

Migrating a retail banking platform



BBD has leveraged its nearly 40-year partnership with a leading African bank to migrate their retail banking platform legacy application to a new technology stack, by making use of service-orientated architecture to create a new omni-channel solution.

Objectives

- › Deprecate the banking platform legacy applications and replace with new technology
- › Agnostic, new channel user interfaces for staff, across multiple channels
- › Optimise architecture and design to establish a solid foundation for speed to market
- › Standardising apps to rationalise multiple apps into one app for a single view

Benefits

- › Supported technology stack
- › Improved systems and operating stability
- › Enhanced performance capability and reliability
- › Simplified interfaces for enhanced user experience
- › Support and maintenance costs lowered

Overview of the solution

Replacing core legacy systems are a bank's worst nightmare and many will do so in a "big bang" project, built separately from the legacy application. Due to the cost of legacy projects and many legacy replacements failing, bank leadership are loathe to undertake projects of this scope. Hence at BBD we look at how we can evolve legacy applications, which may take a bit longer, but deliver results which are less disruptive and with a higher success rate.

To transform this bank's existing retail banking platform system, BBD is utilising new technologies. This will allow for the banking platform's user interface to transform by deploying new preferred technologies, such as HTML5 and Angular, in a phased approach. Back-end components are also being evolved as part of the scope. With many banks moving to a digital, self-service banking model - the app BBD is morphing could also be utilised by the bank for web, mobile and digital instances.

Where possible, BBD is reusing existing services to ensure cost efficiencies. Layered architecture, in line with best practices, are being implemented in such a way that front-ends can be decoupled from the service layer. Using multiple front-end applications to consume these newly developed services have resulted in standardised business capabilities across multiple channels.



| Approach

Through a vertical transformation approach, BBD has been able to slice off full end-to-end functionality across the application, ensuring business capability can be migrated as a whole i.e. enhanced and optimised functions in the new tech. Through the use of a vertical approach, BBD has ensured minimal impact on the business. Once all functionality is transformed, BBD will decommission the old framework and components.

The existing banking platform user interface is being transformed from its current VB6 based controls into HTML5 web pages using an Angular framework, through a phased approach, replacing one OCX at a time. Scaffolding is being developed to allow for seamless integration between the new web pages and the old framework, till all old components are transitioned. At this stage the banking platform framework, as well as the scaffolding, is being removed. The end state will allow for these HTML5 pages to fall in line with the current technology strategy of the bank.

Many of the back-end C++ components have been transformed into C#. For the first phase of this project the remaining C++ components are being wrapped with a C# proxy, enabling them to be exposed as services, falling in line with service orientated architecture. As part of BBD's project scope definition, the C++ components have been earmarked for re-factoring in either C# or Java. The dependency on HIS and CS5 have been removed by enabling direct communication with services exposed on the ESB.

Business and data logic are being extracted from the current components into newly developed services, exposed on the ESB. These have become available for consumption by disparate applications and channels, ensuring they are applicable across the enterprise. At the end of the project the bank's dependency on their previous retail banking platform will be completely severed.

| Impact of BBD's partnership

While this project is still a work in progress, the concept of morphing the legacy technology stack into a new technology stack has been proven. HTML5 pages are currently co-existing with legacy VB6 components, while C++ and C# components are co-existing and communicating with both the legacy and new technology stacks.