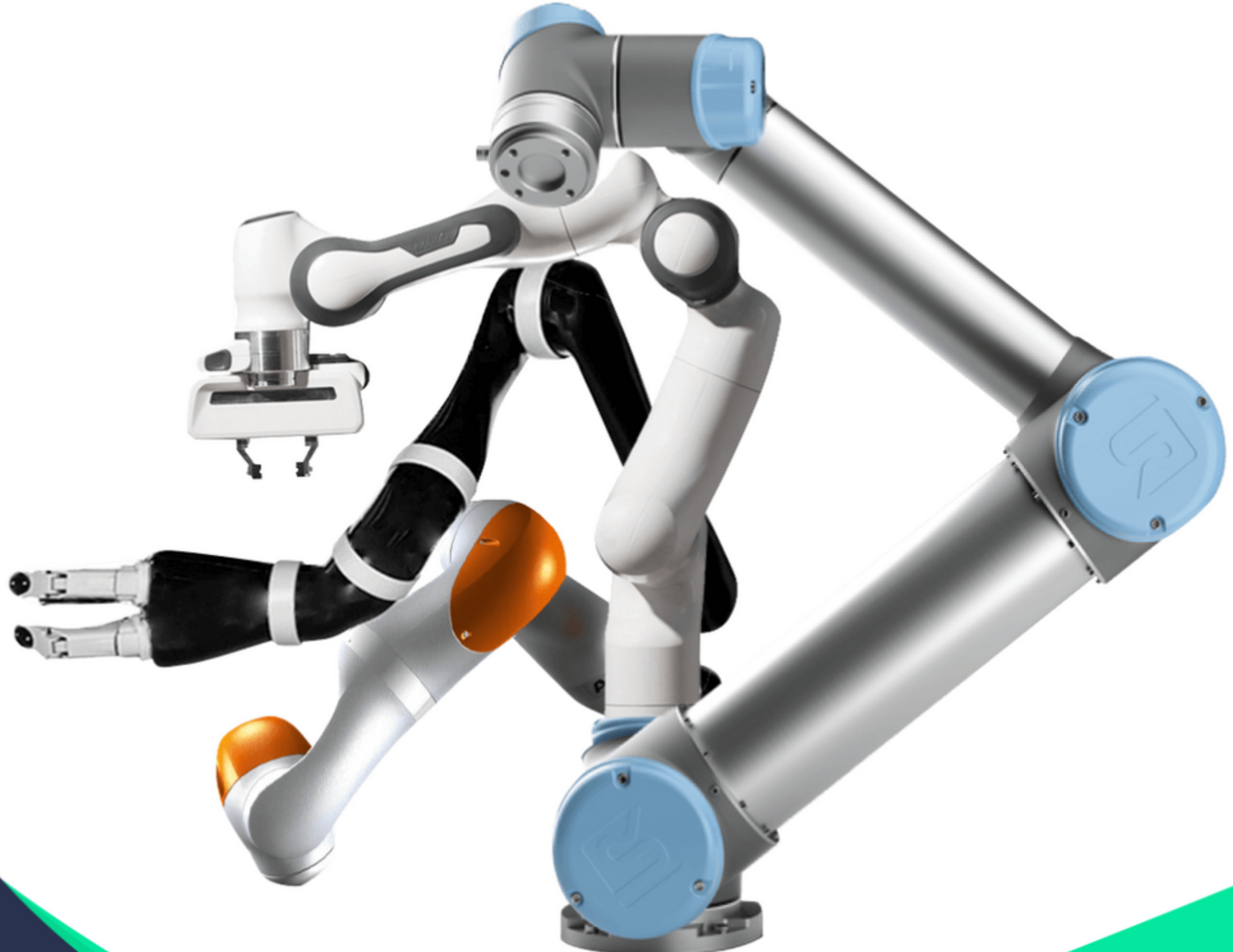
PICKNIK



Retro ROS 2 Launch

Make ROS Easy Again!

September 5, 2023

Tyler Weaver
Staff Software Engineer
tyler@picknik.ai

Tyler Weaver

- Racing Kart Driver
- MoveIt Maintainer
- Rust Evangelist
- Docker Skeptic

Comfort Zone: ROS 1 Launch

```
<launch>
  <arg name="pipeline" default="ompl" />
  <arg name="capabilities" default=""/>

  <node name="move_group" pkg="moveit_ros_move_group" type="move_group"
    output="screen">
    <param name="default_planning_pipeline" value="$(arg pipeline)" />
    <param name="capabilities" value="$(arg capabilities)" />
  </node>
</launch>
```

Trigger: ROS 2 Launch

```python
import os
from launch import LaunchDescription
from launch.actions import DeclareLaunchArgument, OpaqueFunction
from launch.substitutions import LaunchConfiguration, PathJoinSubstitution
from launch.conditions import IfCondition, UnlessCondition
from launch_ros.actions import Node
from launch_ros.substitutions import FindPackageShare
from launch.actions import ExecuteProcess
from ament_index_python.packages import get_package_share_directory
from moveit_configs_utils import MoveItConfigsBuilder


def generate_launch_description():
    declared_arguments = []
    declared_arguments.append(
        DeclareLaunchArgument("rviz_config",
            default_value="kinova_moveit_config_demo.rviz",
            description="RViz configuration file",
        )
    )
    return LaunchDescription(declared_arguments + [OpaqueFunction(function=launch_setup)])


def launch_setup(context, *args, **kwargs):
    launch_arguments = {
        "robot_ip": "xxx.yyy.zzz.www",
        "use_fake_hardware": "true",
        "gripper": "robotiq_2f_85",
        "dof": "7",
    }

    moveit_config = (
        MoveItConfigsBuilder("gen3", package_name="kinova_gen3_7dof_robotiq_2f_85_moveit_config")
        .robot_description(mappings=launch_arguments)
        .trajectory_execution(file_path="config/moveit_controllers.yaml")
        .planning_scene_monitor(publish_robot_description=True, publish_robot_description_semantic=True)
        .planning_pipelines(pipelines=["ompl", "stomp", "pilz_industrial_motion_planner"])
        .to_moveit_configs()
    )
```

```python
# Start the actual move_group node/action server
run_move_group_node = Node(
    package="moveit_ros_move_group",
    executable="move_group",
    output="screen",
    parameters=[moveit_config.to_dict()],
)

rviz_base = LaunchConfiguration("rviz_config")
rviz_config = PathJoinSubstitution([FindPackageShare("moveit2_tutorials"), "launch"

# RViz
rviz_node = Node(
    package="rviz2",
    executable="rviz2",
    name="rviz2",
    output="log",
    arguments=["-d", rviz_config],
    parameters=[
        moveit_config.robot_description,
        moveit_config.robot_description_semantic,
        moveit_config.robot_description_kinematics,
        moveit_config.planning_pipelines,
        moveit_config.joint_limits,
    ],
)

# Static TF
static_tf = Node(
    package="tf2_ros",
    executable="static_transform_publisher",
    name="static_transform_publisher",
    output="log",
    arguments=["--frame-id", "world", "--child-frame-id", "base_link"],
)

# Publish TF
robot_state_publisher = Node(
    package="robot_state_publisher",
    executable="robot_state_publisher",
    name="robot_state_publisher",
    output="both",
    parameters=[moveit_config.robot_description],
)

nodes_to_start = [
    rviz_node,
    static_tf,
    robot_state_publisher,
<launch>
  <arg name="robot_ip" default="xxx.yyy.zzz.www" />
  <arg name="use_fake_hardware" default="true" />
  <arg name="gripper" default="robotiq_2f_85" />
  <arg name="dof" default="7" />

  <let name="robot_description"
    value="$(command 'xacro $(find-pkg-share kortex_description)/robots/gen3.xacro
      robot_ip:=$(var robot_ip) use_fake_hardware:=$(var use_fake_hardware)
      gripper:=$(var gripper) dof:=$(var dof)')" />
  <let name="robot_description_semantic"
    value="$(command 'xacro $(find-pkg-share kinova_moveit_config)/config/gen3.srdf')"
  />

  <!-- MoveGroup -->
  <node pkg="moveit_ros_move_group" exec="move_group" output="screen">
    <param name="robot_description" value="$(var robot_description)" />
    <param name="robot_description_semantic" value="$(var robot_description_semantic)" /
    <param from="$(find-pkg-share cpp_parameters)/config/moveit.yaml" />
  </node>
</launch>
```

Crisis: YAML parsing of params

```
<let name="robot_description_semantic"
    value="$(command 'xacro $(find-pkg-share kinova_moveit_config)/config/gen3.srdf
  />
...
<param name="robot_description_semantic" value="$(var robot_description_semantic)"
```

```
' using yaml rules: yaml.safe_load() failed
mapping values are not allowed here
  in "<unicode string>", line 11, column 13:
      <!--GROUPS: Representation of a set of joi ...
                ^
```

- SRDF parameter loaded into variable `robot_description_semantic`
- value attribute is parsed as yaml
- yaml parser fails when it finds `:` character
- Open issue on ros2/launch: https://github.com/ros2/launch/issues/729

Recovery?

- Try quoting string to get it to stop parsing

Recovery?

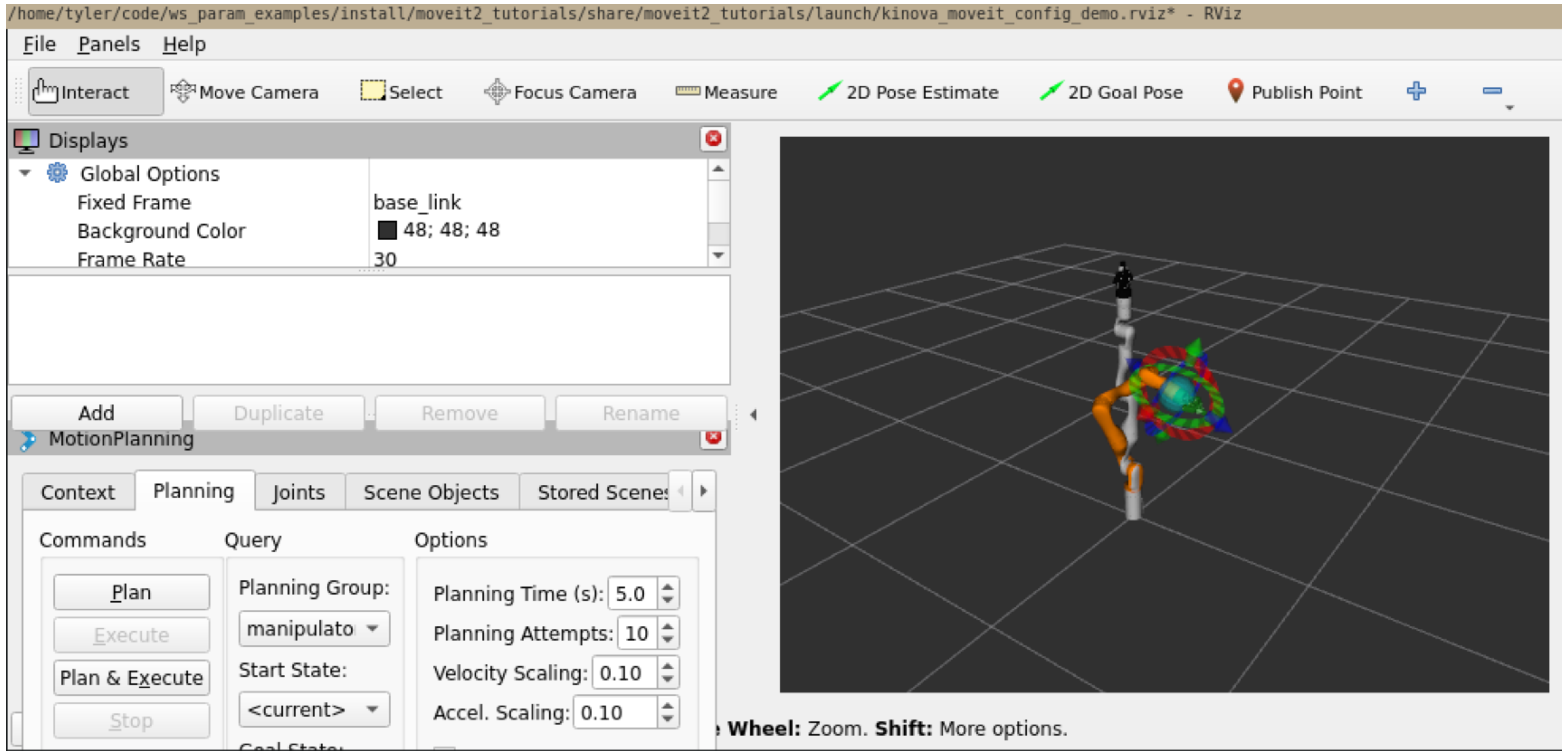
- Try quoting string to get it to stop parsing
- Realize error is parsing comment

Recovery?

- Try quoting string to get it to stop parsing
- Realize error is parsing comment
- Remove `:` character from comment in `srdf` files

Recovery?

- Try quoting string to get it to stop parsing
- Realize error is parsing comment
- Remove `:` character from comment in `srdf` files
- Profit!



/home/tyler/code/ws_param_examples/install/moveit2_tutorials/share/moveit2_tutorials/launch/kinova_moveit_config_demo.rviz* - RViz

Fixed Frame base_link
Background Color ■ 48; 48; 48
Frame Rate 30

Planning Group: manipulato
Start State: <current>

Planning Time (s): 5.0
Planning Attempts: 10
Velocity Scaling: 0.10
Accel. Scaling: 0.10

Wheel: Zoom. Shift: More options.

Better Place: ROS 2 XML Launch

- Launch demo with **43** lines of XML vs **>500** lines of Python
- Single `moveit.yaml` config for MoveIt
- ROS 2 XML Launch Docs: docs.ros.org/en/rolling/How-To-Guides/Migrating-from-ROS1/Migrating-Launch-Files.html
- Comparing Python/XML/YAML: docs.ros.org/en/rolling/How-To-Guides/Launch-file-different-formats.html

Thank You!

PICKNIK