

RosInterface Plugin API reference

This plugin provides an interface with the ROS API

simROS.advertise

Description	Advertise a topic and create a topic publisher.
Lua synopsis	int publisherHandle=simROS.advertise(string topicName, string topicType, int queueSize=1, bool latch=false)
Lua parameters	topicName (string): topic name, e.g.: '/cmd_vel' topicType (string): topic type, e.g.: 'geometry_msgs::Twist' queueSize (int, default: 1): (optional) queue size latch (bool, default: false): (optional) latch topic
Lua return values	publisherHandle (int): a handle to the ROS publisher
See also	

simROS.advertiseService

Description	Advertise a service and create a service server.
Lua synopsis	int serviceServerHandle=simROS.advertiseService(string serviceName, string serviceType, string serviceCallback)
Lua parameters	serviceName (string): topic name, e.g.: '/cmd_vel' serviceType (string): topic type, e.g.: 'geometry_msgs::Twist' serviceCallback (string): name of the callback function, which will be called with a single argument of type table containing the service request payload; it must return another table containing the response
Lua return values	serviceServerHandle (int): a handle to the ROS service server
See also	

simROS.call

Description	Call the service associated with this service client.
Lua synopsis	table result=simROS.call(int serviceClientHandle, table request)
Lua parameters	serviceClientHandle (int): the service client handle request (table of): the message to publish
Lua return values	result (table of): the response message, if the call succeeded
See also	

simROS.deleteParam

Description	Delete a parameter in the ROS Parameter Server.
Lua synopsis	simROS.deleteParam(string name)
Lua parameters	name (string): name of the parameter
Lua return values	-
See also	

simROS.getParamBool

Description	Retrieve a boolean parameter from the ROS Parameter Server.
Lua synopsis	bool exists, bool value=simROS.getParamBool(string name, bool defaultValue=false)
Lua parameters	name (string): name of the parameter defaultValue (bool, default: false): default value returned when parameter does not exist
Lua return values	exists (bool): true if the param exists otherwise false value (bool): the value of the requested parameter
See also	

simROS.getParamDouble	
Description	Retrieve a double parameter from the ROS Parameter Server.
Lua synopsis	bool exists, double value=simROS.getParamDouble(string name, double defaultValue=0.0)
Lua parameters	name (string): name of the parameter defaultValue (double, default: 0.0): default value returned when parameter does not exist
Lua return values	exists (bool): true if the param exists otherwise false value (double): the value of the requested parameter
See also	

simROS.getParamInt	
Description	Retrieve an integer parameter from the ROS Parameter Server.
Lua synopsis	bool exists, int value=simROS.getParamInt(string name, int defaultValue=0)
Lua parameters	name (string): name of the parameter defaultValue (int, default: 0): default value returned when parameter does not exist
Lua return values	exists (bool): true if the param exists otherwise false value (int): the value of the requested parameter
See also	

simROS.getParamString	
Description	Retrieve a string parameter from the ROS Parameter Server.
Lua synopsis	bool exists, string value=simROS.getParamString(string name, string defaultValue=)
Lua parameters	name (string): name of the parameter defaultValue (string, default:): default value returned when parameter does not exist
Lua return values	exists (bool): true if the param exists otherwise false value (string): the value of the requested parameter
See also	

simROS.getTime	
Description	Return the current ROS time (i.e. the time returned by ros::Time::now()).
Lua synopsis	double time=simROS.getTime(int flag=0)
Lua parameters	flag (int, default: 0): unused: set to zero
Lua return values	time (double): ROS time expressed in seconds
See also	

simROS.hasParam	
Description	Check wether a parameter exists in the ROS Parameter Server.
Lua synopsis	bool exists=simROS.hasParam(string name)
Lua parameters	name (string): name of the parameter
Lua return values	exists (bool): true if the parameter exists, false otherwise
See also	

simROS.imageTransportAdvertise	
Description	Advertise a topic and create a topic publisher using ImageTransport.
Lua synopsis	int publisherHandle=simROS.imageTransportAdvertise(string topicName, int queueSize=1)
Lua parameters	topicName (string): topic name, e.g.: '/cmd_vel' queueSize (int, default: 1): (optional) queue size
Lua return values	publisherHandle (int): a handle to the ROS publisher
See also	

simROS.imageTransportPublish	
Description	Publish a message on the topic associated with this publisher using ImageTransport.

Lua synopsis	simROS.imageTransportPublish(int publisherHandle, string data, int width, int height, string frame_id)
Lua parameters	publisherHandle (int): the publisher handle data (string): the image data width (int): image width height (int): image height frame_id (string): frame id
Lua return values	-
See also	

simROS.imageTransportShutdownPublisher

Description	Shutdown the advertisement associated with this publisher using ImageTransport.
Lua synopsis	simROS.imageTransportShutdownPublisher(int publisherHandle)
Lua parameters	publisherHandle (int): the publisher handle
Lua return values	-
See also	

simROS.imageTransportShutdownSubscriber

Description	Unsubscribe the callback associated with this subscriber using ImageTransport.
Lua synopsis	simROS.imageTransportShutdownSubscriber(int subscriberHandle)
Lua parameters	subscriberHandle (int): the subscriber handle
Lua return values	-
See also	

simROS.imageTransportSubscribe

Description	Subscribe to a topic using ImageTransport.
Lua synopsis	int subscriberHandle=simROS.imageTransportSubscribe(string topicName, string topicCallback, int queueSize=1)
Lua parameters	topicName (string): topic name, e.g.: '/cmd_vel' topicCallback (string): name of the callback function, which will be called as: topicCallback(string data, number width, number height) queueSize (int, default: 1): (optional) queue size
Lua return values	subscriberHandle (int): a handle to the ROS subscriber
See also	

simROS.publish

Description	Publish a message on the topic associated with this publisher.
Lua synopsis	simROS.publish(int publisherHandle, table message)
Lua parameters	publisherHandle (int): the publisher handle message (table of): the message to publish
Lua return values	-
See also	

simROS.publisherTreatUInt8ArrayAsString

Description	After calling this function, this publisher will treat uint8 arrays as string. Using strings should be in general much faster that using int arrays in Lua.
Lua synopsis	simROS.publisherTreatUInt8ArrayAsString(int publisherHandle)
Lua parameters	publisherHandle (int): the publisher handle
Lua return values	-
See also	

simROS.searchParam

Description	Search a parameter in the ROS Parameter Server, looking in the closest namespace.

Lua synopsis	bool found, string name=simROS.searchParam(string name)
Lua parameters	name (string): name of the parameter
Lua return values	found (bool): true if the parameter has been found name (string): name of the parameter which has been found
See also	

simROS.sendTransform

Description	Publish a TF transformation between frames.
Lua synopsis	simROS.sendTransform(table transform)
Lua parameters	transform (table of): the transformation expressed as a geometry_msgs/TransformStamped message, i.e. {header={stamp=timeStamp, frame_id='...'}, child_frame_id='...', transform={translation={x=..., y=..., z=...}, rotation={x=..., y=..., z=..., w=...}}}
Lua return values	-
See also	

simROS.sendTransforms

Description	Publish several TF transformations between frames.
Lua synopsis	simROS.sendTransforms(table transforms)
Lua parameters	transforms (table of): an array of geometry_msgs/TransformStamped messages
Lua return values	-
See also	

simROS.serviceClient

Description	Create a service client.
Lua synopsis	int serviceClientHandle=simROS.serviceClient(string serviceName, string serviceType)
Lua parameters	serviceName (string): topic name, e.g.: '/cmd_vel' serviceType (string): topic type, e.g.: 'geometry_msgs::Twist'
Lua return values	serviceClientHandle (int): a handle to the ROS service client
See also	

simROS.serviceClientTreatUInt8ArrayAsString

Description	After calling this function, this service client will treat uint8 arrays as string. Using strings should be in general much faster that using int arrays in Lua.
Lua synopsis	simROS.serviceClientTreatUInt8ArrayAsString(int serviceClientHandle)
Lua parameters	serviceClientHandle (int): the service client handle
Lua return values	-
See also	

simROS.serviceServerTreatUInt8ArrayAsString

Description	After calling this function, this service server will treat uint8 arrays as string. Using strings should be in general much faster that using int arrays in Lua.
Lua synopsis	simROS.serviceServerTreatUInt8ArrayAsString(int serviceServerHandle)
Lua parameters	serviceServerHandle (int): the service server handle
Lua return values	-
See also	

simROS.setParamBool

Description	Set a boolean parameter in the ROS Parameter Server.
Lua synopsis	simROS.setParamBool(string name, bool value)
Lua parameters	name (string): name of the parameter value (bool): value of the parameter
Lua return values	-

See also

simROS.setParamDouble

Description	Set a double parameter in the ROS Parameter Server.
Lua synopsis	simROS.setParamDouble(string name, double value)
Lua parameters	name (string): name of the parameter value (double): value of the parameter
Lua return values	-
See also	

simROS.setParamInt

Description	Set a integer parameter in the ROS Parameter Server.
Lua synopsis	simROS.setParamInt(string name, int value)
Lua parameters	name (string): name of the parameter value (int): value of the parameter
Lua return values	-
See also	

simROS.setParamString

Description	Set a string parameter in the ROS Parameter Server.
Lua synopsis	simROS.setParamString(string name, string value)
Lua parameters	name (string): name of the parameter value (string): value of the parameter
Lua return values	-
See also	

simROS.shutdownPublisher

Description	Shutdown the advertisement associated with this publisher.
Lua synopsis	simROS.shutdownPublisher(int publisherHandle)
Lua parameters	publisherHandle (int): the publisher handle
Lua return values	-
See also	

simROS.shutdownServiceClient

Description	Shutdown the service client.
Lua synopsis	simROS.shutdownServiceClient(int serviceClientHandle)
Lua parameters	serviceClientHandle (int): the service client handle
Lua return values	-
See also	

simROS.shutdownServiceServer

Description	Shutdown the service server.
Lua synopsis	simROS.shutdownServiceServer(int serviceServerHandle)
Lua parameters	serviceServerHandle (int): the service server handle
Lua return values	-
See also	

simROS.shutdownSubscriber

Description	Unsubscribe the callback associated with this subscriber.

Lua synopsis	simROS.shutdownSubscriber(int subscriberHandle)
Lua parameters	subscriberHandle (int): the subscriber handle
Lua return values	-
See also	

simROS.subscribe

Description	Subscribe to a topic.
Lua synopsis	int subscriberHandle=simROS.subscribe(string topicName, string topicType, string topicCallback, int queueSize=1)
Lua parameters	topicName (string): topic name, e.g.: '/cmd_vel' topicType (string): topic type, e.g.: 'geometry_msgs::Twist' topicCallback (string): name of the callback function, which will be called with a single argument of type table containing the message payload, e.g.: {linear={x=1.5, y=0.0, z=0.0}, angular={x=0.0, y=0.0, z=-2.3}} queueSize (int, default: 1): (optional) queue size
Lua return values	subscriberHandle (int): a handle to the ROS subscriber
See also	

simROS.subscriberTreatUInt8ArrayAsString

Description	After calling this function, this subscriber will treat uint8 arrays as string. Using strings should be in general much faster that using int arrays in Lua.
Lua synopsis	simROS.subscriberTreatUInt8ArrayAsString(int subscriberHandle)
Lua parameters	subscriberHandle (int): the subscriber handle
Lua return values	-
See also	

Script functions

Script functions are used to call some lua code from the plugin side (tipically used for event handlers).

subscriberCallback

Description	Callback for ROS subscriber.
Lua synopsis	simROS.subscriberCallback(table message)
Lua parameters	message (table of): the topic payload (i.e. the message)
Lua return values	-
See also	

imageTransportCallback

Description	Callback for ROS ImageTransport subscriber.
Lua synopsis	simROS.imageTransportCallback(string data, int width, int height)
Lua parameters	data (string): the image data width (int): image width height (int): image height
Lua return values	-
See also	