



链滴

[ROS]using rosbag and rospaly

作者: [lixiang0](#)

原文链接: <https://ld246.com/article/1472523050481>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<p>####1.记录数据（创建 bag 文件）</p>

<p>本教程将指导如何记录正在 ROS 系统中运行的主题的数据。主题的数据将保存在 bag 文件中。

首先，执行以下命令：</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">roscore
</span></span><span class="highlight-line"><span class="highlight-cl">roslaunch turtlesim t
rtlesim_node
</span></span><span class="highlight-line"><span class="highlight-cl">roslaunch turtlesim t
rtle_teleop_key
</span></span></code></pre>
```

<p>上述命令将启动 2 个节点，turtlesim 仿真窗口和键盘方向控制窗口。如果你选择启动 turtle_ke
board 的窗口，可以看到下面的内容：</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">Reading from keyboard
</span></span><span class="highlight-line"><span class="highlight-cl">-----
-----
</span></span><span class="highlight-line"><span class="highlight-cl">Use arrow keys to
move the turtle.
</span></span></code></pre>
```

<p>在 turtle_keyboard 窗口按下方向键将看到小乌龟在移动。注意要选择键盘控制的窗口。

####1.1 记录所有的主题</p>

<p>首先检查下当前系统已经发布的主题有那些。执行：</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">rostopic list -v
</span></span></code></pre>
```

<p>可以看到下面的内容：</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">Published topics:
</span></span><span class="highlight-line"><span class="highlight-cl">* /turtle1/color_s
nsor [turtlesim/Color] 1 publisher
</span></span><span class="highlight-line"><span class="highlight-cl">* /turtle1/cmd_vel
[geometry_msgs/Twist] 1 publisher
</span></span><span class="highlight-line"><span class="highlight-cl">* /rosout [rosgra
h_msgs/Log] 2 publishers
</span></span><span class="highlight-line"><span class="highlight-cl">* /rosout_agg [ro
graph_msgs/Log] 1 publisher
</span></span><span class="highlight-line"><span class="highlight-cl">* /turtle1/pose [t
rtlesim/Pose] 1 publisher
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">Subscribed topics:
</span></span><span class="highlight-line"><span class="highlight-cl">* /turtle1/cmd_vel
[geometry_msgs/Twist] 1 subscriber
</span></span><span class="highlight-line"><span class="highlight-cl">* /rosout [rosgra
h_msgs/Log] 1 subscriber
</span></span></code></pre>
```

<p>列出来的已发布的主题是仅有的潜在的能被记录到数据日志文件的数据类型，同时发布的数据也
记录。teleop_turtle 发布的/turtle1/cmd_vel 命令数据是作为 turtlesim 进程的输入。/turtle1/color
sensor 和/turtle1/pose 是 turtlesim 输出的数据。

现在，我们将记录发布的数据。打开一个新的命令行窗口。执行：</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">mkdir ~/bagfiles
</span></span><span class="highlight-line"><span class="highlight-cl">cd ~/bagfiles
</span></span><span class="highlight-line"><span class="highlight-cl">roslaunch turtlesim t
rtlesim_node
</span></span><span class="highlight-line"><span class="highlight-cl">roslaunch turtlesim t
rtle_teleop_key
</span></span></code></pre>
```

```
</span></span></code></pre>
```

<p>这里我们这是创建了一个临时的目录用来保存数据，然后运行 `rosvag record -a`，-a 表示保存有的已发布的主题的数据到一个 bag 文件里。</p>

<p>返回 `turtle_teleop` 窗口，移动小乌龟 10S 左右。</p>

<p>在运行 `rosvag` 的窗口按下 `ctrl+c`。现在，检查下 `~/bagfiles` 目录下的内容。你可以看到一个年、数据、时间、后缀为 `.bag` 为名字的文件。这个文件保存了 `rosvag record` 执行期间所有已任意点发布主题的数据。</p>

<p>####2.检验和播放 bag 文件</p>

<p>前文通过 `rosvag record` 命令已经将数据保存到了 bag 文件，现在我们通过 `rosvag info` 和 `rosvag play` 文件来检查和回放数据。首先查看 bag 文件保存了什么。可以使用 `rosvag info` 来查看，这命令将检查 bag 文件的内容而不用回放。在 bag 文件的目录下执行：</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">rosvag info &lt;bag文件名&gt;
```

```
</span></span></code></pre>
```

<p>可以看到类似的信息：</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">path: 2014-12-10-20-08-34.bag
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">version: 2.0
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">duration: 1:38s (8s)
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">start: Dec 10 2014 20:08:35.83 (1418270915.83)
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">end: Dec 10 2014 20:10:14.38 (1418271014.38)
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">size: 865.0 KB
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">messages: 1247
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">compression: non [1/1 chunks]
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">types: geometry_msgs/Twist [9f195f881246fdfa2798d1d3eebca84a]
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">rosgraph
```

```
msgs/Log [acffd30cd6b6de30f120938c17c593fb]
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">turtlesim/olor [353891e354491c51aabe32df673fb446]
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">turtlesim/ose [863b248d5016ca62ea2e895ae5265cf9]
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">topics: /rosout 4 msgs : rosgraph_msgs/Log (2 connections)
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">/turtle1/md_vel 169 msgs : geometry_msgs/Twist
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">/turtle1/color_sensor 6149 msgs : turtlesim/Color
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">/turtle1/pse 6149 msgs : turtlesim/Pose
```

```
</span></span></code></pre>
```

<p>上面显示了主题的名字，类型，以及包含在 bag 文件中的每一种消息的数目。我们可以看到在 `rosvag` 输出里广播的主题，五分之四的主题在我们记录间隔里实际发布。通过 `rosvag record -a` 命令记录所有节点发布的所有消息。</p>

<p>接下来的教程将回放 bag 文件，重现系统运行时的行为。首先你停止 `turtle_teleop` 窗口，保持 `turtlesim` 窗口运行。在命令行窗口，在 bag 文件的目录下运行：</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">rosvag play &lt;your bagfile&gt;
```

```
</span></span></code></pre>
```

<p>可以看到类似: </p>

```
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl"> [ INFO] [1418271315.162885976]: Opening 2014-12-10-20-08-34.bag
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> Waiting 0.2 seconds after advertising topics... done.
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> Hit space to toggle paused, or 's' to step.
</span> </span> </code> </pre>
```

<p>rosviz play 的默认模式下将等待 0.2 秒之后才开始播放 bag 文件的内容。等待的这段时间许消息的任意订阅者提示接下来将发布的消息以及接下来的具体的消息。如果 rosviz play 立马发布消息, 订阅将收不到一开始发布的消息。具体要等待多久可以通过 -d 选项来设置。
最终, 回放的效果跟实际通过鼠标控制的效果是一样的。也可以通过 -s 选项来选择开始播放的时间点 -r 选项是选择播放的速度: </p>

```
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl"> rosviz play -r 2 &lt;your bagfile&gt;
</span> </span> </code> </pre>
```

<p>可以看到运行的效果跟之前的不一样, 这是因为回放是以 2 倍的速度去点击键盘的。 </p>

<p>####3.保存数据的子集</p>

<p>当运行复杂系统的时候, 比如 pr2 软件集, 将有成百的主题发布。对于一些主题, 比如摄像头图流, 有可能发布大量的数据。这样的系统写入所有主题的日志文件到一个单独的 bag 文件是不可行的。 rosviz record 命令支持部分主题的日志输出, 仅仅只记录用户关注的主题。
执行: </p>

```
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl"> rosviz run turtlesim turtlesim_node
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> rosviz run turtlesim turtlesim_node
</span> </span> </code> </pre>
```

<p>在 bag 文件的目录下执行: </p>

```
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl"> rosviz record -O subset /turtle1/cmd_vel /turtle1/pose
</span> </span> </code> </pre>
```

<p>-O 选项指定保存日志到 subset.bag 文件, topic 参数指定了需要记录的两个主题。通过键盘的向键移动小乌龟, 然后按下 Ctrl+c 终止记录。 </p>

<p>现在通过使用 rosviz info subset.bag 来检查 bag 文件的内容。可以看到类似的: </p>

```
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl"> path: subset.bag
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> version: 2.0
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> duration: 12.6s
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> start: Dec 10 2014 20:20:49.45 (1418271649.45)
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> end: Dec 10 2014 20:21:02.07 (1418271662.07)
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> size: 68.3 KB
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> messages: 813
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> compression: none [1/1 chunks]
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> types: geometry_msgs/Twist [9f195f881246fdfa2798d1d3eebca84a]
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> turtlesim/pose [863b248d5016ca62ea2e895ae5265cf9]
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> topics: /turtle1/cmd_vel 23 msgs : geometry_msgs/Twist
</span> </span> </code> </pre>
```

```
</span></span><span class="highlight-line"><span class="highlight-cl"> /turtle1/p  
se 790 msgs : turtlesim/Pose  
</span></span></code></pre>  
<p>####4.rosbag record/play 的限制</p>  
<p>在之前的教程中可以看到小乌龟的运行路径不是很准确的跟着原始的键盘输入运行，大概的形状能看起来一样，但是小乌龟的运行的不是准确的路径。造成这样的原因是：小乌龟的运行路径对系统间小的改变敏感，rosbag 不能很精确的记录系统运行时的行为，比如消息被 roscord 记录和处理，被 rospaly 产生和处理。对于像 turtlesim 一样的节点，当命令消息处理的时候，监视器的时间改变会妙的改变行为，所以不要期望会精确的复现行为。</p>
```