Data Logging and Playback in ROS

ECE 495/595 Lecture Slides

Winter 2017

Instructor: Micho Radovnikovich

Summary and Quick Links

These slides contain the following concepts:

Bag Files

- Data is logged and replayed in ROS using *bag files*.
- Messages being published on any topic in the ROS system can be saved to a bag file.
- When replaying, the bag player node will publish the exact same messages that were recorded, in the same time sequence.
- This helps software development, since it is possible to test an algorithm on recorded data while maintaining the same conditions of the real-time system.
- Full documentation of terminal command interface at http://wiki.ros.org/rosbag/Commandline

Recording a Bag File

- Recording a bag file is done on the Linux command line with the command *rosbag record*.

- To specify particular topics to record, simply list them after the command:

```
rosbag record /topic1 /topic2 /topic3
```

- If you want to record every topic currently being published, use the *-a* option instead of listing particular topics:

```
rosbag record -a
```

Recording a Bag File

- By default, the name of the recorded bag file will be a time stamp reflecting the system time when the recording was started.

```
rosbag record /topic1 /topic2 /topic3
```

- This would create a bag file with just a time stamp for a name:

```
2016-02-24-20-01-45.bag
```

Recording a Bag File

- If the *-o* (lowercase) option is used, the file can be saved with a prefix in front of the time stamp.

```
rosbag record /topic1 /topic2 /topic3 -o abc
```

- The above command would save the recorded bag file as

```
abc_2016-02-24-20-01-45.bag
```

Recording a Bag File

- The time stamp can be completely suppressed with the *-O* (capital) option.

```
rosbag record /topic1 /topic2 /topic3 -O abc
```

- The above command would save a bag file called

```
abc.bag
```

Playing Back a Bag File

- Recorded bag files can be replayed using the *rosbag play* command.

- Simply input the name of the bag file to play it back:

```
rosbag play abc.bag
```

- If the bag file is in a different directory than the terminal, a relative or absolute file path can be input as well.

Playing Back a Bag File

- There are a lot of useful options that affect the behavior of the bag playback.
  - **-s** – Start the bag somewhere in the middle. The argument is the number of seconds from the start of the bag.
  - **-u** – Stop the bag somewhere in the middle. The argument is the number of seconds from the start of the bag.
  - **-r** – Speed up or slow down the playback. The argument is the time scaling factor.
  - **-l** – Loop the bag playback indefinitely.

- At any time, pressing space bar will pause or un-pause bag playback.

Playing Back a Bag File

- Start a bag 10 seconds into the recording

```
rosbag play abc.bag -s 10
```

- Play only the first 100 seconds of a bag

```
rosbag play abc.bag -u 100
```

- Play a bag recording at 3x speed, and loop it indefinitely

```
rosbag play abc.bag -r 3 -l
```

```
rosbag play abc.bag -lr 3
```

Playing Back a Bag File

- A *ros::Time* can be published from a bag file, which will simulate the actual time from when the data was recorded.
- This is done using the *--clock* option.

```
rosbag play abc.bag --clock
```

- This is primarily used when code compares the time stamps in message headers to the current ROS time, which is by default set to whatever the Linux system time is.

Playing Back a Bag File

- In order to switch ROS to using a simulated time instead of system time, the *use_sim_time* parameter must be set to True before running any nodes.

```
rosparam set use_sim_time true
```

- This parameter can also be set in a launch file

```
<param name="use_sim_time" value="true" />
```

- Different launch files can be set up to run the system in offline bag playback mode, or in real-time mode by simply setting this parameter to True or False.

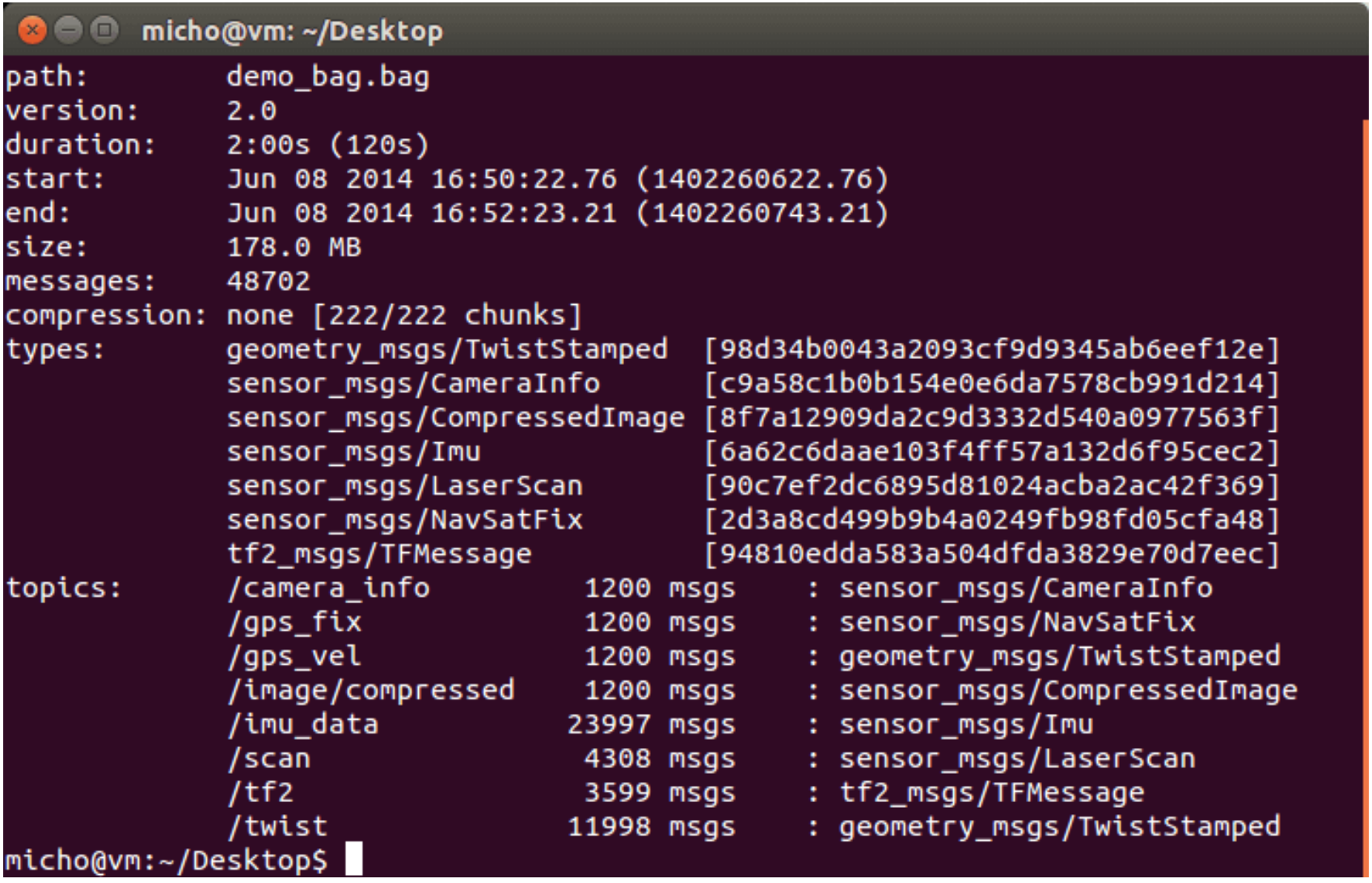
Bag File Introspection

- One way to see which messages are contained in a bag file is to simply play the file and use either *rqt_graph* or *rostopic info* to see the topics.

- However, more details are available about the bag if the *rosbag info* command is used.

```
rosbag info abc.bag
```

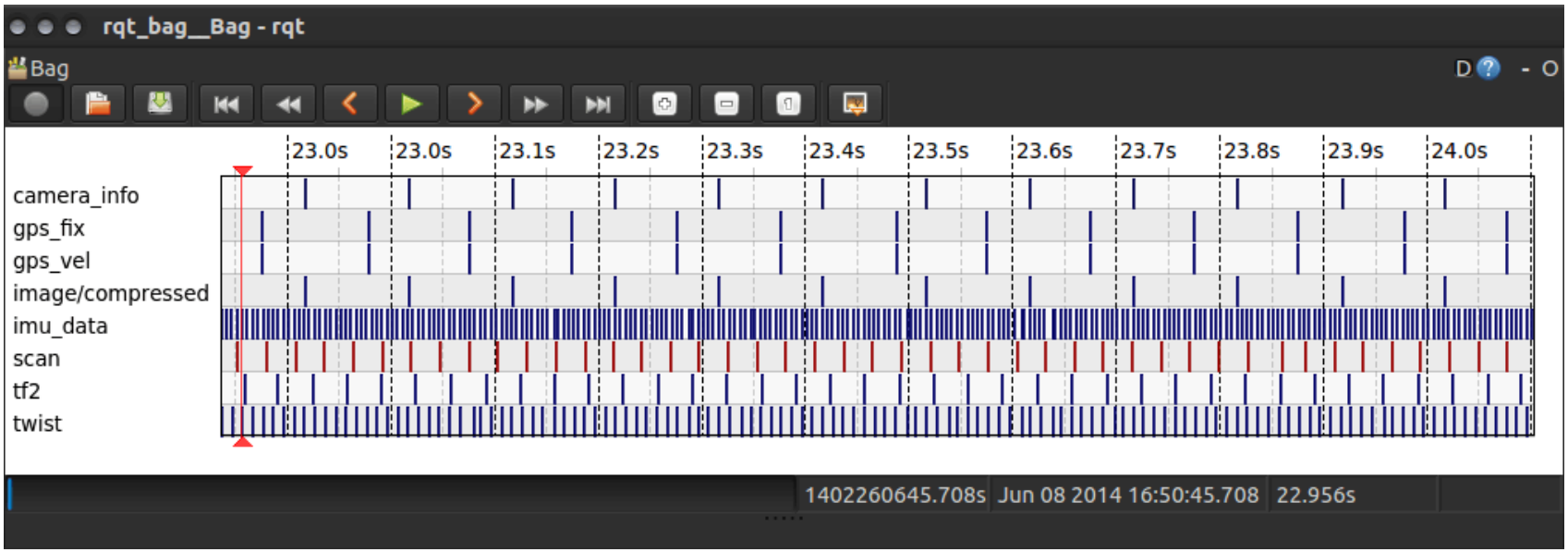
Bag File Introspection



Bag File Introspection

- Another tool to help manage bag files is the *rqt_bag* tool.

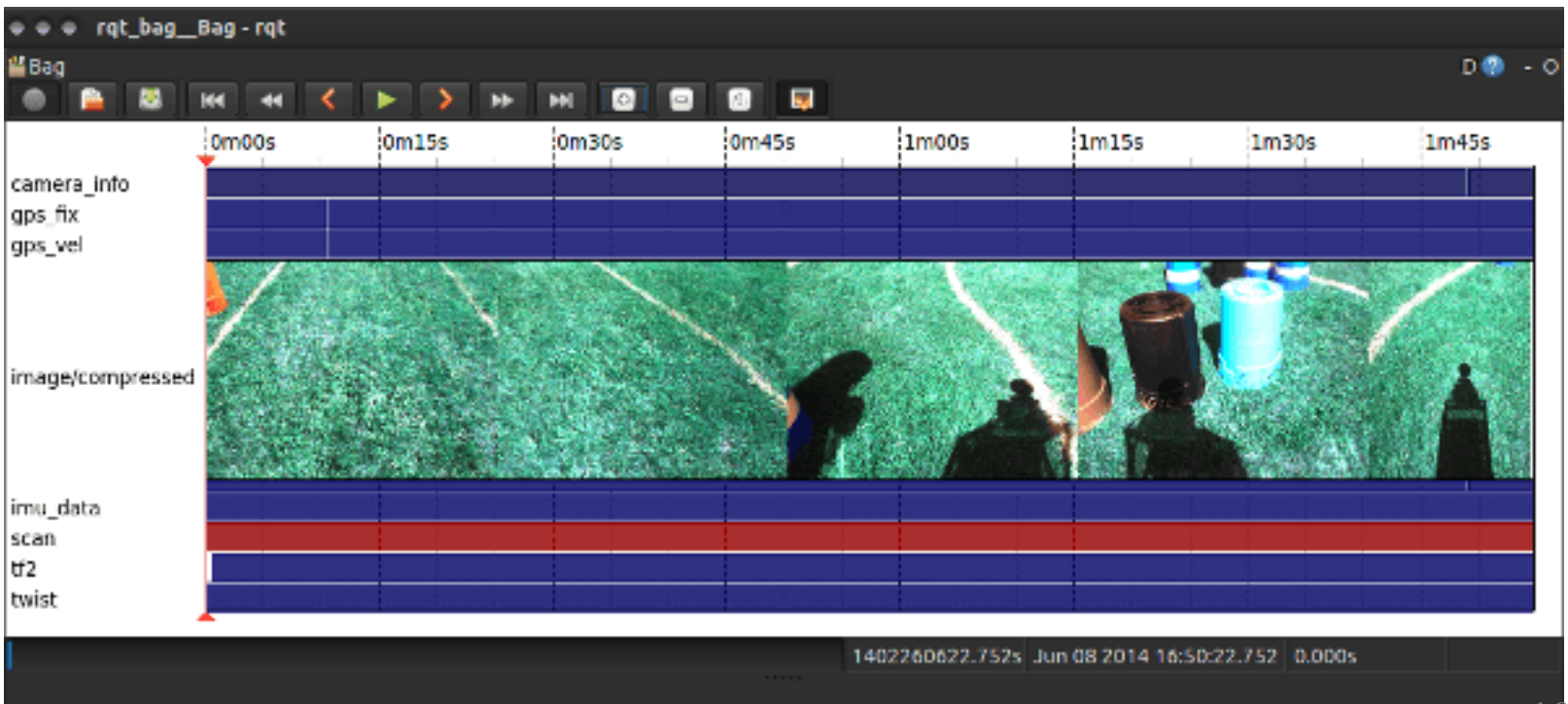
- The GUI visualizes received messages as blue tick marks, which are useful for intuitively verifying message frequency, as well as message dropouts.



Bag File Introspection

- If the bag file contains image topics, they can be visualized using thumbnails.

- The colors might look strange. This is because some image topics are formatted using a different colorspace, and *rqt_bag* assumes it is in RGB8 format.



Bag File Introspection

- *rqt_bag* allows publishing a subset of messages, as well as plotting and display of raw numerical values.
- The biggest drawback to rqt_bag is that it can be slow at loading bag files, especially if they are large.
- In order to actually publish the messages contained in the bag, rqt_bag must be configured to do so, as it won't publish the data by default.
- Therefore, rqt_bag is typically used if the extra features are required, but if it is desired to just play the data back for other code to use it, then just use rosbag play.