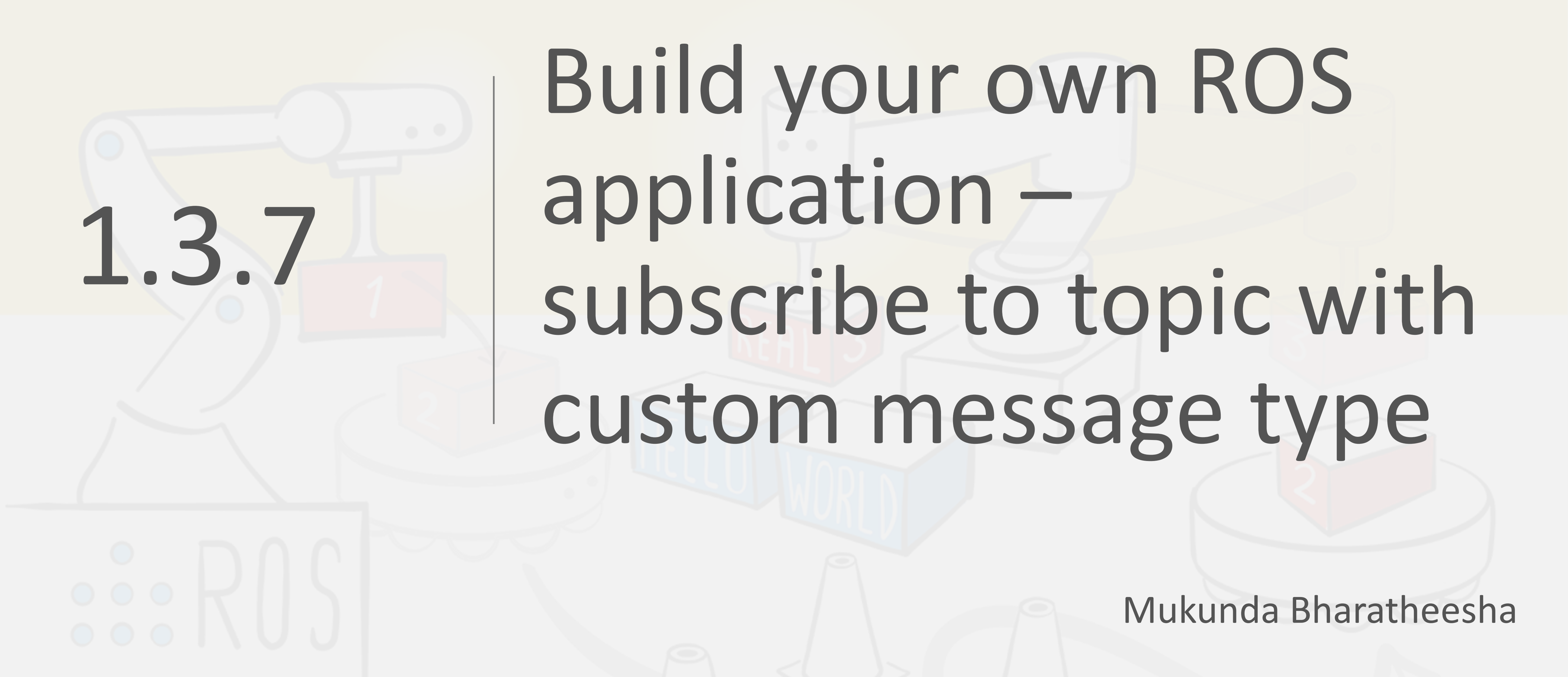

1.3.7

Build your own ROS application – subscribe to topic with custom message type

Mukunda Bharatheesha

Create subscriber

Use the template and start editing it. We will first create a new file, and copy the contents of the example file into it.

```
$ roscd hrwros_week1/scripts

$ touch sensor_info_subscriber.py

$ cp template_subscriber_script.py sensor_info_subscriber.py
```

Next, we will take a look inside our new file. Open it in your favourite editor, and follow along with the lines we indicate. #L16 means line 16.

Import sensor information

```
#L36: from hrwros_msgs.msg import SensorInformation
```

Then give a meaningful names and log message (#L41,42,49)

```
#L51: rospy.init_node('sensor_info_subscriber', anonymous=False)

#L53: rospy.Subscriber('sensor_info', SensorInformation,
sensorInfoCallback)

#L59: sensorInfoListener()
```

Create subscriber

Make the subscriber script executable

```
$ chmod +x sensor_info_subscriber.py
```

To run our new subscriber node:

Run roscore in a new course command shell. Remember you need to source your workspace in every new shell!

```
$ source $HOME/hrwros_ws/devel/setup.bash
$ roscore
```

Then, in a separate CCS:

```
$ source $HOME/hrwros_ws/devel/setup.bash
$ rosrun hrwros_week1 sensor_info_subscriber.py
```

Create subscriber

Check with rostopic list:

```
$ rostopic info /sensor_info
```

Important Note:

Inspect the topic with the rostopic info when debugging your ROS problems.
To actually see the contents of the topic, you will need to publish it from a new CCS.

```
$ rosrun hwros_week1 sensor_info_publisher.py
```