

●●● white paper



Strategies for Effective Change Management

Graduating from Version Control

Strategies for Effective Change Management: Graduating from Version Control

A White Paper by Rocket Software

Today's software development process must evolve to meet business demands. Simple tools that were originally designed to fulfill relatively isolated software development tasks and projects are no longer adequate. For all but the simplest of projects, the capacity of version control tools to meet current business and rigorous development requirements has been exceeded. This white paper examines how enterprise SCM solutions offer the means for aligning IT process with overall business needs.

Version 1.2 August 2014

© Rocket Software, Inc. or its affiliates 1990 – 2014. All rights reserved. Rocket and the Rocket Software logos are registered trademarks of Rocket Software, Inc. Other product and service names might be trademarks of Rocket Software or its affiliates.



Strategies for Effective Change Management: Graduating from Version Control



Introduction

Successful companies understand that evolving and adapting their business processes over time is essential for survival. It's a reality facing nearly every segment of business every single day; and IT—controlling many of the core processes that run today's enterprises—is the central driver for change. Faced with ever-growing, market-wide pressures, IT teams are seeking solutions from software configuration and change management vendors to meet the rapidly evolving needs of their businesses. Graduating from simple versioning tools to an enterprise software configuration management (SCM) solution could mean the difference between success and failure.

Surviving the Market

According to IDC, IT development teams must meet the following requirements to survive the next five years:

- ❖ Deliver quality, bug-free code
- ❖ Reduce complexity around integration
- ❖ Take the costs out of security
- ❖ Make innovations more timely, compelling, and less costly
- ❖ Offer software at costs commensurate with the benefits received and measured by relevant metrics



Quality software delivered on time and that works as intended is an absolute expectation, not a nebulous goal. To meet this demand, IT must commit to every stage of the design, development, and deployment lifecycle of their products.

So what can IT do to survive in this complex and demanding marketplace? Ever-improving SCM products are addressing critical development needs of developers, IT managers, and CIOs by:

- ❖ Helping IT organizations reduce spending and improve service
- ❖ Addressing the demand to more tightly align IT with business needs that are often driven by internal governance issues and external regulatory and standards compliance
- ❖ Reducing the complexity of integrating the work of geographically distributed development teams in multi-platform development environments
- ❖ Simplifying the development process associated with complicated applications
- ❖ Providing a better methodology for reusing artifacts, meeting shorter development cycles, and controlling and securing fast, reliable, and remote access
- ❖ Meeting the increased demand for greater administrative visibility into IT by providing a system of actionable metrics, offering role-based views of all application development and maintenance projects, and enhancing accountability standards

Savvy, forward-looking IT organizations have already adopted, and are benefiting from, the use of robust, integrated SCM products to meet this ever-increasing demand to adapt, control, streamline, monitor, and improve IT business processes.

Just Using Version Control?

There is still a tendency among some in IT to view development as a single, monolithic stage rather than accepting the fact that software development should be managed as a set of distinct, identifiable stages. Working with version control alone, these developers ignore the proven benefits of adopting a more comprehensive, process-oriented SCM solution, including: the lack of visibility and management of IT business processes, people and assets; limited regulatory and standards compliance; a slower time to market; increased business disruption due to software defects; and lower returns on investment. These deficiencies are all by-products of ignoring the benefits of process management in application development.

IT teams and individual developers who find themselves frustrated by today's complex, competitive environment must objectively evaluate their approach to the development process and the needs of the



business in general. One of the first steps in this evaluation process is recognizing the limitations of version control in managing the software development lifecycle. A version control system is an excellent tool for keeping track of versions of files. It is woefully inadequate in providing the infrastructure necessary for automating and enforcing software development best practices. A second, more critical step is accurately analyzing what an organization needs—and wants—if it is to survive and grow.

Version Control: Understanding Its Place in the Development Lifecycle

Version control is just one element in managing the design and deployment process of software. At its most basic level, version control provides a ‘history’ of a file or a group of files, keeping track of who changed what and when. Should a bug appear in a program, version control allows the source code to be inspected to discover which change or set of changes caused the error. Changes can be undone—temporarily or permanently—by having the version control software recreate a previous version of some file or files. Version control also allows for management of conflicting changes— for instance, when one developer deletes part of a file that another developer wants to keep, or two developers edit the same lines of code in the same file. A version control system can keep track of these problems and, in select cases, help with a resolution.

More advanced version control products serve as ‘gate-keepers, tracking who is working on what at the development check-out/check-in, merge, and promote levels. This “gate-keeping” requires a highly manual oversight and approval process, creating bottlenecks and making it very people-intensive. The cost of development and lack of efficiency can greatly increase over time, ultimately affecting the ROI of projects. In addition, version control products provide few, if any, accountability and traceability features, leaving a huge gap for regulated firms. Finally, many version control products need constant code workarounds, external plug-ins, and continual revisions in an attempt to keep up with software development needs.



While many version control tools may be adequate for simple, one-off development projects, their deficiencies in meeting the requirements of complex projects and diverse development environments are clear.

| Comparison of Version Control and Enterprise SCM Features | | |
|--|-----------------|----------------|
| Features | Version Control | Enterprise SCM |
| Check-out/check-in support | ❖ | ❖ |
| Version history | ❖ | ❖ |
| Branching & tagging | ❖ | ❖ |
| IDE integration | ❖ | ❖ |
| Source compare & merge | ❖ | ❖ |
| Change sets support | ❖ | ❖ |
| Enterprise application inventory management (multi-platform) | | ❖ |



Comparison of Version Control and Enterprise SCM Features

| Features | Version Control | Enterprise SCM |
|---|-----------------|----------------|
| Single point of observation and control across a multi-platform environment | | ❖ |
| IT business process automation | | ❖ |
| Workflow | | ❖ |
| Concurrent and multi-team development | | ❖ |
| Multiple release management | | ❖ |
| Automated distribution and deployment | | ❖ |
| Incident tracking | | ❖ |
| Approval and authorization | | ❖ |



| Comparison of Version Control and Enterprise SCM Features | | |
|---|-----------------|----------------|
| Features | Version Control | Enterprise SCM |
| Requirements management | | ❖ |
| Role-based interfaces | | ❖ |
| Management metrics and reporting | | ❖ |

Enterprise SCM solutions take over where version control products leave off, providing greater accountability and requirements management, more insight into the development process, increased automation, and better support for multi-team development. These features enhance productivity, decrease errors, and increase ROI.

Moving Up: Enterprise Software Configuration and Change Management

Version control tools address only one very limited aspect of the software development lifecycle. In today's highly diverse, competitive, and demanding IT environment, these tools simply fall short. Fortunately, the software configuration and change management environment has significantly evolved, and powerful SCM solutions are available. For developers and IT teams, finding the best SCM solution is a matter of evaluating needs, adopting a process-focused strategy, assessing the cost-to-benefit ratio, and selecting a solution.

Evaluating Needs

In selecting an enterprise-grade SCM solution, a series of simple questions can help define evolving requirements:

- ❖ Is there a defined, repeatable, and automated process for promoting application components to their respective environments (e.g., testing, QA, and production)? If so, how is the process monitored and enforced?



You'll Soon Find You Can't Live Without the ALM Hub

IT is in a constant state of flux. Business users are expecting the latest and greatest like never before. Because of this, development and operations teams are expected to deploy fixes and functionality faster than ever before. Which means the more complex your IT environment and development and release processes become, the more points of failure proliferate.

This is why the ALM Hub has become a must-have solution to monitor and orchestrate all the moving parts, processes, and people that touch the application lifecycle. Its ability to seamlessly bring together systems and workflows will minimize manual challenges and maximize productive collaboration. From anytime, anywhere mobile access, to end-to-end process automation, to error-free deployments, to point-and-click compliance reports to new levels of visibility, you will find that once you adopt the ALM Hub you won't know how you survived without it!

The ALM Hub from Rocket Software: your organization's key for unlocking business growth, doing more with existing technology investments & resources, and the compliant development "safety net" across your entire enterprise.

For more information on where to get an ALM Hub, contact Rocket Software at @rocketsoftware.com.



- ❖ Based on your development environment, decide which SCM features are important now. However, keep in mind that future needs may evolve and a tool with more features than you currently need may be more advantageous over the long term.
- ❖ Determine the configuration of your SCM solution (i.e., which source code, data, or documentation will be stored where; who will have access to what; how distribution of objects will be handled at various levels).
- ❖ Make sure that all users are aware of and ultimately accept the goals and processes that your SCM is designed to support.
- ❖ Determine if you will support multiple platforms, what they are now, and what they might be in the near and distant future.
- ❖ Assess where your development teams are located and how quickly and easily they will access the files and objects needed.
- ❖ Evaluate vendor responsiveness to your requirements and a demonstrated ability to provide the best ongoing support and service.

Refining your needs will help you identify a product that clearly fulfills your requirements.

Assessing the Cost-to-Benefit Ratio

Pressure to improve IT processes to meet business demands for higher productivity and improved time to market, while facing ongoing IT budget constraints is now a fact of life for IT. SCM products offer effective solutions to help alleviate much of this pressure. Demonstrable benefits of adopting and properly using an enterprise software configuration and change management solution include:

- ❖ Greater visibility and management of IT business processes, people, and assets
- ❖ Ongoing regulatory and standards compliance
- ❖ Faster time to market
- ❖ Reduced business disruption due to software defects
- ❖ Higher return on investment

Numerous industry case studies provide clear evidence that the outlay for an enterprise SCM solution easily outweighs the costs—and headaches—associated with losing data, releasing “buggy” software, and failing to meet release deadlines and regulatory standards. For many development teams, the expense of adopting the correct SCM product can often be offset by the savings in production time and reduction in errors during the very first project.



Conclusion

Today's software development process must evolve to meet business demands. Simple tools that were originally designed to fulfill relatively isolated software development tasks and projects are no longer adequate. For all but the simplest of projects, the capacity of version control tools to meet current business and rigorous development requirements has been exceeded. Enterprise SCM solutions offer the means for aligning IT process with overall business needs.

