

**THE STATE OF**

# Git in the Enterprise

By Eric J. Bruno



Sponsored by **CollabNet**

**InformationWeek**  
marketing services

**UBM**  
Tech

## Git in the Enterprise



### Executive Summary

By Eric J. Bruno

Git was created for development speed and according to our survey results, this is exactly why organizations from a wide range of sectors have turned to it. Its speed and powerful branching and merging have led to its adoption within companies of all sizes, both in terms of employee size and annual revenue. Git's adoption in the enterprise, is clearly seen as a shining point.

Specifically, Git is enabling development teams to be more agile and innovative, and to move at a fast pace. Survey data reveals that Git is almost always associated with leading-edge mobile, social and cloud-based applications, along with more frequent product iterations.

As with all tools, Git has shortcomings that lead to challenges and concerns among its users, and in particular with management in enterprise settings. Those key challenges are with regard to security, governance and administration at scale.

To make up for these shortcomings, enterprises are adopting a hybrid approach to source code management (SCM) usage, making it a vital if not major part of their development process. They also prefer to complement SCM with central administration and governance functions.

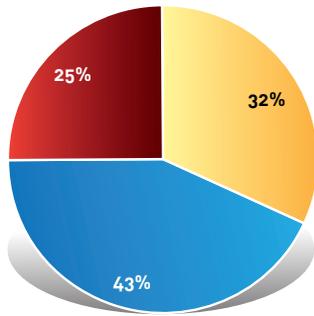
Source code management (SCM) tools – an otherwise mature technology – have seen an uptick in interest and activity in recent years, not at least due to the adoption of the agile methodologies. Historically, the main use of SCM tools is to manage change revision in application source code. The core set of features allows developers to view changes (known as deltas) between any two software module revisions, roll back troublesome or unintended changes, and ensure that multiple developers working on the same code base don't overwrite each other's work. Thanks to advanced conflict resolution, branching and merging, and support for distributed teams, SCM tools have become instrumental to the development process. In addition, managers' expectations for SCM tools include secure access to source code, central access to known source revisions, and packaging and deployment of tagged/named software releases.

### A Look at Git Adoption

Of all the SCM tools available, Git is a more recent open source entry into the market. Git was built with an emphasis on developer speed and powerful branching and merging, making it ideal for the growing number of development

# The State of Git in the Enterprise

**Figure 1.** Which of these titles best fits your role in the organization?



■ Management (incl. CIOs, IT mgmt., senior corporate mgmt., LOB mgmt.)

■ Developers (incl. programmers, software quality assurance, architects)

■ Other

**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

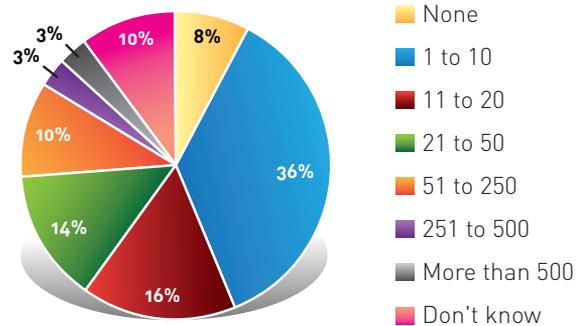
groups adopting the agile methodology. In a recent UBM Tech survey, which included responses from both developers and their managers, we specifically asked how Git was used, to get a feeling for its adoption in the enterprise. Survey responses were divided up by role, with 32% in a management role, 43% in a development role, and the remaining 25% in a support or other role (see Figure 1). For most of the survey questions, responses for developers and managers were very similar, with some exceptions that we'll explore later.

Looking deeper at the responses overall, we find consistent Git adoption across a wide range of industries. We do see slightly stronger adoption among technology-based organizations (i.e., hardware and software manufacturers) when compared to those in the government or finance space. However, these are in line with expectations, as the latter organizations tend to be late adopters in general.

According to the results, survey respondents work for companies of varying sizes, with the full set of size ranges well represented. Around 30% of the respondents work for large companies with more than 10,000 employees, another 30% in the range of 1,000 to 9,999 employees, and the remaining 40% at smaller companies with 100 to 999 employees. The survey results were filtered so as not to include those that worked for companies with less than 100 employees.

In terms of revenue, responses were again well distributed, with a strong showing from those organizations that earned at least \$1 billion dollars annually. About 20% of

**Figure 2.** How many Git repositories do you manage at your entire organization?



**Base:** 62 Git users

**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

the respondents willing to answer the question work for an organization that earns between \$1 billion to \$5 billion annually. About 27% work for a company that earns more than \$5 billion dollars, and only 3% work for a company that earns less than \$6 million annually. Companies of varying revenue sizes, as well as nonprofit and government agencies, were represented as well.

Let's take a quick look at how Git is being used within these organizations, according to the survey responses.

## Summary of Git Usage

Looking at how Git is used, we find that implementations are relatively small across the entire organization. For instance, only 16% of Git-only users manage 50 or more Git repositories, whereas most organizations fall within the minimal range of one to 10 repositories (see Figure 2).

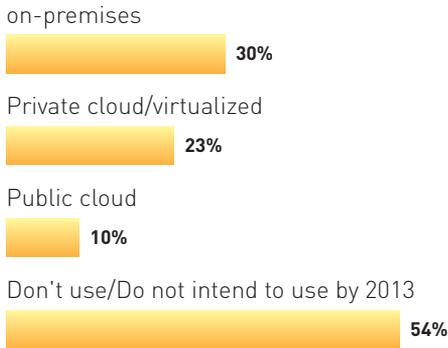
As for the location of these repositories, organizations clearly prefer to keep their source code on-premises. This may surprise initially, given that GitHub (delivered via public cloud) often is associated with Git development. However, this clearly suggests that enterprise customers have a strong preference for deployment methods other than the public cloud (see Figure 3). This is also increasingly the case for organizations that require both commercial support and development services for Git.

## Git Clients and Support

For the majority of survey respondents who use a Git client tool, the most popular choices include those that are either built into the integrated development environment (IDE) or those that work as a file system extension. In terms of vendors that organizations turn to for management or support, 88% of the respondents chose to complement

# The State of Git in the Enterprise

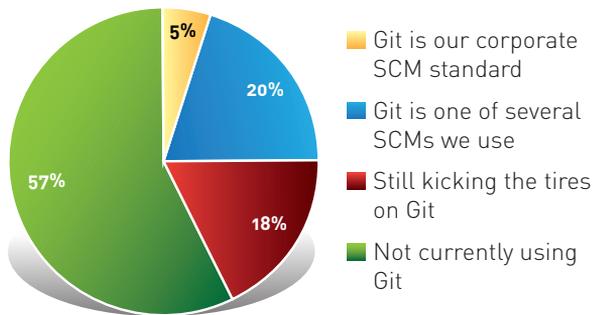
**Figure 3.** How do you use (or intend to use by 2013) Git deployment?



**Note:** Multiple responses allowed

**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

**Figure 4.** Which best describes your use of Git today?



**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

Git with a commercial tool for source code management (SCM) or application life-cycle management (ALM).

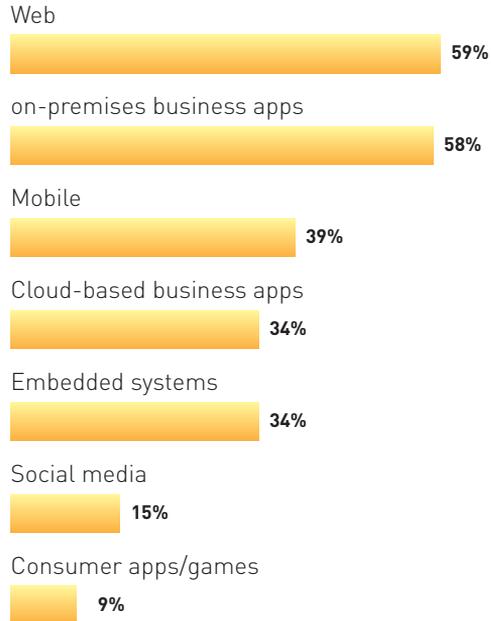
The survey responses we've examined so far give the general landscape for Git adoption. Let's look deeper at the responses, especially as they apply to enterprises, the agile methodology, and users' biggest concerns.

## Git Has Arrived in the Enterprise

Of all those who responded to our survey, almost half said that they use Git or plan to do so in the near future. In fact, nearly 25% have indicated that Git is one of the standard SCM tools they use, while 5% have declared it *the* corporate standard - implying they use no others - see Figure 4.

Almost 80% of the respondents further qualify their primary customer as enterprise in nature (either an in-house or business-to-business enterprise application),

**Figure 5.** What kind of software development are you involved in?



**Note:** Multiple responses allowed

**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

with the remaining 21% being of a consumer focus. Looking at the application types in more detail, we see that the majority are labeled as business applications, combining both on-premises, as well as cloud-based. Applications in the embedded, social media, or games spaces make up a smaller percentage of the responses (see Figure 5).

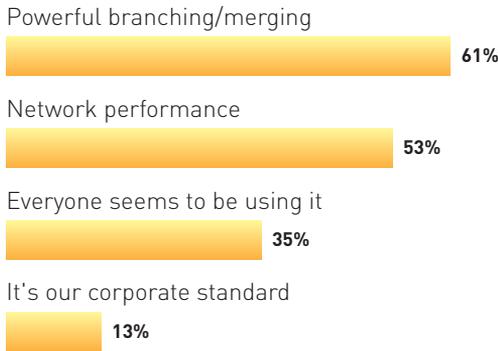
Taking a look back at company sizes both in terms of employees and annual revenue, the data supports the enterprise nature of Git adoption. Adjusting the figure to those who did not provide a response, this figure moves closer to 50%. Additionally, when cross-referencing the responses for the number of developers at the organization with the total number of employees, we see that almost 60% of the respondents work for organizations with at least 100 developers and 1000 employees.

These results add up to indicate that Git has arrived in the enterprise, with strong adoption across large companies and those developing applications with a business focus. Accordingly, distributed access is an important factor in terms of Git adoption, with 53% of its users citing its solid network performance as a positive trait (Figure 6).

This is one of those figures that varied when comparing responses from managers versus developers. Whereas

# The State of Git in the Enterprise

**Figure 6.** What do you like about Git?

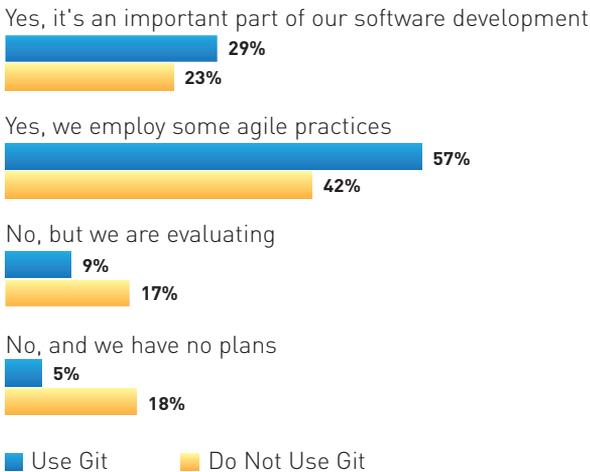


**Note:** Multiple responses allowed

**Base:** 105 respondents using or planning to use Git

**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

**Figure 7.** Do you practice agile software development?



**Base:** 62 Git users and 142 non-Git users

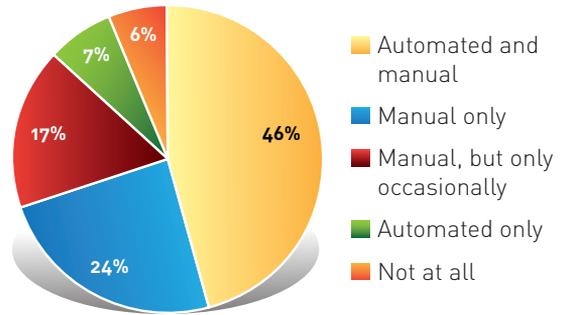
**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

both indicated that branching/merging is the most important Git trait, it had a much stronger showing with developers (at 79%) than with managers (53%). However, a far greater percentage of managers (29%) chose network performance as the trait they like most when compared with developers (at 16%).

## Innovation and Agility Thrive on Git

Beyond just the enterprise, the survey results show that Git usage has grown within organizations adopting agile methodologies, as well as those innovating at a fast pace.

**Figure 8.** How do you conduct code reviews?



**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

Git users show a higher preference for Agile development in general, although it doesn't end there. For instance, 70% of the respondents indicate that they're using Agile at least partially. When this data is cross-referenced for Git usage, we see this percentage soars to 86%, while it sinks to 65% for non-Git users (see Figure 7).

Looking back at the results, nearly 80% of the developers who took the survey chose branching and merging activity as a positive Git trait, and 68% chose it as the trait they liked most. Further, 41% of the respondents indicate they push code back to the central server at least once per day, while over 59% indicated they push more than once per day. Branching and merging, as well as syncing with the central repository, are common, important actions that have driven Git's growth.

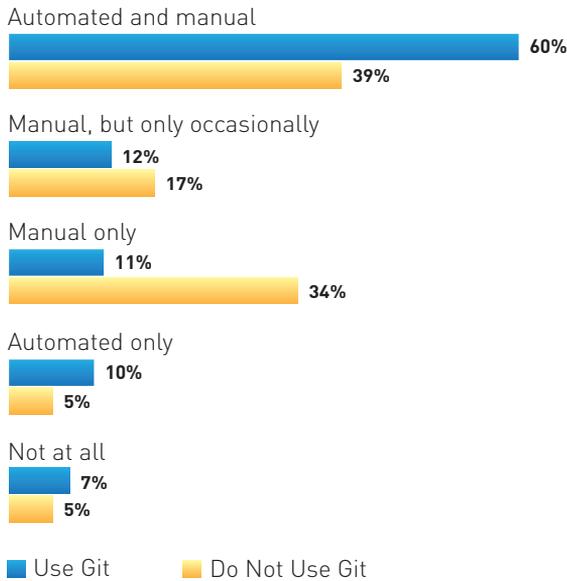
As part of the agile process, code reviews and demonstrations are commonly practiced. Only 6% of the survey respondents said they *don't* conduct code reviews at all, while nearly half of the respondents indicate they use a mixture of a manual and automated code review process (see Figure 8).

Looking at this response when cross-referenced by Git users versus non-Git users, we see some interesting results. The survey reveals that 70% of Git users use some form of automated code review process, compared with only 44% of non-Git users (see Figure 9).

Gerrit is widely considered to be the leading code review tool in use. That makes sense, given that it promotes enterprise-grade use of Git, and that it's the standard tool for entire development communities, such as Android developers. The survey results indicate there's some room for it to grow, as nearly half of the respondents indicate they don't use Gerrit, and 38% indicate they've never heard of it (see Figure 10).

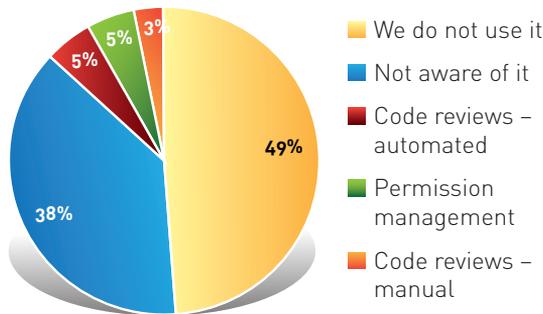
# The State of Git in the Enterprise

**Figure 9.** How do you conduct code reviews?



**Base:** 62 Git users and 142 non-Git users  
**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

**Figure 10.** What is your primary use of Gerrit?



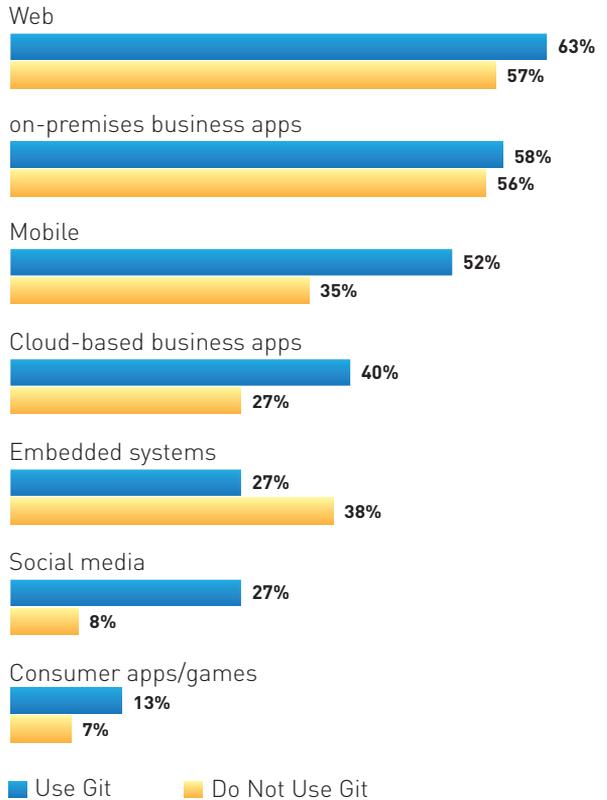
**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

## Cutting-Edge Applications

Looking at innovation, and fast, cutting-edge development groups, we see Git users are more likely to be working on apps for mobile, social media, and the cloud when compared with non-Git users (see Figure 11).

Overall, the survey results indicate that fast teams leading the way in terms of innovation, application type, and agile processes use Git. If you take the inverse of this statement, this trend implies that Git best positions development teams to be agile and innovative, and enables them to move at their fastest pace.

**Figure 11.** What kind of software development are you involved in?



**Note:** Multiple responses allowed  
**Base:** 62 Git users and 142 non-Git users  
**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

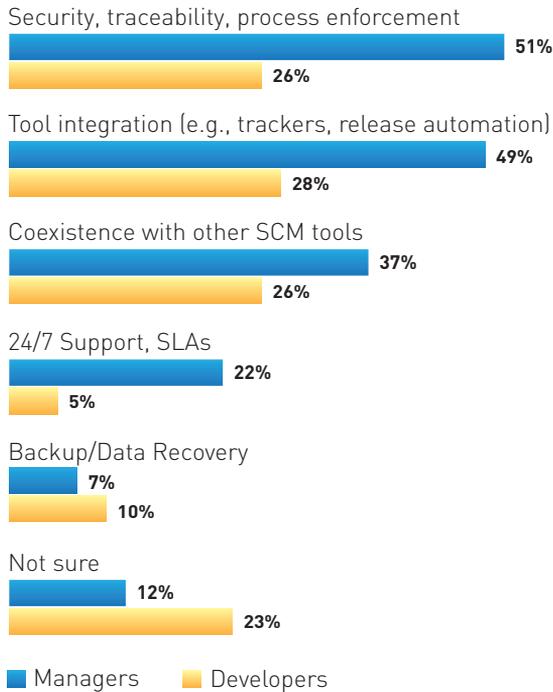
## Addressing Reservations About Git: Hybrid SCM Approach, and Tool Governance

In spite of the advances in innovation, enterprise adoption and faster development pace, Git users (and management especially) do have real concerns. Among these are integration, security and support. Overall, integration was chosen most as a concern (37%). Coexistence with other SCM tools follows closely behind at 36%, while security/traceability/process enforcement came in next at 33%. These numbers vary significantly for developers versus managers. For instance, more managers indicated security as the top issue, with 51% being concerned about it, closely followed by tool integration (49%) and coexistence with other SCM tools (37%); see the full breakdown in Figure 12.

That is not entirely surprising, since Git breaks with one traditional paradigm of all other version control systems: Sacrificing immutable commit history for greater developer flexibility and possibility to remove problematic content

# The State of Git in the Enterprise

**Figure 12.** What do you consider Git's top challenges?



**Note:** Multiple responses allowed

**Base:** 41 managers and 39 developers using or planning to use Git  
**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

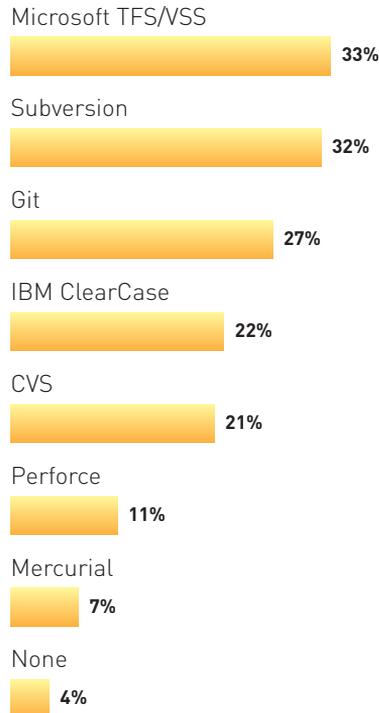
forever (obliterate feature). From a management perspective that's a security and compliance risk.

When asked about features that need improvement, survey respondents most often cited administration, security management, and performance on Windows as their top concerns. Additionally, only 7% of the respondents indicated they use Git for the version control of binaries; of the remaining, 45% said they use a non-Git tool such as SVN, while 13% use some sort of transparent, hybrid approach.

Looking at the hybrid approach in more general terms, we see that most enterprises use two or more SCM tools (see Figure 13). Subversion, Microsoft TFS/VSS and Git being the three most-used SCM tools overall. When the data is cross-referenced, we see the majority of Git shops (36%) also use Subversion in their hybrid approach, with CVS coming in distant second at 27%.

Additionally, for 84% of those who responded, central management of multiple SCM tools and repositories was considered critical or somewhat important. This included features such as role-based access control (RBAC), central monitoring and end-to-end traceability (see Figure 14). This clearly suggests that while organizations like Git, they

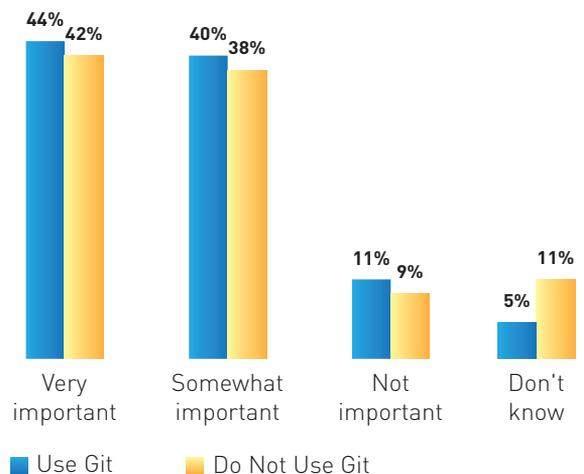
**Figure 13.** By the end of 2013, which SCM tools do you plan to use?



**Note:** Multiple responses allowed

**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

**Figure 14.** How important is central management of multiple SCM tools and repositories (i.e., RBAC, monitoring, traceability)?



**Base:** 62 Git users and 142 non-Git users

**Data:** UBM Tech survey of 248 development and business technology professionals at companies with 100 or more employees who use source code management (SCM) tools, January 2013

# The State of Git in the Enterprise

consider it imperative to complement management tools (whether open source or commercial).

In summary, the data shows that while Git does have perceived shortcomings - in particular with regard to security and integration - its positive traits for developers mean its adoption in the enterprise is quickening. Git, when used with other SCM and management tools to address enterprise concerns, makes for a powerful tool in a multi-pronged approach to productive software development.

## Conclusion

Git was created for development speed, and according to our survey results, this is exactly why organizations from a wide range of sectors have turned to it. Its speed and powerful branching and merging have led to its adoption within companies of all sizes, both in terms of employee size and annual revenue. Git's adoption in the enterprise, often led by a more conservative bunch, is clearly seen as a shining point.

Specifically, Git is enabling development teams to be more agile and innovative, and to move at a fast pace. Survey data reveals that Git is almost always associated with leading-edge mobile, social and cloud-based applications, along with alternative development languages, newer tool usage and more frequent product iterations.

As with all tools, Git has perceived shortcomings that lead to challenges and concerns among its users, and in particular with management in enterprise settings. Those key challenges are with regard to security, governance and administration at scale. To make up for these shortcomings, enterprises are adopting a hybrid approach to SCM usage, making it a vital, or perhaps even major, part of their development of their development process. They also prefer to complement SCM with central administration and governance functions.

## About CollabNet

CollabNet is a leading provider of Enterprise Cloud Development and Agile ALM products and services for software-driven organizations. With more than 10,000 global customers, the company provides a suite of platforms and services to address three major trends disrupting the software industry: Agile, DevOps and hybrid cloud development. CollabNet TeamForge® with Gerrit provides central governance for Git, providing enterprise-grade security and compliance. Learn more at [www.collab.net/gotgit](http://www.collab.net/gotgit).

## About the Author

Eric J. Bruno is a technology consultant specializing in software architecture, design and development. His experience ranges from client/server and highly distributed development, to multitiered Web and transactional software development.

