MONTAVISTA
SOFTWARE

Application:
Robot Operating System (ROS)

CGX
MontaVista

Introduction

MontaVista:

With a history of more than 20 years, MontaVista is proudly empowering millions of devices across diverse industries worldwide.

Lately, some of the fastest evolving technology areas have been in the area of **Machine Learning (ML)** and **Artificial Intelligence (AI)**, combined with increasingly complex, **distributed sensor and actuator compute nodes**. The word "robot" in this context is used to describe a complex set of diverse, stationary and mobile compute modules collaborating to perform increasingly complex tasks.

Applications for this span from **intelligent warehouses** and **industrial production lines** to **telemedicine** and **self-driving vehicles**. Technologies aiding in those tasks range from simple Computer Vision to Simultaneous Localization And Mapping (SLAM) and AI inference up to conversational user interfaces.

Whenever you need the highest reliability Linux embedded applications, MontaVista's experts will help you to find the best solutions.

Robot OS (ROS):

The Robot Operating System (ROS) is an **open-source robotics middleware suite** which facilitates software development for **Robots and Robot-like industrial applications**. ROS2 has been designed with support for real-time use cases, **enabling developers to build more robust and predictable robotic applications.**

Although ROS is not an operating system itself, ROS is frequently used on the Linux Operating System. The supported ROS releases are typically available as pre-built distributions for the current versions of Ubuntu Linux distributions for AMD64 and ARM64 (and recently for Windows).

This document presents an overview of MontaVista's CGX Linux and MV-Secure services in the context of a system design based on Robot OS (Version 2).

Anatomy of a ROS2 System

A ROS2-based system is composed of several interconnected components that work together to enable robotic applications. Here's a high-level overview of the anatomy of a typical ROS2-based (distributed) system:

Nodes:

Nodes are the fundamental units of computation in ROS2. They are individual processes that perform specific tasks, such as controlling sensors, processing data, or executing algorithms.

Multiple nodes typically run concurrently within a ROS2 system, each responsible for different aspects of the robot's functionality.

Depending on the needs, nodes can run on dedicated hardware (e.g. NVidia Jetson with hardware acceleration for GPU intensive tasks like AI or Computer Vision) to medium sized embedded systems (e.g. for communication with sensors and actuators) all the way to containerized or virtualized nodes running on shared hardware.

Nodes communicate with each other by publishing and subscribing to topics to send and receive messages.

Topics and Messages:

Messages are data structures used to represent information exchanged between nodes. Each message type corresponds to a specific data format, such as sensor readings, motor commands, or robot poses.

Topics are named buses over which nodes exchange messages. Nodes can publish messages to a topic to share data or subscribe to a topic to receive data. Topics enable asynchronous communication between nodes, allowing them to send and receive data independently.

Data Distribution Service (DDS):

The DDS is a standardized, data-centric middleware that facilitates the exchange of data between different Nodes of a distributed ROS system. It plays a crucial role in the communication infrastructure of a ROS System.

ROS2 supports the use of various DDS implementations. Popular choices are Fast RTPS or Cyclone DDS, to handle communication between nodes. It provides features like discovery, serialization, and transport for efficiently exchanging messages between nodes.

Users can choose the DDS implementation that best fits their needs for performance, reliability, and scalability. Sophisticated implementations of the DDS can handle communication in large-scale robotic deployments with hundreds or thousands of nodes, while ensuring robustness and fault tolerance.

ROS-Industrial (ROS-I):

ROS-Industrial is an open-source project which builds on the features of ROS2 and extends the advanced capabilities of ROS into the realm of manufacturing automation and robotics. The ROS-Industrial repository includes interfaces for common industrial manipulators, grippers, sensors, and device networks.

For more information about ROS-I, see: https://rosindustrial.org/

Solution Description

The hardware of a ROS based robot is typically built around a powerful ARM or embedded x86 processor, typically running a flavor of the Linux OS.

This section shows some typical use cases in the lifecycle of your ROS2 based robot.

Prototyping your Robot in the lab:
A key strength of the ROS2 ecosystem is the wide array of hardware sensors and actuators available in Open Source. This enables very quick prototyping with real hardware, or virtual, using the different simulation environments bundled with ROS.

For an overview of available hardware drivers, take a look at:
https://github.com/orgs/ros-drivers/repositories

Bringing your Robot from the lab into the real world:

While the ROS2 documentation currently walks the user through setting up their ROS workspace on a Ubuntu (Debian Linux Based) - for an professional CICD development workflow, a Yocto based build system should be used. Let the Yocto experts at MontaVista help you to integrate ROS runtime with your embedded development flow.

For an overview of available Yocto versions and their support for ROS meta-packages, see here: https://github.com/ros/meta-ros?tab=readme-ov-file#meta-ros

Keeping your Robot healthy

Once your Robot is set-up in an industrial environment, it will most likely be connected to the Internet for remote access, maintenance and service.

First and foremost at this time is to ensure your ROS and Linux OS are designed and implemented to the best cybersecurity guidelines - such as ensuring the chain of trust from power-up to boot-up, and using properly configured SELinux settings.

During its useful life, the software running on the robot needs to be remotely updatable and the manufacturer has to ensure that all known vulnerabilities are fixed and deployed to the robots in the field. Both the USA and Europe have recently released updated cybersecurity guidelines, see:
https://www.cisa.gov/sites/default/files/2023-10/SecureByDesign_1025_508c.pdf
https://www.enisa.europa.eu/topics/certification/cybersecurity-certification-framework

Those requirements can be challenging to meet for smaller and medium sized engineering teams. MontaVista has decades of experience keeping Linux systems running reliably - even in hostile environments such as telecommunications infrastructure.

Summary

ROS2 is a great toolkit and middleware platform to rapidly prototype and develop robots and other industrial distributed devices.

MontaVista's CGX Linux enhances embedded industrial systems with unmatched robustness and connectivity, designed for rigorous industrial needs.

The secure environment of MontaVista's CGX, supported by secure boot and comprehensive security features, guarantees robust protection and data integrity for your robots in the field.

If you are looking for a secure, reliable, and connected industrial hardened embedded system solution, MontaVista's CGX Linux meets all of these demands and more, please get in touch with us today to harness the unique benefits of the ROS on MontaVista's secure Linux platform.

We invite all interested parties to visit www.mvista.com and/or contact us at sales@mvista.com or by any of the means listed below.

MontaVista Software, LLC, is a leader in embedded Linux commercialization.

For over 20 years, MontaVista has been helping Linux developers get the most out of open source by adding commercial quality, integration, hardware enablement, expert support, and the expert resources of the MontaVista development community. Because MontaVista customers enjoy faster time-to-market, more competitive device functionality, and lower total cost, more devices have been deployed with MontaVista than with any other Linux.

For more information about MontaVista, visit http://www.mvista.com
Media Contact:
MontaVista Software LLC | E: pr-contact@mvista.com | T: +1 (669) 777-6841

**Linux® is a registered trademark of Linus Torvalds in the United States and other countries.*
MontaVista® is a registered trademark of MontaVista Software, LLC.
All other names mentioned are trademarks, registered trademarks or service marks of their respective owners.