

Industry:

Financial Services,
Insurance

Organization:

A leading global systems
integrator

Description:

One of the world's largest business consulting, systems integration and managed services firms serving Global 2000 companies, medium-sized businesses, government agencies and other organizations.

This company provides business and technology strategy, systems design, architecture, applications implementation, network, systems integration and managed services.

The company's service offerings are designed to help clients generate revenue, reduce costs and access the information necessary to operate their business on a timely basis.

Business Problem:

The company was reengineering and developing an insurance claims system and needed to increase productivity and leverage existing development skills.

Solution:

IBM Rational Rapid Developer
IBM Rational Unified Process (RUP)
IBM WebSphere Application Server

Key Benefits:

Increased developer productivity by 40 to 50 percent. Enabled developers with little J2EE experience to work effectively on complex *n*-tier systems

Opened new business opportunities by using a technology-neutral development platform that supports a wide range of application servers, Web servers, databases and messaging transports.

Accelerated and streamlined the build-deploy-test cycle

Promoted reuse, compliance and knowledge sharing by creating and using code patterns

Rapidly integrated with back-end systems using comprehensive support for XML

Rational® software

Leading Global Systems Integrator Accelerates Development and Expands Business Opportunities Using IBM Rational Rapid Developer

Industry leaders in business consulting, systems integration and managed services are always looking for ways to accelerate the delivery of high-value solutions to their clients.

When one such world leading company began a development effort to reengineer and extend a property and casualty claims system for the insurance industry, the development team used IBM® Rational® Rapid Developer to increase productivity, speed development and pursue new business opportunities. Rational Rapid Developer is an architected rapid application development (ARAD) environment that masks the underlying complexities of *n*-tier development, allowing the system integrator's development team to rapidly design, construct and maintain reliable, scalable business applications.

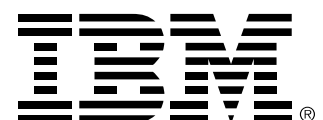
The system integrator's claims system development team has realized several key advantages since adopting Rational Rapid Developer, including increased developer efficiency, improved reuse and standards adherence, simplified project management, and rapid construction and deployment to a range of target environments. Taken all together, these advantages add up to considerable time savings. The company's managing director notes, "As a conservative estimate, we're seeing a 40 percent to 50 percent productivity gain. Without Rational Rapid Developer, I think we would have to add at least six months to the schedule."

From a business perspective, Rational Rapid Developer provides the company with another important advantage. Prospective customers for the claims system include hundreds of insurance companies, including most of the top 100 insurance companies, each with their own preferred platforms, application servers, databases, messaging transports, and existing systems and other IT assets. Because Rational Rapid Developer supports a broad spectrum of

target deployment environments, the development team will be able to rapidly and automatically reconstruct and redeploy the claims system to meet each prospective customer's technology requirements. Rational Rapid Developer will also enable the company's customers to maintain their own version of the claims system with in-house development teams, without requiring a team of J2EE experts. "In addition to accelerated development, one of the biggest benefits of Rational Rapid Developer for us is that it is technology-neutral, and it makes interfacing with other applications much simpler," says a senior manager on the project. "From a maintenance point of view, the ability to hand off our system to clients is another key advantage of Rational Rapid Developer."

High Impact Value for Clients

On the property and casualty claims system project, the system integrator is demonstrating its ability to create measurable, high-impact value for its clients — in this case a leading provider of insurance and other financial services. "We're partnering with a client — a well-recognized Fortune 50 company — to take their existing, client-server-based claims system, and redevelop it as a J2EE *n*-tier application. Our client had successfully implemented this client-server system and had done thorough feasibility studies with their claims adjusters and call centers. A great deal of thought went into the screen flow and the navigation to enable them to capture more information and settle claims more efficaciously. They brought us in because of our expertise in doing analysis in the insurance market and in being able to develop J2EE applications with our architecture and methodology. Once it is completed, our client will begin using the new J2EE solution, and we will market it as a solution to other insurance providers," the senior manager explains.



“As a conservative estimate, we’re seeing a 40 percent to 50 percent productivity gain. Without Rational Rapid Developer, I think we would have to add at least six months to the schedule.”

– Managing Director,
Global Systems
Integrator

Proof-of-Concept Nets Productivity Gains

Before the system integrator’s development team considered Rational Rapid Developer, they performed a feasibility study of their own to ensure that the project made sense from business and technical perspectives. The senior manager recalls, “We did an initial proof-of-concept to make sure that the architecture was sound and that it was a feasible project. We also wanted to ensure that there would be a receptive market for it, so we talked to many of the top 100 players in the insurance market for property and casualty. The proof-of-concept was a subset of the overall functionality. It included about 13 use cases or about five percent of the total application. While developing the proof-of-concept we came across many of the complexities of developing a J2EE application, specifically all the skills that were necessary — EJBs, servlets, JSP, database, and user interface/front-end. We found that it made it challenging to manage the project, because acquiring resources with these skills caused bottlenecks at specific points in the project. For example, there were points at which we would be waiting for the UI developer to finish his screen before we could test certain functionality. So we started looking for a more model-driven approach.”

At that point, the development team decided to evaluate Rational Rapid Developer. The senior manager continues, “We were fairly skeptical going in, because in our experience with CASE (Computer Aided Software Engineering) tools in the past, the end result was not always what was promised. We decided to redo our initial proof-of-concept using Rapid Developer to see how it would compare in terms of the code that was generated, how quickly the developers could adjust to the tools, and how flexible it was in being able to change platforms. Because we were developing a commercial application we had to be able to deploy to different platforms using various application servers, Web servers, and database management systems. So we had to be flexible, and we wanted to be able to change those quickly. We were impressed with how quickly our developers were able to start using Rational Rapid Developer — and how quickly we were able to replicate the first version of the proof-of-con-

cept. We had built the original for IBM WebSphere® Application Server 3.5, but when we tried to upgrade it to WebSphere 4.0 we encountered problems in changing the code—we found it was actually quicker to redevelop the application using Rational Rapid Developer than it was to go in and hand-code the upgrade. Without Rapid Developer, the proof of concept required about eight people working for three months. With Rapid Developer it took less than four weeks with three developers.”

Iterative Development – A Match for the System Integrator’s Methodology

Rational Rapid Developer fits well with the iterative development approach of IBM® Rational Unified Process®, which is used throughout the company, as well as its own methodology. “We have our own methodology for integrating various legacy applications using middleware messaging,” says the senior manager. “We developed a methodology to quickly assemble systems using legacy applications, commercial software and custom development. There is an architecture designed around principles of integrating these applications using adapters, middleware and workflow integration. It also incorporates a methodology for developing requirements, using iterative cycles, and driving it all through return on investment (ROI). Rational Rapid Developer fits this methodology because it enables us to integrate applications very quickly. One of the advantages we saw right away was the ability to interface with applications through XML, which is a large part of the methodology’s architecture,” he says.

He continues, “With Rational Rapid Developer we have the ability to quickly transform from one back-end system to another just mapping the DTD (Data Type Definitions). We see that as a competitive advantage. When we implement a claim solution we expect every client is going to have a different policy administration system and we’re going to have to integrate with every one of those. With Rational Rapid Developer we can just map it to a DTD and that is going to allow us to do our installations more quickly. Time-to-market for these claim solutions will be faster and less expensive for our customers.”

Simplifying Project Management; Leveraging Developer Skills

Today's complex applications require expertise in a wide range of disciplines and technologies, including front-end or user interface development, XML, EJBs, J2EE infrastructure, legacy system integration, and a multitude of diverse application servers and databases. While the system integrator has many engineers on staff that possess these skills, bringing them all together on a project when and where they are needed is not easy in an environment with many concurrent projects. As project leader, the senior manager reports that Rational Rapid Developer greatly simplified the task of assembling a project team. Because Rational Rapid Developer automatically constructs J2EE infrastructure code across all tiers of *n*-tier applications, it enables consultants who are relatively new to J2EE development to become productive very quickly. "Rational Rapid Developer made staffing a great deal easier. Any time you try to staff a project, you need a certain balance of resources with different skill sets. It isn't always easy to get someone that is available at a specific point in time — especially in a consulting company where there are always projects going on simultaneously. Rational Rapid Developer really helped mask a lot of the complexities of *n*-tier development from the developers. Everyone was productive using the tool, even if they had little experience with J2EE development prior to working with Rapid Developer," he explains.

The Power of Patterns Drives Reuse and Standards Compliance

Just as Rational Rapid Developer helped the development team leverage general development experience on a complex J2EE project, it also helped amplify their individual expertise in specific areas of development and extend it to other team members. By developing code templates in Rational Rapid Developer, the company's developers are able to simplify and speed development tasks for their teammates. The senior manager explains, "Rational Rapid Developer allowed us to have a person who had specific expertise in one area develop a reusable template for us, and then provide instructions for all the other developers to implement it in the use case that they were

developing. For example, one developer had significant experience in middleware. He created a code template for integrating with WebSphere MQ (MQ Series) and through JMS, put it into a library, and then explained to everyone how to use it to make a middleware message-based call."

A senior consultant at the company asks, "How do you communicate across the team that these are the steps needed to use a particular component, this is the sequence, and these are the variables that you use? A crude method would be to have an oral or written explanation. A better way is to create a pattern in Rational Rapid Developer. Once you put it in a template in Rapid Developer, a developer can double click on it and all of the code generation is taken care of by the pattern that was fit into the template. All we have to worry about is what name to give it." The senior manager concludes, "That capability in Rational Rapid Developer leveraged all of the knowledge that the team had and it enforced standards. Rational Rapid Developer is also flexible — it allows us to include externally developed components, as well as reusable code templates, so we can just develop a library of functions and add them throughout the application."

Streamlining and Accelerating Builds and Deployment

With Rational Rapid Developer, the system integrator's consultants also have the ability to develop and perform preliminary testing using a lightweight application server, and then simply reconstruct and redeploy to their target environment for thorough system testing. The senior manager observes, "Rational Rapid Developer provided us with significant benefits in doing the builds and deployments. Previously, it was a much longer cycle for us to check in all the code, do the build to target the environment, deploy it, and test that the deployment was correct. Rational Rapid Developer removed much of the complexity from the process. We can deploy a single page and test it, or we can do the entire deployment and target different environments. Many developers deploy against their own laptops running a local version of Oracle's application server. Later they deploy to our testing environment, which includes IBM

"Given our success on this engagement, we are planning on leveraging Rational Rapid Developer on future J2EE development efforts, because this development platform is now a part of our best practices."

– Managing Director,
Global Systems
Integrator

WebSphere Application Server.”

“I feel the biggest advantage of developing with Rational Rapid Developer is that we spend more time on the high value business logic, rather than worrying about configurations, compilations, getting the patterns right, or adhering to the UI standards. We don’t have to worry about any issues that could surface if we redeploy to a different technology environment. That is a very, very big benefit. And with Rational Rapid Developer, distributed development is much simpler than in a standard J2EE development environment,” says the senior consultant.

New Best Practice Sets High Expectations for the Future

The senior manager jokes that there has been one drawback of using Rational Rapid Developer — his team is now expected to get more done in less time. “If anything, our biggest problem is that we’ve set new guidelines for how quickly we can prototype things and demonstrate them. Now that has become the expectation. Within the insurance industry, people are seeing how quickly we are able to deploy our systems and that has become the new standard for expectations from our team.”

The managing director adds that he expects Rational Rapid Developer to provide the company with a competitive advantage in upcoming engagements as well. “Given our success on this engagement, we are planning on leveraging Rational Rapid Developer on future J2EE development efforts, because this development platform is now a part of our best practices.”

About Rational

About Rational Rational provides a software development platform that improves the speed, quality, and predictability of software projects. This integrated, full life-cycle solution combines software engineering best practices, market-leading tools, and professional services. Ninety-six of the Fortune 100 rely on Rational tools and services to build better software, faster. This open platform is extended by partners who provide more than 500 complementary products and services.

Rational Software

Dual Headquarters

18880 Homestead Road
Cupertino, CA 95014

20 Maguire Road
Lexington, MA 02421

Toll-free: (800) 728-1212
e-mail: info@rational.com
Web: www.rational.com

International Locations:
www.rational.com/worldwide

IBM, the IBM logo, and WebSphere are trademarks of International Business Machines Corporation in the United States, other countries, or both. Rational and Rational Unified Process are trademarks or registered trademarks of Rational Software Corporation in the United States, other countries or both. Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. Other company, product or service names may be trademarks or service marks of others. © Copyright Rational Software Corporation, 2003. All rights reserved. Rational Software Corporation is a wholly owned subsidiary of IBM Corp.

Made in the U.S.A.

CS594A 5/03. Subject to change without notice.