WHITE PAPER

Leveraging Life-Cycle Management for Software and Business Adaptability

Sponsored by: Borland

Melinda-Carol Ballou
October 2006

IDC OPINION

Global 2000 organizations must unite application life-cycle management and quality across complex, distributed environments and be increasingly accountable to global compliance and other pressures. Staff at all levels — from management to development — are being driven to connect the software development life cycle from project inception through high-quality deployments. The coordination of quality through requirements, testing, change management, modeling, and portfolio prioritization impacts successful implementations, business, and IT productivity, and savvy IT organizations are focusing avidly on this area. This coordination is particularly important in the context of pressures on Global 2000 corporations, including offshoring and outsourcing, regulatory compliance and emerging new development paradigms with service oriented architecture (SOA), software as a service (SaaS), and open source. The purpose of this white paper is to lay out the role that life-cycle management and development play to enable effective and adaptive business/IT collaboration. It also discusses the overall impact on corporate and IT success of requirements, change management, modeling, testing, and best-practice life-cycle approaches.

IN THIS WHITE PAPER

This white paper considers the interaction between requirements and other life-cycle phases, with an emphasis on testing, change and code management, and effective processes to enable more adaptive approaches to software development and changing competitive environments.

SITUATION OVERVIEW: MARKET TRENDS AND EVOLUTION

Business users, analysts, and IT staff struggle to communicate effectively about business requirements. As a result of the cultural divide across groups and ineffective common communication vehicles, IT staff all too often create projects with little relevance to key business needs. Even where communication and requirements capture are decent initially, business change and lack of a common language across groups still drive misunderstanding and poor iterative development approaches. In addition, poor coordination between requirements, modeling, testing, change, and IT portfolio management results in disconnects for quality control, irrelevant applications, and poor prioritization of IT resources for change requests and iterative development.

The disconnect between requirements and testing impacts user quality viscerally. Amorphous or outdated requirements lead to irrelevant functional tests, inadequate parameters for performance testing, and anemic service-level agreements (SLAs), among other problems.

The Top 5 Trends Driving Software Life-Cycle Quality Improvement

The overall impact of poor life-cycle quality coordination is particularly challenging in the context of increasingly complex development environments. We see five critical trends that are disrupting existing, ineffective approaches and enabling a push toward more consistent life-cycle approaches for software development. They are complex sourcing, Sarbanes-Oxley (SOX) and other regulatory compliance initiatives (in the context of a global economy), the emergence of SOA approaches for development, open source adoption, and SaaS and "on-demand" models for licensing and deployment.

Distributed development of internal resources, offshoring and outsourcing, and the need for collaboration across groups that are separated geographically and culturally demand effective communication vehicles. Such vehicles include best-practice approaches to requirements gathering, testing, and change management, as well as automated technologies to enable more effective communication about content, modeling, business processes, and so forth.

In addition to complex sourcing, regulatory compliance initiatives mandate auditability and rigor with regard to project and program evolution, quality, and change management. Organizations with poor life-cycle management in core areas — including requirements, testing, and change management — run significant risks with regard to regulatory compliance requirements. The need for compliance runs the gambit from legislation such as SOX and Basel II to localized compliance initiatives for Global 2000 organizations managing their businesses worldwide. Thus, any company doing business globally must address this range of compliance issues.

Evolution toward a SOA approach for development by many organizations also demands effective requirements gathering and management. There is no point in investing in SOA development unless core business needs are represented and tested in the resulting services that are created. Challenges in the ability of Global 2000 organizations to effectively communicate business needs and processes result in failure and wasted investment in SOA initiatives.

Drivers for increased complexity that have also facilitated life-cycle adoption include open source (greater availability of "free" quality software for development and life-cycle management) and SaaS hosted and shared infrastructure models for licensing and tools deployment. While open source has seeded a broader, previously uninitiated market for life-cycle adoption, SaaS helps facilitate adoption of life-cycle products by decreasing barriers posed by implementation and internal support costs.

These factors create a "perfect storm" of complexity and disruption that increases both the drivers for life-cycle adoption of automated technology and enables software purchases and implementation through increased expenditures (e.g., regulatory compliance and complex sourcing). As a result, Global 2000 organizations are moving to adopt more effective approaches to requirements, modeling, testing, and change management. Coordination in this context enables a "quality life-cycle" approach across life-cycle phases.

Pursuing Effective Requirements Management Across the Quality Life Cycle for Testing and Change Management

Requirements and quality management are inextricably linked. If requirements pursued by IT in creating software are inaccurate, irrelevant, and/or outdated, projects fail. Effective testing demands functional tests that reflect key business demands and that lay out a framework for appropriate performance testing and optimization to meet user needs and SLAs. The creation of functional tests early and iteratively in the life cycle helps solidify requirements that might otherwise remain amorphous and nebulous. Establishing an approach for performance testing also sets the course for development and deployment. As the project progresses, change management decisions should be prioritized based on key business imperatives (e.g., effective, current, relevant requirements). In turn, once made, those changes need to be tested to ensure that they don't break the application and create defects and also that they are in line with key functionality determined by melding requirements with functional tests. This triumvirate of life-cycle processes should be closely married and integrated for project success.

The benefits are clear for agile iterative approaches to development that include shorter cycle times, the ability to more quickly respond to key business needs (because the business must prioritize and the delivery times are faster), and requirements that are more limited, delineated, and clear cut. IDC sees the strong emergence of agile approaches in a variety of contexts (e.g., Web development and SOA). Yet as organizations pursue agile development approaches, the possibilities for failure as a result of poor requirements, testing, and change management increase. Stitching these smaller projects together and being responsive to changing, iterative requirements demand both effective test and change management as part of the quality life-cycle process. Again, criteria for changes and prioritization of testing resources must be tied to requirements that have the greatest business impact. Pulling together these aspects of the life cycle within automated tool suites is a technology requirement to address the complex "perfect storm" of emerging development environments described earlier. These tools must be accompanied by effective process and organizational change to support a culture for life-cycle quality management that includes the elements of requirements, test and change management, and build automation and modeling.

Technology Requirements and Criteria: Laying Out an Approach

Criteria for successful and agile requirements and quality management within the enterprise include the links between key business stakeholders and IT development staff. In determining core functional requirements for quality management, organizations might find it helpful to ask a series of questions that target tools functionality, process, and organizational evolution. How mature is the organization from a life-cycle process and organizational perspective, and where are the greatest pain points? How well integrated are vendors' requirements tools with testing? How intuitive and accessible are the requirements tools for mere mortals (e.g., business personnel and staff unfamiliar with arcane user interfaces or complex modeling paradigms)? Do links exist between requirements, modeling and change management, and testing with change management? Can the quality life cycle, including testing and other resources, be effectively prioritized by leveraging requirements? What impact analysis is available with regard to existing systems, defect tracking, and metrics to obtain performance optimization? What are key process and organizational issues, and how are they related to and integrated with technology solutions? How do business users struggle with disparate approaches and inadequate coordination with IT? What are key benefits of a different requirements management approach that coordinates throughout the life cycle? What are the risks and costs of refusing to incorporate a quality life-cycle approach? These questions and others can serve as a starting point for making the transition to coordinated life-cycle approaches.

From an IDC perspective, functionality across the quality life cycle includes intuitive requirements development and gathering capabilities to pull in key business information, which, in turn, will feed requirements management and modeling tools, integration between requirements and test management tools to facilitate creation of appropriate test scripts for functional tests, current and relevant information with regard to performance and deployment needs, and bidirectional data feeds to garner information with regard to resource capabilities to help feed metrics and decisions about appropriate staff allocations on comparable future projects. In addition, the ability to coordinate requirements with testing facilitates creation of more effective, targeted requirements. Integration with change management and defect tracking and analysis tools helps facilitate prioritization of change requests and metrics to support key business benefits and improved resource allocation. Build management and code analysis enable additional insight and metrics for a quality development process and better code management. All of these factors potentially cut costs, but even more importantly, they enable business responsiveness and adaptability to highly competitive global markets.

Acquisitions and Mergers: Targeting Coordinated Solutions

It is significant that a series of vendors across the life-cycle arena have purchased and/or closely integrated requirements, change management, and, where possible, testing tools and processes over the past several years. The acquisition of requirements tools by SCM vendors over the course of the past two to three years underscores the growing market trend toward confluence in those areas (e.g., IBM Rational, Borland/Starbase/Caliber, Telelogic/DOORS/Continuus, and Serena/RTM/Apptero). Concurrently, acquisitions and attempts to more closely ally testing with both requirements and SCM underscore growing vendor commitment in this area (Compuware/SteelTrace, and Borland/Segue). The push toward pulling in build management and code analysis is exemplified in other recent M&A activity (IBM/BuildForge, BMC/Identify, and Borland/Gauntlet). Another obvious area of confluence is IT portfolio management to enable better prioritization for IT projects and programs and coordination with the overall IT portfolio (ten acquisitions over three years, four within the past year, most recently HP/Mercury). At their best and at a granular level, effective requirements can help to inform and drive decisions with regard to testing, change, and even portfolio management by providing business context and traceability back to key business benefits.

The increase in depth and breadth across application development life-cycle management products does not, however, address the core areas of process and organizational change that are necessary prerequisites to successful, quality life-cycle implementations.

The biggest barrier to adoption for these products is the human barrier. Companies must put in place effective process and workflow strategies, among others, to enable effective and consistent communication between business and IT. They must also establish an organizational framework with sufficient executive management clout to sustain the initiative and support success. These integrated approaches can facilitate leverage of information from the combined toolsets. Coordination with effective methodology and organizational approaches can enable the cultural transition to those shifts in process.

This white paper considers Borland's approach to these Global 2000 challenges.

Borland Points of Differentiation and Approach to the Life-Cycle Management Problem

Borland's push into the life-cycle arena beyond development began with its Starbase/CaliberRM acquisition three years ago, augmenting the existing Borland modeling and development tools with requirements (CaliberRM) and software change and configuration management capabilities (StarTeam). In the past year and a half, Borland acquisitions have focused on process and methodology (with TeraQuest consulting staff and content), basic portfolio management capabilities (with Legadero), and most recently, quality control and testing (with Segue). As Borland divests of its development environments to squarely target application life-cycle management across areas, a push toward coordination of requirements with a quality life-cycle approach has been a main focus. It incorporates Borland's past focus on and experience with development and is the primary point of departure for the company.

With CaliberRM combined with the recently released DefineIT, Borland lays the groundwork for combining more intuitive requirements gathering with back-end management for auditability and traceability. Integration between requirements and the Together modeling suite (shipping now) positions Borland to create models that are aligned with requirement demands by users. This release of Together also provides Eclipse integration and focus on SOA (addressing several of the trends discussed earlier in this white paper). Based in part on its earlier close partnership with Segue prior to acquiring the company in 2Q06, Borland is launching integration between its requirements tools and the Segue testing and test management products. In conjunction with StarTeam for change and configuration management integration (shipping now) and Gauntlet's code and build analysis and management capabilities (integration due to ship in 4Q06), this combined approach for quality life cycle enables coordination across key life-cycle phases. These product capabilities include the potential for closer business alignment for software initiatives, improved quality and prioritization of software changes, and comprehensive management of development in the context of existing infrastructure. Cost savings could accrue with regard to software success due to greater relevance, coordination of testing with core requirements and performance needs, greater rigor, better prioritization and control for change management, and more effective defect management in coordination with testing. These capabilities also lay the stage for greater IT and business responsiveness to global markets in the context of a comprehensive, quality life-cycle approach. Although Borland is divesting its IDE business, its prior historical focus on development provides a strong context for software project deployment as it seeks to shift its business to target the overall software development life cycle.

Yet the complexity of this approach has its challenges. Borland is up against the largest vendors in the software market as it targets a mature, integrated life-cycle management arena (e.g., Microsoft, IBM, CA, Compuware, HP/Mercury). In its efforts to pursue this space, Borland needs to emphasize its agnosticism with regard to IDEs, platforms and technologies and the productivity and ease of use of many of the tools in its portfolio. Also, Borland's tools offerings can leverage capabilities from a CMMI process and organizational consulting perspective resulting from its TeraQuest acquisition, which is differentiating.

CHALLENGES AND OPPORTUNITIES

Combining the full breadth of solutions that have been separately acquired is demanding, both for users and for Borland. Bidirectional integration across products must work seamlessly. Key to this strategy is the underlying process approach and shift enabled by Borland's consulting arm. As users implement an increasingly integrated quality life-cycle approach with automated tools, they must also incorporate effective process and organizational change to help disparate parts of the organization better coordinate with one another. Borland is already engaged in establishing closer integration across products (the majority of which were acquired more than three years ago).

In addition, it is vital to retain integration with non-Borland tools and partners who now are "co-opetitive" in the testing arena (such as HP/Mercury and Compuware) and elsewhere, as Borland has been able to do previously to some extent (in the requirements, SCM, and modeling arenas).

Borland will increasingly be up against both niche players and behemoth vendors and must execute nimbly and adaptively in a highly crowded, consolidating and challenging market. The opportunity for Borland resides in the multiple factors driving user adoption and market growth presented earlier and the productivity and usability of its chosen tool suite combined with strong process and organizational approaches.

SUMMARY

Overall, the five disruptive elements discussed in this white paper are pushing Global 2000 organizations toward more effective quality life-cycle management and governance. Companies have no choice in these highly competitive environments, dominated by regulatory compliance initiatives and complex sourcing. They must adopt better coordinated, more rigorous approaches to requirements and quality management or face the risk of failure and competitive loss.

Increasingly, IDC sees Global 2000 organizations beginning to address the coordination between requirements and other life-cycle phases more effectively as a result of industry and market pressures. These factors are combining to drive organizations toward greater rigor and at the same time enable funding of these types of initiatives (via compliance and the need to manage complex, distributed staffing).

Shifting existing practices and behavior to adopt coordinated approaches to requirements, modeling, testing, and change management is key, however. For success, it is vital that business and IT staff transition to coordinated practices and organizational governance approaches as they adopt integrated life-cycle management technologies. In this way, they will be better able to gain the adaptability and cost benefits of a coherent, cohesive approach to application development and a dynamic, responsive IT/business relationship for competitive advantage in today's global markets.