BearingPoint Accelerates Development, Simplifies Project Management, and Expands Business Opportunities Using Rational Rapid Developer

Industry:
Financial Services, Insurance

Organization:
BearingPoint, Inc.

Description:
BearingPoint, formerly KPMG Consulting, Inc., is 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. BearingPoint provides business and technology strategy, systems design, architecture, applications implementation, network, systems integration and managed services. With global headquarters in McLean, Virginia, BearingPoint'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:
BearingPoint was reengineering and developing an insurance claims system for a client and needed a rapid application development platform to increase productivity and leverage existing development skills.

Solution:
Rational Rapid Developer
Rational Unified Process (RUP)

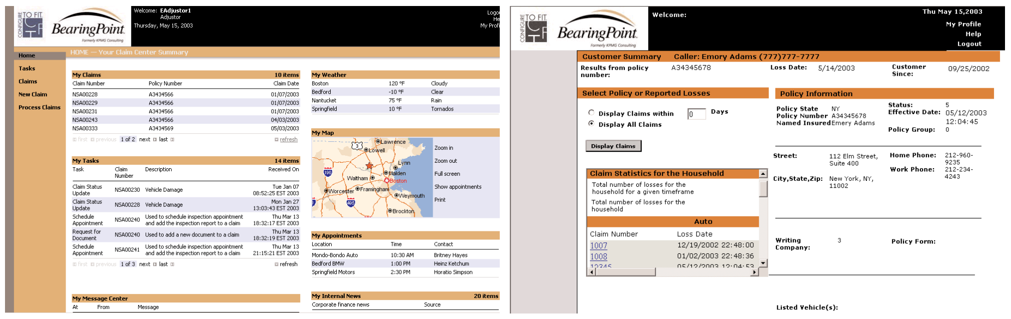
Key Benefits:
Increased developer productivity by 40 to 50 percent

Simplified staffing of key project by enabling developers with little J2EE experience to work effectively on a complex *n*-tier system

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

Saved time in day to day tasks by accelerating and streamlining the build-deploy-test cycle

Improved communication and promoted re-use by creating and using code patterns



Welcome: EAdjustor1
Adjustor
Thursday, May 15, 2003

Home
Tasks
Claims
New Claim
Process Claims

HOME — Your Claim Center Summary

My Claims — 10 items

Claim Number	Policy Number	Claim Date
NSA00228	A3434566	01/07/2003
NSA00229	A3434566	01/07/2003
NSA00231	A3434566	01/07/2003
NSA00243	A3434566	04/03/2003
NSA00333	A3434569	05/03/2003

first previous 1 of 2 next last refresh

My Tasks — 14 items

Task	Claim Number	Description	Received On
Claim Status Update	NSA00230	Vehicle Damage	Tue Jan 07 08:52:25 EST 2003
Claim Status Update	NSA00228	Vehicle Damage	Mon Jan 27 13:03:43 EST 2003
Schedule Appointment	NSA00240	Used to schedule inspection appointment and add the inspection report to a claim	Thu Mar 13 18:32:17 EST 2003
Request for Document	NSA00240	Used to add a new document to a claim	Thu Mar 13 18:32:19 EST 2003
Schedule Appointment	NSA00241	Used to schedule inspection appointment and add the inspection report to a claim	Thu Mar 13 21:15:21 EST 2003

first previous 1 of 3 next last refresh

My Message Center

At	From	Message

My Weather

Boston	120 °F	Cloudy
Bedford	-10 °F	Clear
Nantucket	75 °F	Rain
Springfield	10 °F	Tornados

My Map

Zoom in
Zoom out
Full screen
Show appointments
Print

My Appointments

Location	Time	Contact
Mondo-Bondo Auto	10:30 AM	Britney Hayes
Bedford BMW	1:00 PM	Heinz Ketchum
Springfield Motors	2:30 PM	Horatio Simpson

My Internal News — 20 items

Corporate finance news	Source

Welcome:
Thu May 15,2003
My Profile
Help
Logout

Customer Summary Caller: Emory Adams (777)777-7777

Results from policy number: A34345678 Loss Date: 5/14/2003 Customer Since: 09/25/2002

Select Policy or Reported Losses

○ Display Claims within 0 Days
[✓ Display All Claims]

Display Claims

Claim Statistics for the Household

Total number of losses for the household for a given timeframe
Total number of losses for the household

Auto

Claim Number	Loss Date
1007	12/19/2002 22:48:00
1008	01/02/2003 22:48:36
12345	05/12/2003 12:04:53

Policy Information

Policy State NY
Policy Number A34345678
Named Insured Emery Adams
Status: 5
Effective Date: 05/12/2003 12:04:45
Policy Group: 0

Street: 112 Elm Street, Suite 400
City,State,Zip: New York, NY, 11002
Home Phone: 212-960-9235
Work Phone: 212-234-4243

Writing Company: 3
Policy Form:

Listed Vehicle(s):

Caption info

Staffing a development project as a systems integrator can be a real challenge. In fact, project managers in all types of development organizations share a common dilemma. 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 some organizations have developers with these skills, they are often scarce commodities and bringing them all together on a project when and where they are needed is not easy.

At BearingPoint, Inc., Senior Manager Steven Edwards and his team have found an effective solution. On a recent project, the team used Rational Rapid Developer to leverage their existing development skills in new areas as they reengineer and develop a property and casualty claims system for the insurance industry. IBM® Rational® Rapid Developer™ is an architected rapid application development (ARAD) environment that masks the underlying complexities of J2EE development, allowing the BearingPoint team to rapidly design, construct and maintain reliable *n*-tier business applications. As project leader, Edwards reports that Rational Rapid Developer greatly simplified the task of assembling a project team. "Rational Rapid Developer made staffing a great deal easier. I no longer needed to find people who specifically knew EJBs or JSPs. Any time you try to staff a project, you need a certain balance of these resources. 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 competitive projects going on. Rational Rapid Developer really helped mask a lot of the complexities of J2EE development from the developers. They could be productive using the tool, even though only a few of the developers had experience with J2EE development prior to working with Rapid Developer," Edwards explains.

From a business perspective, Rational Rapid Developer provides BearingPoint with another key 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

"In addition to making our staffing model a lot easier, 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."

Steven Edwards
Senior Manager,
BearingPoint, Inc.,

of target deployment environments, the BearingPoint team will be able to rapidly reconstruct and redeploy the claims system to meet each prospective customer's technology requirements. Rational Rapid Developer will also enable BearingPoint'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 making our staffing model a lot easier, 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. Edwards notes, "From a maintenance point of view, the ability to hand off our system off to clients is another key advantage of Rational Rapid Developer."

A Partnership to Create Measurable Value

On the property and casualty claims system project, BearingPoint is demonstrating it's 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 name in insurance and other financial services – to take their existing, client-server based claims system, and to 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 appropriately. 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 configure-to-fit 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," Edwards explains.

Bringing in Rational Rapid Developer for the Test Drive

Before the BearingPoint 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. Edwards 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 a lot 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 very difficult 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 BearingPoint team decided to evaluate Rational Rapid Developer. Edwards 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 concept. 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 Perfect Match for BearingPoint's "Configure-To-Fit"

Rational Rapid Developer is a perfect fit for the iterative development approach of IBM® Rational Unified Process®, which is used throughout BearingPoint, as well as BearingPoint's configure to fit methodology. "Configure-To-Fit is a methodology for integrating various legacy applications using middleware messaging" says Edwards. "It had its birth in the telco industry. 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 configure to fit architecture," Edwards 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."

Empowering Developers with Little or No J2EE Experience

Because Rational Rapid Developer automatically constructs J2EE infrastructure code across all tiers of *n*-tier applications, it enables developers who are relatively new to J2EE development to become productive very quickly. Ravi Ramachandran, Manish Khot, and Sachin Shingala, Senior Consultants at BearingPoint, started the claims system project with limited J2EE development experience. They quickly found that Rational Rapid Developer was easy to use, and that it helped them develop their skill set as they developed quality software. Ramachandran explains, "I had a general skill set covering most areas, but my knowledge of EJBs was not very detailed. Rational Rapid Developer insulated me from all those detail intricacies and made it easy to catch on quickly." Khot agrees, "My story is the same. I didn't have much EJB or J2EE experience coming into this project. Initially, when I joined, I was assigned the task of JavaScript development and the front-end work. As time went by, I was able to pick up quite a lot of knowledge pretty quickly." Shingala adds, "I didn't have a lot of EJB experience either, and Rational Rapid Developer really helped a lot in masking that. I didn't need a lot of experience in order to develop effectively. After the initial learning curve, it became very easy to implement complex use cases."

Before the team began using Rational Rapid Developer, Edwards and three developers attended training classes. After the initial training, new team members received training and mentoring from more experienced teammates. "Every time we brought a new person on the team we gave them the training materials and allowed them to go through it on their own. Then we had them sit side-by-side with a developer who had already gained Rapid Developer experience. They were only a few feet away, and they would check in every hour or so, to make sure they were doing things the right way or if they had any questions. It didn't take more than a week for them to come up to speed and be productive," Edwards says. "When we started using Rational Rapid Developer, we had one front-end developer who went through the original training. After that he was able to build pages in Rational Rapid Developer and deploy them using EJBs, servlets, or pure Java – and he had no previous Java™ experience," he adds.

The Power of Patterns Drives Reuse and Standards Compliance

Just as Rational Rapid Developer helped the BearingPoint 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,

"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."

Steven Edwards
Senior Manager,
BearingPoint, Inc.,

BearingPoint's developers are able to simplify and speed development tasks for their teammates. Edwards 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, Sachin had the most 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."

Ramachandran adds, "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 was what name to give it." Edwards 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."

Simplifying and Accelerating Builds and Deployment

Rational Rapid Developer automatically constructs and deploys complete n-tier applications, streamlining the develop-build-deploy-test process for BearingPoint developers. With Rational Rapid Developer, BearingPoint developers also have the ability to develop and perform preliminary testing using a lighter weight application server, and then simply reconstruct and redeploy to their target environment for thorough system testing. Edwards 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 a lot 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 WebSphere Application Server."

From a developer's standpoint, Ramachandran agrees that Rational Rapid Developer has simplified many development tasks, enabling the developers to concentrate on the application's high-value business logic, instead of low-level coding details. "I feel the biggest advantage of developing with Rational Rapid Developer is that we spend more time on the 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 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."

Raising the Bar on Developer Productivity and Time-to-Market

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

The success Edwards and his team have had with Rational Rapid Developer has not gone unnoticed. "We had an internal architectural review. As part of that process we had to sit down and explain our methodology for development. We were working with an experienced J2EE developer and he was initially very skeptical. When he finally understood what we were getting from Rational Rapid Developer, he

"[IBM] 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."

Steven Edwards
Senior Manager,
BearingPoint, Inc.,

asked us how he could get a copy of it, because he was so excited about how quickly he could prototype applications," he reports

Edwards 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."

"I feel the biggest advantage of developing with Rational Rapid Developer is that we spend more time on the business logic, rather than worrying about configurations, compilations, getting the patterns right, or adhering to the UI standards."

Ravi Ramachandran,
Senior Consultants,
BearingPoint

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