Orocos RTT Cheatsheet

GENERAL COMMANDS

help <command>	Prints detailed information for a command.
path(string)	Adds a path to the search path.
import(string)	Imports a package by name.
displayComponentTypes	Displays the components that can be loaded.
ls	Introspect a components ext. configuration.
cd	Navigate through components.
quit	Terminates the deployer.

COMPONENT CONFIGURATION

loadComponent(inst_name, comp_type)	Instantiates a component.
unloadComponent(inst_name)	Destroys an instance by name.
setActivity(inst_name, period, priority, scheduler)	Make a component runnable (periodically).
<component>.setperiod(double)	Sets the period of a component (in seconds).
<component>.setpriority(int)	Sets the priority of a component.
<component>.configure()	Configures a component (calls configureHook).
<component>.start()	Starts a component (calls startHook).
<component>.stop()	Stops a component (calls stopHook).
<component>.cleanup()	Cleans a component up (calls cleanupHook).

SERVICES

loadService(target_inst_name, service_name)	Load a service into the target component instance.
SERVICE: dot (“rtt_dot_service” package)	Package to generate a component graph.
dot.dot_file	Property for output file for the (.dot) graph.
dot.generate()	Generates a component graph.
SERVICE: scripting (built-in)	Service to load .ops from file.
scripting.runScript(pathToFile)	Executes an (.ops) script from file.

COMMUNICATION

var ConnPolicy varname	Creates a *default* connection policy.
connect(output_port, input_port, cp)	Connect two ports with a connection policy.
stream(oro_port, rsb.transport.socket.scope(scope_uri))	Connects a port to the RSB middleware.

ENVIRONMENT VARIABLES

RTT_COMPONENT_PATH	Search path for Orocos packages.
GAZEBO_MODEL_PATH	Search path for Gazebo model repos.
GAZEBO_PLUGIN_PATH	Search path for Gazebo plugins.

RSB RELATED COMMANDS

rsb0.14 server	Starts the RSB socket server.
rsb0.14 logger socket:/	Starts a logging tool for RSB.

USE THE OCL::FILELOGGER

loadComponent("file_reporter", "OCL::FileReporting")	Load the file logging component.
loadService("file_reporter", "marshalling")	Attach the marshalling service for rt-save file operations.
connectPeers("file_reporter","controller")	Connect to component to be logged.
file_reporter.reportComponent("controller")	Explicitly tell the logger to report on the specific component.
file_reporter.configure	
file_reporter.start	