



UniCredit modernizes mainframe development and testing with IBM Rational Developer for System z software.

Overview

■ Challenge

Looking to modernize its development platform for core financial business software, UniCredit Group's HVB IS sought to improve development and testing efficiency for new and existing applications.

■ Solution

More than 200 developers at HVB IS began using IBM Rational Developer for System z software to develop mainframe applications for IBM System z servers.

■ Key Benefits

HVB IS can now use a similar development environment for COBOL and Web applications. The new environment is helping the organization attract younger developers, while developers currently using Rational Developer for System z are seeing an estimated 15 percent increase in development productivity and 10 percent increase in testing productivity.

With more than 170,000 employees, 40 million customers and 9,000 branches, UniCredit Group is one of the largest financial services organizations in Europe. When UniCredit Group merged with HypoVereinsbank (HVB), the HVB Information Services (HVB IS) organization continued its mission of developing IT solutions for modern banking operations. Today, HVB IS serves two major core business segments: application development and IT operations and services.

HVB IS develops core financial business applications that run on its mainframe platform, which includes IBM System z® servers with IBM DB2® and IBM Information Management System (IMS™) software. The group is also responsible for maintaining legacy COBOL applications, many of which have been performing reliably for 15 years or more.

HVB IS adopted IBM Rational® Developer for System z software to accelerate mainframe development, improve the efficiency of its COBOL development teams, and move to a similar environment for mainframe and Web development. Almut Geiger, product specialist at HVB IS, explains, "We wanted to modernize our mainframe development, increase efficiency in the development of mainframe programs and become more efficient in testing those programs. Another goal was to make mainframe and COBOL development more attractive to younger programmers. And, while we have more and more mainframe work, we are also doing Java™ development. In the past, the development environments for these efforts have been very different. We wanted to move to a single environment for developing, debugging and testing batch and online applications. For these reasons, we selected Rational Developer for System z."

UniCredit modernizes mainframe development and testing with IBM Rational Developer for System z software.

Key Components

Software

- *IBM Rational Developer for System z*
- *IBM DB2*
- *IBM Information Management System (IMS)*

Hardware

- *IBM System z*

“We are working more and more with UniCredit’s Italian teams. At the moment the only way to debug our code on the Italian mainframe is with Rational Developer for System z.”

– Almut Geiger, product specialist at HVB IS

Modernizing the development environment

While some programmers at HVB IS use the IBM ISPF for z/OS® tool set to develop and document batch programs, approximately 200 HVB IS developers now use Rational Developer for System z as a common workbench with integrated tools to support end-to-end development. As an Eclipse platform–based integrated development environment (IDE), Rational Developer for System z provides HVB IS developers with a modern development platform and advanced development tools. “Programmers that are using Rational Developer for System z have found that development is more comfortable than with our existing set of development tools,” says Geiger. “Debugging in particular is much faster. Features like syntax checking, content assist, multiple views and error identification make development easier. For example, when there were coding errors before, a developer had to find the correct line number in the code and then try to determine what was wrong with the code. Now they just click on an icon, and they are led directly to the position of the error.”

Programmers can also debug by setting breakpoints, monitoring variable values and stepping through the execution of a COBOL application one statement at a time. “This makes debugging much easier,” says Geiger. “And it is much easier to learn because you don’t have to memorize commands to be productive.”

Some HVB IS projects would be significantly less efficient without Rational Developer for System z. Explains Geiger, “We are working more and more with UniCredit’s Italian teams. At the moment the only way to debug our code on the Italian mainframe is with Rational Developer for System z.”

A customizable solution

HVB IS tailored the Rational Developer for System z environment to meet its team’s specific needs. It used Common Access Repository Manager (CARMA), which Rational Developer for System z uses to establish an interface with its software configuration manager (SCM). “Using the CARMA interface, we can browse, edit, compare, check out and check in code directly from within Rational Developer for System z,” notes Geiger.

She adds, “We also have some very old COBOL code in-house with specialized data structures and macros. We customized the parser and remote syntax checks to handle these situations. Because Rational Developer for System z is based on Eclipse, an open source platform, it is easy to enhance and customize. This is one of its greatest advantages. Also, many of our young developers are already familiar with Eclipse from their university courses or from just using it at home. As a result, they learn to use Rational Developer for System z very quickly.”

Increased development and testing efficiency

At HVB IS, Rational Developer for System z has helped improve development and testing efficiency for UniCredit’s financial systems and applications. Geiger explains, “The biggest advantage of Rational Developer for System z is the modernization of mainframe development. Many mainframe developers have worked without an IDE for a very long time—there have been few improvements to their development platform. Of course, it takes people some time to get familiar with a new environment, but when they become familiar with Rational Developer for System z, we are seeing an estimated 15 percent increase in development efficiency and a 10 percent increase in testing efficiency.”

HVB IS is now also better positioned to move toward a service-oriented architecture (SOA) in the coming years. HVB IS teams already working with IBM Rational Software Architect technology to model and build Web applications will work more closely with Rational Developer for System z development teams. “So far, we have focused on developing financial applications with Rational Developer for System z. Over time, we will use it for building Web applications and Web services for SOA. And we plan to use templates in Rational Developer for System z to increase reuse,” Geiger concludes. “Our team now develops, debugs and tests with more confidence, and it is clear that more and more people here will be using Rational Developer for System z.”

For more information

To learn more about IBM Rational Developer for System z, contact your IBM Business Partner or IBM representative, or visit:

ibm.com/software/awdtools/rdz/

“The biggest advantage of Rational Developer for System z is the modernization of mainframe development...it takes people some time to get familiar with a new environment, but when they become familiar with Rational Developer for System z, we are seeing an estimated 15 percent increase in development efficiency and a 10 percent increase in testing efficiency.”

— Almut Geiger, product specialist at HVB IS



© Copyright IBM Corporation 2008

IBM Corporation
Software Group
Route 100
Somers, NY 10589
U.S.A.

Produced in the United States of America
September 2008
All Rights Reserved

IBM, the IBM logo, ibm.com and Rational are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at ibm.com/legal/copytrade.shtml

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc., in the United States, other countries, or both.

Other company, product and service names may be trademarks or registered trademarks or service marks of others.

The information contained in this documentation is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this documentation, it is provided "as is" without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this documentation or any other documentation. Nothing contained in this documentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM (or its suppliers or licensors), or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

This document illustrates how one organization uses IBM products. Many factors have contributed to the results and benefits described; IBM does not guarantee comparable results elsewhere.

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates.