



Unlocking the Container:

Enhancing Operational Performance through Containerization

2025

Executive summary

A container-first approach offers businesses a streamlined way to develop, deploy, and manage applications. By packaging applications and their dependencies into consistent, portable units, containers ensure that software runs the same way across different environments. This consistency speeds up development and deployment, making it easier to respond to changes.

From a top-down perspective, employing a container-first approach in software and application development provides businesses with:



Cost savings

Containerization reduces infrastructure costs by enabling efficient resource utilization and lowering operational overhead.



Productivity improvements

Developers can work more efficiently with consistent environments, reducing time spent on setup and troubleshooting.



Faster time to market

Containers accelerate development and deployment cycles, allowing businesses to bring products and updates to market more quickly.



Enhanced security

Containers offer improved security through isolation and effective vulnerability management, resulting in reliable software customers can trust.

Containerization can revolutionize software/application development and operations by providing a scalable, efficient, and secure framework. By adopting a container-first approach, businesses can achieve faster deployment cycles, reduce costs, and maintain a competitive edge in the market. In this white paper, we'll explore how containerization addresses common business challenges and how to get started.



Navigating the modern business labyrinth

Much like Theseus navigating the intricate labyrinth of ancient Greece, modern businesses find themselves in a complex and evolving maze of challenges. In today's fast-paced market, businesses face mounting pressure to innovate and stay ahead of competitors while managing costs and complexity. [A 2024 survey performed by LBMC](#) highlighted that 70% of business executives expect national economic conditions to worsen in the first half of the year, driving concerns around inflation and cash flow. Additionally, [another survey performed by ITS Managed IT Services](#) found that nearly 50% of businesses have already raised their prices in response to rising costs, which reflects the broader economic pressures on businesses to maintain profitability while facing intense competition and rapidly evolving market conditions.

Traditional software development and deployment methods often fall short in this dynamic environment. They can be too rigid to adapt quickly to technological changes, leading to delays in integrating new tools and frameworks. This slow pace can hinder a company's ability to release new features and updates, impacting its competitive edge.

70%

of business executives expect **national economic conditions to worsen** in the first half of the year

50%

of businesses have already **raised their prices** in response to rising costs



Rising costs and complexity of digital transformation

Digital transformation has provided some tools to address these obstacles, but organizations now face new software development and deployment challenges. One significant challenge is that the [rising cost of cloud services](#) is projected to surpass \$1 trillion globally by 2024, with a compound annual growth rate of 16%. This surge in costs underscores the financial and operational challenges that businesses face as they integrate modern technologies into their operations.

With the growing complexity of modern applications — often involving microservices and diverse technologies — conventional deployment methods can become cumbersome and error-prone. Companies relying heavily on these outdated processes frequently encounter persistent issues, including costly delays caused by inconsistencies between development, testing, and production environments. These challenges and rising costs underscore the urgent need for more agile and cost-effective solutions.

Growing demand for data security and compliance

Additionally, consumers are more concerned about how their data is handled, and companies that emphasize security and compliance can enhance their reputation and maintain long-term customer relationships. [71% of consumers](#) say they would stop doing business with a company if it shared their sensitive data without permission. As data security and regulatory compliance become increasingly critical, companies must prioritize these aspects in their app development processes. Protecting sensitive information is essential to prevent data breaches and safeguard user and business data from unauthorized access. Compliance with stricter regulations, such as GDPR and CCPA, is necessary to avoid legal penalties and build customer trust.

Integrating robust security and compliance measures not only helps avoid operational disruptions that could impact business continuity and customer satisfaction, but it also protects intellectual property and ensures applications are adaptable to future regulatory changes and emerging security threats. By addressing these concerns early in app development, businesses can mitigate risks, meet legal requirements, and stay competitive in a rapidly evolving market.

As organizations increasingly adopt cloud or hybrid cloud environments, they require solutions that offer portability and seamless integration across various platforms. Containers offer the agility, scalability, and security necessary to meet these evolving demands, paving the way for more effective and resilient software solutions.



72% of customers

say they would stop doing business with a company if it **shared their sensitive data** without permission

\$1 trillion globally

The rising cost of cloud services is projected to surpass \$1 trillion globally by 2024, with a compound annual **growth rate of 16%**

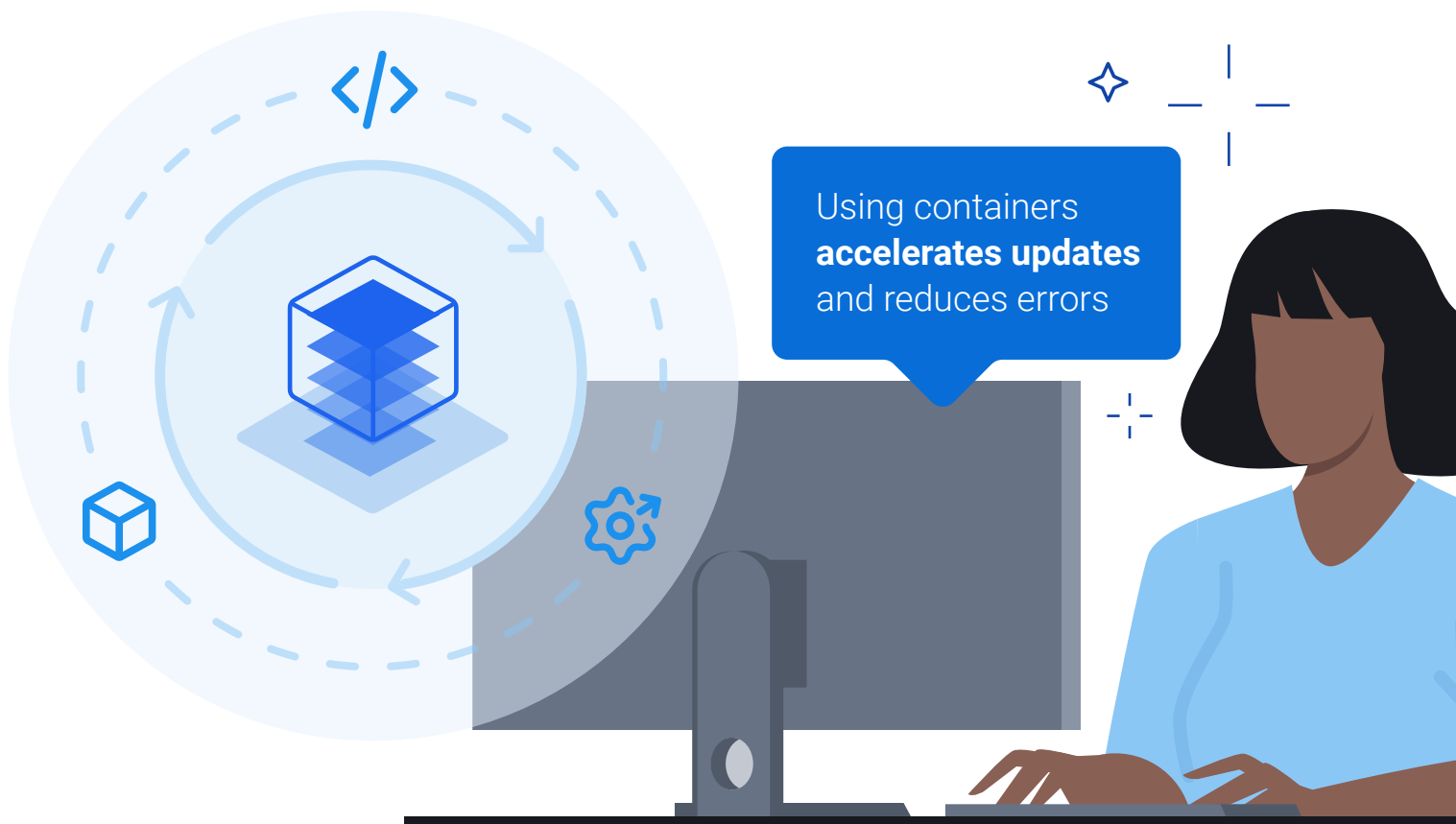


Escaping the rat maze

A container-first approach is a method of developing and managing software where containers are the primary tool for packaging and running applications. Imagine containers as self-contained boxes that include everything an application needs — its code, libraries, and configurations — so it can run smoothly regardless of where it's deployed.

Just as shipping containers revolutionized logistics by standardizing the way goods are packed, transported, and handled across different modes of transport — software containers standardize how applications are packaged and run. This standardization **ensures applications perform consistently across various environments, from development to production.** This consistency simplifies deployment, reduces errors, and accelerates updates, much like how shipping containers simplify and streamline global trade.

In simple terms, a container-first approach means that instead of dealing with complex and inconsistent environments for different stages of development, you start by creating these self-contained packages. This ensures that the application works the same way no matter where it's deployed — whether on a developer's computer, a testing server, or in production.



Rising above the mess

By focusing on containers from the beginning, businesses can organize the mess, streamlining the development process to reduce errors, save on costs, and improve the speed and flexibility of deploying new features and updates. This approach offers several business benefits that can enhance operations, such as **consistency across environments, better efficiency, enhanced security, and faster time to market.**

Containers package an application and its dependencies, ensuring it behaves identically in development, testing, and production. This uniformity eliminates the “it works on my machine” problem, reducing deployment issues and streamlining the software development process.

This, in turn, helps foster efficiency by streamlining operations, simplifying the deployment process, and optimizing resource utilization. Since containers are lightweight and share the same operating system kernel, businesses can run multiple applications on the same hardware without over-provisioning resources. This leads to reduced infrastructure costs and improved scalability, allowing organizations to adjust resources based on demand without unnecessary expenditure.

Achieving more with less

Imagine running a busy kitchen in a restaurant. Traditional software deployment is like having a separate kitchen for every dish you want to prepare. Each kitchen needs its own equipment, ingredients, and space, leading to wasted resources and higher costs. You might end up with some kitchens underused while others are overloaded.

Containerization, on the other hand, is like using a set of modular cooking stations in one big kitchen. Each station is designed to handle different dishes, but they all share the same equipment and workspace efficiently. This way, you can prepare multiple dishes at once without needing extra kitchens or equipment. As demand changes, you can easily adjust the number of cooking stations you use without building new kitchens from scratch. This approach saves on costs, makes better use of resources, and helps the kitchen run smoothly and scale as needed.



Additionally, by isolating applications within their own containers, potential security vulnerabilities are contained and managed more effectively. This isolation helps prevent breaches from spreading across different applications, thereby strengthening overall risk management and protecting sensitive data.

With streamlined deployment processes and the ability to quickly move applications across different environments, businesses can accelerate the release of new features and updates. This rapid deployment capability helps companies stay competitive and responsive to market demands, ensuring they can adapt and innovate quickly.



Playing to your advantage

By packaging applications and their dependencies into standardized units, containers offer a host of strategic advantages that can impact a company's bottom line and operational agility. From lowering infrastructure costs and boosting developer productivity to streamlining deployment processes, **containerization provides a comprehensive solution for modernizing software development and deployment**. Understanding these benefits can help organizations leverage containerization to stay competitive and respond swiftly to market demands.

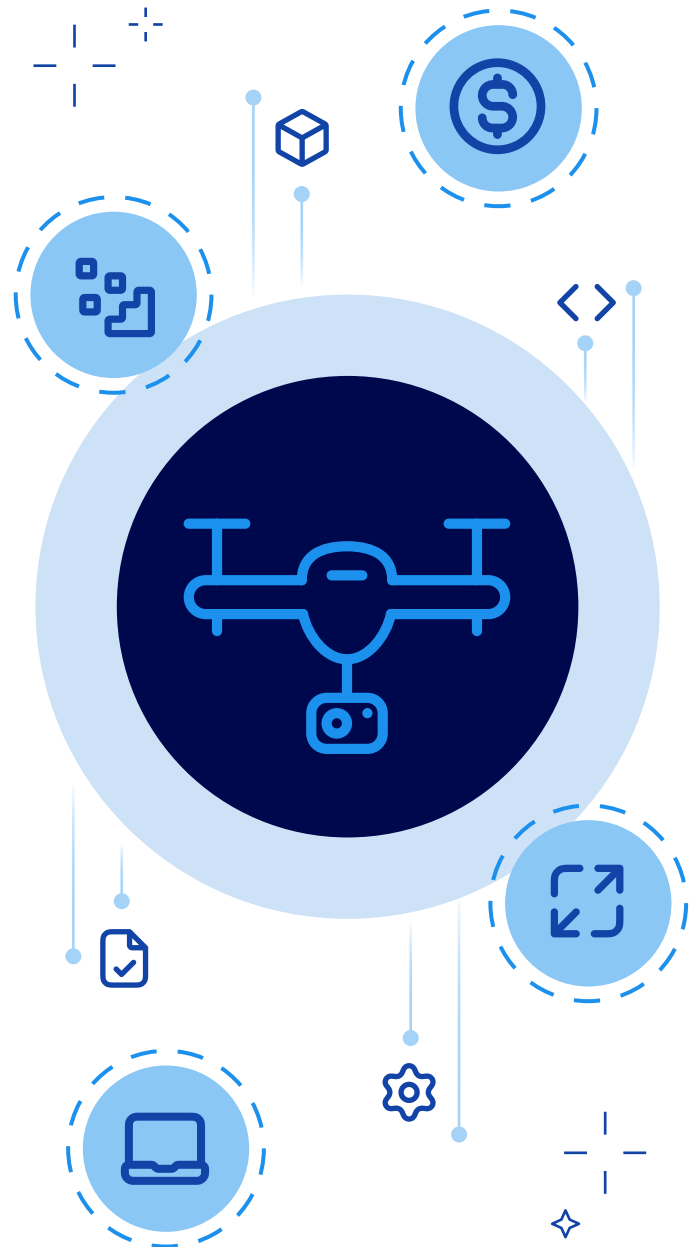
Cost reduction

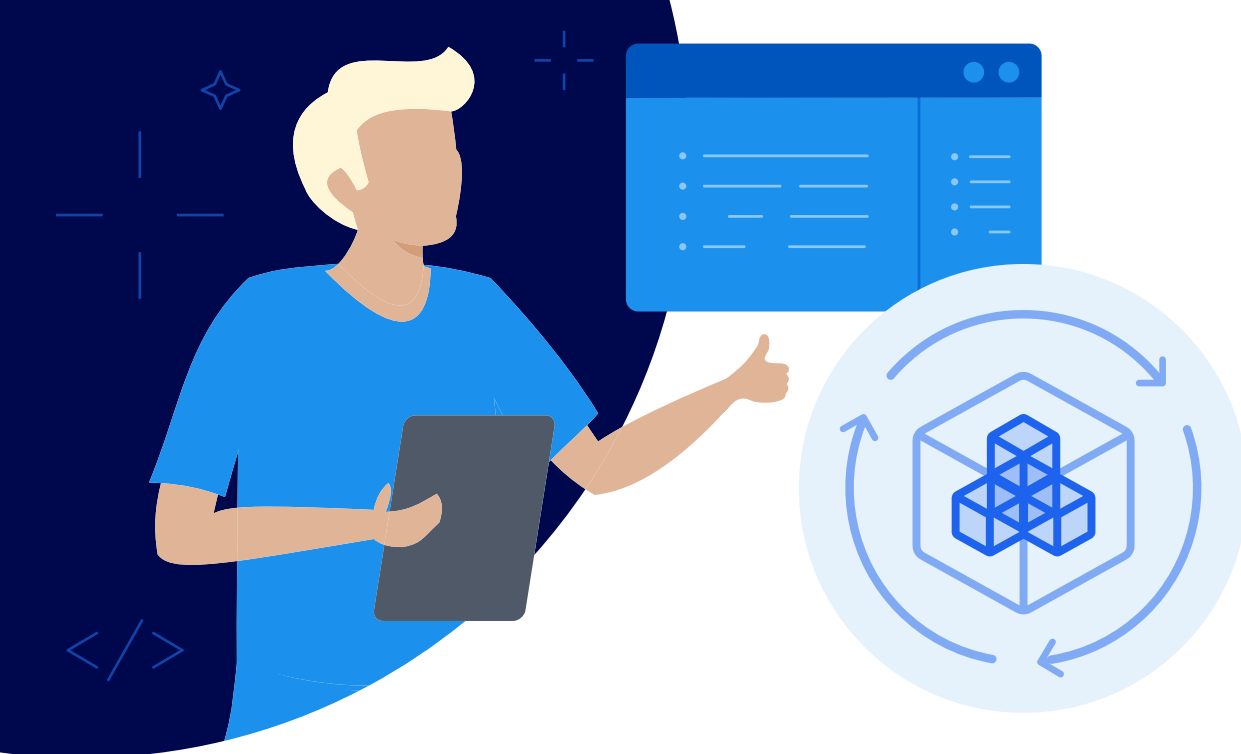
Containerization offers cost benefits by optimizing resource utilization and reducing infrastructure expenses. Traditional software deployment often involves running multiple virtual machines, each requiring its own operating system and resources. This can lead to over-provisioning and underutilization of hardware, driving up costs. In contrast, containers are lightweight and share the same operating system kernel, allowing multiple containers to run on a single host.

Think of traditional software deployment like managing a fleet of delivery trucks, where each truck represents a virtual machine. Each truck requires its own fuel, maintenance, and parking space, even if some of them are only partially loaded. This results in wasted resources and higher costs because you're paying for more capacity than you actually use.

By contrast, containerization is like switching to a more efficient delivery system, such as a network of delivery drones. These drones are much lighter and more flexible, sharing the same charging station and operating system, which means they can be used more efficiently and economically. Just as drones optimize delivery by reducing the need for extra infrastructure and maintenance, containers optimize resource utilization by running multiple applications on a single host without the overhead of separate operating systems. This shift cuts down on costs and maximizes the use of available resources.

This efficient use of resources means **businesses can maximize the performance of their existing infrastructure and reduce the need for additional hardware**. Additionally, containers enable better scalability, as they allow for dynamic allocation of resources based on demand, further cutting down costs associated with over-provisioning while maintaining excess capacity.





Enhanced developer experience

Containerization transforms the developer experience by providing a unified and consistent environment for application development. Unlike traditional methods that often result in discrepancies between development and production environments, containers ensure the application runs the same way everywhere. This consistency reduces the time and effort required to troubleshoot and resolve issues.

Containers streamline the onboarding process for new developers. Since each container includes all necessary dependencies and configurations, new team members can quickly set up their development environments and start contributing without lengthy setup procedures. This efficient setup boosts productivity and enhances overall job satisfaction by letting developers focus on the code instead of dealing with complex configuration issues. With an improved experience, developers are more effective and efficient: they ultimately just enjoy the work more. Happy developers mean better quality work.

Optimized deployment

Containerization fundamentally changes the deployment process by packaging applications and their components into single, self-contained units. This packaging makes it easier to deploy applications consistently across various stages, from development to production, without the usual environment-related hurdles.

Developers can also use containers to enhance deployment pipelines to make them more predictable and less prone to errors. With container orchestration tools handling tasks like scaling and load balancing, the deployment process becomes more automated and efficient. This efficiency translates into faster releases of new features and updates, allowing businesses to quickly adapt to market demands and maintain a competitive edge.

Embracing containerization equips organizations with the agility and efficiency needed to thrive in today's competitive landscape. Take a look at this example with CARIAD to see how containerization is empowering them to respond swiftly to changing market demands and deliver high-quality solutions with greater ease and cost-effectiveness.



Case study

Building a secure and compliant software framework at CARIAD with Docker

CARIAD, the automotive software division of the Volkswagen Group, is focused on advancing automotive mobility through innovative software solutions that enhance safety, sustainability, and comfort across VW Group brands. The company provides a sophisticated software platform that supports constant updates, advanced vehicle functions, and an extensive digital ecosystem suitable for various car models. To address the complexities of automotive software development, CARIAD has partnered with Docker to implement a containerized development environment. This move is aimed at streamlining development processes, ensuring compatibility across diverse hardware, and boosting overall efficiency and innovation.

CARIAD's rapid expansion and integration of various legacy systems created challenges, including operational fragmentation and the need for cross-platform compatibility. To overcome these obstacles, CARIAD adopted Docker containerization technology to unify development environments and enhance operational efficiency. By integrating Docker Desktop, which supports both Windows and Linux platforms through a hybrid architecture, CARIAD has improved consistency in software deployment and reduced compatibility issues. This strategic shift not only streamlined development workflows but also enhanced security, compliance, and scalability, positioning CARIAD to effectively manage and innovate within its extensive software portfolio, ultimately accelerating its software delivery process.

Read the full story to see how [CARIAD and Docker are revolutionizing automotive software development](#).



C A R I A D



Sprinting ahead of the pack

Now that we've seen how CARIAD used Docker to streamline their software development and tackle complex challenges, let's shift our focus to some broader industry insights. Docker commissioned Forrester Consulting to conduct a [Total Economic Impact™ \(TEI\) study](#) to assess the potential return on investment (ROI) enterprises could achieve by deploying Docker Business. The study aims to provide a framework for evaluating the financial impact of Docker Business on organizations. To gather insights, Forrester interviewed five representatives experienced with Docker, aggregating their experiences into a composite organization – a global technology manufacturing company with \$25 billion in annual revenue and over 5,000 software developers.

Prior to adopting Docker, these organizations struggled with inefficient legacy systems and costly IT infrastructures reliant on thousands of virtual machines (VMs). These challenges resulted in high expenses for data center space, hardware, maintenance, and energy, and slowed application development and deployment, which hindered agility and competitiveness.

By shifting to Docker, the organizations significantly reduced their data center footprint and enhanced application delivery speeds. Docker containers allowed for greater server density, cost savings, and improved efficiency, enabling faster development cycles and continuous delivery. This shift not only accelerated time to market and improved response to customer feedback but also allowed the organizations to reinvest savings into innovation and talent, thereby positioning themselves more competitively in a dynamic market.



Key statistics

126%

Return on investment (ROI)

\$66.9M

In net present value (NPV)

3 mo.

Improved time to market for revenue-generating applications

6%

Application developer efficiency increase

3x

Reduction in servers due to improved VM density



Transforming your business with containerization

Navigating today's fast-paced market can feel like maneuvering through a complex labyrinth, with businesses facing pressure to innovate while managing costs and complexity. Traditional software development methods often fall short, proving too rigid to adapt swiftly to technological advancements, causing delays in feature integration and impacting competitiveness. As applications become more complex and involve diverse technologies, conventional deployment methods can lead to costly delays and inefficiencies. With increasing concerns about data security and regulatory compliance, companies must prioritize robust measures to build trust and avoid legal penalties.

Docker containers offer the agility, scalability, and security needed to address these challenges, providing a more resilient and effective solution for modern software development and deployment. By adopting Docker, you'll be well on your way to overcoming some of your business challenges and staying competitive in a rapidly evolving market.

Begin your containerization journey today

Unlock unparalleled operational benefits and secure a competitive edge.

Let's get to know each other



Appendix

Glossary

Containerization

The process of packaging an application and its dependencies into a single, portable unit called a container.

CI/CD Pipeline

Continuous Integration and Continuous Deployment pipeline, a method to automate the process of testing and deploying code.

Microservices

An architectural style that structures an application as a collection of small, autonomous services modeled around a business domain.

Kubernetes

An open source platform used for automating the deployment, scaling, and operation of application containers.

Docker

A platform used to develop, ship, and run applications inside containers.

Orchestration tools

Software used to manage and coordinate the deployment, scaling, and operations of containers.

Additional Resources

[Docker Blog](#) | [Pricing and Subscriptions](#) | [Docker Resources](#) | [Contact us](#)

