



PROJECT MANAGER FORCE PROJECTION

RAS-G IOP Update: NDIA Robotics Division December 2016

Mark Mazzara
PM Force Projection
Robotics Interoperability Lead



Agenda

- RAS-G IOP Update
- ROS-M Update
- NAMC/VRA Standard Cycle Projects



Interoperability Profiles (IOPs) Status

- RAS-Ground IOP Standard
 - IOP V2 published February 2016
 - IOP V3 under development – target publish date December 2017
- IOP Instantiations
 - Instantiations included in upcoming RFPs:
 - MTRS Inc II
 - CRS(I)
 - RCIS
 - Upcoming IOP Instantiation developments:
 - SMET
 - Leader Follower
 - CRS(H)
 - EOD Robotic Payload

IOP V3 Priorities

- Define optimal level of interoperability & modularity for SMET & ground vehicle applique autonomy strategy
 - Enable evolutionary upgrade of autonomy kit to support continued advancement
 - Do not compromise industry's ability to innovate
 - Focus on SMET & Heavy Tactical Vehicles
 - Resolve concerns over safety criticality (JAUS/Ethernet vs. J1939/CAN vs. others)



IOP V3 Development

- Current WGs
 - Test & Compliance
 - Payloads & Platforms – looking at some new connector options
 - XML
 - Autonomy & Behaviors
 - Comms
 - Controls
 - Applique
 - Weapons
- J1939
 - Starting a J1939 Profiling Rules document
 - Leveraging message sets being generated in TARDEC Autonomous Ground Resupply (AGR) project
- New JAUS Messages
 - Dead Man Switch
 - Loading Specifications (i.e. CG for trailer, etc.)
 - Driver Assist / Driver Warning
 - Convoy Specification

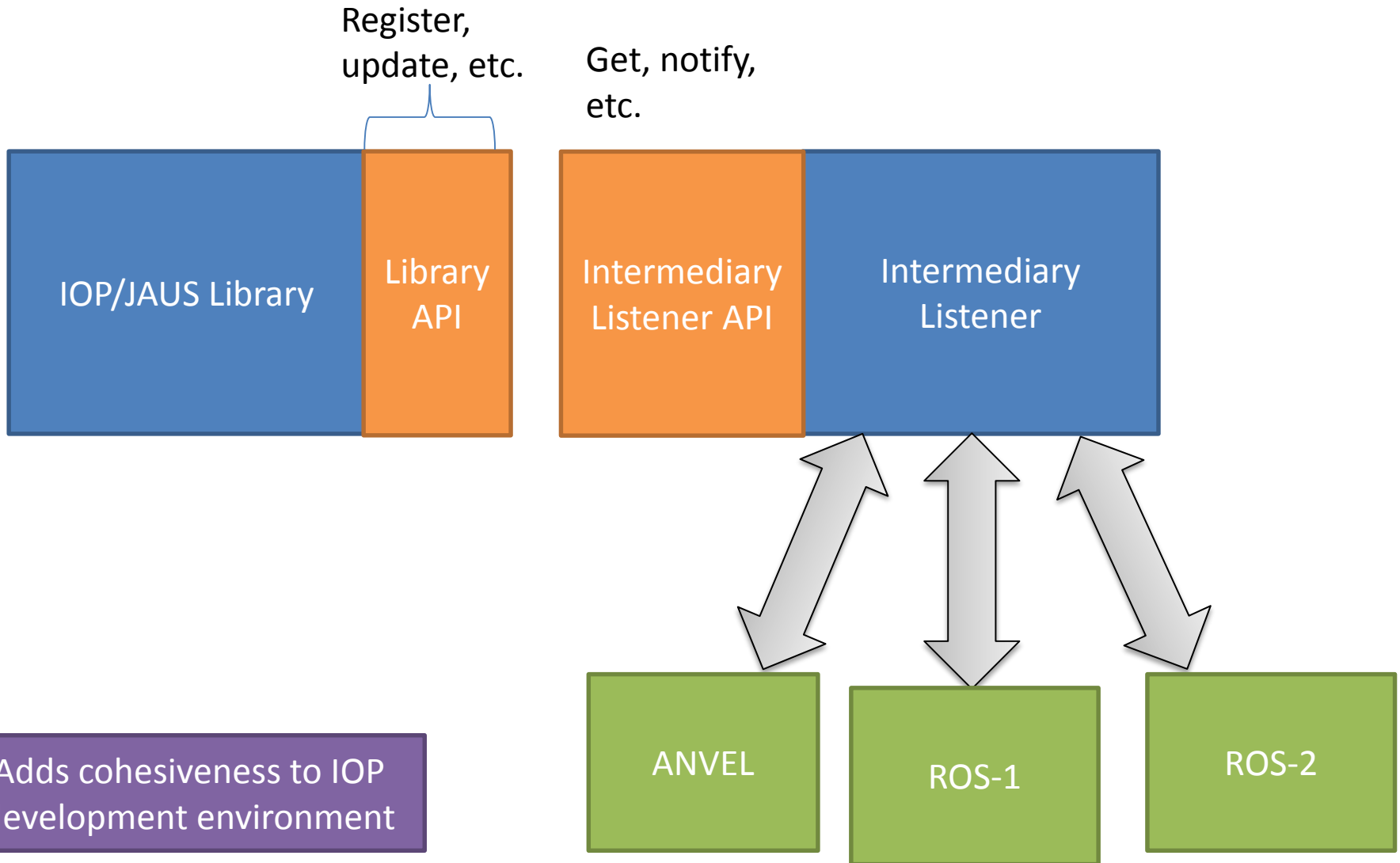


TARDEC IOP Activities

- TARDEC RAS-G IOP Lab
 - Conformance Verification Tool (CVT) v2.4.2-1.beta released Nov 2016
 - CVT March 2017 update will no longer be “beta”
- Joint Communications Architecture for Unmanned Systems (JCAUS) testing in IOP lab – January 2017
- Common CBRN Sensor Interface (CCSI)
 - Obtaining tools in house & determining how to test
- AEODRS
 - Formally assessing how well AEODRS complies w/ IOP
 - Conducting comparison between IOP tools & AEODRS tools
- IOP/JAUS Library & IOP to X Bridge



IOP/JAUS Library & IOP to X Bridge





Agenda

- RAS-G IOP Update
- **ROS-M Update**
- NAMC/VRA Standard Cycle Projects



Robotic Operating System – Military (ROS-M) Update

- ROS-M development being led by TARDEC
 - Goals include software re-use, increased & improved competition, higher development efficiency, common development environment & new capabilities
- TARDEC working closely w/ industry partners in ROS-M WGs for Business Process, Cybersecurity, Software Process & Software Stack
- IOP/ROS-M Activities
 - IOP to ROS-M Bridging
 - Determining how to profile current IOP to ROS-M
 - Evaluating long term utility to migrating toward Data Distribution Service (DDS) architecture ala ROS-M
- PM FP working w/ TARDEC on long term plan for ROS-M & software repositories



TARDEC ROS-M Development Phases

Complete

Phase I:
Concept
Exploration

Concept Definition Report

- Determine scope of project
- Identify potential programmatic and technical issues
- Identify potential business models

Complete

Phase II:
Concept
Definition

Working Group Reports

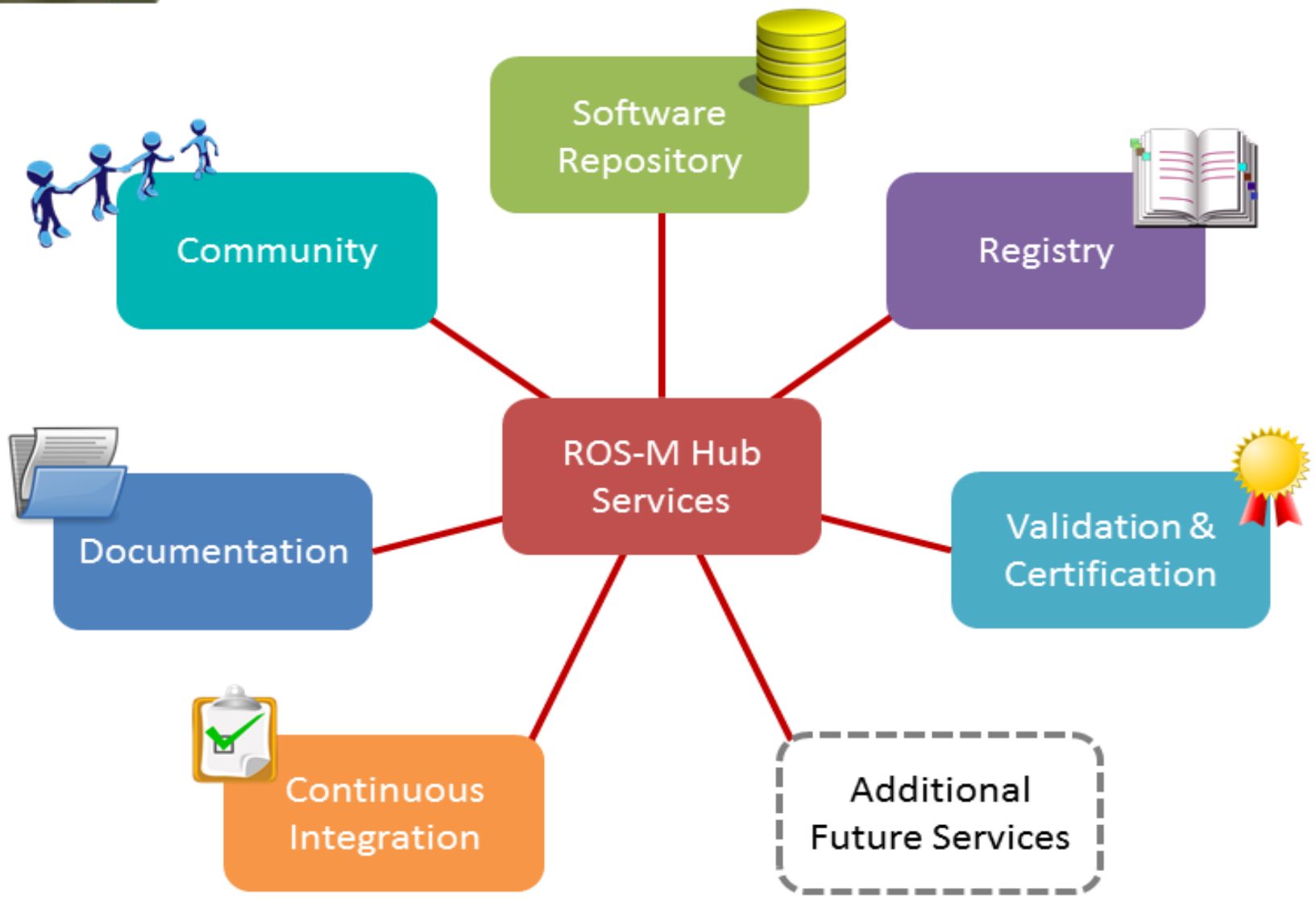
- Refined Concept Definition Report
- Community of Interest for ROS-M Development
- Technical Implementation Suggestions
- ROS-M Business considerations

Funded

Phase III:
Detailed
Implementation
Planning and
Execution

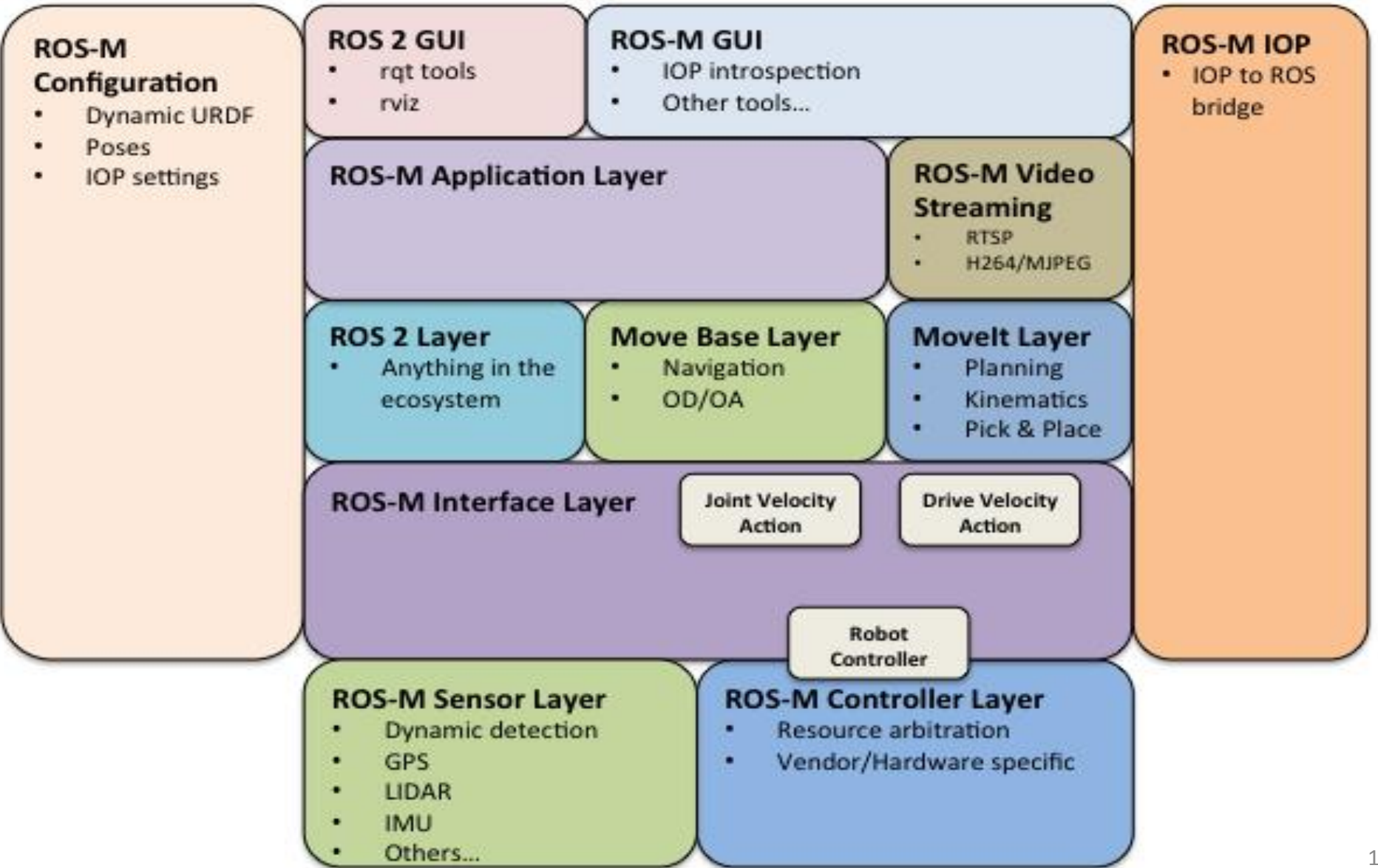
- Alpha ROS-M Security Features
- Alpha ROS-M Repo & Registry
- Initial ROS-M Components
- Prototype Demonstration

Future
Development:
Further
Refinement and
Development





ROS-M Conceptual Model





ROS-M Development Environment

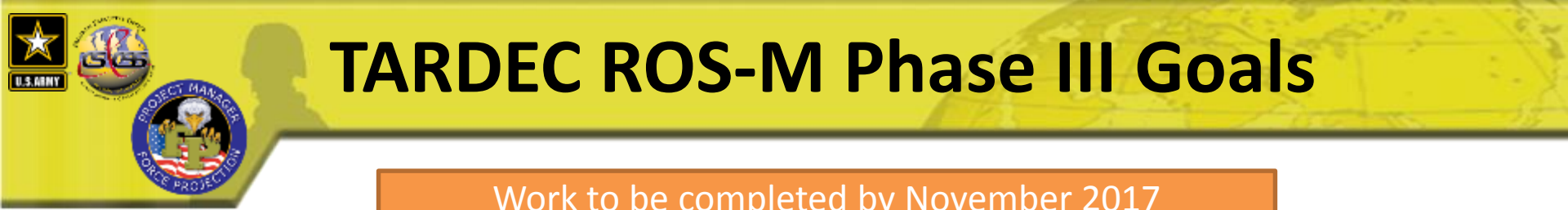
Candidate Solutions

ROS-M Hub Service	Candidate Solution
Software Repository	Git (https://git-scm.com/) GitLab Community Edition (https://about.gitlab.com/) Bitbucket (self-hosted) (https://bitbucket.org/) GitHub Enterprise (self-hosted) (https://enterprise.github.com/)
Registry	Custom solution based on extending rosindex (http://rosindex.github.io/)
Validation	
Continuous Integration	Jenkins (https://jenkins.io/) BuildBot (http://buildbot.net/) GitLab CI (https://about.gitlab.com/gitlab-ci/) Travis CI (https://travis-ci.org/)
Documentation	Confluence (self-hosted) (https://www.atlassian.com/software/confluence) MediaWiki (https://www.mediawiki.org/wiki/MediaWiki) Trac (https://trac.edgewall.org/) ROS Wiki (http://wiki.ros.org/)
Community	ROS Answers (http://answers.ros.org/questions/) Stack Overflow: (http://stackoverflow.com/) Askbot: (http://askbot.org/doc/index.html) AnswerHub: (http://www.dzonesoftware.com/products/answerhub-question-answer-software) Question2Answer: (http://www.question2answer.org/)



ROS-M Candidate Metadata Contents

- **Package name** – The name used within the development environment for the ROS-M package. Example: “cv_bridge”
- **Level of Maturity** – The state of the package (experimental, development, stable, deprecated)
- **Last Updated Date** - The date when this package was last modified.
- **Maintainer(s)** – The names and contact info for the current maintainers of this package. This can be a company name or individuals.
- **Author(s)** – The names of the original authors of this package. This can be a company name or individuals.
- **License** – The license associated with this package. Example: “BSD”
- **Link to source and/or binaries** – Links pointing to where a developer can find the source code, binaries, or an associated store/site where they can be purchased
- **Package description** – A short description of what the package offers in terms of functionality.
- **Package compatibility** - What versions of ROS-M does this package work with?
- **Metatags** – Metatags associated with this package.
- **Link to tutorial(s)** – Tutorials may take up several pages and should be separate from the main package page
- **Link to bug/feature tracker** – A link to where bugs and features are tracked for this particular package. This may be an external link.
- **Overview section** – a more in-depth explanation of the package (as compared to the “package description” section). This may include discussion of various algorithms or approaches.
- **Example usage section** – A short, basic example use of the package.
- **Link to the code API** – link to the API documentation, such as a Doxygen page. This might be an external link.
- **Link to a FAQ/QA** – Link to the QA site with tags relevant to this package (if possible)
- **Changelog** – This lists any changes that are made for each revision for this package
- **Change list** – This lists any changes that are made in each stack release for this package
- **Reviews** – API or code review meeting notes as well as the ability to request a new review of this package
- **Dependencies** – A list of packages that this package depends upon
- **Used by** – A list of packages that are known to use this package
- **Projects and Programs** - a list of military projects and programs that are known to have used this package.
- **Version** - specify which version of package is being used



TARDEC ROS-M Phase III Goals

Work to be completed by November 2017

Repository/Registry and Software Build Infrastructure Development

- Establish the registry/repository infrastructure to store and distribute ROS-M–based software components and tools

ROS Core Testing

- Perform and test the application of the Defense Information Systems Agency security guides and SE Linux with ROS

Autonomy Component Development and Maturation

- Update and/or create the software packages necessary for a typical military autonomy vehicle

Prototype System Development and Integration

- Application of the autonomy components on a physical asset (GVRBot & Representative Industry Bot)

Demonstration of System Functions

- Create a demo to present to the PM and other interested parties that highlights the benefits ROS-M brings/will bring to future military development



Agenda

- RAS-G IOP Update
- ROS-M Update
- **NAMC/VRA Standard Cycle Projects**



Vehicle Robotics Alliance (VRA) FY18 Projects

- VRA Annual Cycle Projects
 - These projects are solicited once per year and include the following' year's projects of interest to the Government. Project requests can be planned funded or unfunded efforts. Unfunded efforts that receive a #1 or #2 rating on their submitted proposal will go into the basket for a period of 36 months and may be funded if money becomes available and the proposal is selected from the basket by the Government.
- PM FP generated topics are summarized on the following slides



PM FP Generated VRA FY18 Topics

- Architecture
 - ASM-18-01: Robotics Cybersecurity Prototype Solutions
 - ASM-18-02: Nett Warrior Control of Robotic Systems
 - ASM-18-03: Net Enabled Robotic Prototype Solutions
- Autonomy
 - Autonomy-18-01: Tactical Wheeled Vehicle Autonomy Risk Reduction Solutions
 - Autonomy-18-11: Robotic Enhancement Program
 - Autonomy-18-13: D7R-11 (Dozer) Teleoperation Capability
- Collaboration
 - Collaboration-18-02: Extended Range Communications
- T&E
 - T&E-18-01: Common Robotic Test Standards



PM FP Generated VRA FY18 Topics (cont.)

- External
 - External-18-01: Additive Manufacturing of Robotic Parts
 - External-18-02: EOD Robotic Payload (ERP) Risk Reduction Prototype Solutions
- M&S
 - M&S-18-01: Video Technical Manuals Prototype Solutions
- Platforms
 - Platforms-18-02: SMET Risk Reduction Prototype Solutions
 - Platforms-18-03: CRS(H) Risk Reduction Prototype Solutions
 - Platforms-18-05: D7R-II (Dozer) Sifter Attachment
- Powertrain
 - Powertrain-18-01: Alternative Power Sources & Charging Methods for Robotic Systems



Summary

- IOP V2 instantiation documents in current RFPs
- IOP V3 development ongoing for publish December 2017
- ROS-M development continues – Phase III results expected November 2017
- PM FP FY18 VRA topics currently out for industry review – topics give gov't the ability to fund approved projects at any time during FY18-FY20 if funding becomes available