Assessing DevOps Maturity
Using a Quadrant Model

Contents

Businesses that rely on software to deliver new products and services must provide customers with the latest technology enhancements and an engaging user experience.

To meet this need, many companies are aggressively investigating and implementing modern software development and delivery methods such as:

» **Agile:** Promotes adaptive planning, evolutionary development, early delivery and continuous improvement, as well as rapid, flexible response to change.

» **Continuous integration (CI):** Automatically builds and validates changes immediately, allowing developers to identify errors early and fix them more quickly.

» **Continuous delivery (CD):** Automatically and continuously builds, tests and deploys software to ensure it can be released at any time.

When used in combination, these practices create the foundation of a culture known as DevOps. This culture supports faster, more frequent releases of higher quality software applications, by focusing on automation and cross-functional collaboration. In addition to releasing higher quality applications more quickly, companies that embrace DevOps can increase market share and deliver innovative, or even market-disrupting, solutions. Due to the proven business and technology benefits of DevOps, it is becoming the de facto software development methodology.

For many IT professionals,the number of tools, technologies and practices which are associated with DevOps can be overwhelming. This alphabet soup of terms like TDD, BDD, IaC, PaaS, SaaS and CI/CD can cause confusion and analysis paralysis, preventing effective adoption of DevOps. In some cases, this confusion has led to misdirected DevOps initiatives which end in costly failures. To confidently address these challenges, companies need to simplify their approach to DevOps. They need a new model that helps them understand where they are, where they want to go and how they can best get there. This paper offers a DevOps maturity assessment model comprised of four quadrants. IT leaders can use the model to do the following:

» Assess their current DevOps maturity.

» Develop and manage a plan to implement DevOps.

» Create strategies to address the organizational, technological and cultural obstacles of embracing a DevOps initiative.

Assessing Development Practices

The DevOps Quadrant Maturity Model is derived from the real-world transformations of Fortune 500 organizations in a variety of industries. This model helps companies assess the state of their current development practices, where they need to improve and what steps must be taken to reach their goals. To frame the space representing non-DevOps and DevOps organizations, the model establishes two axes: the phases of the software development lifecycle on the horizontal x-axis, and the levels of adoption of various DevOps practices on the vertical y-axis (see Figure 1).
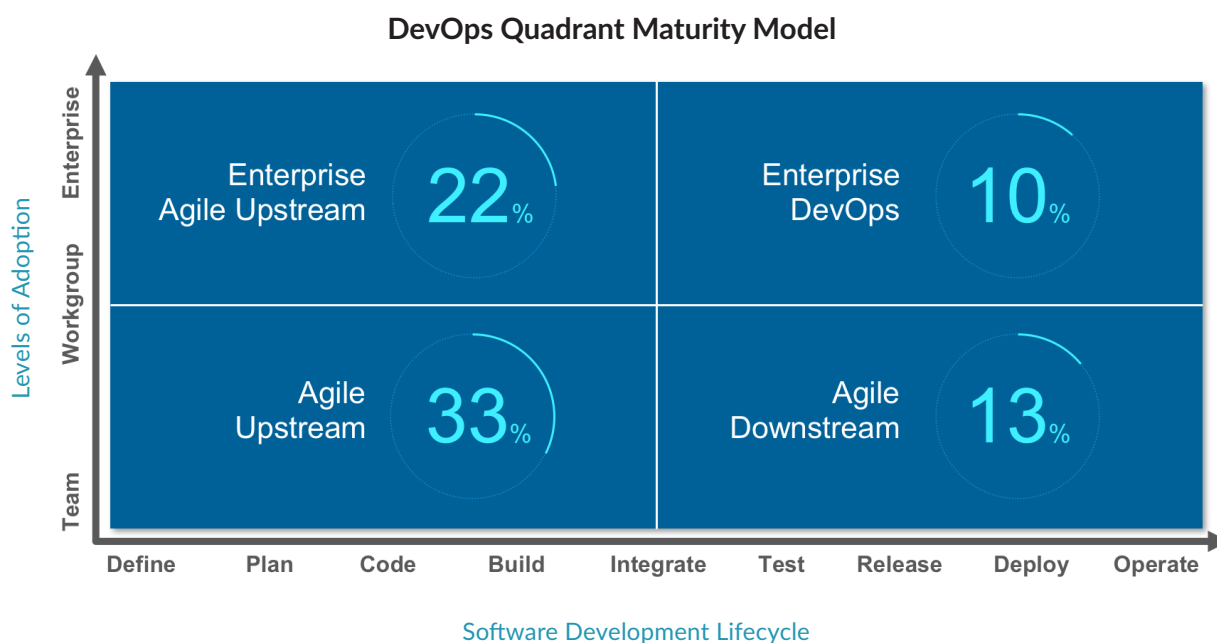
DevOps Quadrant Maturity Model

Figure 1: Two axes of DevOps adoption

53%
of companies are currently investigating or planning a DevOps journey

Source: CloudBees® webinar, *Introduction to the Four Quadrants of DevOps Maturity*, 2016

The x-axis includes the phases of a standard software development lifecycle: define, plan, code, build, integrate, test, release, deploy and operate. Every company practices some form of these phases. The model divides these phases into two groups based on traditional organizational and cultural boundaries and observation of industry practices. Upstream represents the project planning and execution phases, and downstream represents the quality, delivery and operation phases. For firms that have begun applying agile practices and principles to the development lifecycle, the two groups are identified as:

1. **Agile upstream**, comprised of practices such as Scrum, iterative project planning and execution, and continuous integration

2. **Agile downstream**, comprised of practices such as automated testing, dynamic provisioning, automated deployment and continuous delivery.

According to research from Forrester, an IT research and consulting firm, 33 percent of companies practice agile upstream, while only 13 percent have embraced agile downstream.[1] That means that 67 percent of upstream activities and 87 percent of downstream practices still rely on some form of waterfall development, with serial phases for each step of the software development life cycle. Using this non-agile methodology, organizations experience lengthier time to market and slower response to market demand and customer needs. They also compromise innovation and overall competitiveness.

The y-axis represents the levels of adoption for agile practices, which may be deployed at the team, workgroup or enterprise level. According to Forrester, at the team level only 22 percent of organizations have adopted agile upstream practices, and a mere 13 percent have embraced agile downstream. The adoption of enterprise DevOps in the downstream is quite rare, with a mere 10 percent of companies practicing agile methodologies across the development life cycle at an enterprise level.

For the more than 90 percent of organizations that are not consistently taking the enterprise DevOps approach, there is a real downside. They cannot share development tools. They lack shared best practices, which inhibits their ability to measure and understand development practices and efficiency, implement consistent security and meet compliance mandates..

28%
of companies are using continuous delivery on all projects

Source: *Continuous Delivery: The New Normal for Software Development*, by Evan Research, 2015

---

[1] *How Can You Scale Your Agile Adoption?* by Diego Lo Giudice, vice president and principal analyst, Forrester Research, February 5, 2014.

Exploring the DevOps Quadrant Maturity Model

The DevOps Quadrant Maturity Model includes four quadrants that position development practices along the two axes of DevOps adoption (see Figure 2).
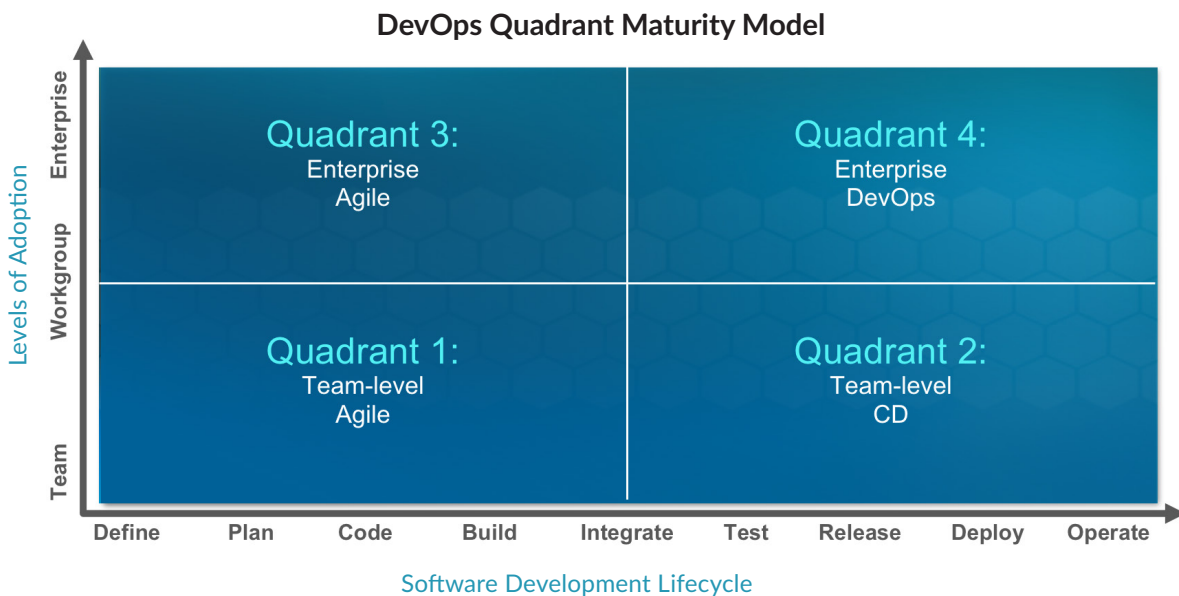
DevOps Quadrant Maturity Model

Figure 2: DevOps Maturity Model

The following table explains the characteristics of each quadrant, both in terms of positive attributes and areas that require improvement so that organizations can achieve enterprise DevOps.

QUADRANT CHARACTERISTICS

	Positive Characteristics	Areas for Improvement
Quadrant 1: (Upstream) Team-level agile	» Agile development practices adopted by one or just a few teams » Continuous integration implemented as a team-level practice	» Team-specific tools integration, disconnected from enterprise and downstream tools team level » Team-specific project management and development processes » Often identified by the existence of waterfall-Scrum methodology

|	Positive Characteristics	Areas for Improvement
Quadrant 2: (Downstream) Team-level CD	» Support for test automation, dynamically provisioned environments, configuration management and container usage on specific projects » Team-level CI extended to CD » Team-level integration of development tools with delivery tools	» Project management and development practices aligned only with operations as team-specific one-offs, not standardized practices » Lack of standard, shared solutions for tools and infrastructure management and development processes; operate as a team-specific, one-off project
Quadrant 3: (Upstream) Enterprise agile	» Standard, enterprise-wide agile development practices » Business aligned with agile specifications » Adoption of common project management and development processes	» Development tools limited to team level, to support delivery process » Tools integrated only at team level
Quadrant 4: (Downstream) Enterprise DevOps	» Support for standard tools and processes for automated test, dynamically provisioned environments, configuration management and containers » Business and customers aligned with agile specifications » Tools integrated to a common platform supporting reporting and metrics, security, governance and traceability » Culture of collaboration, learning and innovation	

Which Quadrant Are You In?
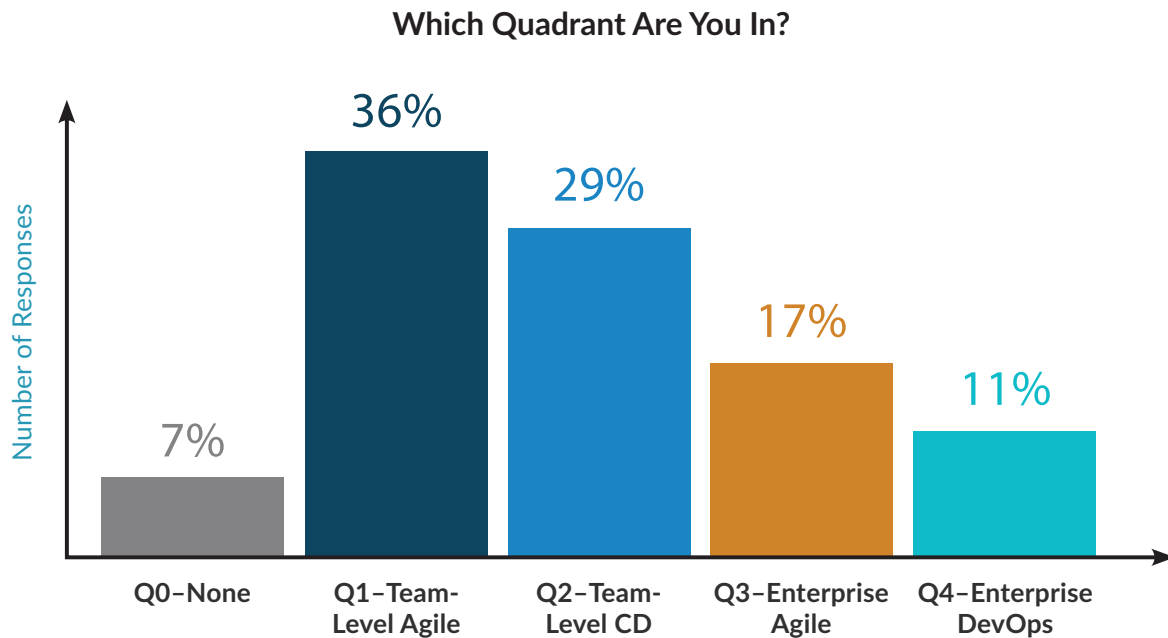
Figure 3: Results from a poll conducted during a CloudBees Webinar, May 2016

CHASMS PREVENTING THE ADOPTION OF ENTERPRISE DEVOPS

What stops organizations from embracing enterprise DevOps? Conflicting priorities in the approach to upstream and downstream agile practices impact the DevOps trinity: people and culture, process and practices, and tools and technology. Resolving these conflicts is necessary for organizations to cross the DevOps chasm, moving either from quadrants one or three to quadrants two or four.
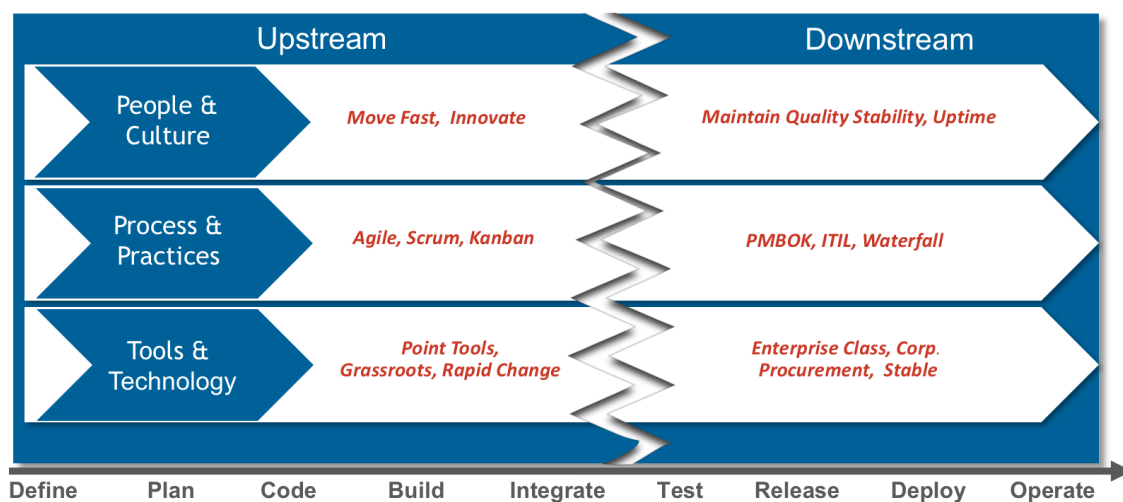

Figure 4: Crossing the DevOps chasm

Reviewing Adoption Patterns

In observing clients and other companies, we see several common adoption patterns for DevOps practices and technologies. Applying these four patterns to the maturity model provides a foundation for discussing a company's DevOps journey.

ADOPTION PATTERN 1

Driven by the explosive adoption of agile, for years the first adoption pattern has been the path most commonly taken to a DevOps transformation. Often, an organization begins in quadrant one, with team-level agile at the upstream stage (planning, project management, execution) (see Figure 5). Recognizing the benefits across the organization, they expand into working groups and then implement enterprise agile in the upstream as step 2. In step 3, enterprise DevOps, the organization aligns on a DevOps strategy, with teams using common CI and CD processes and tools in the downstream (quality release and delivery).
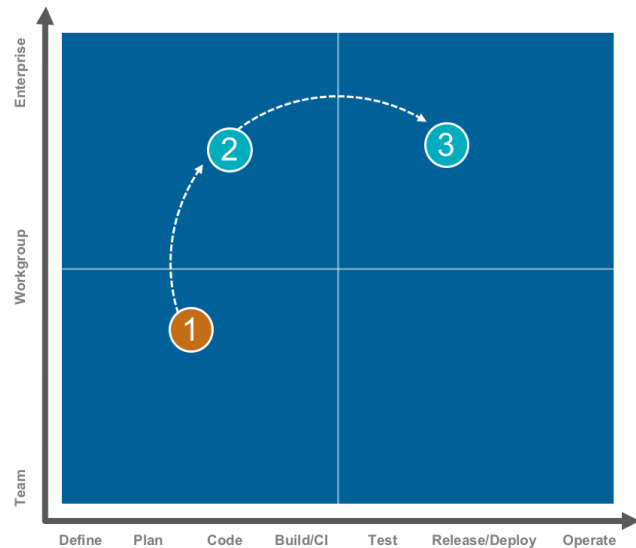
Figure 5: Adoption pattern 1

ADOPTION PATTERN 2

The second pattern is similar to the first, but here the movement toward DevOps modernization begins within the development team embracing agile practices through the adoption of CI. This is a common starting point because development teams are able to adopt CI processes and tools independent of other stakeholders (see Figure 6). As CI speeds up the coding and building steps, teams seek to increase flow by adopting agile definition and planning practices. These organizations then follow the path of adoption pattern 1.
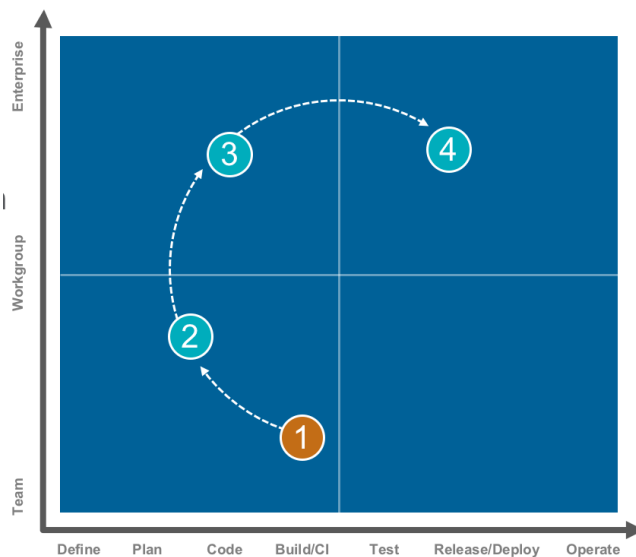
Figure 6: Adoption pattern 2

ADOPTION PATTERN 3

The third adoption pattern is the most common, with one or more teams adopting upstream agile planning and project management first (see Figure 7). The second step is where teams extend CI to CD, connecting downstream. Recognizing the value of these steps, the enterprise embraces the upstream in step 3. The company then scales those practices from a few working groups to the entire enterprise, aligning on a DevOps strategy and using common CI/CD processes and tools in step 4.

ADOPTION PATTERN 4

The fourth adoption pattern shows an emerging trend, where the drive toward modernization comes from the right side of the quadrant (see Figure 8). For example, operations may want to improve or modernize practices. In the first step, enterprise DevOps, a focus on infrastructure drives the operations team to implement cloud technology, creating an elastic infrastructure. Seeking faster delivery through the downstream, or CD in step 2, the company begins to drive automation through the infrastructure with code. In step 3, teams may implement CD as a Service, enhancing repeatability and reliability with the infrastructure. Step 4, enterprise agile, shows the organization aligning on a DevOps strategy using common agile and CI/CD processes and tools.
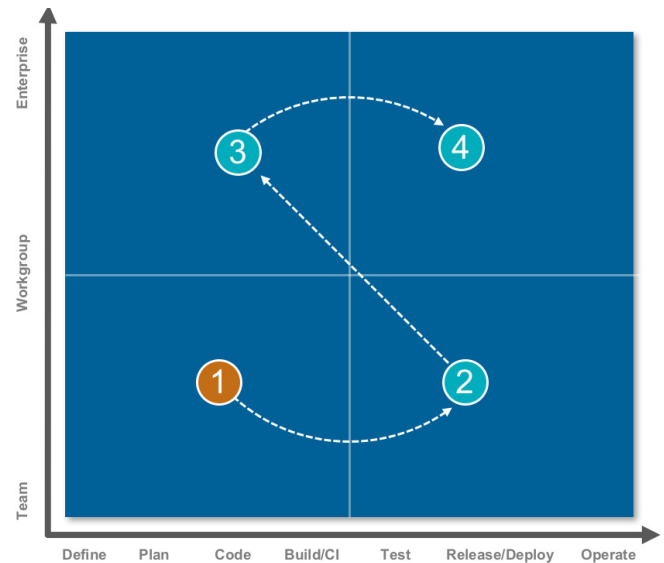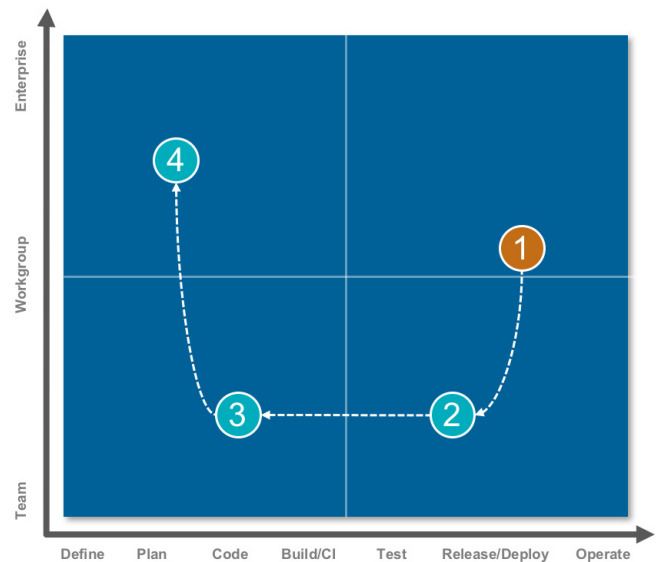

Figure 7: Adoption pattern 1

Figure 8: Adoption pattern 2

Getting Started with a DevOps Maturity Assessment

The DevOps Quadrant Maturity Model strives to simplify a complex problem and provide IT professionals with a structure for assessing DevOps maturity. Obviously, companies need to take additional steps, potentially using the guidance of expert consultants and vendors, to move forward in their strategies.

Here are some tips to help companies launch their DevOps initiative:

» **Be pragmatic, not dogmatic:** As with agile tenets and principles, acknowledge that not everything can be known from the start. Approach and plan the initiative using existing knowledge, but be prepared to apply lessons that adjust, refine and tune your journey along the way.

» **Transform incrementally:** Treat the process as an evolution, not a revolution. Aim for gradual progress.

» **Start with a Go Team:** Pick a project that has low risk and high value. Assign a team to it with the right talent to drive change and innovation in that product's development processes.

» **Show value:** Communicate the success of modern development practices widely and frequently to upper management and other teams. People will see the benefits and develop an interest in adopting these practices in their own teams.

LEARN MORE

For more information on successfully navigating the journey to enterprise DevOps, visit CloudBees at www.cloudbees.com or follow us on Twitter (@CloudBees), LinkedIn, Facebook and Google+.