

The Evolution of Build and Release Management for Effective Software Delivery

A Customer Survey with Case Studies

Judith Hurwitz, President and CEO

Robin Bloor, Partner



Published October, 2007, this paper is an update to
the Hurwitz customer benefit study entitled:

Transforming the Application Development Assembly Line

which was first published by Hurwitz & Associates
in November, 2005



A GreenPaper™ Customer Benefit Study

© Copyright 2007, Hurwitz & Associates. All rights reserved. No part of this publication may be reproduced without the prior written permission of the copyright holder or the copyright holder's licensee.

□ 233 Needham Street □ Newton, MA 02464 □ Tel: 617-454-1030 □
www.hurwitz.com



Contents

Introduction	4
Rational Build Forge After the Acquisition	5
Software Development Trends and Rational Build Forge	7
Proliferation of Technology and the Global Workforce	7
The Advent of SOA	8
The Influence of the Internet	8
Compliance and Governance	9
Impact on the Build and Release Process	10
The Rational Build Forge Survey	10
1. Respondent Demographics and Product Usage Patterns	12
2. Customer Satisfaction Indications and Measurements	15
3. Respondent Customer Experiences: The Technical Benefits	17
4. Respondent Customer Experiences: The Business Benefits	20
Case Studies	22
Rational Build Forge integration and process automation yields substantial productivity gains in global software company	22
Rational Build Forge allowed a global insurance company to manage compliance requirements in a cost effective manner	23
A financial services company replaces home-grown tool with Rational Build Forge to deliver on their service oriented architecture strategy	24
Conclusion	26



Introduction

The nature of applications development and deployment is changing dramatically. The change is being driven by:

- The escalating complexity of corporate software development environments, both for building new applications and maintaining legacy applications.
- The organization of software development using geographically disparate and, often, global development teams.
- The growth in importance of the Internet as an infrastructure platform for customer facing applications.
- The emergence of Service Oriented Architectures (SOA), resulting in the creation of virtual applications composed of business services that are linked together.
- The pressures for compliance from internal and external sources that have placed greater accountability on development teams to prove that their applications are built using reliable and secure practices.

The impact of these trends is that organizations are looking for tools that will allow them to package software that is reliable, predictable, and scalable no matter how that software is built or used. These trends necessitate the need for sound build and release management practices to ensure the high quality and timely delivery of software to power the business.

Without effective ways to reuse common components, automate production throughout each phase of the software development life-cycle, manage quality through repeatable processes and track results for the purpose of auditing and reproducibility, companies will struggle to keep pace and stay competitive.

The Rational Build Forge survey - A brief summary

To validate these trends, we conducted an in-depth quantitative and qualitative study of 18 Rational Build Forge customers – this report shares the results. We studied a broad cross section of the Build Forge installed base, including a mix of large and small development teams across a variety of industries.

We wanted to understand the motivations that caused these companies to examine Rational Build Forge, and quantify the business and technical benefits these companies had achieved from implementing their products. The results of the survey we carried out were quite compelling.

Customers reported significant improvements in their development speed, team productivity, and product quality that led to quantifiable cost savings in their organizations. The table on the following page provides more details of the benefits.

The impact of these trends is that organizations are looking for tools that will allow them to package software that is reliable, predictable, and scalable no matter how that software is built or used.

Benefit	Average Improvement	Highest Improvement
Speed of builds and releases	110%	500-2000% (or 5-20x)
CM team productivity	42%	90% or greater
Release frequency	40%	90% or greater
Error reduction	30%	70-80%
Developer productivity	28%	81-90%
Development cost savings	25%	50-70%

Customers reported significant improvements in their development speed, team productivity, and product quality that led to quantifiable cost savings in their organizations.

Further, we found that these benefits translated into significant, measurable ROI for the customers surveyed, typically within 3-6 months of purchase.

Customer	Payback on Investment	Increased Productivity	Overall Time Savings	1 Year Accumulated Savings	3 Year Accumulated Savings
1	Over 200%				
2	1-20%	60-80%	Over 200%		
3		40-60%			
4	101-200%				Over 200%
5		101-200%			
6	1-20%	40-60%	20-40%		40-60%
7	20-40%	40-60%	1-20%	60-80%	

This paper begins with an update on BuildForge’s evolution since the acquisition by IBM in May, 2006. We then discuss the current development trends that are driving the need for better build and release management practices. Finally, we will share results of the detailed customer survey and some in-depth case studies to explain how customers are using Rational Build Forge software to address these development challenges to create measurable results in their organizations.

Rational Build Forge After the Acquisition

Prior to its acquisition by IBM in 2006, BuildForge was a fast growing software company and the dominant provider of build and release management software. Its acquisition by IBM was not surprising for three reasons:



The Evolution of Build and Release Management

- BuildForge had been an IBM Business Partner since 2004 and they had many successful implementations with common customers.
- BuildForge's build and release management software was purchased and adopted by the Rational development teams for internal use.
- IBM was, and still is, building a comprehensive software development stack under the Rational brand, which was missing a product with BuildForge's build and release management capabilities.

The acquisition made sense; however, the question was how successfully BuildForge would be integrated into the IBM mainstream. In the year since the acquisition, Hurwitz & Associates has observed the following changes:

- Rational Build Forge has matured as a product, and is providing added value and usability for its customers. The Rational Build Forge user interface has been given an impressive overhaul providing a much improved ease of use.
- Rational Build Forge has been tailored to fit neatly into the Rational product set and has the interfaces needed for it to fit seamlessly as the build and release engine for most of IBM's software development environments. Ultimately, Build Forge will be the underlying engine for many of Rational's new product offerings.
- Rational Build Forge has gained significant traction in corporate development environments, and has extended its global market reach. While BuildForge previously had a small corporate client base, the majority of its customers were ISVs. Now the customer base is more balanced between ISVs and corporate software development organizations, and has reached into virtually every industry vertical.
- IBM's strong presence across the globe has contributed to significant growth and market adoption in Europe and Asia. The introduction of local language support in version 7.0.2 promises to accelerate their worldwide momentum even further.
- Rational Build Forge is also expanding its presence in the small and mid sized market with the introduction of Rational Build Forge Express which provides a lower entry point and financing options for smaller teams. We believe this will prompt smaller teams to adopt better build and release management practices.

It is clear that Rational Build Forge has made great progress since the acquisition by IBM, in terms of product maturity, market penetration, and continued industry leadership. We view its build and release management solution as a key enabler to help companies address many of the daunting challenges that face today's development environments.

Rational Build Forge has gained significant traction in corporate development environments, and has extended its global market reach.

Software Development Trends and Rational Build Forge

In the mainframe era that ended in the 1980s, there was a remarkable consistency in the software development environments that companies used. There were only a handful of different programming languages, accompanied by a smattering of 4GLs, database products and other related tools. The applications companies built handled structured text and nothing else, and the development process was relatively straightforward.

In the last 25 years, there has been an explosion in both the number and diversity of applications. Today applications don't just run on mainframes and servers, but on PCs, laptops, PDAs, cell phones, embedded chips and even RFID tags. They manipulate every possible kind of data including unstructured text, images, sound and video. Software is ubiquitous, multifaceted, and constantly changing.

The application explosion that has occurred has been accompanied by an explosion in programming languages and specialized development tools. In fact, the number of different programming languages is now larger than the number of human languages.

A consequence of the growing complexity and variety of development options is that, for most companies, the build and release process is neither documented nor standardized. It is usually poorly automated, often consisting of collections of non-standard build scripts.

Many of the current trends in software development and some of the most pressing customer issues hinge on the necessity to incorporate intelligent automation into the software build and release process. Let us examine them.

Proliferation of Technology and the Global Workforce

Software continues to pervade every aspect of human activity. While this trend itself is not new, software is proliferating at a dramatic pace. Three factors have increased the velocity of this trend:

- Computer technology continues to increase in power, capability and accessibility, while it simultaneously decreases in cost. This trend continually opens up new opportunities for software applications.
- The Internet now provides a ubiquitous platform for accessing and distributing knowledge, information and skills. Therefore, the barriers to building and deploying software are low and falling.
- The number of people involved in building software has exploded. Estimates suggest that there are about 13 -14 million programmers and that their number is growing at around 10 -12 percent a year, with the population growing fastest in developing countries.

Many of the current trends in software development and some of the most pressing customer issues hinge on the necessity to incorporate intelligent automation into the software build and release process.



The Evolution of Build and Release Management

While technology collapses in price, the skills pool just keeps on expanding. It would seem logical to conclude that software is getting less expensive to create and that opportunities for new software in mature business areas are drying up. This is far from the reality.

Given the speed with which software is evolving, developers are faced with a constantly moving target. On the one hand, technology changes so that software has to be rewritten for different platforms and access devices. On the other hand, this same software has to be rewritten because of changes in the approach to business strategy. So software developers seem to be continuously “reinventing the wheel”. Unless development teams are able to find better solutions to reuse assets, leverage their work more efficiently, and build more adaptive software architectures, it will be difficult to achieve any true economies of scale that lead to sustainable competitive advantage.

The Advent of SOA

The Service Oriented Architecture (SOA) movement is an effort to create this kind of economies of scale. With SOA, the business applications of an organization are gradually transformed into a set of reusable components that can be threaded together to directly support end-to-end business processes. The reuse of software components is not only possible with SOA, it is a guiding principle of the architecture.

The corporate adoption of SOA is now happening rapidly, driven by the fact that this development approach reduces overall effort and accelerates the speed of software delivery. This movement brings a strong emphasis on building versatile and high value components, and then determining how they can best be leveraged across diverse business applications.

SOA development is still immature in many areas. Effective SOA development requires greater standardization of development platforms, testing and the build and release processes in order to manage and troubleshoot all the interdependencies. SOA encourages swift development cycles and more frequent software releases, promising faster time to market, but this will not be easily achieved without the appropriate level of automation.

The Influence of the Internet

The Internet has made it possible for companies to hire programmers anywhere in the world. This “globalization” trend began with high tech companies establishing development shops in India, China and elsewhere, and then moving to a 24 x 7 development process. Corporate software development operations quickly followed

SOA encourages swift development cycles and more frequent software releases, promising “faster time to market” but this will not be easily achieved without the appropriate level of automation.



The Evolution of Build and Release Management

suit. Software contracting has become a global business with programmers working for consultancies or simply forming their own teams and bidding for business on a global basis. This trend is currently accelerating.

Another trend that is driven by the Internet is the creation of “mash-ups”. With a mash-up, software components are linked together via web services interfaces to create “virtual applications.” This trend has been enabled by SOA. Most IT users are familiar with simple application components or services such as currency exchange rates or weather predictions or mapping software. Because of web services programming standards, such capabilities can easily be linked up with newly written programming code or scripts to form useful applications.

The building of mash-ups has been stimulated by important web businesses such as Google, Yahoo!, and Salesforce.com publishing their programming interfaces and inviting anyone who is inclined to use any of the services they provide. Google, in particular, appears to be spawning a sub-industry of programmers building mash-ups that link up to Google Maps.

There are indications that the building of mash-ups is a developing trend within businesses – although there are concerns about how it can be controlled. In particular it adds further complexity to the test, build and release process, calling for effective automation and formal audit trails.

Compliance and Governance

The concerns about mash-ups are part of a larger software management issue. A number of factors have come together to raise concerns about the governance of IT. The simple fact is that software is an increasing percentage of the fabric of the business, and the volume and variety of software that support businesses continues to grow. These critical processes now have too much impact on the business to stay below the governance radar.

In addition, the number of components that need to be managed to assemble an application is also growing. We are faced with the possibility that anyone in the chain of activities who can build or maintain corporate software might seek to subvert it, or might simply make errors that would be catastrophic for the business.

Formal oversight procedures are now being considered in many organizations either in response to legislation (Sarbanes Oxley, Gramm Leach Bliley, HIPAA, etc.) or simply at the request of internal auditors who are concerned about security issues. In this context, the need for oversight of the development process is clear. In practice, this means implementing a build and release management system which provides

The simple fact is that software is an increasing percentage of the fabric of the business, and the volume and variety of software that support businesses continues to grow. These critical processes now have too much impact on the business to stay below the governance radar.

full traceability back to the source code for every component used and guaranteed reproducibility of the final artifact so that the same source creates the same executable without fail.

Impact on the Build and Release Process

The trends we have identified and discussed surfaced directly or implicitly in the survey we conducted of Rational Build Forge customers - many of whom are pioneers in the way that they deploy and use software in their businesses.

Customers interviewed reported the following:

- For reasons of compliance and governance, the development process needs greater oversight. In particular it profits from the “bill of materials” and detailed audit trail that an automated build and release process can provide.
- Software is increasingly built from multiple components created by multiple development environments, with mash-ups being an extension of this. The result is a more complicated build and release process that necessitates the need for more automation.
- Many applications are now built by global teams linked together by the Internet. This enables a “follow the sun” development strategy with teams handing off work to others on a daily basis around the globe, but it requires a more disciplined, accessible, standardized and automated development, build and release process.
- The need for faster development cycles and greater numbers of software builds within a development cycle is driven by the fact that software is increasingly a part of the products or services that a company provides. This requires a more nimble and automated development process that allows software to be assembled at a more rapid pace.

Traditionally, the build and release process for many organizations has consisted of hand-built scripts and informal, ad hoc activities. It is clear that many organizations are now finding that this is no longer an acceptable way to manage this mission critical development function.

The Rational Build Forge Survey

In the year before the IBM acquisition, Hurwitz & Associates conducted a survey of 18 BuildForge customers to understand how they were using the product, and the technical and business benefits they had experienced. In June, 2007 we were asked to revisit the survey and conduct in-depth interviews with some new Rational Build Forge customers to determine whether the original survey results are still relevant and to discover any new factors that have emerged as the industry matures.

For reasons of compliance and governance, the development process needs greater oversight. In particular it profits from the “bill of materials” and detailed audit trail that an automated build and release process can provide.



The Evolution of Build and Release Management

After conducting a new round of interviews with Rational Build Forge customers, we concluded that the motivations for buying and using Rational Build Forge remained largely the same with some notable new additions as companies mature their build and release practices. Customers continue to purchase Build Forge to improve team productivity, product quality, and accelerate software delivery through increased automation and better handoffs between roles, rapid code-build-test cycles, and better software traceability.

Hurwitz & Associates posed a detailed set of 66 questions to Rational Build Forge customers about customer environments, usage of the software, and benefits received on both business and technical fronts.

The survey results are grouped together under the following categories:

Respondent Demographics and Product Usage Patterns

Customer Satisfaction Indications and Measurements

Respondent Customer Experiences: The Technical Benefits

Respondent Customer Experiences: The Business Benefits

After conducting a new round of interviews with Rational Build Forge customers, we concluded that the motivations for buying and using Rational Build Forge remained largely the same with some notable new additions...

1. Respondent Demographics and Product Usage Patterns

The first set of charts and figures provides information on the demographics of the customers that were surveyed, and the different environments in which they have deployed Rational Build Forge.

Rigorous Support Requirements for Development Organizations

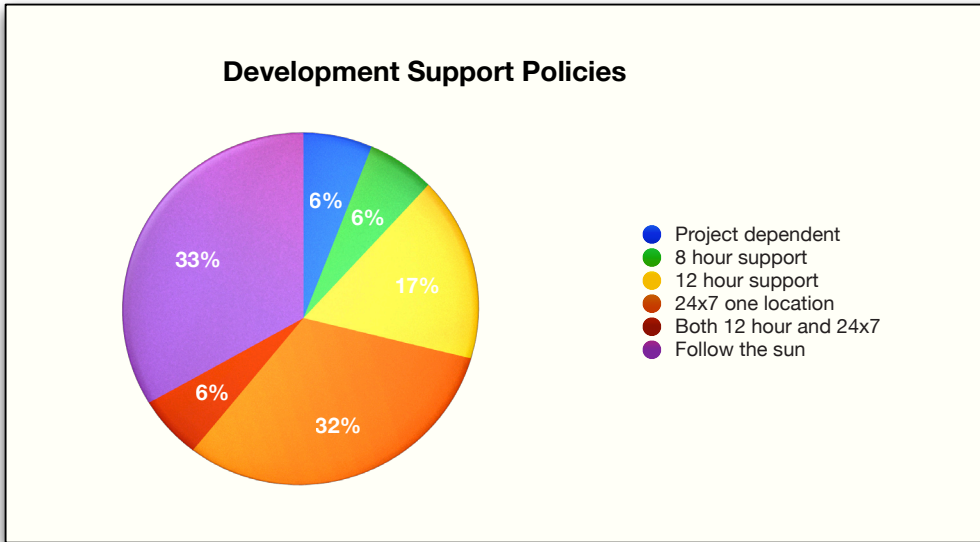


Figure 1: Development Support Policies of Respondents

Figure 1 illustrates that most respondents provide a high level of development support for customers. A total of 71% of respondents provided some level of 24 x 7 or global follow-the-sun support.

The need to deliver this level of support demands high quality and timely delivery of software, making Rational Build Forge an essential technology for the majority of respondents. It also underscores the importance of faster development cycles since cross-team handoffs are often required at the end of each group’s work day for projects to progress.

Product Benefits Achieved Across Many Industries

Figure 2 (shown on the next page) characterizes the industries represented by the respondents in the survey. While a little more than half of the respondents were software companies (i.e. Independent Software Vendors), the remaining percentage is evenly distributed across healthcare, banking/finance, telecommunications, and hardware companies, illustrating that Rational Build Forge provides benefits to software development projects across a wide variety of industries.

The need to deliver this level of support demands high quality and timely delivery of software, making Rational Build Forge an essential technology for the majority of respondents.

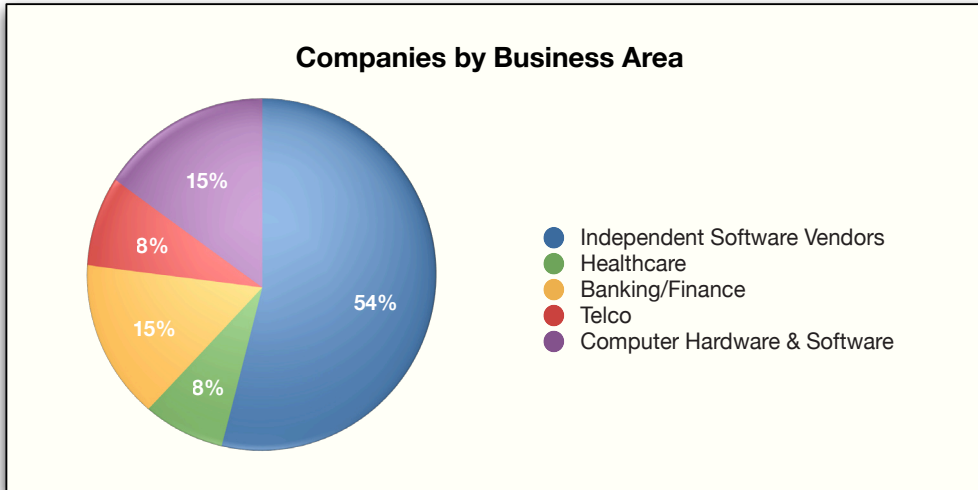


Figure 2: Type of Industries of the Respondent Organizations

In comparison with Rational Build Forge’s current customer base, the financial services and healthcare sectors are under-represented in this survey. Since the original survey was conducted, the adoption of Rational Build Forge within corporate IT Departments has grown significantly. Today, many customers across all of these industries view Build Forge as a critical part of their competitive advantage.

Improvements Achieved by Large and Small Development Teams

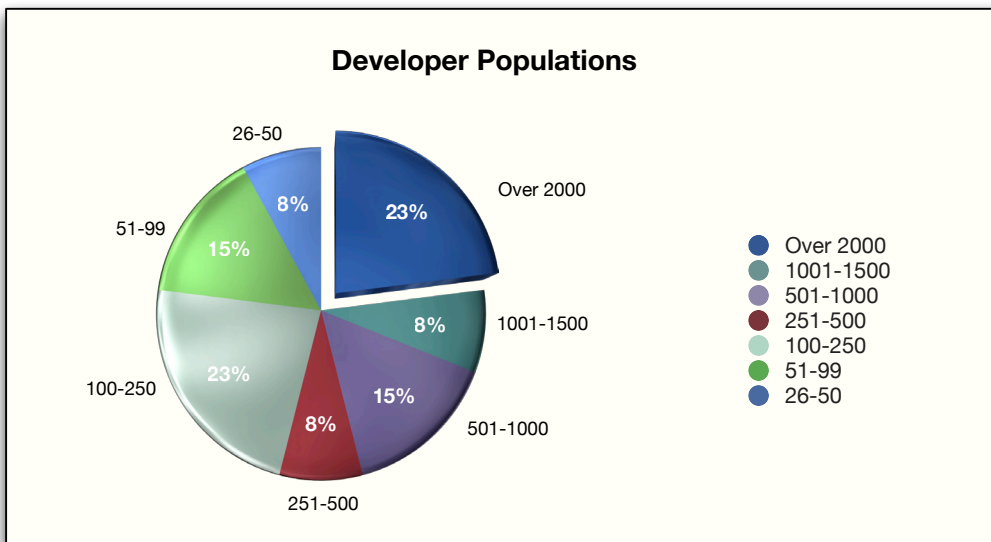


Figure 3: The Number of Developers in the Respondent Organizations

The Evolution of Build and Release Management

Figure 3 illustrates the spread in the population of developers within the respondent organizations. Rational Build Forge's build and release management solution is predictably useful for large development teams where product complexity is high and team communication is difficult. Interestingly, small teams achieved similar benefits to the large organizations in the areas of productivity and efficiency. Just under a third of all respondents reported more than 2,000 developers in their organization.

The number of developers using Rational Build Forge in the four largest companies who responded to the survey varied from 250-2000 developers, with 23% of Rational Build Forge's customer base having development teams as small as 26-99 developers.

The recent introduction of Rational Build Forge Express, IBM's build and release management offering for small to mid-sized teams, is significant because it will provide smaller teams with a cost-efficient alternative to begin automating and managing their builds and releases. There is also a migration path to Rational Build Forge Enterprise Edition, so companies can scale Build Forge and expand its use as they grow.

Support for Internal Products, Web Services, and Packaged Offerings

Survey responses indicate that Rational Build Forge is used to manage a diverse range of software projects, including packaged products shipped directly to end users, web-based applications provided as services to third parties, embedded systems, and products intended for internal use.

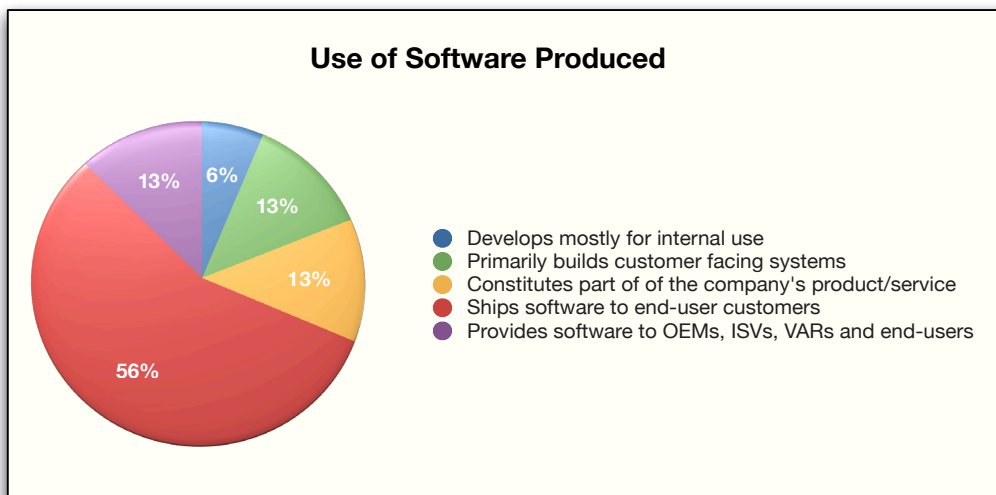


Figure 4: Types of Distribution for Respondents' Software

Figure 4 details the types of software offerings produced by survey respondents.

Rational Build Forge's build and release management solution is predictably useful for large development teams where product complexity is high and team communication is difficult. Interestingly, small teams achieved similar benefits...

Rational Build Forge Customers have Globally Distributed Development and Often Use Outsourced or Offshore Teams

The survey also gleaned meaningful results about the distributed nature of development teams. Among the survey respondents:

- None of the companies developed software from just one location.
- Only two organizations developed software entirely in the United States. 89% of respondents had development teams in Europe, the Asia Pacific region or elsewhere in the world. In fact, 22% of the companies had the majority of their developers outside the US.

Respondents reported that Rational Build Forge provides real-time access to information that is essential for distributed development teams to collaborate and work together efficiently. While it is not one of the primary benefits cited in this study, many respondents noted that Rational Build Forge assists with communication between teams and facilitates smooth team handoffs.

The majority of customers (76%) reported that Rational Build Forge met or exceeded their expectations...

2. Customer Satisfaction Indications and Measurements

Customers Express High Satisfaction with the Rational Build Forge Purchase

Based on the survey responses, customers agree that Rational Build Forge products have lived up to its promises. Figure 5 summarizes the level of customer satisfaction with Rational Build Forge. The majority of customers (76%) reported that Rational Build Forge met or exceeded their expectations, with 18% stating they were too early in their implementation to tell.



Figure 5: Overall Level of Satisfaction with Rational Build Forge Capabilities

Only one customer who had experienced a turnover of management and lost their Rational Build Forge administrator reported that the product had not met their expectations to date.

Documented Return on Investment by Rational Build Forge Customers in the First Six Months

Respondents confirmed they have received a significant return on investment from using Rational Build Forge. In general, the more experienced the respondents were with using Rational Build Forge, the greater the ROI results they reported. The largest, most experienced organizations reaped the largest ROI in terms of increased productivity and overall time savings achieved.

Table 1 shows specific customer findings across financial, productivity, and time savings, depicting the measurements that were important for each of their respective companies.

Many of the companies surveyed indicated cost reductions in excess of \$1 million annually, with one large ISV estimating their savings at \$25 million annually. ROI was typically obtained in the first three to six months of use.

Customer	Payback on Investment	Increased Productivity	Overall Time Savings	1 Year Accumulated Savings	3 Year Accumulated Savings
1	Over 200%				
2	1-20%	60-80%	Over 200%		
3		40-60%			
4	101-200%				Over 200%
5		101-200%			
6	1-20%	40-60%	20-40%		40-60%
7	20-40%	40-60%	1-20%	60-80%	

Table 1: Return on Investment Obtained by the Rational Build Forge Purchase

Respondents confirmed they have received a significant return on investment from using Rational Build Forge.

3. Respondent Customer Experiences: The Technical Benefits

A Desire for Centralization, Efficiency, and Improved Reliability

Respondents were very consistent in their objectives for their use of Rational Build Forge. Figure 6 illustrates the technical objectives that Rational Build Forge customers rated as “important” or “very important”.

Centralized build management for all projects was ranked as the most significant of these benefits -- cited by 93% of respondents. This was closely followed by the ability to increase overall efficiency to meet product schedules, the need for improved quality, and the ability to integrate end-to-end development cycles. Also rated highly was team productivity, the ability to quickly identify and diagnose errors, and increasing development speed.

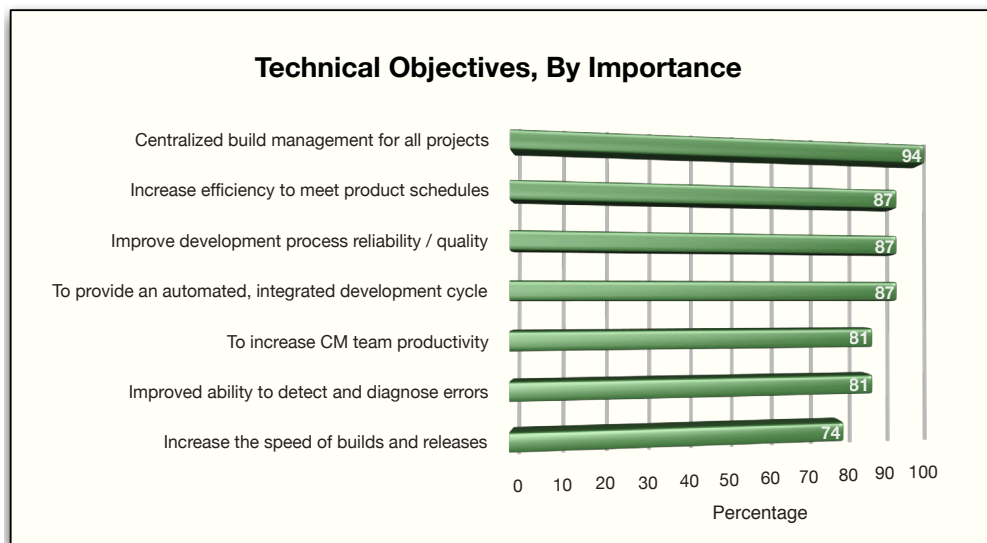


Figure 6: Objectives Ranked by Importance

When analyzed together, these objectives convey a desire for the majority of respondents to improve the efficiency and reliability of their development cycles. We infer from this data that internally developed solutions were showing signs of strain – resulting in missed production schedules and delays in delivering products and solutions to customers.

The following additional objectives were cited by the respondents as being important to their organizations:



The Evolution of Build and Release Management

- Enabling agile development (i.e., implementing continuous integration for faster development cycles).
- The ability to have more visibility into bottlenecks in the development processes.
- Establishing a common system for geographically distributed teams.
- The ability to maintain security and customize access for specific team roles.

Technical Benefits Derived With Rational Build Forge

The survey asked respondents to provide both qualitative and quantitative estimates on the impact of using Rational Build Forge within their organization. As mentioned earlier, survey respondents included new customers (who had used Rational Build Forge for less than six months) as well as more seasoned customers, and we wanted to determine if there was a significant difference in the benefits obtained between these groups.

We discovered that new customers were able to achieve results similar to long-time customers. This validates Rational Build Forge's claims that the product can be implemented and deployed in a relatively short period of time. From this data, we can infer that customers experience a rapid time to value when using Rational Build Forge. The results also suggest that benefits continue to increase over time.

We discovered that new customers were able to achieve results similar to long-time customers. This validates Rational Build Forge's claims that the product can be implemented and deployed in a relatively short period of time.

Benefit	Average Improvement	Highest Improvement
Speed of builds and releases	110%	500-2000% (or 5-20x)
CM team productivity	42%	90% or greater
Release frequency	40%	90% or greater
Error reduction	30%	70-80%
Developer productivity	28%	81-90%
Development cost savings	25%	50-70%

Table 2: Operational Benefits Received from the use of Rational Build Forge

Table 2 above provides a summary of the technical benefits derived for new and mature Rational Build Forge customers.

Dramatic Acceleration of Product Builds

Far and away the dominant benefit reported by the respondents was the ability to increase the speed of development cycles significantly. On average, the improvement was 110%, but many customers cited performance improvements of 5 to 20 times that of previous levels.

While it is not unprecedented that a fully automated capability can deliver such a level of improvement when it replaces a semi-automated capability, these results are certainly substantial.

Significant Improvements in Team Productivity, Release Frequency, Product Quality, and Cost Savings

The following six additional benefits were cited with significant improvements ranging from 20 to 40 percent over previous solutions employed by the respondents:

- **Increased CM Team Productivity:** Rational Build Forge's automation increased the productivity of the configuration management team significantly (over 40%), although several responses were 90% or higher. Respondents commented that Rational Build Forge removes the support burden of home-grown systems from the CM team and allows them to concentrate on more important tasks.
- **Release Frequency:** 40% of respondents commented that Rational Build Forge improved their ability to deliver releases more frequently, with some responses as high as 90%. The significance of this benefit will vary based on whether development teams choose to ramp up the number of releases, but it certainly indicates that teams are able to conduct more iterative code-build-test cycles as prescribed in Agile Development and Extreme Programming methodologies.
- **Overall Quality and Error Reduction in the Build Process:** These two factors relate to each other in that error reduction is a major factor in overall quality. Both factors show similar percentage improvements ranging between 30% and 40%, with responses as high as 70-80%. We can conclude that Rational Build Forge eliminates many build errors and brings to light product errors, and thus significantly improves the quality of the development process.
- **Developer Productivity:** Customers stated that Rational Build Forge made developers 30% more productive, in some cases as high as 81-90%. The product makes developers more efficient in the following ways: providing an automated way to test their code against the production environment, providing automated feedback about the build results, and rapidly directing them to the source of errors. Admittedly, this is a smaller aspect of a developer's daily activity, but even small improvements such as 10-15% can result in substantial cost savings across large development teams.

Far and away the dominant benefit reported by the respondents was the ability to increase the speed of development cycles significantly.

- **Development Cost Savings:** Respondents reported an average 26% reduction in development costs, with some responses as high as 50-70%. Cost savings were attributed to productivity gains for the CM and development teams and fewer hardware purchases due to more efficient use of hardware resources.

Replacement of Home Grown Tools Enables Greater Focus on Core Competencies

The survey revealed a trend among development organizations to abandon internally developed and open source build systems for more robust commercial build products. Most respondents had previously managed the application build and release process using proprietary build solutions or customized open source offerings that were generally created from a combination of scripts and standard utilities. The process of maintaining their in-house build and release capability was becoming onerous for many respondents, which increased the appeal of a commercial product.

Many customers commented that implementing Rational Build Forge allowed them to focus more in two areas:

- Bringing their products and services to market faster (i.e. succeeding at the company's core competency).
- Enabling CM teams to devote more time to improving efficiency and quality with less time spent on repetitive, administrative tasks.

4. Respondent Customer Experiences: The Business Benefits

For the final part of the survey, respondents were asked to consider the business rather than technical benefits of Rational Build Forge. The major business benefits derived by the respondents are identified in Figure 7 (see following page). Most respondents replied that Rational Build Forge's effectiveness significantly increased the visibility of the CM team's value within the organization, and enhanced their own understanding of their development processes. As detailed "before and after" statistics are captured by the system, teams are able to demonstrate the positive impact they have on quality and productivity to the rest of the development organization.

Reducing operational costs was the second most significant benefit that Rational Build Forge provided. The average improvement was 22 percent but varied throughout the survey replies. One mature customer reported an improvement of over 70 percent.

The remainder of the business benefits reported by respondents varied from planning and knowledge retention through to compliance and time to market. The pattern in the statistics was noteworthy since some customers reported significant benefits while others reported a lower impact. This was the case for compliance, time to market and customer experience which highlights the wide variety of companies that deploy Rational Build Forge.

The process of maintaining their in-house build and release capability was becoming onerous for many respondents, which increased the appeal of a commercial product.

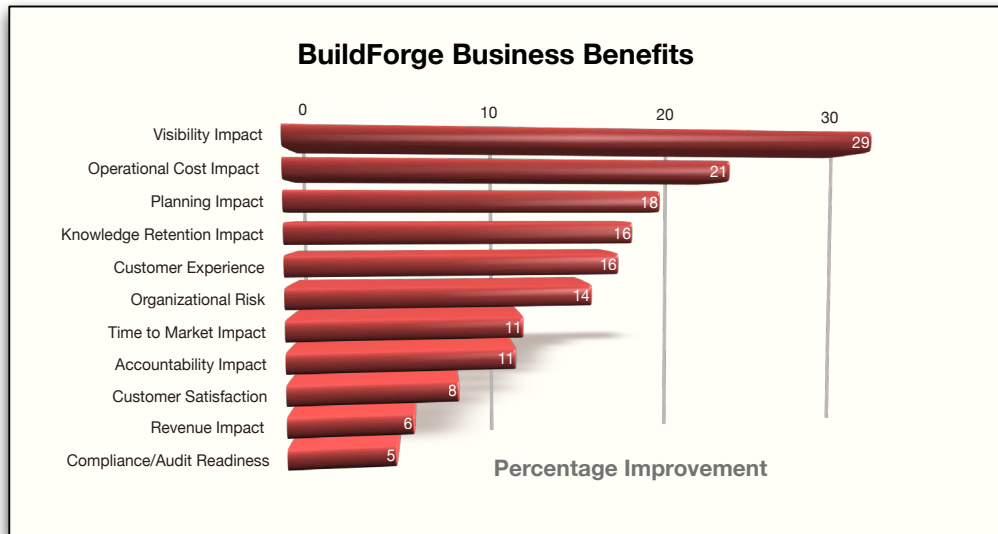


Figure 7: Top Business Benefits Rational Build Forge Respondents Reported

For example, financial services, healthcare, and other IT groups emphasize the need for compliance, while ISV’s may view it with less importance. Similarly, an ISV’s product or service is often entirely software based and thus both customer satisfaction and time to market are of critical importance.

Rational Build Forge Improves Customer Relationships and Competitive Advantage

With a large percentage of software going to customers as commercial products and customer-facing services (71%), the benefits that Rational Build Forge provides by increasing the number of builds, reducing errors, improving product quality, and reducing development time have a direct impact on the companies’ sources of revenue.

Many respondents cited exceptional improvement in their ability to maintain or improve customer relationships and gain competitive advantage as a result of using Rational Build Forge.

Now we’ll turn to some specific customer case studies to better understand how these business and technical benefits were achieved.

Case Studies

Rational Build Forge integration and process automation yields substantial productivity gains in global software company

A large, global commercial software solutions company realized it needed to change the way it developed products. The software release function was in disarray. All build processes were internally developed and were evolving haphazardly. To meet these challenges, the company established an Engineering Services Group (ESG) whose charter was to establish best practices to bring quality and efficiency to build and release processes for 50 product teams. After carefully considering different commercial products and scoping an in-house development project, the company selected Rational Build Forge because of its flexibility and scalability.

We spoke to a Senior Computer Scientist who was instrumental in the decision. He said, “We really needed to change the way we were doing things. The build processes were ad hoc. Each team was reinventing the wheel.” There was no commonality or standardization and releases were difficult to reproduce. The company realized it could provide development teams with better processes without forcing them to change their tools. With Rational Build Forge, the company could plug-in and integrate all its existing tools – including source control (Perforce), defect tracking, test automation suites, a localization framework, CD manufacturing, and other proprietary systems. After running a pilot to validate the approach, the company purchased the product.

The company’s initial experience with Rational Build Forge was positive, and consequently it was quickly and widely implemented. Within 12 months, the company had deployed the solution to all 50 product groups across 7 business units. The company has approximately 100 active projects across hundreds of servers. The ESG team is able to support hundreds of users including development, QA, project management, release engineering, and executive management without adding additional head-count. Currently 75% of the company’s development, quality, and production teams are using Rational Build Forge.

Rational Build Forge made it possible to turn point products into more creative, productive (and profitable) product suites by eliminating disjointed and inconsistent development processes. All the products can now be released synchronously using Rational Build Forge.

The company also wanted to accelerate development cycles by adopting Agile Development principles, but knew its processes wouldn’t scale. For instance, it used

“We really needed to change the way we were doing things. The build processes were ad hoc. Each team was reinventing the wheel.”

to implement various language versions of its products one at a time. Now all the translations fire off at once across a pool of servers. The company originally did 18 builds a month using four people. Now it does 360 build per month with only two people. This is a 20 fold improvement with half the staff. The Senior Computer Scientist said, “We couldn’t operate at this level of efficiency without Rational Build Forge. Software companies seem to put the least emphasis on build systems and installers. Installers are important, because the first thing the customer sees is the install process. The build process has even wider impact. It affects many other parts of a software organization, affecting quality and speed. It is a critical part of the development process, and many corporations don’t realize it.”

The team has estimated that the efficiency and quality gains provided by Rational Build Forge will save the company \$25 million dollars annually.

Rational Build Forge allowed a global insurance company to manage compliance requirements in a cost effective manner

One of largest and oldest insurance companies has grown from a U.S. centric focus to a global company over the last few decades. In addition to selling insurance, the company has expanded into retail banking. To manage this business growth, the company employs an army of about 3,000 software developers across the globe. In addition, the company outsources approximately one third of its development work to third party contractors in India and China.

The company’s adoption of Rational Build Forge was driven primarily by compliance and audit pressures. More specifically, the company’s application support director became aware of a fundamental weakness its build and release process, particularly around the development of their web applications. “We had a good source code management system, but our controls over the build process were nonexistent”, he said. “It was a huge flaw in our development process.” Although it was clear that a build management system needed to be put in place, it was unclear how a single system would meet the needs of their diverse product teams.

He investigated software build technology from several providers and chose Rational Build Forge because it enabled him to move towards a standard and flexible approach to build management. “Build Forge is a command driver that controls the build process without enforcing any particular approach to scripting”, he said. “Because of that it was possible to implement Build Forge quickly on a large number of applications.”

During the process of implementing Rational Build Forge, the team discovered that many applications had no formal build scripts at all, and in 25% of the cases

“We had a good source code management system, but our controls over the build process were nonexistent”, he said. “It was a huge flaw in our development process.”



The Evolution of Build and Release Management

developers did not understand what it took to build their applications. Build Forge became the de facto standard for all of the company's build and release management processes. Now builds can be performed with the push of a button, and all of the critical information is captured. In addition, developer training requirements have been minimized since the team can now execute builds in a consistent manner without having detailed knowledge about how the build works. "Build Forge is easy to learn and use, and it tracks and manages all the build activity across our distributed organization", the director said. "Best of all, it scales well, so we can use it around the globe."

The Build Forge project met its goal because it put "all expected controls in place and addressed all the concerns of the company's internal auditors." To date, the team has implemented 300+ applications in the Build Forge software and each generates a detailed bill-of-materials for auditing purposes.

In addition, the director discovered that there were other unanticipated benefits from using Build Forge. These included:

- A reduction of ad hoc machine purchases by developers for builds because Build Forge server pooling enables more efficient utilization of hardware.
- A reduction in lead time for builds - it went from 5 days down to 1 hour in some cases.
- Faster and more frequent builds. The teams now average 7,500 software builds a month due to better automation and continuous integration.
- An increase in productivity in the build management process.

"Rational Build Forge provides the framework where our application development really hangs together", he said. "I'm now much more confident about the software we deliver."

A financial services company replaces home-grown tool with Rational Build Forge to deliver on their service oriented architecture strategy

This financial services company started in business in the early 20th century to provide automobile insurance. Over the next several decades, the company evolved into a provider of a wide range of insurance and financial services offerings. Technology has been a key enabler of the company's growth strategy. Like many development organizations, the company is in the process of making a transition from home grown development tools to commercial tools. In addition, the company is beginning to outsource its legacy applications while keeping new software development in house. Today, about 25 percent of legacy development is outsourced.

Now builds can be performed with the push of a button, and all of the critical information is captured.



The Evolution of Build and Release Management

The IT organization supports a diverse set of applications that drive the business, including a mix of web applications, client/server agent software, and legacy applications. According to the head of development, this diversity is a departure from the past where most of the application development was mainframe batch applications that were difficult to manage, and were developed using homegrown tools. The developers of these tools had long retired and therefore it was almost impossible to maintain the software. As a result, they were deploying an enterprise modernization strategy, but needed to support distributed, web, and legacy applications with the least amount of overhead possible.

At the same time, the company needed to be much more innovative in its customer facing software due to intense competitive pressures. The company found that it needed to innovate by providing more ways for partners to add web services interfaces and for customers to be able to access services online. The company has decided to look to a Service Oriented Architecture (SOA) approach to help the company's software development process deliver faster and more customer centric results. At the start, the company selected IBM's Websphere as a development platform for SOA. It also turned to the Rational platform for configuration management, testing, and Rational Build Forge as a build and release management solution.

The decision to purchase Rational Build Forge was more than just a decision to replace home-grown tools with newer and better technology; it was a decision to focus on their core competency of delivering world class financial services to their customers instead of developing internal software. The financial advantages of this strategy were clear. IT management concluded that the company would spend over 2x the cost of purchasing Build Forge to build it in house in addition to the extensive ongoing cost to maintain it. Management also concluded that Rational Build Forge was much easier to use than their homegrown tool which would reduce training costs. Best of all, and the Build Forge system would support their web, distributed, and mainframe applications through a single build and release management console.

Beyond the cost savings and efficiencies in managing their applications, IT development management discovered several other benefits from using Build Forge software. The bill of materials provides automated tracking and auditing which allows the company to embed compliance management tasks into their everyday development operations. In addition, the team intends to explore an Agile development strategy in the near future, and the built-in integrations of Rational Build Forge with Rational ClearCase and Rational ClearQuest are viewed as a key enabler because they make it easier to execute iterative code-build-test development cycles. Overall, the IT organization has found that Rational Build Forge has helped the development organization gain more credibility with the business. According to the manager

IT management concluded that the company would spend over 2x the cost of purchasing Build Forge to build it in house in addition to the extensive ongoing cost to maintain it.

of development, “We have seen an overall higher level of satisfaction with the development organization”, he said. “Now when a request comes in we can implement changes quickly so the company can get to market faster. Our line of business counterparts have noticed the improvement.”

Conclusion

In a dramatically changing development environment, customers are increasingly looking for technology that will allow them to transform their businesses through the use of software without increasing costs and overhead. Increasingly, the applications these organizations are building are composites – developed from software components repurposed from existing applications.

Additionally software development has become a distributed or even global activity for many organizations and this a growing trend. This is being stimulated by the fact that software development itself has become an essential element in the products and services offered by many organizations. In such circumstances, the speed of software development can determine the time to market, which can have a significant impact on the company’s market presence and profitability.

Why Are Customers Choosing Rational Build Forge?

As the software development process increases both in importance and complexity, organizations naturally seek to proactively manage, automate and enhance the process. Our survey of Rational Build Forge customers validates that Rational Build Forge contributes to product quality, team productivity, and faster time to market to improve the success of software projects.

Customers reported a wide variety of benefits. For some, build speed and frequency was of the essence. Others were keen to meet governance and compliance goals. Some were eager to reduce costs through automation, while others focused more on raising quality and reducing errors. For many, the need to bring the build and release process under control was driven simply by the increase in scale of their software development activities and the fact that it had acquired a global character.

The survey provides ample evidence that the majority of Rational Build Forge customers were able to achieve all of these diverse goals and quickly cover the cost of their investment. Hurwitz and Associates believes that the scale of software development in many organizations, its importance to the business and the growing demand for formal governance of the development process will lead to widespread take-up of automated build and release technology in the next few years. Manual and poorly automated procedures will, in time, cease to be acceptable in most IT shops. We believe Rational Build Forge is breaking new ground in development automation by moving companies towards creating a highly efficient and dependable software factory.

Manual and poorly automated procedures will, in time, cease to be acceptable in most IT shops.



About Hurwitz & Associates

Hurwitz & Associates is a consulting, research and analyst firm that focuses on the customer benefits derived when advanced and emerging software technologies are used to solve business problems. The firm's research concentrates on understanding the business value of software technologies, such as Service Oriented Architecture and Web services, and how they are successfully implemented within highly distributed computing environments. Additional information on Hurwitz & Associates can be found at www.hurwitz.com.