

Approaching 2006 - CM Predictions

By Mario Moreira - January 2006

What is your New Year's Resolution? What about better identification, control, audit, and report? How about better Configuration Management (CM)? As we look in the horizon, what do we think will be hot in the CM landscape? It is a clean slate but most of us already have our plates full. As we plan for our Configuration Management (CM), we should consider where the CM field may be headed for 2006.

While cost will continue to play a key role in 2006 in software engineering, companies are focusing more and more on productivity gains in the form of painless integrations with one eye on the application lifecycle. With this in mind, what shifts might we see in CM for 2006? The envelope please...



The Predictions

What might we see by way of changes for CM in 2005? While, IMHO, I do not see dramatic changes occurring in the CM space, there are several areas where I predict shifts in direction. Here are some areas I perceive we may see changes:

Prediction #1: CM Tools that do Much More than Version Control

I predict we will see a shift toward establishing CM tools that have a broader range of functionality and in particular functionality that does not have to be integrated at the customer's expense. The classic CM tools focused on source control. More recently, some CM tool vendors have expanded on source control focusing on branching and merging, change sets, and build management. However, I see a shift where the consumers are looking for CM tools that better integrate with other tools in the application lifecycle and tools and require less integration effort. Two future scenarios come to mind:

- A CM technology that has more integration capability with other tools in the project lifecycle. This would include CM tools that more tightly integrate with process tools, requirements tools, reporting tools, deployment tools, etc. Some vendors are focusing more on ensuring their CM solution comes with supporting technologies in other spaces.
- A CM technology that requires or no integration effort to other tools. An example of this is a CM tool that has built-in defect tracking functionality. Many companies have had to struggle with integrating their CM technology with defect tracking or 'pass' on this capability. Even if they have established the integration, it requires time and effort to maintain the integration.

The advantages of a tight or effort-free integration are two-fold. To those that purchase this technology, the advantage of an integration between source control and defect tracking promote cost avoidance of establishing and maintaining the integration and the knowledge of the change relationship between code changes and defects. To the vendor, the advantage is that they can promote a tool that saves a company time and money. Here are examples of newer CM vendor products that either come with built-in integrations or promote tighter integrations:

- Source Control and Defect Tracking: AccuRev provides a built-in integration between its version control system and its defect tracking system. This is essentially out-of-the-box with no actual technology integration effort. To learn more about AccuRev, consider reading: <http://www.cmcrossroads.com/review/AccuRev37.pdf>
- Source Control and Process Workflow: MKS Source Integrity Enterprise is a source control tool that has the MKS Integrity Suite available to it. The MKS Integrity

Configuration Management Journal

Manager provides a framework for specifying and managing the workflow including approval and change management and can be used to manage issue tracking, metrics and related reporting, queries and charts. To learn more about MKS, consider reading: <http://www.cmcrossroads.com/review/mksintegrity.pdf>

- Source Control and Build Management: BuildForge FullControl provides a non-invasive management layer over existing build tools and integrates complex build, test, and deployment activities into a reliable, automated, and repeatable process. To learn more about BuildForge, consider reading: <http://www.cmcrossroads.com/toolspot/buildforge.php>
- Source Control and Peer Review: SmartBear Code Collaborator enables peer review of source code changes before or after files are checked into version control and automates audit trails, enforce workflow, and generate reports. To learn more about SmartBear, consider reading: <http://www.cmcrossroads.com/toolspot/smartbear.php>

Note: IBM Rational ClearCase, Serena Version Manager, and Telelogic Synergy also have integrated CM solutions (amongst others).

Prediction #2: Continued focus on Distributed CM

I predict that there will be continued effort to offshore software development and testing personnel. While this is not new, we will see a continued need for deploying and improving distributed CM systems for a more productive project infrastructure between local and remote personnel.

So what does this mean to CM professionals? We should become more knowledgeable of what distributed functionality CM tools provide and what CM processes can be used or improved upon that support distributed development. Some roles that CM can play include:

Global Change Control

It becomes critical to ensure baselines at all sites are managed together and in harmony. This includes the Requirements, Planning, Design, Code, and Infrastructure baselines where changes must be articulated to the appropriate site(s). Your current change control process may have to be adjusted to allow for a global approach using shared web-based document repositories, requirements technologies, and planning tools.

Global CM Infrastructure

The CM version control, replication, build, and release technology provides the infrastructure for the identification and control of much of the software development processes. While it is essential to have this infrastructure in place, due consideration must be given to a distributed or global CM strategy and accompanying infrastructure. A global CM strategy should include a:

Distributed Access Technology approach – Will you have the code base at one site or multiple sites? This will help you determine the technology that can help remote sites gain access to the source code. For multiple sites, consider either a Remote Client Snapshot or a Remote Server Repository technology. For single site repository, consider Terminal Emulation, Terminal Services, or Web Interface technology.

Code Availability approach - Will personnel at all sites have access to modify 'all code' or will personnel from specific sites have only 'sections of code' available for modification. This is important to determine since it can impact your project management approach and your branching and merging approach.

Branch/Merge approach – Will you have a branch per site or have all developers work on the same branch? This is important to determine since it can help the project better control changes coming from remote sites depending on how you structure your branching and who controls merging. Note, your approach to Code Availability (see above) may influence your branching and merging approach.

Prediction #3: Increased focus on CM Auditing

I predict that we will see a continued increase in process improvement initiatives with hopes of increasing productivity or ensuring compliance to set standards.

From a process improvement perspective, some of the drive for consistency is due to the distributed development model that needs a more consistent approach in order for two or more sites to work effectively and some of the drive is due to cost in hopes of improving productivity and doing more for less. Some initiatives will be oriented toward establishing a more lean or Agile (development lifecycle) methodology for development.

From the compliance perspective, we will continue to see certification or regulatory initiatives based on variations of

Configuration Management Journal

Capability Maturity Model Integration (CMMI), ISO 9001, Sarbanes-Oxley Act (SOX), and other related compliance or certification models.

So what does this mean to the CM professional? It is more important than ever for CM professionals to understand the certifications or compliance models such as the CM Key Process Area (KPA) in CMMi and SOX and understand the implications to their everyday work. In addition, when development methodologies are introduced to improve productivity, it is important to understand these approaches, such as incremental or iterative. You do not have to love the new approaches, but consider understanding them and their implications.

Summary

To summarize, as we look into 2006, I predict we will see large movements in the following areas:

- We will see a true shift from standalone CM technologies to integrated CM technologies. Those that claim to be integrated will be 'weeded' out by those that are truly or tightly integrated. Development shops will see the advantages of those CM vendors that truly have CM technologies that have built-in integrations with other tools or very tight labor-saving and productivity-gaining integrations.
- The software engineering field continues to be more global than ever. A distributed form of CM will continue to play a major role.
- The growing need for process and compliance support will continue. It is important for CM professionals to, at least, have an understanding of some of the leading process improvement or compliance models.

Finally, it is important for CM professionals to be ready to adapt to change that supports the organization. When considering a new CM technology or change to an existing CM technology, consider the cost avoidance or productivity improvements from built-in or tightly integrated technologies to a CM system and keep one eye on the full application lifecycle. When looking at development methodologies, understand that there are many models and that each model can have some benefit to your organization. For a compliance focus, ensure you understand the implication of the compliance model to CM and the development process. All of this will make the CM professional, both, more knowledgeable and valuable to the organization. Knowledge of these areas may also help in resume building and job hunting. ;-) **Have a great 2006!!!**

Mario Moreira is a Director/Architect of Technology, an Author of CM publications, and has worked in the SCM field since 1986. He has experience with numerous SCM technologies and processes and has implemented SCM on over 100 applications/products, which include establishing global SCM infrastructures. He has an MA in Mass Communication with an emphasis on communication technologies. Mario also brings years of Project Management, Software Quality Assurance, Requirement Management, facilitation, and team building skills and experience.

References

- Chapter 4 (Establish an SCM Infrastructure for an Application), section "5.3 Define a Global SCM/ Development Strategy", p90-95 of the "Software Configuration Management Implementation Roadmap" by Mario E. Moreira, 2004, John Wiley & Sons, Ltd Publishing
- Information on SCM technology:
 - AccuRev - <http://www.accurev.com/accurev.html>
 - MKS Source Integrity Enterprise - <http://www.mks.com/products/sie>
 - BuildForge FullControl - http://www.buildforge.com/products/buildforge_fullcontrol.htm
 - SmartBear Code Collaborator - <http://www.codehistorian.com/codecollab-overview.php>
 - IBM Rational ClearCase - <http://www-306.ibm.com/software/awdtools/clearcase/>
 - Serena Version Manager - <http://www.serena.com/Products/professional/vm/home.asp>
 - Telelogic Synergy - <http://www.telelogic.com/corp/products/synergy/index.cfm>

This article was reprinted by permission 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