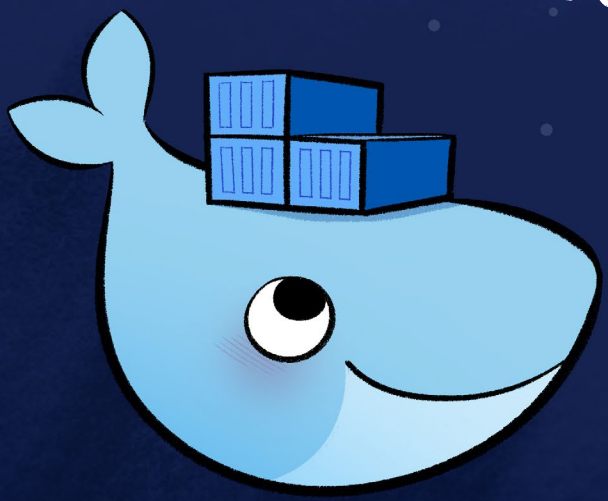
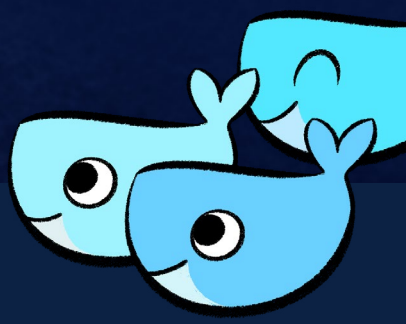
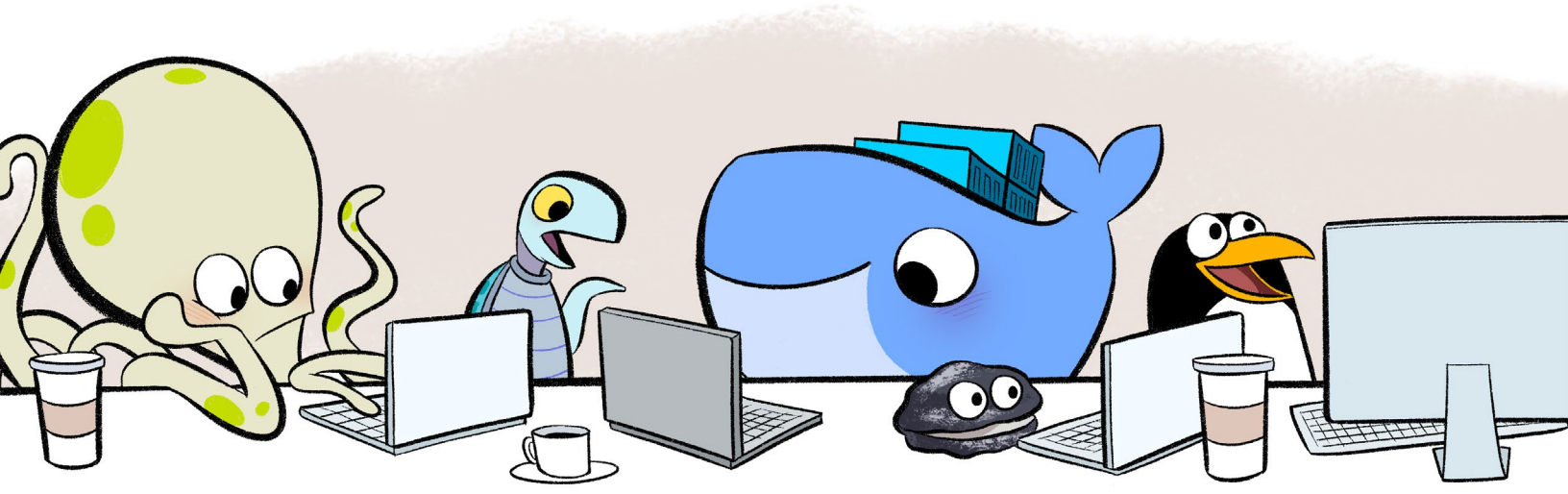
The State of Application Development in 2022 *and Beyond*





docker

Contents

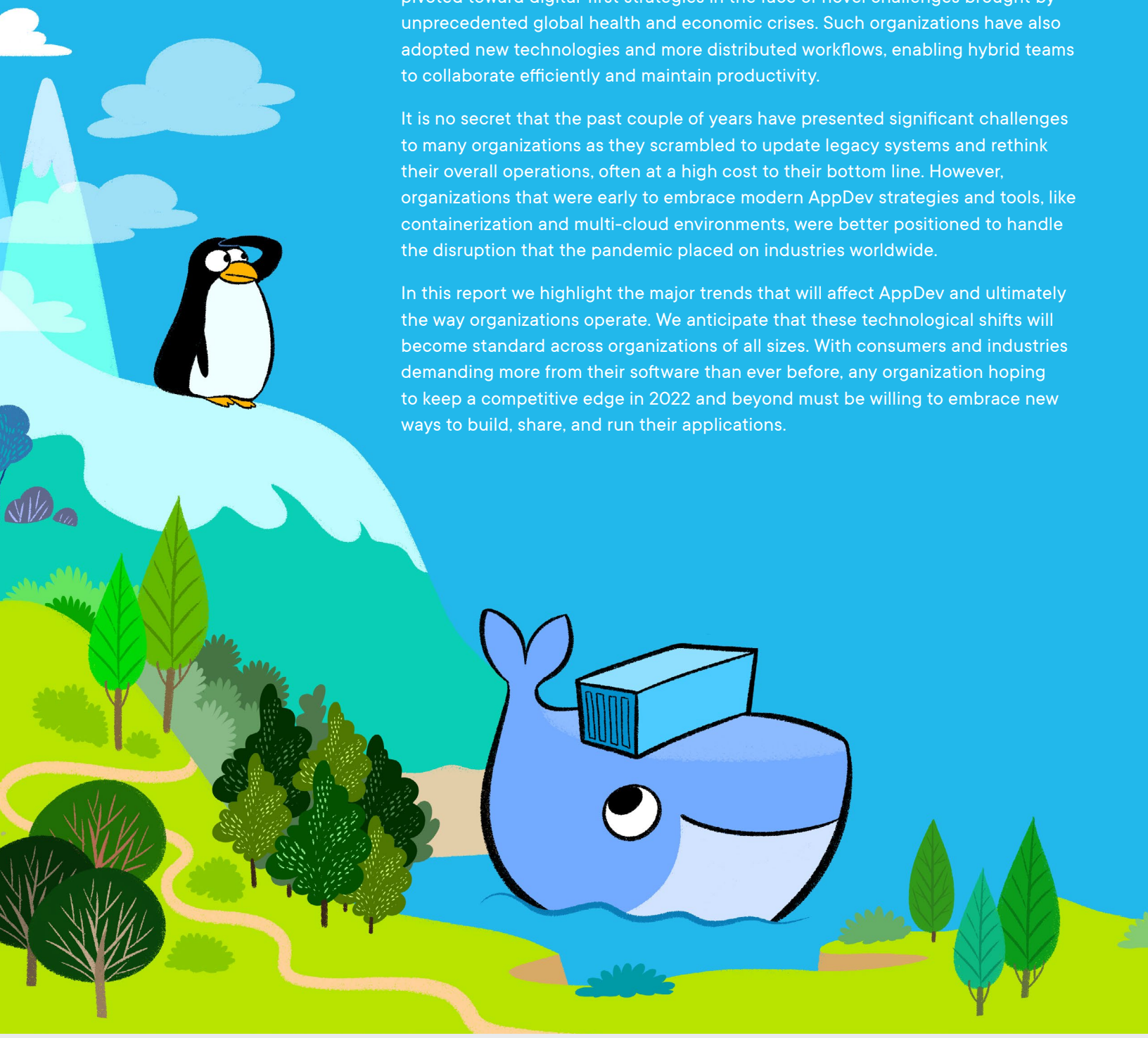


Introduction

Application development (AppDev) trends have been guiding industries (tech and non-tech alike) toward a more cloud-native and distributed model for nearly a decade. These changes have accelerated in recent years, as organizations pivoted toward digital-first strategies in the face of novel challenges brought by unprecedented global health and economic crises. Such organizations have also adopted new technologies and more distributed workflows, enabling hybrid teams to collaborate efficiently and maintain productivity.

It is no secret that the past couple of years have presented significant challenges to many organizations as they scrambled to update legacy systems and rethink their overall operations, often at a high cost to their bottom line. However, organizations that were early to embrace modern AppDev strategies and tools, like containerization and multi-cloud environments, were better positioned to handle the disruption that the pandemic placed on industries worldwide.

In this report we highlight the major trends that will affect AppDev and ultimately the way organizations operate. We anticipate that these technological shifts will become standard across organizations of all sizes. With consumers and industries demanding more from their software than ever before, any organization hoping to keep a competitive edge in 2022 and beyond must be willing to embrace new ways to build, share, and run their applications.



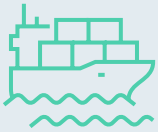
Introducing the Six Trends to Watch in 2022

If the past couple of years are any indication, 2022 is likely to be another roller coaster of a year, and development teams can expect even more surprises that may undermine their productivity. **Here are six AppDev trends that organizations should keep on their radar and proactively address to minimize disruption and remain competitive.**

1
Work is More Distributed Than Ever

2
CI/CD Will Be the Basic Building Block

3
Containerization Will Be the Norm



4
Secure Software Supply Chain Becomes Vital

5
DevSecOps Will Rise to the Occasion

6
Organizations Will Continue to Automate

1. Work is More Distributed Than Ever

Changes in cloud computing and containerized microservices have encouraged a departure from centralized software architecture models. Organizations have broken up traditionally monolithic applications operating on a centralized server into discrete parts that form microservices. Microservices operate in their own self-contained environments. They are often distributed across different networks, cloud providers, and geographical locations. Indeed, microservices have changed the environments where software operates, enabling a fundamental shift in how AppDev teams collaborate to create applications.

To comply with recent government regulations, many organizations shifted to a work-from-home or hybrid work model. Many undertook substantial efforts to refine their organizational structure, workflows, and available tools to accommodate a more distributed workforce. Due to this shift, 2022 promises to be a year of even more distributed software, teams, and workflows.

Remote and hybrid AppDev teams will continue to grow in prevalence. According to The State of Remote Engineering 2021, **75 percent of the 1,100 software engineers surveyed said they preferred having the option of a hybrid system.** This system would allow them to split their schedules between remote and in-office work. In addition, about one-third of the engineers stated they prefer to work all their hours remotely.

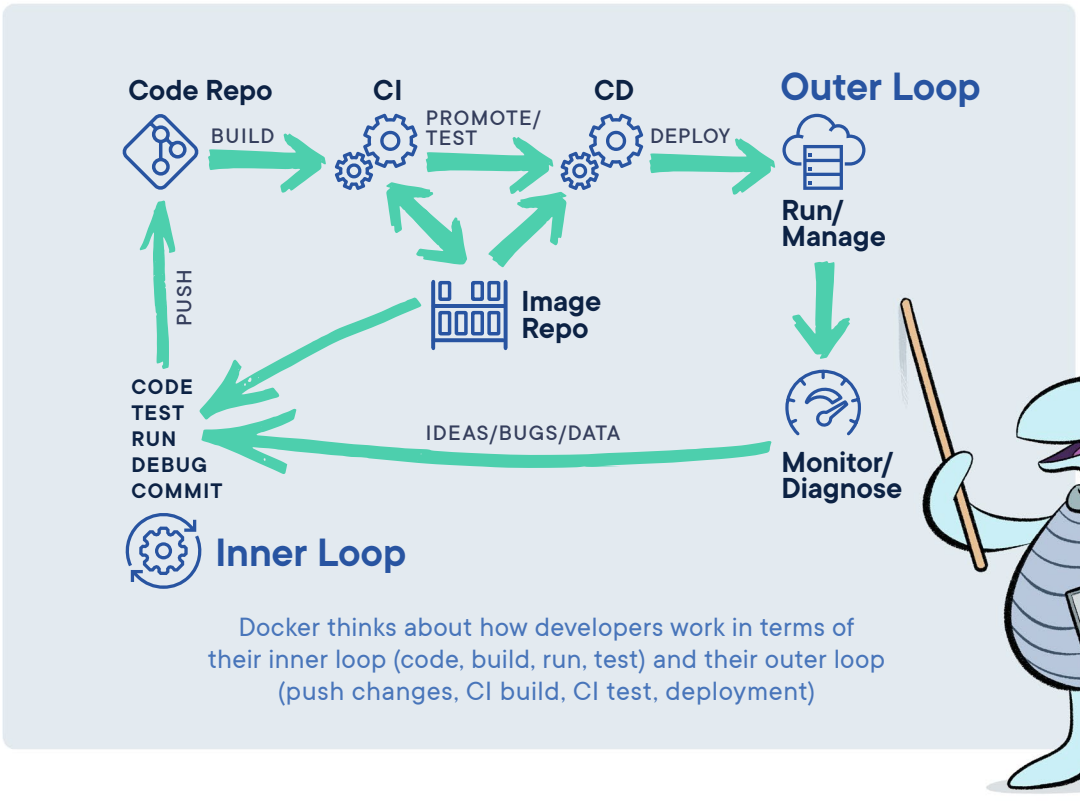
Fortunately, a rich ecosystem of tools to support a more distributed and collaborative software development model has bloomed into capable systems. Docker Business enables organizations to coordinate large teams working at cloud scale across Docker and other AppDev environments. Developers get the powerful tools and services they have come to expect from Docker. At the same time, centralized visibility and management controls empower admins to manage environments and effortlessly onboard and off-board remote developers.

75%

of the **1,100 software engineers** surveyed said they preferred having the **option of a hybrid system.**

2. CI/CD Will Be the Basic Building Block

While we can trace the origins of continuous integration and continuous deployment (CI/CD) back to the mid–2000s, the adoption of CI/CD principles has been a gradual one. In 2022, CI/CD will no longer be just an option. It will be *the* fundamental building block of software development.



Docker thinks about how developers work in terms of their inner loop (code, build, run, test) and their outer loop (push changes, CI build, CI test, deployment)

44%

of developers are using some form of **continuous integration and deployment** with Docker containers.

CI/CD offers a variety of benefits perfectly suited for distributed development environments. Because the CI/CD model is based on continually integrating and deploying small code changes, it lends itself well to a decentralized software development model. Regardless of their physical location, developers can make small changes to the codebase during their workday and synchronize the changes with the project remotely, relying on version control tools to avoid conflicts or overwriting other contributions.

Automation also plays a key role to enable a truly continuous process. This approach includes the adoption of automated testing, verification, and deployment steps to process code existing at all stages within the CI/CD lifecycle. An effective CI/CD implementation enables development teams to focus on creating and refining features, minimizing the day-to-day operational concerns of moving code from development environments to deployment.

Docker plays an integral part in many organizations' CI/CD pipelines, fitting seamlessly into their automated testing and deployment strategies. In fact, a survey found that 44% of developers are using some form of continuous integration and deployment with Docker containers. Docker containers can become the basis of a new CI/CD pipeline or enhance an existing CI/CD pipeline to streamline testing and deployment. As an example, many development teams have found success in configuring GitHub Action CI/CD pipelines with Docker.

3. Containerization Will Be the Norm

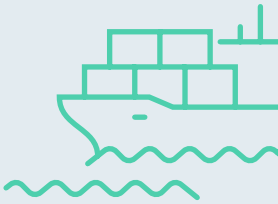
Containerization represents one of the most revolutionary departures from traditional software development methods. It has completely changed the way organizations design, develop, and deploy software. In fact, we would go so far as to say that containerization should be the basis of distributed software, distributed AppDev teams, and CI/CD.

Containerized applications improve software scalability and security by relying on a model of lightweight services with rich connectivity between them. Isolating an application's discrete parts into their own services allows developers to create flexible environments and more capable software. It also allows developers to quickly iterate on changes to specific microservices without affecting the software as a whole.

According to the 2020 State of Cloud Native Development report, **60 percent of back-end developers now use containers.** This statistic represents a 10 percent increase from 2019. With such indicators showing the growing importance of containerization in AppDev, containers are likely to become the default environment for back-end systems this year and in the years to come.

Docker provides a complete framework for creating portable containerized applications. With help from Docker's library of Docker Official and Verified Publisher Images, developers have quick access to all the dependencies necessary to easily spin-up new containers and in minimal time. In addition, Docker's centralized management tools make it easy to operate at scale, whether working with a few simple containers or managing a complex enterprise deployment. Organizations can also stay updated on any changes and need-to-know details of their containerized environments.

60% of back-end developers **now use containers**



4. Secure Software Supply Chain Becomes Vital

According to the 2021 State of the Software Supply Chain, the number of **software supply chain attacks increased by**

650%

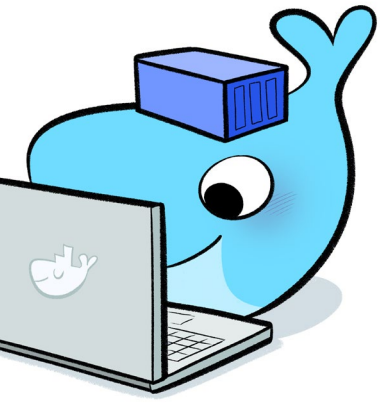
from 2020 to 2021.

Software has grown more interconnected as it evolved. Gone are the days when an in-house development team would write all the code in an enterprise application. Today, developers combine many third-party libraries and technologies to quickly create solutions uniquely tailored to their organization's needs. However, such productivity enhancements come at the cost of creating more dependencies and security risks.

While using third-party codet has made it possible for developers to be more nimble and productive, and applications to be more powerful and feature-rich, it has also introduced its own share of security concerns. A modern application is heavily dependent on the software supply chain responsible for assembling all the necessary dependencies allowing the application to function.

As a recent example, the Log4j vulnerability discovered in December 2021 caused a global uproar as many organizations were dependent on this common Java logging library and scrambled to find a fix. At one point, malicious groups had attempted to exploit the flaw on over 40 percent of global networks. Needless to say, introducing code that internal teams did not write or verify creates serious security concerns for organizations and enterprises.

The risk of introducing vulnerabilities or malicious code into the software supply chain remains a constant concern in AppDev. In fact, opportunistic attackers have taken note of the software supply chain's growing interconnectedness. According to the 2021 State of the Software Supply Chain, the number of software supply chain attacks increased by 650 percent from 2020 to 2021 . The primary attack vectors are dependency confusion, typosquatting, and malicious code injection.



Docker Business provides a set of powerful tools for securing, managing, and verifying your software supply chain. Docker's Image Access Management offers finer-grained control over which Docker images (that is, executable packages of software that includes everything needed to run an application: code, runtime, system tools, system libraries, and settings) your team members can use in containers. With help from Docker's Official Images and Verified Publisher Program, organizations have direct access to repositories verified and secured by Docker and its trusted partners.

Docker provides a shortcut for securing the software supply chain over your application's entire lifetime by minimizing the need to seek out dependencies from potentially unreliable repositories. In addition to sourcing images from Docker's secured repositories, Docker Business users can also take advantage of built-in automated vulnerability scanning to help detect known vulnerabilities within their images before they lead to issues.

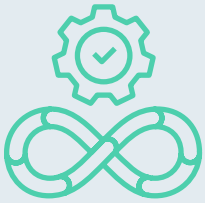
5. DevSecOps Will Rise to the Occasion

With cloud-native software serving a foundational role in almost every industry, the need to ensure security throughout the entire application lifecycle is paramount. A March 2021 attack on Microsoft exposed the data of 30,000 customers. In another example, 530 million users' private data leaked from Facebook that August. Organizations must prepare for an increasingly hostile digital environment or risk the high cost of fixing security breaches, experiencing operational disruptions, and losing customers.

Gartner predicts that organizations adopting a "cybersecurity mesh architecture" by 2024 **may reduce the financial impact of security incidents by 90 percent on average.** The holistic view of cybersecurity found in a cybersecurity mesh architecture parallels another critical development: the rise of DevSecOps.

DevSecOps rose to popularity in the 2010s, paving the way for a more seamless integration and collaborative effort between development, security, and operations teams. Recognizing a more proactive approach to software security, the DevSecOps philosophy integrates security into every phase of the AppDev lifecycle–from the initial design phase to deployment and throughout its service life.

The ability to log, audit, and visualize access and changes to your application environments is essential to any software security practice. Docker provides complete visibility into containers, accompanied by detailed logs describing who made what changes, enabling organizations to audit all the activity across their environments.



Gartner predicts that organizations adopting a "cybersecurity mesh architecture" by 2024 may **reduce the financial impact** of security incidents **by 90%** on average.

Organizations adopting a DevSecOps methodology will appreciate Docker's built-in security features. Docker Business features, such as Image Access Management, ensures that all the links to the organization's software supply chain are trusted sources. By handling these essential tasks, Docker helps DevSecOps teams focus their attention on security details specific to their applications, rather than worrying about introducing vulnerabilities from their supply chain.

6. Organizations Will Continue to Automate

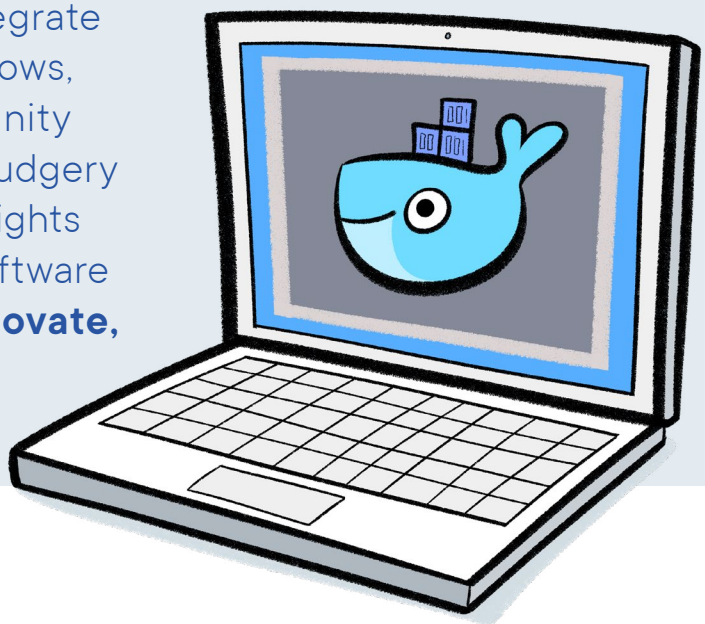
As artificial intelligence (AI) and machine learning (ML) grow more capable, automation is increasingly working its way into familiar DevOps and DevSecOps practices. While organizations traditionally assigned automation to handle mundane and repetitive tasks, the growing capabilities of AI and ML systems now allow these tools to help with more complex AppDev work. For instance, AI may inform decision-making and provide automated remediation to common problems with known solutions.

In 2022, organizations capable of synergizing automation with human efforts will enjoy a substantial competitive advantage. Rather than automation completely replacing human work, it will augment it. By designing systems that integrate automation with human workflows, organizations have an opportunity to free their best minds from everyday drudgery while simultaneously providing valuable insights gleaned from AI. This approach provides software development teams with more room to innovate, experiment, and create.

Automation empowers all the core demands of security, scalability, and speedy development cycles. With automation playing key roles in compiling, testing, and deploying code, CI/CD pipelines can become truly continuous. Automation can also play a vital role in orchestrating the underlying infrastructure, allowing applications to respond to demand and sniff out potential security vulnerabilities or other problems. With the superpower of automation on their side, developers are untethered from repetitive tasks and free to innovate new ideas.

Many organizations rely on Docker to provide the container environments essential for automation tools and CI/CD pipelines. In addition to enabling an organization's automation efforts, Docker also provides automation tools natively, such as automated vulnerability scanning and Docker Automated Builds. Automated Builds enable Docker Business users to automatically build images from source code and push them to Docker repositories.

By designing systems that integrate automation with human workflows, organizations have an opportunity **to free their best minds** from everyday drudgery while simultaneously providing valuable insights gleaned from AI. This approach provides software development teams with more room to **innovate, experiment, and create.**



How Docker Business Can Help in 2022 and Beyond

The global pandemic was responsible for much of the disruptions impacting nearly every industry. When government regulations forced offices to close, many development teams found themselves having to experiment with new ways to stay productive and collaborate from home. AppDev teams faced an unprecedented demand for new applications while operating under reduced capacity. It was clear that new tools and methods were necessary to tackle these challenges in a scalable and more secure way. **Docker Business was designed with developers in mind, bringing together the Docker tools developers know and love with the added security and management features enterprises can trust.**

Docker Business provides access to a powerful suite of tools and services to build, share, run and manage containerized applications, including:

Commercial Use of Docker Desktop

Docker Desktop is a powerful developer tool that increases developer productivity and collaboration. It enables developers to integrate Docker into their existing AppDev stack and workflow. Docker Desktop removes the complexity of developing container applications on MacOS and Windows, streamlining installation and setup, and helping teams focus their efforts where they are needed most. Plus, Docker Desktop handles the details of managing their containerized inner-loop development environments. Software development teams can easily create consistent development environments across their entire organization by letting Docker Desktop take care of the complexity .

Centralized Management Console

Administrators and managers can view, manage, and edit their Docker environments directly from their web browser. Gain detailed visibility into the various teams and individuals contributing to projects, and seamlessly administer their access and permissions.

Image & Registry Access Management

Administrators can control the registries and content their developers can access through access management controls on Docker Hub. Set up specific rules for which images and registries team members can access for their environments. Admins can also restrict access to trusted sources, such as Docker Official Images and Verified Publisher Images. Built-in automated vulnerability scanning provides an extra layer of security, automatically scanning container images for known vulnerabilities and providing recommendations for remediation.

SSO

Docker offers native support for single sign-on (SAML 2.0 and Azure Active Directory). By enabling SSO, organizations can easily automate the onboarding and management of Docker users at scale. Docker users can authenticate using their organization's standard identity provider (IdP).

Auditability

With a robust suite of visibility tools, Docker Business enables organizations to audit the actions of specific contributors and teams, tracking activity across various environments and containers to gain detailed insights into how teams are collaborating on projects.

Premium Support

For Docker Business customers operating in enterprise environments, Docker offers an additional Premium Support package that provides prioritized access to Docker's expert support team. These experts work diligently to ensure developers are getting the most from their Docker Business experience.

Docker Business was **designed with developers in mind,** bringing together the Docker tools developers know and love with the added security and management features enterprises can trust.

Do More with Docker Business

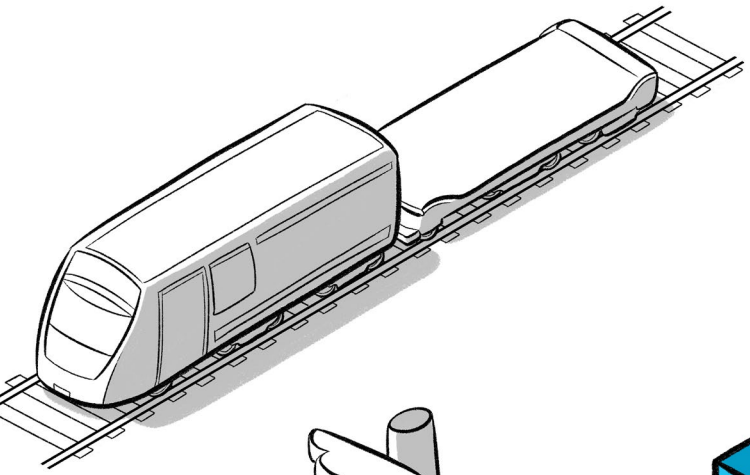
Docker Business brings together the Docker tools, services, and community developers know and love, with added enterprise-grade features. In addition to helping developers build, share, and run applications with Docker Desktop, Docker Business incorporates comprehensive management and security controls. The result is empowered developers building modern, secure, and reliable apps at scale without compromising on security and compliance.

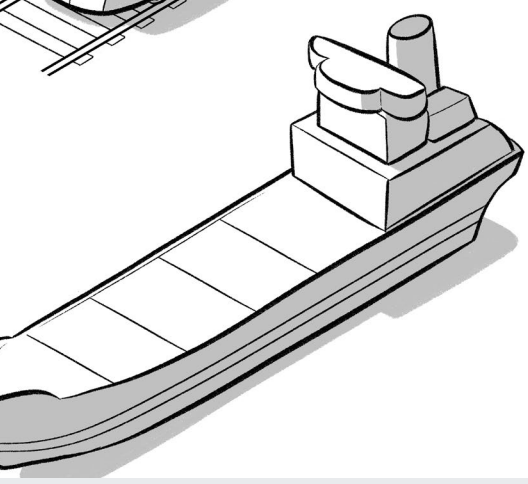
Developers familiar with Docker may know that Docker's fundamental components are open-source. This flexibility gives teams the option to create their own "DIY" implementations using Docker Engine without relying on Docker Desktop. While specific use-cases may require a custom build, the DIY route brings its own set of challenges.

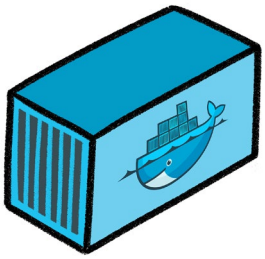
Docker Desktop is the quickest method of gaining access to all of Docker's features and creating collaborative container environments. It's ready to install in Windows and Mac environments. Though a DIY implementation is possible, Docker Desktop makes it simple to set up local development environments with minimal effort, reducing overhead costs and time spent configuring individual developers' environments to work with a DIY solution.

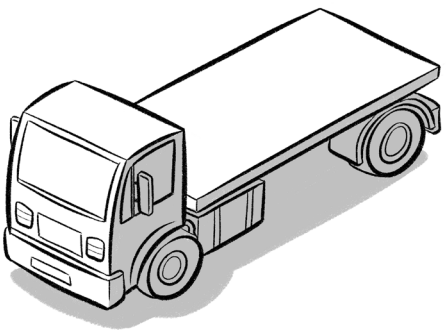
Docker Desktop also handles maintenance tasks, such as regular patches and updates issued by Docker, allowing developers to keep their attention where it's most needed. Docker Desktop provides a shortcut for saving substantial engineering and development expenses. Organizations avoid the costs and potential technical pitfalls of creating and operating a DIY in-house implementation.

With Docker Desktop, developers and other teams are free to dive directly into the essential work of creating and improving their software. Rather than worrying about building their own complex container environment and ensuring their home-spun solution is successfully running internally, developers can dive right into the heart of addressing the issues most important to the business. Docker Desktop helps organizations maximize their technology investment's ROI by freeing skilled developers' time to design the applications of tomorrow.

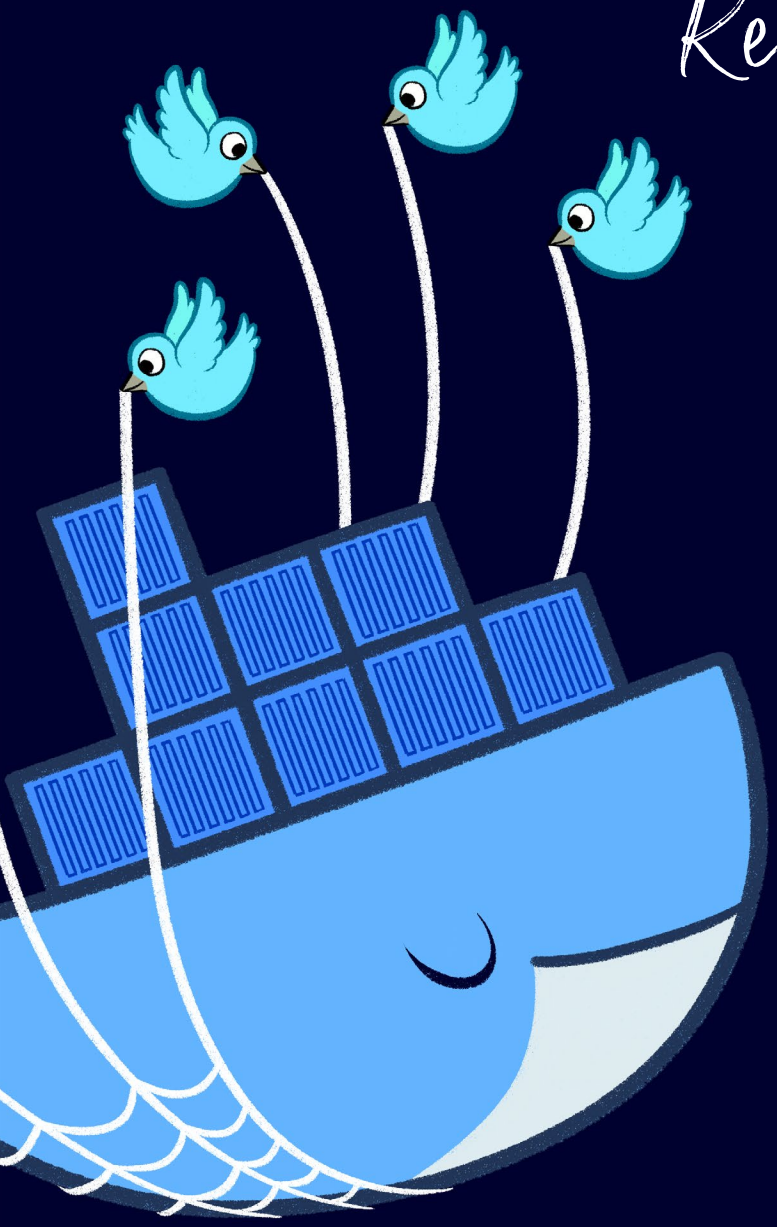








Read more about considerations for evaluating Docker Desktop alternatives



Ready to Try Docker Business?

Whether your organization is just beginning to grow or it's already operating at scale, Docker Business provides the tools and services you need to build modern and secure applications.

Interested in learning what Docker Business can do for your organization?
Contact your sales representative today.

Get started today

Interested in learning what Docker Business can do for your organization?
Contact your sales representative today.

docker