



**Serena StarTool**  
**Software Change Management**

Improving Application  
Availability with Integrated  
Change Information

**serena™**

Automating Change to  
Enterprise Code and Content



**Improving Application Availability  
with Integrated Change Information**

**Table of Contents**

The Case for Integration . . . . . 2

Crucial Components: SCM and Application Availability Technologies . . . . . 3

Stand-Alone Application Management Hinders Availability . . . . . 4

Integration: Recovering to Maximum Availability . . . . . 5

Avoiding Revisionist History . . . . . 7

## The Case for Integration

There is no such thing as a spontaneous application failure or an unprovoked performance problem. These predicaments just don't happen by themselves.

While arriving as unwelcome surprises, application failures and sudden, slow performance are usually the result of changes to the application. When these slow-downs and stoppages occur, the resulting application downtime damages the business system availability expectations of customers, employees and business partners.

The developers responsible for resolving stoppages and slow-downs have traditionally been faced with the pressures of quickly returning business operations to normal while identifying the root cause of the problem. Strategies for problem correction are devised using tools to analyze the code of a failing program, provide a detailed analysis of a system dump, or evaluate and report errors between data definitions and data formats. However, changes aimed at correction are often implemented rapidly, without regard to the change history of the failing or offending program. Often times, a change made to correct a single problem results in what has been known as a “bad-fix injection” — a situation where new errors are inadvertently introduced to a program when known errors are corrected.

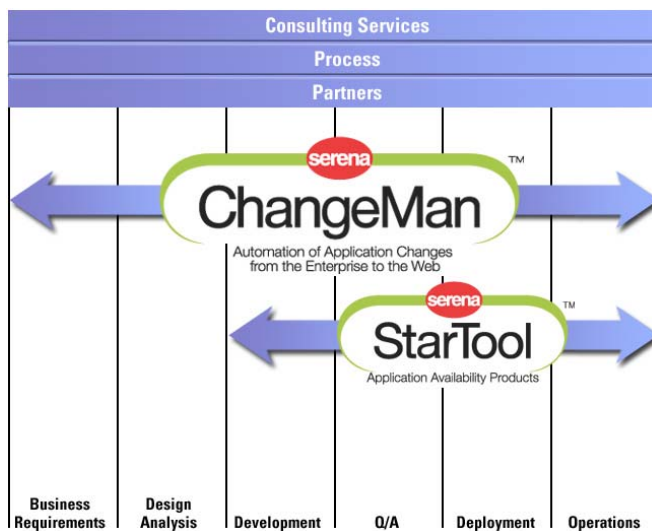
These new errors are often introduced due to lack of developer knowledge of, or access to, the past change history of a program. Additionally, new errors are introduced when a developer makes changes to wrong source code, unintentionally working with the incorrect version. Instead of returning to business as usual, new errors increase downtime and further reduce application availability.

Failures and slow-downs in the production environment must be remedied as quickly as possible. While Application Availability tools are essential for identifying the root cause of a problem, knowledge of application and program change history is vital to reducing downtime. What was changed prior to the problem situation, when it was changed, why it was changed and who made the change provide key information for achieving continued stability. Access to change histories from Application Availability tools ensures that developers work on the correct version of the source code, avoid a bad-fix injection and minimize application downtime. Making change histories available to Application Availability tools requires integration with the technologies that protect, control, manage and monitor application changes—Software Change Management (SCM) systems.

This white paper discusses the challenges when stand-alone Application Availability and SCM technologies are used separately, and emphasizes the capabilities available when these vital solutions are integrated.

## Crucial Components: SCM and Application Availability Technologies

In the OS/390® or z/OS™ environment, the combination of Software Change Management (SCM) and Application Availability technologies ensure the quality, stability, security, accountability and recoverability of critical business systems in the production environment. SCM is used to facilitate the efficient entry of application components into the production environment and Application Availability technology is used to measure, capture and identify errors that exist in production while supporting efforts during testing to help ensure a quality deliverable. While SCM is most frequently used for pre-production activities, Application Availability is most frequently used for post-production.



Serena ChangeMan and Serena StarTool products are essential tools in today's mainframe environments. Serena ChangeMan solutions manage the life cycle and Serena StarTool products automate developer and programmer activities that are a part of the life cycle.

SCM automates the entire application life cycle, accelerates application development and improves software reliability. Superior SCM technologies offer an open architecture with a central point of control, a consistent change lifecycle methodology, increased accountability and quality assurance, and protection of software assets. Additionally, these powerful systems provide key functions such as library management for source control and protection; version control to prevent code regression and simplify recovery to prior versions of a program; impact analysis to determine the potential impact of changes; and configuration management to identify and maintain the relationships between versions of application components.

Equally important, superior SCM technology provides these functions while keeping track of every change that is made to every component, building a change history to identify what has been changed, when the change was made, the reason for the change and the name and user ID of the person responsible for the change.

While this information is deemed crucial for developers planning on implementing new changes, it is equally crucial for those who are required to correct application failures or poor performance identified by Application Availability technologies.

Application Availability technologies ensure continued smooth performance of business systems, identify the root cause of an application failure, and analyze source code to report errors in logic and data. These technologies help development and systems teams ensure the continued high-quality and high-availability of business systems by delivering functions such as file and data management, missing source code recovery, interactive source code debugging, linking failing instructions to a line of source code after an abend, and speeding up I/O performance.

Separately, SCM and Application Availability technologies provide the support required to facilitate movement into the production environment and to quickly capture, identify and remedy production errors. Organizations employing these technologies usually realize increased application quality and availability along with reduced failures. Furthermore, these organizations can realize an increase in productivity as systems and application development teams use these tools to automate processes.

But failures will occur. And for almost all organizations, recovery from failure must be fast and accurate. And without immediate, integrated access to SCM change histories along with the power of Application Availability technologies, the human intervention required can be the cause of additional errors.

### **Stand-Alone Application Management Hinders Availability**

Traditionally, SCM and Application Availability technologies have been seen as “bookends” for the production environment. Today though, when faced with requirements for high-volume, high-performance business processing with little or no downtime, it makes sense for IT organizations to have SCM and Application Availability technologies operate in conjunction with one another. The unification of these tools provide a holistic view of the entire production environment — from pre-production to post-production and historical activities.

Operating in stand-alone fashion, IT organizations will continue to face delays in recovery and reduced availability. Without integration between these two powerful technologies, when addressing problem situations, developers lack the ability to:

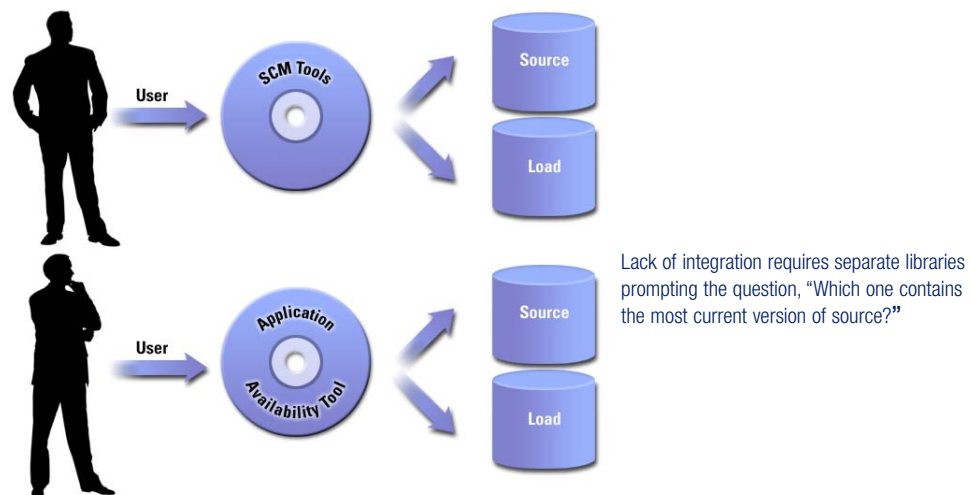
- Guarantee they are working on the correct version of source code.
- Review and compare changed lines of code.
- Ensure synchronization between a source and load module.
- Analyze past changes to a program.
- Validate data integrity during the change process.

Organizations committed to the highest standards of information delivery can benefit enormously through integration of SCM and Application Availability technologies. When working diligently to correct a problem situation, the ability to immediately access change history information, guarantee access to the correct version of source code, analyze past changes and validate data integrity reduces the likelihood of a bad-fix injection. Most importantly, integration between SCM and Application Availability technologies helps keep downtime to a minimum and availability at a maximum.

## Integration: Recovering to Maximum Availability

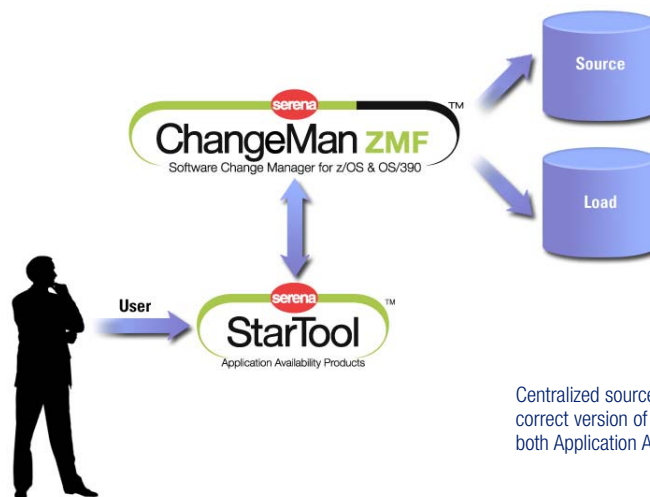
Working together, complementary technologies such as SCM systems and Application Availability tools improve application quality, deliver the highest level of performance and recover to maximum availability. Arming developers with interactive, critical change history information during a failure situation, and delivering a holistic view of the production environment will help organizations respond faster and more accurately to problem situations.

Integration between SCM and Application Availability technologies begins with a consolidation of the storage mechanisms for the most crucial application component — the source code. Lacking integration, users of individual SCM and Application Availability tools are required to manage multiple sets of source storage libraries. Populated by different and unique compile procedures, multiple libraries create uncertainty between the correct versions of source and the correct versions of load modules. Also questionable is the accuracy of the synchronization between all application components. With separate libraries and without integration, there is no assurance that the source copy stored for the dump analysis is identical to the source stored in the production library. Nor is there assurance that the load module running in production is identical to the load module required for a debugging session. The disparity can result in changes to the wrong version of the source code and a bad-fix injection.



Centralizing source storage into a single, safe location shared by SCM and Application Availability technologies ensures accessibility to the exact same source component for all pre-production and post-production activities including debugging, dump analysis, data/source manipulation and validation and SCM activities.

Centralized source storage further enhances the integrity of all development and corrective activities. It is crucial for quickly solving an application abend or performance problem because it ensures the correct version of the source code is immediately available, eliminating the need to “hunt” for the correct version, saving time and avoiding the errors that can be introduced by working with the wrong source. Integration and centralized source storage also provides the ability to perform immediate analysis of individual lines of code to identify the exact changes that occurred within a program and eliminates the need for detailed, exhaustive research — especially important when faced with a production problem. Most importantly, integration assures that corrective changes introduced by a developer do not inadvertently eliminate new functions introduced in previous changes.



Centralized source storage ensures that the correct version of source is always available for both Application Availability and SCM tools.

Without integration, developers lack direct access to the change history of a failing or problem program and face the unintended consequences of undoing past changes. Integration between SCM and Application Availability technologies provides developers with instant, interoperable access to change histories, providing immediate understanding of past changes, improving awareness of business issues and eliminating questions of why a particular change was introduced. Furthermore, change histories made available to Application Availability tools provide the benefit of an historical overview and insight as to the types of changes that have occurred over time. Change histories revealed during a debugging session, for example, makes it possible to identify problematic patterns that can be corrected accurately to prevent future failures.

Lastly, integration between SCM and Application Availability tools provides developers with the ability to simultaneously edit source and data, enabling rapid validation of record layouts, record formats and record types during the change lifecycle. It provides the ability to ensure the source code is ready to accommodate data while helping to ensure the faster implementation of quality code into the production environment.

## Avoiding Revisionist History

Facts are crucial to quickly identifying and correcting application problems. The facts provided by an SCM system reduce time required to research errors, provides immediate access to history and source and deliver the true historical account of change history to ensure the correct problems are addressed at the correct time.

Delivering the facts is made possible through the integration of SCM and Application Availability technologies. And true integration is only possible when infrastructures, databases and source code of the technologies can be fully shared and delivered from the same vendor.

Recognizing the pressing need for integration between SCM and Application Availability technologies, Serena Software has joined the powerful functionality of its Serena™ ChangeMan® ZMF and Serena™ StarTool® products. Serena is the industry-leading provider of Enterprise Change Management products. With more than 20 years of expertise, Serena has amassed more in-depth knowledge about managing change to enterprise applications than any company in the market. While other companies may provide separate, stand-alone SCM or Application Availability tools for the OS/390 or z/OS environment, none have the comprehensive, integrated solutions needed to protect production applications, improve Application Availability and reduce downtime — except Serena.

**Visit [www.serena.com](http://www.serena.com) to find out more**

To learn more about these products, and how to begin implementing comprehensive change management in your enterprise, visit Serena at [www.serena.com](http://www.serena.com) or write us at [info@serena.com](mailto:info@serena.com).

**serena™** Automating Change to  
Enterprise Code and Content

---

Serena is a trademark of SERENA Software, Inc. ChangeMan and StarTool are registered trademarks of SERENA Software, Inc. OS/390 and z/OS are either trademarks or registered trademarks of International Business Machines Corporation in the United States and/or other countries.

Copyright © 2002, 2003 SERENA Software, Inc. All Rights Reserved. WP\_401\_002\_1102v2