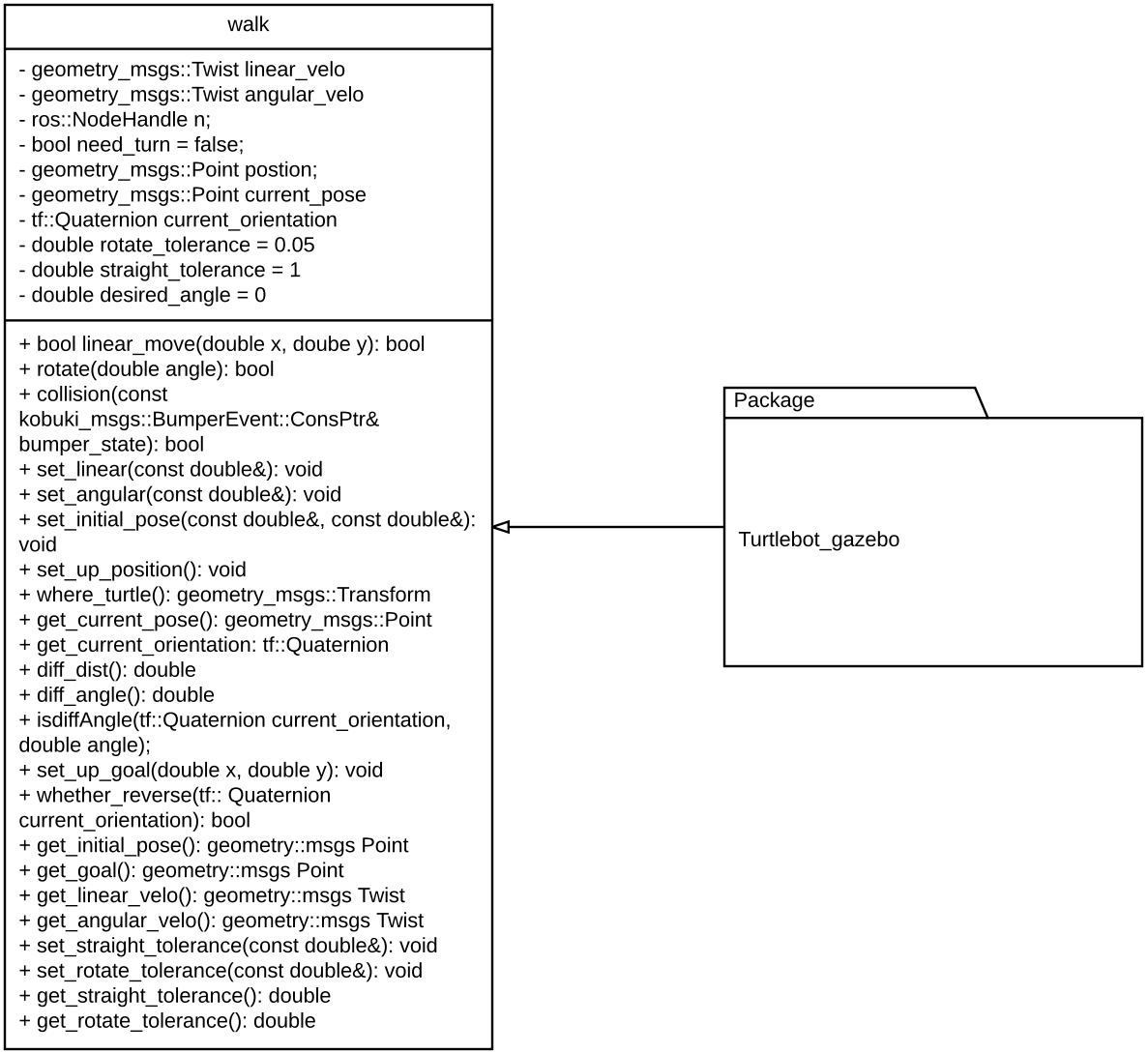
walk
- geometry_msgs::Twist linear_velo
- geometry_msgs::Twist angular_velo
- ros::NodeHandle n;
- bool need_turn = false;
- geometry_msgs::Point postion;
- geometry_msgs::Point current_pose
- tf::Quaternion current_orientation
- double rotate_tolerance = 0.05
- double straight_tolerance = 1
- double desired_angle = 0
+ bool linear_move(double x, doube y): bool
+ rotate(double angle): bool
+ collision(const kobuki_msgs::BumperEvent::ConsPtr& bumper_state): bool
+ set_linear(const double&): void
+ set_angular(const double&): void
+ set_initial_pose(const double&, const double&): void
+ set_up_position(): void
+ where_turtle(): geometry_msgs::Transform
+ get_current_pose(): geometry_msgs::Point
+ get_current_orientation: tf::Quaternion
+ diff_dist(): double
+ diff_angle(): double
+ isdiffAngle(tf::Quaternion current_orientation, double angle);
+ set_up_goal(double x, double y): void
+ whether_reverse(tf:: Quaternion current_orientation): bool
+ get_initial_pose(): geometry::msgs Point
+ get_goal(): geometry::msgs Point
+ get_linear_velo(): geometry::msgs Twist
+ get_angular_velo(): geometry::msgs Twist
+ set_straight_tolerance(const double&): void
+ set_rotate_tolerance(const double&): void
+ get_straight_tolerance(): double
+ get_rotate_tolerance(): double
Package
Turtlebot_gazebo