OpenLegacy
Digital to the Core

Case Study

Major bank embarks on a digital transformation project with OpenLegacy

Took only one hour to create a new API, and just two weeks for 6 new business workflows

Financial | Tandem Mainframes | Digital Banking | Microservices

Founded in 1884, the bank is the third-largest in its country of operations and part of a multi-national financial services corporation. With 40,000 employees and tens of millions of customers, the bank processes over 40% of total transactions in the country. Witnessing the digital transformation happening around it, the bank committed to a digital transformation of its own.

The Challenge

The foundation of this transformation process is the ability to leverage—and extend—the bank's current technology stack. At the heart of it is a proprietary mainframe system, which runs the bank's core business applications such as payments and account management. Relying on this tried-and-tested technology backbone, while layering on new services and applications in an agile, flexible, standard fashion is the key to a successful digital transformation.

As early as a decade ago, the bank realized the need to open up their various IT platforms and applications so they can share data and integrate business processes. To that end, it even instituted a policy of opening up business flows as APIs.

Over the years, numerous vendors—including some of the biggest legacy transformation software vendors in the world—have tried to open up their legacy applications and make them available for the rest of the bank's IT environment. All of these attempts achieved various levels of success, but to date, none could be considered successful. These modernization projects were complex, slow, and non-scalable. Even worse, they led to an ever-increasing spiral of complexity with a bevy of in-house tools, mainframe gateways, ESBs, middleware, dispatchers, routers, messaging queues, and the list goes on. Each system had a different way of handling the backend legacy platform, which ironically made it harder and more cumbersome to integrate with legacy applications after each such "integration" project had finished.

This level of complexity and rigidity is not uncommon among enterprises. Legacy systems at their core are valuable assets, but they are trapped behind layer upon layer of integration implementations.

> *After years of mainframe modernization projects that didn't realize their full potential, OpenLegacy's API integration platform is the light at the end of the tunnel. Within an hour, we went from nothing to a working API connected to our payments app, powered by OpenLegacy.*

Following years of unsuccessful, costly integration attempts, the bank was left with the same needs and challenges—only exacerbated: No way to upgrade or update the mainframe environment, an urgent need to better use the mainframe to increase profitability and return on investment, and strong demands from business stakeholders to open up legacy applications and integrate them within new, emerging company initiatives such as the Customer Digital Experience and the End-User Journey.

The Solution

The bank approached OpenLegacy to expose all of their core banking applications, including Account Opening, Checking and Debit Accounts, Consumer Credit (Personal, Autos, Mortgage), Payments and Investments. They consist of hundreds of business processes that handle all aspects of most consumer applications, including utility and credit card bills, tax payments, consumer products payments (e.g. car, mortgage) and more.

In the current environment, the bank's tellers used either the mainframe's "Green Screens" directly, or internal apps that were very clunky and unstable as a result of previous "spaghetti" integration projects. Exposing the application's workflows as APIs would pave the way to building new, intuitive user interfaces for these workflows, serving both the teller and later on the consumer.

OpenLegacy's connector for the mainframe environment, combined with its COBOL parser and automated API generator, resulted in a working prototype exposing six real-world mainframe business flows (e.g. checking balance, account validation) as APIs. The prototype only took two weeks to complete. It completely bypassed all existing middleware and messaging technologies, and instead connected directly with the mainframe environment: From analyzing back-end code and automatically generating APIs from it, to invoking transactions and processing input and output.The new APIs generated by OpenLegacy were tested and deployed using the bank's existing environment, to ensure that they function properly in the current infrastructure as well as follow the bank's internal integration standards.

The development was done using modern, scalable, open architecture (such as REST API) and development methodologies (Agile, Scrum)—complying with the bank's requirements.

The Result

Incredibly fast time to market

With OpenLegacy, creating an API for one business flow (eg checking an account's balance, validating an account) took less than an hour—compared to 1-2 months of development time in the old setup.

Unleashing digital transformation

Now that the API building blocks are in place, the bank's business units can dream up—and realize—unlimited new consumer services, offerings, and products. One example is a new Student bundle that includes products and services geared towards the student market: Car and tuition loans, discounted fees on checking and savings accounts, and a credit card with special benefits. Previously, developing such a product was prohibitively labor intensive and expensive.

> *OpenLegacy helps us set the mainframe free, finally. Instead of being a prohibitive factor for new projects, it's now the enabler—with OpenLegacy-generated APIs that connect to any other system in our environment.*

Significantly lower cost of development with a minimal risk solution

For application developers, OpenLegacy is a game-changer. The simple, elegant OpenLegacy platform allows any Java developer within the organization to call mainframe business workflows directly in order to add new functionality or expose existing one to external applications. There's no need to go through layer upon layer of connectors, ESBs, and middleware, to re-write COBOL code, or create a new legacy environment. Deployment is also a breeze with OpenLegacy, freeing up expensive DevOps resources: OpenLegacy automatically and instantaneously deploys new versions—compared to the current environment that requires over a week to deploy.

Reduced complexity and TCO

At run-time, the complexity and TCO are reduced because the OpenLegacy-generated microservices run in the cloud and communicate directly to the mainframes. This means the application doesn't have to communicate through expensive middleware.

About OpenLegacy

OpenLegacy's Digital-Driven Integration enables organizations with legacy systems to release new digital services faster and easier than ever before. Connecting directly to even the most complex core systems, OpenLegacy automatically generates the digital-ready components needed to integrate legacy assets into exciting new innovations. With OpenLegacy, industry-leading companies release new apps, features, and updates while spending a fraction of the time and resources, so they quickly and easily become digital to the core.

© OpenLegacy Inc. All Rights Reserved
CS_MajorBank_7Apr2020