



# Configuration Management

BEST PRACTICES

## ENABLING DEVOPS, CONTINUOUS DELIVERY AND QA WITH INCREDIBUILD

BY BOB AIELLO

DevOps best practices are driving the IT industry to new capabilities such as continuous delivery. DevOps is a set of principles and practices that help improve communication and collaboration, delighting customers with frequent delivery of features and bug fixes. Implementing DevOps can be challenging. Commonly requires best practices include:

- Continuous integration
- Continuous delivery and deployment
- Comprehensive testing to keep up with accelerated deployments
- Fast builds to keep up with demand
- Enhancing knowledge through instrumenting and fully transparent builds

## Continuous Integration is Expected

Continuous integration is a development practice that requires developers to integrate code into a shared repository several times a day. Check-ins are then verified by an automated build, allowing teams to detect problems early.

Continuous integration and automated application deployments have become standard practices that lead to high quality systems, which can be updated as often as necessary to meet and exceed customer expectations. QA, along with both manual and automated testing, have been challenged to keep up with the accelerated delivery cycles as is often seen in agile development, necessitating improved methodologies including *continuous testing* – where automated tests are built with the application from the very beginning of the lifecycle and continue through application deployment to production. These tests range from unit tests to robust API and service virtualization testing. DevOps presents many other challenges too.

## DevOps Challenges

While DevOps is driving agile development and setting a new high bar for delivering feature rich systems, these new capabilities come with many challenges. Developers must contend with

increasing complexity and the challenges of shorter development and deployment lifecycles. To succeed with DevOps you need to have the right tools to master this complexity and achieve faster development cycles. .

## Taming Complexity

Most technology professionals specialize in a few technical competencies and large systems often require the collaboration of teams of experts who have deep knowledge in each of these disciplines -- from GUI development with strong skills in user interfaces (UI) and user experience (UX) to backend service-oriented architecture (SOA). Other professionals focus on scalable infrastructures or micro services to manage the technical complexity that often must be delivered with each enterprise application. DevOps must not only bridge the gap between development and operations, but frequently DevOps must also help smooth out the communications and collaboration between technology teams even within the development organization. DBAs, for example, often have a siloed view of the data and are unaware of the impact of their changes on the application as a whole. Technical complexity allows for feature rich systems, but only if our teams understand how to support and enhance the software technology itself. Product and business users are becoming more technically savvy themselves and have come to expect instant results – especially when market demand may make the difference between the company thriving or going out of business. These demands have led many organizations to adopt agile development, which, in turn, has necessitated enhanced deployment automation. DevOps has led to improved tools and processes to support fully automated application development processes. In fact, tools are becoming first class citizens because it is just impossible to implement DevOps processes from continuous integration to continuous delivery and deployment without them. This is exactly where IncrediBuild impacts and enables DevOps transformation.

## IncrediBuild – Enabling DevOps

IncrediBuild is a comprehensive solution that makes effective use of virtualization and helps to provide the tools and processes necessary to meet the challenges of high performance computing inherent in successful DevOps. Continuous integration servers can delegate builds to available servers, but IncrediBuild takes this process a step further by distributing a single build across one or more available machines within your existing infrastructure. This approach provides much faster build cycles than any other existing approach. This capability not only applies to builds, but can also be used to accelerate other development methodologies such as code scanning and automated testing.

IncrediBuild helps to get the day-to-day DevOps work completed by providing the tools and processes necessary to accelerate and fully visualize the most complex development processes

## Enabling DevOps, Continuous Delivery and QA with IncrediBuild

from compilation and testing to code analysis file conversion, packaging or even rendering and simulation.



**FIGURE 1 - INCREDIBUILD BUILD VISUALIZATION**

This article will explain how the power of IncrediBuild’s capabilities work and how they can help your team successfully implement optimized DevOps best practices.

## Fast Builds and Virtualization

Fast builds are a must-have in today’s demanding technology environment. But speed is not the only requirement. Developers must often build, package, and deploy multiple variants of the code to meet the challenges of accelerated application delivery cycles. Time is one factor and complexity can often be an even greater challenge. IncrediBuild instruments your application build to provide detailed technical insights into your application architecture that provides knowledge and transparency into what is often the black box of proprietary compilers and complex build automation tools. With IncrediBuild, developers and build engineers have the tools to understand exactly what is going on behind the scenes when building complex applications and, more importantly, rapidly build variants of the code to accelerate continuous testing, including unit testing, code scanning, and security code vulnerability scanning. IncrediBuild does this by virtue of a workload coordinator known as a build controller, along with one or more build agents on your existing hardware using CPU cycles that previously went completely unused.

Agent	Package	Status	Work Power	CPU Speed	Avail CPU	File Cache	Logged On Users	Build Title	Logging Level	Physical Memory	Performance...	Disk Space	Build Group
Win7x32	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		1,867 MHz	91%	4096 MB	Win7x32/Workse...			Total: 3,063 MB	11.9	Total: 74.4 GB	(Default)
Xps32	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		2 Cores	2 Cores	Used: 30.14 MB				Avail: 2,241 MB		Free: 46.4 GB	
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			2,009 MHz	100%	4096 MB				Total: 1,983 MB	16.1	Total: 146.0 GB	(Default)
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			2 Cores	2 Cores	Used: 4090 MB	XP32/Workse...			Avail: 1,627 MB		Free: 136.3 GB	(Default)
Intel-tp	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		2.494 MHz	99%	4096 MB	Intel-tp/Workse...			Total: 3,070 MB	13.3	Total: 266.2 GB	(Default)
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			2 Cores	2 Cores	Used: 514 MB				Avail: 1,592 MB		Free: 239.6 GB	
Win-wiki (corruption)	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		3,502 MHz	99%	4096 MB	Win-wiki/Admins...			Total: 1,555 MB	26.1	Total: 46.0 GB	corruption
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			4 Cores	4 Cores	Used: 545 MB				Avail: 825 MB		Free: 28.2 GB	
XP32-3 (Chrome_test)	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		2,333 MHz	99%	4096 MB	XP32-3/Admins...			Total: 3,037 MB	15.0	Total: 146.5 GB	Chrome_test
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			4 Cores	4 Cores	Used: 1,218 MB				Avail: 1,318 MB		Free: 33.4 GB	
Garfield (Chrome_test)	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		2,673 MHz	99%	4096 MB			Minimal	Total: 8,183 MB	22.5	Total: 878.8 GB	Chrome_test
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			4 Cores	4 Cores	Used: 4092 MB				Avail: 7,184 MB		Free: 652.5 GB	
Host-8 (Chrome_test)	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		2,526 MHz	99%	4096 MB	HOST-8/Lumina...		Defaulted	Total: 8,120 MB	24.7	Total: 232.8 GB	Chrome_test
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			8 Cores	8 Cores	Used: 4070 MB				Avail: 6,479 MB		Free: 19.0 GB	
Host-2 (Chrome_test)	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		3,502 MHz	98%	4096 MB	HOST-2/Adminst...		Minimal	Total: 10,286 MB	37.5	Total: 465.7 GB	Chrome_test
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			8 Cores	8 Cores	Used: 4077 MB				Avail: 7,208 MB		Free: 367.8 GB	
Host-3 (Chrome_test)	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		3,391 MHz	97%	4096 MB	HOST-3/Adminst...		Minimal	Total: 8,104 MB	36.9	Total: 465.7 GB	Chrome_test
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			8 Cores	8 Cores	Used: 4076 MB				Avail: 5,503 MB		Free: 98.1 GB	
Host-4 (Chrome_test)	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		3,391 MHz	97%	4096 MB	HOST-4/Adminst...		Minimal	Total: 8,174 MB	36.6	Total: 465.7 GB	Chrome_test
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			8 Cores	8 Cores	Used: 4093 MB				Avail: 6,272 MB		Free: 362.9 GB	
Host-5 (Chrome_test)	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		3,391 MHz	98%	4096 MB	HOST-5/Adminst...		Defaulted	Total: 16,321 MB	35.5	Total: 465.7 GB	Chrome_test
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			8 Cores	8 Cores	Used: 4095 MB				Avail: 8,524 MB		Free: 305.9 GB	
PC-victor (Chrome_test)	XGE - Obsolete	Enabled		3,430 MHz	99%	64 MB	PC-victor/Victor		Minimal	Total: 32,662 MB	36.6	Total: 135.1 GB	Chrome_test
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			8 Cores	8 Cores	Larger size re...				Avail: 18,350 MB		Free: 5.2 GB	
Win-intel2013	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		3,502 MHz	98%	4096 MB				Total: 1,023 MB	25.0	Total: 100.1 GB	(Default)
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			8 Cores	8 Cores	Used: 514 MB				Avail: 729 MB		Free: 73.1 GB	
Win7x64-2	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		2,336 MHz	99%	4096 MB	win7x64-2/Admin...		Minimal	Total: 3,062 MB	13.6	Total: 233.7 GB	(Default)
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			2 Cores	2 Cores	Used: 4074 MB				Avail: 1,479 MB		Free: 145.0 GB	
Win7x64 (Chrome_test)	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...	Enabled		2,333 MHz	92%	4096 MB	Win7x64/Admins...		Minimal	Total: 4,061 MB	15.4	Total: 298.0 GB	Chrome_test
	Visual Studio C/C++, XGE - Obsolete, Nintendo 3DS...			4 Cores	4 Cores	Used: 4093 MB				Avail: 2,844 MB		Free: 74.6 GB	

FIGURE 2 – COORDINATOR MONITOR

## Putting the Continuous back into Continuous Integration

Continuous integration is a common practice in many firms, but the demand created by on-commit build requests backlogs many continuous integration servers. Developers often sit idly by waiting for their requested build to execute or, perhaps worse, some CI servers actually truncate requested builds, grouping commits from multiple developers together into one batch request. While often necessary, this defeats the whole purpose of continuous integration and can make it more difficult for developers to understand who broke the build as changes from multiple developers are bunched together in response to a CI server becoming backlogged with too many requests. IncrediBuild's distributed build capabilities can put the **continuous** back in continuous integration by providing faster builds and avoiding the backlog scenario, which can render CI servers ineffective. IncrediBuild speeds up the builds through efficient use of virtualization and provides transparency into the build processes. IncrediBuild's unique in its ability to use resources without duplicating data - no tool chains, source code or build environment needs to be setup up on the remote machines in order to use such resources. You do not even need to make sure its runs the same operating system version. This is by nature creates a major IT scalability resource for IT to accelerate its development with no additional hardware cost. While IncrediBuild can effectively use your existing machines on your local network, some organizations prefer to use a private, external, or hybrid cloud based infrastructure enabling scalability and increased performance, especially at peak time.

## Supporting Multiple Build Variants

Continuous delivery puts a huge strain on QA and testing, which requires the use of *continuous testing* in order to keep up with the demands for quality and testing. Not only are unit testing and test driven development (TDD) essential to producing clean code, but multiple variants of the code often must be built with one build created to facilitate code scanning and another to instrument the code for performance testing. Many organizations are taking automated testing

## Enabling DevOps, Continuous Delivery and QA with IncrediBuild

to a deeper level with API and service virtualization testing. These procedures, while often effective, also put a huge strain on the application build capabilities of the team. IncrediBuild allows you to support multiple complex builds and keep up with the demands by distributing the load across your existing machines. Faster builds improves productivity and quality through supporting multiple build variants in support of QA and testing.

These procedures also help to improve the in-depth understanding of the builds themselves. IncrediBuild accomplishes this by providing advanced capabilities not seen in other automated build tools. For example, process virtualization and task delivery are an essential aspect of this technology, which helps to improve understanding and management of the build process itself. Another benefit is ease of use, which comes from IncrediBuild's tight integration with other leading products.

## Integration with Other Products

Seamless integration with Integrated Development Environments (IDEs), including Visual Studio and robust source code management and workflow automation solutions such as Microsoft Team Foundation Server (TFS), and CI tools such as Jenkins, Team City, and others helps to accelerate not only your build automation but your application lifecycle workflow as well. IncrediBuild does this by integrating seamlessly with the TFS build controller while distributing the workload across multiple VMs in your existing topology.

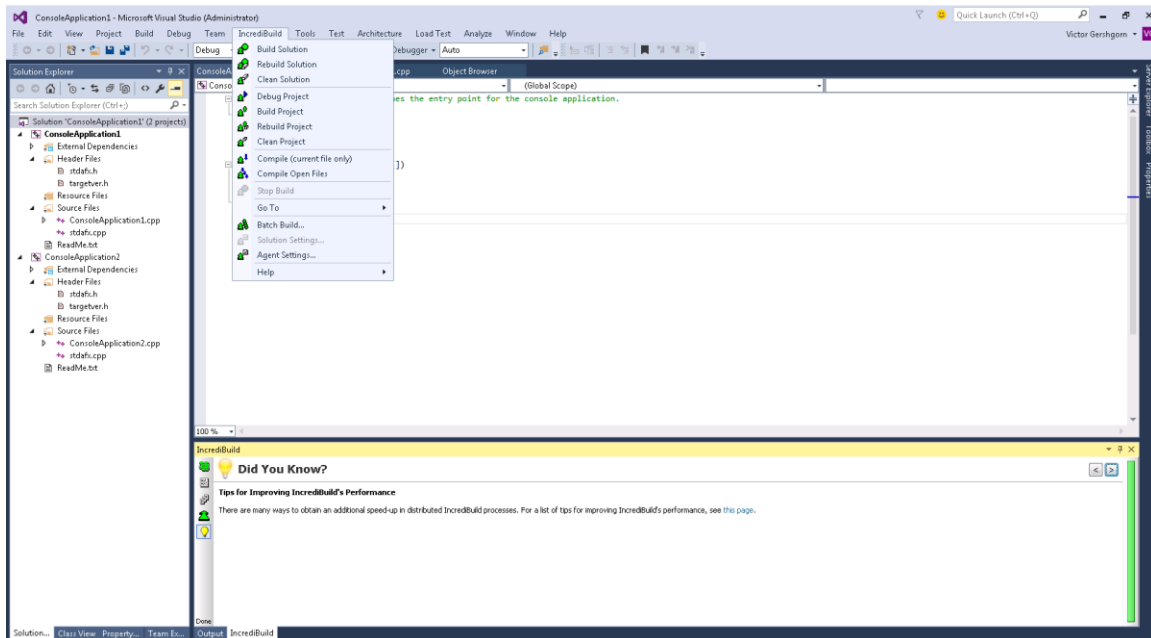


FIGURE 3 - INCREDIBUILD INTERFACE FROM WITHIN VISUAL STUDIO

IncrediBuild, now also available on Linux, accelerates build tools such as Make, MSBuild, Scon, WAF, Gmake, Jam, nAnt, Jom, BJam, and Jam+, as well as development tools, including static/dynamic code analysis (Coverity, Klocwork and others), automatic code generation tasks, code verification, code obfuscation and tools developed in-house. IncrediBuild also provides a rich command line interface allowing customers more flexibility when executing builds.

These tools provide the capabilities and processes necessary to effectively meet the challenges of DevOps continuous delivery and continuous deployment. This is particularly important in large ISVs and financial services firms and other large enterprises that must meet with industry regulatory and compliance requirements.

## Build Transparency: Regulatory Compliance

Many large firms must comply with the both international and local regulatory requirements. Many more wish to achieve the designation of being ISO 9000 compliant, which requires the establishment of a quality management system (QMS). One key aspect of meeting these regulatory requirements is the capability to maintain a separation of duties between the developer who wrote the code and the operations engineer, who builds, packages, deploys and monitors the code in production. Companies typically employ a team of dedicated build engineers to meet these industry requirements. Build engineers are often backlogged with complex application builds that they may not even fully comprehend.

IncrediBuild helps to implement DevOps best practices by empowering the build engineer with an instrumented view of exactly what is transpiring in the application build and can be used to provide a virtualized build engineer, which may meet the audit requirements of many firms. Faster builds means that your build engineers can be more productive and ensure that all applications are built to your organizations quality standards.

## Conclusion

IncrediBuild provides essential tools that help to implement fast builds that are fully traceable, audited, and well understood. This powerful virtualization technology enables your team to build multiple variants of the code, which enables your team to handle the heavy workload inherent in continuous testing. This technology also helps to keep your continuous integration servers running in a fast and effective manner – leading the way to continuous delivery and continuous deployment. DevOps best practices are essential for your team's success. Powerful tools like IncrediBuild are fundamental to ensuring that your team can meet the demands of this popular methodology successfully. The end result is a more reliable application build, package and deployment pipeline along with happier customers and a successful enterprise!

[1] Ambler, Scott W. and Lanes, Scott, Disciplined Agile Delivery: A Practitioners Guide to Agile Software Development in the Enterprise, Pearson 2012

[2] <http://agilemanifesto.org/principles.html>

[3] Humble, Jez and Farley, David, Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation, Addison-Wesley 2010

[4] Beizer, Boris, Software System Testing and Quality Assurance, International Thomson Publishing Press, 1996

[5] Aiello, Bob and Leslie Sachs. 2010. Configuration Management Best Practices: Practical Methods that Work in the Real World. Addison-Wesley Professional.

Bob Aiello is a consultant, technical editor for CM Crossroads, and the author of Configuration Management Best Practices: Practical Methods that Work in the Real World, Addison-Wesley Professional (<http://cmbestpractices.com>). Mr. Aiello has more than twenty-five years' experience as a technical manager in several top NYC financial services firms where he had company-wide responsibility for CM, often providing hands-on technical support for enterprise source code management tools, SOX/Cobit compliance, build engineering, continuous integration, and automated application deployment. Bob has served as the vice chair of the IEEE 828 Standards working group (CM Planning) and is a member of the IEEE Software and Systems Engineering Standards Committee (S2ESC) management board. Mr. Aiello holds a Masters in industrial psychology from NYU and a B.S. in computer science and math from Hofstra University. You may contact Mr. Aiello at [bob.aiello@ieee.org](mailto:bob.aiello@ieee.org), link with him at <http://www.linkedin.com/in/bobaiello>.