

Managing Web Development with Surround SCM

Enforce your process with flexible branching, shadow folders, and security

by Fernando Cremer

Change control is a critical aspect of development. While Web development tends to be more fluid, you should not abandon process entirely. Surround SCM's flexible branching model, shadow folders, robust security, and other advanced features help you control change throughout the Web development process.

One of the biggest challenges Web developers face is how to implement a source code management (SCM) application that meets Web development best practices. Most Web development shops use a development process that consists of the following stages:

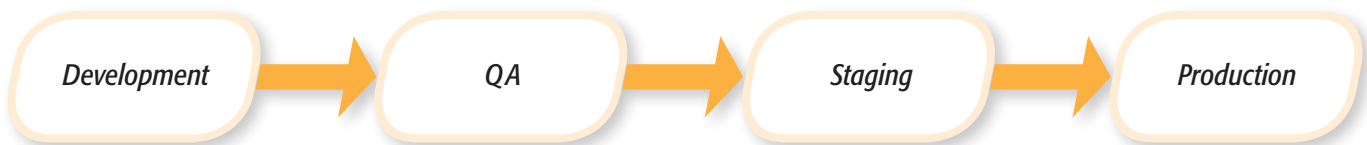


Figure 1: Web development process

Development—In this stage, developers modify the content and make changes to functionality.

QA—After the developers complete the initial changes, QA tests them.

Staging—To protect the production code, changes are integrated and verified in a mirror copy of the live environment. This is the last stop before going live with changes.

Production—This is the live Web site. Changes go live only after they have gone through the previous stages successfully.

Read on to learn how to manage Web development with Surround SCM's branches and shadow folders.

Branching in Surround SCM

Branches allow you to set up different, parallel environments that contain the same file structure. Changes in one branch don't affect the code in other branches. Promoting and rebasing provides an easy yet controllable way to merge changes between two branches.

The waterfall model is the best way to set up Surround SCM branches for the Web development process. In this model, each stage is a branch and changes flow from Development to Production.

Adapting Figure 1 to a waterfall branch model is easy. Development is the mainline branch in Surround SCM. QA, Staging, and Production are baseline branches off the mainline. Staging is a child of QA and Production is a child of Staging. (See Figure 2.) As you can see, changes flow down the tree like water flowing down a stepped waterfall.

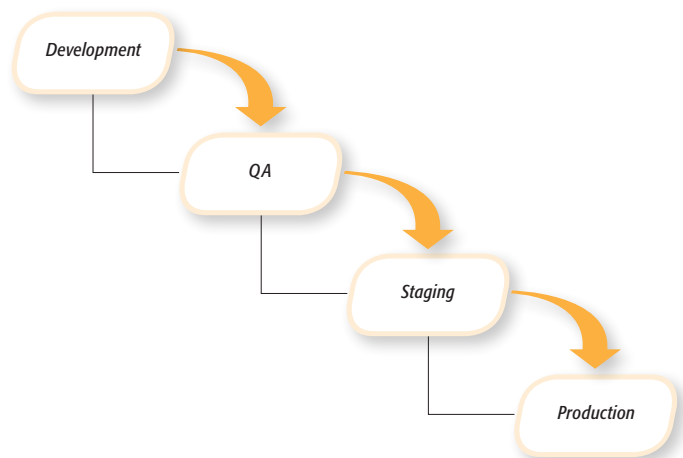


Figure 2: Web development process in Surround SCM

In a waterfall branch model, changes move into the next branch (stage) through rebase actions. With this branching approach, you can rebase to move changes between each branch, where each branch represents a different stage in the lifecycle. Rebase actions also allow you to specify a label, timestamp, or workflow state. For example, a group of developers work on a fix related to change request 344-876. As they check in changed files, they apply the 'CR# 344-876' label to the files. A high-level user, such as an administrator or project lead, can then rebase the changed files to the QA branch for verification.

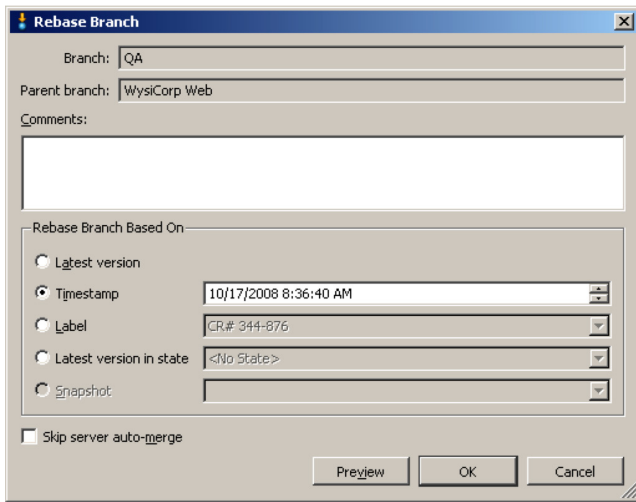


Figure 3: Rebase branch dialog

In the following example, WysiCorp uses Surround SCM to control the files and development process for their Web site (see Figure 4). All development takes place in the WysiCorp Web mainline branch. Under the Production branch, snapshot branches capture the production environment at specific milestones. Capturing these milestones provides a way to easily roll back to a previous version of the environment in case newly introduced changes cause problems.

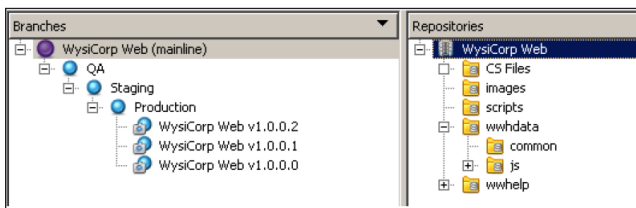


Figure 4: Web development branching model in Surround SCM

Publishing Files with Shadow Folders

Surround SCM shadow folders provide a way to publish read-only files to a network share, making them available to QA for testing and verification.

In a Web development environment, these shares are folders on the various web servers for the Web site. For example, WysiCorp uses separate IIS web servers for each stage of the development process (see Figure 5.)

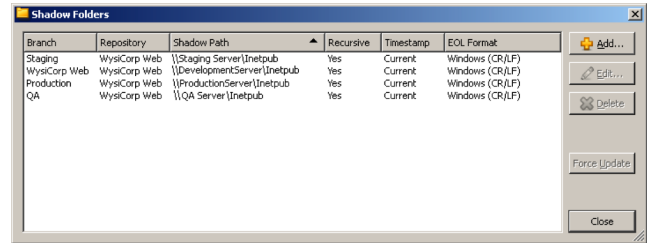


Figure 5: Shadow folders correspond to development stages and branches

After checking in changes, the developer points a browser to <http://DevelopmentServer/> to view the changes in a Web browser. When testers are ready to review changes rebased to the QA branch, they browse to <http://QA Server/>. Once the changes are approved by QA, they are verified on the staging server (<http://StagingServer/>) before being pushed live.

Enforcing Security

Surround SCM provides configurable and granular security to restrict who can perform rebase actions on each branch. To enforce your organization's process, set up security groups that match each role in your development group. You can also apply security to a specific branch to provide an additional level of security. For example, if a team is supposed to submit changes in the development branch only, you can make their access to the remaining branches read-only.

WysiCorp's process dictates that members of the development group should only make changes in the Development branch. Once changes have gone through a review process, the changes are rebased to the appropriate branch by a team lead. To enforce this process, a Development security group is created and permissions to all other branches are set to read-only. This ensures developers can check in changes to the Development branch only. Figure 6 shows how one of the restricted branches is set up.

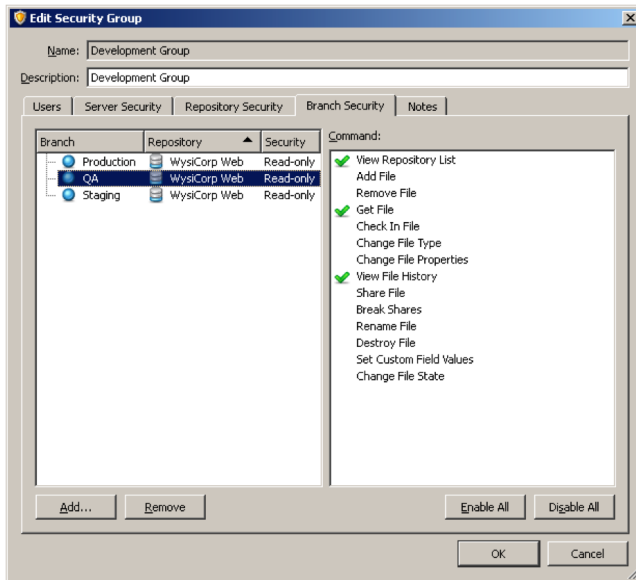


Figure 6: Branch security settings

Additional Process Enhancements

Branching, shadow folders, and security are the basics in setting up a Web development process in Surround SCM. The following features can further enhance your process:

Workflow—Track the various processes within each branch using Surround SCM’s workflow. Is a file ready for rebase? Has the file gone through the QA process? Use the workflow to ensure changes move from one stage to the next only when ready.

Triggers—Extend Surround SCM’s functionality with powerful, in-application programmable triggers. Enhance team collaboration and process visibility with email notifications. Want to know when a file is ready to be reviewed? Set up a trigger to send an email when a file changes workflow state or when a rebase takes place

Want to prevent checkouts when a file needs to be reviewed? Set up a pre-event trigger that prevents users from checking out files when the workflow state indicates a need for review.

IDE Integrations—Surround SCM provides tight integration with popular Web development environments and IDEs, like Dreamweaver and Visual Studio, allowing you to perform most Surround SCM actions from within your preferred tool.

Next Steps

Seapine solutions can help you manage Web development projects from start to finish. Our comprehensive tool set facilitates collaboration, coordinates tasks and workflows, and provides access to source files anytime, anyplace. Seapine’s Web development solutions offer:

- Centralized digital asset management
- Remote real-time, secure access to issues and source code
- Team-based collaboration and communication
- Role-based security
- Automated testing of Web-based applications
- Quality metrics and reporting for management
- Cross-platform support

Unlock your Web development potential with Seapine solutions. Visit www.seapine.com/webdev.html to learn more.

About the Author

Fernando Cremer is a solutions consultant at Seapine Software, Inc. He has over 10 years of experience in customer support and service, 7 of which have been in the software industry. Fernando is responsible for software services, demonstrations, and training for Seapine’s award-winning change management products.