



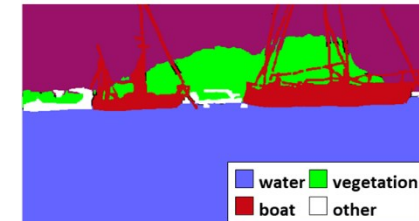
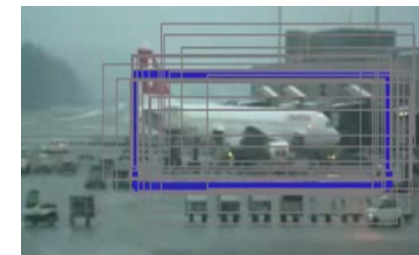
UNIVERSITÀ
di **VERONA**

Dipartimento
di **INFORMATICA**

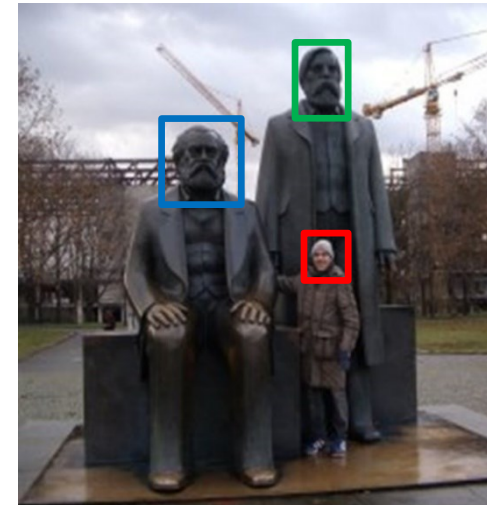
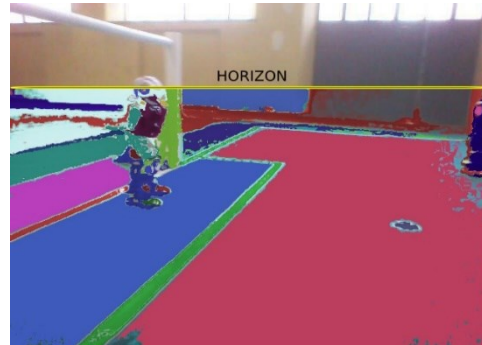
*Corso di Laboratorio Ciberfisico
Modulo di Robot Programming with ROS*

Esercitazione rosbag

Docente:
**Domenico Daniele
Bloisi**



Maggio 2018



References and credits

Queste slide si basano sul materiale contenuto nel libro

YoonSeok Pyo, HanCheol Cho, RyuWoon Jung, TaeHoon Lim,

“ROS Robot Programming - A Handbook Written by TurtleBot3 Developers”

<http://www.robotis.com/service/download.php?no=719>

rosvag

- I dati contenuti nei messaggi ROS possono essere registrati in appositi file
- Il file che contiene i messaggi prende il nome di **bag** e ha l'estensione “.bag”
- Il vantaggio offerto dai file di bag è quello di avere una registrazione che può essere utilizzata **più volte**, riproducendo ogni volta l'esatto scenario operativo in cui la bag è stata registrata

rosvag per i dati dei sensori

- Un esempio dell'utilità dei file di bag è dato dalla registrazione dei messaggi contenenti i dati prodotti dai sensori del robot
- Durante gli esperimenti con il robot reale, i dati dei sensori possono essere registrati in una bag
- I messaggi registrati possono essere poi caricati senza la necessità di ripetere l'esperimento, permettendo così di sviluppare con maggiore facilità algoritmi che richiedano modifiche frequenti dei parametri

Usare rosbag

Rosbag è un programma che crea, riproduce e comprime bag di messaggi. Una bag è un file contenente i dati relativi a messaggi serializzati

- `rosbag record` → record all the topics
- `rosbag info bag-name` → info on the recorded bag
- `rosbag play --pause bag-name` → play the recorded bag, starting paused
- `rosbag play -r #number bag-name` → play the recorded bag at rate #number

Comandi rosbag

Command	Description
<code>rosbag record [OPTION] [TOPIC_NAME]</code>	Record the message of a specific topic on the bsg file
<code>rosbag info [FILE_NAME]</code>	Check information of a bag file
<code>rosbag play [FILE_NAME]</code>	Play a specific bag file
<code>rosbag compress [FILE_NAME]</code>	Compress a specific bag file
<code>rosbag decompress [FILE_NAME]</code>	Decompresses a specific bag file
<code>rosbag filter [INPUT_FILE] [OUTPUT_FILE] [OPTION]</code>	Create a new bag file with the specific content removed
<code>rosbag reindex bag [FILE_NAME]</code>	Reindex
<code>rosbag check bag [FILE_NAME]</code>	Check if the specific bag file can be played in the current system
<code>rosbag fix [INPUT_FILE] [OUTPUT_FILE] [OPTION]</code>	Fix the bag file version that was saved as an incompatible version

Esempio rosbag

Apriamo un
terminal e
digitiamo
\$ roscore

```
roscore http://localhost:11311/
bloisi@bloisi-U36SG:~$ roscore
... logging to /home/bloisi/.ros/log/4d85da46-576b-11e8-9e4c-2709ac87ed01/roslau
nch-bloisi-U36SG-2511.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://localhost:38804/
ros_comm version 1.12.13

SUMMARY
=====

PARAMETERS
* /rostdistro: kinetic
* /rosversion: 1.12.13

NODES

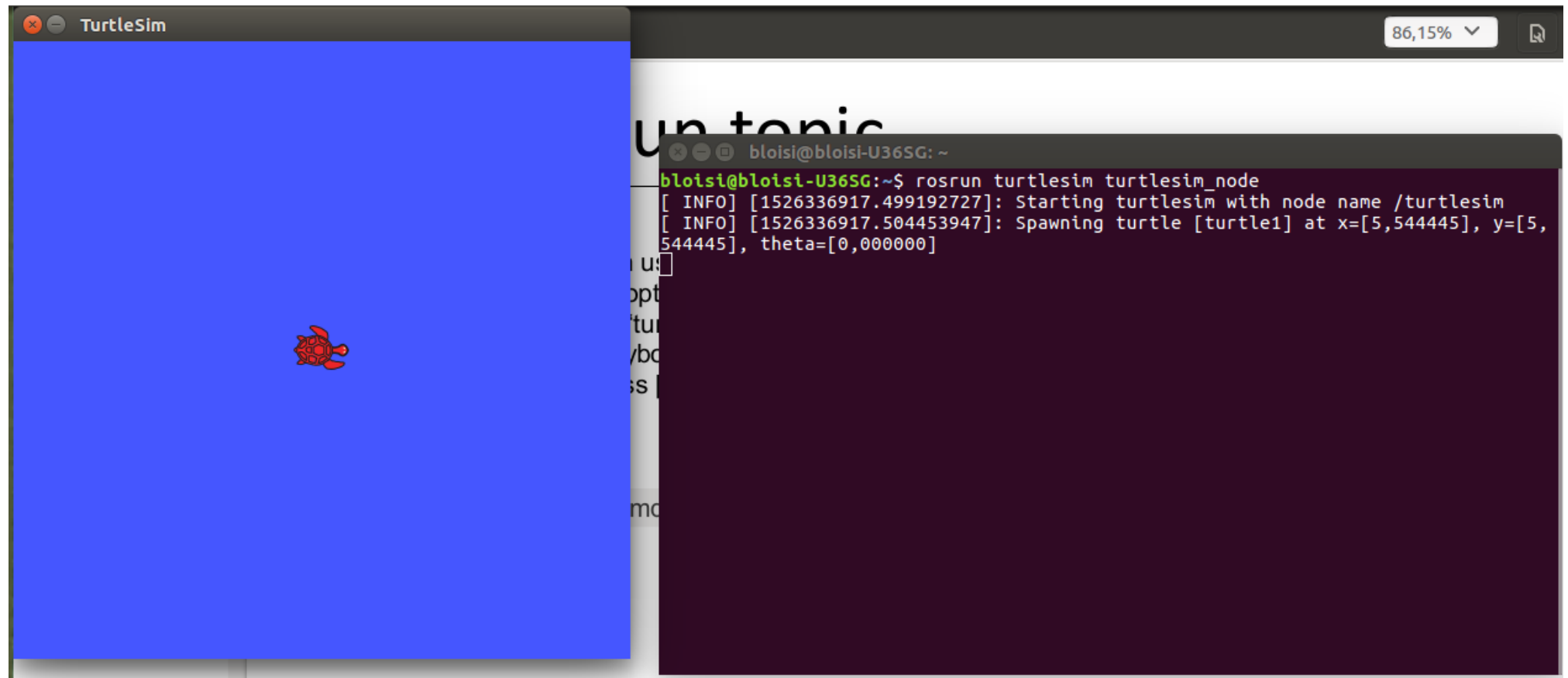
auto-starting new master
process[master]: started with pid [2523]
ROS_MASTER_URI=http://localhost:11311/

setting /run_id to 4d85da46-576b-11e8-9e4c-2709ac87ed01
process[rosout-1]: started with pid [2536]
started core service [/rosout]
```

Esempio rosbag – turtlesim

Apriamo un secondo terminal e digitiamo

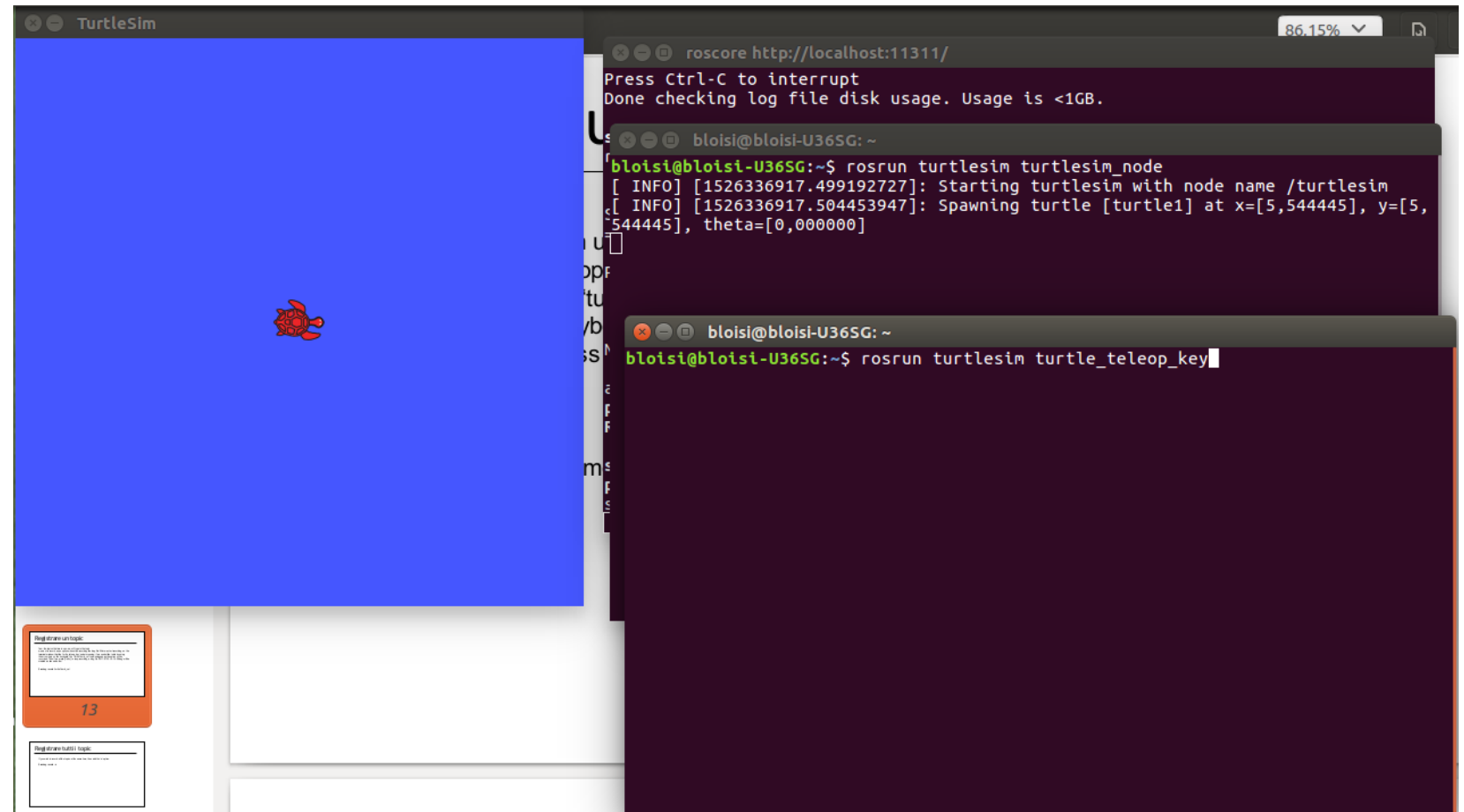
```
$ rosrun turtlesim turtlesim_node
```



Esempio rosbag - teleop

Apriamo un terzo terminal e digitiamo

```
$ rosrun turtlesim turtle_teleop_key
```

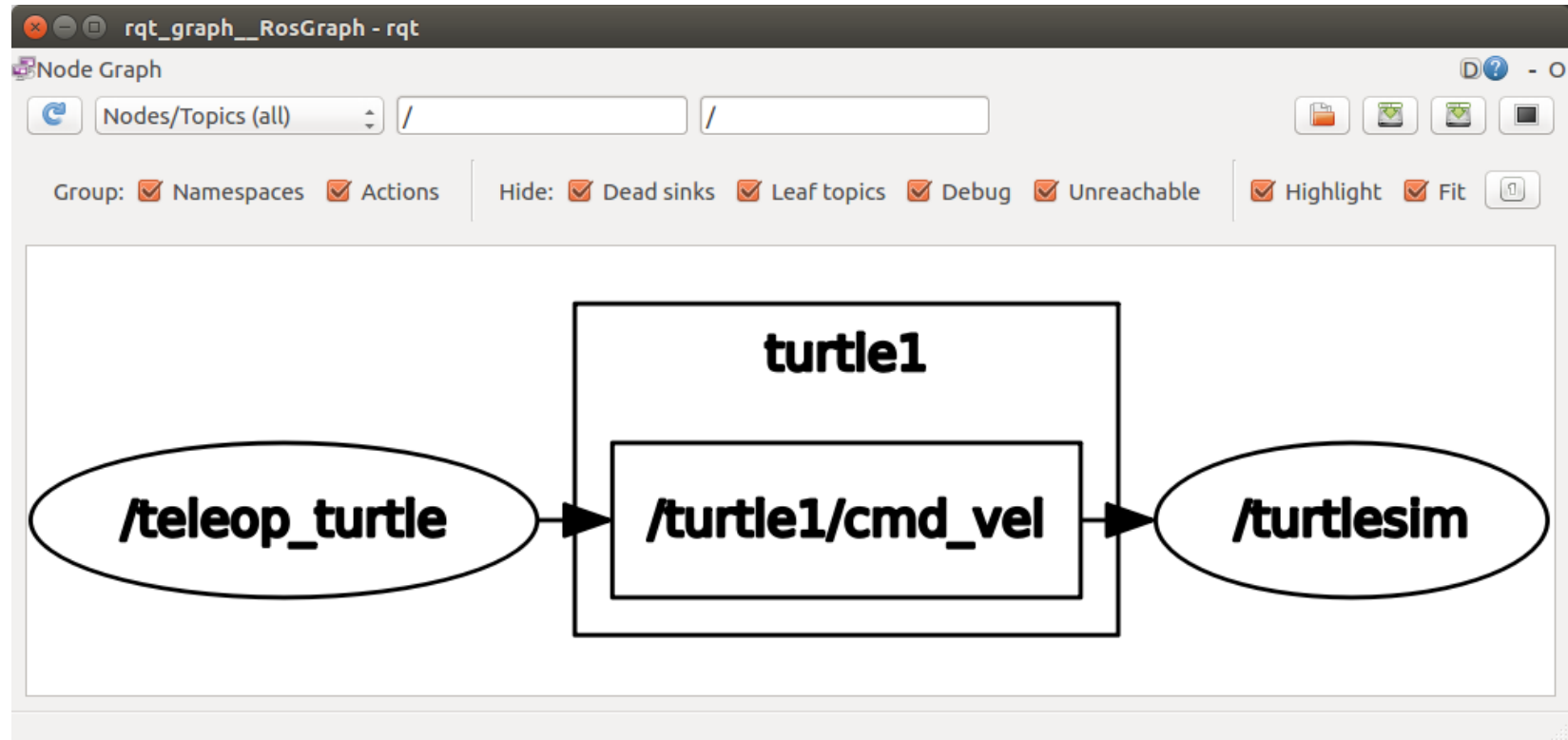


rqt_graph

Apriamo un quarto terminal e digitiamo

```
$ rqt_graph
```

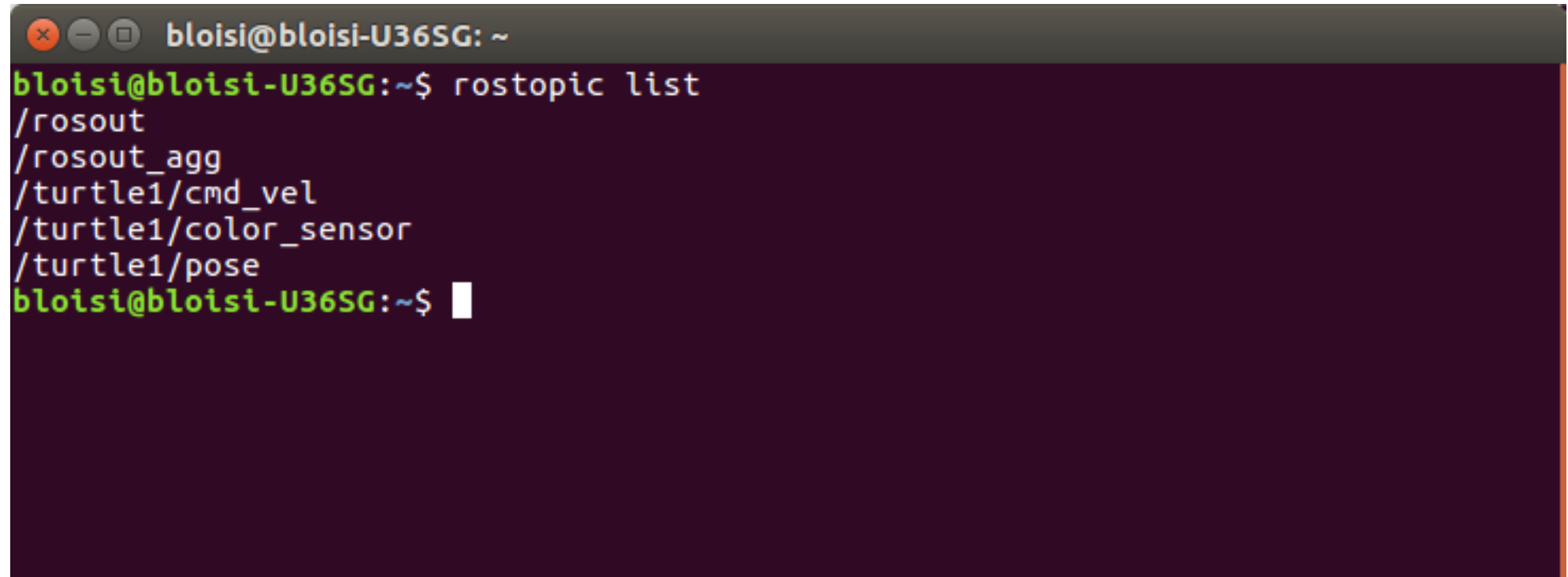
per verificare che i due nodi siano in collegamento tra loro



rostopic

Controlliamo anche a lista dei topic attivi

```
$ rostopic list
```



```
bloisi@bloisi-U36SG: ~  
bloisi@bloisi-U36SG:~$ rostopic list  
/rosout  
/rosout_agg  
/turtle1/cmd_vel  
/turtle1/color_sensor  
/turtle1/pose  
bloisi@bloisi-U36SG:~$
```

Registrazione un topic

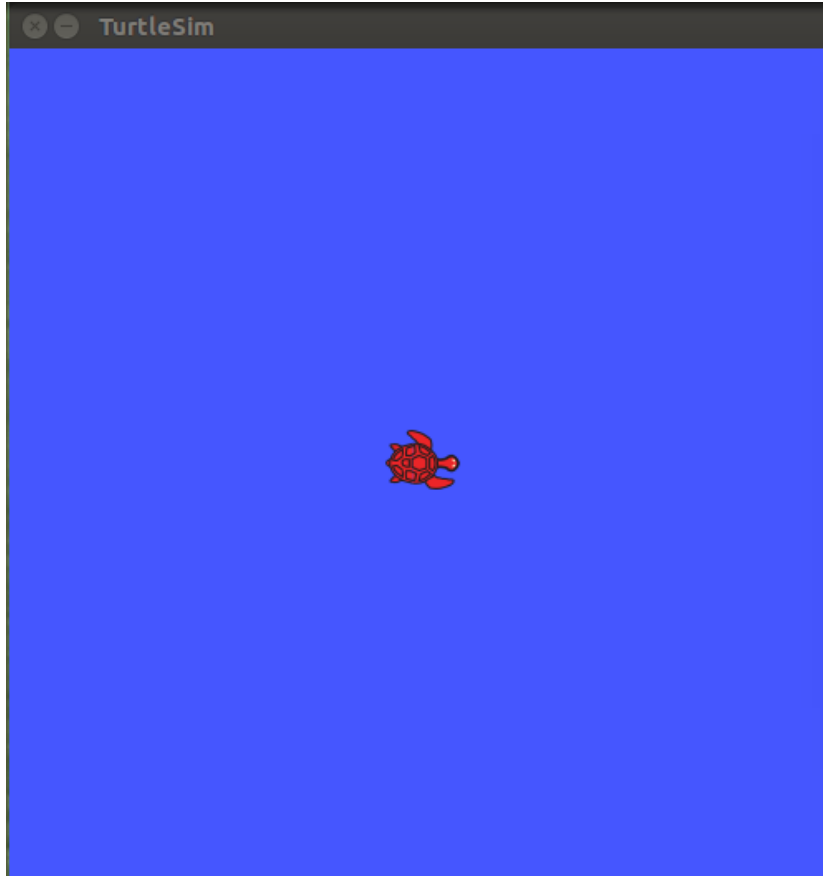
Tra tutti i topic attivi, possiamo scegliere quali registrare usando i comandi e le opzioni di rosbag

```
rosbag record <topic name>
```

Per esempio, per registrare i comandi inviati tramite cmd_vel digitiamo

```
$ rosbag record /turtle1/cmd_vel
```

Esempio - Registrare un topic



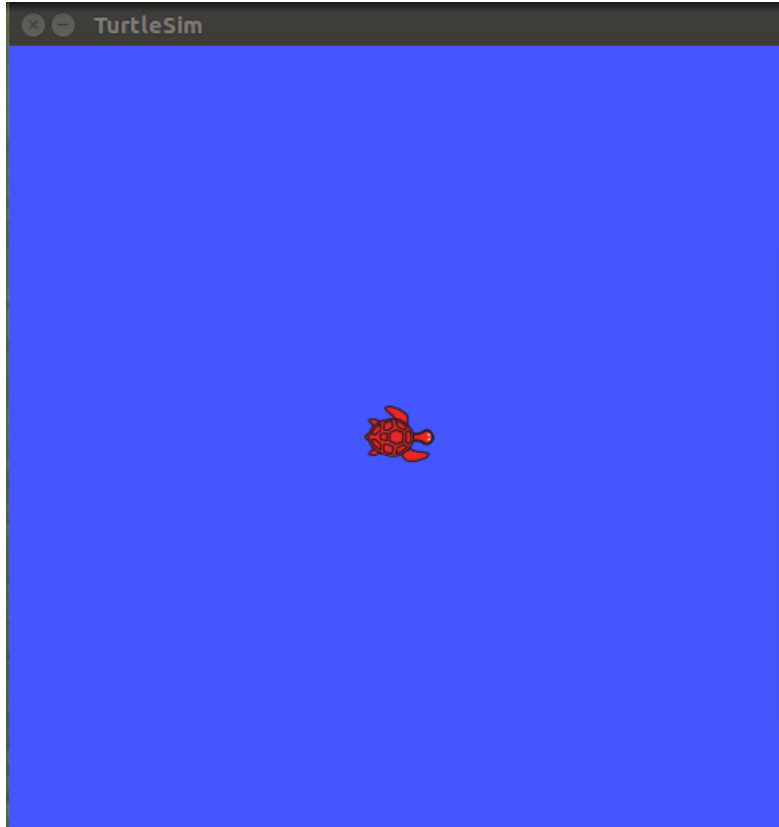
```
roscore http://localhost:11311/
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

bloisi@bloisi-U36SG: ~
bloisi@bloisi-U36SG:~$ roslaunch turtlesim turtlesim_node
[ INFO] [1526336917.499192727]: Starting turtlesim with node name /turtlesim
[ INFO] [1526336917.504453947]: Spawning turtle [turtle1] at x=[5,544445], y=[5,544445], theta=[0,000000]

bloisi@bloisi-U36SG: ~
bloisi@bloisi-U36SG:~$ roslaunch turtlesim turtle_teleop_key
Reading from keyboard
-----
Use arrow keys to move the turtle.

bloisi@bloisi-U36SG: ~
bloisi@bloisi-U36SG:~$ rostopic list
/rosout
/rosout_agg
/turtle1/cmd_vel
/turtle1/color_sensor
/turtle1/pose
bloisi@bloisi-U36SG:~$ rosbag record /turtle1/cmd_vel
```

Esempio – Registrazione in corso...



```
roscore http://localhost:11311/
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

bloisi@bloisi-U36SG: ~
bloisi@bloisi-U36SG:~$ rosrn turtlesim turtlesim_node
[ INFO] [1526336917.499192727]: Starting turtlesim with node name /turtlesim
[ INFO] [1526336917.504453947]: Spawning turtle [turtle1] at x=[5,544445], y=[5,544445], theta=[0,000000]

bloisi@bloisi-U36SG: ~
bloisi@bloisi-U36SG:~$ rosrn turtlesim turtle_teleop_key
Reading from keyboard
-----
Use arrow keys to move the turtle.

bloisi@bloisi-U36SG: ~
bloisi@bloisi-U36SG:~$ rostopic list
/rosout
/rosout_agg
/turtle1/cmd_vel
/turtle1/color_sensor
/turtle1/pose
bloisi@bloisi-U36SG:~$ rosbag record /turtle1/cmd_vel
[ INFO] [1526337037.406675213]: Subscribing to /turtle1/cmd_vel
[ INFO] [1526337037.414123426]: Recording to 2018-05-15-00-30-37.bag.
```

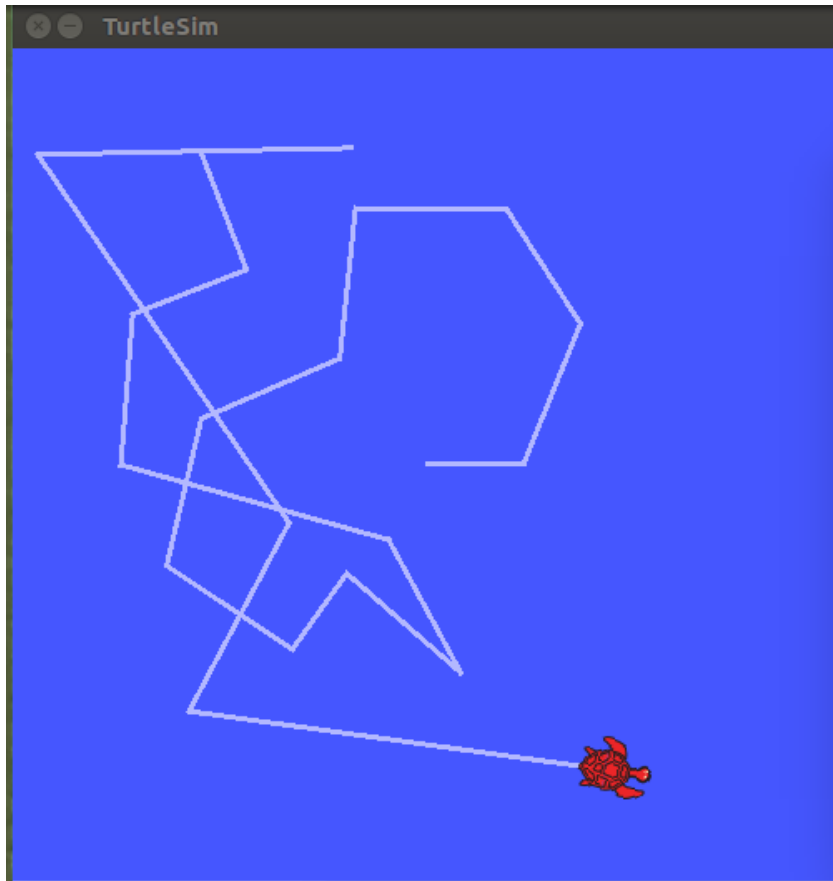
Registrazione tutti i topic

Se si vogliono registrare tutti i topic attivi, si può usare l'opzione '-a'.

```
$ rosbag record -a
```

Terminare la registrazione

[Ctrl-C] ci permette di terminare la registrazione della bag

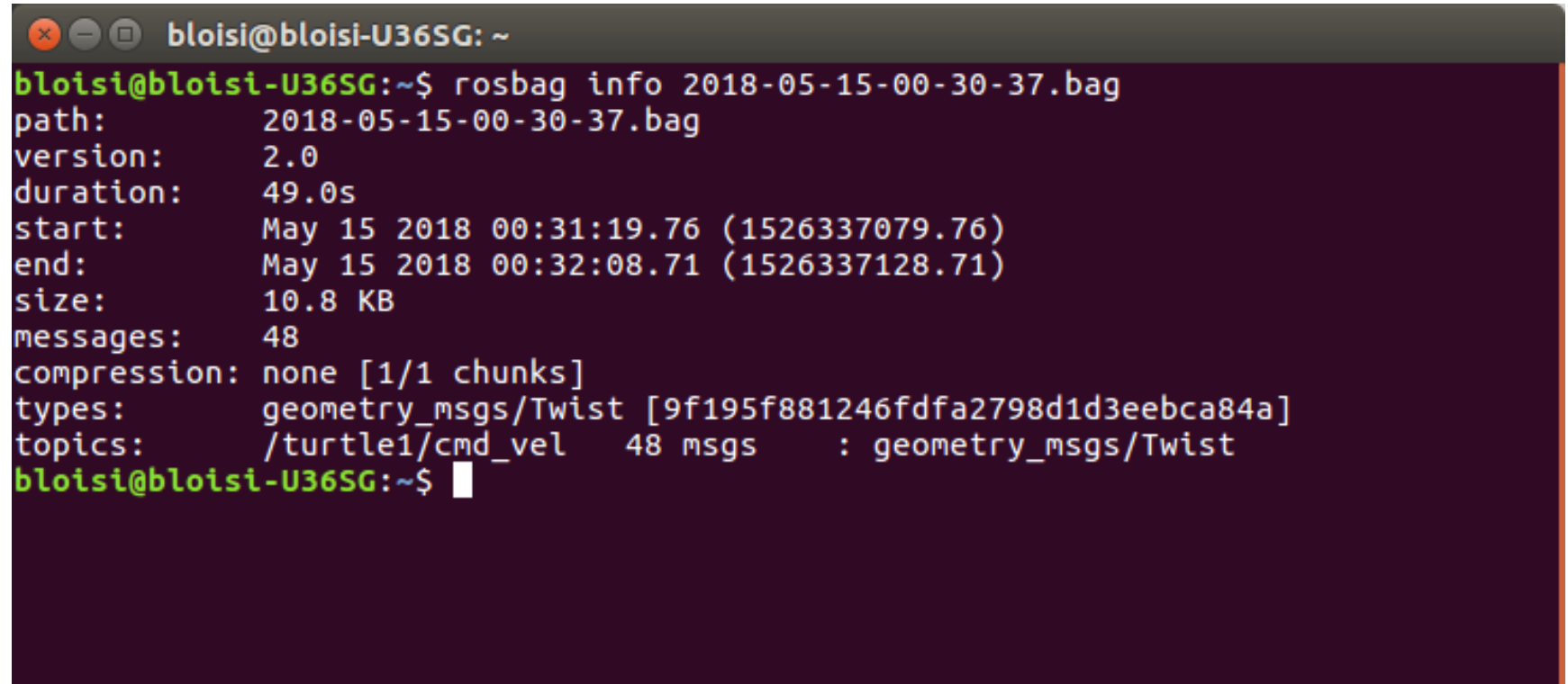


```
bloisi@bloisi-U36SG: ~  
bloisi@bloisi-U36SG:~$ rostopic list  
/rosout  
/rosout_agg  
/turtle1/cmd_vel  
/turtle1/color_sensor  
/turtle1/pose  
bloisi@bloisi-U36SG:~$ rosbag record /turtle1/cmd_vel  
[ INFO] [1526337037.406675213]: Subscribing to /turtle1/cmd_vel  
[ INFO] [1526337037.414123426]: Recording to 2018-05-15-00-30-37.bag.  
^Cbloisi@bloisi-U36SG:~$
```


rosvag info

Il comando info stampa a video informazioni sulla bag fornita come parametro. Nel nostro esempio, avremo informazioni sul file 2018-05-15-00-30-37.bag digitando

```
$ rosvag info 2018-05-15-00-30-37.bag
```



```
bloisi@bloisi-U36SG: ~  
bloisi@bloisi-U36SG:~$ rosvag info 2018-05-15-00-30-37.bag  
path:          2018-05-15-00-30-37.bag  
version:       2.0  
duration:      49.0s  
start:         May 15 2018 00:31:19.76 (1526337079.76)  
end:           May 15 2018 00:32:08.71 (1526337128.71)  
size:          10.8 KB  
messages:      48  
compression:  none [1/1 chunks]  
types:         geometry_msgs/Twist [9f195f881246fdfa2798d1d3eebca84a]  
topics:        /turtle1/cmd_vel  48 msgs      : geometry_msgs/Twist  
bloisi@bloisi-U36SG:~$
```

rosvag play

Ora che la bag è stata registrata, possiamo provare a riprodurla tramite

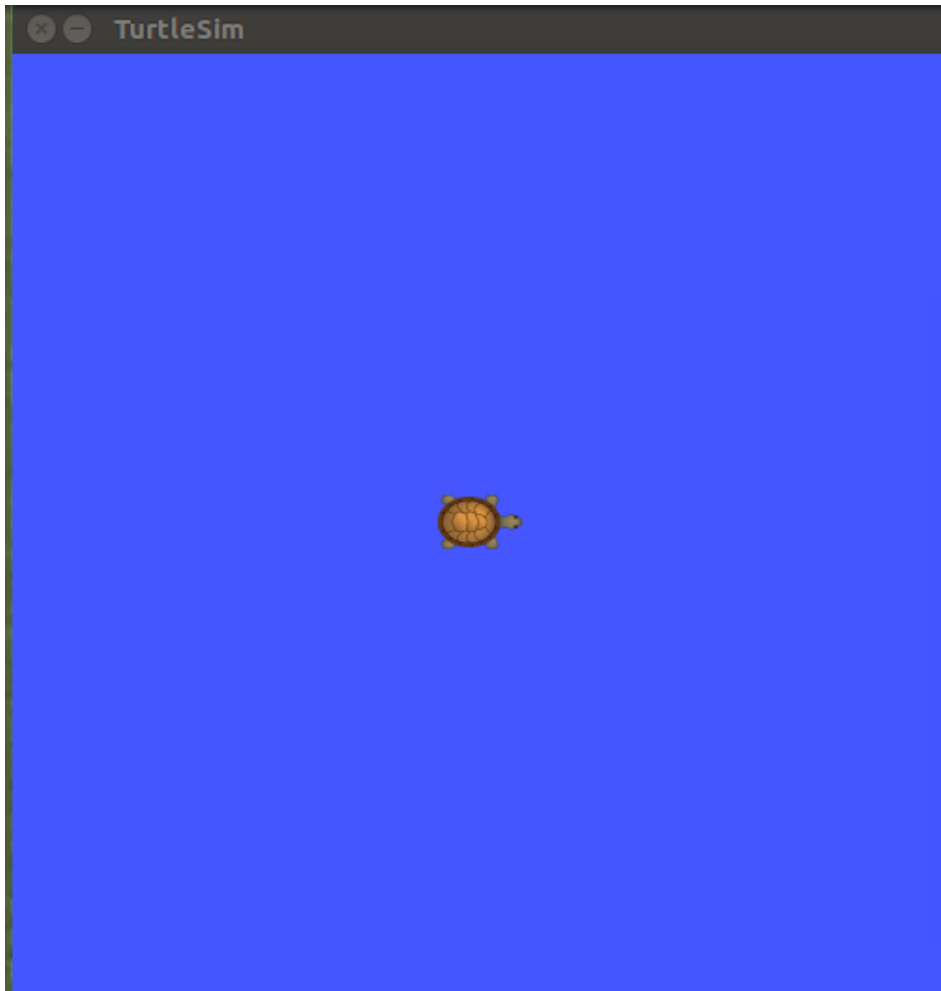
```
rosvag play <bagfile name>
```

Per esempio, per riprodurre la nostra bag

1. Terminiamo tutti i nodi attivi
2. Lanciamo il nodo turtlesim_node
3. Digitiamo

```
$ rosvag play 2018-05-15-00-30-37.bag
```

rosvim play – esecuzione



```
bloisi@bloisi-U36SG: ~  
bloisi@bloisi-U36SG:~$ rosvim turtlesim turtlesim_node  
[ INFO] [1526336917.499192727]: Starting turtlesim with node name /turtlesim  
[ INFO] [1526336917.504453947]: Spawning turtle [turtle1] at x=[5,544445], y=[5,  
544445], theta=[0,000000]  
^C  
bloisi@bloisi-U36SG:~$ rosvim turtlesim turtlesim_node  
[ INFO] [1526337414.146208633]: Starting turtlesim with node name /turtlesim  
[ INFO] [1526337414.151953802]: Spawning turtle [turtle1] at x=[5,544445], y=[5,  
544445], theta=[0,000000]  
□
```

```
bloisi@bloisi-U36SG: ~  
bloisi@bloisi-U36SG:~$ rosvim play 2018-05-15-00-30-37.bag
```

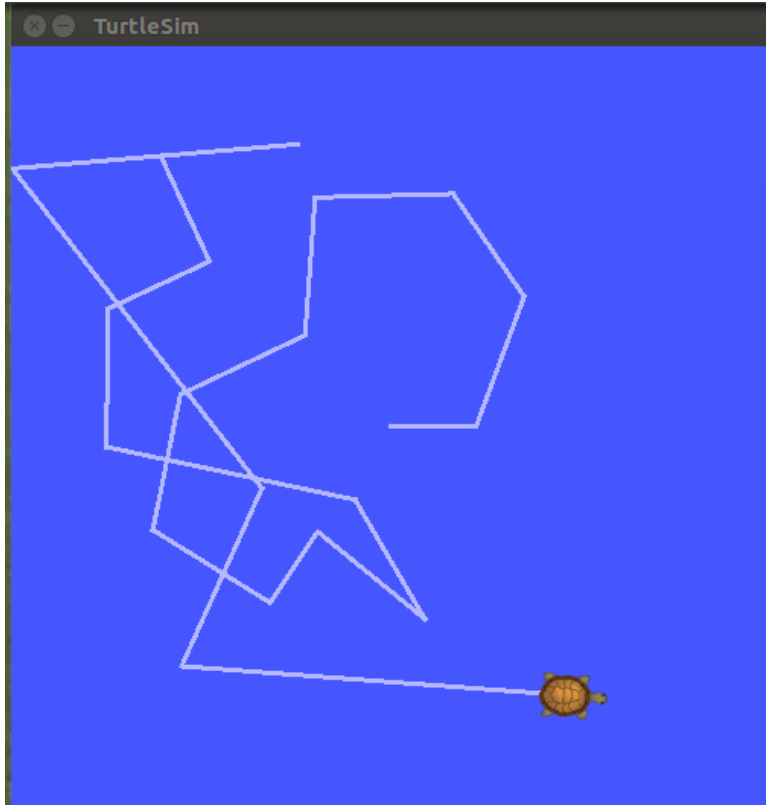
rosvbag play – esecuzione in corso

The image shows a desktop environment with a TurtleSim window on the left and a terminal window on the right. The TurtleSim window displays a blue field with a white path and a small turtle icon at the start of the path. The terminal window shows the execution of ROS commands and a table of bag play statistics.

```
bloisi@bloisi-U36SG: ~  
bloisi@bloisi-U36SG:~$ rosrn turtlesim turtlesim_node  
[ INFO] [1526336917.499192727]: Starting turtlesim with node name /turtlesim  
[ INFO] [1526336917.504453947]: Spawning turtle [turtle1] at x=[5,544445], y=[5,544445], theta=[0,000000]  
^C  
bloisi@bloisi-U36SG:~$ rosrn turtlesim turtlesim_node  
[ INFO] [1526337414.146208633]: Starting turtlesim with node name /turtlesim  
[ INFO] [1526337414.151953802]: Spawning turtle [turtle1] at x=[5,544445], y=[5,544445], theta=[0,000000]
```

[RUNNING]	Bag Time:	Duration:
[RUNNING]	1526337090.591895	10.836131 / 48.957648
[RUNNING]	1526337090.692036	10.936272 / 48.957648
[RUNNING]	1526337090.791511	11.035747 / 48.957648
[RUNNING]	1526337090.891664	11.135900 / 48.957648
[RUNNING]	1526337090.991832	11.236068 / 48.957648
[RUNNING]	1526337091.091962	11.336197 / 48.957648
[RUNNING]	1526337091.192129	11.436365 / 48.957648
[RUNNING]	1526337091.248360	11.492596 / 48.957648
[RUNNING]	1526337091.348569	11.592805 / 48.957648
[RUNNING]	1526337091.448749	11.692984 / 48.957648
[RUNNING]	1526337091.548924	11.793160 / 48.957648
[RUNNING]	1526337091.649109	11.893345 / 48.957648
[RUNNING]	1526337091.749240	11.993475 / 48.957648
[RUNNING]	1526337091.849477	12.093713 / 48.957648
[RUNNING]	1526337091.949632	12.193868 / 48.957648
[RUNNING]	1526337092.049816	12.294051 / 48.957648
[RUNNING]	1526337092.150008	12.394243 / 48.957648
[RUNNING]	1526337092.250157	12.494392 / 48.957648
[RUNNING]	1526337092.312010	12.556246 / 48.957648
[RUNNING]	1526337092.412182	12.656418 / 48.957648
[RUNNING]	1526337092.512428	12.756664 / 48.957648
[RUNNING]	1526337092.612576	12.856811 / 48.957648
[RUNNING]	1526337092.712744	12.956980 / 48.957648

rosv bag play – risultato finale

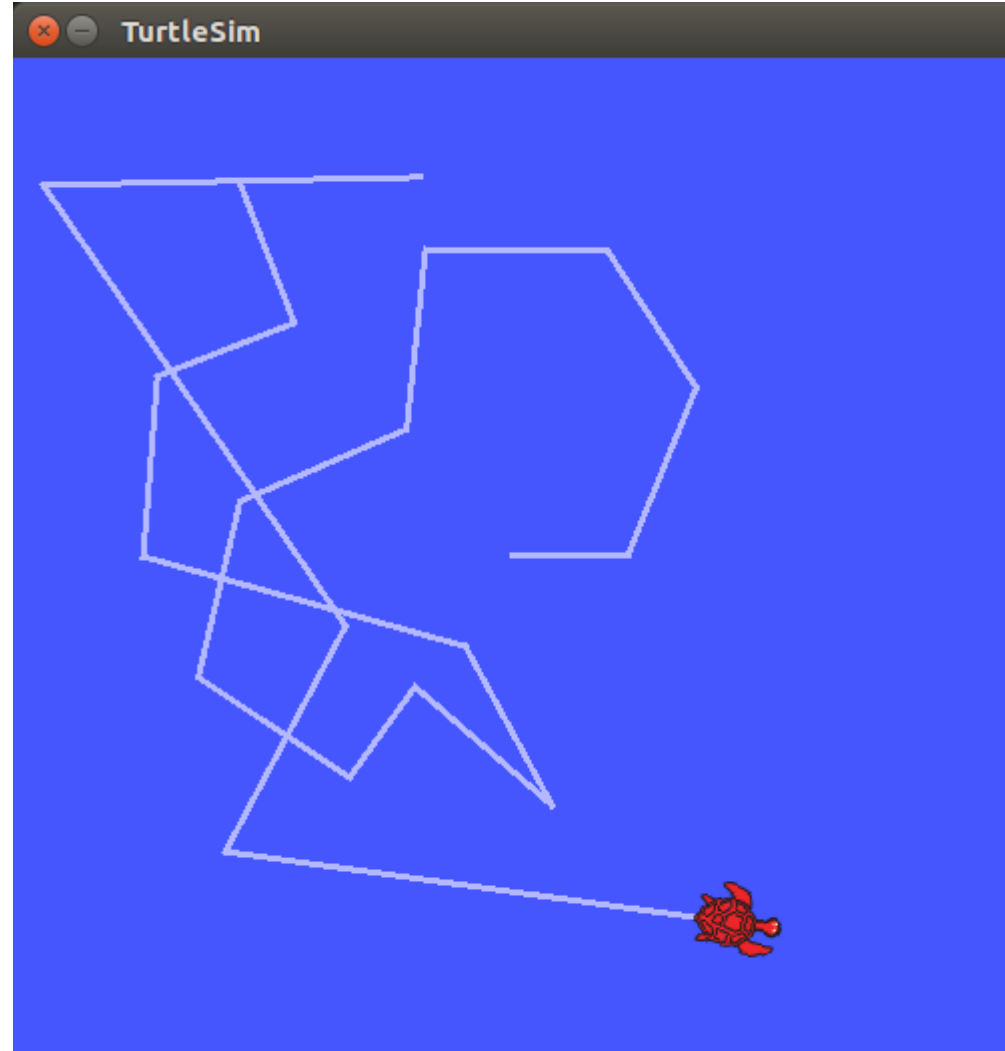


```
bloisi@bloisi-U36SG: ~  
bloisi@bloisi-U36SG:~$ rosv run turtlesim turtlesim_node  
[ INFO] [1526336917.499192727]: Starting turtlesim with node name /turtlesim  
[ INFO] [1526336917.504453947]: Spawning turtle [turtle1] at x=[5,544445], y=[5,544445], theta=[0,000000]  
^C  
bloisi@bloisi-U36SG:~$ rosv run turtlesim turtlesim_node  
[ INFO] [1526337414.146208633]: Starting turtlesim with node name /turtlesim  
[ INFO] [1526337414.151953802]: Spawning turtle [turtle1] at x=[5,544445], y=[5,544445], theta=[0,000000]  
□
```

```
bloisi@bloisi-U36SG: ~  
[RUNNING] Bag Time: 1526337126.557608 Duration: 46.801843 / 48.957648  
[RUNNING] Bag Time: 1526337126.657763 Duration: 46.901998 / 48.957648  
[RUNNING] Bag Time: 1526337126.757943 Duration: 47.002178 / 48.957648  
[RUNNING] Bag Time: 1526337126.858106 Duration: 47.102342 / 48.957648  
[RUNNING] Bag Time: 1526337126.958296 Duration: 47.202532 / 48.957648  
[RUNNING] Bag Time: 1526337127.058506 Duration: 47.302742 / 48.957648  
[RUNNING] Bag Time: 1526337127.158664 Duration: 47.402900 / 48.957648  
[RUNNING] Bag Time: 1526337127.258812 Duration: 47.503048 / 48.957648  
[RUNNING] Bag Time: 1526337127.358996 Duration: 47.603232 / 48.957648  
[RUNNING] Bag Time: 1526337127.459208 Duration: 47.703444 / 48.957648  
[RUNNING] Bag Time: 1526337127.559368 Duration: 47.803604 / 48.957648  
[RUNNING] Bag Time: 1526337127.659556 Duration: 47.903791 / 48.957648  
[RUNNING] Bag Time: 1526337127.759781 Duration: 48.004016 / 48.957648  
[RUNNING] Bag Time: 1526337127.857417 Duration: 48.101653 / 48.957648  
[RUNNING] Bag Time: 1526337127.957637 Duration: 48.201872 / 48.957648  
[RUNNING] Bag Time: 1526337128.057879 Duration: 48.302115 / 48.957648  
[RUNNING] Bag Time: 1526337128.158112 Duration: 48.402348 / 48.957648  
[RUNNING] Bag Time: 1526337128.258319 Duration: 48.502555 / 48.957648  
[RUNNING] Bag Time: 1526337128.358572 Duration: 48.602807 / 48.957648  
[RUNNING] Bag Time: 1526337128.458740 Duration: 48.702976 / 48.957648  
[RUNNING] Bag Time: 1526337128.558883 Duration: 48.803119 / 48.957648  
Done.  
bloisi@bloisi-U36SG:~$
```



roslaunch play – confronto



I bag file possono essere molto grandi

Un bag file registrato per un breve periodo di tempo comporta la creazione di file aventi dimensioni contenute

Se, invece, si ha bisogno di registrare messaggi per un lungo periodo di tempo, allora la dimensione del bag file può crescere fino ad occupare molta memoria

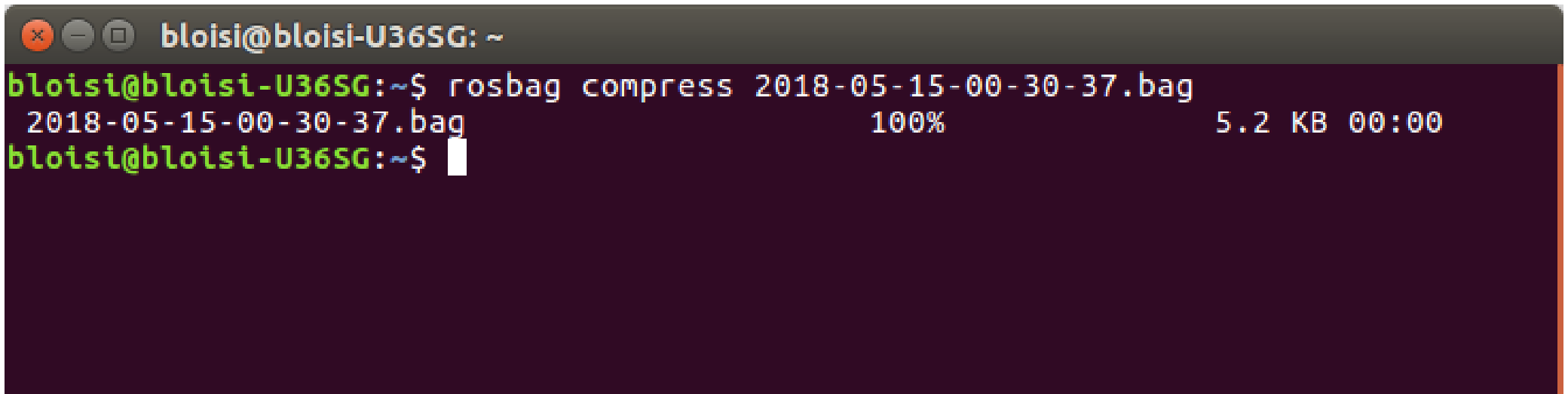
Si provi per esempio a scaricare la ROS bag a questo indirizzo

<http://www.dis.uniroma1.it/~bloisi/didattica/RobotProgramming/face.bag>

rosvag compress

ROS fornisce la possibilità di comprimere i bag file grazie all'opzione compress

```
$ rosvag compress 2018-05-15-00-30-37.bag
```



```
bloisi@bloisi-U36SG: ~  
bloisi@bloisi-U36SG:~$ rosvag compress 2018-05-15-00-30-37.bag  
2018-05-15-00-30-37.bag          100%          5.2 KB 00:00  
bloisi@bloisi-U36SG:~$
```


rosvag compress – esecuzione

```
bloisi@bloisi-U36SG: ~  
bloisi@bloisi-U36SG:~$ rosvag compress 2018-05-15-00-30-37.bag  
2018-05-15-00-30-37.bag          100%          5.2 KB 00:00  
bloisi@bloisi-U36SG:~$ ls  
2018-05-15-00-30-37.bag  
2018-05-15-00-30-37.orig.bag
```

rosvag decompress

Per riportare il bag file al suo formato originale, è possibile utilizzare decompress

```
$ rosvag decompress 2018-05-15-00-30-37.bag
```

Esercizio

Si vuole utilizzare un bag file per poter creare la mappa di un ambiente esplorato con un Turtlebot3 waffle

Utilizzeremo i seguenti componenti ROS

- **SLAM Package**
- **Rviz**
- **rosbag**

Virtual SLAM with saved bag file

Per completare l'esercizio utilizzeremo un Turtlebot3 waffle simulato e una rosbag già registrata

In particolare, del turtlebot3 utilizzeremo i dati provenienti dal sensore laser a 360°

Scarichiamo la bag dall'indirizzo

https://github.com/ROBOTIS-GIT/bags/blob/master/TB3_WAFFLE_SLAM.bag

roscore

Lanciamo

\$ roscore

```
roscore http://localhost:11311/
bloisi@bloisi-U36SG:~$ roscore
... logging to /home/bloisi/.ros/log/4d85da46-576b-11e8-9e4c-2709ac87ed01/roslau
nch-bloisi-U36SG-2511.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://localhost:38804/
ros_comm version 1.12.13

SUMMARY
=====

PARAMETERS
* /rostdistro: kinetic
* /rosversion: 1.12.13

NODES

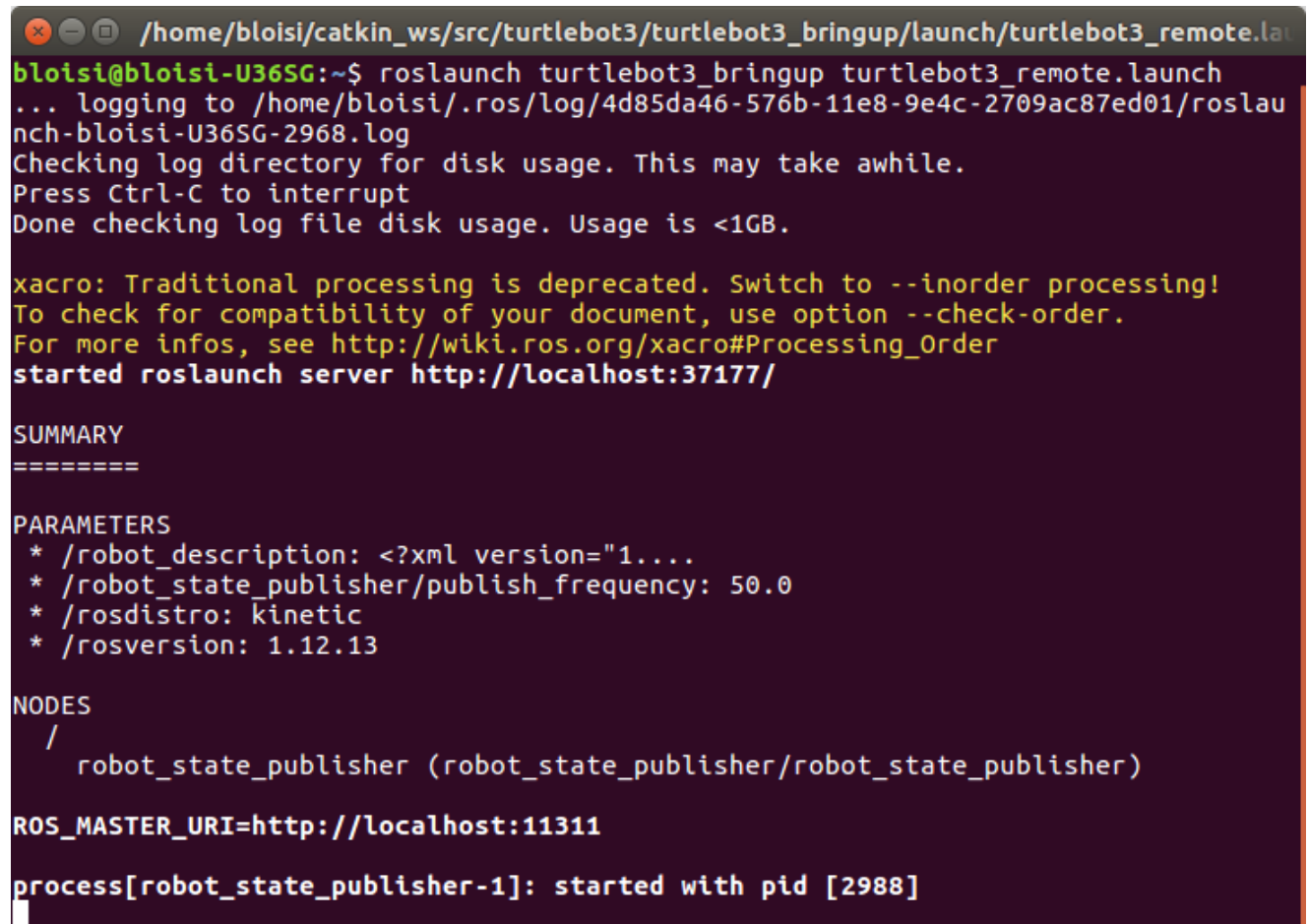
auto-starting new master
process[master]: started with pid [2523]
ROS_MASTER_URI=http://localhost:11311/

setting /run_id to 4d85da46-576b-11e8-9e4c-2709ac87ed01
process[rosout-1]: started with pid [2536]
started core service [/rosout]
█
```

turtlebot3_remote

```
$ export TURTLEBOT3_MODEL=waffle
```

```
$ roslaunch turtlebot3_bringup turtlebot3_remote.launch
```



```
/home/bloisi/catkin_ws/src/turtlebot3/turtlebot3_bringup/launch/turtlebot3_remote.la
bloisi@bloisi-U36SG:~$ roslaunch turtlebot3_bringup turtlebot3_remote.launch
... logging to /home/bloisi/.ros/log/4d85da46-576b-11e8-9e4c-2709ac87ed01/roslau
nch-bloisi-U36SG-2968.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

xacro: Traditional processing is deprecated. Switch to --inorder processing!
To check for compatibility of your document, use option --check-order.
For more infos, see http://wiki.ros.org/xacro#Processing_Order
started roslaunch server http://localhost:37177/

SUMMARY
=====

PARAMETERS
* /robot_description: <?xml version="1...
* /robot_state_publisher/publish_frequency: 50.0
* /roscpp_core: kinetic
* /roscpp_core: 1.12.13

NODES
/
  robot_state_publisher (robot_state_publisher/robot_state_publisher)

ROS_MASTER_URI=http://localhost:11311

process[robot_state_publisher-1]: started with pid [2988]
```

RViz

```
$ export TURTLEBOT3_MODEL=waffle
```

```
$ rosrun rviz rviz -d `rospack find turtlebot3_slam`/rviz/turtlebot3_slam.rviz
```

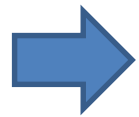
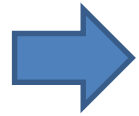
A terminal window with a dark purple background and a grey title bar. The title bar contains the text "bloisi@bloisi-U36SG: ~" and standard window control icons. The terminal shows two lines of command execution. The first line is "bloisi@bloisi-U36SG:~\$ export TURTLEBOT3_MODEL=waffle" and the second line is "bloisi@bloisi-U36SG:~\$ rosrun rviz rviz -d `rospack find turtlebot3_slam`/rviz/turtlebot3_slam.rviz". A white cursor is visible at the end of the second line.

```
bloisi@bloisi-U36SG: ~
bloisi@bloisi-U36SG:~$ export TURTLEBOT3_MODEL=waffle
bloisi@bloisi-U36SG:~$ rosrun rviz rviz -d `rospack find turtlebot3_slam`/rviz/t
urtlebot3_slam.rviz
```

RViz

```
$ export TURTLEBOT3_MODEL=waffle
```

```
$ rosrun rviz rviz -d `rospack find turtlebot3_slam`/rviz/turtlebot3_slam.rviz
```



The screenshot shows the RViz interface for a TurtleBot3 Waffle robot. The main window displays a 2D grid with a white robot icon in the center. The left sidebar contains the 'Displays' panel with the following settings:

- Global Options**
 - Fixed Frame: map
 - Background Color: 48; 48; 48
 - Frame Rate: 30
 - Default Light:
- Global Status: Error**
 - Fixed Frame: Fixed Frame [map] do...
- Grid**
 - Status: Ok
 - Reference Frame: map
 - Plane Cell Count: 100
 - Normal Cell Count: 0
 - Cell Size: 0,45
 - Line Style: Lines
 - Color: 160; 160; 164
 - Alpha: 0,5
 - Plane: XY
 - Offset: 0; 0; 0
- TF**
 -
- LaserScan**
 -
- Map**
 - Status: Warn
 - Topic: /map
 - Alpha: 0,7

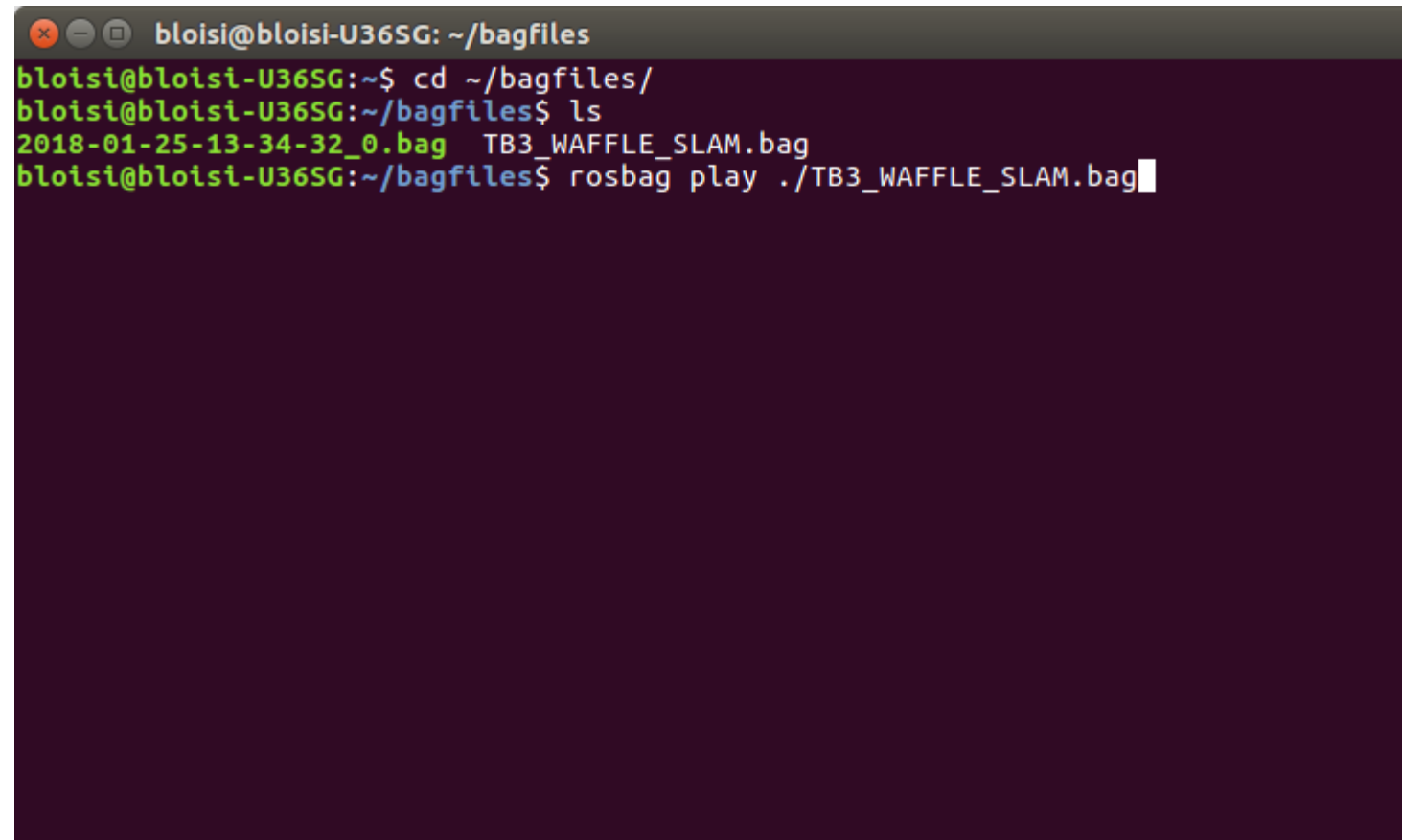
At the bottom, the 'Time' panel shows the following values:

- ROS Time: 1526298123.63
- ROS Elapsed: 15.82
- Wall Time: 1526298123.66
- Wall Elapsed: 15.73
- Experimental:
- Reset:
- Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click: Zoom. Shift: More options.
- FPS: 31

rosvag play

Spostiamoci nella directory dove è stata salvata la bag e digitiamo

```
$ rosvag play ./TB3_WAFFLE_SLAM.bag
```

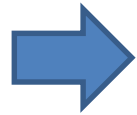


```
bloisi@bloisi-U36SG: ~/bagfiles
bloisi@bloisi-U36SG:~$ cd ~/bagfiles/
bloisi@bloisi-U36SG:~/bagfiles$ ls
2018-01-25-13-34-32_0.bag  TB3_WAFFLE_SLAM.bag
bloisi@bloisi-U36SG:~/bagfiles$ rosvag play ./TB3_WAFFLE_SLAM.bag
```

bag playing in terminal

```
bloisi@bloisi-U36SG: ~/bagfiles
[RUNNING] Bag Time: 1491540670.700201 Duration: 7.760979 / 360.058742
[RUNNING] Bag Time: 1491540670.702572 Duration: 7.763350 / 360.058742
[RUNNING] Bag Time: 1491540670.703450 Duration: 7.764228 / 360.058742
[RUNNING] Bag Time: 1491540670.704459 Duration: 7.765238 / 360.058742
[RUNNING] Bag Time: 1491540670.713223 Duration: 7.774002 / 360.058742
[RUNNING] Bag Time: 1491540670.713934 Duration: 7.774713 / 360.058742
[RUNNING] Bag Time: 1491540670.713956 Duration: 7.774735 / 360.058742
[RUNNING] Bag Time: 1491540670.715845 Duration: 7.776624 / 360.058742
[RUNNING] Bag Time: 1491540670.716441 Duration: 7.777220 / 360.058742
[RUNNING] Bag Time: 1491540670.721837 Duration: 7.782616 / 360.058742
[RUNNING] Bag Time: 1491540670.722950 Duration: 7.783729 / 360.058742
[RUNNING] Bag Time: 1491540670.741850 Duration: 7.802629 / 360.058742
[RUNNING] Bag Time: 1491540670.743046 Duration: 7.803824 / 360.058742
[RUNNING] Bag Time: 1491540670.743095 Duration: 7.803873 / 360.058742
[RUNNING] Bag Time: 1491540670.744679 Duration: 7.805458 / 360.058742
[RUNNING] Bag Time: 1491540670.745658 Duration: 7.806437 / 360.058742
[RUNNING] Bag Time: 1491540670.751776 Duration: 7.812555 / 360.058742
[RUNNING] Bag Time: 1491540670.755691 Duration: 7.816470 / 360.058742
[RUNNING] Bag Time: 1491540670.756861 Duration: 7.817639 / 360.058742
[RUNNING] Bag Time: 1491540670.757389 Duration: 7.818168 / 360.058742
[RUNNING] Bag Time: 1491540670.759160 Duration: 7.819939 / 360.058742
[RUNNING] Bag Time: 1491540670.759865 Duration: 7.820644 / 360.058742
[RUNNING] Bag Time: 1491540670.762386 Duration: 7.823164 / 360.058742
```

bag playing in RViz



The screenshot shows the RViz interface for a robot simulation. The main window displays a 2D grid with a robot model in the center. The robot is a small grey rectangle with a black circle representing the camera. The robot is positioned on a grey path that leads to a red dashed line, indicating a goal or a specific location. The background is a dark grey grid.

The left sidebar contains the following settings:

- Displays**
 - Global Options
 - Fixed Frame: map
 - Background Color: 48; 48; 48
 - Frame Rate: 30
 - Default Light:
 - Global Status: Ok
 - Fixed Frame: OK
 - Grid
 - Status: Ok
 - Reference Frame: map
 - Plane Cell Count: 100
 - Normal Cell Count: 0
 - Cell Size: 0,45
 - Line Style: Lines
 - Color: 160; 160; 164
 - Alpha: 0,5
 - Plane: XY
 - Offset: 0; 0; 0
 - TF:
 - LaserScan:
 - Map
 - Status: Ok
 - Topic: /map
 - Alpha: 0,7

Buttons at the bottom of the sidebar: Add, Duplicate, Remove, Rename.

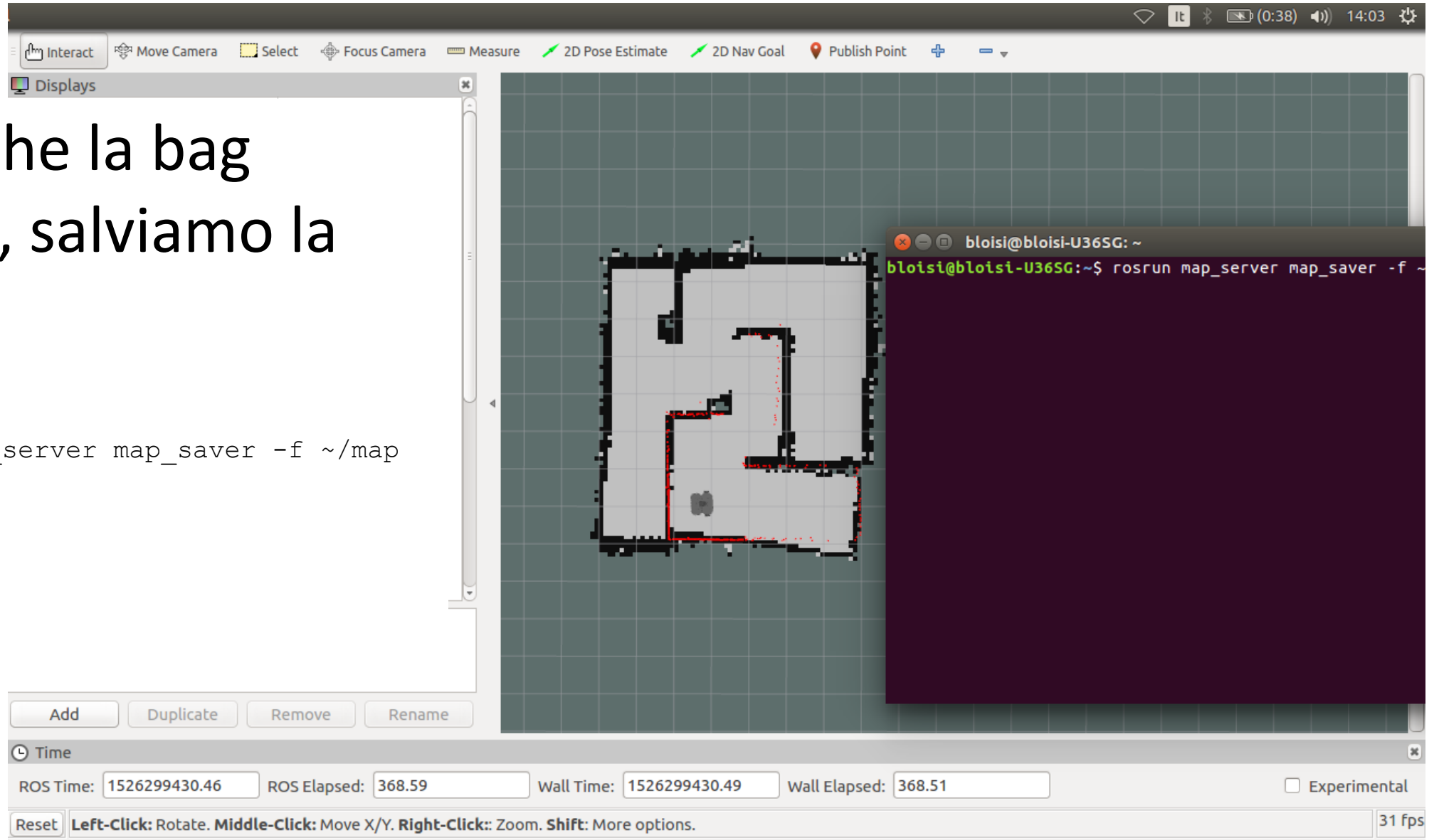
The bottom status bar shows the following information:

- Time
 - ROS Time: 1526298338.14
 - ROS Elapsed: 230.34
 - Wall Time: 1526298338.17
 - Wall Elapsed: 230.27
- Experimental:
- Reset
- Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click: Zoom. Shift: More options.
- 31 fps

Salviamo la mappa

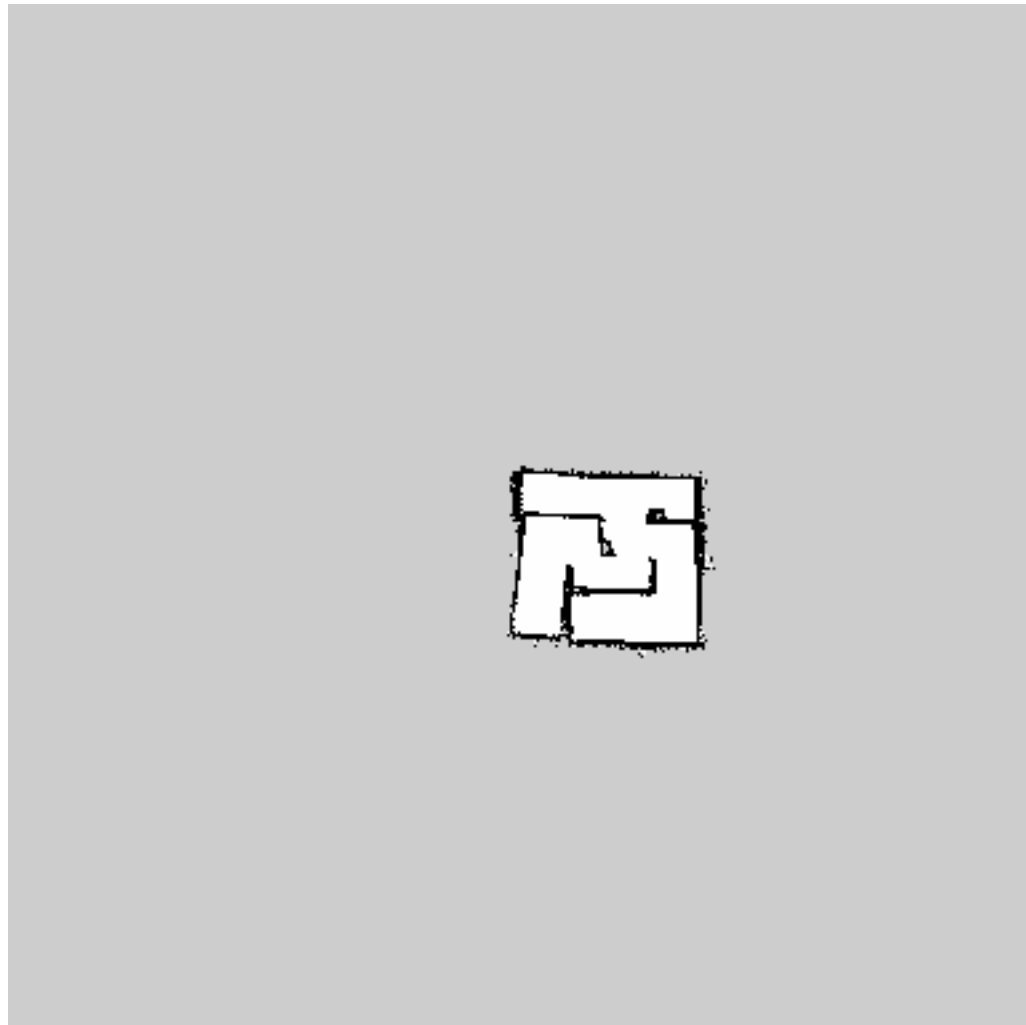
Prima che la bag
termini, salviamo la
mappa

```
$ rosrun map_server map_saver -f ~/map
```



The screenshot shows a ROS2 GUI window with a toolbar at the top containing icons for Interact, Move Camera, Select, Focus Camera, Measure, 2D Pose Estimate, 2D Nav Goal, and Publish Point. A 'Displays' panel on the left shows a 2D occupancy grid map with a robot's current position and path. A terminal window is overlaid on the map, displaying the command `roslaunch map_server map_saver -f ~/map` and its output. At the bottom, a 'Time' panel shows ROS Time: 1526299430.46, ROS Elapsed: 368.59, Wall Time: 1526299430.49, and Wall Elapsed: 368.51. A 'Reset' button and a legend for mouse actions (Left-Click: Rotate, Middle-Click: Move X/Y, Right-Click: Zoom, Shift: More options) are also visible. The bottom right corner shows '31 fps'.

la nostra mappa



Esercizio

Si salvi in memoria un bag file contenente i dati registrati usando un Turtlebot3 waffle simulato nel mondo Turtlebot3 Home

Usare poi la bag per creare una mappa dello scenario Turtlebot3 House

Quali topic è necessario registrare?



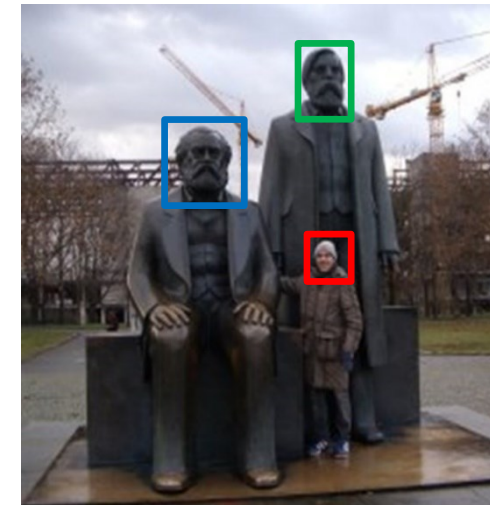
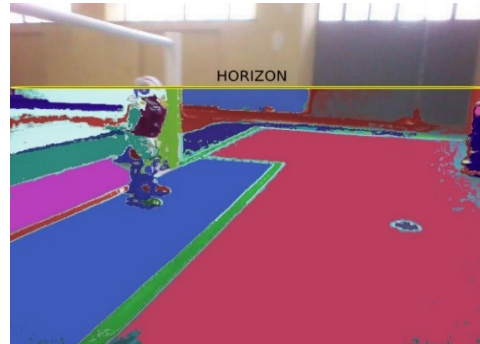
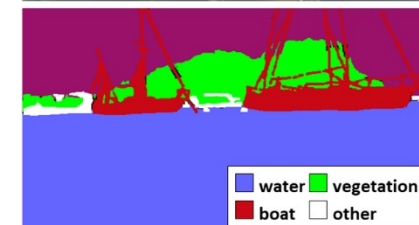
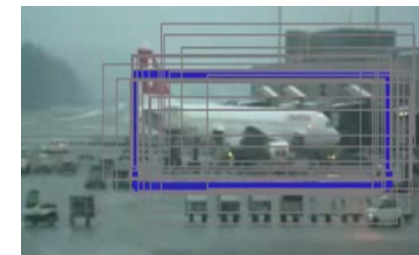
UNIVERSITÀ
di **VERONA**

Dipartimento
di **INFORMATICA**

*Corso di Laboratorio Ciberfisico
Modulo di Robot Programming with ROS*

Esercitazione rosbag

Docente:
Domenico Daniele
Bloisi



Maggio 2018