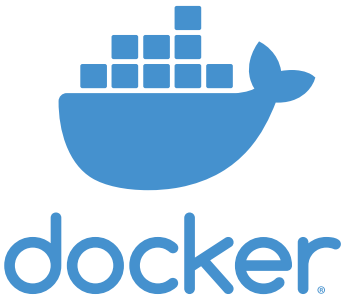


Definitive Guide to Enterprise Container Platforms

OCTOBER 2019

CONTENTS

Digital Transformation and the Software Era

The new era of digital transformation is both being fueled by and is a result of rapid advancements in a few key areas—the growth of public clouds, the rapid adoption of microservices and DevOps and the growing influence of open source technologies. The challenge for many organizations is being able to keep up with these technology advancements in real-time, especially with budgets already allocated to supporting existing applications in operation.

Today's mandate is clear: Organizations need to go beyond digital transformation to adopt modern application development and delivery models that are cloud-first and cloud-ready. Modern applications enable faster response to changing market needs, easier and more frequent release cycles, greater resiliency and the ability to leverage cloud-based services. These benefits lead to faster innovation, which can be applied in two areas of the business:

1. Accelerating new (greenfield) application development, and
2. Modernizing the existing portfolio of applications (brownfield) and extending them with new capabilities

> "Our biggest area of need has been our transformation from what has historically been an industrial manufacturing company … into a software, a hardware and firmware design and manufacturing company."
>
> Brian Magnusson
> VP, Innovation and Technology
> Lindsay Corporation

Unfortunately, today's applications have become very complex, creating challenges for organizations wishing to modernize. Modern applications are highly distributed and modular. A single application may now include multiple cloud services, UI and backend languages, databases and even serverless functions. At the same time, organizations are rapidly moving to a hybrid and multi-cloud operating model to optimize costs and gain access to these new cloud-based services.

To overcome these challenges, organizations need an end-to-end solution that containerizes applications. But the solution must then take modernization a step further, allowing organizations to build, share and run modern applications quickly, effectively and securely across any environment.

BACKGROUND:
INTRODUCTION TO CONTAINERS

A container is a standard unit of software that packages up code and all its dependencies so an application can run quickly and reliably from one computing environment to another. Docker popularized container technology when it launched in 2013 with an open source Docker Engine. It leveraged existing computing concepts from the Linux world, specifically primitives known as cgroups and namespaces, and made it easy to use with both Linux and Windows-based applications.

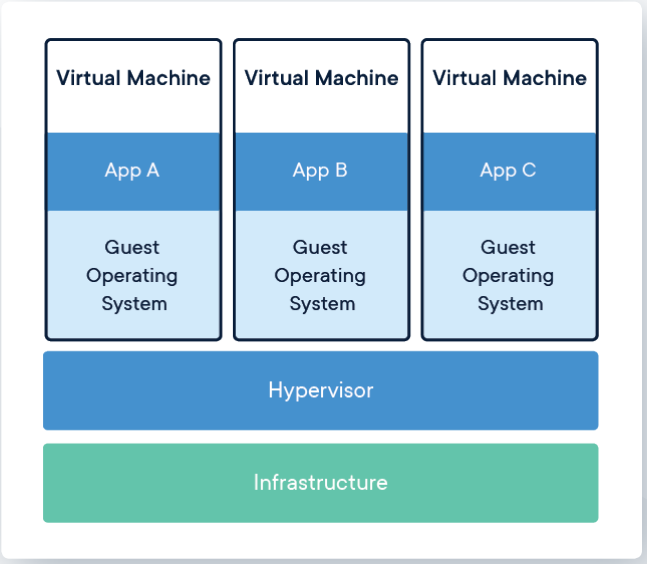
Containers revolutionized computing because it introduced a new way to build and run applications:

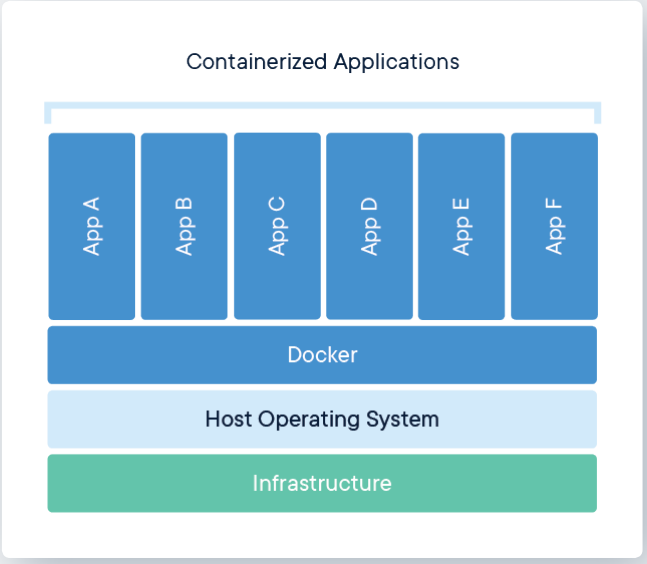
- **Standard**: Docker created the industry standard for containers, so they could be portable anywhere
- **Lightweight**: Containers share the machine's OS kernel and therefore do not require an OS per application, driving higher server efficiencies and reducing server and licensing costs
- **Secure**: Applications are safer in containers and Docker provides the strongest default isolation capabilities in the industry

Containers and virtual machines have similar resource isolation and allocation benefits, but function differently because containers virtualize the operating system instead of hardware.

Virtual machines (VMs) are an abstraction of physical hardware turning one server into many servers. The hypervisor allows multiple VMs to run on a single machine. Each VM includes a full copy of an operating system, the application, necessary binaries and libraries - taking up tens of GBs. VMs can also be slow to boot.

Containers are an abstraction at the app layer that packages code and dependencies together. Multiple containers can run on the same machine and share the OS kernel with other containers, each running as isolated processes in user space. Containers take up less space than VMs (container images are typically tens of MBs in size), can handle more applications and require fewer VMs and Operating Systems. By encapsulating and isolating everything in a container, the container will always run the same, regardless of the environment it is running in. Containers can uniquely turn very diverse set of application services into standardized software units.





75% of global organizations will be running containers in production by 2022

"Best Practices for Running Containers and Kubernetes in Production", Gartner, February 2019

The Emergence of Container Platforms

Over the past few years, the industry has standardized around the container as the format to solve some of the key challenges around application portability and developer productivity. Docker containers famously solved the "it works on my machine" problem by abstracting away application dependencies and spurred the growth of microservices-based application architectures.

But containers on their own are just a technology. Just as in the 1950s, shipping containers introduced a new way to package and distribute goods across the entire transportation system including trucks, trains, cargo ships, and ports that truly revolutionized goods distribution. In a similar way, today's container platforms are built on industry-leading standards like the Docker container runtime and Kubernetes orchestration. However, it is the system of tools operationalizing containers that will transform a company. A container platform goes beyond Kubernetes and orchestration in building a complete solution around the lifecycle of the containerized application, including how those applications are created, where they are stored and how they get integrated into other IT tools and processes.

BACKGROUND:
INTRODUCTION TO KUBERNETES

Kubernetes is an open-source container orchestration engine for automating deployment, scaling and management of containerized applications. Originally created by engineers at Google and now hosted by the Cloud Native Computing Foundation (CNCF), Kubernetes has seen rapid growth and adoption in a few short years.

Kubernetes is primarily focused on the running of containerized applications, including the following features:

- Service discovery and load balancing
- Service placement and bin packing
- Self-healing
- Storage orchestration
- Automated rollouts and rollbacks
- Secret and configuration management
- Horizontal scaling
- Batch execution

Docker packages a certified distribution of Kubernetes in Docker Desktop and Docker Enterprise, along with additional tools that make a more complete, enterprise-ready solution.

Enterprise Platform Requirements for the Digital Era

As traditional businesses transform into application companies, speed, choice and security become absolutely essential. Yet achieving those qualities is increasingly complicated given technical debt, vendor stacks and the risks and requirements around protecting data and systems in distributed hybrid and multi-cloud scenarios. Container platforms enable a new wave of software-based innovation, but they need to support the broader company's objectives.

Speed

Today's market dynamics and competitive environment are changing so rapidly that companies who can respond and react quickly will be the most successful. However, most CIOs surveyed think they are falling behind in their digital transformation efforts. Things that slow an organization down include unnecessary barriers to developer productivity, friction between developers and operators and steep learning curves for new technologies. The container platform must be designed for high-velocity innovation across the entire software development lifecycle and should be intuitive for both developers and operators. Platforms should extend to developers' systems to enable "shift left" testing and security.

Be wary of:

- Platforms that force adoption of specific software development patterns or require learning specific coding techniques
- Platforms that require additional tooling for each new type of technology or application
- Platforms that only allow "late" testing, risking design defects that are not detected until applications are deployed to staging or production
- Platforms that focus more on deploying Kubernetes than the software development process

Choice

New technology stacks and frameworks are being introduced every day. The container platform must work with both existing technology investments and next generation technologies alike. It should be independent of both the underlying infrastructure as well as the applications and frameworks, providing a consistent and uniform operating model for different application types intended for different operating environments to prevent lock-in.

Be wary of:

- Platforms specializing in limited types of application frameworks or technology stacks
- Platforms that support only one infrastructure or operating system
- Platforms that cannot address the full spectrum of existing and new applications

Security

The new digital economy has also introduced a new wave of cybercrime. Security has become more complex and high-stakes and organizations are faced with the challenge of empowering developers to be productive while protecting themselves against risk. The new application architectures are highly dynamic and scalable, outgrowing traditional security models so the container platform needs to be built from the ground up with security in mind.

Be wary of:

- Platforms that require 3rd party security tools due to lack of built-in capabilities
- Platforms that ignore the developers' systems as a key to creating safe applications
- Platforms that cannot support required application security models, such as Active Directory authentication for Windows applications

Components of a Complete Container Platform

A container platform should provide a complete solution for the building, sharing and running of containerized applications. Oftentimes, focus is placed strictly on the running of containerized applications, however the impact of containers on an organization is the transformational change it brings to the entire software development process. A container platform should address the needs of both developers and operators who often work in different paradigms. Container platforms should also improve the handoff between these groups.

BUILD

- Developers work on code
- Compile, assemble and build apps

SHARE

- Teams collaborate
- Source content from others

RUN

- DevOps teams deploy apps
- Manage, monitor and patch

From concept to fruition, a containerized application passes through multiple lifecycle stages. Each of these stages introduce new requirements for the container platform:

Build

Containerized applications usually start with the developers who are building applications in a local environment—typically a desktop or laptop. The container platform needs to provide developers a simple and secure way to rapidly build containerized applications and microservices. It should easily integrate with existing software development tools like developers' preferred IDE (e.g. Visual Studio, Eclipse, IntelliJ), and it should work across different languages and frameworks (e.g. .NET, Java, Node.JS)—so developers can select whatever makes the most sense for a particular project. Developers also need a quick and simple way to build in compliance with corporate and architecture standards—without impacting developer productivity.

CRITICAL COMPONENTS:

Speed

- Offers a simple desktop-based experience and intuitive design to build modern, container-based applications and microservices without being an expert
- Provides a desktop container development environment that integrates with existing developer workflows
- Develops with certified images and application templates

Choice

- Works with any application, framework or language
- Develops applications for any OS and any infrastructure using a single development pattern
- Is configurable to existing CI processes and tools

Security

- Provides predefined and customizable application templates that adhere to corporate and IT standards
- Features secure orchestration with a fully-conformant Kubernetes
- Standardizes deployment and configuration of developer environments using existing endpoint management solution

Share

Today's digital economy requires the ability to innovate rapidly on top of previous innovation and close collaboration among teams. In mid-sized to large organizations, this increasingly involves DevOps teams who embrace Continuous Integration and Continuous Delivery (CI/CD) principles for agile delivery. A container platform should include tools that make it both simple and secure to collaborate on applications while assisting in the rapid delivery of applications through the software pipeline.

CRITICAL COMPONENTS:

Speed

- Features an automated policy engine to support the full lifecycle of containers, push images and mirror containerized content to distributed teams
- Replicates approved images across multiple clusters, putting the latest content right where it's needed

Choice

- Provides access to trustworthy content
- Features Webhooks and API interfaces for integrating with CI/CD processes and tools
- Configures to run in any hybrid-cloud environment

Security

- Facilitates an automated container image supply chain for improved security, governance and faster delivery
- Performs binary-level scanning of images for known vulnerabilities
- Integrates role-based access control (RBAC) with internal user directories to implement fine-grained access policies

Run

The container platform should make it easy to deploy, manage, update and secure containerized applications with globally consistent environments. It should allow developers to run applications on any data center or cloud and any architecture or OS without requiring deep expertise. These environments—which may be distributed and/or hybrid—need to be secure while being easy to operate, which starts at the foundation with a secure container runtime, a 100% portable security model and a standard set of APIs and tools. Finally, the platform needs to offer a highly scalable operator experience that integrates well with storage, networking, logging and monitoring tools and provides a management plane for ongoing operations.

CRITICAL COMPONENTS:

Speed

- Features an intuitive design and guided workflows for unified management of applications and fast troubleshooting
- Simplifies lifecycle management (Day 1 and Day 2) with automated tools for installation, upgrades, backup, restore and scaling up and down

Choice

- Supports any OS and deployment infrastructure—bare metal, VM, or hybrid-cloud
- Configures to CD processes and tools

Security

- Deploys Kubernetes environments automatically with out-of-the-box secure defaults
- Signs images digitally from the source and prevents unvalidated content from being deployed to production
- Delivers secure multi-tenancy through advanced access controls that integrate with enterprise authentication and authorization tools

There are different container platform options available in the market, and organizations have the ability to build their own container platform piece by piece. Organizations will need to assess their requirements to determine which platform fits their needs.

Introducing the Docker Platform

Over the last five years, containers have become the fundamental unit of software for building applications and providing standardized units for development, shipment and deployment. At Docker, our history is rooted in making containers simple and ubiquitous. But modern applications have become much more complex than a few individual containers. As stated in the introduction, today's applications are highly distributed and modular and must be designed to securely, efficiently and cost-effectively run across hybrid and multi-cloud environments. Docker is focused on making these diverse and distributed modern applications as easy to develop and deliver as we made containers.

In the Forrester New Wave™: Enterprise Container Platform Software Suites, Q4 2018 report, Docker was cited as a leader in the enterprise container platform category with the Docker Platform (then known as "Docker Enterprise") receiving a "differentiated" rating in eight of ten criteria. Forrester added that the Docker Platform "leads the pack with a robust container platform well-suited for the enterprise".

The Docker Platform is the industry-leading, standards-based container platform for rapid development and progressive delivery of modern applications. Only Docker delivers a consistent and secure desktop-to-cloud application pipeline to Kubernetes environments in any cloud, choice of tools and languages and central point of collaboration that simplifies and streamlines processes across dev and ops.

Speed – The Docker Platform enables organizations to rapidly deliver engaging new customer experiences and transform existing processes. It does this by enabling developers to accelerate time-to-productivity and production with faster developer on-boarding and streamlined workflows. Operators are then able to bring applications to market faster by modernizing the way they're built, managed and secured.

Choice – With the Docker Platform, organizations have optimal agility and flexibility to meet their business needs. They can easily adapt to the next technologies on their own timeline while leveraging existing knowledge and processes. Developers have the freedom to select the best tools, languages, application stacks and deployment environments for each project. Operators have the freedom to pursue the right operational strategy for the business—across any app, OS and infrastructure.

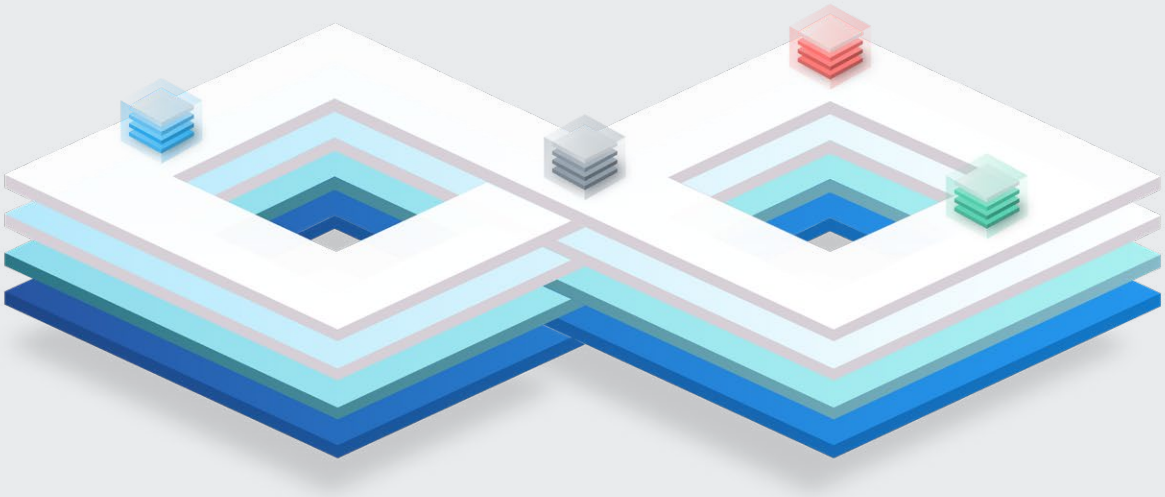
Security – The Docker Platform enables organizations to continuously ensure compliance and mitigate risk without slowing down innovation. It is the only platform that can provide trusted and certified end-to-edge security with automated governance and compliance throughout the application lifecycle with a multi-layered security approach that is delivered by design and by default. It offers scalable security solutions built on a policy-based governance model that supports secure multi-tenancy and integrates with the software development process to operate at the speed of DevOps.

A Complete End-to-End Solution

The Docker Platform makes it possible to build, share and run modern, containerized applications as simple, repeatable processes for application teams and perfectly orchestrated, always-on experiences for customers.

Docker Enterprise Platform

Securely build, share and run any application, anywhere



Build

Rapidly build containerized applications

Docker Application
Docker Desktop Enterprise

Share

Securely share and access certified and approved content

Docker Hub
Docker Trusted Registry

Run

Deploy, manage and secure modern applications anywhere

Docker Kubernetes Service
Universal Control Plane
Lifecycle Management

It features the following key end-to-end capabilities to help build, share and run modern applications:

Build: Rapidly build containerized applications and microservices in a secure way while leveraging existing tools and skill sets:

- Docker Desktop Enterprise: Simple desktop-based experience and intuitive design to build modern, container-based applications and microservices without being a Docker expert.
- Docker Application: A new first-class object across the full Docker Platform that defines any type of multi-service application—on-premises or cloud native, Windows or Linux—as a standard, portable package for sharing and running anywhere.

Share: Seamlessly find and securely share certified and approved content leveraging Docker Hub, the world's largest library of container content and Docker Trusted Registry (DTR):

- Docker Hub: Use validated and secure container content from Docker Hub to support a diverse set of application stacks and infrastructures.
- Docker Trusted Registry (DTR): Manage and operate a globally consistent private content repository for distributed development teams with the most advanced private registry for container images.

Run: Deploy, manage and secure modern applications with Docker Kubernetes Service, a consistent Kubernetes environment that runs on any cloud:

- Rapidly deploy Kubernetes clusters anywhere with secure defaults out-of-the-box and without being an expert.
- Use consistent and commercially supported Kubernetes from the developer's desktop to production servers.
- Access the Control Plane to centrally manage and operate containerized applications.

In addition, the Docker Platform delivers:

- **Extensibility** that connects the container platform easily into other enterprise tools, including preferred storage and networking solutions, logging and monitoring tools.
- **Enterprise support and services** with advisory capabilities to help resolve potentially complex process, orchestration, integration and workflow challenges.
- **Certification** to assure interoperability with the ecosystem and other data center platforms.

How the Docker Platform Spurs Innovation

Docker has a unique approach that provides a broader view of digital transformation. It is one that focuses not just on the new microservices and cloud-native applications; it also looks at an organization's existing application portfolio and how to bring it forward into the digital era:

- **Modernize brownfield applications** - For applications that are being actively maintained, containerizing is the first step to further application modernization. Once containerized, organizations often begin the work of segmenting off specific capabilities and building new microservices to replace old monolithic architectures or replacing key building blocks and moving them to cloud-based services (e.g. moving to DBaaS).
- **Accelerate greenfield applications** – Next, organizations need to build new and compelling experiences for their customers and flexible, responsive systems for their businesses. New greenfield applications can come in many different flavors and architectures. Docker's methodology puts the emphasis in the innovation process, so developers can make the technology and architecture choices that best fit the needs of the application. With Docker's focus on choice and flexibility, organizations have the freedom to use existing stacks or explore new ones.

Finally, Docker's approach is to prepare organizations for whatever is next—whether that is new business models, new opportunities or responding to new competitive threats.

> "What we sought out to do (with Docker Enterprise) is create a multi-lane highway that could accelerate application delivery into the cloud in a way that gave us better portability, better speed, and better agility for our development teams...a model where we could containerize our traditional legacy applications and get them to the cloud, modernize some of them into microservices, and fuel innovation around net-new microservices."
>
> Eric Drobisewski
> Senior Architect
> Liberty Mutual

The Docker Platform has tangible benefits that immediately deliver a high return-on-investment:

- **Unified operations**. When everything is standardized and follows the same operational patterns, it's easier for IT teams to explore new technology areas—and for the company to adapt and embrace new services. With Docker, Franklin American runs a single cluster that supports the development, test and production environments.
- **Leverage existing teams and processes**. This goes back to standardization. With a common platform, processes become repeatable. It's easier and faster to experiment or just make iterative changes. Cornell University accelerated application deployment times by 14x; Kadaster, the Dutch land registry, went from one new deployment a month to as many as 500.
- **Respond to risks and threats**. The agility and standardization offered by a container platform makes it easier to apply consistent security to protect the organization from threats. Bosch uses Docker to reduce security and compliance risks by accessing app content on the internet. The company can now securely serve 62,000 global developers with highly available, complaint infrastructure and over 1,000 secure image repos.
- **Increase data center utilization by 3x**. Even with virtualization, most data centers operate at—at best—20 percent utilization. Containerization increases utilization 50 or 60 percent by eliminating redundant operating systems and further consolidating systems.
- **Decrease IT operating costs**. Lifecycle management and infrastructure standardization make system patching, application updates and even rollbacks much faster. Docker is helping Citizens Bank accelerate software development and reduce server costs by 40 percent and storage costs by 90 percent.
- **Fund innovation**. As a Fortune 100 insurance company put it in their DockerCon presentation, companies can "self-fund innovation" since the savings from the Docker Platform can be reinvested into innovation.

Taking the Next Steps

What organizations should do next depends on where they are now, and what they want to achieve.

- **For organizations that have not started on the containerization journey:**
  - Onboard developers to modern application development models with Docker Desktop and start learning about Docker Applications.
  - Identify a first project. Show early success by bringing a set of containerized applications to production—either brownfield or greenfield—and begin planning an innovation team.
- **For organizations that have begun using containers and/or container orchestrators:**
  - Focus on ways to accelerate developer productivity with a secure software supply chain that is integrated with existing software pipeline tools.
  - Invest in training and certification to make sure teams know how to get the most out of a container platform.
  - Expand from initial pilot projects to production at scale by extending modern application practices to a broader set of applications.

Click here to learn more and get started with Docker Enterprise or contact sales for more information.

www.docker.com