

Top 10 Popular Open Source DEVOPS Tools

Author, Balakrishnan Subramanian

A Data Science Foundation White Paper

February 2020

www.datascience.foundation

Data Science Foundation

Data Science Foundation, Atlantic Business Centre, Atlantic Street, Altrincham, WA14 5NQ
Tel: 0161 926 3641 Email: admin@datascience.foundation Web: www.datascience.foundation
Registered in England and Wales 4th June 2015, Registered Number 9624670

1. OVERVIEW

First coined around 2009, the term [DevOps](#) refers to an approach to IT that emphasizes collaboration between the development and operations groups. It arose out of the agile software development movement and applies some of the same principles to the Application Lifecycle Management (ALM) process. DevOps is difficult to define because it's more of a movement or a philosophy than a rigid set of rules or practices. Open source DevOps tools are used as a way to streamline the process of development and deployment.

2. INTRODUCTION

1. Stages of DevOps lifecycle

The DevOps tools can be categorized in five groups depending on its purpose in the particular stage of DevOps lifecycle:

- Continuous Integration
- Configuration Management
- Continuous Inspection
- Containerization
- Virtualization

3. TOP 10 OPEN SOURCE DEVOPS TOOLS

1. Ansible

1. Description

This open source tool provides software-related services like application deployment, configuration management etc.

2. Key features

- It provides agentless architecture.
- It is powerful because of workflow orchestration.
- It is simple and easy to use.

2. Chef

1. Description

This tool is used for checking the configurations which are applied everywhere and also helps in automating the infrastructure.

2. Key features

- It ensures that your configuration policies will remain flexible, versionable, testable and readable.
- It helps in standardizing and continuously enforce the configurations.
- It automates the whole process of ensuring that all systems are correctly configured.

3. Docker

1. Description

Docker's, on the other hand, uses the concept of Containers which virtualizes the Operating System. It can be used to package the application (for example, WAR file) along with the dependencies to be used for deploying in different environments.

2. Key features

- The portability is made possible via its special containerization technology, quite often found in self-contained units.
- It packages everything than an application is required to run: libraries, system tools, runtime, etc.

4. ELK Stack

1. Description

The Elastic Stack—more commonly known as ELK Stack—combines Elasticsearch, Logstash and Kibana. Elasticsearch is a modern search and analytics engine based on Apache Lucene, while Logstash provides data processing and enrichment. Kibana offers logs discovery and visualization.

2. Key features

- It Supports Server Platforms like All OS-x, Window and Red hat.
- Basic Components Involved - Search and storage.

5. Git

1. Description

It is a free and open source tool and supports most of the version control features of check-in, commits, branches, merging, labels, push and pull to/from GitHub etc.

2. Key features

- It is pretty easy to learn and maintain for teams initially looking at a tool to version control their artifacts.

6. Jenkins

1. Description

Jenkins is an automation server. It is an open source tool and written in Java. It helps many projects in automating, building and deploying.

2. Key features

- It helps in distributing the work on multiple machines and platforms.
- It acts as a continuous delivery hub for the projects.
- Supported operating systems are Windows, Mac OS X, and UNIX.

7. Nagios

1. Description

It is an open source tool. This tool is written in C language. It is used for network monitoring, server monitoring, and applications monitoring.

2. Key features

- It helps in monitoring Windows, Linux, UNIX, and Web applications.
- It provides two methods namely agent-based and agentless for Server monitoring.
- While monitoring the network, it checks network connections, routers, switches, and other required things also.

8. Puppet Enterprise

1. Description

It is an open source tool. While developing the software this tool will ensure that all the configurations are applied everywhere. Basically, it is a configuration management tool.

2. Key features

- It can work for hybrid infrastructure and applications.
- It provides Client-server architecture.
- It supports Windows, Linux, and UNIX operating systems.

9. Selenium

1. Description

Selenium is a free open source automated functional testing tool to test web applications. It is normally installed as a Firefox browser plugin and helps to record and playback of test scenarios.

2. Key features

- It is very easy to learn.

10. Splunk

1. Description

Splunk is a software platform which converts the machine data into valuable

information. For this, it gathers the data from different machines, websites etc. Splunk is headquartered in San Francisco.

2. Key features

- Splunk Enterprise will help you in aggregating, analyzing and finding answers from your own machine data.
- Splunk Light provides features for small IT environments.
- With the help of Splunk Cloud, Splunk can be deployed and managed as a service.

4. PROS AND CONS OF TOOLS

S.No	Tools	Pros	Cons
1.	Ansible	<ul style="list-style-type: none"> - Agentless – does not require any tools except SSH and Python on the target machine (may use other connection protocols for managing network devices) - Relatively easy to learn - Python-based and can be extended easily - Push based and imperative – the user can control the order of execution of different tasks which relates more to the usual way of doing things then description-based approach. 	<ul style="list-style-type: none"> - Since it is push-based it is more difficult to manage large sets of machines. - The web console is a community add-on – there’s a commercial version made by RedHat but it’s not free. - Missing modules for compliance/security. - Windows support is not on par with Linux/Unix support.
2.	Chef	<ul style="list-style-type: none"> - The most flexible solution - Ruby-based Domain Specific Language - Has an excellent ecosystem with tools for automation, cloud monitoring, compliance and security management. - Good for managing large-scale environments. 	<ul style="list-style-type: none"> - Agent-based. - The full ecosystem is insanely expensive - Ruby-based DSL requires a steeper learning curve for people not fluent in Ruby. - Pull-based/descriptive, so no control of the order of execution of task.
3.	Docker	<ul style="list-style-type: none"> - Return on Investment and Cost Savings. - Rapid Deployment - Security - Simplicity and Faster Configurations. 	<ul style="list-style-type: none"> - Missing features - Data in the container - Run applications as fast as a bare-metal serve - Provide cross-platform compatibility.

- | | | | |
|----|-----------|--|--|
| 4. | ELK Stack | <ul style="list-style-type: none"> - Robust solution - Variety of plugins - Logstash allows you to create customized log processing pipeline - Incredible Kibana visualizations. - Data redundancy and replication - High availability - Only one .git directory per repository - Superior disk utilization and network performance - Collaboration friendly. - It comes with a wide range of plugins - Jenkins integrates and works with all major tools like CVS, Subversion, Git, build tools like Apache Ant and Maven and even other tools like Kubernetes and Docker. - Jenkins comes with a decent API suite. | <ul style="list-style-type: none"> - Steep learning curve - Kibana has no default “logging” dashboards - Requires intensive management - Authentication and Alerting are paid features. |
| 5. | Git | <ul style="list-style-type: none"> - Git is less preferred for handling extremely large files or frequently changing binary files. - GIT does not support ‘commits’ across multiple branches or tags. | <ul style="list-style-type: none"> - Git is less preferred for handling extremely large files or frequently changing binary files. - GIT does not support ‘commits’ across multiple branches or tags. |
| 6. | Jenkins | <ul style="list-style-type: none"> - Unpredictable costs - Lack of governance - Lack of analytics | <ul style="list-style-type: none"> - Unpredictable costs - Lack of governance - Lack of analytics |
| 7. | Nagios | <ul style="list-style-type: none"> - It has an open configuration which is easy to add a custom scripts to extend the services available. - There’re many devices which the Nagios system can monitor. The requirement is an SNMP protocol on that device. - Alert, notification or comment the status of the system. It has varieties of alert tools. - It has many plugins and add-ons which are free to download and develop. | <ul style="list-style-type: none"> - Many features are not available on a free version of Nagios. Features such as wizards or interactive dashboard are available on Nagios XI, which is very expensive. - There’re many configuration files which are very hard to configure. - Nagios core has a confusing interface. - Nagios can’t monitor network throughput (bandwidth uses or available). - Nagios can’t manage the network, just monitor the network. |

8.	Puppet Enterprise	<ul style="list-style-type: none"> - Most mature - Has support for almost all Operating systems. - Has a large community support - Has web UI built-in in the free/open-source version - Good for large-scale environment. 	<ul style="list-style-type: none"> - Agent-based - A steep learning curve, DSL based on Ruby - Pull-based/descriptive so no control of the order of execution of tasks.
9.	Selenium	<ul style="list-style-type: none"> - Wide range of supported languages - Lots of online resources - Has a large community support - Good multi browser support / Parallel testing - Excellent for what it does (when you learn to implement it properly to prevent flaky/brittle tests) - Easy integration with the likes of Maven/TestNG/Jenkins etc - Multi OS support. - It can be used by anyone within an organization (i.e. IT, managers, CEO etc.) - It comes with a lot of plugins and customizations. 	<ul style="list-style-type: none"> - Can be a tad harder to use than some other things, Selenium IDE isn't great - Web based stuff only, e.g no windows applications - Cannot interact with everything in the browser, e.g some pop out dialogs require additional handling elsewhere - Not ideal for image testing stuff.
10.	Splunk	<ul style="list-style-type: none"> - It automatically discovers useful information in data to make your work even more concise. - It makes your system smarter by saving searches and tagging useful information. - It offers alerts to automate the monitoring of systems 	<ul style="list-style-type: none"> - Pricing gets a bit higher for large data volumes. - The optimization of searches is more of an art than just science. - Dashboard is a bit harsh as compared to tableau. - It is continuously making attempts to replace it with open source alternatives.

5. DEVOPS TOOL COMPARISON TABLE

S.No	Tools	Platform	Functions
1.	Ansible	Windows machines as well Linux/Unix machine.	Server and configuration management tool.

2.	Chef	Windows & Mac	Configuration Management Tool.
3.	Docker	OS and platform independent	It delivers configuration for one process at a time, making Docker files simpler than bash script for process configuration.
4.	ELK Stack	Microsoft Windows, macOS and Linux	Collects and Analyse logs from an excel file to a database or server.
5.	Git	Microsoft Windows, macOS and Linux/Unix	It allows you to track changes in your file and, by using it you can easily coordinate the work among your team
6.	Jenkins	Windows, Mac, Linux, FreeBSD, etc.	Continuous Integration tool.
7.	Nagios	Windows, Mac, Linux.	Monitoring Tool.
8.	Puppet Enterprise	Puppet Master works only on Linux/Unix but Puppet Agent also works on windows.	<ul style="list-style-type: none"> - Puppet allows you to define distinct configurations for every host. - The tool allows you to continuously monitor servers to confirm whether the required configuration exists or not and it is not altered. If the config is changed, Puppet tool will revert to the pre-defined configuration on the host.
9.	Selenium	Microsoft Windows, macOS and Linux	Support for parallel test execution reduces the time taken in test execution.
10.	Splunk	Microsoft Windows, macOS and Linux	Store, search, analyze and visualize the machine-generated data.

6. THE ROAD AHEAD

The world of DevOps is full of unique and outstanding open source tools. The above mentioned popular DevOps tools can help effectively bridge the gap between development and production environments when compared previously. You can opt for the tool that suits your business needs and can instantly observe the difference in your business operations. And not only do these different DevOps tools function well inidually, they also play well together.

About the Data Science Foundation

The Data Science Foundation is a professional body representing the interests of the Data Science Industry. Its membership consists of suppliers who offer a range of big data analytical and technical services and companies and individuals with an interest in the commercial advantages that can be gained from big data. The organisation aims to raise the profile of this developing industry, to educate people about the benefits of knowledge based decision making and to encourage firms to start using big data techniques.

Contact Data Science Foundation

Email: admin@datascience.foundation

Telephone: 0161 926 3641

Atlantic Business Centre

Atlantic Street

Altrincham

WA14 5NQ

web: www.datascience.foundation

Data Science Foundation

Data Science Foundation, Atlantic Business Centre, Atlantic Street, Altrincham, WA14 5NQ

Tel: 0161 926 3641 Email: admin@datascience.foundation Web: www.datascience.foundation

Registered in England and Wales 4th June 2015, Registered Number 9624670