Tutorial for usage of the RCLL with the Fawkes Robotics Framework in the Gazebo simulation and YAGI

Nicolas Limpert

[1] Fachhochschule Aachen - University of Applied Sciences
[2] MASCOR Institute

Abstract. This document is supposed to serve as a tutorial for getting started with the software stack provided by the Team Carologistics[3] in 2015.
Fawkes[4] is a robotics framework and fully capable of handling the RoboCup Logistics League (RCLL) [5].
The Fawkes implementation meant for the Festo Robotino robots additionally make use of an agent written in the C-Language Integrated Production System (CLIPS).
In this tutorial we instead want to make use of YAGI (Yet Another Golog Interpreter)[6].

1 Installation

Most of this section is gathered from the official documentation at the Fawkes Robotics Website:

- `https://trac.fawkesrobotics.org/wiki/InstallingFawkes`
- `https://trac.fawkesrobotics.org/wiki/FawkesOnFedora`
- `https://trac.fawkesrobotics.org/wiki/Carologistics/Gazsim-Setup-2015`

1.1 Fawkes Dependencies

It is preferred to make use of Fedora 23 as the time of this writing. Fawkes is mainly developed against Fedora and using Fedora 23 eases up the installation process due to readily available dependencies solved by Fedora's repositories. Although you can of course feel free to have a look at the aforementioned website in order to install it on any other Unix-based operating system.
We will consider the installation in Fedora 23 in this tutorial.

After a fresh installation of Fedora perform the following
(refer to) in order to install the dependencies required to build Fawkes:

[3] http://www.carologistics.org
[4] http://www.fawkesrobotics.org
[5] http://www.robocup-logistics.org
[6] http://yagi.ist.tugraz.at/

```
$ sudo dnf groupinstall development-tools development-libs
$ sudo dnf install fawkes-devenv
$ sudo rpm -e --nodeps tolua++ tolua++-devel
$ sudo dnf install compat-lua compat-lua-devel\
 compat-tolua++ compat-tolua++-devel
```

1.2 ROS (optional)

For debugging reasons it is recommended to make use of the visualization functionalities of the Robot Operating System called Rviz.
To make use of Rviz we have to install the full ROS (the currently used version is "Jade").
In order to install ROS (see `http://wiki.ros.org/jade/Installation/Source`) we want to do the following;

```
$ sudo mkdir /opt/ros/catkin_ws_jade
$ sudo mkdir /opt/ros/jade
$ sudo chown <yourusername>:<yourusername> /opt/ros/*
$ cd /opt/ros/catkin_ws_jade

$ rosinstall_generator desktop_full --rosdistro\
 jade --deps --wet-only\
 --tar > jade-desktop-full-wet.rosinstall
$ wstool init -j8 src jade-desktop-full-wet.rosinstall

$ rosinstall_generator navigation --rosdistro\
 jade --deps --wet-only\
 --tar > jade-navigation.rosinstall
$ rosinstall_generator ar_track_alvar --rosdistro\
 jade --deps --wet-only\
 --tar > jade-ar_track_alvar.rosinstall
$ wstool merge -t src jade-navigation.rosinstall
$ wstool merge -t src jade-ar_track_alvar.rosinstall
$ wstool update -t src
```

Where <yourusername> is the username that was given during the installation process of Fedora.
Next, build the whole workspace (this can take some minutes):

```
$ cd /opt/ros/catkin_ws_jade
$ ./src/catkin/bin/catkin_make_isolated --install\
 --install-space=/opt/ros/jade -DCMAKE_BUILD_TYPE=Release
```

1.3 Gazebo Models and Plugins

There are custom models and plugins used by the simulation software Gazebo. E.g.

```
$ cd ~
$ git clone git@github.com:robocup-logistics/gazebo-rcll.git
$ cd gazebo-rcll/plugins
$ make -j4
```

1.4 LLSF-Refbox

The RCLL-Refbox [7] is the software responsible for handling the game's scores to each of the teams.
In order to install it do the following;

```
$ cd ~
$ git clone http://git.fawkesrobotics.org/llsf-refbox.git
$ cd llsf-refbox
$ make -j4
```

1.5 Environmental variables

In order to promote the LLSF-Refbox installation directory, the Gazebo models and plugins path and so on we have to modify the file /.bashrc. Insert the following contents at the bottom of this file:

```
source /usr/share/gazebo/setup.bash
source /opt/ros/jade/setup.bash
export FAWKES_DIR=~/fawkes-robotino
export GAZEBO_RCLL=~/gazebo-rcll
export GAZEBO_PLUGIN_PATH=$GAZEBO_PLUGIN_PATH:\
 $GAZEBO_RCLL/plugins/lib/gazebo
export GAZEBO_MODEL_PATH=$GAZEBO_RCLL/models
export GAZEBO_MODEL_PATH=$GAZEBO_MODEL_PATH:\
 $GAZEBO_RCLL/models/carologistics

export LLSF_REFBOX_DIR=~/llsf-refbox
export GAZEBO_WORLD_PATH=\
 1~/gazebo-rcll/worlds/carologistics/llsf.world
```

1.6 YAGI

YAGI's website `http://yagi.ist.tugraz.at/` provides access to the release of YAGI including the agent for the LLSF2014.
As we are in need of an agent for the LLSF2015 we continued development of the agent to handle the communication used by the Refbox in 2015.
The installation can be done in the following way:

[7] http://www.robocup-logistics.org/refbox

```
$ sudo dnf install antlr3-C-devel
$ git clone git@github.com:balamesh/yagi.git
$ cd yagi
$ git checkout nlimpert/llsf2015
$ mkdir build
$ cd build
$ cmake ..
$ make -j4
```

1.7 Fawkes Robotino

The aforementioned Fawkes Robotics Software-stack is provided by the team Carologistics in a tarball[8].
In order to get / install this Software-stack perform the following:

```
$ wget\
 https://files.fawkesrobotics.org/releases/fawkes-robotino-2015.tar.bz2
$ tar -xjf fawkes-robotino-2015.tar.bz2
$ cd fawkes-robotino
$ make -j4
```

Important: After installation make sure you insert the additional skill `explore_zone_wrapped` required to call exploration goals out of YAGI. This skill and the required `init.lua` is found in the repository `https://github.com/balamesh/ti_praktikum_2016`.
You need to copy the files `explore_zone_wrapped.lua` and `init.lua` to `fawkes-robotino/src/lua/skills/robotino/`.

[8] https://www.fawkesrobotics.org/blog/2016/02/22/rcll2015-release/

2 Usage

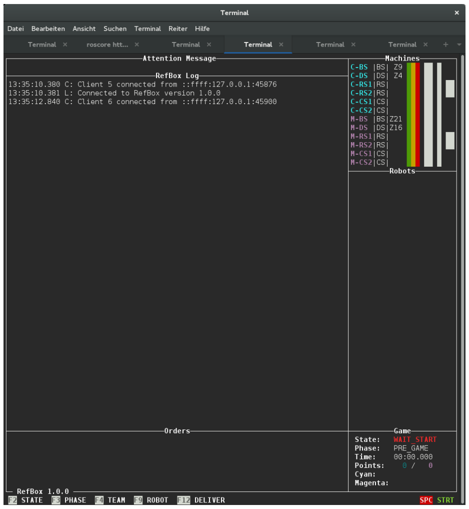
2.1 Starting the simulation

In order to start the whole simulation including the ROS Environment, the Gazebo simulation software, the LLSF-Refbox and an instance of one Fawkes in order to control the first robot in the simulation do the following;

```
$ cd fawkes-robotino/bin
$ ./gazsim.bash -x start -r -n 1
```

There should be 2 windows opening. One of them is a set of terminals including the following:

1. `gazebo including the world-plugin`
2. `roscore`
3. `refbox-shell`
4. `refbox-user-interface`
5. `fawkes`
6. `fawkes-comm`



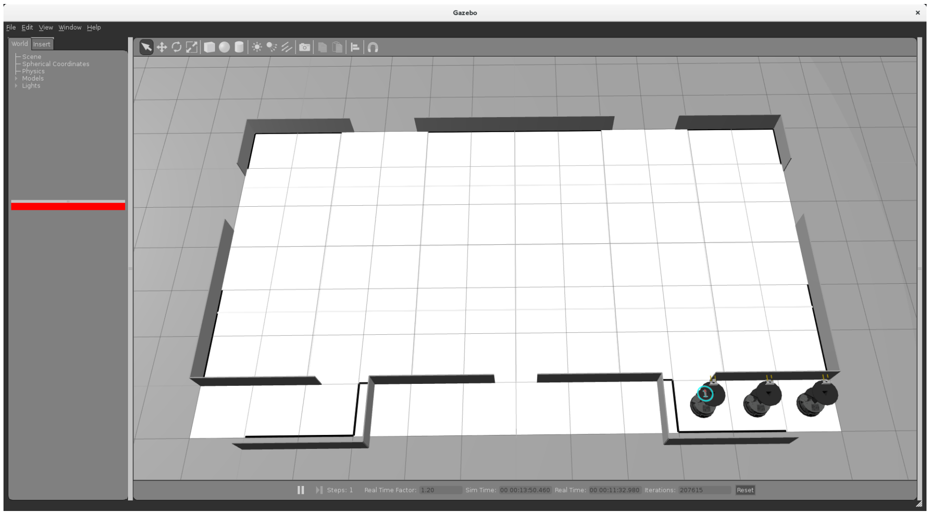
```
13:35:10.380 C: Client 5 connected from ::ffff:127.0.0.1:45876
13:35:10.381 L: Connected to RefBox version 1.0.0
13:35:12.840 C: Client 6 connected from ::ffff:127.0.0.1:45900
```

```
C-BS |BS| Z9
C-DS |DS| Z4
C-RS1|RS|
C-RS2|RS|
C-CS1|CS|
C-CS2|CS|
M-BS |BS|Z21
M-DS |DS|Z16
M-RS1|RS|
M-RS2|RS|
M-CS1|CS|
M-CS2|CS|
```

```
State:     WAIT_START
Phase:     PRE_GAME
Time:      00:00.000
Points:      0 /   0
Cyan:
Magenta:
```

Fig. 1. Starting screen of the LLSF-Refbox in the simulation

An example of the refbox starting screen is shown in Figure 1.



Steps: 1 Real Time Factor: 1.20 Sim Time: 00 00:13:50.460 Real Time: 00 00:11:32.980 Iterations: 207615

Fig. 2. Refbox user-interface

The other window is the Gazebo simulation environment, shown in figure 2.

2.2 Launching the simulation

In order to start the game we have to tell the refbox that we first would like to announce the desired team. Do the following:

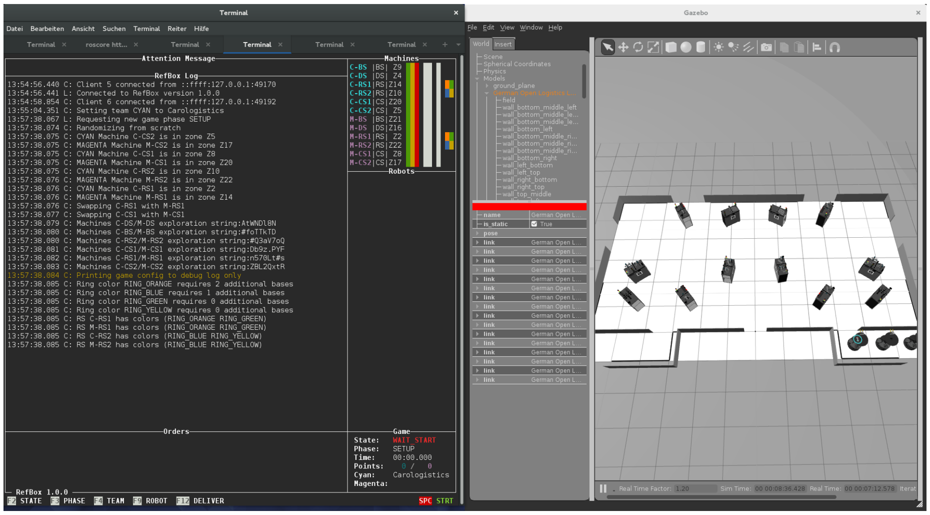
1. Switch to the window shown in figure 1
2. Press F4
3. Press "Return" twice

This sets the team CYAN to "Carologistics".

Change the phase from `PRE_GAME` to `SETUP` by pressing `F3` and afterwards set the game state from `WAIT_START` to `RUNNING` by pressing the `SPACE`-Key.
Important: After changing from the phase `PRE_GAME` to `SETUP` the gazebo-plugin spawns the MPSes. This usually takes a while so be patient until you see all 12 MPSes spawned. A fully launched simulation looks like shown in figure 3.

Launch YAGI Agent Before changing from the `SETUP`-phase to `EXPLORATION` we have to launch the YAGI-Agent. To do this perform the following steps:

1. Launch a new terminal
2. cd to `~/yagi/build`
3. run `./yagi -b FawkesBackend -f ~/yagi/domains/llsf2015/llsf2015.y` (in case the agent you'd like to launch is located at `~/yagi/domains/llsf2015/llsf2015.y`)



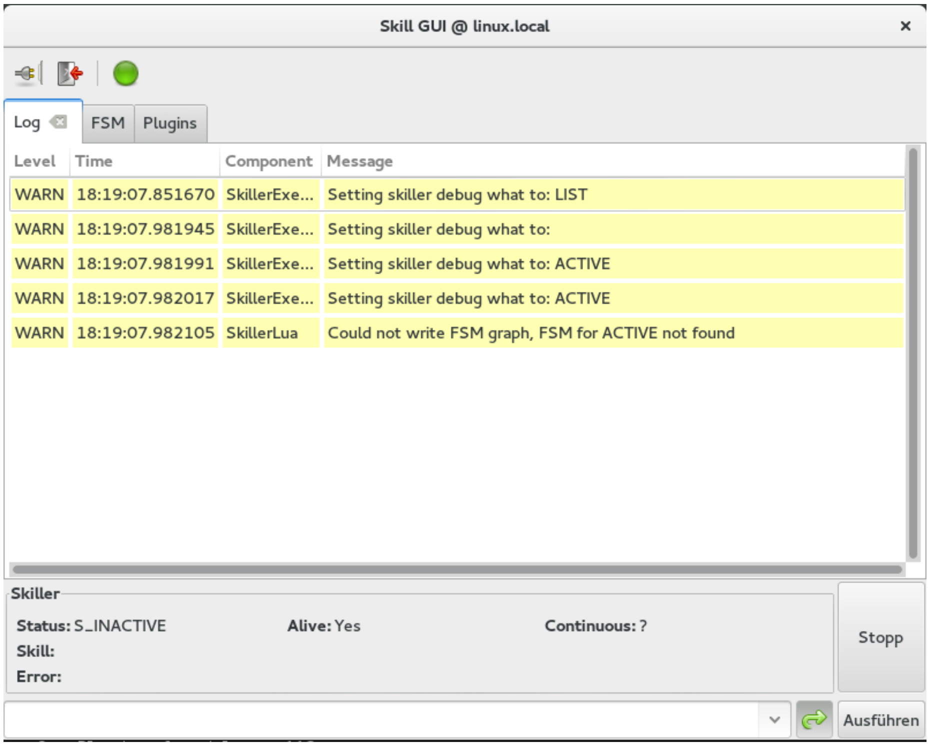
```
Attention Message

RefBox Log
13:54:56.440 C: Client 5 connected from ::ffff:127.0.0.1:49170
13:54:56.441 L: Connected to RefBox version 1.0.0
13:54:58.854 C: Client 6 connected from ::ffff:127.0.0.1:49192
13:55:04.351 C: Setting team CYAN to Carologistics
13:57:38.067 L: Requesting new game phase SETUP
13:57:38.074 C: Randomizing from scratch
13:57:38.075 C: CYAN Machine C-CS2 is in zone Z5
13:57:38.075 C: MAGENTA Machine M-CS2 is in zone Z17
13:57:38.075 C: CYAN Machine C-CS1 is in zone Z8
13:57:38.075 C: MAGENTA Machine M-CS1 is in zone Z20
13:57:38.075 C: CYAN Machine C-RS2 is in zone Z10
13:57:38.076 C: MAGENTA Machine M-RS2 is in zone Z22
13:57:38.076 C: CYAN Machine C-RS1 is in zone Z2
13:57:38.076 C: MAGENTA Machine M-RS1 is in zone Z14
13:57:38.076 C: Swapping C-RS1 with M-RS1
13:57:38.077 C: Swapping C-CS1 with M-CS1
13:57:38.079 C: Machines C-DS/M-DS exploration string:AtWNDl8N
13:57:38.080 C: Machines C-BS/M-BS exploration string:#foTTkTD
13:57:38.080 C: Machines C-RS2/M-RS2 exploration string:#Q3aV7oQ
13:57:38.081 C: Machines C-CS1/M-CS1 exploration string:Db9z.PYF
13:57:38.082 C: Machines C-RS1/M-RS1 exploration string:n570Lt#s
13:57:38.083 C: Machines C-CS2/M-CS2 exploration string:ZBL2QxtR
13:57:38.084 C: Printing game config to debug log only
13:57:38.085 C: Ring color RING_ORANGE requires 2 additional bases
13:57:38.085 C: Ring color RING_BLUE requires 1 additional bases
13:57:38.085 C: Ring color RING_GREEN requires 0 additional bases
13:57:38.085 C: Ring color RING_YELLOW requires 0 additional bases
13:57:38.085 C: RS C-RS1 has colors (RING_ORANGE RING_GREEN)
13:57:38.085 C: RS M-RS1 has colors (RING_ORANGE RING_GREEN)
13:57:38.085 C: RS C-RS2 has colors (RING_BLUE RING_YELLOW)
13:57:38.085 C: RS M-RS2 has colors (RING_BLUE RING_YELLOW)

Machines
C-BS |BS| Z9
C-DS |DS| Z4
C-RS1|RS|Z14
C-RS2|RS|Z10
C-CS1|CS|Z20
C-CS2|CS| Z5
M-BS |BS|Z21
M-DS |DS|Z16
M-RS1|RS| Z2
M-RS2|RS|Z22
M-CS1|CS| Z8
M-CS2|CS|Z17

Robots

Orders

Game
State:    WAIT_START
Phase:    SETUP
Time:     00:00.000
Points:     0 /   0
Cyan:     Carologistics
Magenta:

RefBox 1.0.0
F2 STATE  F3 PHASE  F4 TEAM  F9 ROBOT  F12 DELIVER                SPC STRT
```

Fig. 3. Exemplary launched simulation including Refbox-UI and Gazebo with spawned MPSes

Launch Skillgui (Optional) In order to monitor the executed behaviour by the state machine one can make use of the Skillgui provided by the Fawkes Robotics Framework. After compilation of `fawkes-robotino` one can launch the skillgui by executing `fawkes-robotino/bin/skillgui`.
In the skillgui (shown in figure 4) you have to click the top left button in order to be able to connect to the particular robot. In our case it is the entry `"Fawkes on Gazsim Robotino 1"`. Afterwards click the `FSM`-Header in order to view the state machine.

Launch Rviz (Optional) After running the simulation you can connect to the ROS-Environment running on the simulated robot by making use of `rviz`[9].
The github repository `ti_praktikum_2016` contains a subfolder called `.rviz` which can be copied to the home-folder in order to set the default settings.
In order to run rviz simply type in `rosrun rviz rviz` into a terminal.

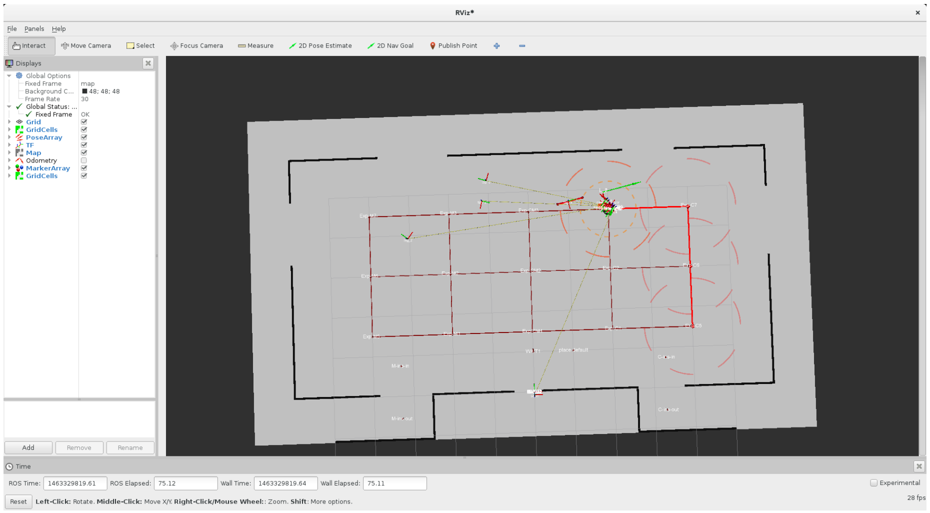
[9] http://wiki.ros.org/rviz



Level	Time	Component	Message
WARN	18:19:07.851670	SkillerExe...	Setting skiller debug what to: LIST
WARN	18:19:07.981945	SkillerExe...	Setting skiller debug what to:
WARN	18:19:07.981991	SkillerExe...	Setting skiller debug what to: ACTIVE
WARN	18:19:07.982017	SkillerExe...	Setting skiller debug what to: ACTIVE
WARN	18:19:07.982105	SkillerLua	Could not write FSM graph, FSM for ACTIVE not found

Status: S_INACTIVE **Alive:** Yes **Continuous:** ?
Skill:
Error:

Fig. 4. Skillgui



Global Options
Fixed Frame: map
Background C...: 48; 48; 48
Frame Rate: 30
Global Status: ...
Fixed Frame: OK

ROS Time: 1463329819.61 ROS Elapsed: 75.12 Wall Time: 1463329819.64 Wall Elapsed: 75.11

Left-Click: Rotate. **Middle-Click:** Move X/Y. **Right-Click/Mouse Wheel:**: Zoom. **Shift**: More options.

28 fps

Fig. 5. rviz