rmf_simulation

Release 1.0.0

Open Source Robotics Corporation

Mar 08, 2022

CONTENTS:

Simulation plugins for use with OpenRMF.

CHAPTER

ONE

RMF_SIMULATION API

1.1 Class Hierarchy

1.2 File Hierarchy

1.3 Full API

1.3.1 Namespaces

Namespace crowd_simulator

Contents

Classes

Typedefs

- *Typedef crowd_simulator::AgentPtr*

Namespace rmf_building_sim_common

Contents

Classes

- *Struct DoorCommon::DoorElement*
- *Struct DoorCommon::DoorUpdateRequest*
- *Struct DoorCommon::DoorUpdateResult*
- *Struct LiftCommon::LiftUpdateResult*
- *Struct MotionParams*
- *Class DoorCommon*
- *Class LiftCommon*

Functions

- *Function rmf_building_sim_common::compute_desired_rate_of_change*
- *Function rmf_building_sim_common::compute_ds*
- *Template Function rmf_building_sim_common::get_element_required*
- *Template Function rmf_building_sim_common::get_sdf_attribute_required*
- *Template Function rmf_building_sim_common::get_sdf_param_if_available*
- *Template Function rmf_building_sim_common::get_sdf_param_required*

Typedefs

- *Typedef rmf_building_sim_common::DoorMode*
- *Typedef rmf_building_sim_common::DoorRequest*
- *Typedef rmf_building_sim_common::DoorState*
- *Typedef rmf_building_sim_common::LiftRequest*
- *Typedef rmf_building_sim_common::LiftState*

Namespace rmf_dispenser_common

Contents

Classes

Namespace rmf_ingestor_common

Contents

Classes

Namespace rmf_plugins_utils

Contents

Classes

Enums

- *Enum Simulator*

Functions

- *Function rmf_plugins_utils::compute_desired_rate_of_change*
- *Function rmf_plugins_utils::compute_ds*
- *Template Function rmf_plugins_utils::convert(const Eigen::Quaterniond&, IgnQuatT&)*
- *Template Function rmf_plugins_utils::convert(const Eigen::Vector3d&, IgnVec3T&)*
- *Template Function rmf_plugins_utils::convert_pose*
- *Template Function rmf_plugins_utils::convert_quat*
- *Template Function rmf_plugins_utils::convert_to_pose*
- *Template Function rmf_plugins_utils::convert_vec*
- *Template Function rmf_plugins_utils::get_element_required*
- *Template Function rmf_plugins_utils::get_sdf_attribute_required*
- *Template Function rmf_plugins_utils::get_sdf_param_if_available*
- *Template Function rmf_plugins_utils::get_sdf_param_required*
- *Template Function rmf_plugins_utils::make_response*
- *Function rmf_plugins_utils::simulation_now*

Namespace rmf_readonly_common

Contents

- *Classes*

Classes

- *Class ReadonlyCommon*

Namespace rmf_robot_sim_common

Contents

- *Classes*
- *Enums*
- *Functions*

Classes

- *Struct AckermannTrajectory*
- *Struct SlotcarCommon::ChargerWaypoint*
- *Struct SlotcarCommon::PowerParams*
- *Struct SlotcarCommon::UpdateResult*
- *Struct TrajectoryPoint*
- *Class SlotcarCommon*

Enums

- *Enum SteeringType*

Functions

- *Template Function rmf_robot_sim_common::get_element_val_if_present*

1.3.2 Classes and Structs

Struct CrowdSimInterface::Object

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_crowd_simulator_common.hpp

Nested Relationships

This struct is a nested type of *Class CrowdSimInterface*.

Struct Documentation

struct crowd_simulator::*CrowdSimInterface*::**Object**

Public Functions

AnimState **get_next_state** (bool *condition*)

Public Members

AgentPtr **agent_ptr**

std::string **model_name**

std::string **type_name**

bool **is_external** = false

AnimState **current_state**

Struct ModelTypeDatabase::Record

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_crowd_simulator_common.hpp

Nested Relationships

This struct is a nested type of *Class ModelTypeDatabase*.

Struct Documentation

struct crowd_simulator::*ModelTypeDatabase*::**Record**

Public Members

std::string **type_name**

std::string **file_name**

AgentPose3d **pose**

std::string **animation**

std::string **idle_animation**

double **animation_speed**

Struct DoorCommon::DoorElement

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_door_common.hpp

Nested Relationships

This struct is a nested type of *Class DoorCommon*.

Struct Documentation

struct rmf_building_sim_common::*DoorCommon*::**DoorElement**

Public Functions

inline DoorElement()

inline DoorElement(**const** double *lower_limit*, **const** double *upper_limit*, **const** bool *flip_direction* = false)

Public Members

double **closed_position**

double **open_position**

double **current_position**

double **current_velocity**

Struct DoorCommon::DoorUpdateRequest

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_door_common.hpp

Nested Relationships

This struct is a nested type of *Class DoorCommon*.

Struct Documentation

struct rmf_building_sim_common::*DoorCommon*::**DoorUpdateRequest**

Public Members

std::string **joint_name**

double **position**

double **velocity**

Struct DoorCommon::DoorUpdateResult

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_door_common.hpp

Nested Relationships

This struct is a nested type of *Class DoorCommon*.

Struct Documentation

struct rmf_building_sim_common::*DoorCommon*::**DoorUpdateResult**

Public Members

std::string **joint_name**

double **velocity**

double **fmax**

Struct LiftCommon::LiftUpdateResult

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_lift_common.hpp

Nested Relationships

This struct is a nested type of *Class LiftCommon*.

Struct Documentation

struct rmf_building_sim_common::*LiftCommon*::**LiftUpdateResult**

Public Members

double **velocity**

double **fmax**

Struct MotionParams

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_utils.hpp

Struct Documentation

struct rmf_building_sim_common::**MotionParams**

Public Members

double **v_max** = 0.2

double **a_max** = 0.1

double **a_nom** = 0.08

double **dx_min** = 0.01

double **f_max** = 10000000.0

Struct MotionParams

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Struct Documentation

struct `rmf_plugins_utils::`**MotionParams**

Public Members

double **v_max** = 0.2

double **a_max** = 0.1

double **a_nom** = 0.08

double **dx_min** = 0.01

double **f_max** = 10000000.0

Struct SimEntity

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Struct Documentation

struct `rmf_plugins_utils::`**SimEntity**

Public Functions

inline SimEntity(uint64_t *en*)

inline SimEntity(std::string *nm*)

inline const std::string &**get_name**() **const**

inline uint64_t **get_entity**() **const**

Public Members

Simulator **sim_type**

uint64_t **entity**

std::string **name**

Struct AckermannTrajectory

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_slotcar_common.hpp

Struct Documentation

struct rmf_robot_sim_common::**AckermannTrajectory**

Public Functions

inline AckermannTrajectory(**const** Eigen::Vector2d *&_x0*, **const** Eigen::Vector2d *&_x1*, **const** Eigen::Vector2d *&_v1* = Eigen::Vector2d(0, 0), bool *_turning* = false)

Public Members

Eigen::Vector2d **x0**

Eigen::Vector2d **x1**

Eigen::Vector2d **v0**

Eigen::Vector2d **v1**

bool **turning** = false

Struct SlotcarCommon::ChargerWaypoint

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_slotcar_common.hpp

Nested Relationships

This struct is a nested type of *Class SlotcarCommon*.

Struct Documentation

struct rmf_robot_sim_common::*SlotcarCommon*::**ChargerWaypoint**

Public Functions

inline ChargerWaypoint(double *x*, double *y*)

Public Members

double **x**

double **y**

Struct SlotcarCommon::PowerParams

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_slotcar_common.hpp

Nested Relationships

This struct is a nested type of *Class SlotcarCommon*.

Struct Documentation

struct rmf_robot_sim_common::*SlotcarCommon*::**PowerParams**

Public Members

double **nominal_voltage** = 12

double **nominal_capacity** = 24

double **charging_current** = 2

double **mass** = 20

double **inertia** = 10

double **friction_coefficient** = 0.3

double **nominal_power** = 10

Struct SlotcarCommon::UpdateResult

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_slotcar_common.hpp

Nested Relationships

This struct is a nested type of *Class SlotcarCommon*.

Struct Documentation

struct rmf_robot_sim_common::*SlotcarCommon*::**UpdateResult**

Public Members

double **v** = 0.0

double **w** = 0.0

double **speed** = 0.0

Struct TrajectoryPoint

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_slotcar_common.hpp

Struct Documentation

struct rmf_robot_sim_common::**TrajectoryPoint**

Public Functions

inline TrajectoryPoint(**const** Eigen::Vector3d &*_pos*, **const** Eigen::Quaterniond &*_quat*)

Public Members

Eigen::Vector3d **pos**

Eigen::Quaterniond **quat**

Class AgentPose3d

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_crowd_simulator_common.hpp

Class Documentation

class crowd_simulator::**AgentPose3d**

Public Functions

inline AgentPose3d()

inline AgentPose3d(double *x*, double *y*, double *z*, double *roll*, double *pitch*, double *yaw*)

inline double **x**() **const**

inline double **y**() **const**

inline double **z**() **const**

inline double **roll**() **const**

inline double **pitch**() **const**

inline double **yaw**() **const**

inline void **x**(double *x*)

inline void **y** (double *y*)

inline void **z** (double *z*)

inline void **roll** (double *roll*)

inline void **pitch** (double *pitch*)

inline void **yaw** (double *yaw*)

template<typename **IgnMathPose3d**>
inline *IgnMathPose3d* **convert_to_ign_math_pose_3d** ()

Class CrowdSimInterface

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_crowd_simulator_common.hpp

Nested Relationships

Nested Types

- *Struct CrowdSimInterface::Object*

Class Documentation

class crowd_simulator::**CrowdSimInterface**

Public Types

enum AnimState
Values:

enumerator WALK

enumerator IDLE

using ObjectPtr = std::shared_ptr<*Object*>

Public Functions

inline CrowdSimInterface ()

rclcpp::Logger **logger** () **const**

void **init_ros_node** (**const** rclcpp::Node::SharedPtr *node*)

bool **init_crowd_sim** ()

double **get_sim_time_step** () **const**

size_t **get_num_objects** () **const**

ObjectPtr **get_object_by_id** (size_t *id*) **const**

void **one_step_sim** (double *time_step*) **const**

double **get_switch_anim_distance_th** () **const**

std::vector<std::string> **get_switch_anim_name**() **const**

bool **enabled**() **const**

template<typename **SdfPtrT**>
bool **read_sdf**(*SdfPtrT* &*sdf*)

template<typename **IgnMathPose3d**>
void **update_external_agent**(size_t *id*, **const** *IgnMathPose3d* &*model_pose*)

template<typename **IgnMathPose3d**>
void **update_external_agent**(**const** *AgentPtr* *agent_ptr*, **const** *IgnMathPose3d* &*model_pose*)

template<typename **IgnMathPose3d**>
IgnMathPose3d **get_agent_pose**(size_t *id*, double *delta_sim_time*)

template<typename **IgnMathPose3d**>
IgnMathPose3d **get_agent_pose**(**const** *AgentPtr* *agent_ptr*, double *delta_sim_time*)

Public Members

std::shared_ptr<*ModelTypeDatabase*> **_model_type_db_ptr**

struct Object

Public Functions

AnimState **get_next_state**(bool *condition*)

Public Members

AgentPtr **agent_ptr**

std::string **model_name**

std::string **type_name**

bool **is_external** = false

AnimState **current_state**

Class MengeHandle

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_crowd_simulator_common.hpp

Inheritance Relationships

Base Type

- `public std::enable_shared_from_this< MengeHandle >`

Class Documentation

class crowd_simulator::**MengeHandle** : **public** std::enable_shared_from_this<*MengeHandle*>

Public Functions

inline MengeHandle(**const** std::string &*resource_path*, **const** std::string &*behavior_file*, **const** std::string &*scene_file*, **const** float *sim_time_step* = 0.0)

void **set_sim_time_step**(float *sim_time_step*)

float **get_sim_time_step**() **const**

size_t **get_agent_count**()

void **sim_step**() **const**

AgentPtr **get_agent**(size_t *id*) **const**

Public Static Functions

static std::shared_ptr<*MengeHandle*> **init_and_make**(**const** std::string &*resource_path*, **const** std::string &*behavior_file*, **const** std::string &*scene_file*, **const** float *sim_time_step*)

Class ModelTypeDatabase

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_crowd_simulator_common.hpp

Nested Relationships

Nested Types

- *Struct ModelTypeDatabase::Record*

Class Documentation

class crowd_simulator::**ModelTypeDatabase**

Public Types

using RecordPtr = std::shared_ptr<*Record*>

Public Functions

RecordPtr **emplace**(std::string *type_name*, *RecordPtr record_ptr*)

size_t **size**() **const**

RecordPtr **get**(**const** std::string &*type_name*) **const**

struct Record

Public Members

std::string **type_name**

std::string **file_name**

AgentPose3d **pose**

std::string **animation**

std::string **idle_animation**

double **animation_speed**

Class DoorCommon

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_door_common.hpp

Nested Relationships

Nested Types

- *Struct DoorCommon::DoorElement*
- *Struct DoorCommon::DoorUpdateRequest*
- *Struct DoorCommon::DoorUpdateResult*

Class Documentation

class rmf_building_sim_common**::DoorCommon**

Public Functions

rclcpp::Logger **logger**() **const**

std::vector<std::string> **joint_names**() **const**

MotionParams &**params**()

std::vector<*DoorUpdateResult*> **update**(**const** double *time*, **const** std::vector<*DoorUpdateRequest*> &*request*)

Public Static Functions

template<typename **SdfPtrT**>
static std::shared_ptr<*DoorCommon*> **make**(**const** std::string *&door_name*, rclcpp::Node::SharedPtr *node*, *SdfPtrT &sdf*)

struct DoorUpdateRequest

Public Members

std::string **joint_name**

double **position**

double **velocity**

struct DoorUpdateResult

Public Members

std::string **joint_name**

double **velocity**

double **fmax**

Class LiftCommon

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_lift_common.hpp

Nested Relationships

Nested Types

- *Struct LiftCommon::LiftUpdateResult*

Class Documentation

class rmf_building_sim_common**::LiftCommon**

Public Functions

rclcpp::Logger **logger**() **const**

LiftUpdateResult **update**(**const** double *time*, **const** double *position*, **const** double *velocity*)

std::string **get_joint_name**() **const**

double **get_elevation**() **const**

bool **motion_state_changed**()

Public Static Functions

template<typename **SdfPtrT**>
static std::unique_ptr<*LiftCommon*> **make**(**const** std::string *&lift_name*, rclcpp::Node::SharedPtr *node*, *SdfPtrT &sdf*)

struct LiftUpdateResult

Public Members

double **velocity**

double **fmax**

Class TeleportDispenserCommon

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_dispenser_common.hpp

Class Documentation

class rmf_dispenser_common::**TeleportDispenserCommon**

Public Types

using FleetState = rmf_fleet_msgs::msg::FleetState

using FleetStateIt = std::unordered_map<std::string, *FleetState*::UniquePtr>::iterator

using DispenserState = rmf_dispenser_msgs::msg::DispenserState

using DispenserRequest = rmf_dispenser_msgs::msg::DispenserRequest

using DispenserResult = rmf_dispenser_msgs::msg::DispenserResult

Public Functions

void **send_dispenser_response**(uint8_t *status*) **const**

void **fleet_state_cb**(*FleetState*::UniquePtr *msg*)

void **dispenser_request_cb**(*DispenserRequest*::UniquePtr *msg*)

void **on_update**(std::function<void) *FleetStateIt*, std::vector<rmf_plugins_utils::*SimEntity*>& > *fill_robot_list_cb*, std::function<rmf_plugins_utils::*SimEntity***const** std::vector<rmf_plugins_utils::*SimEntity*>&, bool&> *find_nearest_model_cb*, std::function<void**const** rmf_plugins_utils::*SimEntity*&> *place_on_entity_cb*, std::function<boolvoid> *check_filled_cb*

void **init_ros_node**(**const** rclcpp::Node::SharedPtr *node*)

Public Members

bool **dispense** = false

DispenserRequest **latest**

std::string **guid**

double **last_pub_time** = 0.0

double **sim_time** = 0.0

bool **item_en_found** = false

bool **dispenser_filled** = false

rclcpp::Node::SharedPtr **ros_node**

std::unordered_map<std::string, *FleetState*::UniquePtr> **fleet_states**

DispenserState **current_state**

Class TeleportIngestorCommon

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_ingestor_common.hpp

Class Documentation

class rmf_ingestor_common::**TeleportIngestorCommon**

Public Types

using FleetState = rmf_fleet_msgs::msg::FleetState

using FleetStateIt = std::unordered_map<std::string, *FleetState*::UniquePtr>::iterator

using IngestorState = rmf_ingestor_msgs::msg::IngestorState

using IngestorRequest = rmf_ingestor_msgs::msg::IngestorRequest

using IngestorResult = rmf_ingestor_msgs::msg::IngestorResult

Public Functions

void **send_ingestor_response**(uint8_t *status*) **const**

void **fleet_state_cb**(*FleetState*::UniquePtr *msg*)

void **ingestor_request_cb**(*IngestorRequest*::UniquePtr *msg*)

void **on_update**(std::function<void) *FleetStateIt*, std::vector<rmf_plugins_utils::*SimEntity*>& > *fill_robot_list_cb*, std::function<rmf_plugins_utils::*SimEntity***const** std::vector<rmf_plugins_utils::*SimEntity*>&, bool&> *find_nearest_model_cb*, std::function<bool**const** SimEntity&> *get_payload_model_cb*, std::function<void> *transport_model_cb*std::function<voidvoid> *send_ingested_item_home_cb*

void **init_ros_node**(**const** rclcpp::Node::SharedPtr *node*)

Public Members

bool **ingest** = false

IngestorRequest **latest**

std::string **_guid**

bool **ingestor_filled** = false

double **last_pub_time** = 0.0

double **last_ingested_time** = 0.0

double **sim_time** = 0.0

rclcpp::Node::SharedPtr **ros_node**

std::unordered_map<std::string, Eigen::Isometry3d> **non_static_models_init_poses**

std::unordered_map<std::string, *FleetState*::UniquePtr> **fleet_states**

IngestorState **current_state**

Class ReadonlyCommon

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_readonly_common.hpp

Class Documentation

class rmf_readonly_common**::ReadonlyCommon**

Public Types

using BuildingMap = rmf_building_map_msgs::msg::BuildingMap

using Level = rmf_building_map_msgs::msg::Level

using Graph = rmf_building_map_msgs::msg::Graph

using Location = rmf_fleet_msgs::msg::Location

using Path = std::vector<*Location*>

Public Functions

void **set_name**(**const** std::string &*name*)

std::string **get_name**() **const**

rclcpp::Logger **logger**()

template<typename **SdfPtrT**>
void **read_sdf**(*SdfPtrT* &*sdf*)

void **init**(rclcpp::Node::SharedPtr *node*)

void **on_update**(Eigen::Isometry3d &*pose*, double *sim_time*)

Public Members

rclcpp::Node::SharedPtr **ros_node**

Class SlotcarCommon

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_slotcar_common.hpp

Nested Relationships

Nested Types

- *Struct SlotcarCommon::ChargerWaypoint*
- *Struct SlotcarCommon::PowerParams*
- *Struct SlotcarCommon::UpdateResult*

Class Documentation

class rmf_robot_sim_common::**SlotcarCommon**

Public Functions

SlotcarCommon()

rclcpp::Logger **logger**() **const**

template<typename **SdfPtrT**>
void **read_sdf**(*SdfPtrT* &*sdf*)

void **set_model_name**(**const** std::string &*model_name*)

std::string **model_name**() **const**

void **init_ros_node**(**const** rclcpp::Node::SharedPtr *node*)

UpdateResult **update**(**const** Eigen::Isometry3d &*pose*, **const** std::vector<Eigen::Vector3d> &*obstacle_positions*, **const** double *time*)

bool **emergency_stop**(**const** std::vector<Eigen::Vector3d> &*obstacle_positions*, **const** Eigen::Vector3d &*current_heading*)

std::array<double, 2> **calculate_control_signals**(**const** std::array<double, 2> &*curr_velocities*, **const** std::pair<double, double> &*displacements*, **const** double *dt*, **const** double *target_linear_velocity* = 0.0) **const**

std::array<double, 2> **calculate_joint_control_signals**(**const** std::array<double, 2> &*w_tire*, **const** std::pair<double, double> &*displacements*, **const** double *dt*) **const**

void **charge_state_cb**(**const** std::string &*name*, bool *selected*)

void **publish_robot_state**(**const** double *time*)

struct UpdateResult

Public Members

double **v** = 0.0

double **w** = 0.0

double **speed** = 0.0

1.3.3 Enums

Enum Simulator

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Enum Documentation

enum rmf_plugins_utils::**Simulator**

Values:

enumerator Ignition

enumerator Gazebo

Enum SteeringType

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_slotcar_common.hpp

Enum Documentation

enum rmf_robot_sim_common::**SteeringType**

Values:

enumerator DIFF_DRIVE

enumerator ACKERMANN

1.3.4 Functions

Function rmf_building_sim_common::compute_desired_rate_of_change

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_utils.hpp

Function Documentation

double `rmf_building_sim_common::compute_desired_rate_of_change`(double *_s_target*, double *_v_actual*, **const** *MotionParams* &*_motion_params*, **const** double *_dt*)

Function rmf_building_sim_common::compute_ds

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_utils.hpp

Function Documentation

double `rmf_building_sim_common::compute_ds`(double *s_target*, double *v_actual*, **const** double *v_max*, **const** double *accel_nom*, **const** double *accel_max*, **const** double *dt*)

Template Function rmf_building_sim_common::get_element_required

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_utils.hpp

Function Documentation

template<typename **SdfPtrT**, typename **SdfElementPtrT**>
bool `rmf_building_sim_common::get_element_required`(*SdfPtrT* &*_sdf*, **const** std::string &*_element_name*, *SdfElementPtrT* &*_element*)

Template Function rmf_building_sim_common::get_sdf_attribute_required

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_utils.hpp

Function Documentation

template<typename **T**, typename **SdfPtrT**>
bool `rmf_building_sim_common::get_sdf_attribute_required`(*SdfPtrT* &*sdf*, **const** std::string &*attribute_name*, *T* &*value*)

Template Function rmf_building_sim_common::get_sdf_param_if_available

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_utils.hpp

Function Documentation

template<typename **T**, typename **SdfPtrT**>
void `rmf_building_sim_common::`**`get_sdf_param_if_available`**(*SdfPtrT &sdf*, **const** std::string *&parameter_name*, *T &value*)

Template Function rmf_building_sim_common::get_sdf_param_required

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_utils.hpp

Function Documentation

template<typename **T**, typename **SdfPtrT**>
bool `rmf_building_sim_common::`**`get_sdf_param_required`**(*SdfPtrT &sdf*, **const** std::string *&parameter_name*, *T &value*)

Function rmf_plugins_utils::compute_desired_rate_of_change

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

double `rmf_plugins_utils::`**`compute_desired_rate_of_change`**(double *_s_target*, double *_v_actual*, **const** *MotionParams &_motion_params*, **const** double *_dt*)

Function rmf_plugins_utils::compute_ds

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

double `rmf_plugins_utils::`**`compute_ds`**(double *s_target*, double *v_actual*, **const** double *v_max*, **const** double *accel_nom*, **const** double *accel_max*, **const** double *dt*, **const** double *v_target* = 0.0)

Template Function rmf_plugins_utils::convert(const Eigen::Quaterniond&, IgnQuatT&)

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

template<typename **IgnQuatT**>
inline void `rmf_plugins_utils::convert`(**const** Eigen::Quaterniond &*_q*, *IgnQuatT* &*quat*)

Template Function rmf_plugins_utils::convert(const Eigen::Vector3d&, IgnVec3T&)

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

template<typename **IgnVec3T**>
inline void `rmf_plugins_utils::convert`(**const** Eigen::Vector3d &*_v*, *IgnVec3T* &*vec*)

Template Function rmf_plugins_utils::convert_pose

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

template<typename **IgnPoseT**>
inline Eigen::Isometry3d `rmf_plugins_utils::convert_pose`(**const** *IgnPoseT* &*_pose*)

Template Function rmf_plugins_utils::convert_quat

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

template<typename **IgnQuatT**>
inline Eigen::Quaterniond `rmf_plugins_utils::convert_quat`(**const** *IgnQuatT* &*_q*)

Template Function rmf_plugins_utils::convert_to_pose

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

template<typename **IgnPoseT**>
inline auto `rmf_plugins_utils::convert_to_pose`(**const** Eigen::Isometry3d &*_tf*)

Template Function rmf_plugins_utils::convert_vec

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

template<typename **IgnVec3T**>
inline Eigen::Vector3d `rmf_plugins_utils::convert_vec`(**const** *IgnVec3T* &*_v*)

Template Function rmf_plugins_utils::get_element_required

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

template<typename **SdfPtrT**, typename **SdfElementPtrT**>
bool `rmf_plugins_utils::get_element_required`(*SdfPtrT* &*_sdf*, **const** std::string &*_element_name*, *SdfElementPtrT* &*_element*)

Template Function rmf_plugins_utils::get_sdf_attribute_required

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

template<typename **T**, typename **SdfPtrT**>
bool `rmf_plugins_utils::get_sdf_attribute_required`(*SdfPtrT* &*sdf*, **const** std::string &*attribute_name*, *T* &*value*)

Template Function rmf_plugins_utils::get_sdf_param_if_available

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

template<typename **T**, typename **SdfPtrT**>
void `rmf_plugins_utils::get_sdf_param_if_available`(*SdfPtrT &sdf*, **const** std::string *&parameter_name*, *T &value*)

Template Function rmf_plugins_utils::get_sdf_param_required

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

template<typename **T**, typename **SdfPtrT**>
bool `rmf_plugins_utils::get_sdf_param_required`(*SdfPtrT &sdf*, **const** std::string *&parameter_name*, *T &value*)

Template Function rmf_plugins_utils::make_response

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

template<typename **ResultMsgT**>
std::shared_ptr<*ResultMsgT*> `rmf_plugins_utils::make_response`(uint8_t *status*, **const** double *sim_time*, **const** std::string *&request_guid*, **const** std::string *&guid*)

Function rmf_plugins_utils::simulation_now

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_utils.hpp

Function Documentation

rclcpp::Time `rmf_plugins_utils::simulation_now`(double *t*)

Template Function rmf_robot_sim_common::get_element_val_if_present

- Defined in file_latest_rmf_robot_sim_common_include_rmf_robot_sim_common_slotcar_common.hpp

Function Documentation

template<typename **SdfPtrT**, typename **valueT**>
bool `rmf_robot_sim_common::get_element_val_if_present`(*SdfPtrT* &*_sdf*, **const** std::string &*_element_name*, *valueT* &*_val*)

1.3.5 Typedefs

Typedef crowd_simulator::AgentPtr

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_crowd_simulator_common.hpp

Typedef Documentation

using `crowd_simulator::AgentPtr` = std::shared_ptr<Menge::Agents::BaseAgent>

Typedef rmf_building_sim_common::DoorMode

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_door_common.hpp

Typedef Documentation

typedef rmf_door_msgs::msg::DoorMode `rmf_building_sim_common::DoorMode`

Typedef rmf_building_sim_common::DoorRequest

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_door_common.hpp

Typedef Documentation

typedef rmf_door_msgs::msg::DoorRequest `rmf_building_sim_common::DoorRequest`

Typedef rmf_building_sim_common::DoorState

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_door_common.hpp

Typedef Documentation

typedef rmf_door_msgs::msg::DoorState `rmf_building_sim_common::DoorState`

Typedef rmf_building_sim_common::LiftRequest

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_lift_common.hpp

Typedef Documentation

using `rmf_building_sim_common::`**`LiftRequest`** = rmf_lift_msgs::msg::LiftRequest

Typedef rmf_building_sim_common::LiftState

- Defined in file_latest_rmf_building_sim_common_include_rmf_building_sim_common_lift_common.hpp

Typedef Documentation

using `rmf_building_sim_common::`**`LiftState`** = rmf_lift_msgs::msg::LiftState

INDEX

C

R