

Assignment - 4

Name : Shree Varshan V

Identify any of the following things and Make a document about their functionality and importance

- 3 ROS Topics**
- 3 ROS Services**
- 3 Actions**
- Navigation messages, Geometry Messages, and sensor messages.**

Answer

- 3 ROS Topics

Rostopic contains the rostopic command-line tool for displaying debug information about ROS Topics, including publishers, subscribers, publishing rate, and ROS Messages.

Topics are named buses over which nodes exchange messages. Topics have anonymous publish/subscribe semantics, which decouples the production of information from its consumption. In general, nodes are not aware of who they are communicating with. Instead, nodes that are interested in data subscribe to the relevant topic; nodes that generate data publish to the relevant topic. There can be multiple publishers and subscribers to a topic.

Topics are intended for unidirectional, streaming communication. Nodes that need to perform remote procedure calls, i.e. receive a response to a request, should use services instead. There is also the Parameter Server for maintaining small amounts of state.

• Topic Types

Each topic is strongly typed by the ROS message type used to publish to it and nodes can only receive messages with a matching type. The Master does not enforce type consistency among the publishers, but subscribers will not establish message transport unless the types match. Furthermore, all ROS clients check to make sure that an MD5 computed from the msg files match. This check ensures that the ROS Nodes were compiled from consistent code bases.

- **Topic Transports**

ROS currently supports TCP/IP-based and UDP-based message transport. The TCP/IP-based transport is known as TCPROS and streams message data over persistent TCP/IP connections. TCPROS is the default transport used in ROS and is the only transport that client libraries are required to support. The UDP-based transport, which is known as UDPROS and is currently only supported in roscpp, separates messages into UDP packets. UDPROS is a low-latency, lossy transport, so is best suited for tasks like teleoperation.

ROS nodes negotiate the desired transport at runtime. For example, if a node prefers UDPROS transport but the other Node does not support it, it can fallback on TCPROS transport. This negotiation model enables new transports to be added over time as compelling use cases arise.

- **Topic statistics**

roscpp and rospy offer integrated measurement of the following parameters for every connection:

- Period of messages by all publishers (average, maximum, standard deviation)
- Age of messages, based on header timestamp (average, maximum, standard deviation)
- Number of dropped messages
- Traffic volume (Bytes)

All measurements are performed on a window, that resizes automatically depending on the number of messages published.