

P E R S P E C T I V E

Phase II of the Core Transformation Journey: Getting There



YOUR INNOVATION PARTNER

An earlier paper discussed ways to secure the buy-in of top management and key stakeholders for a core transformation decision. In this, we touch upon some of the IT challenges during and after transformation.

Typically, banks have built layers of IT infrastructure, from mainframe and client server applications to ERP systems and custom applications, creating an IT snarl beset with problems of cost, rigidity and lethargy. Ensuring the peaceful co-existence of new applications with the old is one of the challenges of core transformation. Another is the minimization of risks during data migration. All this while, the bank has to ensure that business is disrupted as little as possible, preferably not at all.

Getting Applications to Co-Exist

The way to get new and legacy applications to communicate is by defining the right enterprise architecture, which ironically involves adding a layer of Service Oriented Architecture (SOA) on top of the others.

The success of SOA adoption hinges on the robustness of the bank's IT policy and process framework. Putting check points where necessary and automating at least some part of enforcement improves the framework's effectiveness. These policies determine how applications must interact, share functionalities and generally behave with one another. Yet again, technology provides the tools to define these policies, and once the rules are established within the interfacing infrastructure, all systems automatically comply.

That apart, middleware technology can enable existing applications to fit into the revised enterprise architecture by giving them a "facade" to communicate with new applications. Thus, even non-standard legacy applications can interact with new systems in a standardized manner.

Mitigating Risks of Data Migration

The risks of data migration manifest in several forms, as described below:

Quality issues with existing data: When the quality of existing data is suspect on account of duplication, inaccuracy or incompleteness, it

is advisable to cleanse it prior to migration in order to avoid the same flaws being carried over to the new system. Corrective options include data quality tools, simple manual effort or assumption of default values until the right ones become available.

Incomplete migration of existing data: To prevent partial data migration, a complete data dictionary of the existing system is needed to match all data elements with the corresponding ones in the new system. Auditing and testing with migrated data through one or more simulation runs prior to going live is highly recommended. Reverse mapping of data from the new system to the existing one works as a secondary check.

Quality issues in migrated data not seen in old data: Quality issues cropping up in the new system on account of the following reasons are trickier to handle:

- a. A bug in the new system
- b. Incorrect translation of old data into a new format for the new system
- c. Incorrect mapping of migrated data to data elements of the new system – for example, a mismatch between the totals on the internal counters of the two systems
- d. Wrong assumption of default values for additional data elements required in the new system
- e. New system highlighting errors that were masked by the old one

Again, there is no magic bullet for any of the above; mitigation is possible only through painstaking review, audit and testing of migrated data. Also, the efficacy of data migration improves by having multiple iterations of data migration during the testing phase and using specialized migration tools rather than relying on manual processes.

Different results reported by the new system despite data being correct: Sometimes, the new system throws up different results from the old one, in spite of there being no variation in the migrated data. This is likely due to a difference in the way data is processed by the two systems or the existence of a bug. Thorough testing with migrated data is the best preventive action.

Lack of proper control over migration scripts and programs: Often, migration scripts, programs and tools are not subject to the same stringent controls that ensure correctness of version, patches, upgrades etc., as the main system. This slackness is because of the wrong assumption that these tools which are meant for one time use do not need rigorous control like the main system which will be used repeatedly. Often, testing may be done with one version of a migration script and actual migration with another, without anyone detecting the discrepancy. Clearly, data migration tools, scripts and programs demand the same controls and checks as any other software.

Adopting data migration best practices is recommended for the reasons described below:

- Although technical implementation can be handled by the IT department, business users must participate in review and testing, since they are the actual owners of the data and therefore understand it best
- Data migration should be treated not like a parallel activity, but as an integral part of the main transformation program. The data migration strategy and plan must be decided upfront. Any modifications to be made during implementation should be subject to stringent change control, like any other aspect of the program
- There must be a clear cutover strategy for migration, with the approach and limitations - such as the cutover window, reconciliation approach / window, fallback strategy and channel cutover strategy - highlighted in order to prevent business disruption

Ensuring Business Continuity

Besides data migration, there are several sources of risk to business continuity - unexpected events, systemic failure or a failed core transformation - each of which must be managed.

In theory, a perfectly planned and executed core systems transformation should pose no threat to

the running of operations. Unfortunately, this is a less than ideal world in which problems creep up from time to time. That being said, the risks associated with banking transformation can be mitigated to a great extent by following these best practices:

Taking a phased risk-managed approach: By opting for phased implementation instead of big-bang transformation, banks give up quicker and larger financial benefits in return for less risk and lower overall gains accrued gradually over the implementation lifecycle.

Having a risk mitigation plan: Whereas risks are usually identified at the outset of any transformation, mitigating actions rarely are, leaving banks floundering in the wake of disaster. Therefore, it is crucial to have a mitigation plan laid out for every significant risk.

Testing thoroughly: Likewise, a testing plan must be drawn up right at the beginning and executed preferably with migrated data. Conducting multiple simulation runs across various lifecycle events further reduces the risk.

Right-sizing infrastructure: Over-sizing may entail additional expenditure, but under-sizing is a recipe for disaster. While transforming their core systems, banks need to ensure adequate infrastructure to support the intended scale of business.

Performance and stress testing: Performance and stress testing, especially under peak load utilization helps banks to assess the readiness and stability of their infrastructure.

Monitoring proactively: Banks must establish alert and monitoring mechanisms to detect impending problems well in time during the core systems implementation lifecycle.

Auditing regularly: While pre-implementation audit is necessary to highlight issues that might crop up on going live, post-implementation audit could send out an early warning signal of impending problems after switchover.

Conducting preventive maintenance and upgrades: This improves reliability and minimizes chances of failure.

Having a rollback plan: In the extreme case of total failure after switchover, it may be necessary to revert to the old systems. Hence, a rollback plan is essential.

Transformation risks apart, business may be disrupted on account of natural calamity or hardware / software failure. The way to defend against this is through effective disaster recovery planning and execution.

Ideally, banks must have an “active-active” disaster recovery infrastructure plan, which enables automatic switchover to backup systems configured to deliver a predetermined level of service; however, the financial and technological requirements may render this infeasible. In the absence of such an arrangement, manual intervention will be required to restore normalcy. While this implies a break in service, the extent of disruption can be minimized with advance planning.

The risk of hardware or software failure can be mitigated by installing alternative resources which take over when the main instances go down – for example, having an additional web server helps maintain continuity when there’s an outage of the primary server. Although this creates hardware redundancy, it also imparts scalability over the longer term. Similarly, software redundancy can be built through applications that clone themselves, so that when one instance fails, another takes its place. Such practices eradicate single point sources of failure in both hardware and software to ensure business continuity.

24x7 solutions with the in-built capability to switch to a stand-in application / offer a graded level of fallback in the event of failure provide additional defence against business disruption. Ideally, all business critical applications must have this feature. When applications are not 24x7 enabled, it is possible to compensate that through third party solutions – for example, when the functioning of a web server is interrupted, a third party solution could redirect web traffic seamlessly to alternative servers.

Of course, the best safeguard for the bank is to be able to switch over to a completely manual mode and still provide essential banking services when all systems fail, much like the airline industry which issues manual boarding passes if computer systems fail, so that customers may still fly.

Last but not least, banks must routinely test their disaster preparedness plans even when it is not mandatory to do so.

Author

Balwant C Surti

Head - Solutions Architecture and
Design Group, Finacle
Infosys Technologies Limited



YOUR INNOVATION PARTNER

PERSPECTIVE

Universal Banking Solution | System Integration | Consulting | Business Process Outsourcing

Infosys Technologies Limited

Plot No. 44, Electronics City, Hosur Road, Bangalore - 560100. India
Tel.: + 91 80 28520261, Fax: + 91 80 28521747, e-mail: finacleweb@infosys.com
www.infosys.com/finacle

Join us on Twitter, LinkedIn and Finacle Whiteboard at www.infosys.com/finacle/networking.asp

*COPYRIGHT NOTICE: Copyright ©2009 Infosys Technologies Limited, Bangalore, India. ALL RIGHTS RESERVED.
Finacle logo is a registered trademark of Infosys and Infosys acknowledges the proprietary rights of the trademarks and product names of other companies mentioned in this document. Infosys believes the information in this publication is accurate as of its publication date; such information is subject to change without notice.