



U.S. ARMY



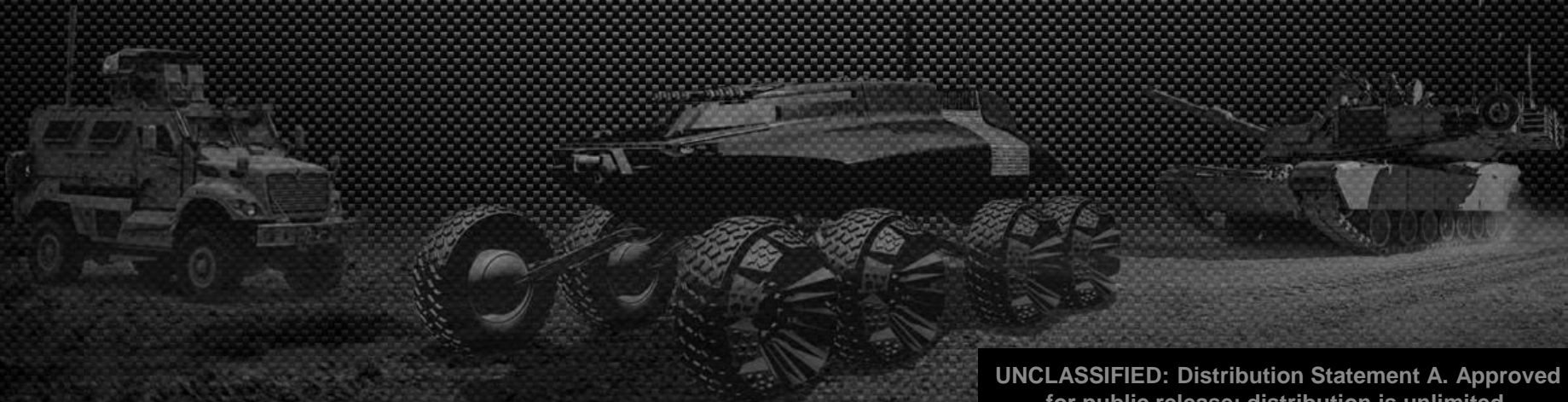
U.S. ARMY  
**RDECOM**



U.S. ARMY TANK AUTOMOTIVE RESEARCH, DEVELOPMENT AND ENGINEERING CENTER

# Introduction to Robotic Technology Kernel (RTK)

TARDEC Ground Vehicle Robotics



UNCLASSIFIED: Distribution Statement A. Approved  
for public release; distribution is unlimited.

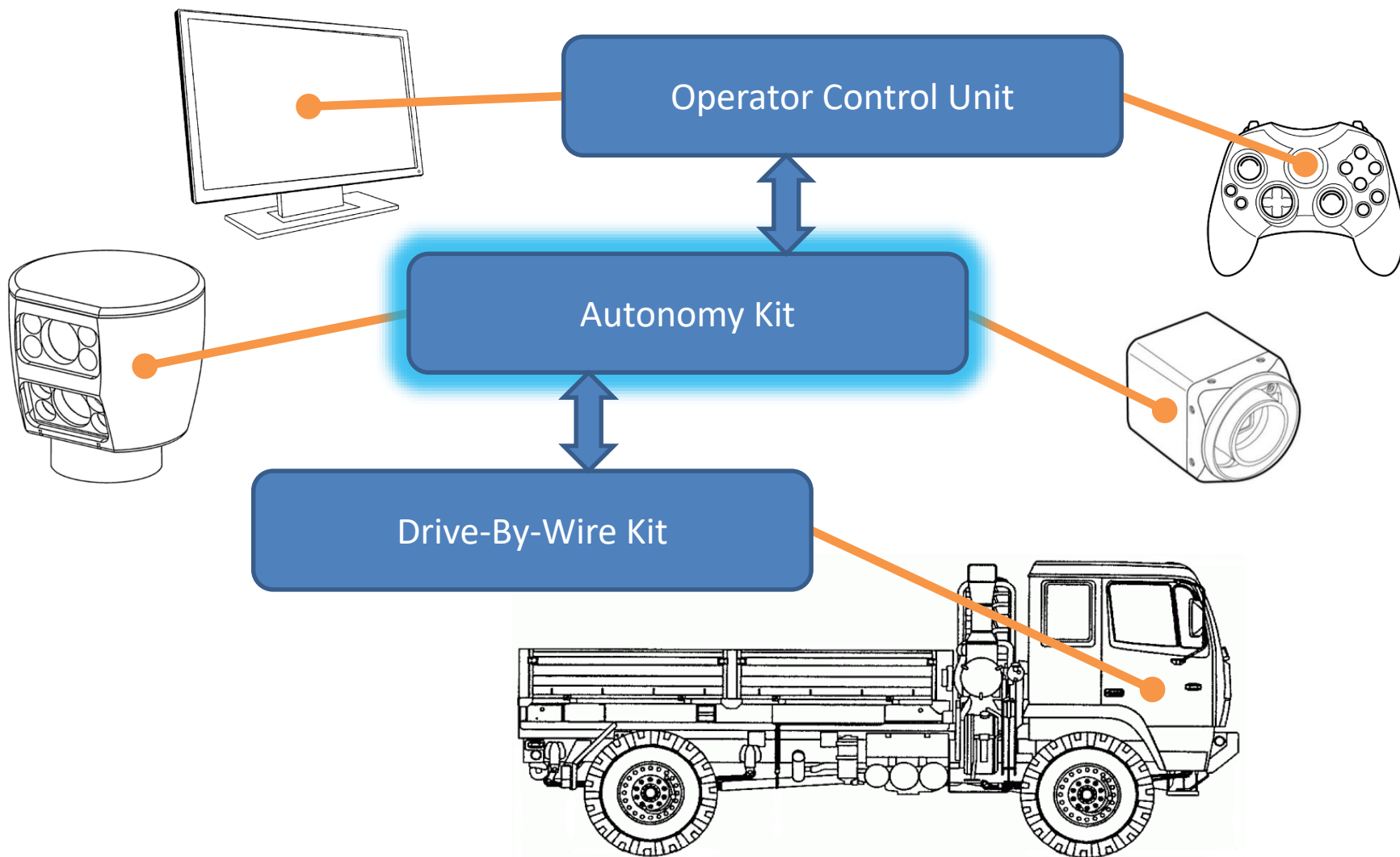


- RTK Overview
- Architecture
- Vehicle Configuration
- Development
- Useful Tools



# RTK Overview

# Autonomy Kit/Drive-by-Wire Kit Concept



# Robotic Technology Kernel (RTK)



The Robotic Technology Kernel (RTK) is a Robot Operating System (ROS)-based modular autonomy software library for S&T development that provides a set of common robotic capabilities across a variety of platforms and efforts.

DSAT

MUER

TORVICE

CAAR

Wingman

CoVeR

**TARDEC  
Robotic  
Technology  
Kernel**



## Cross-Platform

New RTK capabilities are enjoyed by all RTK-enabled platforms

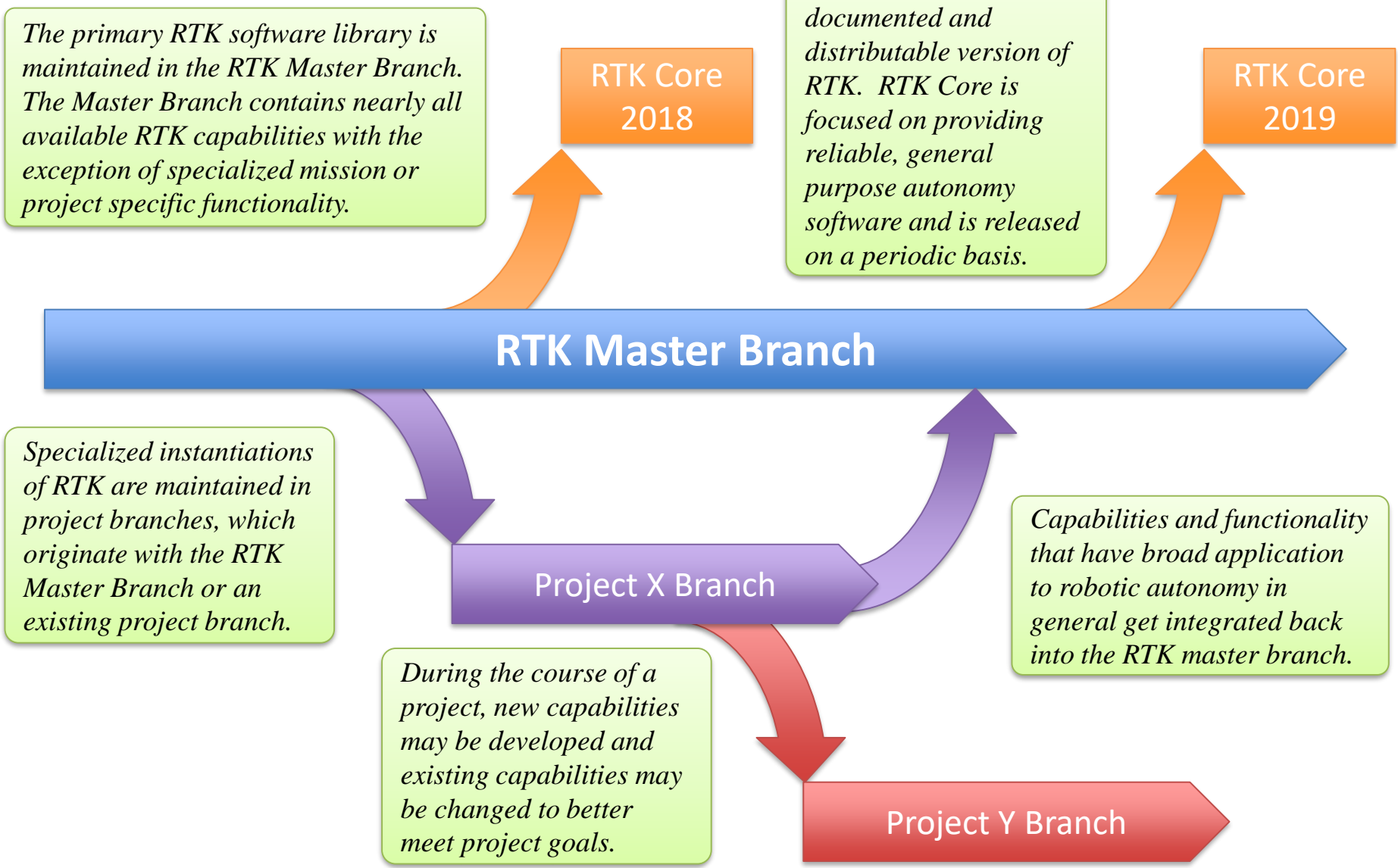
## Cross-Effort

Efforts deliver new capabilities to the RTK, which in turn is leveraged for new efforts.

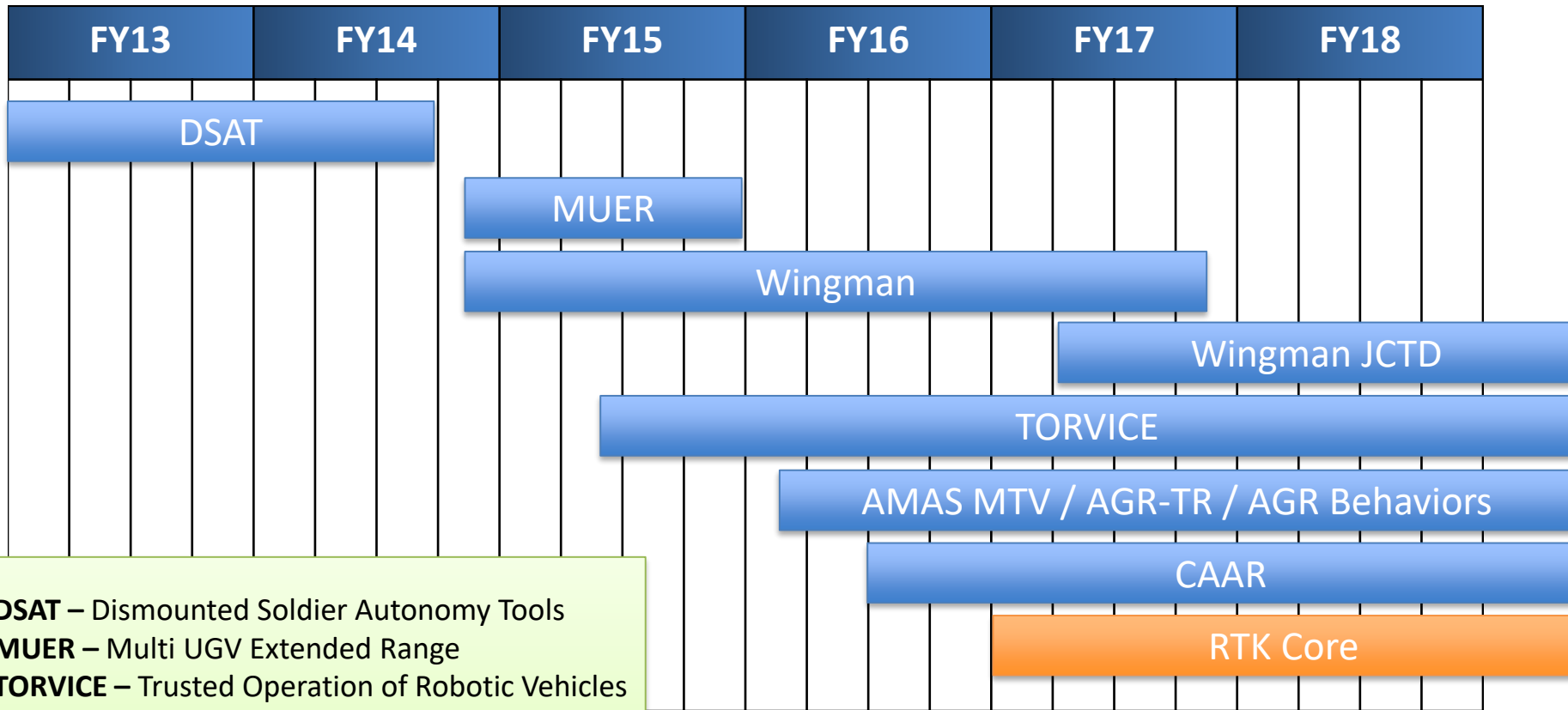
## Cross-Controller

Any RTK-enabled platform can be controlled by any RTK compatible controller or WMI.

# RTK Software Development Concept



# RTK Past and Present



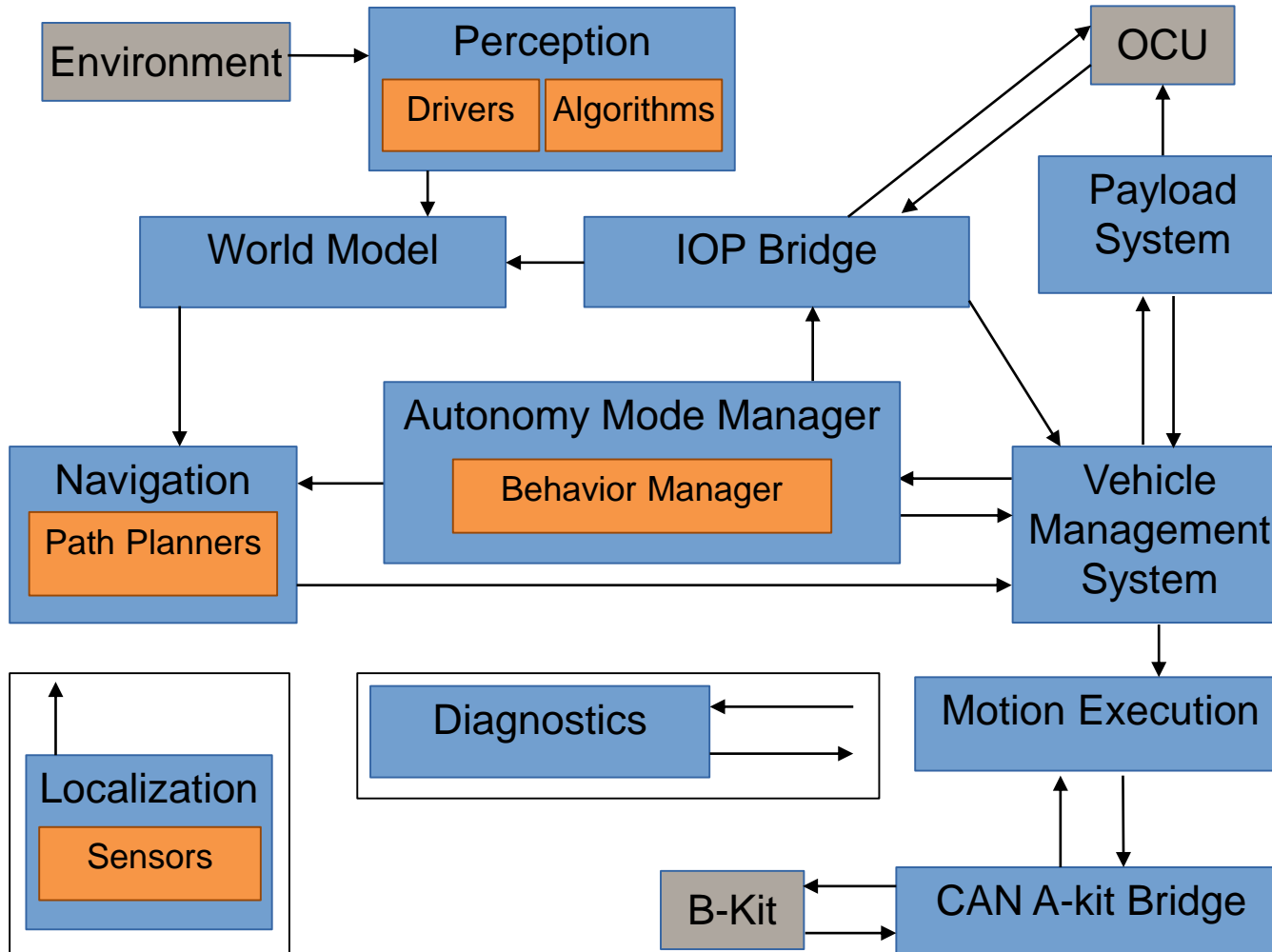
**DSAT** – Dismounted Soldier Autonomy Tools  
**MUER** – Multi UGV Extended Range  
**TORVICE** – Trusted Operation of Robotic Vehicles in a Contested Environment  
**AMAS** – Autonomous Mobility Applique Kit  
**AGR** – Autonomous Ground Resupply  
**TR** – Tactical Resupply  
**CAAR** – Coalition Assured Autonomous Resupply

In addition to the development efforts listed here, RTK has been and continues to be used on several other projects to provide autonomous capability.



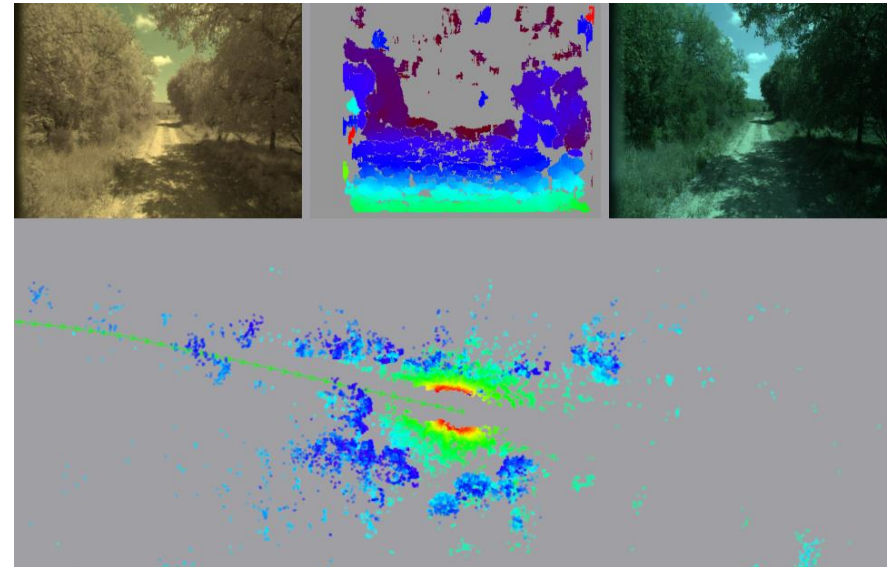
# Architecture





This subsystem contains the sensor drivers and algorithms that detect and interpret things of interest from the environment.

- Stereo Cameras
- LIDARs
- Ultra Wideband Radios (UWBs)
- RADAR
  
- Output:
  - Disparity
  - Material Classification
  - Ground Point Clouds
  - Non-Ground Point Clouds



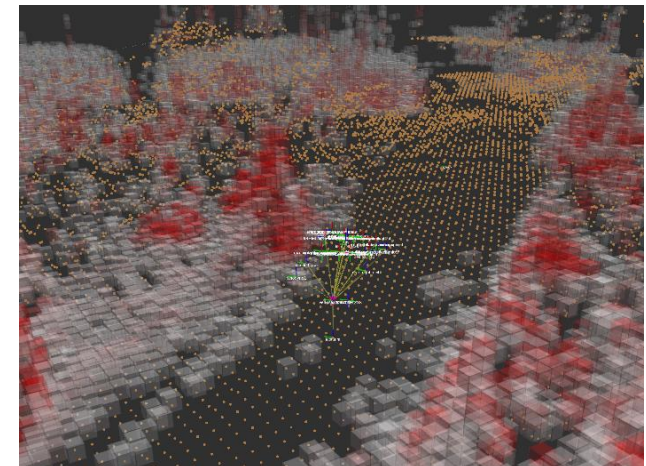
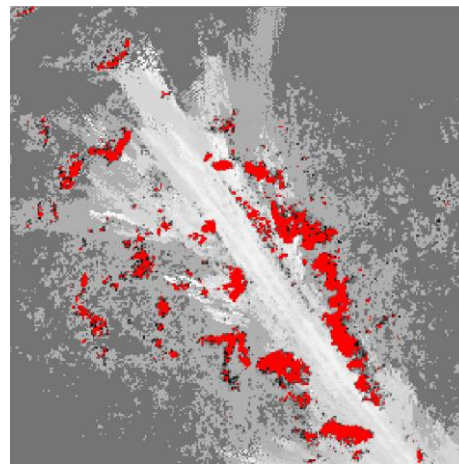
This subsystem fuses data from multiple sensors to provide both relative and absolute pose of the vehicle.

- GNSS
    - GPS
    - DRTK
    - DGPS
  - Wheel Speed Encoders
  - IMU
  - Gyro
  - RANGER
  - Visual Odometry
- 
- Output:
    - far\_field -> near\_field -> vehicle TF tree
      - map -> odom -> base\_link in standard ROS terms
    - far field and near field odometry



This subsystem is used for storing and fusing data from multiple sources. Services are provided to request data as a costmap or list of dynamic objects and zones.

- Input:
  - Plugin-based sensor fusion from Perception & Localization
  - “No go” zones
  - Known Obstacles
- Internal operations:
  - Octree-based sparse voxel map
  - Persistent Map
- Output:
  - Voxels
  - 2D Costmaps



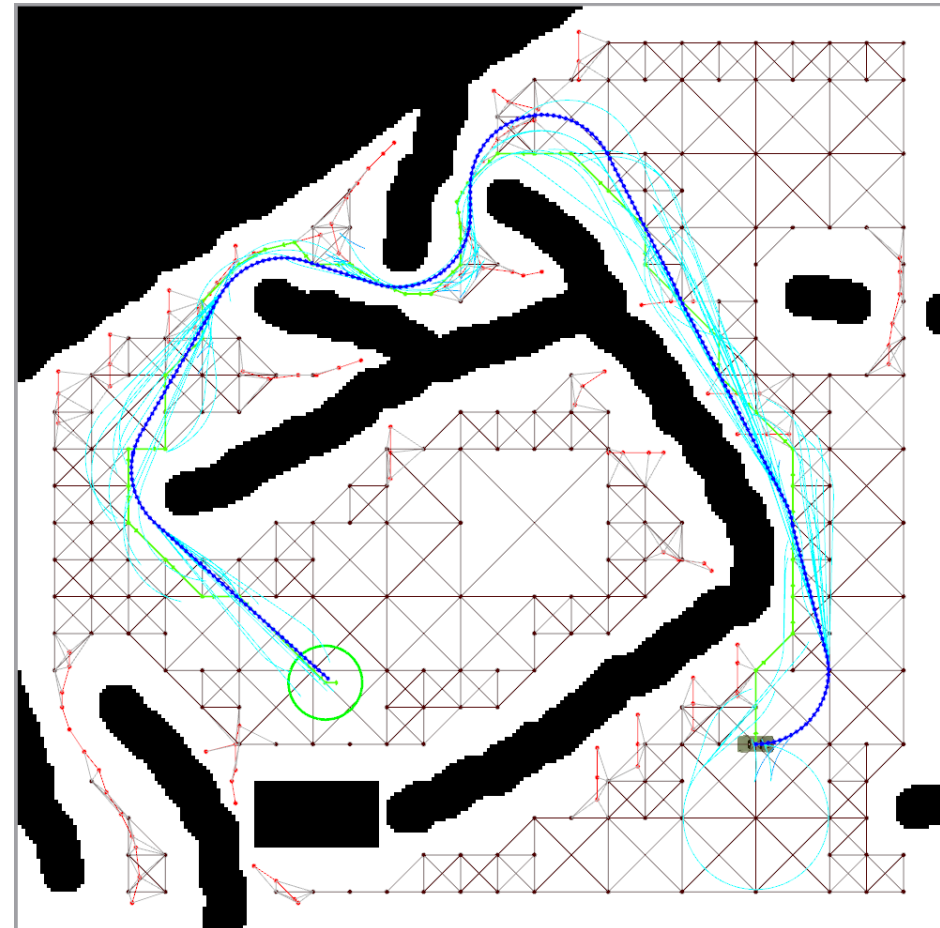


This subsystem configures and monitors behavior modules.

- Input:
  - Mode commands
- Output:
  - Path planner configuration

This subsystem generates paths and speed/steering commands to guide the vehicle to its destination.

- Input:
  - Filtered 2D Costmaps
  - Vehicle Odometry
- Internal operations:
  - Behavior switch
  - Path Planning
    - A\*
    - Maverick
    - Vaquero
    - Vaquerito
- Output:
  - Speed & Curvature Setpoint



This subsystem monitors overall health and ensures commands are safe to perform.

- Input:
  - Direct teleoperation
  - Mode commands
  - Camera control and configuration
  - Speed and steering
  - Vehicle status
- Output:
  - Mode confirmation
  - Verified speed and steering

This subsystem handles primitive control and status between the autonomy kit and the drive-by-wire kit.

- Input:
  - Speed & Curvature Setpoint
  - Vehicle Odometry
- Output:
  - Drive-by-Wire Commands





- Diagnostics
  - This subsystem provides a way to monitor the health of hardware in the system.
- IOP Bridge
  - This subsystem is the interface between the autonomy kit and the operator control unit.
- CAN-A-Kit Bridge
  - This subsystem handles traffic to and from the CAN bus.
- Payload System
  - This subsystem handles control, configuration, and status of pan/tilt, camera, and remote weapon systems that can be controlled by the operator.



# Vehicle Configuration

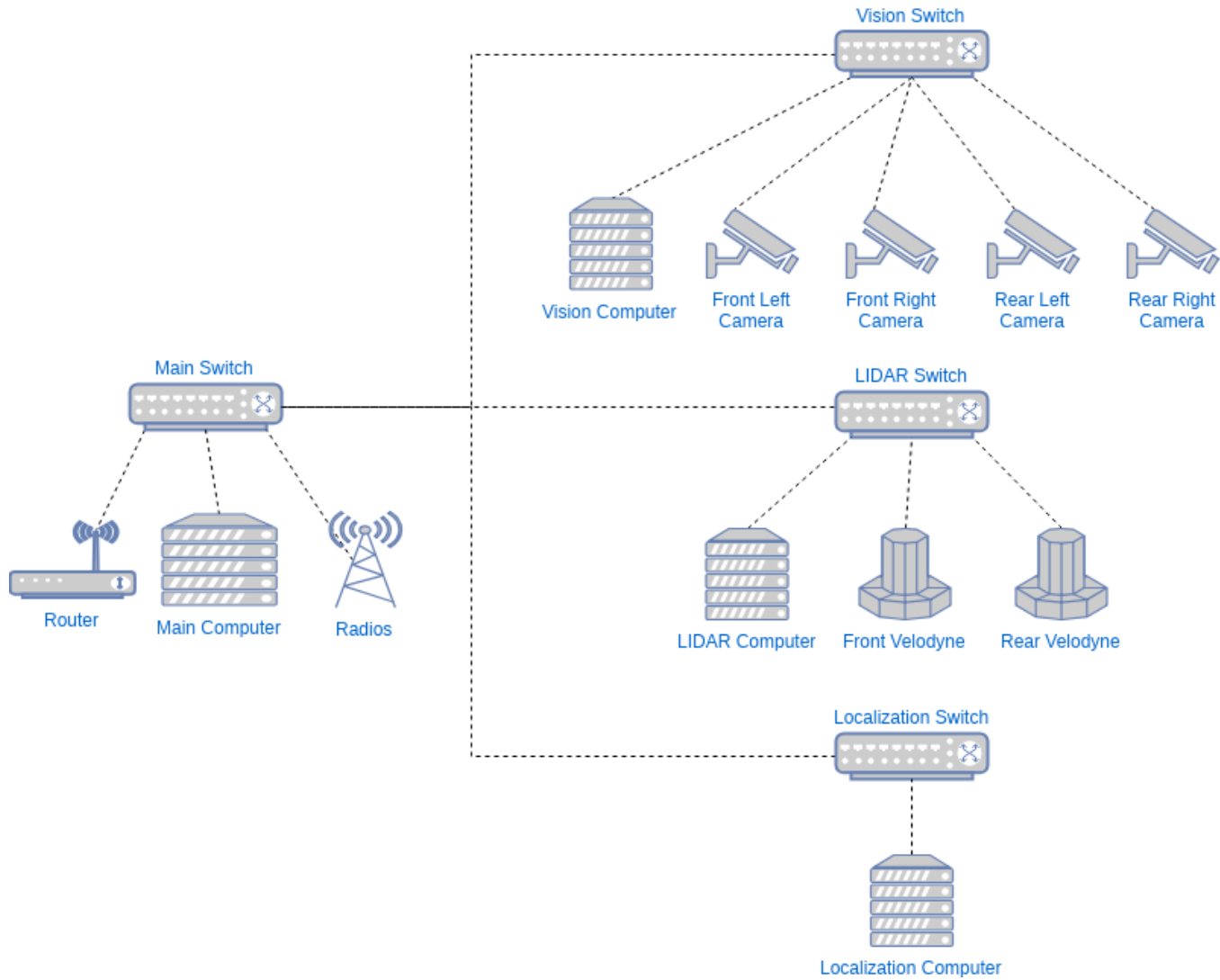
- GNSS\*
  - GPS
  - RTK
  - DGPS
- IMU\*
- LIDAR\*
- Wheel Speed Encoders\*\*
- Gyro\*\*
- Stereo Cameras
- Teleop Cameras
- Data Radios
- UWB Radios
- RADAR

\* Required

\*\* Not required, but a really good idea

- Main
  - ROS Master
  - Navigation
  - World Model
  - Motion Execution
  
- Localization
  - Hardware Drivers
    - GPS
    - IMU
    - Wheel Speed
  - State Estimation
  
- Vision
  - Camera Drivers
  - Image Rectification
  - Disparity Segmentation
  - Material Classification
  
- LIDAR
  - Velodyne Drivers
  - Ground Segmentation

# Networking





# Development

- Create a ROS workspace
  - Clone repositories
  - Configure the workspace
  - Install dependencies
  - Build it
- Make changes
  - Edit code
  - Deploy to a vehicle
  - Test the changes
  - Push them to a new branch
  - Create a Merge Request to request a code review

- “deploy” script copies compiled binaries to /opt/rtk/indigo (or kinetic)
- Ubuntu 14.04 / ROS Indigo:
  - Upstart script in /etc/init/dsat.conf
  - sudo service dsat start/stop/restart
- Ubuntu 16.04 / ROS Kinetic:
  - Systemd unit in /etc/systemd/system/dsat.service
  - sudo systemctl start/stop/restart dsat





- DI2E: Widely-Available Release Repository
  - RTK Core
  - WMI Core
- Dismount-Git: Primary Development Repository
  - RTK Master Branch
  - RTK Project Branches
- Developer Handbook: Environment Setup, Development Workflow, Style Guide, etc.
- RTK User Guide
  - Architecture Diagrams, Flow Charts, Use Cases
- RTK Architecture Guide
  - Low-level ROS package documentation
  - ROS node APIs



# Useful Tools

- Stereo Camera Calibration
  - Create calibration boards
  - Collect data
  - Perform calibration
  - Apply calibration files
  
- LIDAR Alignment
  - Necessary if using more than one LIDAR
  - Launch the calibration system
  - Run rviz to monitor progress
  - Use calibration system to manually align LIDARs
  - Save calibration values into platform launch file

- `roslaunch mrzr_8803 mrzr_8803.launch playback:=true`
- Runs all nodes on the local computer
- Disables hardware drivers
- Useful for playing back bag files to debug behavior
- `rosbag play -clock -r 0.5 rtk_raw_test_track_playback*.bag \`  
`local_xy_origin:=local_xy_old`

- Like rosconsole but better

SWRI Console (classifier, disparity\_segmenter, image\_coupler, imperx\_stereo\_pipeline, classifier, disparity\_segmenter, image\_coupler, imperx\_stereo\_pipeline) - + x

File Edit Options

```

/magic_carpet/carpet (5)
/navigation/active_route_source (1)
/navigation/comms_map (12)
/navigation/costmap_processor (1)
/navigation/costmap_source (1)
/navigation/kinodynamic_planner (2)
/navigation/maverick_map_processor
/navigation/maverick_planner_drive
/navigation/path_following_control
/navigation/straight_path_node (38)
/navigation/vaquerito_planner_node
/navigation/waypoint_planner (20)
/navigation/nodelets (30)
/play_1524775268084810454 (4)
/rear/disparity_segmenter (21)
/rear/stereo_heartbeat_aggregator
/route_interpolator (7)
/route_manager (6)
/route_transformer (14)
/vaquero_route_recorder (18)

```

Clear All Clear Messages

Severity

- Debug
- Info
- Warn
- Error
- Fatal
- Follow Newest Messages

Connected to ROS Master

```

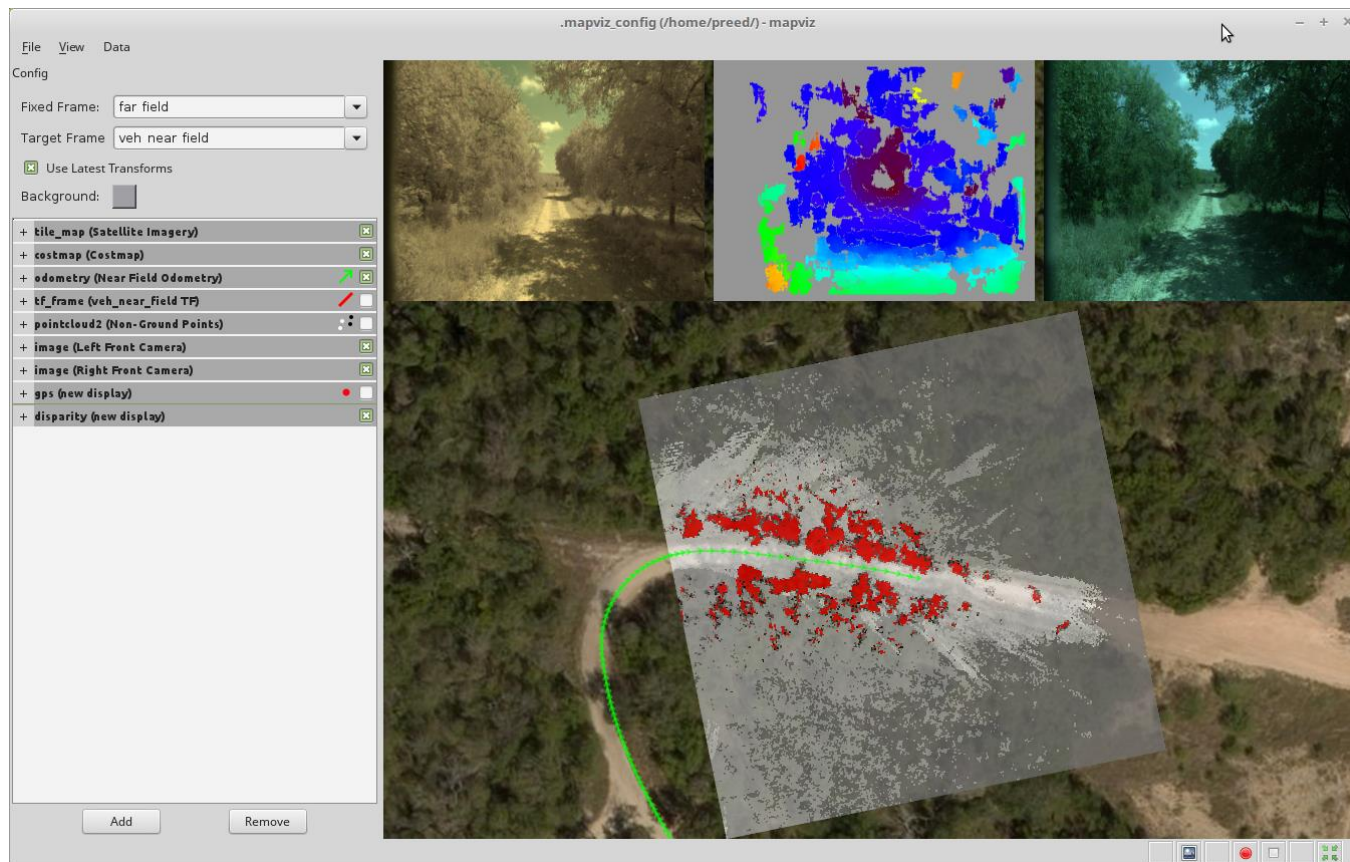
[I 423210: 34:26:467] Read parameter prefilter_cap = 13
[I 423210: 34:26:487] Read parameter correlation_window_size = 10
[I 423210: 34:26:497] Read parameter correlation_window_size = 10
[I 423210: 34:26:518] Read parameter min_disparity = 1
[I 423210: 34:26:518] Read parameter min_disparity = 1
[I 423210: 34:26:548] Read parameter disparity_range = 64
[I 423210: 34:26:548] Read parameter disparity_range = 64
[I 423210: 34:26:568] Read parameter uniqueness_ratio = 2
[I 423210: 34:26:568] Read parameter uniqueness_ratio = 2
[I 423210: 34:26:588] Read parameter texture_threshold = 5
[I 423210: 34:26:588] Read parameter texture_threshold = 5
[I 423210: 34:26:608] Read parameter speckle_size = 270
[I 423210: 34:26:608] Read parameter speckle_size = 270
[I 423210: 34:26:638] Read parameter speckle_range = 14
[I 423210: 34:26:638] Read parameter speckle_range = 14
[I 423210: 34:26:648] Read parameter normalize_rows = false
[I 423210: 34:26:648] Read parameter normalize_rows = false
[I 423210: 34:26:658] Read parameter uncrop_disparity = false
[I 423210: 34:26:658] Read parameter uncrop_disparity = false
[I 423210: 34:26:668] Read parameter check_for_errors = true
[I 423210: 34:26:668] Read parameter check_for_errors = true
[I 423210: 34:26:678] Read parameter min_intensity_range = 30.000000
[I 423210: 34:26:678] Block Matcher: OpenCV Block Matcher
[I 423210: 34:26:678] Read parameter min_intensity_range = 30.000000
[I 423210: 34:26:678] Block Matcher: OpenCV Block Matcher
[I 423210: 34:26:708] Read parameter queue_size = 5
[I 423210: 34:26:708] Read parameter queue_size = 5
[I 423210: 34:26:739] Read parameter approximate_sync = true
[I 423210: 34:26:739] Read parameter approximate_sync = true
[I 423210: 34:26:759] Read parameter max_interval = 0.050000
[I 423210: 34:26:759] Read parameter max_interval = 0.050000
[I 423210: 34:27:349] Reconfigure request received.
[I 423210: 34:27:359] Reconfigure request received.
[I 423210: 34:27:359] Block Matcher: OpenCV Block Matcher

```

Include

Exclude

- Top-down ROS visualization tool, similar to rviz



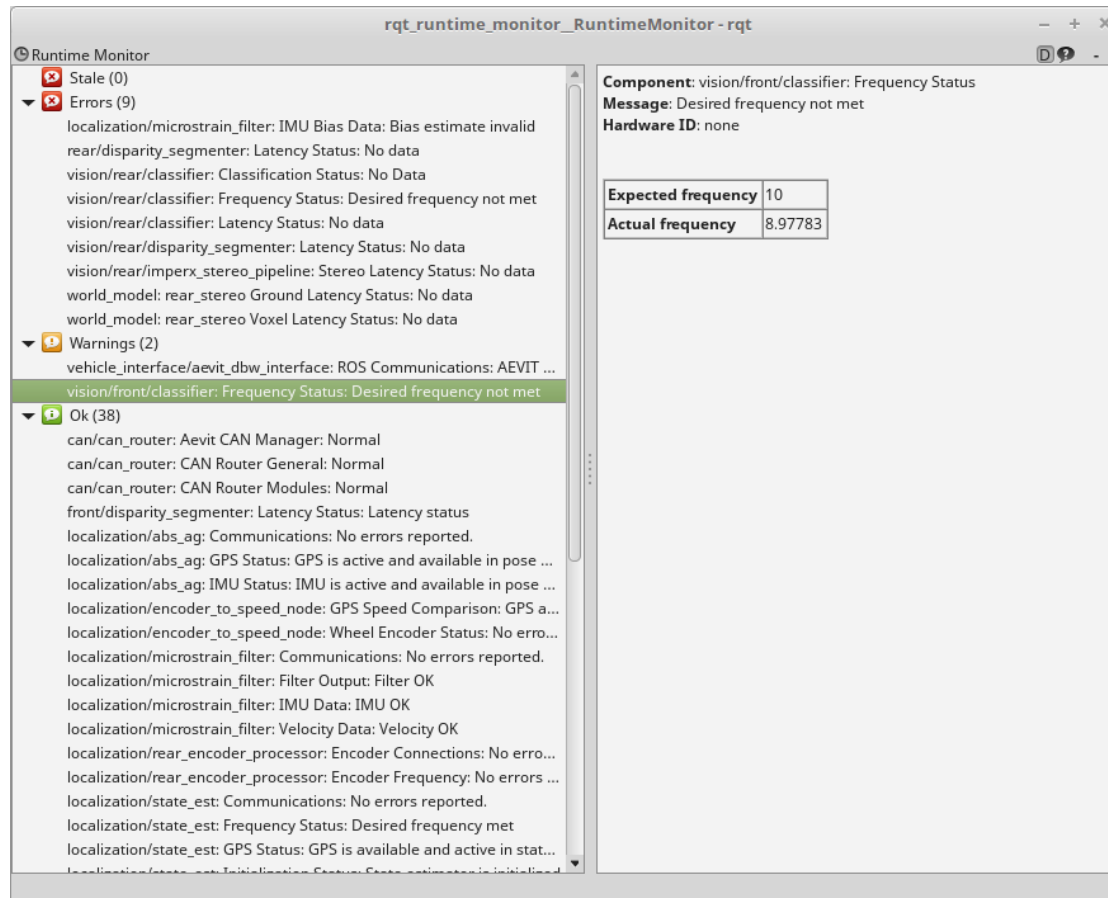
- Searchable, inspectable database of bag metadata

The screenshot displays the Bag Database web application interface. At the top, there is a search bar with the text "sensor\_msgs/Image" and a search button. Below the search bar, there are checkboxes for search criteria:  Filename,  Description,  Tags,  Path,  Location,  Vehicle,  Message Types, and  Topic Names. The search results are displayed in a table with the following columns: File Name, Location, Vehicle, Duration (s), Start Time, End Time, Size (MB), and Tags. The table lists 5283 bags, with the first few rows showing details for files like "10\_meter\_data.orig.bag", "151\_video\_2015-01-13-14-05-58.bag", and "15m\_15to15\_angles\_2017-01-10-18-39-19.bag".

File Name	Location	Vehicle	Duration (s)	Start Time	End Time	Size (MB)	Tags
10_meter_data.orig.bag			26.831	10/12/2017 14:49:16	10/12/2017 14:49:43	7105.866	
151_video_2015-01-13-14-05-58.bag	US-90, San Antonio, ...		140.428	1/13/2015 14:05:48	1/13/2015 14:08:09	8204.480	
15m_15to15_angles_2017-01-10-18-39-19.bag			111.549	1/10/2017 18:39:19	1/10/2017 18:41:11	15097.855	
15m_all_angles_2017-01-10-17-55-11.bag			213.754	1/10/2017 17:55:12	1/10/2017 17:58:45	38460.480	
2011-05-16-cars1.bag			70.214	5/16/2011 10:25:50	5/16/2011 10:27:00	3754.655	
2011-05-16-cars2.bag			79.150	5/16/2011 10:25:50	5/16/2011 10:27:00	3754.655	
2011-05-16-cars3.bag			83.880	5/16/2011 10:25:50	5/16/2011 10:27:00	3754.655	
2011-05-16-cars4.bag			7.789	5/16/2011 10:25:50	5/16/2011 10:27:00	3754.655	
2011-05-16-people1.bag			137.341	5/16/2011 10:25:50	5/16/2011 10:27:00	3754.655	
2011-05-16-image_raw_111279				5/16/2011 10:25:50	5/16/2011 10:27:00	3754.655	
2011-05-16-1				5/16/2011 10:25:50	5/16/2011 10:27:00	3754.655	
2011-05-16-2				5/16/2011 10:25:50	5/16/2011 10:27:00	3754.655	
2011-06-2-1				6/28/2011 08:18:01	6/28/2011 08:18:08	47.959	
2011-06-2-2				6/28/2011 08:21:37	6/28/2011 08:25:05	1934.985	
2011-06-2-3				6/28/2011 08:25:14	6/28/2011 08:30:19	2738.199	
2013-03-1-1				3/14/2013 16:28:15	3/14/2013 16:29:33	1077.798	
2013-03-1-2				3/14/2013 16:28:15	3/14/2013 16:29:34	1077.439	
2013-09-1-1				9/13/2013 08:18:01	9/13/2013 08:18:08	47.959	
2013-09-1-2				9/13/2013 08:21:37	9/13/2013 08:25:05	1934.985	
2013-09-1-3				9/13/2013 08:25:14	9/13/2013 08:30:19	2738.199	

The interface also features a navigation pane on the right side with various icons for file operations. Two preview windows are open: one showing a raw image of a road scene and another showing a satellite map with a red path overlaid on it. A status bar at the bottom indicates "2120 errors" and "Bag Scanner: Scanning for new bag files in /bags."

- Displays diagnostic messages
- Useful on a live vehicle to tell if all the hardware is working









# Questions?