Orchestrating Applications at the Edge: CI/CD at the Edge with K3s and GitLab

Arm Project Cassini Reference Implementation

White Paper

Index

Executive Summary

To meet the growing need for advanced, scalable data processing at the edge of the network, Arm's Project Cassini brings a secure cloud-native experience to edge development, using Kubernetes and CI/CD systems to create a streamlined method for enabling application delivery, maintenance, and control to any number of edge and terminal devices.

The Edge is Where It's At

There is a fundamental shift underway at the edge of the network, with connected devices being asked to do more than ever before. More often than not, it takes too much time and energy to make data travel from an endpoint to the cloud and back, and that means more and more data is being analyzed, processed, and stored at the edge of the network.

These changes in the way data is generated and managed are giving rise to new approaches, in hardware and software, designed to deliver what have traditionally been cloud computing capabilities to edge nodes and terminals.

> More of what used to happen only in the cloud now needs to happen at the edge.

Operating on the Edge

Developing for this new, edge-intensive way of operating requires an updated view of how tasks are divided between the cloud, the edge, and the endpoint, and means taking into account the operating requirements that are unique to the edge.

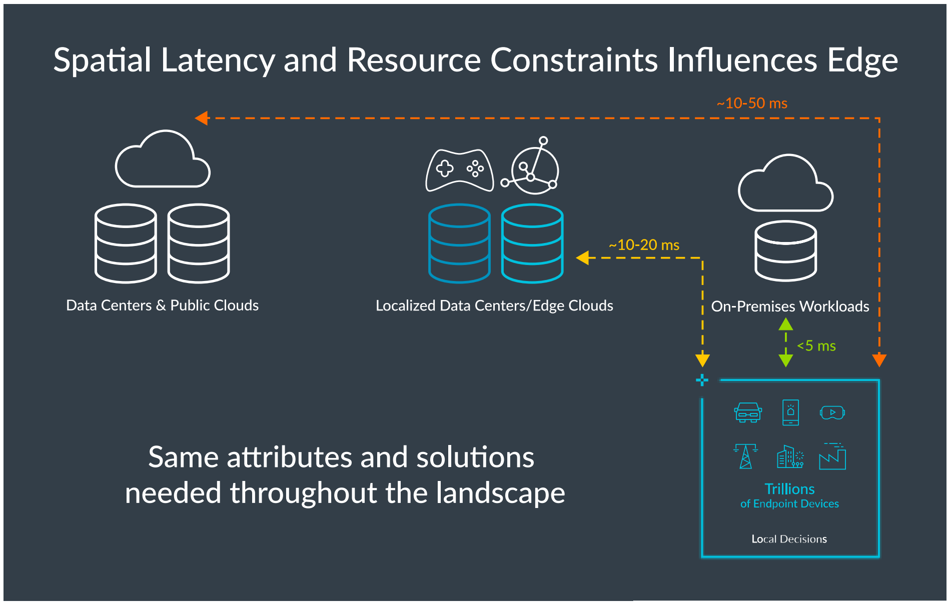
- **Compute resources**
  Edge devices are often resource-constrained systems, equipped with a smaller processor and less onboard memory, and powered by a battery. Many cloud-based applications for data processing are designed to run on systems that have more compute and power resources to draw on.

- **Latency and Bandwidth**
  Real-time operation requires an optimized balance between latency and bandwidth, with localized data centers, edge clouds, and on-premise workloads being serviced quickly enough to enable local decisions. To support efficient local decision-making, what might take more than 100ms to complete with a remote data center or public cloud now needs to happen much faster at the edge – with latencies as low as 5ms or less in some cases.

- **Scalability**
  Edge deployments need to support intelligent data-processing decisions close to where data is sourced, even if that means dealing with very large number of devices generating very large amounts of real-time data. Scalability has to be built into the deployment's baseline architecture, so rapidly growing numbers of IoT components and endpoints don't overwhelm the setup or negatively impact the ability to aggregate, analyze, and store data.

- **Security**
  Edge devices are often exposed to a wider variety of threats than datacenter equipment, which means datacenter security mechanisms don't always translate well to the edge environment. Lack of physical access to edge devices can complicate matters further, making it that much harder to address issues when they arise. The edge landscape is also heavily fragmented, in terms of hardware and software approaches to security, and there is growing need for standardized security principles for edge-specific protection.

Real-time operation at the edge requires very fast responses



Spatial Latency and Resource Constraints Influences Edge

Same attributes and solutions needed throughout the landscape

Bringing a Secure, Cloud-Native Experience to the Edge with Project Cassini

For several years now, developers working to meet the demands of scalable, real-time data processing have used the cloud-native approach to add automation and simplify application design, coding, and customization.

These same cloud-native techniques, which have been used to build applications that run exclusively in the cloud, can be tailored for use in the edge, so developers can be more efficient and more effective while addressing edge-specific operating requirements.

Arm created Project Cassini to meet this need. It is a comprehensive industry initiative aimed at creating a secure, flexible, and scalable approach to development at the edge of the network, using cloud-native techniques, so it's easier for developers to address the operating requirements of edge nodes and terminals.

Two elements of cloud-native architectures – Kubernetes and CI/CD (Continuous Integration/Continuous Delivery) systems – are part of Project Cassini use cases for scalable, real-time data processing at the edge.

- **Kubernetes**
  Containers address the need to have software run reliably when moving from one computing environment to another, be it from a developer's laptop to the test environment, from a staging environment into production and then deployment. The open-source Kubernetes container-orchestration system is known for its use of unified standards, which provide consistent functions and experiences across devices.

- **CI/CD Pipeline**
  The CI/CD pipeline creates a common framework for all developers, testers, and product managers working on a given software-development project. It also reduces the amount of manual work, helps catch critical bugs earlier, and makes it easier to incorporate feedback.

  The Continuous Integration (CI) portion of the pipeline builds and tests applications after code changes, and the Continuous Delivery (CD) portion pushes approved changes to a variety of environments. Using a CI/CD pipeline for software development creates a common framework for everyone, including developers, testers, and product managers.

  In particular, the CI/CD pipeline reduces risk by making it possible to deploy small, incremental updates that can be rolled out gradually to a large number of distributed edge devices, so any breaking changes, which may cause other components to fail, can be addressed quickly before they have a significant impact.

Benefits of using containers at the edge

- *Improved predictability –* with software functions that run the same way, everywhere
- *Higher security –* with isolated microservices that use permission-driven interactions
- *Software re-use –* with standards that guarantee operation in the environment provided
- *Reduced complexity –* with common operations deployed on common environments
- *Greater consistency –* with the same functions deployed every time

Benefits of using CI/CD at the edge

- *Better code and faster releases –* with smaller updates that are easier to isolate, test, and repair
- *Faster time-to-market for new features –* with quick rollouts for more responsive services
- *Smoother operation –* with cleaner code that has fewer non-critical defects
- *Increased automation –* with the ability to deploy code to the production environment with minimum human intervention
- *Higher efficiency –* with more developer time available to focus on code quality and innovation

The Project Cassini Approach: Rancher K3s and GitLab CI/CD

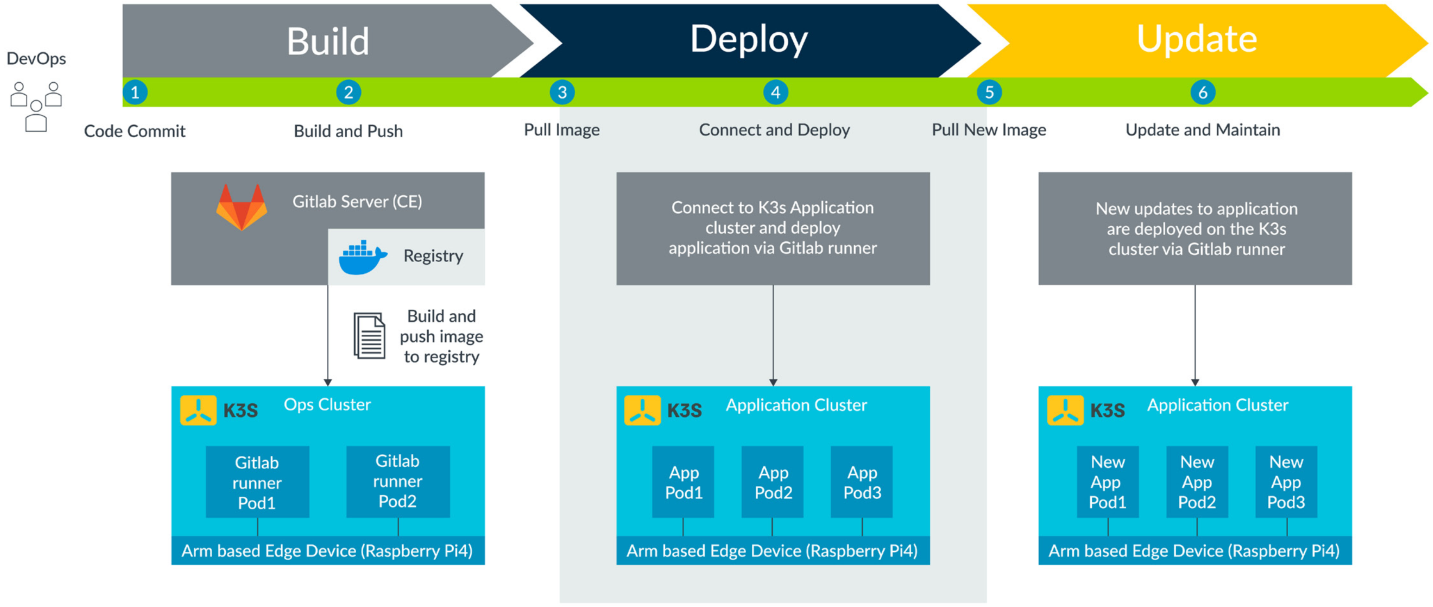
To deliver the benefits of cloud-native development in a format optimized for use with edge devices, Project Cassini combines a Rancher K3s based container solution with GitLab CI/CD system running on Arm-based edge devices (Raspberry Pi)

Rancher K3s is a lightweight, production-grade Kubernetes distribution optimized for the Arm architecture. Its small footprint (around 40 MB) enables quick, reliable application deployments to edge devices and other resource-constrained environments.

GitLab is a web-based DevOps platform that provides a Git-based repository that tracks all changes made to the files in a project. The Project Cassini pipeline uses the Community Edition (CE) of GitLab Server, for a license-free distribution that contains no proprietary code.

As shown in the diagram, the process of orchestrating applications includes three phases – build, deploy, and update. The process phase begins when a developer commits a change to the repository. A simple command is all it takes to trigger the pipeline. A GitLab Runner, supported on the Arm architecture, picks up the build job, executes it, and sends the results to the GitLab server. The GitLab Server hosts a Docker registry to version, store, and distribute the application docker images. In the delivery stage, the GitLab runner automatically pulls the application docker image from the registry, connects to a K3s application cluster, and deploys the application. All the config files related to this particular use case have been kept in this github repository.

The K3s/GitLab CI/CD pipeline automatically moves changes from build to deploy and update

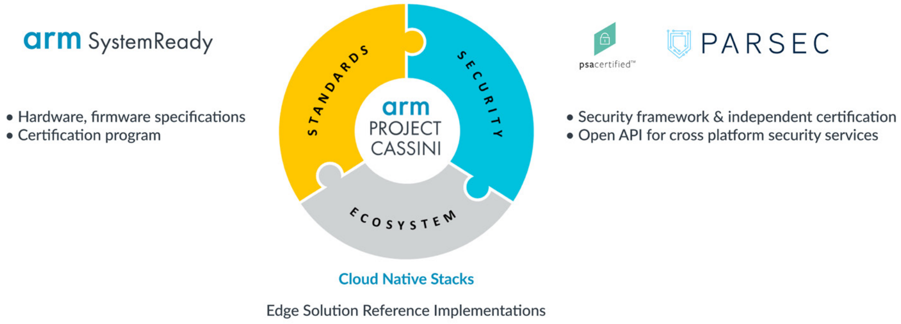


Each time an update is complete, a new version of the application is deployed on the K3s cluster. The process makes it easy to test and deploy cloud native applications. The overall approach is highly scalable, meaning multiple instances can be added based on load.

The CI/CD Pipeline is Just One Part of the Project Cassini Picture

The CI/CD pipeline is a valuable blueprint for edge-driven uses cases for data processing, and an important part of the Project Cassini ecosystem for cloud-native development. The pipeline is made stronger by the other elements of Project Cassini, which address issues of standards and security.

To support standardization, the SystemReady compliance program addresses fragmentation across low-level firmware and hardware features, so off-the-shelf and community software distributions ("distros"), as well as other workloads, run seamlessly on diverse Arm platforms. To address security threats, the Arm Platform Security Architecture (PSA) Certified program provides an objective assessment of quality when implementing the foundation for security, known as the Root of Trust (RoT). PSA Certified is made stronger by PARSEC, a collaboration with Docker that provides secure RoT abstraction to applications across platforms.


arm SystemReady
• Hardware, firmware specifications
• Certification program
STANDARDS
SECURITY
ECOSYSTEM
arm
PROJECT
CASSINI
psacertified™
PARSEC
• Security framework & independent certification
• Open API for cross platform security services
Cloud Native Stacks
Edge Solution Reference Implementations

Project Cassini uses standards, security, and ecosystem to create a cloud-native experience for edge

Tell Us What You Think

We're proud to count some for the most influential leaders in technology among our supporters. Feedback from our partners in critical. We welcome responses and seek commentary on the information we're providing. We look forward to receiving thoughts and ideas on the edge and how it can best be secured and leveraged. To share your feedback with us, we invite you to email: project-cassini@arm.com.

Arm is dedicated to collaboration. We welcome your insights on Project Cassini.

Take the Next Step

To learn more about Project Cassini and how it relates to smart-city deployments, visit www.arm.com/project-cassini.

All brand names or product names are the property of their respective holders. Neither the whole nor any part of the information contained in, or the product described in, this document may be adapted or reproduced in any material form except with the prior written permission of the copyright holder. The product described in this document is subject to continuous developments and improvements. All particulars of the product and its use contained in this document are given in good faith. All warranties implied or expressed, including but not limited to implied warranties of satisfactory quality or fitness for purpose are excluded. This document is intended only to provide information to the reader about the product. To the extent permitted by local laws Arm shall not be liable for any loss or damage arising from the use of any information in this document or any error or omission in such information.