Tracing ROS 2

Christophe Bourque Bédard

Progress Report Meeting
January 14, 2022

Polytechnique Montréal
DORSAL Laboratory

POLYTECHNIQUE MONTRÉAL
TECHNOLOGICAL UNIVERSITY

Summary

1. Introduction
2. ROS 2
3. Tracing ROS 2
4. Overhead evaluation
5. Analysis, Trace Compass plugin
6. Upcoming work and conclusion
7. Questions

Introduction

- Robotics
  - Commercial or industrial applications
  - Safety-critical applications
  - Can be connected over a network (e.g., 5G)
- Key elements
  - Message passing (publish-subscribe) and Remote Procedure Call (RPC)
  - Higher-level scheduling of tasks is challenging
  - Real-time constraints
- Robotics software development can greatly benefit from tracing

ROS 2

- Robot Operating System 2
  - docs.ros.org/en/galactic
- Open source framework and set of tools for robotics software development
  - Well-known in robotics
  - Used for NASA's 2023 Moon rover, VIPER!
- Message passing between "nodes"
  - Publish/subscribe
  - Service/action calls (~RPCs)
- Modular
  - Each node generally accomplishes a very specific task
  - Nodes are put together to perform complex tasks
- Uses Data Distribution Service (DDS) as the middleware
  - OMG standard
- Intra-process, inter-process, and distributed

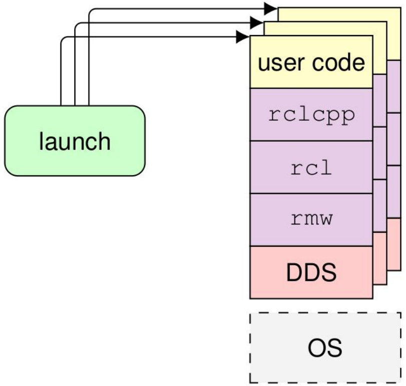

Figure 1. ROS 2 architecture and orchestration.

Tracing ROS 2

- Tools part of the ROS 2 core
  - gitlab.com/ros-tracing/ros2_tracing
- LTTng instrumentation in ROS 2
  - Message publication & reception
  - Subscription & timer callbacks
  - Etc.
  - Constant number of trace events, constant overhead (?)
- And some LTTng instrumentation for a DDS implementation
- Tracing tools closely integrated with ROS 2
  - ROS 2 CLI tools
  - ROS 2 launch/orchestration system

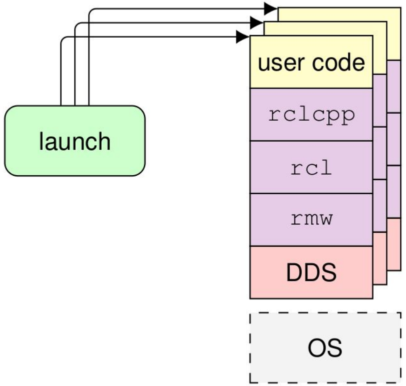

Figure 1. ROS 2 architecture and orchestration.

Overhead evaluation

- Goal: measure tracing overhead in a ROS 2 context
  - Latency overhead using 1 publisher → 1 subscription (inter-process)
  - Tool: gitlab.com/ApexAI/performance_test
- Parameters
  - Publishing rate: 100-2000 Hz
  - Message payload size: 1-256 KiB
  - Quality of service settings: reliable
  - DDS implementation: eProsima Fast DDS
- Setup
  - Ubuntu Server 20.04.2 with PREEMPT_RT (5.4.3-rt1)
  - Intel i7-3770 @ 3.40GHz, 8 GB RAM
  - SMT/Hyper-threading disabled (4 cores, 1 thread/core)
  - CPU power-saving features disabled through the BIOS
  - Run for 60 minutes, discard the first 10 seconds, and use mean latency

Overhead evaluation - results

- Latency overhead is mostly constant

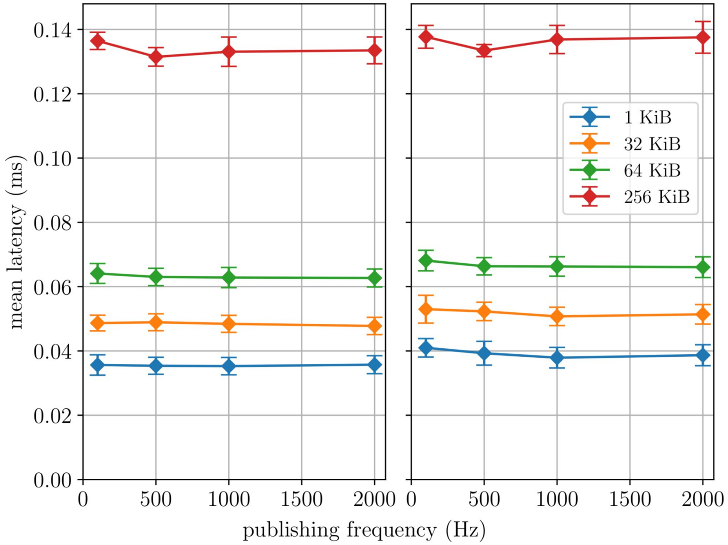

Figure 2. Latencies without tracing (left) and with tracing (right).

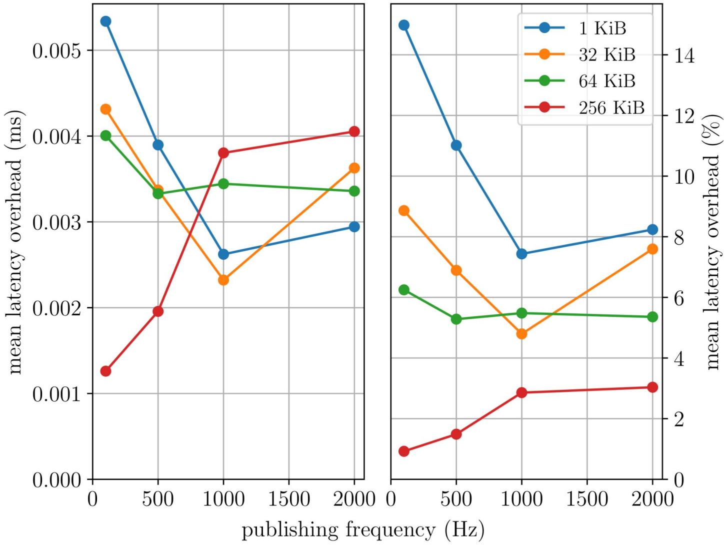

Figure 3. Absolute (left) and relative (right) latency overhead results.

Analysis

- Can extract basic metrics
  - Publishing rate, callback execution rate
- Can visualize
  - Message publications
  - Subscription & timer callbacks
- Example
  - 1 source node publishes messages periodically
  - 1 intermediary node receives those messages and publishes other messages
  - 1 sink node receives those messages

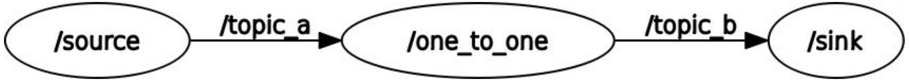

Figure 4. Example node structure.

Analysis (2)

- ROS 2 plugin for Trace Compass (work in progress)

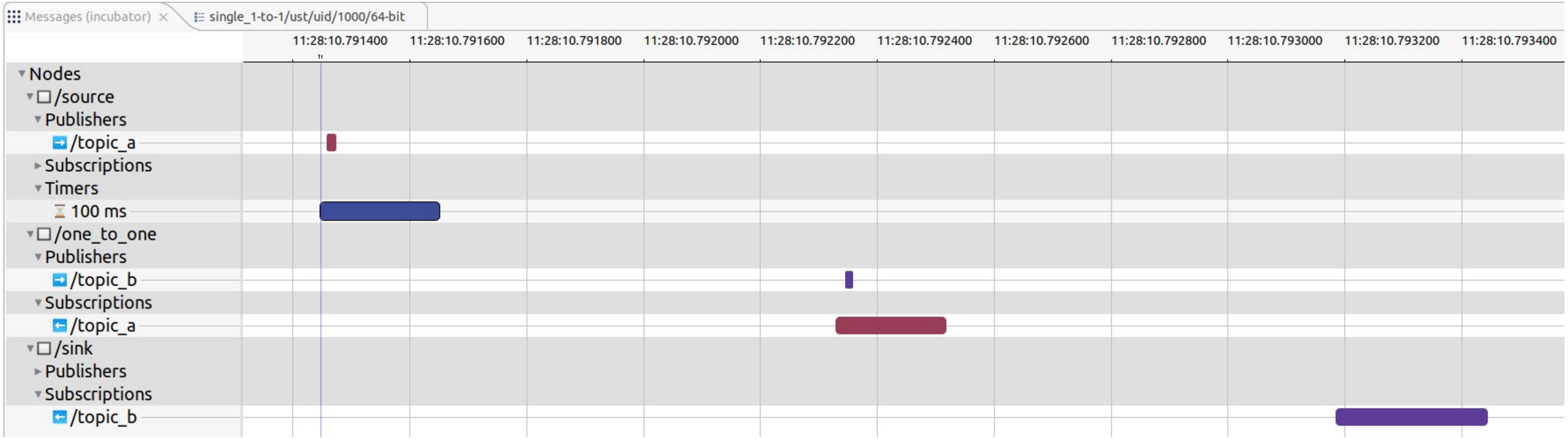

Figure 5. Time graph view showing message publications and timer & subscription callbacks.

Analysis (3)

- Starting point: want to link messages for end-to-end message flow
- However, some message links are not trivial (e.g., asynchronous, cached)
- Furthermore, message links could be N-to-M, not necessarily 1-to-1

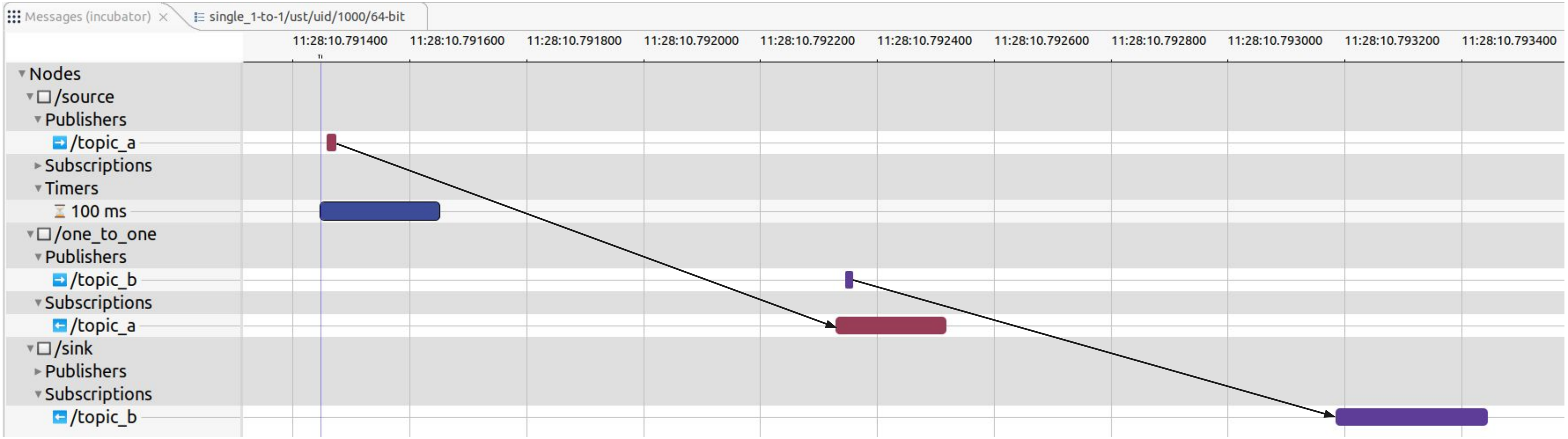

Figure 6. Future plans for this view (arrows).

Upcoming work and conclusion

- Tracking messages across nodes
  - Building a message flow graph using this information
- Computing end-to-end latency automatically
- Critical path analysis at the ROS 2 level

- More and more public interest & users

Questions?

- christophe.bedard@polymtl.ca

- Links
  - docs.ros.org/en/galactic
  - gitlab.com/ros-tracing/ros2_tracing
  - gitlab.com/ApexAI/performance_test
- Other relevant links
  - Recent paper (in review): arxiv.org/abs/2201.00393
  - Recent presentation at a ROS conference: vimeo.com/652633418 (slides)