

SpectrumSCM, Configuration Management the way that it should be

By Bob Aiello - October 2004

Working with SpectrumSCM one becomes immediately aware that the vendor had spent considerable time making significant design choices. Instead of trying to be all things to all people, this tool focuses on serious SCM and those who are ready to do Release Management the way that it should be. These decisions also come with a price. The SpectrumSCM user is forced to think through the entire design straight through to release management. If you are ready get your CM practices in tip top shape, then SpectrumSCM should be pretty high on your list of products to consider for deployment.

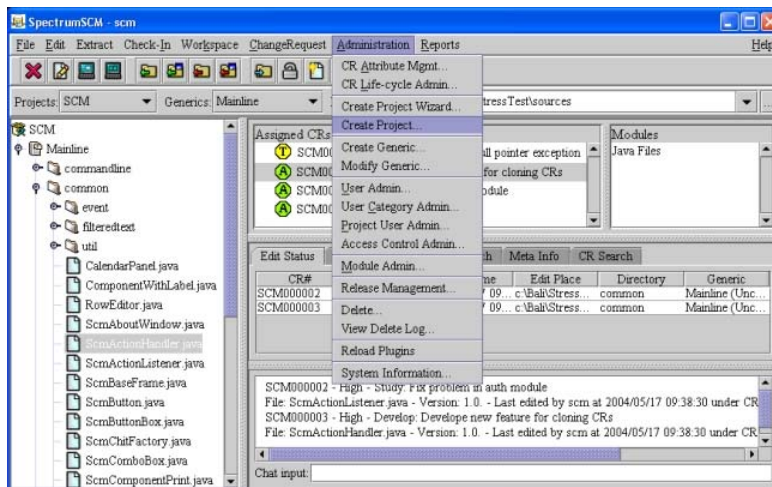
Bob Aiello is a Senior Editor for CM Crossroads and an Associate Director at a major financial services firm in NYC, where he has company wide responsibility for Software Configuration and Release Management best practices. Bob is on the Steering Committee of the NYC Software Process Improvement Network (CitySPIN), where he is also the chair of the CM SIG which meets in Midtown NYC. Mr. Aiello has a Masters in Industrial Psychology from NYU and a BS in Computer Science from Hofstra. You can reach him by email at bob.aiello@cmcrossroads.com

Architecture

The underlying architecture of any CM tool is essential in evaluating its value in terms of functionality, reliability and of course security. SpectrumSCM is written in 100% Pure Java with a Java ODBMS, Java RMI and raw sockets for communications and file transmissions. Spectrum Software's Chief Architect, Corey Brown explained, "Its' Java based architecture guarantees a consistent look and feel across all platforms, including Unix, Linux, Windows, MAC OS and other platforms that support a modern JVM." One big plus is that you can extend/customize the tool using Java giving it all the benefits of Java portability and extensibility.

Change Request Centric Approach

Many CM tools boast the ability to integrate with Change Request and Defect Tracking Systems. SpectrumSCM requires that you plan and implement your CRs from the outset. In



fact, you won't be able to create a release unless you have done so. In some ways, I began to feel that this product is really a workflow (SDLC) and Change Request tool that also happens to include Source Code Management!

This is one of the key areas where SpectrumSCM is well thought out with CRs dependencies a core part of planning the release management process. When I asked Corey how he would

handle customers who did not want to bother initially with CRs. His response was clear that this might not be the best product for them. According to Brown, "Change Management is the base framework from which all other functions in SpectrumSCM are derived, including the release management sub-system. Experienced software developers and configuration managers know that a tight integration between Change Management and Version Control creates an SCM system that provides superior traceability of source level revisions to concrete change statements." We further discussed that CRs and their dependencies must be well thought out and specified or you won't be able to create a release.

CRs are created via XML, which allows users to migrate an existing issue tracking system to SpectrumSCM. Its Change Request Work Breakdown Structure is powerful although obviously requires some upfront planning and design for effective use by most development teams.

Role Based Access Control

Sarbanes-Oxley auditors will love the extensive capabilities of this tool to manage entitlements. Access privileges may be customized extensively, although the default model out of the box certainly seemed to meet most of my needs for large scale projects. Password expiration and SMTP mail authentication also are available and often missing from other CM tools.

Terminology

Some of the terminology seems a bit counterintuitive but may be somewhat more familiar to those who have used Sablime in the past. For example, "Generics" are long lived branches (e.g. development branches). The product's branching capabilities are strong, although you will also have to get used to making the code uncommon (e.g. branched code) or re-commoned (e.g. merging). The online documentation is extensive and well written, and much of the training materials describe the use of CM Patterns.

Overall, I found the time spent learning the tool to be worthwhile, but do not tackle SpectrumSCM unless you are ready to learn their way of doing things. In fact, it seems that SpectrumSCM is actually meant for serious developers who are likely to be very successful at Software Configuration and Management.

Familiar Concepts

Much of using this tool is well thought out and very intuitive too. The concept of Sandboxes should be familiar to most developers and GUI interfaces were well designed and reasonably easy to use.

Training and Implementation

I have to be honest and admit that I found too little emphasis on the importance of training to use this tool. You should expect to need a product engineer onsite, for a couple of weeks, for large project deployment. It was not possible for me to really judge, but this vendor claims that their product does not require much effort in terms of ongoing systems administration.

Performance

Checking in a file or two seemed to be a bit sluggish but while working with a large directory the check-in process positively flew through adding a large structure. Complete performance testing is beyond the scope of this evaluation.

Integrations

SpectrumSCM integrates with Visual Studio (including .net), Borland Control Center, Microsoft Office, Eclipse and many other IDEs and development environments.

What I liked most

This product is very well thought out from both a functional and architectural point of view. The company has been around for over 12 years, although their CM tool is relatively new. They seem to be open to input from their customers. Spectrum Software has done a good job of thinking about exactly how this tool should work. I would recommend that serious developers have SpectrumSCM on their short list of strong CM tools for hands-on evaluation.

Cost:

Client server pricing based on concurrent user licensing. Clients start at \$750 for a single user and server license is \$1,500. An annual support contract is available for \$120 per license.

Company Contact:

Spectrum Software Inc.
11445 Johns Creek Parkway
Suite 300
Duluth, GA 30097
Phone: +1 (770) 448-8662

info@spectrumsoftware.net
www.spectrumsoftware.net

This product review is a publication of CM Crossroads, *the* Configuration Management Community. For more information about this or other CM Crossroads publications please contact us by email at info@cmcrossroads.com