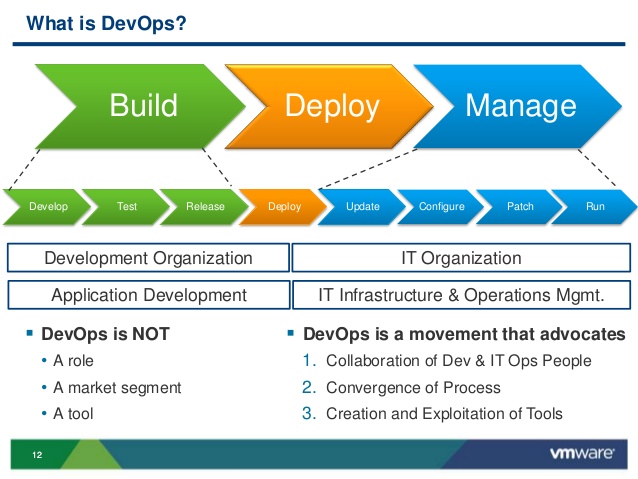
**What is DevOps?**

**DevOps** (combination of "software **DEV**elopment" and "information technology **OP**eration**S**") is a term used to refer to a set of practices that emphasize the collaboration and communication of both software developers and information technology (IT) professionals while automating the process of software delivery and infrastructure changes. It aims at establishing a culture and environment where building, testing, and releasing software can happen rapidly, frequently, and more reliably.



DevOps is a cultural shift and collaboration (between development, operations and testing), there is no single "DevOps tool": it is rather a set, consisting of multiple tools.[[](https://en.wikipedia.org/wiki/DevOps#cite_note-12) Generally, DevOps tools fit into one or more of these categories, which is reflective of key aspects of the software development and delivery process:[[13]](https://en.wikipedia.org/wiki/DevOps#cite_note-13)[[14]](https://en.wikipedia.org/wiki/DevOps#cite_note-14)

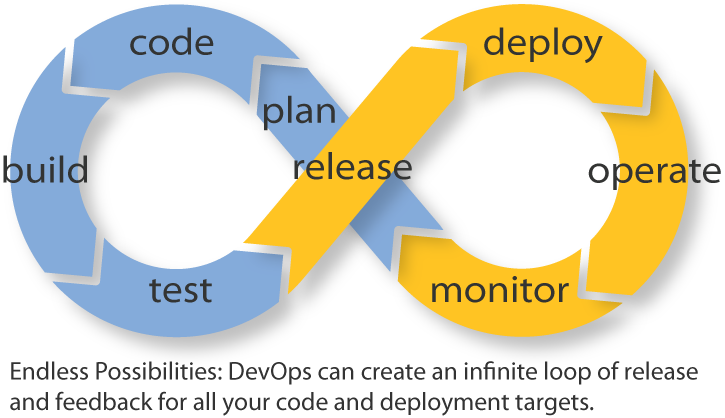
1. Code — Code development and review, version control tools, code merging;
2. Build — Continuous integration tools, build status;
3. Test — Test and results determine performance;
4. Package — Artifact repository, application pre-deployment staging;
5. Release — Change management, release approvals, release automation;
6. Configure — Infrastructure configuration and management, Infrastructure as Code tools;
7. Monitor — Applications performance monitoring, end–user experience.

Though there are many tools available, certain categories of them are essential in the DevOps toolchain setup for use in an organization. Some attempts to identify those basic tools can be found in the existing literature.

Tools such as [Docker](https://en.wikipedia.org/wiki/Docker_%28software%29) (containerization), Jenkins (continuous integration), Puppet (Infrastructure as Code) and Vagrant (virtualization platform)—among many others—are often used and frequently referenced in DevOps tooling discussions

**Relationship to agile and continuous delivery**

Agile and DevOps are similar, but, while agile software development represents a change in thinking and practice (that should lead to organizational change), DevOps places more emphasis on implementing organizational change to achieve its goals



### Continuous delivery

Continuous delivery and DevOps are similar in their meanings (and are, often, conflated), but they are two different concepts:

* DevOps has a broader scope, and centers around:
  + Organizational change: specifically, to support greater collaboration between the various types of worker involved in software delivery:
    - Developers;
    - Operations;
    - Quality Assurance;
    - Management;
    - System Administration;
    - Database Administration;
    - Coordinators
  + Automating the processes in software delivery.
* Continuous delivery, on the other hand, is an approach to automate the delivery aspect, and focuses on:
  + Bringing together different processes;
  + Executing them more quickly and more frequently.

## Goals

The specific goals of DevOps span the entire delivery pipeline. They include improved deployment frequency, which can lead to:

* Faster time to market;
* Lower failure rate of new releases;
* Shortened lead time between fixes;
* Faster mean time to recovery (in the event of a new release crashing or otherwise disabling the current system).

### Benefits of DevOps

Companies that practice DevOps have reported significant benefits, including: significantly shorter time to market, improved customer satisfaction, better product quality, more reliable releases, improved productivity and efficiency, and the increased ability to build the right product by fast experimentation.

However, a study released in January 2017 by F5 of almost 2,200 IT executives and industry professionals found that only one in five surveyed think DevOps had a strategic impact on their organization despite rise in usage. The same study found that only 17% identified DevOps as key, well below software as a service (42%), big data (41%) and public cloud infrastructure as a service (39%).