidation, reduction and parallelization of main work packages in one master plan
- Implementation of a resilient progress review until the Go-live in relation to work package statistical analysis, budget consumption, test case coverage and test progress, trouble shooting and releases
- Force-fitted interlocking of business and IT departments and implementation of cross-departmental projects
- Illustration of the effects of core banking transformation on business processes and IT

Phase 3 – Execution

- Reduction of decision-making timeframes and focusing of project participants with decision-making powers
- Expansion of basis for decision-making by increasing the necessary transparency
- Continuous monitoring and reporting of program's progress
- Structured deviation escalation and adjustment across all project hierarchies
- Close involvement of line organization and business departments through inclusion in test management
- Securing migration through staggered fallback scenarios
- Implementation of a stringent release process with inclusion of participating stakeholders
- Preparation and moderation of point-of-no-return decision of all stakeholders and participants on basis of collectively defined criteria

Customized reports are available for each project phase

For each phase of the project, crucial reports were defined either prior to or during the project's implementation. Some of these reports were regularly updated throughout the project, while others were created only for a certain amount of time. All reports shown in the image below are explained in detail in the corresponding descriptions of the individual phases. The *Production* phase included later in the process (see Image 4: Overview of the Essential Management Elements) is not an element of the transformation and therefore is not scrutinized further in this section.

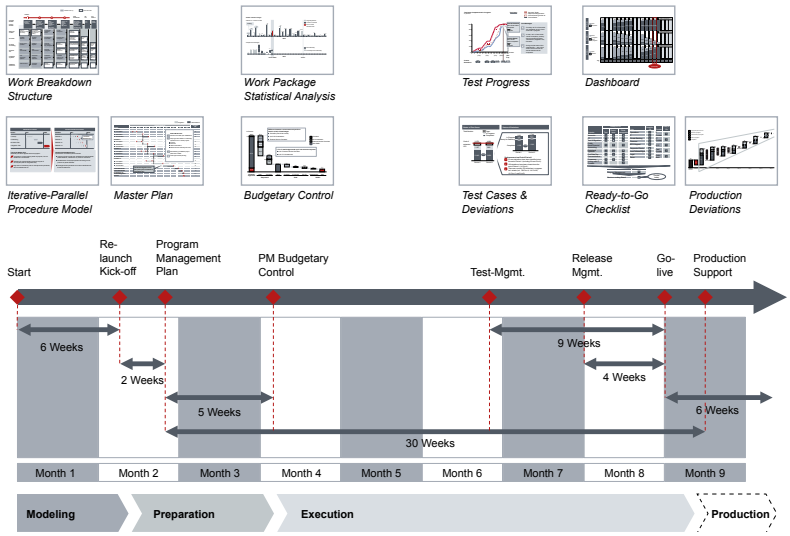

Image 4: Overview of the Essential Management Elements

Phase 1: Modeling

The point of departure for all of the reports discussed below was the newly implemented and approved work breakdown structure (also see Image 5: Work Breakdown Structure of the Master Plan). Together with the newly developed procedure in the field of test management, the work breakdown structure forms the basis for the master plan.

The entire course of the project was measured against the master plan. The master plan defined the benchmark for the success of the project at the end of Migration 2.

The modeling of the situation occurs in the form of a work breakdown structure

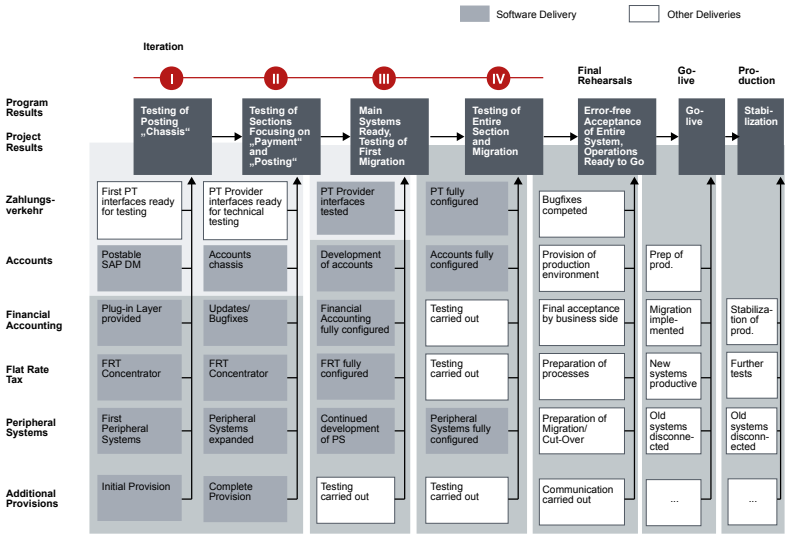

Image 5: Work Breakdown Structure of the Master Plan

One essential element of the work breakdown structure is the depiction of the specialized contents of the projects in the individual steps on the way to the Go-live. This form of illustration makes it possible to portray parallel tasks and content-related dependencies in a transparent and more aggregated manner.

The work breakdown structure presents the specialized contents of the projects step-by-step on the way to the Go-live

In an MS Project Plan, additional structural elements, such as individual phases (Plan – Build – System Test – Integration Test – Acceptance Test), would unnecessarily increase the complexity. In addition, this level of abstraction makes it possible to report to the individual projects within the master plan using specialized project planning tools.

As a result, it was possible to synchronize all projects according to the parallel process needed to achieve the targeted deadline. The content-based process of the entire program was transparent for all project managers.

The iterative-parallel procedure model is an alternative to the waterfall model

In order to meet the contractually obligatory Go-live deadline, it was necessary to increasingly parallelize actions that had until then been planned sequentially (see Image 6: Iterative-Parallel Procedure Model). On the one hand, this procedure increased the complexity of the *critical path* in the project plan, thereby increasing the entire risk for the program, but on the other hand, it was the only option for achieving the highest quality possible by the deadline.

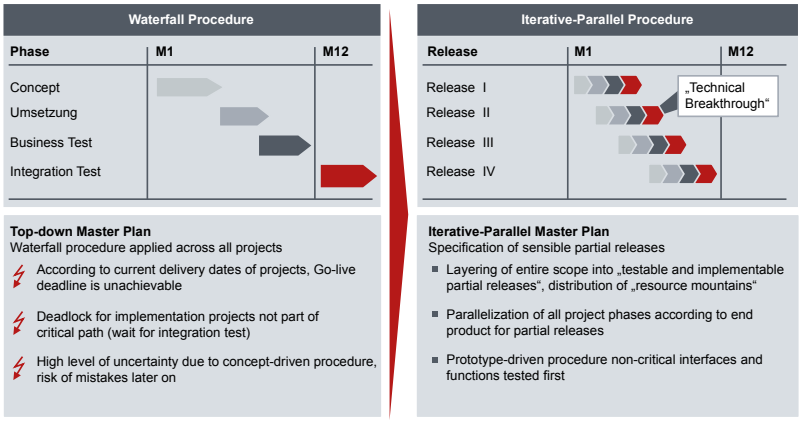

Image 6: Iterative-Parallel Procedure Model

Phase 2 – Preparation

In order to depict the consequences for all projects within the core banking transformation program on a timeline, the information from the work breakdown structure and the procedure were transferred to a classical master plan (see Image 7: Master Plan).

The problem-solving approach presented in the work breakdown structure is transferred to a Master Plan

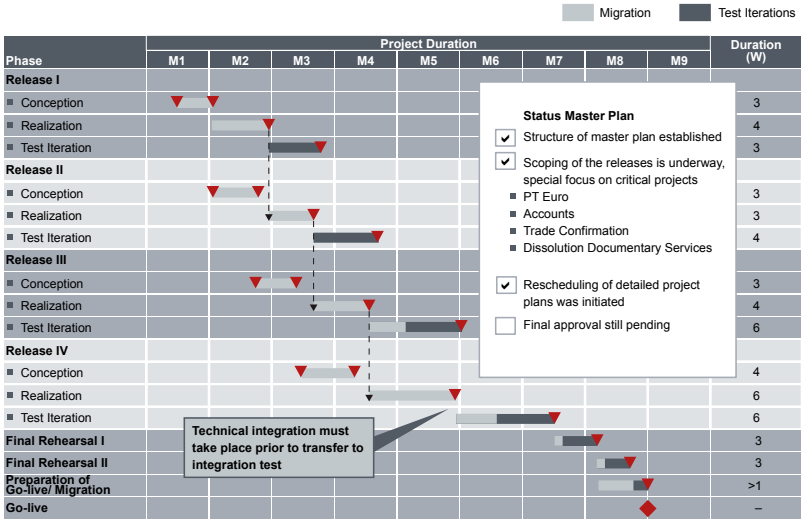

Image 7: Master Plan

Test iterations set the pace for incremental implementation

Of crucial importance was the fact that during the planning phase, the test iterations were established first, as these were to set the pace for the entire program. All test iterations build on one another and thereby contribute incrementally to the entire scope of the test. This leads to a transparent definition of the *critical path* and the delivery date for the corresponding test iteration for all projects.

Test orchestration was centralized while test implementation was decentralized

The early definition of test procedures transferred large amounts of responsibility for the testing process to the individual projects, while the general steering of the test management remained centrally organized. This also meant that the planning power regarding resource planning remained in the hands of the individual projects, which functioned as an additional trust-building measure in the sense of a change-management approach. The contradiction between centralized control and local planning was thereby successfully bridged.

The prompt inclusion of the contracting business side in the entire management of the program, which was triggered after the new start of the program by CORE's analysis in Phase 1 – *Modeling*, ensured that all essential stakeholders were actively involved in the shaping of the program.

Phase 3 – Execution

The definition of the master plan and the high level of parallelization of the project phases across all projects in the program required that management elements be identified in order to recognize and correct any deviations from this planning at an early stage.

Implementation was closely monitored on the basis of corresponding reports

The management element defined at the program level for the entire duration of the implementation phase was the **work package statistical analysis**.

The work package statistical analysis shows the deviations from the planned delivery progress

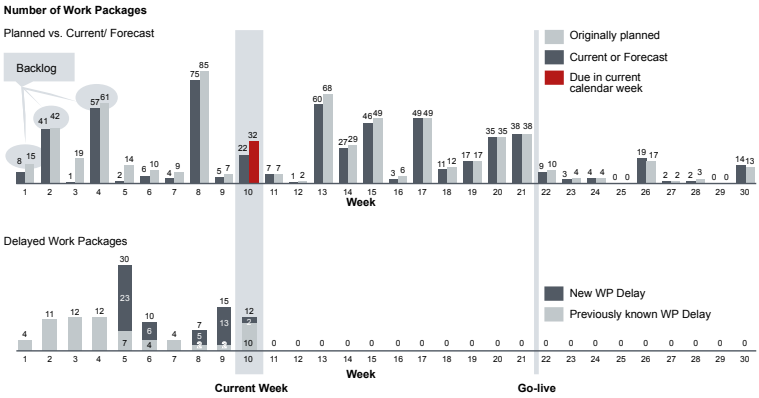

Image 8: Work Package Statistical Analysis

The goal of the **work package statistical analysis** is to illustrate the goal attainment or the backlog in regard to the completion of work packages in relation to the baseline for the program management team, the PMO (Program Management Office) and for the steering committee.

The **point of departure** is the baseline defined at the beginning of Phase 3 – *Execution* for all projects in the program plan. Changes to the baseline are made exclusively through the PMO via a previously established process, as the increased complexity of the *critical path means* that small changes in one project can endanger the plan as a whole.

The work package statistical analysis

- is a weekly evaluation of the completion of work packages according to the work plan,
- shows the number of work packages due to be completed according to the plan and the number of current and forecast work packages, respectively, per calendar week,
- provides a color-coded number of the work packages due in the current calendar week, and
- transparently shows the number of already postponed work packages as *known delay* and the number of delayed work packages as *new delay*.

The backlogs occurring in the upper part of the graphic – depicted as the difference between the bar showing the original plan and the bar showing the current forecast – partially derive from the delays shown at the bottom. As it would overly increase the complexity to denote work packages that have been postponed multiple times or to show when which delayed work package was completed, the detailed analysis must occur in the projects affected by the delay.

In an ideal situation, both bars at the top of such a graphic are of the same height and no delays are displayed at the bottom. The input consists of the work package planning and the work plans for the projects. The prerequisite for this is the availability of all work plans and a binding baseline.

Lessons Learned: The statistical analysis of the work packages makes it possible to quantify the deviations between the plan and the current situation as well as the number of postponed work packages. This analysis does not provide any qualitative information; that type of information is provided in the defined (such as weekly) reporting of the project. Although the effort per work package was comparable within the sub-projects in the sample project described here, there were significant differences in the amount of effort between the projects, differences that had to be taken into account in the analysis. Yet the quantified feeling for the situation sufficed in order to communicate an impression of the progress made in the project. It is worth pointing out that the tools presented here were able to explain the reasons for an accumulation of work packages due at specific points in time. This contributed in large part to the credibility of the approach. Although detailed reports were used intermittently, they did not provide more added value when compared with the less detailed versions.

The qualitative evaluation of work progress occurs in the classical cycle of project reporting

The second crucial management element defined at the program level, aside from the purely effort-based progress review, was **budgetary control**. Its goal is the presentation of the current budget within the program as a whole. The addressees include the program managers and the PMO as well as the steering committee for budget decisions.

The updated budget for the entire program is monitored using a waterfall chart

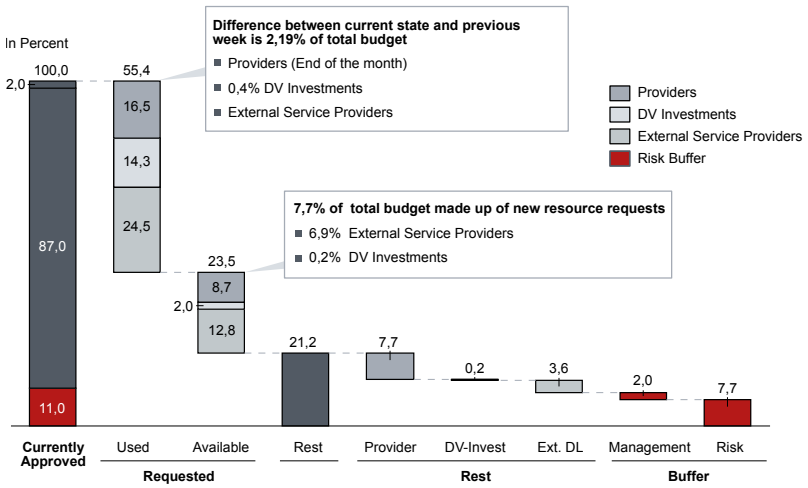

Image 9: Waterfall Chart for Budgetary Control

The **point of departure** is the budget defined and approved at the beginning of Phase 3 – *Execution*, including the available risk buffer. In this project it was necessary to distinguish between pure provider costs (which were mostly based on contracts lasting more than six months with precisely defined billing dates and therefore not influenced directly by program decisions), costs for external service providers in the project (consulting, conceptualizing, implementation) as well as the so-called DV investments (such as hardware and infrastructure costs).

The Waterfall Chart for budgetary control

- is a diagram, updated on a weekly basis, showing the actual budget of the entire program compared to the approved budget,
- depicts the budget in lists according to the dimensions *Requested, Rest* (planned budget – requested budget) and *Buffer*,
- divides the requested budget according to *Used* and what is *Available* in the projects, and

- distributes the budget across the categories *Provider*, *DV Investments* and *External Service Providers*.

In an ideal situation, the budget in the risk-buffer would not be transferred (not even partially) into projects and requested; furthermore, no supplementary budget would be necessary as all possible risks would be covered by the buffer. The input for the budgetary control report consists of the current state of the budget consumption by external parties (time recording) as well as planned invoices from providers and for DV investments. Prerequisites include current time recordings in all projects, the known commissions to providers, or completed or planned DV investments not yet included in the calculations.

Bank management systems are not suited to constant budget monitoring in complex programs

Lessons Learned: Using time recording procedures, updated weekly, as the basis for the used external budget, generally leads to acceptable results for the previous time periods. Significant errors occur when the general resource planning of the projects is not known (such as the sudden increase in resources due to the start of implementation, etc.). In most cases the data available in the clearing systems (SAP FI/CO) is not current enough to be tested in the form of the weekly waterfall chart. For example, invoices from providers are sometimes not recorded in the system until late in the process, even though the project was aware of these costs early on due to the planning cycles. Thus the waterfall chart must include all information that was present in the available clearing systems as well as the data collected in the weekly reports. Of course this means that data must be corrected as soon as the true data (such as value changes in invoices due to rebate systems) is entered into the clearing system. In the sample project, this deviation was located in the lower range of the annual budget and was corrected on a monthly basis.

The total number of test cases was precisely defined in order to determine the test progress

A cumulative approach was chosen for the test iterations, which began during the implementation: During the implementation, the test cases were defined and for each project were decided upon in sum together with the business departments so that the corresponding report always included the sum of all test cases (including those processed earlier). This permitted reporting to be carried out for each iteration step based on the total number of cases, defined early in the process. The image on **test progress** points to this in a goal-oriented manner (all test cases carried out).

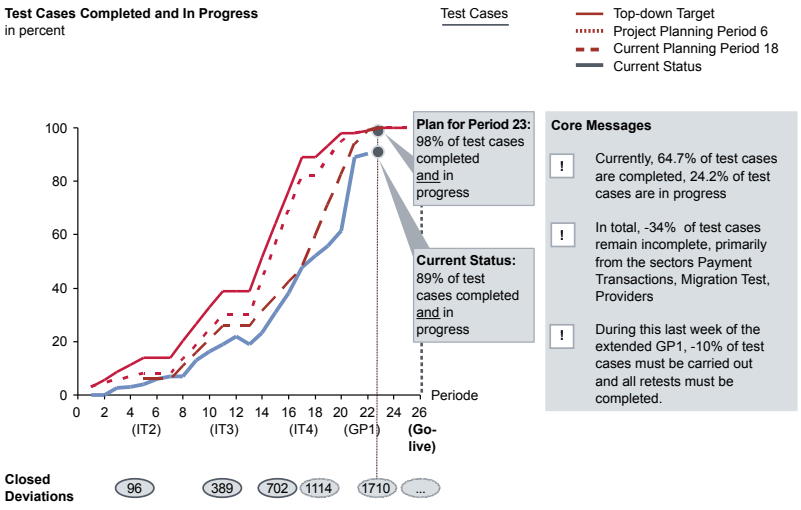

Image 10: Test Progress

In each iteration step, test cases were designated that were to undergo a regression test and those test cases defined that were necessary for the newly implemented functionality. Only the total sum of all test cases is presented in the depiction of progress; detailed illustrations of each project exist as back-up files.

The goal of the test case statistical analysis is the representation of the course of the test and the goal achievement based on the level of completion of the test cases in comparison to the original and possibly reworked plan. The addressees are the program managers, the PMO and the Test project management team.

The **point of departure** is the total number of test cases, which is to be reviewed at the beginning of each respective test iteration. It is entirely possible that changes will occur here during the process, as some test cases will be dropped, new ones will be requested by the business side, and at certain points additional regression tests will be required.

In all, the total number of test cases in the sample project fluctuated by a maximum of ten percent from the original number of cases over a period of five months.

The test progress diagram
- is an evaluation of the processing of test cases, updated on a weekly basis,
- shows the originally planned completion process for the test cases (solid red line),
- shows the adjusted completion process for the test cases on the basis of new planning within the project (dotted lines),
- shows the actual progression of completed test cases (blue),
- shows the number of test cases in *progress/completed* as well as core messages about these test cases in separate text boxes, and
- also provides information on the number of closed deviations.

Ideally, the courses of the curves for the planned progression and the actual progression are coincident and there are no new planning curves. The input for the test progress consists of the number of test cases that are closed, open, and in progress, within all projects. This requires that daily test case statistics from all projects be available.

It is necessary to clarify the testing procedure in terms of planning and degree of completion

Lessons Learned: Although the terms *in progress* and *open* had been clearly defined at the beginning of the project, they still turned out to be important points to be discussed between the program managers and the test management team. In the sample project, both the processing of the test cases as well as the deviations identified during the process were consistently named: *In progress* meant that the test case was being carried out or that an error had occurred within the case that still had to be corrected. Such a test case was termed completed only after a successful retest. By contrast, an *open test* case had not yet been started and still had to be carried out. This allowed the backlog to be identified: In general, *open test* cases lead, with a certain amount of probability, to errors being made and illustrate the work that still remains to be done. The number of *open test* cases functions as a gauge for the project's likelihood of meeting the deadline and the remaining work required of the test team, while the number of test cases *in progress* must also be added to this number.

The number of completed test cases and corrected deviations is compared to the basic values in a weekly rhythm

The qualitative evaluation of the implementation was achieved in a graphic representation of the test cases together with the number of deviations (unexpected system behavior in the test) in the report titled **Test Cases & Deviations**.

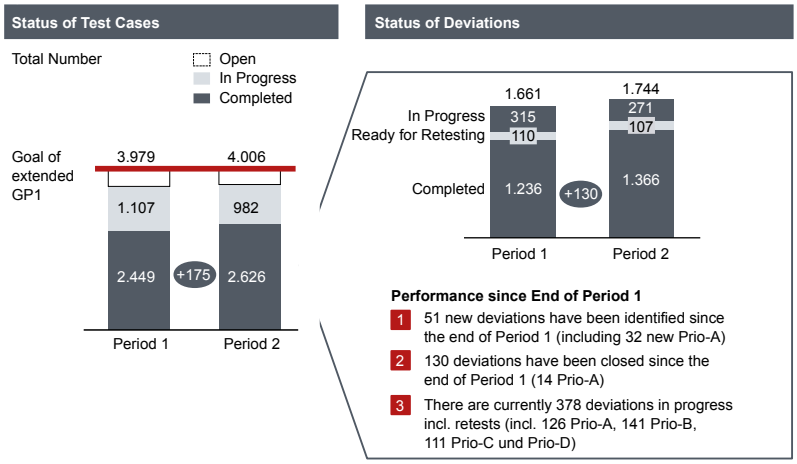

Image 11: Test Cases & Deviations

The goal of this diagram is to show the progress in the processing of the test cases and the processing of deviations in comparison with the progress of the previous week. The addressees are the program managers, the PMO and the Test project management team.

The point of departure is the number of deviations recorded by the target date in comparison to the number of test cases completed by that point. Distinctions are to be made between completed (meaning corrected) deviations, deviations about to undergo retesting, and those deviations still in progress.

In this project, the error priorities were also recorded (the number of A priority errors, etc.).

The test cases and deviations diagram

- is a weekly analysis of deviations in the test and of the processing of test cases,
- shows the current state of the test cases: the number of all test cases in the three states - *open, in progress* or *completed* - in comparison to the previous week,
- shows the current state of all deviations in the three states - *completed, ready for retesting*, and *in progress* - in comparison to the previous week, and
- provides information regarding the number of new and closed deviations in comparison to the previous week (including Prio A), and furthermore illustrates the distribution of deviations *in progress* across the priorities.

There is no *ideal situation* in this report. Basically, no stagnation should occur in the processing of test cases and the correction of deviations, other than at planned intervals (such as when caused by the new positioning of test environments, etc.). Obviously it is best to have as few deviations as possible, whereby one should not underestimate the psychological effect of errors that are allegedly not found.

The input consists of the number of test cases and deviations from all projects that are completed, open and in progress. This report requires that daily statistics on test cases and deviations in all projects are available.

Deviations are to be classified according to how critical they are to the success of the project

Lessons Learned: The number of deviations did not correlate with the number of test cases in the respective iteration step: Test progress therefore did not mean that fewer deviations were found per test case. This would only have been the case if the quality of the implementation had changed at a specific point in time (such as due to new developers, better development methods, tougher quality control during implementation, etc.). In this project in particular, it was noticeable that up until a few days before the Go-live, Prio-A deviations that could prevent acceptance were still being identified, which impeded the release process to a significant extent. This report was therefore used up until the Go-live. Statistical projections regarding the expected errors, calculated on the basis of previous performances, were of limited value in this case. The classification of the deviations according to four priorities showed itself to be useful, whereby the two non-essential groups were combined in the end as no distinction could be made regarding the treatment of these groups. The distinction remained between production-impeding deviations, production-impeding deviations that permit workaround, and non-production-impeding deviations.

A **Dashboard** was created and updated on a daily basis in order to ensure transparency for the decision-making committees regarding the relevant criteria in the last weeks before the Go-live.

A Dashboard is used to ensure transparency about the Go-live criteria

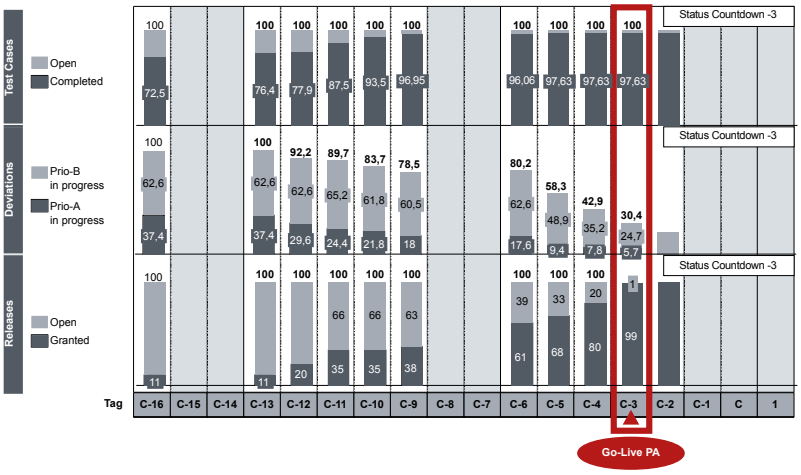

Image 12: Dashboard for Monitoring Progress

The goal of the dashboard is to provide an overview of the status of the essential dimensions during the most important time period before the Go-live in order to manage information and to prepare for the final migration release. The addressees are the program managers, the PMO, the steering committee, and the final Go-live release committee.

The **point of departure** consisted of the reports on the progress of test cases and deviations. In addition, the sum of the progress of the defined releases of the business departments was portrayed.

A critical factor for success is the constant delivery of information to all stakeholders involved

The Dashboard

- is used over a period starting two weeks before the Go-live until the Go-live to provide daily representations of the progress in the test, the correction of deviations and of the releases,
- shows the number of test cases: *carried out* (completed and in progress) and *open*, per day,
- shows the number of deviations *in progress* in the A priorities (*production-impeding*) and B priorities (*workaround possible*), and
- shows the number of open and granted releases for the migration.

In an ideal situation, all test cases have been completed, all deviations corrected and all releases granted one week before the Go-live. The input consists of the status of the tests, the deviations and the releases. The prerequisite is that the daily status is available at the best level of granularity.

Discussions about critical and acceptance-hindering points are summarized in the Ready-to-Go Checklist

Lessons Learned: The dashboard was the main instrument for providing information to the participating stakeholders regarding the most current status in the two weeks before the Go-live. The stakeholders requested the dashboard on a daily basis and it served as the foundation for discussion in all important decision-making councils. In each meeting, the necessary detailed information had to be available in case of follow-up questions, for example, which releases from which departments were still open or which deviations were still acceptance-hindering.

The **Ready-to-Go Checklist** is an essential element needed for the final and physical signature for the release of the migration and therefore the start of the concrete transfer activities on the Go-live weekend.

The **point of departure** is the list of organizational units involved and affected as well as all providers involved. The process dedicated to inquiring about approval from each party not only allows the information about the upcoming Go-live to be ascertained, but also seeks out solidarity in case of difficulties ahead.

The final critical and possibly acceptance-hindering points are discussed on the basis of this checklist and are evaluated in the final release meeting at which all necessary decision-makers are personally present.

This checklist includes all previous information from the projects regarding coverage of test cases and correction of deviations as well as from the business departments regarding the releases and constraints. In this project, all providers involved were also required to explicitly state their willingness to participate in the transformation. This affected both the delivering as well as the receiving sides.

Projects	Ready-to-Go		
	Test	Devia-tions	Risks manageable
Accounts	☑	☑	☑
PT	☑	☑	☑
Test & Migration	☑	☐	☑
Public Bank/Financial Accounting	☑	☑	☑
Peripheral Systems	☑	☑	☑
Flat Rate Tax	☑	☑	☑
Communication			☑
Processes	☑	☑	☑
Linking with IT Provider	☑		☑

Linie	Ready-to-Go
Operations	☑
IT	☑
Private Banking	☑
Finances	☑
BPO-Support	☑
Corporate Clients	☐
Loan/Collateral Mgmt	☑
Legal	☑
Taxes	☑

Provider	Ready-to-Go
IT Provider	☑
PT Provider	☑
GCP	☑
Card Provider	☑
Public Bank	☑

Program Management	☑
Decision-making Board	☐

Decide today

Image 13: Ready-to-Go Checklist

The quality of a transformation for the entire bank is reflected in the subsequent production deviations

The Checklist for the preparation of the final Go-live decision is the Ready-to-Go Checklist

- per project for the categories: test, deviations, risks,
- per line organization unit and
- for the program managers and the decision-making councils.

In an ideal situation, all releases will have been granted prior to the final Go-live decision workshop.

The goal is the fastest possible achievement of the defined “absence of errors” in production processes

The measurement of results and thereby the internal evaluation of the program’s success as a whole have different perspectives:
A representation of the **production deviations** is suited to portraying the effects of the transformation on the bank as a whole. Here, both operative IT topics as well as bank-specific problems are of interest. Production deviations include any and all situations occurring after the Go-live that do not conform to expectations. Aside from classical IT errors, this also refers to a lack of training or communication that led to false expectations.

The **point of departure** is different types of evaluation of the success of the transformation by various organizational units. It is important to make the situation of each respective group clear to both groups.

Criteria for the Success of the Migration:

- It is important for the IT department and for the entire Operations department that the new solution requires less effort than the process in place before the transformation.
- It is important for the business departments and the daily banking operations that the impact on customers is minimized and that all necessary and desired functions are implemented.

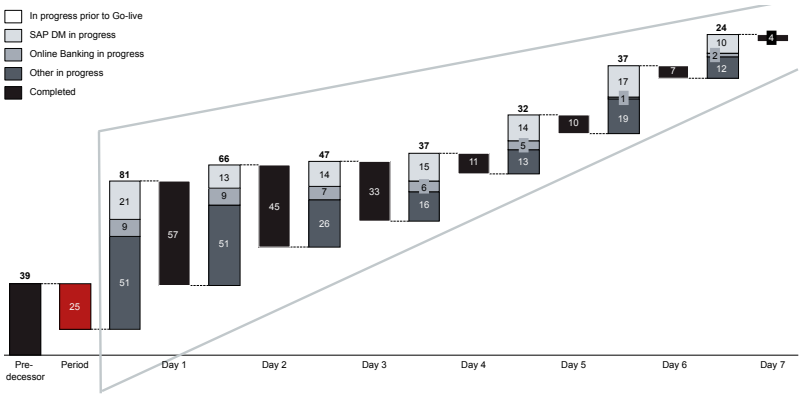

Image 14: Number of Production Deviations

The core tasks of the program may not be transferred to the line organization too early

The core message of this report is that the initial difficulties inherent in any large transformation must be quickly overcome and that, at the latest after two or three weeks, there be fewer difficulties than was normally the case prior to the transformation. Improvement compared to the previous situation is a main goal for all stakeholders and can be transparently demonstrated using this illustration. The trajectory of the curve leads to a theoretical point of complete lack of deviations. Here it is important that the backlog of open deviations be processed using the remaining resources distributed to the projects. Normally the line organization that takes over after the transformation cannot take on the additional effort required to process this backlog.

The report on production deviations

- is a depiction of the development of the production deviations and a breakdown of the main error categories, updated on a daily basis,
- shows the number of recorded (new) and completed production deviations per day, and
- contains explanations with core messages on the distributions among the main error categories by percent.

In an ideal situation, the rapid and clear reduction of the height of the bar of newly recorded deviations is noticeable and the average value curve of the daily production deviations tends towards the lowest possible boundary value.

In our sample project, stability was achieved after less than three weeks and the number of new daily production deviations was clearly below the values measured prior to the transformation.

Summary of the Core Messages

Over a period of nine months and during the main phases of the transformation of a core banking system, the following criteria emerged as being essential for success:

Factors identified as essential for successful Go-live management

- The **early** but differentiated and need-based **integration** of all **stakeholders** in the project as well as moderation between these parties is essential.
- The highest degree of **transparency** throughout the entire company hierarchy regarding the **most critical work packages** increases the efficiency in reporting and minimizes the number of necessary voting sessions.
- Timely **avoidance** of **conflicts of interest** through **communication** of all decisions and facts with the help of **structured reports** and **regular council meetings**.
- A transparent and standardized **progress review** should be **initiated** at the beginning of the project and **consistently maintained** throughout the project.
- **Short reaction times** of the **decision-making council** enabled by meetings at least once a week, particularly in the weeks prior to the Go-live.
- Clear **prioritization** of all **tasks** and the necessary **decisions** as well as continual review regarding adherence to these tasks and decisions.
- Relevant **test scenarios** should be **identified** in a timely manner and the scale of the tests shaped in a need-based manner.
- **Employees** should be **prepared** for the transformation through documentation and simultaneous **early** training courses on the changes occurring on the process and business side.
- Prior to the **release**, clear **rules** must be decided upon collectively by the IT and business departments. The release processes should be stringently implemented and include **all stakeholders**.

Conclusion

The Go-live was successfully carried out for the bank due to the intensive preparations in Phase 2 (*Preparation*) and went mostly unnoticed by the customers. The low number of production deviations and their rapid reduction to values below the expected level led to a reduction in all special monitoring instruments of the production team as early as four weeks after the transformation. The target values of stable production operations were therefore achieved eight weeks earlier than planned.

Basis for success of the PMO

This success was particularly due to the employees in the program, whereby the following points form the basis for successful program management:

- Creation of the highest level of transparency across the different hierarchy levels in the current situation
- Fact-based moderation in order to prevent escalation and conflicts of interest
- Prioritization of upcoming decisions and activities
- Creation of a decision-making council capable of action and with weekly meetings
- Following up on prioritized activities and communicating the decisions

The significance of these individual steps only becomes clear in the total overview of the consequent implementation of these steps during the time-period required until the Go-live.

The COREinstitute

Transformations of complex systems are recurring challenges faced by companies and organizations across all industries and sectors.

The creation of solutions for these challenges requires precise analysis and taking stock of the current situation, on the one hand, and, on the other hand, leaving behind limiting ideas and concepts.

The COREinstitute observes and analyzes the dynamics and systematics of the transformation of complex systems in many regions and relevant industries. It provides a platform for current developments and new problem-solving approaches in transformation management for industry experts, researchers, engineers, and artists.

This work results in answers to specific questions and tools to support transformation programs.

The COREinstitute makes the results of this interdisciplinary work available to the public in discussion series and publications.

COREtransform GmbH
Am Sandwerder 21
14109 Berlin | Germany
www.coretransform.com

Tel: +49 30 26344 020
Mail: institute@coretransform.com

October 2014