In any sizeable network, managing the changes and updates to device configurations can be a full-time job for an entire network-engineering department. However, those tasks are made easier with Network Change and Configuration Management (NCCM) applications like those provided by Austin, TX-based AlterPoint Inc. (www.alterpoint.com). The company’s DeviceAuthority™ Suite, a multi-vendor NCCM product suite, offers real-time visibility, automation, and governance of complex IT infrastructures.

DeviceAuthority Suite has the ability to remediate security vulnerabilities on hundreds of different routers, switches and firewalls, provide Cisco SMARTnet documentation across your entire device inventory, update device passwords, and deliver NSA compliance reports for networks. To provide a platform on which to build these capabilities, AlterPoint turned to Eclipse and the Eclipse Rich Client Platform (RCP).

“We wanted to provide a Telnet client that was similar to the consoles that administrators were familiar with. That type of rich interaction is very difficult to provide in a web application. Eclipse provides a great framework in which to build those types of features.”

“Managing configuration changes on routers, switches, and other network devices presents a lot of complexity,” says Alterpoint’s Giovanni Farris. “It’s important to set standards on your configurations and manage all changes to those standards. In our v3.0 release, we wanted to provide a means by which network engineers and administrators could quickly examine their configuration to locate settings that weren’t meeting certain policies. We were already looking at Eclipse as a development environment when we discovered that we could also use it as an application framework. We decided to create the type of policy enforcement engine we wanted to provide through the Eclipse Rich Client Platform (RCP), as opposed to our existing web client structure.”

Betting on Eclipse
Among the features that AlterPoint wanted to include in DeviceAuthority Suite were capabilities such as allowing network engineers to Telnet into devices and record their actions as they were executing the commands. “We wanted to provide a client that was similar to the consoles that administrators were familiar with,” Farris says. “That type of rich interaction is very difficult to provide in a web application without using an applet or ActiveX control. Eclipse provides a great framework in which to build those types of features. Java and other code developers have had sophisticated integrated development environments for a long time. Unfortunately, network engineers have been stuck switching between numerous applications. Using Eclipse, we’ve been able to put all of the tools they need in one place in a very tightly integrated and highly interactive environment. Engineers need to know exactly what a tool is doing with their network and an Eclipse UI let us provide necessary level interactivity for our end-users.”

Other capabilities that made the Eclipse RCP attractive to AlterPoint were features like syntax coloring capabilities within the editors, the extensible search framework, decorators, and other interface elements they wanted to deliver to their customers. For example, AlterPoint used Eclipse to provide context intelligence.
If a network engineer has an IP address selected, they can right click on a device and have its operations displayed—whether or not they see that particular area of the configuration. “The more we used Eclipse, the more we were able to leverage a lot of the framework components that were in place, like the JFACE structures and extensibility framework,” Farris says. “During development, it was almost like we had a networking framework that we could write tools to. What we call the Integrated Network Environment™ (INE) is the framework for bringing in network devices from the DeviceAuthority server. We’ve written separate plug-ins that act as tools against that same information set. As you drop them in, different options become visible. As we create new tools, they function seamlessly with what already exists.”

Eclipse RCP also allowed Alterpoint to hide or show functionality for different network engineering roles. This approach introduces another level of network security by only allowing certain users to perform certain operations. “For the most part, we were able to hide any functionality that would have confused users by providing perspectives specific to the function they are performing,” Farris says. “We can also drop our features into an existing Eclipse installation and get all the same functionality. For example, network engineers who have Eclipse running can drop INE into their existing Eclipse installation. It just works.”

**RCP in Action**

Right now, Eclipse RCP acts as a common interface for two of AlterPoint’s products, with a third expected to be available within the next twelve months. “Our products manage the configuration of the network,” Farris says. “DeviceAuthority Audit shows the entire network device inventory within a tree structure and lets users perform actions like searching that inventory to execute jobs against it. We added a second product called DeviceAuthority Update that allows users to make changes to their network devices. For example, we allow users to Telnet to one of their routers and we can record that interaction with the device and save it in a script. They can then run it against any number of other devices they have in their inventory. We’ve leveraged some of the syntax highlighting within Eclipse to show them the scripts that they’ve created. In addition, we have leveraged the development cycle build into Eclipse tools to allow end-users to develop IT automation.”

Farris believes that his company has gained as much as its customers from the use of Eclipse and the Eclipse RCP. “Eclipse and RCP have shaved a lot of time off of our development,” he says. “We needed a rich client framework and Eclipse RCP worked out extremely well for us. We used the component model. We were able to leverage the text editors. We used decorators to highlight different objects to show status in different ways. We used the properties and preferences. We just extended those contribution points and received a lot of functionality for free. Eclipse has saved us a lot of time by relieving us of developing features from the ground up.”

*AlterPoint’s DeviceAuthority uses the same type of constructs as the Eclipse IDE. However, the content is substantially different. The views and editors are geared toward network devices showing an inventory and script execution against a device. The item in the problems tab is not a Java compile exception but a violation of a compliance policy and the search results are not local files but network devices.*