

ROS 2: What's new?

February 2020

Jacob Perron

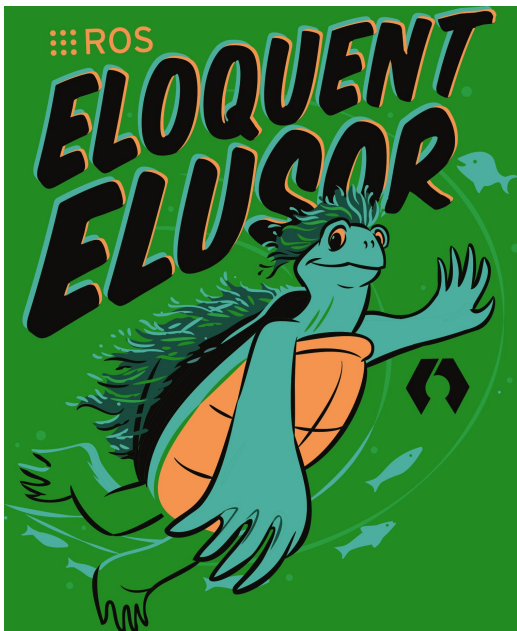
Just in case...

- What is ROS?
 - <http://www.ros.org/>
- Who is Open Robotics?
 - We work on ROS, Gazebo, ROS 2, Ignition
 - <https://www.openrobotics.org>
- What is the difference in ROS and ROS 2?
 - https://design.ros2.org/articles/why_ros2.html
- What is the difference in Gazebo and Ignition?
 - <https://ignitionrobotics.org/docs/all/overview#what-is-ignition-robotics>

Introductions

- Presenter:
 - Jacob Perron <jacob@openrobotics.org>
@jacobperron
- People who actually did the work:
 - ROS 2 development team @ Open Robotics
 - You (the community)

Eloquent Elusor
November 22nd, 2019



Foxy Fitzroy (LTS)
May 23rd, 2020

WIP: Foxy logo

ros2 interface

- Author:
 - Siddharth Kucheria @skucheria
- Code:
 - <https://github.com/ros2/ros2cli/tree/master/ros2interface>
- What is ros2interface?
 - Tool for showing information about ROS interfaces
 - Supports actions and IDL
 - Replaces ros2msg and ros2srv
 - ROS 1: [rosmsg](#) and rossrv

ros2 interface

```
$ ros2 interface -h
```

Show information about ROS interfaces

optional arguments:

-h, --help show this help message and exit

Commands:

list	List all interface types available
package	Output a list of available interface types within one package
packages	Output a list of packages that provide interfaces
proto	Output an interface prototype
show	Output the interface definition

ros2 doctor

- Author:
 - Claire Wang @claireyywang
- Code:
 - <https://github.com/ros2/ros2cli/tree/master/ros2doctor>
- What is ros2doctor?
 - Tool for ROS system check-up
 - ROS 1: [roswtf](#)

ros2 doctor

- Reports:

- RMWs installed
- Package versions (Foxy)
- ROS distribution info
- Network info
- Topic list

```
$ ros2 doctor --report
```

- Checks:

- Warn about newer upstream package versions (Foxy)
- Warn about unsupported ROS distribution
- Warn about missing loopback/multicast IP address
- Warn about unconnected subscriptions and publishers

```
$ ros2 doctor
```


ros2 doctor

- Hello?
 - Check if multiple hosts can communicate with each other
 - Available in ROS Foxy

```
$ ros2 doctor hello
```

Markup for launch

- Authors:
 - Michel Hidalgo @hidmic
 - Ivan Santiago Paunovic @ivanpauno
 - William Woodall @wjwwood
- ROSCon 2019:
 - <https://vimeo.com/379127678>
- What is launch?
 - Configure and start your ROS system
 - ROS 1: [roslaunch](#)
- Code:
 - <https://github.com/ros2/launch>

Markup for launch

```
from launch import LaunchDescription
from launch_ros.actions import Node

def generate_launch_description():
    return LaunchDescription([
        Node(package='demo_nodes_cpp', node_executable='talker', output='screen'),
        Node(package='demo_nodes_cpp', node_executable='listener', output='screen'),
    ])
```

Markup for launch

```
from launch import LaunchDescription
from launch_ros.actions import Node

def generate_launch_description():
    return LaunchDescription([
        Node(package='demo_nodes_cpp', node_executable='talker', output='screen'),
        Node(package='demo_nodes_cpp', node_executable='listener', output='screen'),
    ])
```

```
<launch>
  <node pkg="demo_nodes_cpp" exec="talker" output="screen" />
  <node pkg="demo_nodes_cpp" exec="listener" output="screen" />
</launch>
```

Markup for launch

```
from launch import LaunchDescription
from launch_ros.actions import Node

def generate_launch_description():
    return LaunchDescription([
        Node(package='demo_nodes_cpp', node_executable='talker', output='screen'),
        Node(package='demo_nodes_cpp', node_executable='listener', output='screen'),
    ])
```

```
<launch>
  <node pkg="demo_nodes_cpp" exec="talker" output="screen" />
  <node pkg="demo_nodes_cpp" exec="listener" output="screen" />
</launch>
```

```
launch:
- node: {pkg: "demo_nodes_cpp", exec: "talker", output: "screen"}
- node: {pkg: "demo_nodes_cpp", exec: "listener", output: "screen"}
```

Markup for launch

- How does it work?
 - Map markup to an intermediate representation

```
<launch>  
  <node pkg="demo_nodes_cpp" exec="talker" output="screen" />  
  <node pkg="demo_nodes_cpp" exec="listener" output="screen" />  
</launch>
```

```
Entity (type: 'launch')  
  children: [  
    Entity(type: 'node'),  
    Entity(type: 'node')  
  ]
```

Parser

```
LaunchDescription([  
  Node(...),  
  Node(...),  
])
```

Markup for launch

- TODO

- Parsers for more launch entities
 - E.g. LifecycleNode, ComposableNode, Events
- Documentation
- API review

```
<!-- Eloquent -->  
<node node-name="my_talker" pkg="demo_nodes_cpp" exec="talker" />  
<!-- Foxy -->  
<node name="my_talker" pkg="demo_nodes_cpp" exec="talker" />
```

launch_testing

- Authors:
 - Peter Baughman @pbaughman
 - William Woodall @wjwwood
 - Michel Hidalgo @hidmic
- ROSCon 2019:
 - <https://vimeo.com/378683186>
- Code:
 - <https://github.com/ros2/launch>

launch_testing

- Successor to [rostest](#) from ROS 1
 - Good:
 - Launch ROS system with tests; exit when tests finish
 - Testing in isolation
 - Bad:
 - Tests might run before ROS system is ready
 - Crashes are not detected by tests
 - Testing process output is not straight forward
- Bad things are fixed in ROS 2

launch_testing

```
@pytest.mark.launch_test
def generate_test_description():
    launch_description = IncludeLaunchDescription(
        PythonLaunchDescriptionSource('talker_listener.launch.py'))

    return LaunchDescription([
        launch_description,
        ReadyToTest(),
    ])
```

Continued on next slide ...

launch_testing

```
class TestTalkerListener(unittest.TestCase):

    def test_talker_talks(self, proc_output):
        proc_output.assertWaitFor(
            process='talker',
            expected_output="Publishing: 'Hello World: 3'",
            timeout=5.0
        )

    def test_listener_hears(self, proc_output):
        proc_output.assertWaitFor(
            process='listener',
            expected_output='I heard: [Hello World: 3]',
            timeout=5.0,
        )
```

launch_testing

- Run tests with ros2test:

```
$ ros2 test test_talker_listener.py
```

- Alternatively, with pytest:

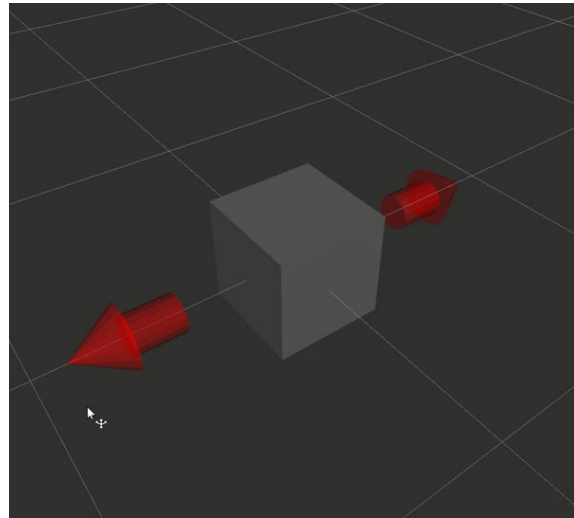
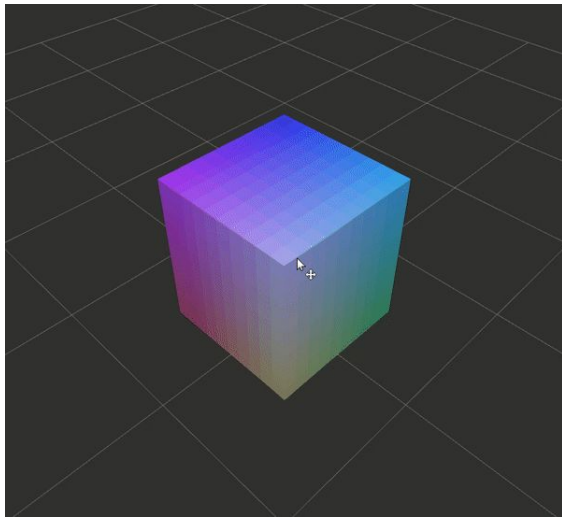
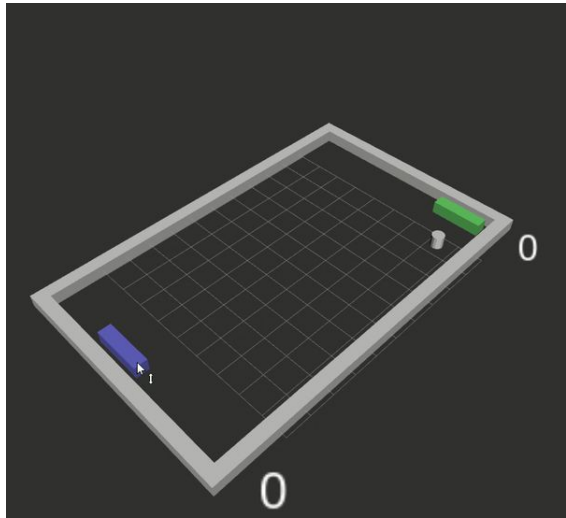
```
$ pytest test_talker_listener.py
```

launch_testing

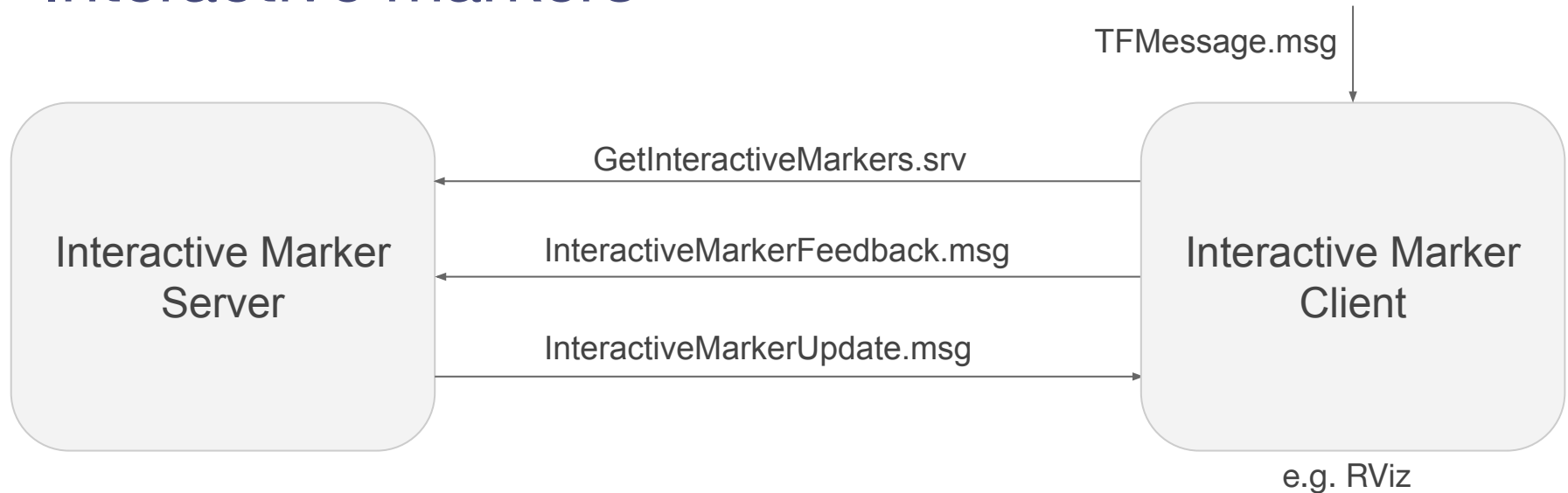
- TODO
 - Run tests in separate processes (e.g. gtests)
 - Documentation and tutorials
 - Record ROS messages for post-shutdown tests
 - Frontend support

Interactive markers

- Author:
 - Jacob Perron @jacobperron
- Code:
 - https://github.com/ros-visualization/interactive_markers/tree/ros2
 - https://github.com/ros-visualization/visualization_tutorials/tree/ros2/visualization_marker_tutorials



Interactive markers



- Many clients can connect to a server
- Updates contain sequence numbers
 - If client misses an update, request all markers

Interactive markers

- Example server usage:

```
auto node = std::make_shared<rclcpp::Node>("foo_node");
interactive_markers::InteractiveMarkerServer server("bar_marker_namespace", node);
visualization_msgs::msg::InteractiveMarker marker;
// ... populate interactive marker
server.insert(marker, std::bind(&feedbackCallback, _1));
server.applyChanges();
rclcpp::spin(node);
```

- Server will update pose of markers by default
 - Feedback callback is optional
- Python support for server (not client)

Interactive markers

- TODO:
 - ROS 1 bridge support
 - Add more tests
 - Documentation
 - Python interactive marker client

Other things in Eloquent Elusor

- RMW support for loaned messages
 - ROSCon 2019: <https://vimeo.com/379127778>
- Navigation2: <https://github.com/ros-planning/navigation2>
- Passing node parameters from CLI
- Stream logging macros (e.g. RCLCPP_INFO_STREAM)
- And more:
<https://index.ros.org/doc/ros2/Releases/Release-Eloquent-Elusor/>

What to expect in Foxy Fitzroy?

- Node to DDS participant mapping improvement
- Changes to rclcpp and rclpy API
- Python API for rosbag2
- MoveIt! 2: <https://github.com/ros-planning/moveit2>
- ROS 2 Java
- Better documentation 🙌
- Node description language: <https://github.com/ros2/design/pull/266>
- Package quality categories: <https://github.com/ros-infrastructure/rep/pull/218>
- And more: <https://github.com/ros2/ros2/issues/830>

More links

- Link to these slides:
 - <http://tiny.cc/ros2-whats-new>
- Eloquent Release page:
 - <https://index.ros.org/doc/ros2/Releases/Release-Eloquent-Elusor/>
- Foxy Release page (WIP):
 - <https://index.ros.org/doc/ros2/Releases/Release-Foxy-Fitzroy/>
- Contribute:
 - ROSCon 2018: <https://vimeo.com/292699328>

Questions?