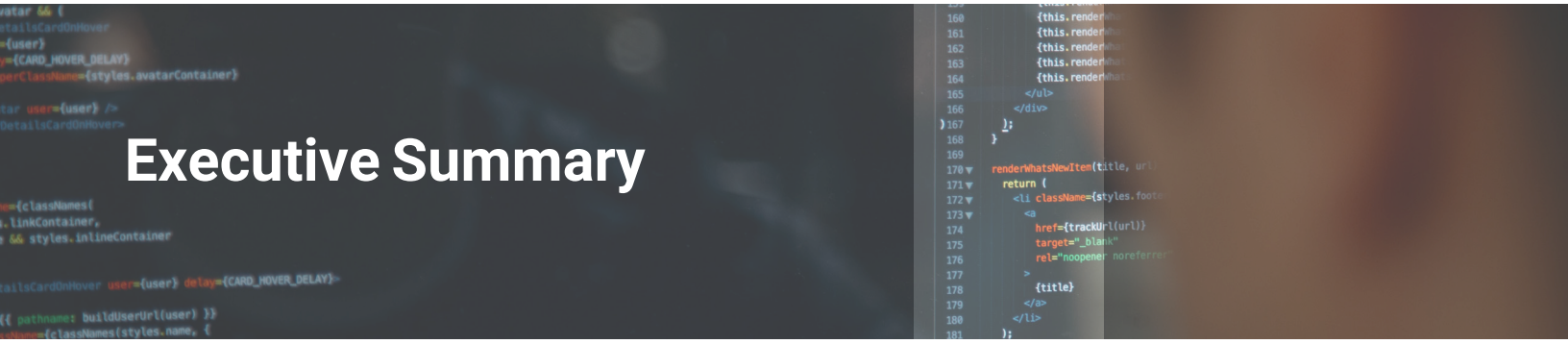
Enabling Embedded Solutions with Container Technology

Bring Portability, Security, and Manageability to Embedded Software Development and Distribution

WNDRVR



Executive Summary

Monumental changes to embedded software development practices are underway to address the complexities of ongoing software lifecycle maintenance and critical security concerns. These challenges have captured the attention of many industry leaders who strive to enable and secure streamlined development and deployment methods.

Complementary technologies have spurred these changes. Such technologies have been driven by the growing adoption of edge computing, autonomous vehicles, medical devices for remote diagnosis and treatment, robotics, aerospace advances, and increasing 5G technology demands as mobile broadband and mmWave microcell installations proliferate.

Adopting cloud-native tools for development empowers teams to collaborate from around the world. Containers make this possible by sharing a common environment with reusable configurations for developing and deploying code. One of the advantages of using containers is that they can be adopted for both existing embedded applications and for new designs. Application developers, whether creating embedded or enterprise-focused programs, can deploy software written in Rust and Python using tools familiar to them. Think of it as a write-once, deploy-anywhere approach.

Real-time operating systems (RTOSes), VxWorks® in particular, are fundamental to embedded systems. The introduction of support for containers in VxWorks to systematically deliver and update software is transformational. Support for containers that can respond dynamically in a real-time, deterministic way to events in the environment is a critical and innovative element for creating a software-defined world. This is especially important for automated manufacturing lines, autonomous vehicle operation, aerospace applications, and medical devices, as digital transformation advances in these sectors.

Adoption has lagged to some degree due to a lack of awareness in the embedded industry of the benefits of container technology. Also, the consolidation of existing solutions around virtualization (virtual machines) implementation as standard has impeded adoption. Another impediment is concern over security issues related to a shared kernel causing a generalized breach among those applications connected to the same host. Finally, adoption has been hampered by a lack of skilled technicians and developers with the tools and expertise to implement container solutions for embedded use cases.

As discussed in this paper, these issues have been addressed in numerous ways, and the multiple benefits of containers enabling innovative embedded applications offer a promising future for the evolving software architecture that supports their creation and distribution.

Demystifying Container Technology

Containerization enables the creation of a standardized bundle of software components — including a collection of all required configuration files, libraries, and utilities — allowing an application to run in a specified environment. Containers for Linux and VxWorks can be confidently deployed across different hardware environments and kernel versions. Containers stay lightweight and manageable by sharing the kernel of the operating system. This simplifies management and updating of code, since the latest iteration of the operating system doesn't need to be distributed each time a container is deployed into a system.

Container Formats and Orchestration

Docker popularized the use of containers with tooling that made it easy and simple for anyone to build and distribute images. Released as open source in 2013, the original creators worked with the community to standardize the specifications based on their work.

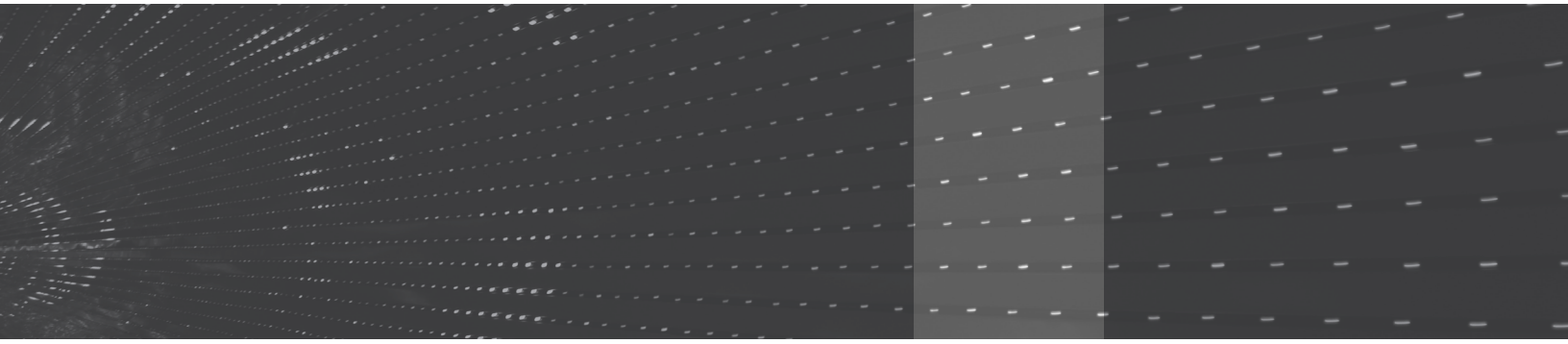
Standardization has advanced container adoption, expanding interoperability across multiple architectures. The Open Container Initiative (OCI) lists three specifications for container developers to follow:

1. **Image specification:** Defines the container images, which are basically file system bundles stored in a registry from which they can be retrieved by a host
2. **Distribution specification:** Provides a means to locate images in the registery and download them
3. **Runtime specification:** Establishes the rules for unpacking the image contents, the file system bundle that will be used by the container when running

VxWorks supports these specifications, enabling use of common formats and tools for developing and deploying containers. VxWorks containerized workloads can even be controlled by Kubernetes to orchestrate containers across devices.

Through Wind River® Studio, orchestration of containers can be extended to manage a large fleet of devices running VxWorks. For example, software upgrades of applications in VxWorks can be configured within Kubernetes and applied automatically.

> "Container technology is opening the world to you, while enforcing security and enabling reuse of components. Containers also foster adoption of DevSecOps and make your software modular and flexible."
>
> 
>
> **—Nicolas Chaillan,** *founder, Ask Sage; former U.S. Air Force and Space Force Chief Software Officer*



What Is Inside a Container?

Typically, containers are associated with deploying microservices in the cloud. However, containers can also be used for deploying traditional services and applications. Wind River takes this one step further by extending support for containerizing embedded software on VxWorks.

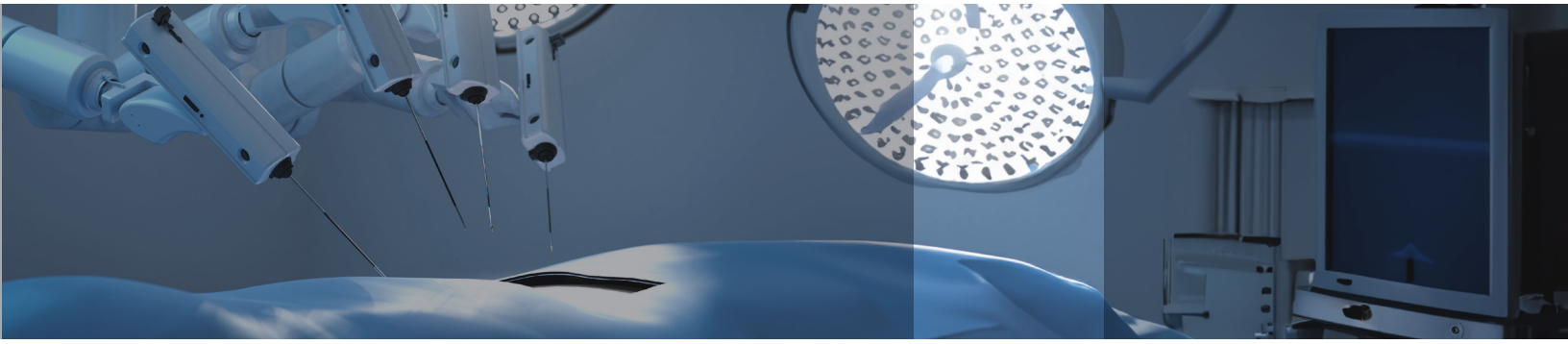
By definition, containers consolidate the vital elements of a set of modular, interlinked applications into a cohesive solution. The concept is based on building large software applications from many smaller, independent blocks, as shown in Figure 1 (page 5). This approach takes advantage of proven open source tools, frameworks, and software to speed software development time and reduce costs. It also breaks free of the legacy practice of building monolithic, inflexible software applications that are difficult to create and update. Instead, this modular approach favors portability, lightweight operation, and agility. These attributes are essential for supporting modern architectures, including cloud-native and service-oriented architecture as used in automotive designs. Wind River supports containers in both VxWorks and Wind River Linux to build next-generation architectures based on cloud-native principles across multiple industries.

Where Are Containers Being Used in Different Markets?

Containers are ideally suited for applications in many vertical markets in which embedded software plays a significant role, including:

- **Avionics:** For commercial and military avionics companies looking to optimize space, weight, and power (SWaP), containerization is transformative. Being able to run applications in their own containers, independent of the underlying stack, enables greater portability and even reusability of legacy software packaged as containers deployed on a newer system. Examples of prime use cases are flight data analysis, flight management systems, 3D cockpit displays, user interfaces, in-flight entertainment systems, and aircraft system monitoring.
- **Automotive:** Automotive applications for container technology are among the most promising developments in the field, particularly as they support autonomous driving and software-defined vehicles. This new approach virtualizes vehicle equipment and consolidates various functions on a single hardware system. Due to modularity

> "Containerization brings three things to the industry that are desperately needed. First, it allows you to manage software, which today is monolithic. Secondly, it lowers development costs. The third thing is that containers unlock new monetization models with software-related revenues."
>
> 
>
> **—Glen De Vos,**
> *Senior VP of Transformation and Special Projects, Aptiv*



and compatibility, the concept of containers is used for updating and upgrading features over the air both faster and more securely. Containerized workloads can also support other use cases, such as in-vehicle infotainment, connected car services, real-time diagnostic alerts, predictive maintenance, fleet management, analytics, and telematics data.

- **Industrial:** Transformation is underway as Industry 4.0 changes are sweeping through markets. By running secure, isolated containers on single or distributed systems, industrial applications can be consolidated into fewer systems, improving scalability, simplifying deployment, and enhancing system update capabilities. It also ensures update- and upgrade-ready forward-looking architectures including predictive maintenance, AI-enabled robots and cobots, manufacturing-line automation, supply chain operations and logistics, augmented reality, virtual reality, and control systems.
- **Telecommunications:** Containers enable top service providers to deploy and securely update 5G and upcoming 6G networks. A world-leading telco equipment manufacturer is implementing initiatives using containers for 5G network slicing to support applications with differing performance requirements, such as augmented reality and virtual reality. 5G base stations equipped with mmWave capabilities use containers to distribute software in smart city environments and to ensure that current software patches are installed. Network utilities such as firewalls and load balancers can also be implemented through containers.
- **Medical:** In a highly regulated environment where device high availability is a matter of life and death, containerized workloads on underlying safety-certified operating systems enable critical environment separation of applications and data. Use cases enhanced through containers include remote patient care, health monitoring, imaging systems, infusion pumps, and surgical and robotic systems. These applications can utilize containers to deliver security patches and critical software updates. Surgical robots are being developed that are fully automated and can perform complete operations without human intervention.

As container technology matures, potential use cases are essentially unlimited.

Isolation of Applications Increases Security and Reliability

Containers isolate applications and their runtime operations for security, conflict avoidance, and management control. Common isolation techniques include the use of namespaces to control access to system resources and capabilities and to place limits on how much can be allocated. Containers sometimes use an overlay file system to share common files but keep any local changes private.

VxWorks has granular control over how its real-time processes (RTPs) can access kernel objects, view files and directories, and interact with kernel resources. By restricting access to system calls, applications can be isolated from each other and given a private sandbox to run in. The file system namespace provides a unique view of the file system to allow containerized applications to manage their libraries and configuration independently. Its overlay file system allows reduced footprint size through reuse of containers with a common base image without interfering with each other. The extensive network functionality in VxWorks — stemming from its history in telecommunications — also makes it possible to control the network interface provided to applications and the endpoints they can communicate with.

Containers provide additional hardening, particularly when features are combined with other security provisions. This can be valuable in embedded environments where container security is essential. For example, a secure boot technology establishes a chain of trust by validating the full range of software components, from the hardware root of trust through the bootloader and kernel, right down to the signed container and the application itself. Combining container security with secure boot provides an end-to-end chain of trust for software running on the device.

Integration of Mixed-Criticality Components

In some cases, containers deploy software developed with higher-level programming languages or open source software that cannot be certified. This results in containers with varying levels of criticality. Successful integration of these components, identifying certifications that have been met or compliance that has been granted, can help avoid the need to redo certain certification processes. To take full advantage of agile development practices and DevSecOps workflows, isolating components according to the level of criticality can retain the advantage of streamlined development while maintaining existing certifications for specialized components.

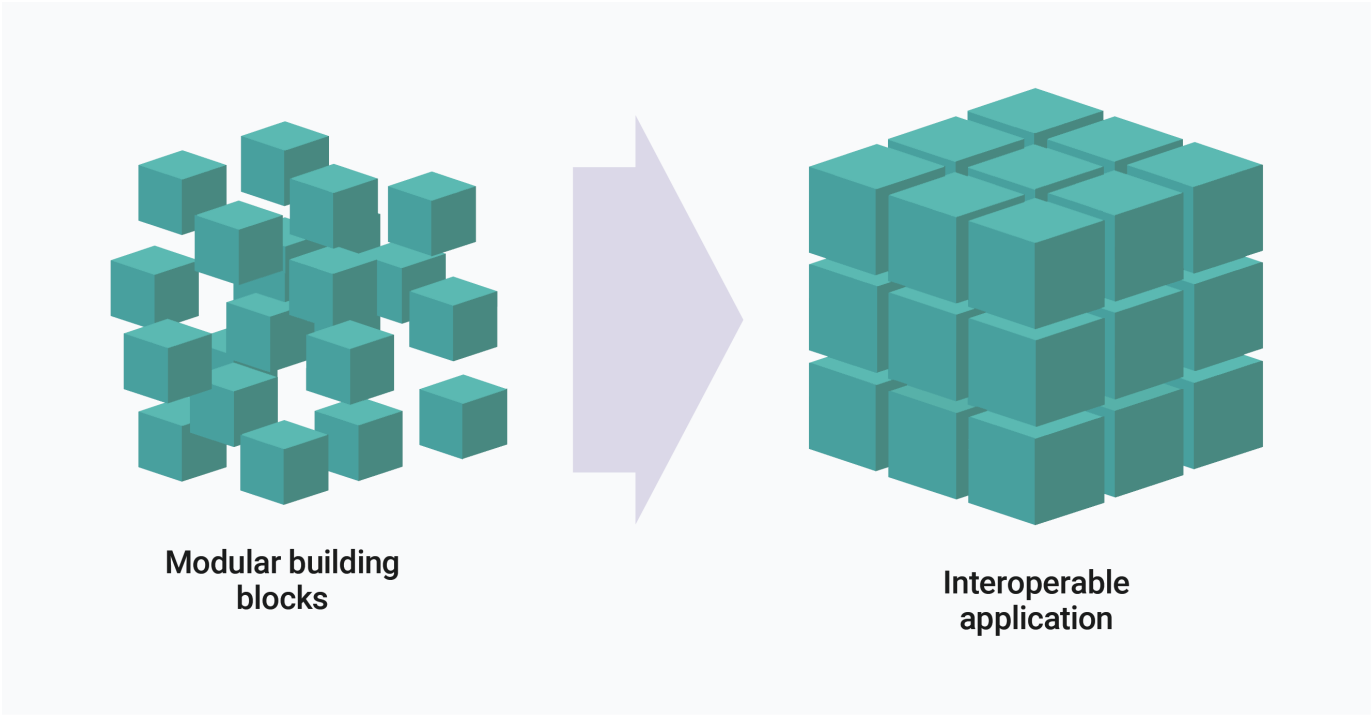
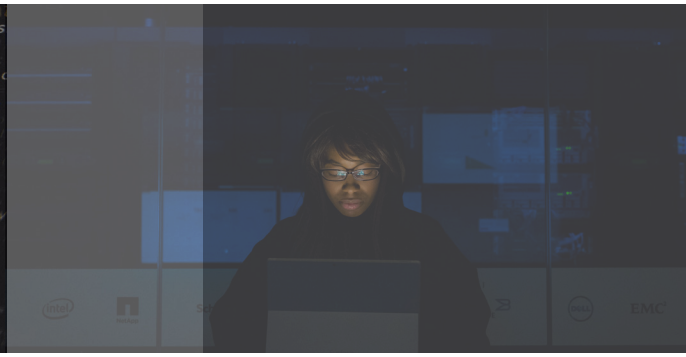


Figure 1. Modular building blocks make it easier to rapidly deliver advanced capabilities

Embedded Deployment of Applications



The challenges of effectively deploying embedded applications that comply with the rigors of vehicular, medical, or edge computing use cases are well suited to container technology. Rob Woolley, principal technologist at Wind River, summarized the benefits in his paper, "Deploying Embedded Applications Faster with Containers." He notes, "Embedded developers can benefit from the infrastructure-agnostic, scalable execution environment enabled by containers. Imagine a design process — from development to test to deployment to production to management — in which developers can share resources, pipelines, and results across the team. Instead of being limited by the number of development boards available, companies could exploit the elasticity of the cloud to set up multiple instances of a system on demand."

Embedded use cases often require low-latency, responsive, deterministic behavior. VxWorks has been engineered to provide an RTOS for these types of deployments, with OCI-compliant container support for enabling small-footprint embedded solutions. It has a long history of use in products that require rigorous security compliance with specialized certifications. VxWorks suits demanding use cases in industries such as medical, aerospace, transportation, and Industry 4.0 robotics and automation.

The charactertistics of containers are ideal for continuous integration and deployment (CI/CD), ready for producing new software updates quickly from new source code changes. Their modularity means they can be leveraged in a DevSecOps workflow with automated orchestration (as shown in Figure 2). Containers built for VxWorks are highly portable across environments that value security and frequent urgent software patches to address cybersecurity concerns.

- Compliant with OCI
  - Image format
  - Runtime specifications
- Runtime
  - Image parsing/validation
  - Instantiation of the container
  - Execution of the application
- Manager
  - Logic for pulling containers from registry
  - Command line tools for development/testing

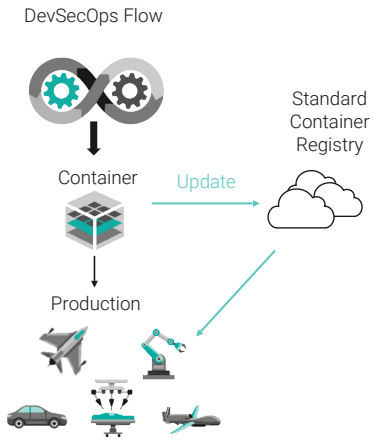


Figure 2. Creating and distributing VxWorks containers

Developing the CAPS Framework

To streamline software development and fend off potential security breaches and malware, Collins Aerospace developed the Containerized Application Platform System (CAPS). A recent whitepaper describes the approach in these terms:

"Based on an easy-to-use, mainstream technology stack, CAPS ensures reliable operations for a collection of containerized microservices in an embedded avionics environment where power is pulled after each flight and the system is never connected to an IT network for maintenance."

Developing and Testing Containers with Wind River Simulation

Wind River advanced simulation software offers an effective way for developers and system architects to test containers to identify and resolve software defects before deploying to the physical device. As the DevSecOps model becomes more widely adopted, containers can be engineered to minimize embedded solution issues, probing for potential problems at the earliest stages of development.

Simulation software can effectively detect and ensure that safety vulnerabilities don't go unnoticed and can be eliminated before a container goes into large-scale distribution. This simulation technology makes it possible to test effectively on virtual hardware, an important capability when exploring potential attack vectors that are difficult or impossible to check on a live system.

Creating a digital twin of a container-enabled system is increasingly important to long-term operation and upgrades. Nicolas Chaillan, founder of Ask Sage and former U.S. Air Force and Space Force chief software officer, commented, "A digital twin is essential when it comes to modeling and simulation. It empowers teams effectively to know exactly how the system is going to behave before bending metal. And a lot of people look at software agility and forget that you can do a lot of that. You don't have to always be in a waterfall universe in hardware. I think you can easily now swap hardware, swap compute with a rack or some type of easily replaceable hardware, compute, or storage capability — on jets, on ships, you name it."

> "A digital twin is essential when it comes to modeling and simulation. It empowers teams effectively to know exactly how the system is going to behave before bending metal. "
>
> —Nicolas Chaillan

Prospects for Container Technology

Rethinking the design of an existing embedded system to embrace a software-defined architecture is not a trivial pursuit. However, momentum is building throughout the industry to embrace containers as a logical evolution of modern development practices. Commenting on the direction in which design is proceeding toward a software-defined model at Aptiv, Senior VP of Transformation and Special Projects Glen De Vos said, "We went through all of these proofs of concept and studies with the automotive OEMs to [say], 'This is why this works.' We're now in the next phase, the software equivalent of that: We're going to show you smart software architecture with containerization and the DevOps environment, and that's the process for now. We think the math is very compelling. In fact, the technology has gone from being unsustainable to being profitable."

Container technology for embedded solutions is still in the early stages of adoption. However, it is part of an industry-wide transformation of how software is designed using modular components, built with DevSecOps practices for increased agility, security, and ongoing maintenance. The shift to cloud delivery of software patches and upgrades can, in the long term, save companies substantial cost and efficiently open new opportunities to reliably meet customer demands and industry requirements.

As awareness of the benefits of embedded container technology grows, and the containerization skills of software developers keep pace with this growth, the technology holds the promise of a bright future.

AI and Container Technology Are a Winning Combination

Container technology combined with artificial intelligence (AI) and machine learning facilitate powerful capabilities. Examples include:

- **Predictive maintenance:** Machine learning models within embedded solutions can assess the probability of parts failure, triggering timely maintenance procedures and preventing unexpected downtime. This is particularly important in the aerospace industry, in which safety and fail-safe operation are paramount.
- **Identifying and combating security threats:** The dangers of security breaches — including malware and unauthorized access to systems — can be lessened by employing AI to monitor the behavior of embedded applications and block intrusions.

> "When it comes to containerization in the future, one important point is that it is going to streamline the adoption of artificial intelligence and machine learning, which is dependent on containerization and the scalability of containers."
>
> —Nicolas Chaillana

- **Balancing resource use:** AI makes it possible to intelligently allocate resources within embedded container applications, creating an optimal balance of processor, memory, and storage resources as well as delivering dynamic scalability to meet application requirements.

Expect AI to provide value and benefits in a host of emerging technologies as containers are widely adopted, encompassing (but not limited to) autonomous vehicles, avionics, smart grid installations, smart city infrastructure, and intelligent agricultural equipment.

Conclusion

As the embedded industry tackles the challenges and complexities of software lifecycle maintenance and critical security concerns, support for containers in VxWorks provides a solution for regular, reliable, and large-scale deployment of updates.

The adoption of container technology in VxWorks opens the door for innovation by leveraging existing industry standards and tools for existing and next-generation designs. Container use provides modular software that can be reused across projects, resulting in cost savings and faster time-to-market.

Containers provide agility to contend with the accelerated pace of change in technologies, map to diverse use cases, and create business opportunities. Visit www.windriver.com/containers to learn more about how Wind River can help you adopt containers.

Additional Resources

What Are Embedded Containers? Discover the ways in which container technology is bridging the divide between enterprise and embedded systems.

Hype Cycle for Container Technology, 2023: Get insights from Gartner on the benefits of container technology to enable digital business strategies.

Container Technology Energizes Edge Computing: See how manufacturers, medical organizations, energy providers, aerospace firms, and others can take advantage of the container support included with VxWorks.

About Wind River

Wind River is a global leader in delivering software for mission-critical intelligent systems. For 40 years, the company has been an innovator and pioneer, powering billions of devices and systems that require the highest levels of security, safety, and reliability. Wind River software and expertise are accelerating digital transformation across industries, including automotive, aerospace, defense, industrial, medical, and telecommunications. The company offers a comprehensive portfolio supported by world-class professional services and support and a broad partner ecosystem. To learn more, visit Wind River at www.windriver.com.

Wind River is a global leader of software for the intelligent edge. Its technology has been powering the safest, most secure devices since 1981 and is in billions of products. Wind River is accelerating the intelligent transformation of mission-critical edge systems that demand the highest levels of security, safety, and reliability.