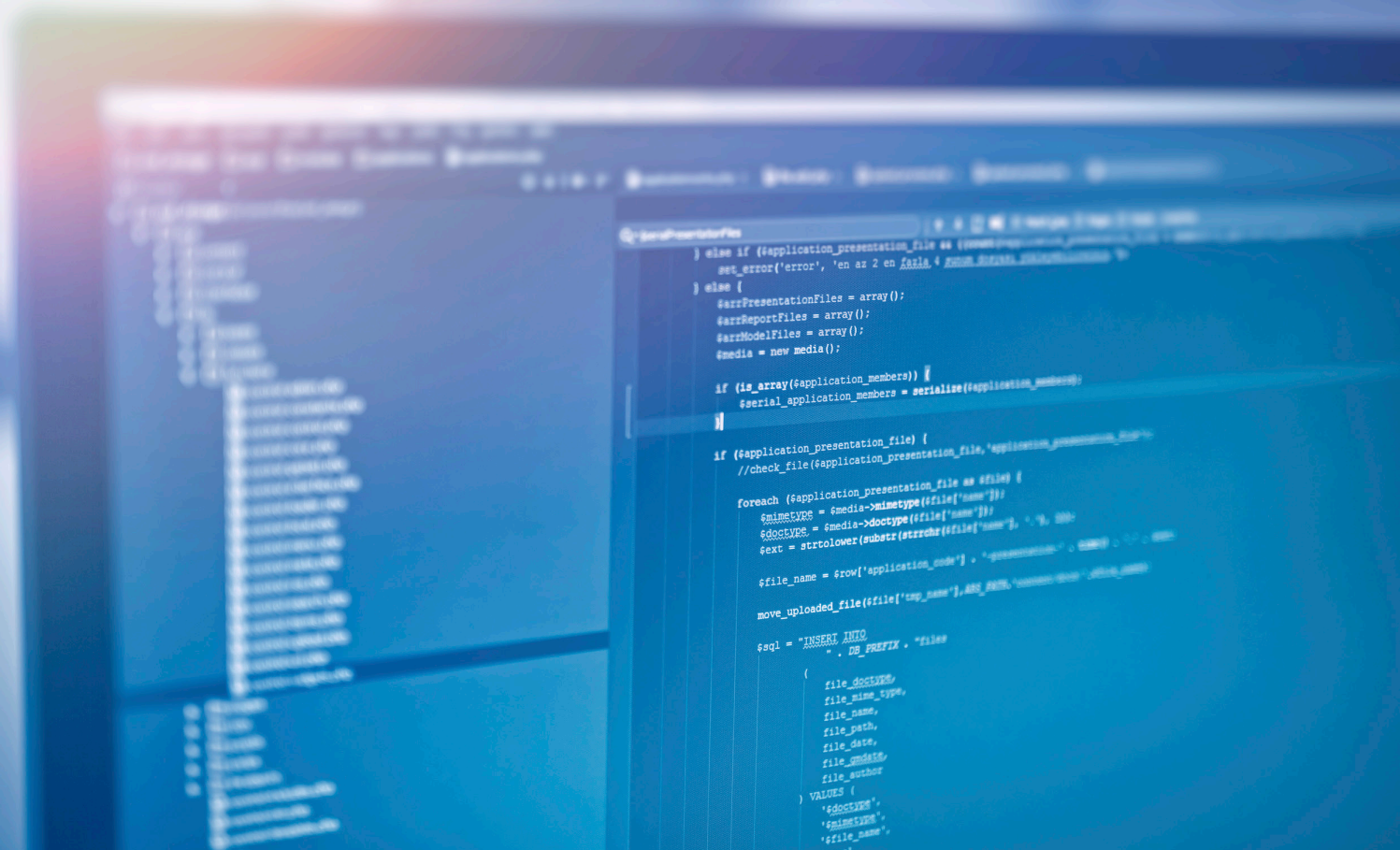


Five Critical Factors for DevOps Success in Large, Complex Enterprises



CIOs are more focused than ever on moving from project-based, Waterfall projects to continuous delivery of working software. To enable this, DevOps has become a priority.

DevOps is both a culture shift and a methodology. It's about emphasizing collaboration across the entire lifecycle of a product. Everyone – from strategy teams, to product management, definition, development, and testing, to support and operations – is involved in a continuous cycle of product delivery, learning, and improvement.

An “Assembly Line” Analogy Doesn’t Work for IT

Developing and supporting software in today's complicated business and technology environment is exceedingly complex. Every project is different, as is every product. Numerous stakeholders are involved, and business needs change mid-stream.

An “assembly line” analogy for continuous delivery comes up frequently in DevOps discussions, but it doesn't work for software. Most organizations can't turn IT – especially enterprise IT – into a software “factory” that drops releases continuously off the end of a long conveyer belt manned by business analysts, developers, and testers. Why?

- **Enterprise silos prevent collaboration.** Teams involved in developing software solutions come from multiple teams in both business and IT. These teams, however, are usually separated from each other, and they work in disparate toolsets that don't talk with one another. These groups and their input must be orchestrated despite these silos – a major challenge for every large company.
- **Compliance is non-negotiable.** Most large organizations face compliance requirements from regulatory bodies, internal policy teams, and best practice frameworks. Non-compliance can be detrimental to project and individual success. It can also come with costly consequences – heavy penalties, fines, and job loss. Compliance is recommended, if not mandatory.
- **Formal processes are necessary.** Large organizations need checks and balances through change control processes and approval gates. Often, these validations and verifications are mandatory – particularly when it comes to regulatory compliance. Whether the processes come from the compliance team, security team, or release team, they must be followed in most large companies.

Developing and supporting software in today's complicated business and technology environment is exceedingly complex. Every project is different, as is every product. Numerous stakeholders are involved, and business needs change mid-stream. Assembly lines aren't equipped to handle these factors.

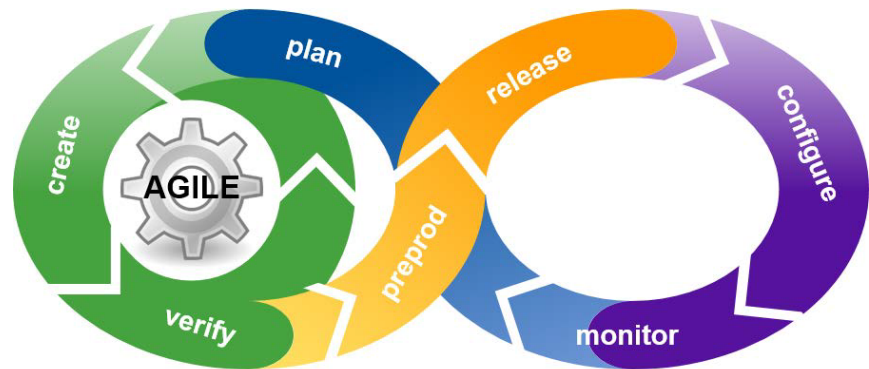
Additionally, a robust assembly line depends on an automated integration of activities. That's where efficiency and quality consistency improve. But for today's large, complex enterprises, automation and integration of people, processes and tools across a diverse, distributed ecosystem is not only challenging – it's nearly impossible.

Gartner's DevOps Framework

Gartner describes 7 activities that a DevOps Toolchain must support in an integrated way. These activities are not sequential. They're not "stages" or "phases." They exist in various forms across projects and teams. They don't reflect organizational structure.

Gartner describes the concept of DevOps through a "DevOps Toolchain," which it defines as "the tools needed to support a DevOps continuous integration, continuous deployment, and continuous release and operations initiative." A robust DevOps Toolchain enables continuous delivery.

This framework can also be used to describe the people and processes needed for a successful DevOps practice.



Gartner describes 7 activities that a DevOps Toolchain must support in an integrated way. These activities are not sequential. They're not "stages" or "phases." They exist in various forms across projects and teams. They don't reflect organizational structure.

The activities are parallel and ongoing, continuously supporting both development and operations as part of an Agile, Lean, DevOps environment. Each is likely to be supported by many tools in a typical organization, and they include:

- **Plan:** An ongoing set of activities that encompasses requirements definition, metrics development, prioritization of new and enhanced features, security planning, and release planning.
- **Create:** Includes activities associated with the creation of a code release candidate, including design, coding, builds, functional testing, and release management.
- **Verify:** Quality assurance activities, including the variety of testing types like acceptance testing, regression testing, and performance testing.
- **Preprod:** The activities that take place when a release is ready for deployment, including staging, approvals, and configuration.
- **Release:** Includes activities needed to move software into production, including release and fallback/recovery activities.
- **Configure:** An activity that occurs through the DevOps Toolchain that includes provisioning and configuration of hardware and software.
- **Monitor:** Activities focused on the health of production environments, including measurement of performance, availability, and other nonfunctional requirements, as well as monitoring of the end user experience. Feedback from

One of the most important aspects of this DevOps framework is that "monitoring" is directly linked to "planning." This connection is critical to DevOps success.

Five Critical Success Factors for Enterprise DevOps

Agile brought development teams together to collaboratively produce software and measure success as a team. DevOps takes this a step further by viewing both upstream and downstream groups as part of the team responsible for holistic, end-to-end success of a solution.

Many think of DevOps success in terms of companies, like Amazon, that can release hundreds of releases every day without disrupting business. Large organizations can't – and shouldn't – move that fast. It puts too much at risk.

DevOps "maturity" is relative and varies from organization to organization. To drive DevOps success in your organization with its unique needs, pay attention to these five critical success factors.

One: Connect the siloes culturally.

Without connecting the various teams that influence product direction and support it for your end users and customers, DevOps is not achievable. Delivery must be in sync with business objectives to make sure you're delivering the right solution. User stories, acceptance criteria, code, and tests must be in continuous alignment. Support and operations teams and activities cannot be ignored.

Agile brought development teams together to collaboratively produce software and measure success as a team. DevOps takes this a step further by viewing both upstream and downstream groups as part of the team responsible for holistic, end-to-end success of a solution. DevOps is about orchestrating these groups, the information they create, and their activities seamlessly and continuously. This will almost always require technology, but it begins with an important culture shift.

Two: Close the loop between monitoring and planning.

Planning encompasses the creation of user stories and other requirements that can come not only from new product releases, but also from an organization's service desk. Requirements to address bug fixes, performance problems, or other issues in a production application must be addressed through the DevOps cycle to enable continuous delivery.

By considering requirements more holistically, teams can continuously deliver an evolving solution blueprint to development, testing, and operations teams. This enables teams to improve on their end-to-end DevOps practices as teams learn from one another. It also enables requirements solutions to serve as the source of corporate knowledge and solution documentation.

Feeding information from monitoring activities back into the planning cycle is a critical disconnect that must be closed for successful DevOps practices.

Three: Measure success based on the entire technology ecosystem.

The complexity of today's enterprise technology environment is mind boggling. Projects are executed simultaneously that may affect many, interdependent systems. Shifting from measuring success on a project or system basis is not nearly as important as knowing that your entire collection of technology solutions support your customers' needs and meet their expectations.

DevOps means a shift from success of one feature or release to success of the system as a whole. Multi-functional teams must ensure that everything needed to ensure a product is successful in the eyes of the customer is accounted for – impacts on dependent systems, performance, measurement of success, scalability, usability, and compliance to name just a few areas of importance.

Four: Have practical DevOps goals.

Don't expect to deliver dozens of releases a day. Large organizations, like banks and healthcare companies, need discipline, checks and balances, and a bit more formality than other companies, like small software product vendors or newer organizations with no legacy technology to consider. Organizations want to accelerate delivery, but they have to manage considerable risk. What's important is that you take steps to bring about more confidence in the software your teams are producing.

Five: Leverage technology to automate and orchestrate.

Today's large enterprises need tools to support the end-to-end communication and collaboration across the organization needed for DevOps success. This won't happen without an integrated DevOps toolchain. This is a big challenge for most CIOs.

Historically, teams have chosen their own development tools, so most departments have a myriad of solutions to support strategy, business requirements, development, testing, release management, and support. There is little consistency. Tools don't talk to one another. There are overlaps and gaps in functionality.

Gartner refers to these collections of disparate, siloed development tool ecosystems as "tool islands." If organizations want to enable the continuous delivery of high-quality technology solutions, they must build bridges between these islands, because these tools must work together seamlessly.

The DevOps toolchain is as business-critical to an organization's development and operations teams as its financial software is to the Finance department and its CRM software is to Sales teams. It is the set of integrated tools that drives the assembly line of Agile, Lean, and DevOps IT to support continuous delivery of high-quality software that accommodates change.

Gartner recommends that CIOs develop and follow a DevOps toolchain strategy, so they can:

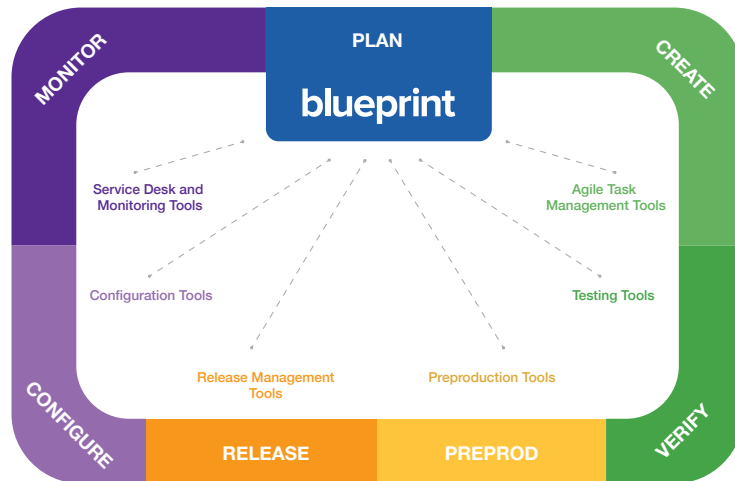
- Streamline activities by reducing manual work, eliminating redundancies, and improving process.
- Identify and automate tool integrations and handoffs.
- Make tool choice and usage decisions for their DevOps professionals.
- Understand the skills and resources needed to support the activities and tools.

Blueprint Excels as Part of a High-Performing DevOps Toolchain

An abundance of tools exist to support each set of DevOps activities. Blueprint plays a key role in business-driven planning, requirements, and metrics. It is also key in that it feeds all other activities with essential requirements needed to align the toolchain on common objectives. To facilitate this, Blueprint integrates with many other business planning, development, testing, and operations tools to keep the continuous delivery engine turning.

An abundance of tools exist to support each set of DevOps activities. Blueprint plays a key role in business-driven planning, requirements, and metrics.

Here are some key ways that Blueprint drives the DevOps Toolchain as defined by Gartner:



- Plan:** In Blueprint, teams define requirements through an integrated collection of text and visual models. They capture business priorities and metrics and align requirements to those important elements. They also establish end-to-end traceability to support coverage and change impact analysis. Blueprint's requirements definition and management capabilities are broad and deep, and second to none in the market.
- Create and Verify:** Blueprint provides the "blueprint" for development and testing by automatically generating user stories, acceptance criteria and tests from processes collaboratively defined with and by stakeholders. It delivers these artifacts through tool integrations to the teams' tools of choice. As requirements change, Blueprint keeps teams up-to-date with a clear, comprehensive understanding.
- Preprod, Release, and Configure:** Blueprint's Nonfunctional Requirements Accelerator helps teams define requirements that facilitate the transition from preprod to release. The Accelerator provides a set of reusable nonfunctional requirement categories and questions teams can ask to ensure solutions are well-positioned for sound operations, maintainability, and ongoing support.
- Monitor:** Blueprint's integration with other DevOps tools enables it to collect metrics and support information that feeds back into the next release's planning cycle. Blueprint helps teams bring these activities together in one place, so teams are prepared with the information they need to deliver the next software release. In addition metrics from the Monitoring activity are weighed against original business objectives recorded in Blueprint to show whether the business value originally envisioned was achieved.

Blueprint is a key enabler of continuous delivery as described by Gartner's DevOps toolchain. For more information on how we support the move to continuous delivery with an Agile, Lean, and DevOps focus, please [contact us today](#).

blueprint®

90 Eglinton Ave E #700
Toronto, ON M4P 2Y3, Canada
1.866.979.BLUE(2583)
www.blueprintsys.com

Blueprint provides industry-leading solutions that accelerate and de-risk the digital transformation of large organizations. With our products - Blueprint Storyteller for Agile, Blueprint Automate for DevOps and Blueprint RegTech for Compliance - organizations receive greater business value from IT, faster and more frequently, while dramatically increasing the efficiency and confidence of compliance.

© 2017 Blueprint Software, Inc. All rights reserved. All product and company names and marks mentioned in this document are the property of their respective owners.