U.S. ARMY COMBAT CAPABILITIES DEVELOPMENT COMMAND – GROUND VEHICLE SYSTEMS CENTER

ROS-Military — ROS 2 Overview

Jerry Towler

Unmanned Systems Lead

Southwest Research Institute

30-Sep-2020

ROS 2 Overview

Production-Enabling Evolution of ROS 1

WHAT IS ROS 2?

- **New system using ROS 1 concepts, but targeting production**
  - Real-time capable
  - Cybersecurity capable
  - Lifecycle capable
  - Additional first-class platforms: Windows, MacOS
- **Other improvements**
  - Pluggable back-end for transport layer-agnostic communication
  - Pluggable front-end for multiple language support
  - Distributed node management
  - Flexible mappings between nodes and processes

WHY ROS 2 FOR ROS-M?

- **ROS-M has always targeted ROS 2 for its cybersecurity possibilities**
  - ROS 1 has a single security story: there is none
  - ROS 2 permits the use of the DDS-Secure profile
    - Authentication
    - Access Control
    - Encryption

WHY ROS 2 FOR ROS-M?

- **ROS-M has always targeted ROS 2 for its cybersecurity possibilities**
  - ROS 1 has a single security story: there is none
  - ROS 2 permits the use of the DDS-Secure profile
    - Authentication
    - Access Control
    - Encryption
- **ROS-M provides base level of capability for acquisition pipeline**

ROS 2 Development

Supporting ROS-Military, RTK, and other programs

REALISTIC CURRENT STATUS OF ROS 2

“Dashing Diademata” released June 2019

“Foxy Fitzroy” released June 2020

REALISTIC CURRENT STATUS OF ROS 2

- **Installed ROS 2 on Rover Robotics rover**
  - Provides realistic ROS 2 test environment
- **Tested ROS 2 for RTK-relevant hardware**
  - Velodyne LIDAR
  - Microstrain IMU

ROS 2 Readiness Evaluation

Led by Matt Schickler

ROS 2 READINESS: INTRODUCTION

OBJECTIVE: Evaluate **ROS 2 Eloquent** readiness for ROS-M and make recommendations for future action.

STRATEGY: Execute targeted gap analysis

- Compare core ROS 2 functionality to ROS 1
- Perform feature survey of community—maintained ROS 2 packages
  - Limit survey to packages of particular interest to ROS-M
  - Identify which packages are available as binary vs. source
- Conduct NAMC user survey among NAMC members to understand current ROS usage, ROS 2 plans, and issues preventing greater ROS 2 adoption
- Identify resources for ROS 2 training and migration

ROS 2 READINESS: USER SURVEY

71% of respondents are currently using ROS 1
- 48% to a significant (50%) or greater extent
- 6% are using it exclusively

41% of respondents are using ROS 2
- 18% to a significant (50%) extent
- **none** are using it exclusively

ROS 2 READINESS: USER SURVEY

53% of respondents are quite to highly likely to start using ROS2 **within the next year**

30% of respondents are quite to highly likely to start porting ROS 1 code to ROS 2 **within the next year**

ROS 2 READINESS: USER SURVEY

Primary factors hindering adoption:
Lack of overall maturity of ROS 2
Key ROS 1 features missing

- Comprehensive set of **device drivers**
- Motion planning framework (**MoveIt!**)
- XML-based **launch files**

ROS 2 READINESS: SUMMARY

ROS 2 Eloquent: core infrastructure approaching feature parity with ROS 1, but **not there yet**

ROS 2 Foxy might be point at which parity is achieved

Completeness and maturity of ROS 2 capabilities lagging behind core functionality

Cartographer (3D SLAM) no longer supported by Google

Still waiting for many ROS packages to be officially released (e.g., MoveIt2! is in Beta)

Training materials are sparse but growing

ROS 2 READINESS: UPDATE

ROS 2 Eloquent: core infrastructure approaching feature parity with ROS 1, but **not there yet**

ROS 2 Foxy might be point at which parity is achieved
Not yet; parameters, launching, rosbag…

Completeness and maturity of ROS 2 capabilities lagging behind core functionality

Cartographer (3D SLAM) no longer supported by Google
SLAM Toolbox selected for ROS 2
But… supported by one person (Steve Macenski)… for free

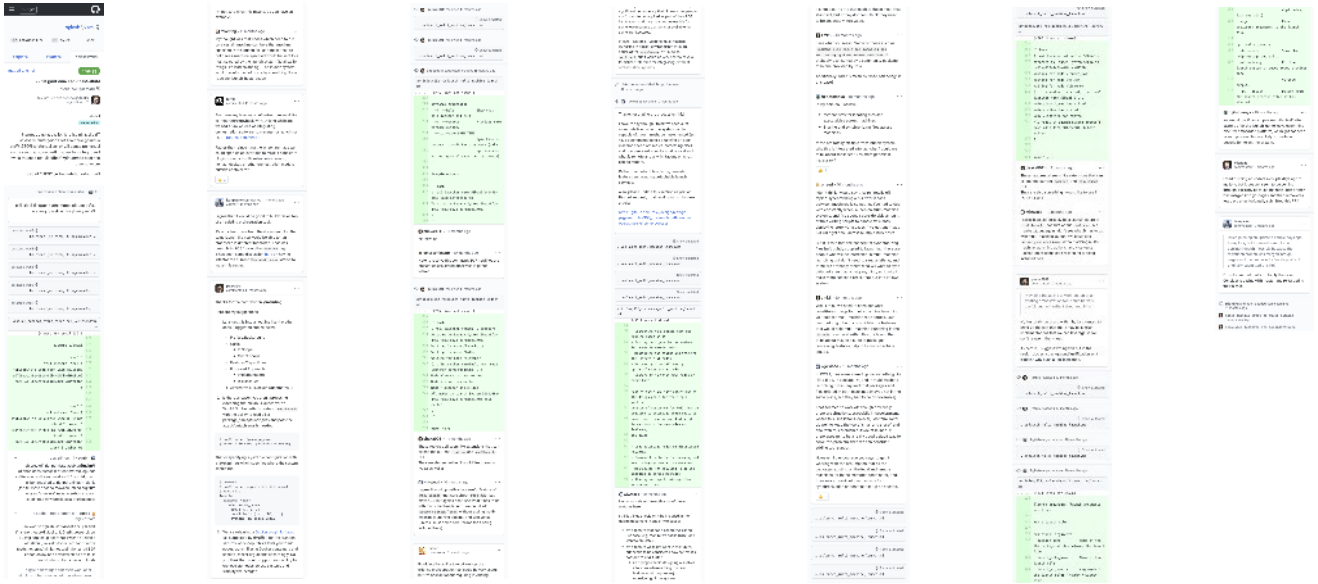
Still waiting for many ROS packages to be officially released (e.g., MoveIt2! is in Beta) **Still in beta**

Training materials are sparse but growing **Still sparse**

Primary Contribution: Multi-Machine Launching

MULTI-MACHINE LAUNCHING

- **Multi-machine launching is a crucial RTK capability**
- **Wrote a design document and shared with community**
  - https://github.com/ros2/design/pull/255
  - 13 community participants left more than 60 comments over 4 months
  - Led to 12 revisions (so far) to address community concerns



MULTI-MACHINE LAUNCHING

- **Multi-machine launching is a crucial RTK capability**
- **Wrote a design document and shared with community**
  - https://github.com/ros2/design/pull/255
  - 13 community participants left more than 60 comments over 4 months
  - Led to 12 revisions (so far) to address community concerns

miker256

This really is really great see @Matt_Lanting. Keep up the great work!

MULTI-MACHINE LAUNCHING

- **Multi-machine launching is a crucial RTK capability**
- **Wrote a design document and shared with community**
  - https://github.com/ros2/design/pull/255
  - 13 community participants left more than 60 comments over 4 months
  - Led to 12 revisions (so far) to address community concerns
- **Wrote a *new* design document and shared with community**
  - https://github.com/ros2/design/pull/297
  - Completely reconsidered complexity and interaction model
  - Issued 18 August 2020

MULTI-MACHINE LAUNCHING

- **Multi-machine launching is a crucial RTK capability**
- **Wrote a design document and shared with community**
  - https://github.com/ros2/design/pull/255
  - 13 community participants left more than 60 comments over 4 months
  - Led to 12 revisions (so far) to address community concerns

Still working on it…

LAUNCH MITIGATION STRATEGY

- **Multi-machine launching is a major undertaking**
- **RTK may need a multi-machine solution sooner**

LAUNCH MITIGATION STRATEGY

- **Multi-machine launching is a major undertaking**
- **RTK may need a multi-machine solution sooner**
- **Two mitigation strategies**
  - ✓ Use SSH to replicate ROS 1 functionality

LAUNCH MITIGATION STRATEGY

- **Multi-machine launching is a major undertaking**
- **RTK may need a multi-machine solution sooner**
- **Two mitigation strategies**
  - ✓ Use SSH to replicate ROS 1 functionality
  - 🚧 Refactor ROS 2 launch system
    - Issue: https://github.com/ros2/launch/issues/114
    - Design document: https://github.com/ros2/design/pull/272
    - Initial code: https://github.com/ros2/launch/pull/454

7

77

ROS 2 Community Interaction

GVSC ISSUES REPORTED

Repository	Issue	Name	Status
ros1_bridge	181	Garbled text output in [INFO] messages	Resolved by OR
launch	309	`_prune_and_count` iterating over collection twice	Unresolved
velodyne	298	`velodyne_convert_node` doesn't work in ROS 2 Dashing	Resolved by community
moveit2	100	Errors when running ubuntu-install.sh	Resolved by community
moveit2	104	Build error: Package 'moveit_core' exports the library 'moveit_exceptions' which couldn't be found	Unresolved
mara_examples	4	Clone and install instructions missing	Resolved by community
rosbag2	447	Buildes prioritizing binaries instead of source	Resolved by OR

GVSC ISSUES REPORTED

Repository	Issue	Name	Status
rmw_cyclonedds	111	Crash when subscribing to a single topic with multiple types	Resolved by vendor
rmw_cyclonedds	126	Crashes when receiving a message on /rosout	Resolved by vendor

ROS 2 PORTS

Repository	Description
gps_umd	Necessary messages and tooling for GPS drivers Used extensively by RTK
microstrain_3dmgx2	Microstrain is the primary IMU vendor for many RTK systems
mapviz	Primary RTK visualization tool
marti_common	Significant dependencies of many parts of RTK
rqt_runtime_monitor	Important development and testing tools
rqt_robot_monitor	
swri_console	

GVSC ISSUES RESOLVED

Repository	Issue	Name	Notes
rcl	258	Treat __name the same as __node	Extremely challenging UX benefit
velodyne	299	Check for intra-process subscribers	Unexpected failure
rclcpp	471	Executor::spin_some() should check for work only once	Unexpected infinite loop 11 participants, 40 comments
novatel_gps_ driver	74	Refactor GPSFix generation	More accurate GPS fix
novatel_gps_ driver	104	Add support for Epson G370	Missing IMU model
novatel_gps_ driver	47	gps_common::GPSStatus ::STATUS_FIX	Community-reported
novatel_gps_ driver	58	Inconsistent Get methods in the nodelet	Community-reported Unnecessary computation

GVSC ISSUES RESOLVED

Repository	Issue	Name	Notes
novatel_gps_ driver	70	gps_common/GPSFix has wrong time data	More accurate GPS time Resulted in significant rewrite
novatel_gps_ driver	75	Parsing GPGSV logs with 0 satellites fails	Unexpected failure
microstrain_ 3dmgx2_imu	12	ROS2 Compatibility	Unavailable in ROS 2
novatel_gps_ driver	58	Inconsistent Get methods in the nodelet	Community-reported Unnecessary computation

GVSC ISSUES RESOLVED

Repository	Issue	Name	Notes
rosbag2	414	Compress by message	Greater recording flexibility
rosbag2	413	Recording bag with message compression not implemented	Magically fixed by previous fix
rosbag2	418	gps_common::GPSStatus ::STATUS_FIX	Community-reported
rosbag2	274	Compress files & messages asynchonously	Required for data-heavy systems like RTK
rosbag2	511	Restructuring compression code	Prompted by previous issue
rosbag2	459	Consolidate ZSTD utility functions	Reduce implementation complexity
rosbag2	233	Add split by time to recording	Required for data-heavy systems like RTK

STILL WORKING ON IT…

ROS bag metadata recreation

- Catastrophic data recording failure

STILL WORKING ON IT…

ROS bag metadata recreation

- Catastrophic data recording failure
- Current status: unrecoverable

STILL WORKING ON IT…

ROS bag metadata recreation

- Catastrophic data recording failure
- Current status: unrecoverable
- We're on it!

STILL WORKING ON IT…

ROS bag metadata recreation

- **Catastrophic data recording failure**
- **Current status: unrecoverable**
- **We're on it!**
  - This makes **two** major rosbag2 changes prompted by GSVC investigation…

DOD RTK SUPPORT

Supported ERDC deployment of RTK within a Singularity process

(Singularity is basically USG Docker)

INTERNAL IMPROVEMENTS

- ROS 2 Foxy Docker Image
- RTK Docker Configuration

COMMUNITY INTERACTION

- ROS 2 Technical Steering Committee
- Tooling Working Group
- Security Working Group

COMMUNITY INTERACTION

- ROS 2 Technical Steering Committee
- Tooling Working Group
- Security Working Group

The U.S. Army is a good ROS citizen

GVSC OSS OPSEC PROCESS

- ✓ **Markings on committed code**
  - Many commits will contain just pieces (small changes to existing files)
  - Add "Distribution A, OPSEC #[number]" in the commit message
- ✓ **License**
  - Contract has all work by DCS and SwRI as unlimited rights
  - Under the DFARS, which is typically used for DoD contracts, the government can release software as open source software once it receives "unlimited rights" to that software (DFARS 252.227-7014)
- ✓ **Messaging/Discussion/Collaboration on forums as the development arm of a Gov entity**
  - Logging all public facing interaction (needs to live on past the website)
  - PAO briefing on how to conduct oneself on behalf of the Gov

ROS 2 STATEMENT OF WORK

- **Launch system –** supporting the ability to launch a set of ROS nodes across multiple physical systems as well as software to convert the ROS 1 XML launch files to the ROS 2 Python equivalent.
- **Parameter server –** supporting the ability to pass values between nodes via a command line interface, expand on the predefined constraints, or allow an ability to override a constraint.
- **DDS configuration –** supporting the ability to view current DDS QoS parameters through the ROS command line interface.
- **Hardware drivers –** supporting the ability to monitor and control sensors necessary for robotic operations.

ROS 2 STATEMENT OF WORK (NEW)

- **Developer tools** – supporting the ability to efficiently develop and test robotic systems
- **DDS security** – supporting the ability to take advantage of all DDS cybersecurity capabilities across the entire robotic system
- **Rust language support** – supporting the ability to write ROS nodes in Rust, an inherently memory-safe language, enabling the development of safer and more secure robotic systems

Self-Evaluation

SELF-EVALUATION

Focus Area	Evaluation

SELF-EVALUATION

Focus Area	Evaluation
Launch System	MML Design—huge interest Launch system refactor—huge impact

SELF-EVALUATION

Focus Area	Evaluation
Launch System	MML Design—huge interest Launch system refactor—huge impact
Parameter Server	*Not much progress*

SELF-EVALUATION

Focus Area	Evaluation
Launch System	MML Design—huge interest Launch system refactor—huge impact
Parameter Server	*Not much progress*
DDS Configuration	NoDL Evaluation/Assessment Security Working Group Membership

SELF-EVALUATION

Focus Area	Evaluation
Launch System	MML Design—huge interest Launch system refactor—huge impact
Parameter Server	*Not much progress*
DDS Configuration	NoDL Evaluation/Assessment Security Working Group Membership
Hardware Drivers	Velodyne LIDAR—bugs fixed, driver improved Microstrain IMU—ported to ROS 2 Novatel GPS—tons of activity

SELF-EVALUATION

Focus Area	Evaluation
Launch System	MML Design—huge interest Launch system refactor—huge impact
Parameter Server	*Not much progress*
DDS Configuration	NoDL Evaluation/Assessment Security Working Group Membership
Hardware Drivers	Velodyne LIDAR—bugs fixed, driver improved Microstrain IMU—ported to ROS 2 Novatel GPS—tons of activity
Dev Tooling	ROS 2 Command-Line UX Improvements Docker configurations rosbag fixes and enhancements

SELF-EVALUATION

Focus Area	Evaluation
Launch System	MML Design—huge interest Launch system refactor—huge impact
Parameter Server	*Not much progress*
DDS Configuration	NoDL Evaluation/Assessment Security Working Group Membership
Hardware Drivers	Velodyne LIDAR—bugs fixed, driver improved Microstrain IMU—ported to ROS 2 Novatel GPS—tons of activity
Dev Tooling	ROS 2 Command-Line UX Improvements Docker configurations rosbag fixes and enhancements
Community Interaction	Technical Steering Committee & Working Groups Hundreds of comments in dozens of issues/PRs Internal DOD community, too

Conclusion