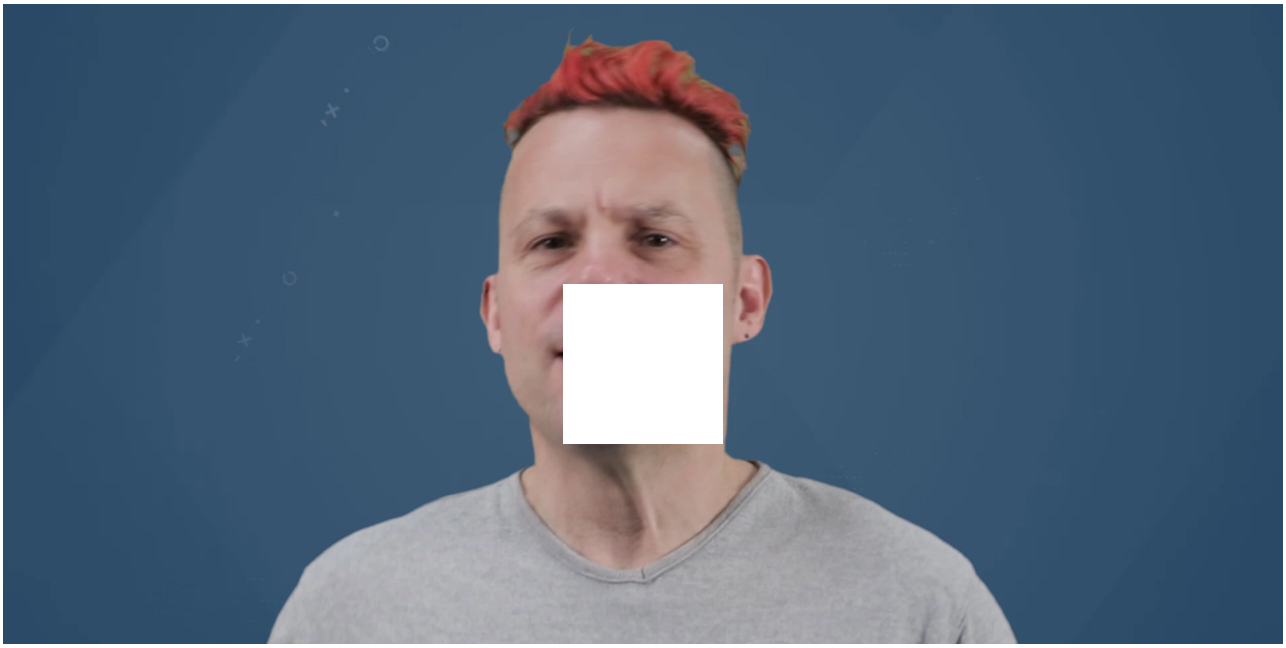
Docker: A cheat sheet

Published March 22, 2017 | Written By James Sanders

Table of Contents

This comprehensive guide about Docker, the leading containerization technology, covers the paradigm shift of containers vs. virtualization, as well as why Docker matters for business users.



00:00 01:47

The widespread adoption of containers has changed the orthodoxy of IT–administrators do not need to run multiple virtual machines to create a barrier between applications or users, nor are programmers beholden to languages such as Java, which have their own strategy for process isolation and lifecycle management.

Leading the way in this change is Docker, an open-source utility that automates the deployment and management of programs inside software containers. Docker is also the center of the container ecosystem, with other solutions such as Kubernetes available for deploying containers through Docker in a computing cluster.

TechRepublic's cheat sheet about Docker is a quick introduction to the software container platform. This "living" guide will be updated periodically as major version updates add functionality to Docker.

SEE: All of TechRepublic's cheat sheets and smart person's guides

Executive summary

- **What is Docker?** Docker is an open-source utility that automates the deployment and management of programs inside software containers.
- **Why does Docker matter?** Compared to virtual machines, running applications inside of Docker containers allows for easier system administration.
- **Who does Docker affect?** Docker runs on Linux, Windows, and OS X hosts, and practically any application can be packaged for use inside a Docker container.
- **When is Docker happening?** Docker was first released as an open source project in March 2013, reached version 1.0 in June 2014, and remains in active development.
- **How do I get Docker?** Docker can be downloaded freely for Linux, Windows, and OS X hosts.

SEE: Cloud Computing Policy (Tech Pro Research)

What is Docker?

Docker is an open-source utility that automates the deployment and management of programs inside software containers. While Linux and Windows natively provide the bulk of functionality for software containers, Docker facilitates the bundling of dependencies and other necessary components into a distributable format. This containerized program can then be deployed on any arbitrary system running Docker. In effect, Docker allows programs to be run the same–independent of the platform on which a given program is run.

If this sounds familiar, the concept of software containers overlaps somewhat with older

technologies. Virtual machines allow for fully independent operating systems to run as a guest on arbitrary hardware, but function identically to running on bare metal. Containers are sandboxed applications running inside the same operating system. Similarly, the write once, run anywhere paradigm of Java is a language-based approach, rather than an environment-based approach, to software portability. While the two compete in an abstract sense, it is possible to run a Java Application Server inside Docker.

For a more in-depth view of software containers and how they compare to other technologies, check out TechRepublic's smart person's guide to containers.

Additional resources

- Containers: replacements or alternatives to virtual machines? (TechRepublic)
- Containers: The pros and the cons of these VM alternatives (TechRepublic)
- Consider this operational challenge before implementing containers (TechRepublic)
- Mini-glossary: Cloud computing terms you should know (TechRepublic)

Why does Docker matter?

The reasons why Docker matters depend on your point of view. As an open source project, it is easily portable and not subject to vendor lock-in or licensing as competing standards (and related technologies such as virtual machines) can be.

In a more general sense, Docker requires a somewhat lower learning curve than virtual machines, and is capable of running on any Linux distribution or on Windows. (The primary competitor for software containers is called rkt; it's an open standard, though presently rkt is functionally exclusive to Container Linux by CoreOS.) Deploying applications in Docker containers rather than inside virtual machines allows for easier administration of systems, as maintenance and patching need only be done on the Docker host, rather than the host system and guest machines. Not running multiple Linux instances in parallel also reduces system overhead. Naturally, by running applications inside containers, administrators can be certain that the processes inside those containers are not interfering with the host system or other containers.

Additional resources

- Video: How Docker brought containers mainstream (ZDNet)
- 4 ways enterprise container technology will work for everyone (ZDNet)
- Why Red Hat makes more money on Docker than Docker does (TechRepublic)
- Why AWS Lambda and serverless computing won't kill Docker in the enterprise (TechRepublic)

Who does Docker affect?

Practically anyone can use Docker. Given the premise of Docker allowing applications to be run on any arbitrary system, Docker is available on various Linux distributions, as well as Windows and OS X. Popular cloud providers such as Amazon Web Services, Google Cloud Platform, and Microsoft Azure provide container hosting as well.

Docker is also the center of its own ecosystem, as other tools have been developed for use in conjunction with Docker, such as Kubernetes, which allows for the deployment of software containers inside a computing cluster.

Additional resources

- Docker launches Enterprise Edition, courts broader corporate adoption (ZDNet)
- Netflix on containers: Five ways they'll boost our business, from lower costs to higher productivity (TechRepublic)
- Containers likely to disrupt Java Virtual Machine usage, survey says (ZDNet)
- What Docker means for enterprises (Tech Pro Research)

When is Docker happening?

Docker was developed as an internal project at dotCloud, and first released as an open source project in March 2013. The 0.9 release introduced the Docker-specific libcontainer library as the default execution environment. Docker reached the 1.0 release milestone on June 9, 2014. According to the release announcement, the 15-month journey from initial release to 1.0 comprised of "8,741 commits from more than 460 contributors [and] 2.75 million downloads."

Additional resources

- Why Docker is finally embracing Kubernetes (TechRepublic)
- Container technology market to grow 40% a year, analysts predict (ZDNet)
- Verizon puts 50,000 Docker containers to work on a massive IoT cluster (TechRepublic)
- Docker rocker: container technology usage doubles; serious money follows (ZDNet)
- Apache Libcloud: The open-source cloud library to link all clouds together (ZDNet)

How do I get Docker?

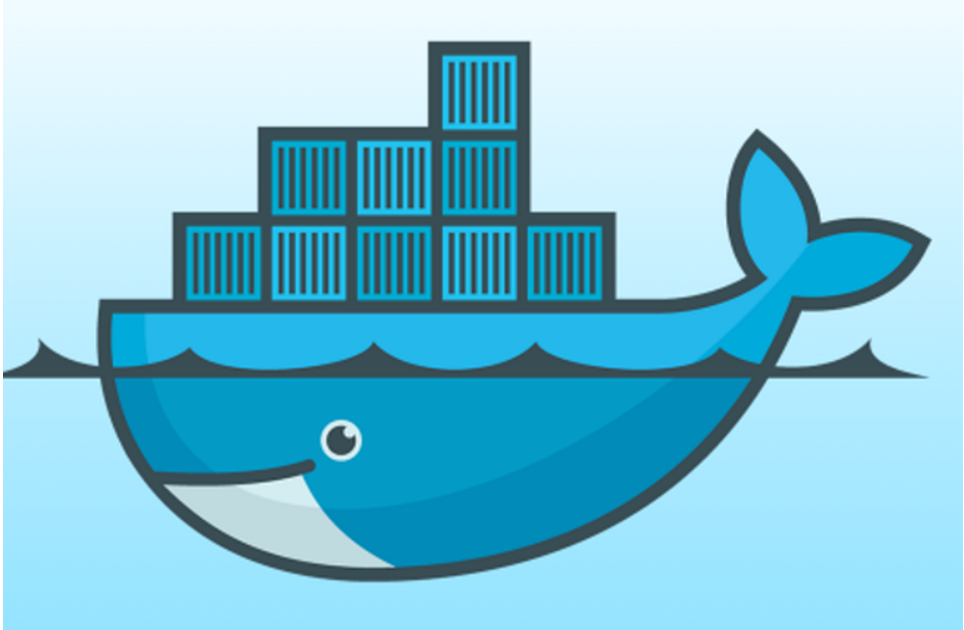
Docker is available for Linux, Windows, or OS X hosts, and can be downloaded freely from the Docker website. The Docker Store provides prepackaged apps such as NGINX, WordPress, Ghost, Drupal, Clojure, and among others for easy deployment. (Despite the

name, many of the available apps are free.)

Docker support is also included in other solutions, such as IBM Bluemix, VMware vSphere, and HPE Helion Stackato.

Additional resources

- Cisco launches validated designs under Docker container partnership (ZDNet)
- How to install Docker on Fedora 25 (TechRepublic)
- HPE to deliver its Docker Ready Server to speed container adoption in the enterprise (ZDNet)
- Many enterprises not ready for 'microservices tsunami' (ZDNet)



Share Article

James Sanders

James Sanders is an analyst for 451 Research. He was formerly a Staff Technology Writer for TechRepublic.

See all of James's content →