Modernize Traditional Applications with Docker Enterprise Edition

Modernize Without Changing Source Code

Enterprises are looking to cloud and microservices to drive agility and digital transformation. These innovations are often applied to new projects but the question remains "but what about the other 99% of my apps that are not microservices?" The overwhelming majority of an application total cost of ownership (TCO) is incurred after the initial deployment meaning IT ops teams bear the brunt of the administration and maintenance burden for each application. As enterprises look to become more agile, strategies need to be implemented to optimize costs of the existing applications to free up time and resources to reinvest into refactoring and innovation.

Docker Enterprise Edition (EE) enables IT ops to modernize traditional applications with the leading container platform without requiring modifications to the source code. By simply containerizing the application without adjusting the source code, legacy applications begin the journey to modernization with hybrid cloud portability, increased security and cost efficiency.

Portable
Migrate across hybrid cloud infrastructure

Secure
Reduce surface area and increase visibility

Efficient
Optimize CapEx and OpEx costs

Benefit: Hybrid Cloud Portability

Docker packages application code and dependencies together into lightweight, stand-alone containers. This eliminates the "works on my machine" problem so that the containers can run in a new environment without issue regardless of the different configurations and dependencies between environments. Once packaged, the container can easily be deployed to any environment with a single Docker command. Quickly enable cloud migration, accelerate tech refresh cycles or burst to the cloud.

Benefit: Increase App Security

Simply packaging legacy applications into Docker containers allows them to inherit the built-in security capabilities of Docker EE without any changing the source code. Docker provides the strongest isolation properties and comes with strong default settings that can be configured to be more limited, if needed for the application. With these features, IT admins can greatly reduce the attack surface area of older applications and provide the minimal amount to host resources required to operate and nothing more.

Additionally, Docker EE provides a secure supply chain for containers to be created, scanned, signed, shared and deployed. Security Scanning provides a deep visibility with a Bill of Materials (BOM) of all the packages and version numbers included in the application and their current vulnerability status. If a CVE be reported in the future, the Docker admin is notified of the change for quick remediation. Containers can also be digitally signed and Docker clusters enabled for verification to guarantee safe transport of applications across the network from one infrastructure to another. Combined these make existing apps safer simply by leveraging Docker EE.
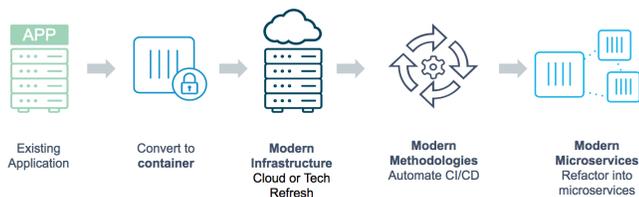
Benefit: CapEx and OpeEx Efficiency

Over 90% of an application's total cost of ownership (TCO) is incurred after the initial deployment. From ongoing maintenance, deployments, support and scaling, these tasks add up to thousands of hours per year per application multiplied by hundreds to thousands

of applications. Modernizing legacy applications with Docker EE allows IT ops teams to dramatically streamline the operational tasks like provisioning, deployment and updates. On average, Docker customers experience up to 75% time savings with provisioning and deployment and overall 10X reduction in total hours spent maintaining and supporting a single application.

Architecturally, Docker containers are lightweight and share the kernel of the underlying operating system meaning they use minimal resources and can increase the number of application instances on a single server. Increased workload density can reduce hardware, cloud and software licensing costs.

How to Get Started

The modernization approach outlines logical steps to gain efficiencies and benefits along the way.

Existing Application → Convert to **container** → **Modern Infrastructure** Cloud or Tech Refresh → **Modern Methodologies** Automate CI/CD → **Modern Microservices** Refactor into microservices

1. Port application to a container and once validated consider where to deploy the application.
2. Migration to cloud or server refresh is a good consideration for this stage.
3. Once deployed and confirmed, consider integrating the container into systems that automate build, integration testing and deployment to further optimize operations and save time.
4. The last stage considers the path to microservices. The application itself can be broken up or have modern services added to the existing container.

Docker EE allows for the flexibility for the application teams to decide how much to modernize.

The Journey to Containers as a Service

The path to a modern and agile IT environment begins with the applications and infrastructure you already have. Containerizing legacy apps, deploying and managing them with Docker EE is the first step to DevOps, microservices and cloud.
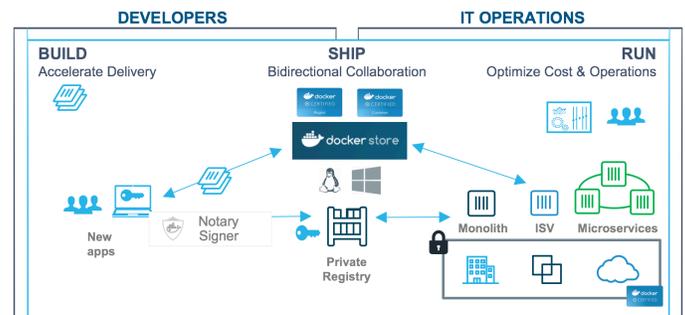
Docker EE provides a bridge to bring forward legacy apps to modern infrastructure, to integrate with modern automation and a flexible architecture to further modernize the application into microservices. Docker EE is a unified platform that brings together legacy, microservices and ISV applications into a standardized packaging format with common APIs and tooling.

www.docker.com/MTA