ROSDiscover: Statically Detecting Run-Time Architecture Misconfigurations in Robotics Systems

Artifact Paper

Christopher S. Timperley, Tobias Dürschmid, Bradley Schmerl, David Garlan, and Claire Le Goues
Carnegie Mellon University, Pittsburgh, PA, USA
{ctimperl, tdurschm, schmerl, garlan, clegoues}@cs.cmu.edu

2022 IEEE 19th International Conference on Software Architecture Companion (ICSA-C) | 978-1-6654-9493-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICSA-C54293.2022.00055

***Abstract*—This is the replication package for the paper, ROSDiscover: Statically Detecting Run-Time Architecture Misconfigurations in Robotics Systems, which is published at the International Conference on Software Architecture (ICSA), 2022.**

The artifact contains (a) the tool ROSDiscover, which is a component and connector architecture recovery tool that recovers and checks robotics systems built in the Robot Operating System (ROS) 1; (b) data set of architecture misconfiguration bugs of real-world open-source ROS 1 systems on GitHub; and (c) scripts and instructions for replicating the results produced in the paper that show that it is both possible to accurately recover run-time architectures of ROS 1 systems, and that these architectures can be used to detect misconfiguration bugs that were found in real systems.

I. Introduction

This replication package contains, scripts for reproducing the results presented in the paper, as well as the source code for the complete ROSDiscover toolchain:

- **ROSDiscover**: This is the tool that is described in the paper. It is designed to, among other purposes, recover run-time architectures from ROS applications provided in the form of a Docker image and an accompanying configuration file. Further instructions on the general use of ROSDiscover can be found in its `README` file, available either in its archival form in the deps/rosdiscover directory of this artifact, or, preferably, in its up-to-date form on GitHub at: https://github.com/rosqual/rosdiscover. ROSDiscover has commands for recovering component models, observing running systems to produce observed architectures, and statically assembling architectures to form recovered architectures.
- **ROSWire**: This is a standalone Python library, used as part of the ROSDiscover toolchain, that provides extensive functionality for building static and dynamic tools for ROS that accept Docker images as their input (rather than assuming that those tools are located on the same machine as the subject of the analysis). More details about ROSWire can be found in its `README` file, available either in its archival form in the deps/roswire directory of this artifact, or, preferably, in its up-to-date form on GitHub at: https://github.com/rosqual/roswire.
- **CXX-Extract**: Provides the implementation of the static component model recovery of ROS nodes from source code written in C++. Available at: https://github.com/rosqual/rosdiscover-cxx-recover

When ROSDiscover is invoked to recover an architecture, it uses ROSWire to locate packages, launch files, etc. ROSDiscover subsequently invokes CXX-Extract when it encounters a node in a launch file it is processing, to parse the source, identify ROS API calls, and produce a component model. ROSDiscover then combines the component models according to the launch files being processed and resolves any parameters, arguments, unbound topics, etc. that may be in the component models to produce an architecture model.

II. Availability and Installation Requirements

The artifact is available at https://doi.org/10.5281/zenodo.5834633.

The artifact contains an archive of the over 50 Docker images of the ROS systems studied in the paper. In the steady state, the package requires around 100GB of disk space. However, during installation, around 250GB of disk space is required. The package has been tested on Linux installations, including Ubuntu and Arch. It requires Docker 17.04 or higher, as it uses multi-stage builds.

Further information is available in the `INSTALL` file included with the package.

III. Summary of Contents

The full contents of the package are described in the `README` of the artifact. The artifact itself contains:

- `images.tar.gz`: The built containers of the ROS systems studied in the paper. This means that the containers do not need to be built from scratch, although the replication package does provide means to do that.
- `paper.pdf`: A copy of the paper that this artifact is associated with.
- `README.rst`: Introduces the artifact.
- `replication-package.zip`: The archive containing all the source code, results, bug data set, and instructions for the artifact.

This work has been supported in part by NASA (Award 80NSSC20K1720), AFRL (Award 19-PAF00747), and the NSF (Award CCF-1750116)

978-1-6654-9493-9/22/$31.00 ©2022 IEEE
DOI 10.1109/ICSA-C54293.2022.00055