Rational® software

Covarity Delivers Web Services and Customer Satisfaction with Rational XDE and Rational Suite Enterprise

At a time when most development organizations are just beginning to talk about Web services strategies, Covarity, Inc. is already deploying Web services applications. With a relatively small yet exceptionally effective development team, Covarity is using Rational® XDE™ Professional .NET Edition and an array of integrated tools from Rational Suite® Enterprise to successfully develop and deploy Web services and other .NET applications. Offering an unparalleled level of support for Microsoft® Visual Studio™ .NET, Rational XDE enables Covarity's development team to design and develop in a single environment — saving time and improving communication. Covarity's analysts, developers and testers also use market leading tools from Rational Suite Enterprise for requirements management, software configuration management and functional testing to improve productivity and team collaboration in every phase of development. By combining Rational XDE with a repeatable development process based on the Rational Unified Process® and Rational's award winning tools Covarity's development team were able to shorten the development life cycle by more than 25 percent, and deliver effective, high-quality solutions to their customers in less time.

Focused on the financial services market, Covarity connects applications across enterprise boundaries, facilitating not only the exchange of information but also real-time understanding of financial data. Many financial institutions, such as banks and other lenders, spend a significant amount of time and money gathering, entering, and analyzing financial data from their borrowers. The process is predominantly manual, involves duplicate data entry and relies on subjective decisions by junior field personnel who often fail to detect problem loans on a timely basis.

Covarity's solution provides greater transparency and effective loan portfolio management while reducing costs. By automatically polling, processing and analyzing borrower financial data, Covarity eliminates manual data entry, analysis, and subjectivity by ensuring standard business rules are applied to every account. Covarity's exception management system automatically alerts loans officers when an account becomes a problem or when trends indicate a loan may become a problem. Covarity's solution is delivered as a Web service, eliminating the need for their customers to invest significant capital in technology infrastructure. Covarity's customers rely on them for reliable, high-quality and timely software, and Covarity relies on Rational tools to help them build it.

Advantages for Everyone On the Team

Jeff Fedor, Covarity's CTO, reports that everyone on the development team benefits from the use of Rational tools and the adoption of the Rational Unified Process. Fedor explains, "From a CTO perspective, I think the most important thing is a repeatable process. That is something that concerns everyone. With a repeatable process, we simply get better code and can manage risk better." Covarity's process is based on the RUP® configuration for small projects. This provides guidance for smaller teams — typically less than 10 people — on how to achieve the proper balance between formality and completeness on one hand, and minimal process overhead on the other.

Fedor continues, "When you add in the tight integration of Rational tools you just get a much greater productivity level. Your team is not flip-flopping around. Everything is in one place — including the design and the code. That is such a huge savings in time, but I think it's not until you start using the tools that you fully realize that."

IBM®

For the software architects and business analysts at Covarity, the ability to trace requirements to the visual model and assess the architectural impact of changes to requirements is particularly important. "From an architectural perspective, the visibility provided by Rational XDE and other tools is key. They can make enlightened decisions about a change request based on its architectural impact. That visibility — from visual modeling to traceability of requirements and architectural impact – is an important benefit for us." Fedor notes.

For developers, the most significant advantages derive directly from Rational XDE, which allows them to move seamlessly between design and development. At Covarity, visual modeling is as much a part of the Microsoft Visual Studio .NET IDE as the code editor and debugger. All of Covarity's development is done in the relatively new C# programming language, and the development team takes full advantage of the fully integrated and automated C# code synchronization capabilities in Rational XDE. Fedor concludes, "From a developer perspective the biggest advantage is being able to work and have your models or your code updated instantly. There is almost no overhead. Whereas before, they'd have to think 'I should go and update my models from the code or vice versa.' And now everything is updated automatically and the whole system remains documented throughout the entire process."

Getting Started

The most recent Web services application project at Covarity was completed in about four months. As with all of Covarity's projects, it started with a process and a vision document, which is what Covarity always uses to capture very high-level requirements and design constraints, and to gain a better understanding of the system they will develop. Fedor recalls, "Using the Rational Unified Process configuration for small projects as a starting point, we settled on a core set of activities and artifacts that we thought were essential — first and foremost was the vision document. We developed the vision document in Rational® RequisitePro® and from there we wrote out our use cases." Fedor and his team use Rational RequisitePro to define project requirements and use cases, and to manage and track requirements as they evolve throughout their projects.

Each use case for the Web services application project was then modeled in Rational XDE. "Not every use case was fully written out in Rational RequisitePro, but every use case was represented in a use case model in Rational XDE. If we couldn't understand everything we needed from the use case model, then we would write it out" Fedor continues. "Because we had this process and game plan, everybody knew where we were at and where we were going, what was required and what was next. Continually using that information throughout the iterations gave us a very repeatable and predictable process."

"Rational XDE Has Changed the Way We Develop"

Once use cases are defined and modeled, the Covarity team starts design realization, which takes the use cases a step further in the process. Fedor explains, "We start to think about use cases from an architectural perspective as opposed to a functionality perspective. And that is when we start to get into where the tight integration between Rational XDE and Microsoft Visual Studio really helps out. The integration is so smooth that we can do a realization of a use case, put together sequence diagrams and any reflection that we need to on our data model; and then push it through by forward-engineering it. And suddenly, we have code for that supporting piece. It is really, really fast compared to our earlier projects. We used to sit at white boards and go through a 'model process day'. Then someone would run off and do some work on it, and the next day we would regroup and take a look at it. Now all that has gone away, and we just have very, very quick iterations."

Fedor explains that Rational XDE has completely changed the way the Covarity team thinks about the development process, "With Rational XDE the time frame between 'let's model' and 'lets move to code' has collapsed. Because Rational XDE is integrated into Microsoft Visual Studio we can quickly iterate through designs and really take advantage of the iterative development process. We take advantage of being able to round-trip engineer the .NET framework quickly, and then we build our own pieces on top of that. Rational XDE has changed the way we develop. Absolutely -- and for the better."

Improving Team Communication

Fedor reports that the development team at Covarity is an exceptionally cohesive group. Clear communication between team members is essential for maintaining a unified team and minimizing misunderstanding. Rational XDE, along with Rational® ClearQuest® and Rational® ClearCase® LT, have helped the Covarity team to not only develop more efficiently, but also to communicate and share information more easily throughout development. Tight integration between Rational ClearCase LT for version control and Rational ClearQuest for defect tracking, provides Covarity with a comprehensive, customizable and scalable software configuration management solution. The Covarity team also uses Unified Change Management (UCM), extending the benefits of that integration with a workflow that defines and organizes work around activities and artifacts within the project.

Fedor explains how the tools work together during a typical Covarity project, "In our process, the analyst will deal one-on-one with a customer or potential user to develop a first iteration of use cases, which are then handed off to the architects. The architects will look at individual use cases or a set of use cases, and then they will use Rational XDE Modeler to take a first crack at the use case realization. Then we get the entire team together, throw Rational XDE up on a projector, and walk through the use case models to understand the motivation behind everything. The great part is that, rather than sitting in a separate tool, the design is integrated right into the team's project. So, one of the architects creates a project and puts it into Rational ClearCase LT. Then everyone else grabs it, and is able to see all the architectural pieces immediately. We use Rational ClearCase LT for source control and we manage all of our XDE and Rational RequisitePro assets in it as well. Everything is stored in Rational ClearCase LT."

UCM links the artifacts managed through Rational ClearCase LT, with the activities managed through Rational ClearQuest, simplifying the process of change throughout Covarity's development process. Covarity also uses the Rational testing tools that are included with Rational Suite Enterprise, and the results of their tests can be used to create new entries in Rational ClearQuest. Fedor notes, "We use [Rational] ClearQuest a lot in Q.A. We also use it for tracking enhancement requests, because invariably, users will have opinions on the way software should work."

Fedor adds that Rational XDE provides an automated, unambiguous way for Covarity's developers to document their work so that the architects can check it against the design. "Once the group has had a chance to get familiar with the motivation behind the use cases, the architects continue on their path. They start to release more areas to be coded, and that is when the developers go to work. Classes begin appearing in their projects, and they take it from there. In our system, prior to checking code back into Rational ClearCase LT, a developer will use Rational XDE to reverse-engineer the code back into the model to document it. So, all the code is automatically documented in the system, and models are constantly being updated. From there, the architects are able to use Rational XDE to monitor the developers to ensure that the architecture is really being implemented as designed."

He concludes, "Rational XDE is simply a great tool, because our entire project — models and code — is all there, and it is constantly being updated in our development environment. As a result we have much greater transparency through the process, and we actually get better code out of the system because, as they say, "sunlight is the best disinfectant." From a project leader or architect's perspective, we can see that the developers are doing the right thing. And the developers can click over to the model and instantly see what they should be building."

"We use [Rational] ClearCase LT for source control and we manage all of our [Rational] XDE and [Rational] RequisitePro assets in it as well. Everything is stored in Rational ClearCase LT."

—Jeff Fedor,
CTO, Covarity, Inc.

Instant UML and the Power of Patterns

Like most organizations, Covarity's development team includes people with a variety of skill sets. At the beginning of the project, not everyone on the team was an accomplished modeler or fluent in the Unified Modeling Language (UML), which Rational XDE users apply to design, model and document software. However, because Rational XDE can automatically generate the UML representation of code, Covarity's developers were able to see visual models of their code instantly and learn UML very quickly. Fedor observes, "That kind of instant UML, or assisted modeling, helped us in two areas. One, we had some developers on the team who weren't as experienced with UML as the architects. So for them, learning to model was greatly simplified because they could learn by example. If they were unsure, they could go and write code, reverse-engineer it, and see the model right there for themselves. Even if you start a junior developer using Rational XDE, they will begin to understand UML very quickly. The second big benefit for us, is that there is a lot of C#-specific support in Rational XDE, including language constructs that are completely supported, so we didn't have to figure out how to represent them ourselves."

Rational XDE saved the Covarity team a lot of time by helping them leverage existing patterns. As a result, they were able to jumpstart their development without having to reinvent the wheel. Fedor says, "In developing a Web service, the facade pattern is, in a way, almost forced on you. We used the facade pattern and the Pattern Wizard in Rational XDE to simplify things. The wizard walked us through selecting the class we wanted and then it automatically built any relationships for those patterns. It also allowed us to implement any new classes that had to be spawned as a result of implementing the pattern. Having those patterns baked right in allowed us to implement them quickly. Implementation was literally just a series of clicks, as opposed to probably a good hour or two of thinking it through. Also, the wizard has some 'smarts'. It is not going to let you implement a pattern where it is either not feasible or technically impossible to do so. And you're not going to make syntax errors with your UML diagrams because Rational XDE does it all for you."

Quality Built In

The constant synchronization between code and architecture helped Covarity meet and exceed its high standards for quality on the Web services project. But there were other factors as well. Covarity's testing team used

Rational® Robot and Rational® TestManager to conduct extensive functional tests, and the developers used Rational® PurifyPlus to check for memory leaks before they checked their code into Rational ClearCase LT. These tools are all included with Rational Suite Enterprise, which comprises all of the tools in Rational Suite® AnalystStudio®, Rational Suite® DevelopmentStudio and Rational Suite® TestStudio®.

In addition, following an iterative development-process — one of the best practices of software development promoted by the Rational Unified Process — was a big contributor to product quality. Fedor notes, "What really helped was sticking to the iterative process and doing a greater quantity of smaller releases. That really cut our bug count down; our bug count was phenomenally low on this project."

Getting A Running Start from

Rational Services According to Fedor, the adoption of Rational tools and the iterative process at Covarity was fairly rapid and uneventful, thanks in large part to a number of Rational's Professional Services. The Covarity team attended a variety of Rational University training courses at their site in Waterloo, Ontario. "We had a great trainer who worked with us to really tailor the course to our needs. We did some remedial UML work, and we actually worked on our project during the training which made it much more tangible for us," Fedor notes. "We also sent a few people to the course on Rational ClearQuest Administration and some others to Rational University courses. We all took the Rational XDE Web-based training available on the Web through the Rational Developer NetworkSM. There were a number of real benefits, including significantly less time required to adopt the tools. It was a great introduction. We were fully supported, so we just dove right in. And for the in-house courses, there was no risk to jumping right in as we had Rational folks right there with us. Wherever we could, we geared

the labs toward actual project work. Everyone came away with a higher degree of confidence in their ability to use the tools effectively."

In addition to training, Covarity also brought in a Rational consultant to help them set up Rational ClearCase LT and ensure that it was fully integrated into their development environment. Fedor adds that the training and consulting provided exceptional value, "I think it brought down all the barriers to adoption. Our people didn't feel like there were tools being thrust upon them, because there was so much support provided along with them."

Fedor and others at Covarity have also been exploring the Rational Developer Network, an online source for news, knowledge and other resources for the Rational community. "We visit the Rational Developer Network for articles, tips and anything else that might be helpful," Fedor says. "We are always looking for best practices. As .NET was coming out, we wanted to see what people were doing with it, and read any studies about it. We looked at a Web services project on the Rational Developer Network to see how other folks were going about modeling Web services, as it is a fairly new area. It has been a real help."

Understanding and Managing Change

Fedor feels that the most important benefits Covarity has realized from using Rational tools are improved visibility, traceability and predictability. He explains, "It is hard for problems to hide now. Because the tools are so well integrated, we have visibility from architecture straight through to code. It is so quick, that it is easy to catch problems, and see where developers stray off the course. Of course, developers aren't out to thwart architects, despite what architects may sometimes think. But somewhere along the way, someone is going to make a design decision, and synchronize the model and the code. With the entire model in your development tool you can see when something has been changed, and you can decide whether a change to the architecture really is required. You can't always get your architecture right from day one. It does evolve. I think Rational XDE facilitates greater communication between architects and developers because these issues are brought to light early. They don't

come out in a code review three months after you ship the product. They come out in the process. The issues are clear, and in the end everyone knows how they are going to be resolved. Everyone is on the same page."

The benefits of traceability, made possible by Rational's integrated tools, extend beyond models and code, to the entire development process, according to Fedor. He concludes, "Typically, what happened before we started using Rational tools is that we would write a great set of documentation and requirements. They would be used at the beginning of the project, but never touched again because it was so difficult to keep them up date. I find that traceability and UCM encourages you to make those changes. Now our documentation and requirements are constantly evolving and always up-to-date. The other issue is that it was often difficult to remember why a particular change was necessary. Now we have it all clearly documented. It really helps us understand the impact of changes, and gives us an historical view so we can see why the changes were made. As a CTO, having that kind of knowledge gives me a huge advantage. For any given change, knowing who requested it and why is key. As software developers, we need to do a better job of understanding what customers actually want. With Rational we're doing it, and that is incredibly powerful."

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.

IBM Rational software

Dual Headquarters

18880 Homestead Road
Cupertino, CA 95014

20 Maguire Road
Lexington, MA 02421

Toll-free: (800) 728-1212
Web: www.ibm.com/rational