

1 S-CD-12 ROS 1 - ROS 2 Bridge

Artifact ID	Artifact Title		
S-CD-12	ROS 1 - ROS 2 Bridge		
Team	Revision	Artifact Date	
BYU Mars Rover	0	22 Oct 2024	
Prepared by:		Checked by:	
Nelson Durrant		Alyssa Fielding	

1.1 Revision History

Revision	Date	Made by	Checked by	Description
0	22 Oct 2024	Nelson Durrant	Alyssa Fielding	Initial upload

1.2 References

Artifact ID	Revision	Title
S-CD-13	0	Containerized Development and Deployment

1.3 Purpose

This artifact describes the reasoning process behind using a containerized ROS 1 to ROS 2 bridge during the ROS 2 code conversion process, as well as links to simple instructions for using the bridge.

1.4 Concept Definition

Motivation

Our rover's code is currently written in ROS 1 Noetic LTS. Using this open-source robotics framework to structure the rover's software has allowed past teams to effectively organize and maintain code using node divisions built on the SRP (single-responsibility principle), and has allowed easy integration with a large variety of open-source robotics packages and applications. However, the OSRF (Open-Source Robotics Foundation) will end support for this distro (ROS 1 Noetic LTS) in May 2025. In order to preserve the advantages of using the Robot Operating System project while still ensuring future software support and compatability, one of the main focuses for this years team will be transitioning our current code from ROS 1 Noetic LTS to ROS 2 Humble LTS. This distribution will be supported until May 2027 and should only require small updates to transition to newer ROS 2 versions in the future.

The ROS 1 to ROS 2 transition, however, will be a larger undertaking that requires a bit more intentionality in the conversion process. It would be incredibly useful to be able to test our new ROS 2 code within our

existing ROS 1 system without having to rewrite the entire software stack in ROS 2 beforehand. A ROS 1 - ROS 2 bridge allows us to achieve this functionality.

Gradual Conversion

The ROS 1 - ROS 2 bridge (as currently implemented) allows simple, easy communication between the ROS 1 and ROS 2 message passing protocols. This allows us to run the original ROS 1 software stack on the rover with certain nodes or packages disabled, run the ROS 2 equivalent of those same nodes and packages in a containerized ROS 2 environment, and test to make sure those new ROS 2 packages match the functionality of the original ROS 1 ones. This iterative build-and-test process should allow us to effectively and gradually convert each of the existing ROS 1 packages to ROS 2 while ensuring that the rover as a whole stays functional and available for testing by other team members and projects.

Containerization

We have also utilized a containerized approach in building the ROS 1 - ROS 2 bridge. For further documentation on the advantages of this strategy, see the artifact "S-CD-13 Containerized Development and Deployment."

Documentation

The code, documentation, and instructions for using the ROS 1 - ROS 2 bridge can be found using this GitHub link: https://github.com/snelsondurrant/docker_ros1_bridge