

CaseStudy1

Showing sonar distances on a radar

Antonio Natali

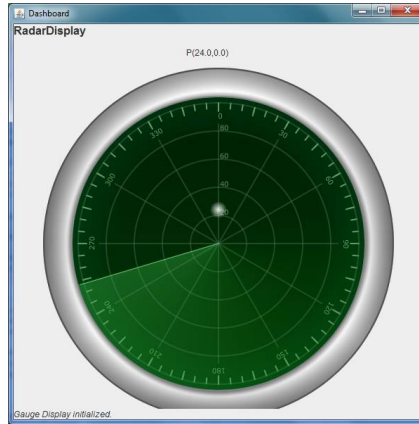
Alma Mater Studiorum – University of Bologna
viale Risorgimento 2, 40136 Bologna, Italy
`antonio.natali@unibo.it`

Table of Contents

CaseStudy1 Showing sonar distances on a radar	1
<i>Antonio Natali</i>	
1 Introduction.....	2
2 The problem to solve	3
55	

1 Introduction

In the project *it.unibo.mbot.intro*, the file [radar/it.unibo.ctxRadarBase.MainCtxRadarBase-1.0.zip](#) includes the implementation of a software system able to display distance values on an output device that simulates the screen of a radar.



To run the application, unzip the file into a working directory and execute

```
java -jar it.unibo.qactor.radar-1.0.jar
```

In the same project *it.unibo.mbot.intro*, the file [src/radarUsage.qa](#) includes a test of the radar application.

```
1  /* =====
2  * radarUsage.qa
3  * =====
4  */
5  System radarUsage
6  Dispatch polarMsg : p( Distance, Angle )
7  Event   polar     : p( Distance, Angle )
8
9  Context ctxRadarUsage ip [ host="localhost" port=8022 ]
10 Context ctxRadarBase ip [ host="localhost" port=8033 ] -standalone
11
12 QActor radartest context ctxRadarUsage {
13 Rules{
14   p(80,0). p(80,30). p(30,50). p(80,60). p(60,70).
15   p(80,90). p(80,160). p(10,130). p(80,150). p(80,180).
16 }
17 Plan init normal [
18   println("radartest STARTS ")
19 ]
20 switchTo dotest
21
22 Plan dotest [
23   delay 1000 ;
24   [ ?? p(X,Y) ] println( sending(p(X,Y)) );
25   [ ?? p(X,Y) ] sendto radarguibase in ctxRadarBase -m polarMsg : p( X, Y ) else endPlan "radartest ENDS";
26   delay 2000 ;
27   [ ?? p(X,Y) ] println( emitting(p(X,Y)) );
28   [ ?? p(X,Y) ] emit polar : p(X,Y) else endPlan "testDone"
29 ]
30 finally repeatPlan
31 }
```

Listing 1.1. radarUsage.qa

2 The problem to solve

The problem now is the following:

With reference to a `mbot` physical robot working in virtual environment, build an application that sends to the radar the data sensed by the virtual and the real sonars. More specifically:

- the data of the *virtual sonar* `sonar1` must be displayed on the direction of angle=`30`;
- the data of the *virtual sonar* `sonar2` must be displayed on the direction of angle=`120`;
- the data of the *virtual sonar* on the virtual robot must be displayed on the direction of angle=`90` at the fixed distance of `40`;
- the data of the *real sonar* on the physical robot must be displayed on the direction of angle=`0`;