MoveIt Overview

Mark Moll, PhD
Director of Research
PickNik Robotics
mamoll

PICKNIK

Outline

- MoveIt Background
- MoveIt Applications: Land, Sea, Space
- Typical Usage Patterns
- Components
- More Resources

The Dream: A Multi Purpose Robot

Powered by MoveIt and ROS

Willow Garage

MoveIt: A Hardened Motion Planning Platform

arm_navigation

Version 1.0

2010

MoveIt!

Beta

2013

MoveIt!

Version 1.0

2019

MoveIt2

Version 2.0

2020

MoveIt3?

Version 3.0

2022?

A Feature-Rich Ecosystem

PICKNIK

Global Planners
- OMPL
- SBPL
- TrajOpt
- STOMP
- CHOMP

Cartesian Planners
- RobotState
- Descartes
- JogArm
- PilzMotion

Inverse Kinematics
- KDL
- IKFast
- TracIK
- LMA
- BioIK

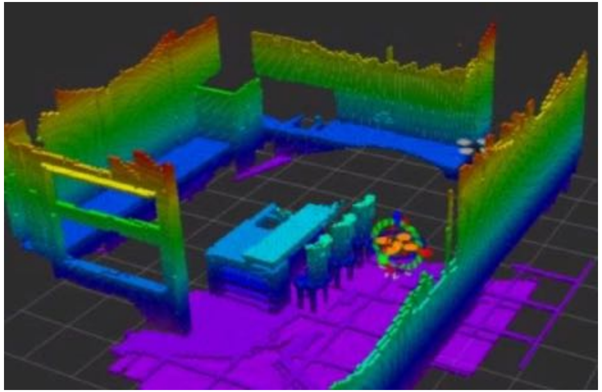
Grasping Libraries
- MoveIt Grasps
- Grasp Pose Detection (GPD)
- Intel OpenVino GPD

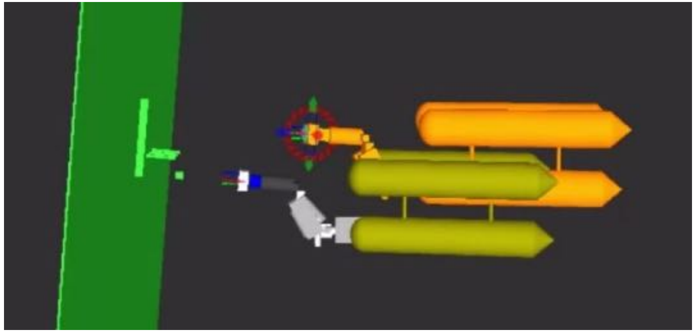
Collision Checking
- Fast Collision Library (FCL)
- Bullet

Perception / Octomap
- Depth Images
- Point Clouds

Flexibility (but also complexity)

- Can handle:
    - Groups of joints
    - Multivariable joints
    - Mimic joints
- Notions of:
    - Cartesian-Space Planning
    - Joint-Space Planning
    - Orientation Constraints
    - Visibility Constraints

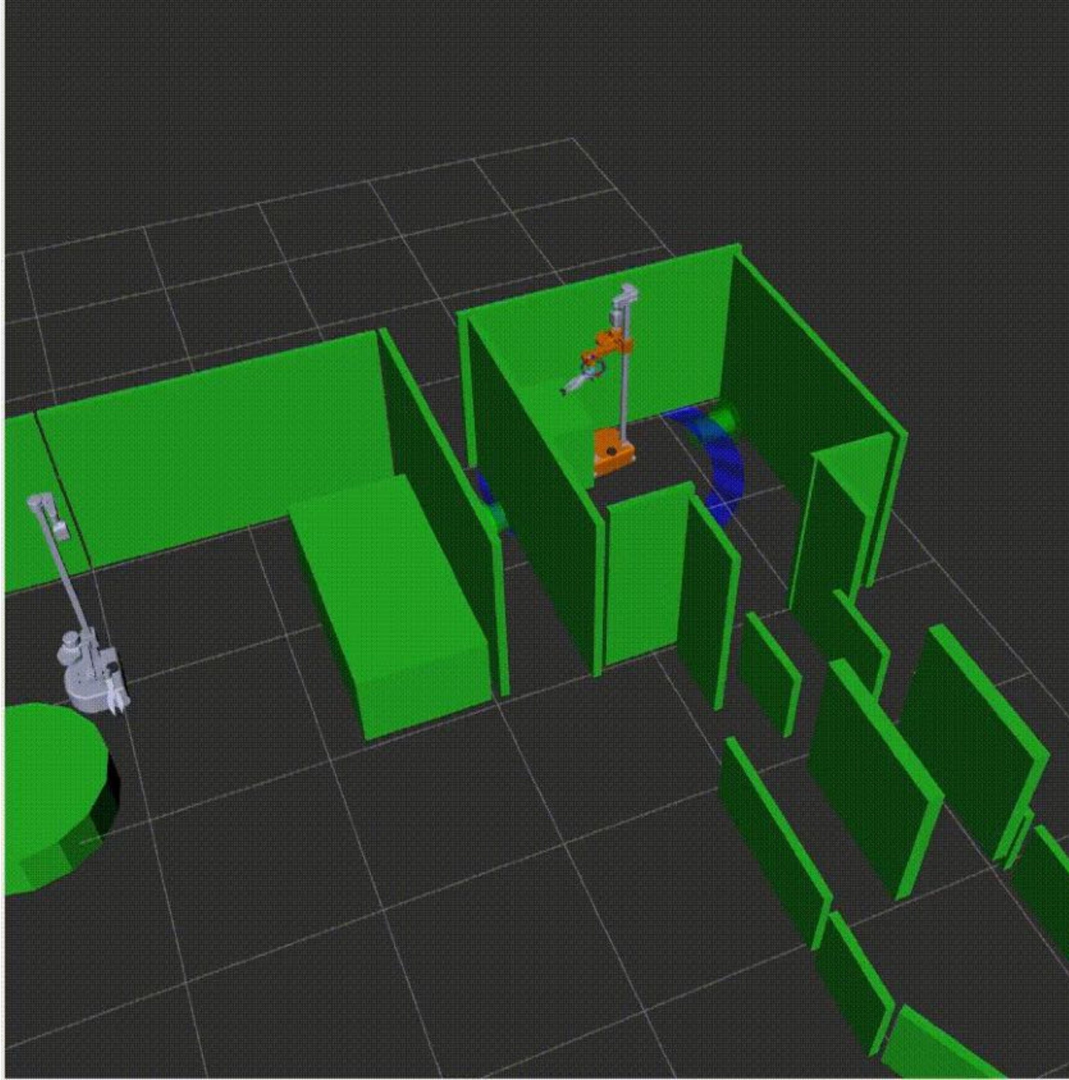




Flexibility (but also complexity)

Differential drive treated as a special joint:

- 3 DOFs: position + orientation
- joint distance/interpolation overridden to correspond to optimal diff-drive paths
- rest of MoveIt "just works"



152 Robot types integrated to work with MoveIt

29,843 Downloads per month of moveit_core

733 Academic citations of MoveIt

162,630 Unique users to moveit.ros.org in 2021

5,600 Members of Discourse, MoveIt's Discussion Forum

1,136 Github users have starred the MoveIt project

262 Github code contributors to MoveIt

167 International attendees of 2020 MoveItWorld online event

An Active Community

PICKNIK

- World MoveIt Day (annually)
- Monthly MoveIt Manipulation Working Group meetings
- ROSCON / ROS World workshops & tutorials

Land, Sea, Space

Land

Agriculture

Plant Harvesting

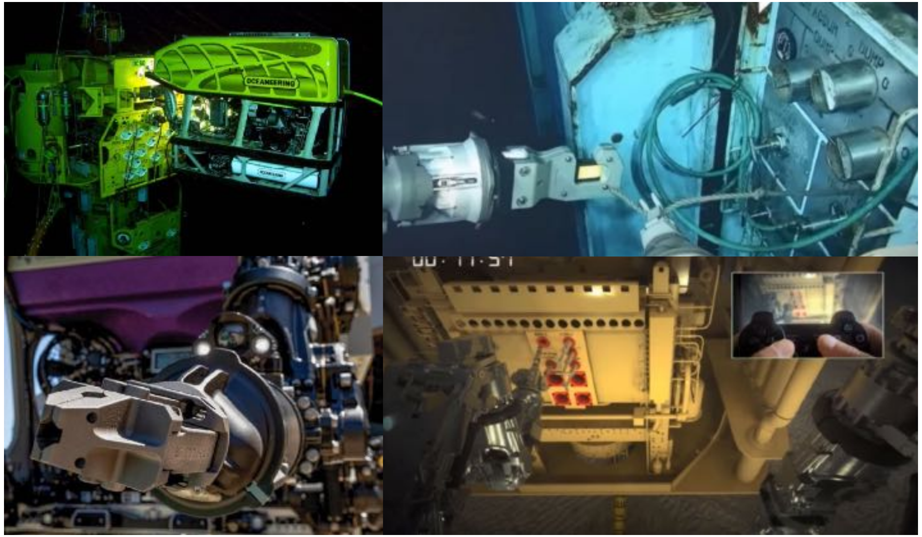
Food

Kitchen Assistant

Logistics

Bin Picking

Sea



ROVs

Remotely Operated Underwater Vehicles

Space

IVR

Inter Vehicle Robotics

OSAM

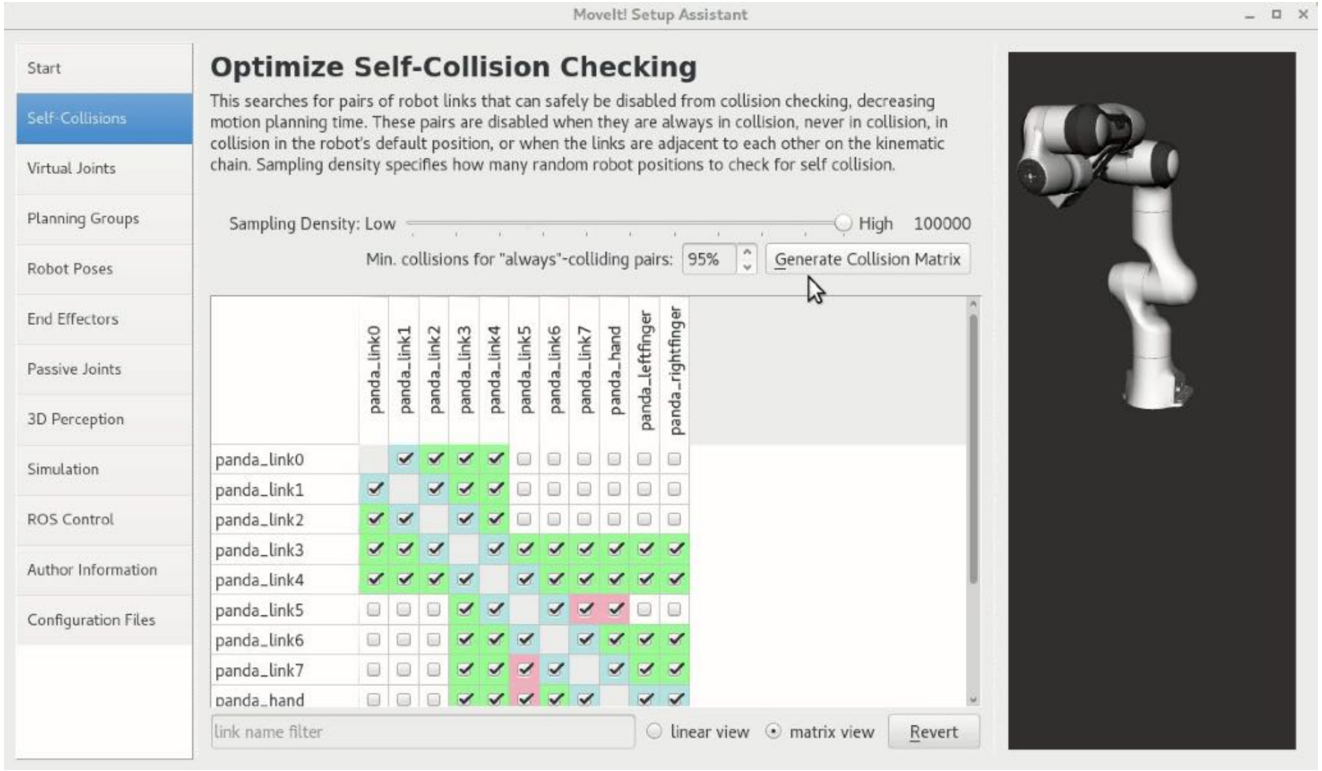
On-orbit Servicing, Assembly, and Manufacturing

Planetary Activities

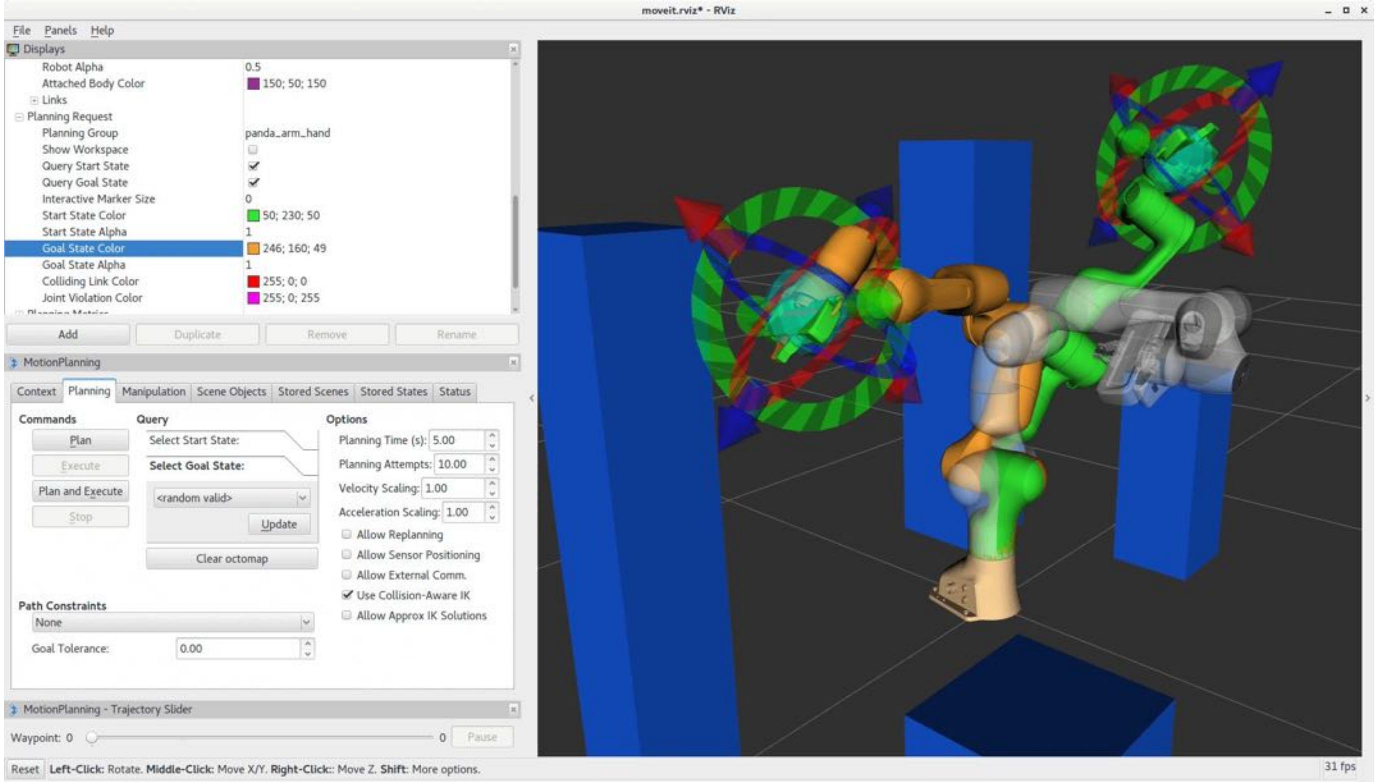
Lunar Base Construction

Typical Usage Patterns

Setup Assistant *(work in progress for MoveIt 2)*

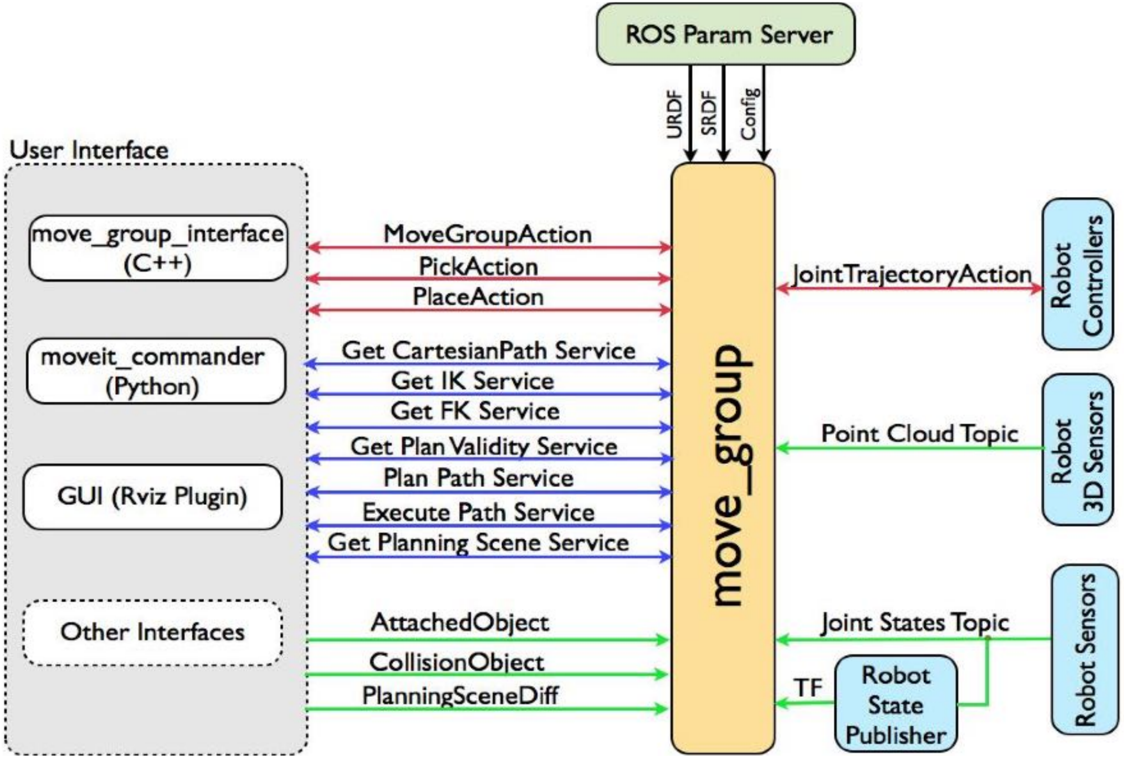


Rviz Motion Planning Plugin



PICKNIK

ROS Services & Actions



PICKNIK

Pro-Tip: Use C++ classes individually

```
robot_model_loader_.reset(new robot_model_loader::RobotModelLoader("robot_description"));
robot_model_ = robot_model_loader_->getModel();
planning_scene_.reset(new planning_scene::PlanningScene(robot_model_));
tf_.reset(new tf::TransformListener(nh_));
psm_.reset(new planning_scene_monitor::PlanningSceneMonitor(
        planning_scene_, robot_model_loader_,  tf_,"my_scene"));
psm_->startStateMonitor("/joint_states", "");
psm_->startPublishingPlanningScene(planning_scene_monitor::PlanningSceneMonitor::
                        UPDATE_SCENE, "my_planning_scene");
visuals_tools_.reset(new MoveItVisualTools(robot_model_, planning_scene_monitor_));
planning_pipeline_.reset(new planning_pipeline::PlanningPipeline(robot_model_ nh_,
                "planning_plugin", "request_adapters"));
trajectory_execution_manager_.reset(new trajectory_execution_manager::
                        TrajectoryExecutionManager(robot_model_));
```

Best for researchers who want to modify code

PICKNIK

Pro-Tip: Use MoveItCpp

Full tutorial here

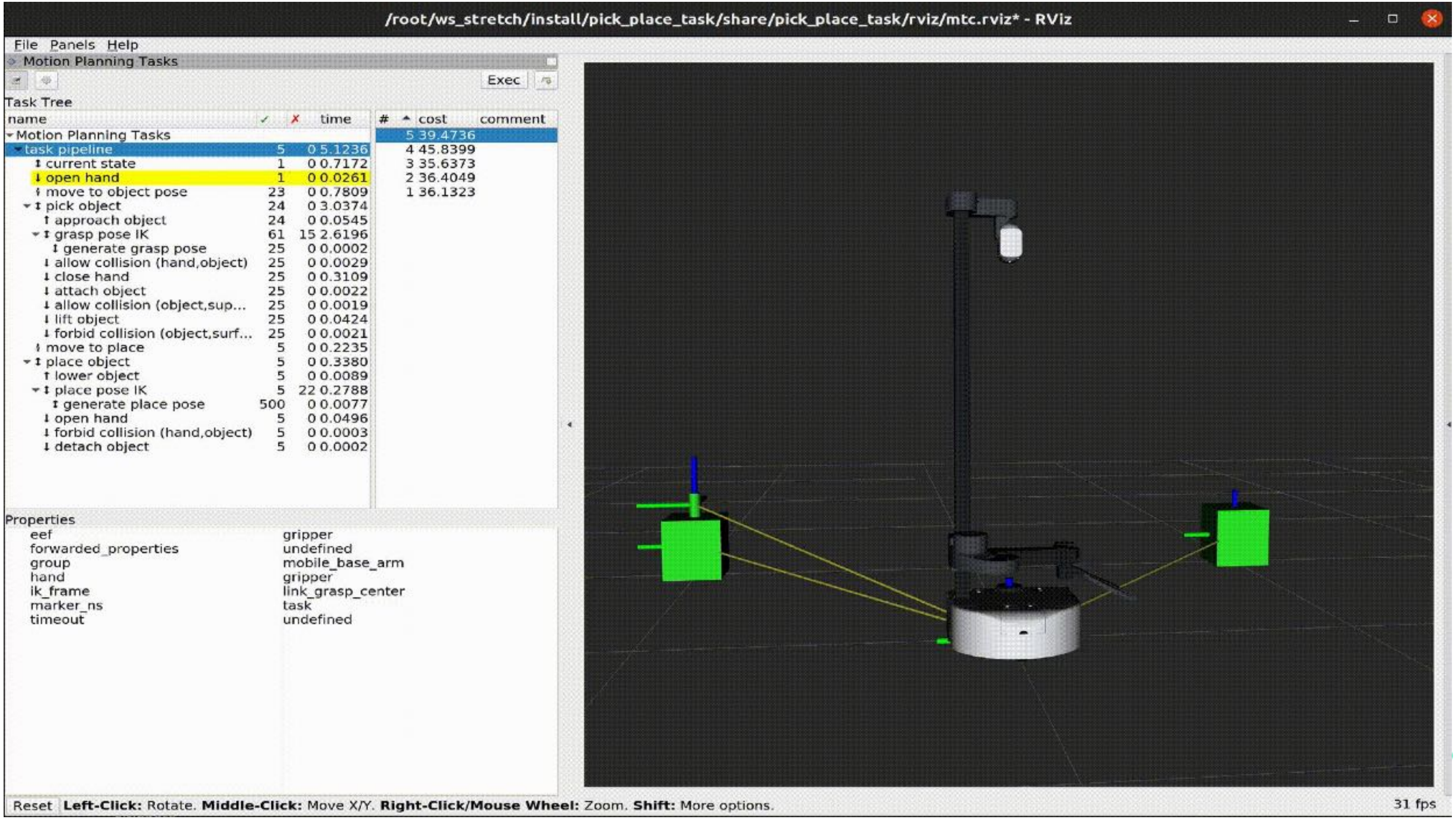
```
static const std::string PLANNING_GROUP = "stretch_arm";
auto moveit_cpp_ptr = std::make_shared<moveit_cpp::MoveItCpp>(node);
moveit_cpp_ptr->getPlanningSceneMonitor()->providePlanningSceneService();
auto planning_components = std::make_shared<moveit_cpp::PlanningComponent>
(PLANNING_GROUP, moveit_cpp_ptr);
planning_components->setStartStateToCurrentState();
planning_components->setGoal("extended");
auto plan_solution = planning_components->plan();

if (plan_solution)
{
    ...
}
```

Best for researchers who want to modify code

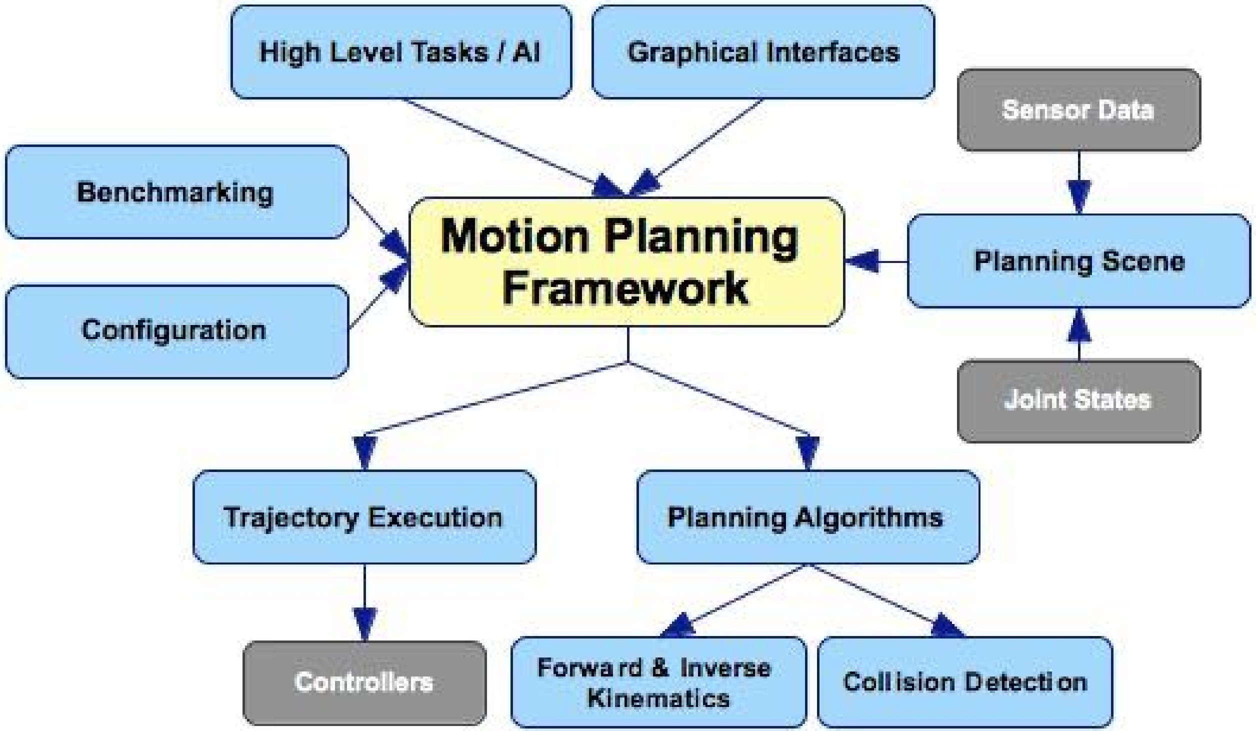
PICKNIK

MoveIt Task Constructor



Components

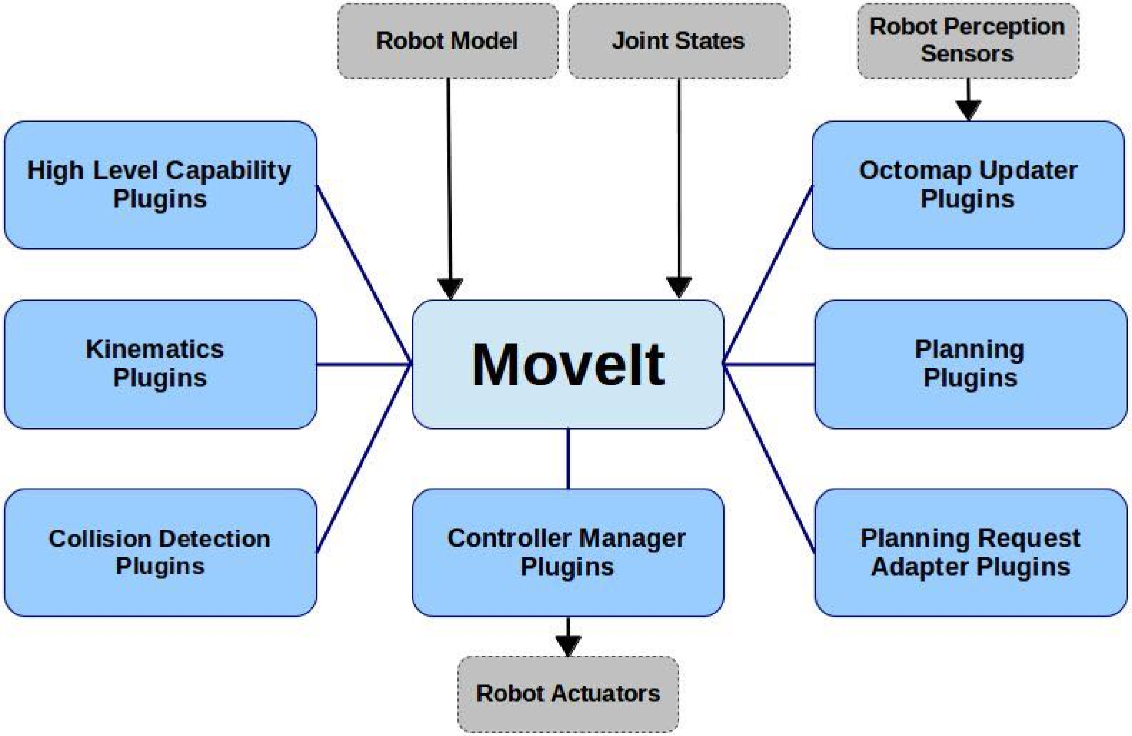
High Level Components



PICKNIK

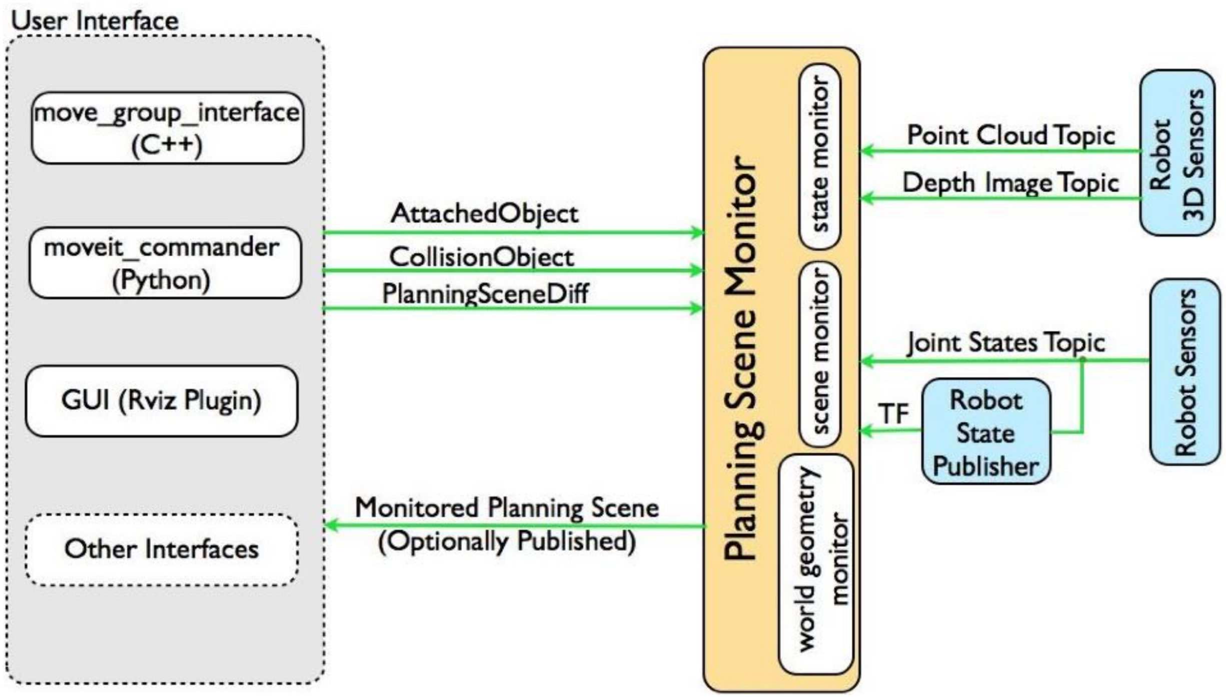
Plugin Framework

Full details [here]



PICKNIK

Planning Scene Monitor



PICKNIK

Motion Planner Plugins

OMPL	STOMP / CHOMP	TrajOpt	SBPL
Kavraki Lab at Rice	Kalakrishnan et al	John Schulman... Pieter Abbeel	Maxim Likhachev's Lab at CMU
Probabilistic, Sampling-Based	Optimization-Based	Sequential Convex Optimization	Graph-Based
Probabilistically Complete	Locally Optimal	Locally Optimal	Resolution Complete
Stochastic	Deterministic	Deterministic	Deterministic
Is a library of many planners Computationally fast More reliable runtime for real-world applications Many variants of algorithms available	Generates smooth well behaved collision free motion paths in reasonable time Can incorporate additional objective functions - collision avoidance and smoothness	Currently in final development	Requires pre-processing phase Requires custom heuristics for different configuration spaces More reliable solutions for real-world applications

PICKNIK

Planner Request Adapters

- **Plugins for pre-/post-processing planner request**

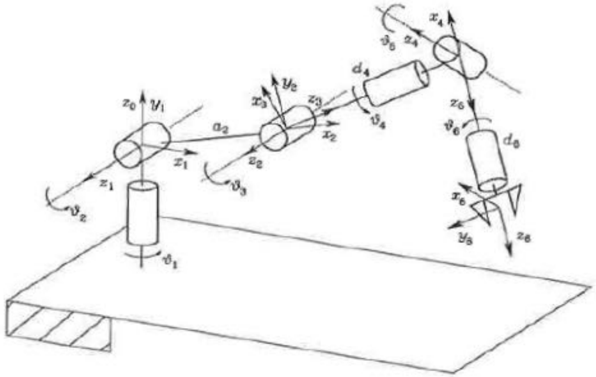
- **Two types of plugins:**
    - **Trajectory generation:** (re)parametrize trajectory to respect velocity, acceleration, jerk limits.

    - **Heuristic fixes for common problems:**
        - set workspace bounds if none defined
        - tweak joints in start state to be within joint limits
        - tweak start state to be collision free

Adapts research theory to real world hardware

PICKNIK

Inverse Kinematics



- KDL - x dof
  - Kinematics Dynamics Library
  - OROCOS
- IKFast - 6 or 7 dof
  - OpenRave Analytical
- Trac-IK
  - Combines KDL with Sequential Quadratic Programming approach
- OPW Kinematics - 6 dof
  - Closed form IK for parallel base, spherical wrist industrial manipulators

Time Parameterization

- TOTG: Time Optimal Trajectory Generation (recommended)
- Ruckig: Only method that supports jerk limits (new!)
- ITPP: Iterative Time Path Parameterization
- TOPP: Time Optimal Path Parameterization

More info here

Comparison of MoveIt 1 and MoveIt 2

PICKNIK

	MoveIt 1	MoveIt 2
ROS 1 Support	✓	via ros1_bridge
ROS 2 Support	✗	✓
Motion Planning	✓	✓
Inverse Kinematics	✓	✓
Perception	✓	✓
Grasping	✓	✓
Setup Assistant	✓	in development
MoveIt Task Constructor	✓	pending
Game Controller Integration for Servo	✓	✓

	MoveIt 1	MoveIt 2
Industrial Trajectory Generator	✓	planned
Probabilistically complete Cartesian Planning	Stale Patch	✓
Composable Nodes	nodelet subsystem	✓
Planning for Differential Drive Bases	✗	✓
Hybrid Planning (global + local planners)	✗	pending
Based on Realtime Capable DDS Messaging	✗	✓
Native Windows Build	via RoboStack	✓
New Feature Development by PickNik	✗	✓
Development Coordinated with ROS 2 Technical Steering Committee	✗	✓
Built for Industrial Security	✗	✓

More Resources

Online Tutorials (http://moveit2_tutorials.picknik.ai)

MoveIt

Version: MoveIt 2 - Foxy

Search tutorials

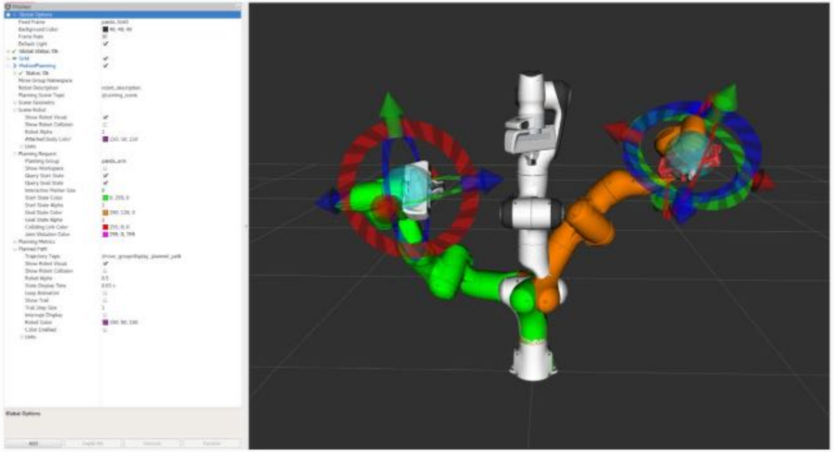
- Getting Started
- MoveIt Quickstart in RViz
- Move Group C++ Interface
- Robot Model and Robot State
- Planning Scene
- Planning Scene Monitor
- Planning Scene ROS API
- MoveItCpp Tutorial
- Planning For Differential-Drive Mobile Base and an Arm
- URDF and SRDF
- Realtime Arm Servoing

MoveIt » Tutorials » MoveIt 2 Tutorials

Edit on GitHub

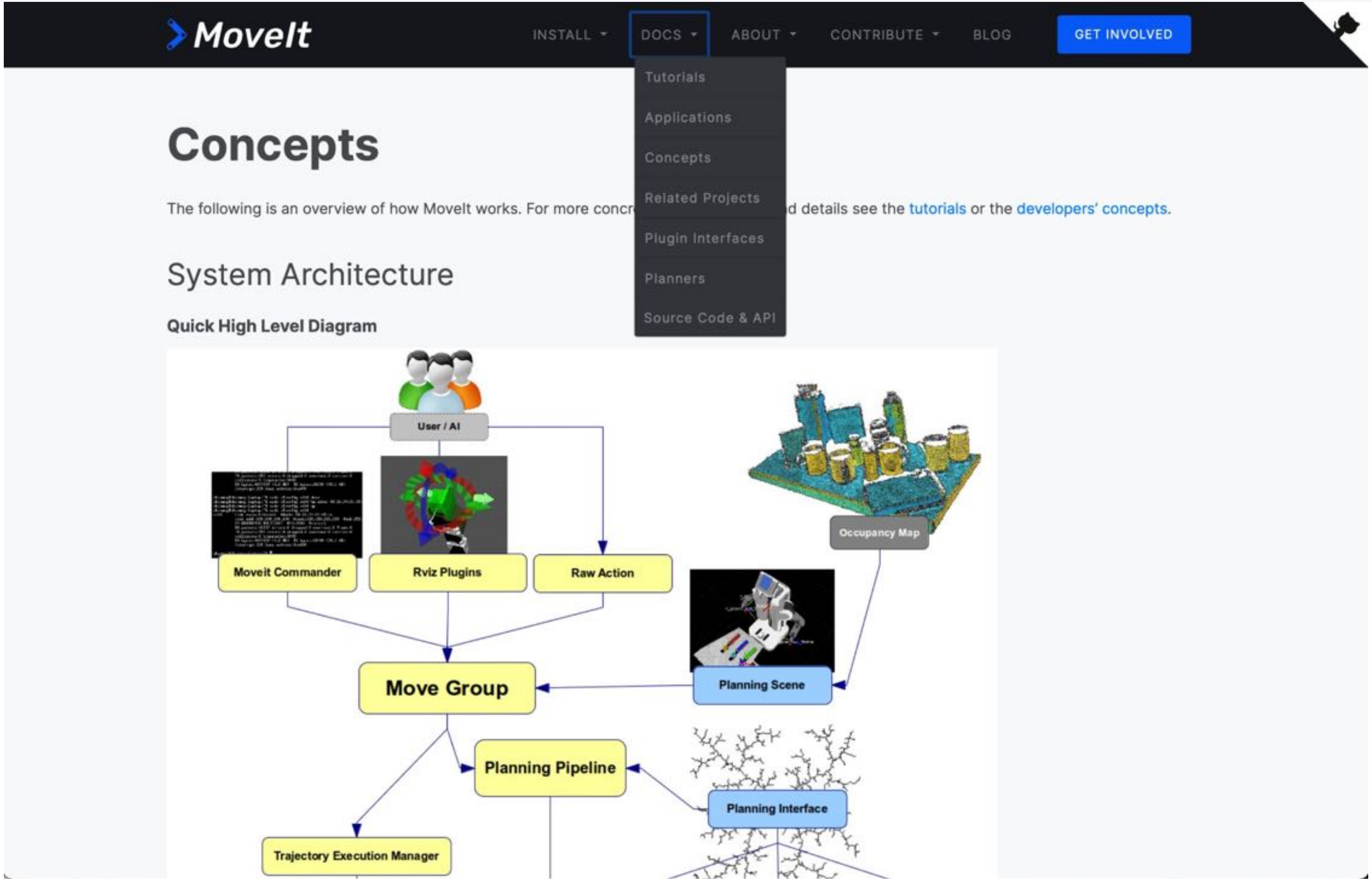
MoveIt 2 Tutorials

These tutorials will quickly get you, and your robot, using the MoveIt 2 Motion Planning Framework.



In these tutorials, the Franka Emika Panda robot is used as a quick-start demo. Alternatively, you can easily use any robot that has already been configured to work with MoveIt - check the list of robots running MoveIt to see whether MoveIt is already available for your robot. Otherwise, you can setup MoveIt to work with your custom robot in the tutorial section "Integration with a New Robot", below.

Main website (http://moveit.ros.org)



Thanks!

PickNik Robotics
picknik.ai
Colorado, USA
@picknikrobotics

Mark Moll
mark@picknik.ai