

Core Bank Transformation in Practice

Large Scale IT System Renovation

Justin Kilimnik

IBM

Brisbane, Australia

justin.kilimnik@au1.ibm.com

Chris Pavlovski

IBM

Brisbane, Australia

chris.pav@au1.ibm.com

Abstract—The banking experience for many people today is fundamentally an application of technology to be able to carry out their financial tasks. While the need to visit a bank branch remains essential for a number of activities, increasingly the need to support mobile usage is becoming the central focus of many bank strategies. The core banking systems that process financial transactions must remain highly available and able to support large volumes of activity. These systems represent a long term investment for banks and when the need arises to modernize these large systems, the transformation initiative is often very expensive and of high risk. We present in this paper our experiences in bank modernization and transformation, and outline the strategies for rolling out these large programs. As banking institutions embark upon transformation programs to upgrade their banking channels and core banking systems, it is hoped that the insights presented here are useful as a framework to support these initiatives.

Keywords— core banking; transformation; finance; payments

I. INTRODUCTION

Core banking traditionally refers to range of functions including deposits, loans, payments processing, merchant support, and card related transactions. For institutional banking this will refer to monetary exchange markets, commercial loans, and wealth management. Although the broader scope will differ between financial institutions, there is general acceptance of these functions as being the traditional core banking domains. These core functions form the mission critical daily operations to be supported by banks. The information technology and communication systems required to support these services have the highest availability requirement with any outages to business continuity having a significant impact to banking revenue and operation. For instance, it is reported that the average cost for downtime for financial and brokerage firms ranges from USD\$1.5 million to USD\$6.5 million per hour [1]. Additionally, the large volume of transactions to be supported also means that high-end platforms or large clusters of commodity hardware are key decision points for these systems.

The choice of platform and technology to build core banking systems has evolved to the point that packages now offer a broad range of capabilities to address the various functional domains. However, in practice these banking packages typically require significant customization for deployment within the targeted institution, often causing

significant delays and increased costs during the project development life cycle. We aim to describe an approach to core banking systems upgrade and/or replacement based upon lessons learned in large transformation projects. Whilst banking institutions have undergone such upgrades in the past, often tackling a full systems replacement in one phase, the same big bang philosophy is unlikely to be the best option today and will foster significant risk due to the increases in complexity of capability offered, diversity of technologies, and the large number of systems to integrate with.

In this paper we discuss the approaches that are applied to core banking system modernization programs. We highlight the lessons learned that are applicable to these programs and note the similarities with transformation programs in alternative industries. We also propose a high-level strategy to carry out a major banking system upgrade that balances the risks associated with large projects, the technologies applied, together with the time and resources required. Given the significant costs associated with core banking replacement (sometimes up to several billion dollars), it is hoped that the presented approach and lessons may be useful to banking organizations embarking upon similar large scale modernization. These observations also have applicability in other industries. Hence we view the main contributions of this paper as follows.

1. Define the scope of core banking modernization programs and the strategies/approaches available for core banking platform replacement.
2. Present lessons learned in banking transformation, with approaches presented that balances risk, costs, & time.
3. We also outline a project schedule with high level tasks that is useful in practice to help plan the program.

In the next section we review the literature related to core bank transformation and modernization. This is then followed in section III by a discussion of the functional scope that defines core banking and the associated support systems. We then outline in section IV several approaches and methods to implement a modernization program for banking. This is then followed by a discussion of the lessons learned in core banking replacement. In section VI we present a core bank program schedule and describe the key tasks associated. Finally, in section VII we summarize and discuss the work presented in this paper and note that there are several areas of further work.

II. RELATED LITERATURE

There has been much recent work on banking system trends that focus upon the application of recent technologies such as mobile devices, SOA, and payment systems [4, 5, 10]. All these advancements rely upon core back end banking systems, the key theme of this paper.

There are several papers that explore the core banking functions and their implementation. The features and architecture for core banking systems are detailed in [2], where a functional architecture of the banking sub-systems is described. This includes treasury modules, retail banking, private institutional banking, wholesale modules, and multi-channel capabilities. A further paper discusses the core banking capabilities in the context of electronic banking [3]. Whilst it may be argued that the majority of banking is now focused towards some form of electronic banking, the authors put forward the notion of the disruptive nature of e-banking and identify eight core capabilities to apply to emerging information technology trends and to reconfigure existing business models.

There are several papers which discuss the application of SOA to core banking [4, 5]. The case for a meta-architecture is argued in [4]. The authors suggest a layered architecture to reduce complexity of design with a set of defined views to improve clarity of understanding of the systems involved; key to this is the use of SOA to provide a layer of banking services for interacting IT systems. Other published works support the approach for an incremental and progressive transformation of core banking systems based upon SOA design principles [5]. The paper also reviews the trends in banking globally observing that banks are taking advantage of technology to improve margins, transforming to become customer centric, and are coming under increased pressure from new imposed regulations. To tackle the challenges for ‘bank renovation’ the authors suggested that SOA can be applied to manage the transformation in a progressive manner, rather than the more risky approach of total replacement as one implementation.

In [6] the success of IT implementation of a banking system is studied, where a case study is used to support the finding that the success of IT systems is closely related to the alignment achieved with understanding the banks’ business operations. Another paper that follows a case study also reviews the management practices and their associated impact to effective implementation [7]. The paper presents a number of findings including the balance of out-sourcing and use of in-house resources, the need to communicate benefits of newer technologies to bank staff, and improve management of IT within the bank.

A further paper studies the attributes of repeatable success in core banking implementations [8]. The authors explore the linkage with business and technical requirements describing the importance of IT delivery execution through planning and project implementation processes. They also note the importance of capturing lessons learnt as a key differentiator to be applied. We follow this line of thinking and explore in greater detail the experiences in core banking system modernization, describing the technical and functional principles to be observed during these implementations. This

area appears less well studied and hence we present our observations and findings so they may be applied by similar projects embarking upon transformation; with a view to help manage the complexity and reduce the significant risk and costs associated with these types of programs.

III. MOTIVATION AND FUNCTIONAL SCOPE OF CORE BANKING

In this section we discuss some of the key motivations for embarking on core bank replacement and begin to describe the key elements and functional scope that embodies core banking. This provides a baseline of functions to be supported. In addition, the extended set of functions that are optionally considered part of the core banking operations are also explained.

The exact scope of core banking will differ between institutions however the presented capabilities appear to be generally accepted by the financial industry. For completeness we also outline the operational banking and channel support systems. The following diagram (Fig. 1) illustrates the position of core bank systems with respect to other banking systems. The following sections provide a more detailed discussion of the components depicted, but we first outline some of the key motivations for pursuing such a program.

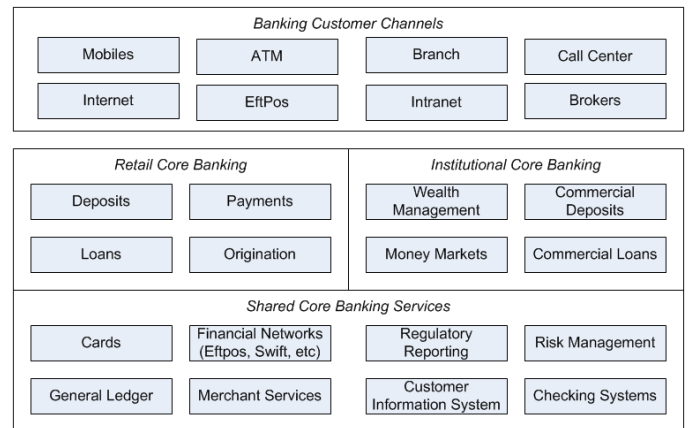


Fig. 1. Functional View of Core and Supporting Bank Systems.

A. Motivation

The motivation to modernize is often due to several factors that have influence upon the bank that eventually gather enough critical mass, creating the inertia to proceed. In general, the momentum commences with existing systems and processes that restrict the business from delivering new financial products rapidly to the market. They may be viewed as inflexible and a limiting factor to business market agility. Other motivating factors that often then emerge include:

- Legacy systems constrain the business to support multiple emerging banking channels, such as mobile and internet, since the design of older systems predate the newer virtual channels.

- Too many financial products create additional complexity and hence product consolidation and reduction is often a key motivation to engage in modernization within the bank.
- In order to manage costs, the need to rationalize and simplify the number of banking systems in operation.
- Increasing transaction volume is placing pressure on contemporary systems that requires either an upgrade or warrants introduction of a new core banking system.
- Competing banks in the region make announcements to modernize and hence viewed by the marketplace as more progressive due to their willingness to embark upon change, as apposed to remaining stagnant.

Finally, the constant change due to mergers, acquisitions and disposal of business units is also a disruptive contributor to the technology landscape. Often this results in duplicated systems, even if merged business units rely upon the same core banking platforms there will be significant difference so as to require separate instances and hence support.

B. Retail Core Bankings Systems

The precise nature of what is termed core banking does vary to some degree by region and institution, however the functional domains typically classified as core banking include: deposits, loans, origination, and payment cards. In the core banking sense these services are generally associated with the retail customer as well as commercial entities; we now briefly describe each of these core banking services further.

Deposits represent the cash management functions for supporting retail customer bank deposits. Whilst historically the bank deposit was a straightforward account that offered a variable interest rate yield, a number of products now require support including: traditional savings accounts, loan offset accounts, term deposits, check accounts, and trust accounts.

Both secured and unsecured loans form the core banking services for both retail and commercial customers. Secured loans are those loans with assets associated, such as home loans, whereas unsecured lending represents higher risk financing; this includes personal loans, credit card debt, and lines of credit. Origination deals with the process of applying for a new loan, and hence involves verification of customer identity, credit checking, and risk profiling. Extensions to the standard loans process may also include mortgage brokers and other parties which require further systems support to cater for brokerage fees and commissions.

Payments card support has expanded considerably over recent years, with many alliances in place between banks, merchants, and other third parties such as Internet payment providers. The card types typically include credit cards, debit only cards, and hi-value credit cards. In each case, a variety of interest rates combined with annual fees will apply to these cards, making the transaction landscape more complex. In addition to the Internet, transactions may be routed via financial networks such as EFTPOS, SWIFT, and Maestro.

C. Institutional and Other Shared Core Banking Systems

Institutional banking represents the set of financial services to support small to medium businesses and large corporations. This includes management of large financial portfolios, international transactions, and engaging the money markets (investment and trading). Hence, some of the core capabilities are similar to retail core banking such as deposits, loans and payments; however, these are oriented to support commercial scale operations for business. Additionally, these systems furnish capabilities to support wealth management (such as superannuation), merchant support services for processing payments over financial related networks, and extended interfaces for international clearing networks.

The money market systems enable investment and trading for the business customer, this includes securities, foreign currencies, derivatives, and shares trading. These systems provide functions for lending and credit.

Further core banking systems include wealth management (e.g. SMSF; often as a separate banking division), merchant services, and checking systems. There is also support required for various local regulatory reporting standards and international clearing networks.

The general ledger is the final destination for recording all transactions processed by the bank. The system provides a range of accounting, budgetary, and reporting functions with key interfaces to the core banking systems such as deposits, loans, and payments. The typical capabilities include transaction journals, reconciliation, financial statements, daily balancing, and budgetary analysis. Additional banking services are supported with interfaces to fixed asset systems, risk management systems, and other financial accounting systems.

IV. ROLL OUT STRATEGIES IN MODERNIZATION

In terms of complexity, duration, and cost, the strategic decisions for implementing a core bank modernization program are not dissimilar from several other industry solutions which undergo large transformation initiatives. Often, one complexity which large programs overlook is the time and effort required for adapting the organization to accommodate the new IT systems together with the new business processes that need to be altered and established. As in many programs, the initial starting point for scope is believed to be focused and manageable, but as the complexity of interfacing systems and the specific needs (the requirements) of the business become clearer, the scope can easily propagate well beyond initial estimates.

Time and again it has been shown that project initiation and managing requirements (both functional and non-functional) are critical to successfully completing large complex programs. Sound project management and architectural guidance during the development and implementation of the program is another absolute to ensure continuity in program direction. In addition, delivery fundamentals in methods, tools, and processes are needed to provide the foundation for a broad team of people to be able to effectively and efficiently implement the scope of the modernization program.

In the following sub-sections we now expand upon our experiences in banking and other industries to illustrate some of the key strategies and considerations to made in defining and executing a core bank modernization program. We commence by describing the overall strategies for delivering functionality and then discuss the roll-out approaches for core banking, on-line channels, and banking support systems.

A. *Phased Deployment Strategies*

The most immediate strategy that is always discussed is to deliver the entire scope of functionality in a single release, i.e. Big Bang. This is often considered the highest risk approach as it requires the largest allocation of both time and duration in order to complete the banking solution, generally over several years. We have observed that historically a Big Bang approach has been successfully implemented by some banking institutions. However, this was perhaps during the early 80's and 90's where the scope of functionality, based on banking financial products, was much smaller in number compared to today's environment. Additionally, the banking channels did not contain the breadth and complexity as present day multi-channel environments. Hence, the prospects of success using Big Bang is considerably more challenging today due to high technology dependence, complex range of products offered by the bank, and the increased number of integration points.

Notwithstanding, for smaller institutions and those willing to accept the risks there are several key lessons and techniques to be observed if attempting such an implementation approach. It will be essential to have multiple initial pilot releases to refine aspects of the system prior to full go-live. In addition, dress rehearsals (or live but limited pilots) of the deployment, operation, and roll-back are also vital. During a dress rehearsal release, the issue of consolidating and merging all transactions (including financial, origination, and analytical) created during those times with the production databases also need to be accommodated. The option can be taken of course to stage these dry runs in non-production environments, but this then diminishes the overall ability to reveal problems that may be expected in the final production system launch.

A phased delivery approach is often the most likely strategy adopted by large banks for their modernization program. One of the most pressing issue to deal with when developing a program with multiple releases is what is the way functionality will be progressively deployed? This can be achieved in several ways including rolling out the system by product, by customer, by region, by branch, or roll out by core banking function. We now explore each of these techniques in more detail.

B. *Rollout by Product*

Rollout by product involves primarily the deposit and mortgage related offerings. In general, the number of financial products offered by banks has grown over time to the point that these have gone beyond a manageable set, hence consolidation is required. As such, the approach to deploy system functionality by product line requires an analysis to consolidate what is in practical use and of strategic importance. In order to develop a phased delivery based on products it is also important to consider the how these inter-relate. For instance, a

deposit account can actually be an offset account that is linked to a loan. A line of credit account is also a form of loan arrangement and one product may be available in multiple systems due to previous attempts to upgrade or transform the core banking landscape. In addition, customers are likely to have products split between banking systems so the transition to a new platform is likely to cause challenges in maintaining a seamless view to the customer. A typical approach to counter this issue of multiple systems is to re-engineer the on-line channel to hide these complexities, using some form of service integration bus. This of course increases the overall costs while these technical measures are used during the transition period.

The key challenge of course for any phased approach is that there will be that two systems in operation until the program is completed and the former system retired; a period of *co-existence* of multiple core bank systems. Hence, it is necessary to consider the impacts associated with the other related core banking systems. This includes treasury, collateral, collections, CRM, fraud management, data warehousing (particularly for regulatory compliance), and brokerage systems. All these systems, along with the on-line banking channels, will require some form of integration whilst multiple core banking systems are in place until final cut-over occurs. Once again, the use of a service integration bus between adjoining systems will hide much of the complexity and if not already deployed would be a key component to include as part of the core banking program.

C. *Rollout by Customer Segement*

A roll-out by customer segment may be by customer type (business or retail), by customer accounts held, or by customers recorded in a particular system. The later segmentation hints at one of the first challenges in this approach, which is to identify which banking system the customer actually resides. Over time many financial institutions have attempted to consolidate customer records into a central repository. This is true also for other industries, where the result often means that multiple customer repositories are in production, with different information stored about the customer in each system. Furthermore, it is not uncommon that customer data is duplicated in these systems and these in fact may not match.

The most pressing challenge when dividing a rollout by customer is how to present a simple and consistent interface to the customer via the on-line channels. Whilst branch and back office staff may be able to deal with the complexity of multiple systems, the same is not true for the customer's on-line channel experience. One immediate solution to this is to hide the complexity by redirecting the customer to the appropriate system; alternatively it is possible for customers to access multiple on-line systems by ensuring that standards exist on user interface experience to hide the differences of multiple on-line systems. In spite of the integration approach adopted, the question of whether to modify the existing channel or deploy anew is to be addressed. There is additional training and organization change required to support new systems, and quite often new systems cannot be rolled out until the data schema for the customer repository is finalized and agreed.

Where possible, the approach of building a new on-line channel that commences after a period of time following the start of the core banking upgrade is more ideal. This gives time

for the fundamentals of customer data and product definition to be finalized, and hence not subjecting the channel development project to excessive change during project delivery. However, in practice it may be too overwhelming to have two large programs running in parallel, even for a period of time, as such often these programs are engaged sequentially.

D. Rollout by Region or Branch

Deploying a core banking program progressively by region or branch is often an approach that appears favourable. This allows the bank to limit customer impact as a series of production pilots that gradually increases the migrated customers. The approach lowers the overall complexity of the program when customers are distributed homogeneously on a small number of banking systems. A further benefit is that the ability to target an initial smaller customer population (associated with a branch or region) is bestowed. Organizational training may also be coordinated by branch roll-out. Whilst this approach is very sound when there are fewer duplicated banking systems, in practice however, there can be several challenges. The diverse nature of products and customer types means that a single branch is likely to need support of all core banking systems that form part of the modernization program. This means that the full functionality is required in all affected systems to enable cut-over of the customer base for each region or branch. Further, the operation of multiple on-line channels and core banking support systems will require support for customer and branch staff access. Finally, it may be generally the case that a subset of the bank branches will contain the majority of customer accounts hence the deployment progress is skewed considerably until the larger branches are accommodated.

In spite of these challenges, a deployment approach that is a combination of a defined customer segment and a branch rollout, offers a balance in managing risk and scope in a reasonable way. A subset of the customer base can be the target of an initial roll-out and by focusing on a branch or region for an initial deployment the organizational change aspects can also be managed in a rational manner. Such an approach is often employed by other industry segments such as telecommunications, energy distribution, and healthcare. This provides a good basis for scheduling deployment over time.

E. Rollout by System or Functional Domain

A further approach to be considered is to sub-divide the modernization program by core banking system, or rather, functional domain. An obvious choice that presents is whether to engage in modernizing the on-line channel or core banking system first. This is further subdivided by functional area, by prioritizing the need to renew either origination, loans, deposits, payments or customer repository as the initial target system. This decision has a flow on effect to other banking systems that are impacted by changes to these core systems.

The key observation to make regarding changes to foundation systems such as core banking is that essentially it is the data that is stored, both transactional and customer related, that dictates many of the design decisions that are to be dealt with by both upstream and downstream systems.

It is generally observed that there are three states of data. These are referred to as data at rest (in database), data in motion (traversing network or memory), and data in use (under active change) [9]. We refine further the state 'data in use' here and note that data on entry requires individual consideration and is largely represented by the on-line channel and banking support systems used to capture data. Moreover, such data to be captured is dependant upon the data model used when at rest, i.e. the core database systems that support banking. Hence, the key starting point is often the main database schema due to its impact to the design of upstream and downstream systems. This is further revealed in the need to rationalize products offered, reduce duplication of customer data, and the need to capture certain information for regulatory control. As such, whether a banking channel or core bank systems is chosen to be renewed first, it will be the central database repositories that dictate the extent of work required to display, integrate, or analyse financial information.

As mentioned, the ability to tackle both the channel and core bank system simultaneously would seem a good choice, but the complexity due to the magnitude of effort would often be prohibitive. Hence, a sound approach is to commence with the core banking system and associated database architecture, followed where practical by upstream systems (such as the banking channels). There is no right or wrong answer to which system to include in a modernization program, and ultimately the decision is based on the individual needs of the bank, which is a function of balancing time, cost, scope, together with the risks, skills, and organizational needs to be addressed.

V. LESSONS LEARNED FROM TRANSFORMATION

We now describe the lessons learned by expanding upon the approaches discussed in the previous section. There are a number of considerations to observe when planning the modernization program such as technology choices, scope of program, and broader risks associated with systems integration. It is often good practice to commence the project with weekly risk management meetings to facilitate resolution and help manage some of the more critical decisions that need to be addressed during implementation.

A. Co-existence Strategy

One of the key issues affecting any large transformation program is how to deal with the existing system being retired or replaced. Even when the legacy system is initially targeted to be retired, it is often the case that plans change as the new platform being developed is not able to provide sufficient functionality in its first release. Hence, the business will often end up with two systems at least in the short term. As such, a key strategy to have in place is co-existence between the new and old system in production. Developing such a strategy, even as a back-up plan will reduce risk. Aspects that need to be considered include: which system will act as the master repository for certain types of data; how to deal with customers in two systems; and ensuring that there is replication, or a data exchange process, in place (either real-time or batch). In addition, how to visually present access to these systems to the external customers and the mechanism to direct (or redirect) traffic also requires design. Although the above brief points are

common challenges to be discussed, a co-existence strategy requires a much deeper discussion,

B. Master Customer Data

Will the master customer data management system be replaced as part of the program or an existing system used? An upgrade or replacement of the master customer database will contribute to the risk due to the reliance by nearly all core banking systems, it follows that changes to this system will impact all integrating systems. As mentioned previously, often customer data is duplicated in several systems, hence the opportunity to address this may be appealing as part of the modernization program. There is always the need to ensure the data is accurate and a data cleansing exercise will need to be factored into the schedule as data is migrated from one system to any proposed new system.

C. Service Integration Bus

Many institutions have introduced an integration bus of some way to help disentangle the traditional forms of point to point integration. The benefits of SOA and Enterprise Integration are well discussed. Integration layers contribute to technical rigor by establishing consistent messaging between platforms, reduces technical dependence between systems by accessing services via a single adaptor technology, and enables reuse of services. Where an integration bus is in place, extending the existing bus platform will be the most practical balance of scope and effort. If no integration bus exists the opportunity to introduce such a layer is granted. Although this means the introduction of another moving part to the overall program, this is still likely to reduce overall program complexity and risk.

D. Incumbent Bank Transformation Programs

Core banking modernization will be a key aspect of an ongoing set of banking transformation initiatives. Transformation is often a continual effort and there is likely to be several other programs underway within the bank, and so consideration is to be given on the impact to those existing projects. Where adjunct platforms are being upgraded it is worthy to consider placing these into a more static maintenance mode, so as to avoid excessive rework and change to integrate and interoperate with those systems.

E. Cloud Technologies

The use of cloud as hardware infrastructure (IaaS) or cloud as an application service (SaaS) is featuring more often as a viable deployment option. If cloud is to form part of the solution, clearly identifying the boundary of what components will be cloud and what shall remain in traditional data centers is vital. Several cloud approaches may be considered including public cloud, on-premise private cloud, and hybrid cloud solutions. The security implications of components deployed on the cloud also needs to be considered as this is more complex to control in terms of data privacy, securing transmissions, and most important – adherence to financial regulatory controls. In an attempt to make better utilization of hardware infrastructure, the use of virtualization has introduced

new complexity to the supported hardware environments. Cloud also offers a way to offload this technical overhead.

F. Package Selection

Package selection poses a significant challenge to banks as quite often there is an attempt to select a banking platform that provides the functionality desired ‘out-of-the-box’. However, in practice there is always the need to customize, sometimes significantly, contributing to unexpected costs and project delays. A vital aspect of the selection process is to consider the organizational impact of the chosen package. Whilst it is a challenge to replace the information technology systems and components, the effort required instituting new business practices and supporting processes is also a significant burden to the bank that needs to be estimated and factored into project schedules. If the intention is to select a major package to replace one or several existing core systems, a sensible criterion to apply is to ensure the functionality expected is of sufficient maturity to be useful with minimal tailoring. Often the upgrade of a previous version of a banking platform to a new version may also result in a program that is similar to a replacement, due to the breadth of change between platform versions. As in any high risk technology adoption it is most practical to carry out a proof-of-concept deployment of selected features to validate aspects of the platform under consideration, in particular the maturity. This will also give the opportunity to consider more seriously the reuse of existing systems to form part of the final solution versus those components to be replaced by a new package.

G. General Large Program Challenges

There are numerous additional challenges that are typical for any large transformation program that are not specific to any industry. This includes skills and resource availability with the selected platform, technology, or implementation language. Supportability of the chosen banking package and is there a local onshore or offshore support base? Accurate estimation of the effort to complete a large program of work, ensuring coverage for vendors, internal IT teams, and the time required for organizational change. The effective capture of requirements and the need to manage these during design and implementation can always pose some of the greatest risks. Change management and project delivery lifecycle management in addition to deployment and transition into business as usual operations. There are many other risks that emerge during any large program and having access to experienced people who have dealt with these problems is the most practical defence.

VI. CORE BANK REPLACEMENT PROGRAM SCHEDULE

We now capture some of the key observations during implementation and outline a project schedule that depicts the scope of core banking activities and how these and other tasks may be positioned with respect to the broader banking domain.

Since core bank replacements are typically hybrid solutions that comprise multiple sub-system types, it is appropriate to consider different development models (a.k.a. method) for each sub-system type. These candidate models need to be identified

during solution outline as part of the planning phase in order to properly frame the program schedule. Examples include using vendor package methods for core package configuration and customisation; agile methods for sub-system types that require a large degree of custom development with extensive User Interface and/or report design; and integration methods for integration bus development and waterfall methods for technical components.

The chosen delivery models need to be wrapped appropriately with project management and governance activities that will form the base framework of the schedule. One example schedule structure is depicted in Figure 2, with the basic assumption that a combination of package and

integration models (which may be waterfall in nature) are used together with agile methods applied across the program.

The project schedule is also useful as a checklist to transformation teams to aide in initial estimation. The schedule shows that a typical program is in the order of 3 to 4 years in elapsed duration. Depending on the actual scope, these programs may deliver more quickly; conversely they may also take much longer when some of the challenges discussed in this paper are not addressed early in the program. Whilst the project schedule is indicative, the key phases of the modernization programs are generally consistent as depicted in Figure 2. Next, we describe these phases and iterations in more detail, discussing some of the major sub-tasks and activities.

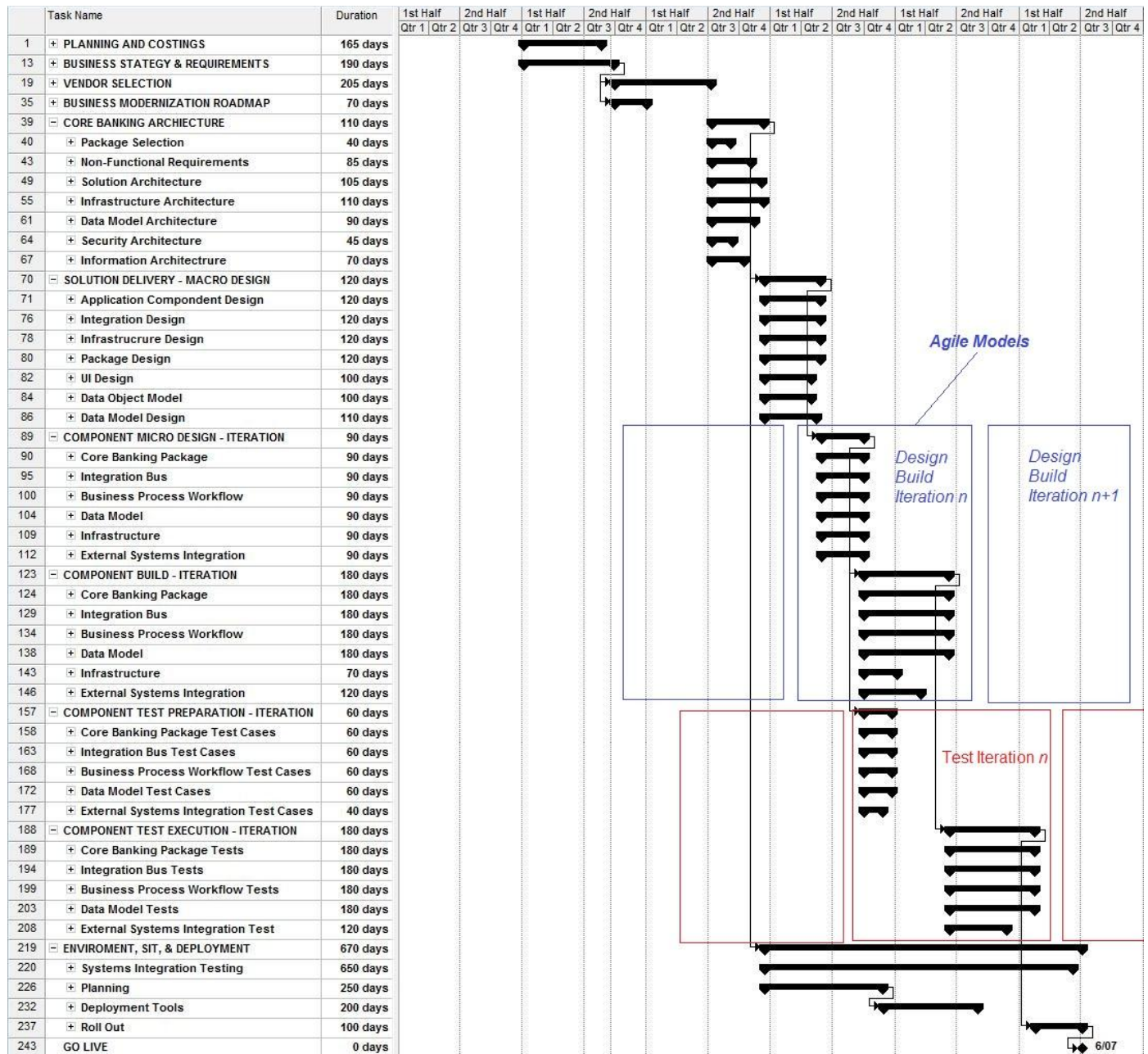


Fig. 2. Project Schedule for Core Bank Transformation Program.

A. *Planning, Requirements, and Vendor Selection*

The planning phase hosts the thinking to develop the entire program from concept and also includes development of the business case for modernization. This also includes the initial funding estimates and necessary preparations to seek board approval for project seed funding. Gathering of business requirements is often considered part of strategy development and will involve workshops to identify the business benefits, how the bank will differentiate from its competitors, and how best to address the needs of its customer base and market. During this phase activities such as (financial) product simplification, initial user experience design concepts, and high level requirements are established. Although linked as a sequential task, often the surveying of candidate product platforms and vendor selection occurs simultaneously, and in practice the core banking packaged supplied by potential vendors will impact the financial products, go-to-market strategy and project scope. Hence, it is usual that the vendor selection be commenced during the business requirements gathering phase, although the final package and vendor selection may commence after the business strategy and requirements are agreed upon by the business.

B. *Business Modernisation RoadMap*

In addition to the scope, candidate banking platforms, and funding model, the roadmap for roll-out, replacement and legacy system retirement (and co-existence) also needs consideration very early in the program. These tasks fall under the business modernization roadmap in the schedule. In particular, system retirement will often be linked to the business cost cases. The target systems for replacement will also start to reveal more information on the detailed system requirements, as the existing business process and human workflow tasks are assessed further to ensure that the capabilities of the candidate banking packages are able to deliver the expected functions. Already assembled will be a small team including the lead program management staff together with business and technical leaders. Often the project may commence with a team size of around 10-20 during the early phases. This may increase to several hundred project resources that include sub-contractors, partners, internal IT staff, subject matter expert and business representatives. The project team will generally expand in number as the solution outline and system is progressively defined.

C. *Core Banking Architecture*

The next phase (core banking architecture) represents the commencement of end-to-end solution design. In practice, this may start earlier, and must take form during the early part of the program. The package selection, which includes the core banking platform, and also key sub-systems such as the integration middleware, database management system (often influenced by the existing core bank platform), and web application server technologies are also defined. It is also often the case that many of the selected packages will already be pre-existing components within the bank or will be determined based upon standard operating environment for development and production. However, some additional components and software licensing is almost always inevitable.

As the non-functional requirements are developed for response times, batch processing, and transaction processing the solution architecture will be established in collaboration with the infrastructure platform engineering team. A method such as a workload placement classifier will aide engineers and architect in the hardware platform decisions [11]. The data model and security architecture requires considerable thought at this time, contributing to the overall design. The key design element visible to stakeholders and users is the information architecture as the screen user interface design is carried out. Prototyping screens is essential to help consolidate the scope and although this is often viewed as out-of-the box capability, more often this is subject to significant change and customization to meet business expectations. At this time, scope will often increase considerably as new requirements (both technical and functional) begin to emerge. Hence, it is good practice to establish sound project management practices including change management early in the program, to ensure that any scope expansion is warranted.

D. *Solution Delivery - Macro Design*

The Macro Design phase expands upon the solution framework established during the architecture phase, with more detail added with an application architecture. In particular, for each major solution domain (i.e. loans, deposits, payments, origination) the service (API) definitions and core business logic are specified for the core banking platform. In addition, the integration bus service interface definitions, business (human) workflow process and user interfaces, and data access services are also defined. The systems integration design requires specification for integrating with other banking systems such as the payments network, general ledger, risk management and related distribution channels. Finally the security services such as authentication, authorization, privacy control, confidential session exchange, data leakage protection, and audit/logging for regulatory controls also form part of the integrated solution.

E. *Component Micro Design and Build Iterations*

Although the initial phases of the schedule resemble a more traditional waterfall solution development life-cycle the design and build phase activities will be more agile in nature with several iterations. Given the scale of a core bank modernization program it may be difficult to commence an iterative development approach prior to the architecture being refined to a reasonable and completed state. Tasks such as screen prototyping and showcases for certain distribution channels would be areas where early iterative development may commence. However, the logic for the business rules and API services has greater dependence on the banking organization and business process definitions.

F. *Testing Iterations, Environment, & Deployment*

While the delivery phases of the Testing appear later in the schedule, it is important to observe that test analysis and preparation commence as early as possible in the agile life-cycle, typically commencing during each design-build iteration. Deployment planning would also naturally commence immediately after the modernization roadmap, but

will be refined further during the initial work as the plans develop for system retirement, system replacement, or upgrade. Prior to final production deployment a consolidated system integration test will also be carried out.

VII. CONCLUSIONS, DISCUSSION, & FURTHER WORK

The ability to carry out a bank modernization program requires solid support from stakeholders and business sponsors. Experience has shown that such programs, as with any large transformation, will be subject to considerable pressure during delivery and will benefit from sound technical and project management practices. We have presented some strategies and lessons learned for core banking transformation and provided some detail to the type of activities involved. This is useful to both program teams and sponsors embarking upon similar work and it is hoped that these insights are useful in aiding delivery teams to complete their programs with reduced risk.

More work is required with approaches and techniques on co-existence between existing and the new platform being deployed. Often it is assumed that the existing system will simply just be retired and hence it is believed to be relatively straightforward. In practice however, it is often the case that the existing system needs to be held and retrained for some time until the new platform is able to gather enough functionality to replace the old system. This often results in significant costs to the businesses due to the need to support two systems that essentially perform a similar function. Thus, strategies and techniques for co-existence and facilitate system retirement may be useful.

There is also further work on studying the impact of cloud and multi-tenant solutions for core banking. Whilst cloud technology trends are viewed as candidate frameworks for other banking components, time critical systems such as core banking may find deployment models in cloud unacceptable to banking institutions and regulatory bodies in the short term. As these technologies improve and gain further trust with the enterprise, it may be likely that migration of key operating platforms to these cloud environments may be inevitable.

ACKNOWLEDGMENTS

We thank Zoran Ojurovic, Ann Valencic and Michael Aaron for their helpful comments and discussion on the content of this paper. This paper is an extended version of the conference paper that appeared in [12].

REFERENCES

- [1] "The Value of Availability," *The Availability Digest*, June 2011. [Online]. Available: www.availabilitydigest.com/public_articles/0606/value_of_availability.pdf
- [2] Z. Ebadi, "Advance Banking System Features with emphasis on Core banking," in *The 9th International Conference on Advanced Communication Technology*, IEEE, February 2007, pp.573-576.
- [3] J-H. Wu, "Core Capabilities for Exploiting Electronic Banking," *J. Electronic Commerce Research*, vol. 7, no. 2, pp.111-122, 2006.
- [4] S. Jafar Hussain, M. Kumar. S. Jamal Hussain, "Meta Architecture to Support Layered Banking Systems," in *2nd IEEE International Conference on Emerging Technologies*, Pakistan November 2006, pp.754-760.
- [5] R. Liua, F. Wua, Y. Patnaikb, S. Kumaranb, "Business Entities: An SOA Approach to Progressive Core Banking Renovation," in *2009 IEEE International Conference on Services Computing*, pp.466-473.
- [6] D. Beimbom, J. Franke, H-T. Wagner, T. Weitzel, "The Influence of Alignment on the Post-Implementation Success of a Core Banking Information System: An Embedded Case Study," in *Proceedings of the 40th Hawaii International Conference on System Sciences*, 2007.
- [7] E. Metwally, "Implementation of IT-based Delivery Systems in Union Trust Bank of Egypt," in *2010 International Conference on Education and Management Technology (ICEMT 2010)*, pp.645-649.
- [8] A. Malan, L Pretorius, JHC Pretorius, "Managing the Implementation of Banking Systems for Repeatable Success," in *PICMET 2008 Proceedings*, 27-31 July, Cape Town, South Africa (c) 2008 PICMET, pp.2366-2375.
- [9] A.P. Noble, R. Kopae, A. Melek, and N. Nandy, "Data Leak Prevention," ISACA White Paper, 2010.
- [10] L. Zhang and Y. Wang, "The Banking System Reference Implementation Based on SOA," in *the Fourth International Joint Conference on Computational Sciences and Optimization (CSO)*, April 2011, pp. 374-377.
- [11] T. Rojahn, R. Lebsack, C. Pavlovski, "Towards a Computing Workload Classifier," *IBM J. Research and Development*, Vol. 58, November 2014, To Appear.
- [12] J. Kilimnik and C. Pavlovski, "Core Banking Modernization," in *the 9th International Conference on Information Technology and Applications (ICITA)*, July, 2014, Sydney, Australia.