opentext™

Case Study

Large European Bank

Leading European bank reduces complexity and cost with Service Virtualization, which simulates real environments to speed and enhances testing.

Challenge

A large European bank recently worked with OpenText™ to deploy a solution that significantly enhanced testing while decreasing complexity. The bank serves retail, private, and corporate banking clients with a focus on Europe and selective operations internationally. Clients rely on a comprehensive range of products and services offered through multiple channels, including advanced mobile applications and Internet banking.

The bank wanted to improve its testing quality and continuous delivery and integration capabilities. With help from OpenText, it found that the best way to accomplish this was to virtualize third party and internal services, testing software with virtual services that simulate real-world behavior.

Here are the specific challenges the bank faced:

- Testing teams were heavily dependent on the SWIFT banking protocol.
- SWIFT is expensive—charging .2 Euros per transaction—which became very costly when approximately 6,500 messages were being sent via SWIFT each month.
- Regression and performance testing teams were using the SWIFT interface extensively in each release.
- The bank needed to enable SWIFT based testing throughout its infrastructure.

Solution

By relying on Service Virtualization, the bank has been able to create more than 150 virtual services that decrease complexity and enhance results for different business teams across the bank's locations. They discovered that virtualization could be used to simulate complex financial networks and backend systems (including the vast SWIFT network for inter-bank exchange) whenever the bank needs to test or develop its internal systems in a truly independent manner.

Service Virtualization helps the bank:

- Deploy new services quickly, and making them always available
- Simulate backend services that do not yet exist
- Simulate complex bank systems while coding a new service (no need to set up difficult-to-maintain test environments)
- Enhance performance testing and optimization

Management at the bank uses the Service Virtualization web interface to extract statistics and monitor how service virtualization implementation progresses across the bank's infrastructure.

At a Glance

- **Industry**
  Financial Service
- **Location**
  Europe
- **Challenge**
  Streamline development and testing, decrease complexity, and cut costs.
- **Products and Services**
  Service Virtualization
- **Success Highlights**
  + 30% of defects can now be detected early, in pre-production environments
  + Significantly reduced the time to identify performance bottlenecks
  + Improved continuous integration and delivery
  + Substantial direct savings of euros per year

"Micro Focus (now part of OpenText™) Service Virtualization is a very good solution because it's easy to use and has a simple interface. It was a great effort by Micro Focus (now part of OpenText™) to come up with a solution that decreases dependency on SWIFT and FIX and supports the needs of financial institutions."

IT MANAGER
Large European Bank

Connect with Us
OpenText CEO Mark Barrenechea's blog

Results

Making Services Always Available

The bank has its own dedicated Services Virtualization team, tasked with maintaining all virtualized services and making sure they are always available for other teams to use (or to download and deploy wherever they need). Other teams focus on business-specific areas such as credit cards, mortgage lending, trading, and so on. Quality assurance and development teams in these areas develop virtualized services internally, on their own. Once an internal developer creates a prototype of a new virtual service, they hand it over to the dedicated Services Virtualization team to finalize, deploy, and maintain the service.

The bank is able to deploy new virtualized services easily and quickly because you don't need any specialized technical skills to use Service Virtualization—even for tasks such as creating a mocked IBM mainframe.

Simulating REST Backend Services

Developers of web applications need to code their functionality before backend services even exist. With this in mind, the bank uses service virtualization to create a simulated virtual model of future REST services. They rely on Service Virtualization to mock up not-yet-existing backend services without the need to spend time coordinating with the backend team or understanding backend inner workings in detail.

Simulating Complex, Legacy Banking Systems

There are several complex systems in banks that are difficult to include in test scenarios. One example is the IBM IMS mainframe—a legacy system that requires a good amount of knowledge to set up even a primitive test. Instead of dealing with the real IMS installation, the bank's developers use Service Virtualization as a "go-between" to first intercept all communication between IMS and other systems and then use the recorded data to build a mockup of the IMS system.

Performance Testing and Optimization

When you tune the performance of a service, you can't simply measure what would happen if one component runs 20% faster. It's an existing component, and you can't make it faster without coding, deployment, and/or replacing it. But if you use a virtualized mockup of such a component, you can measure the whole system and get much better information for further analysis. In the bank's case, it can use Service Virtualization to orchestrate the collaboration of several complex bank systems and test what happens to the whole complex if one of the systems has performance issues.

For more information on Service Virtualization, please visit: **www.microfocus.com/sv**

Learn more at
www.microfocus.com/opentext

opentext™