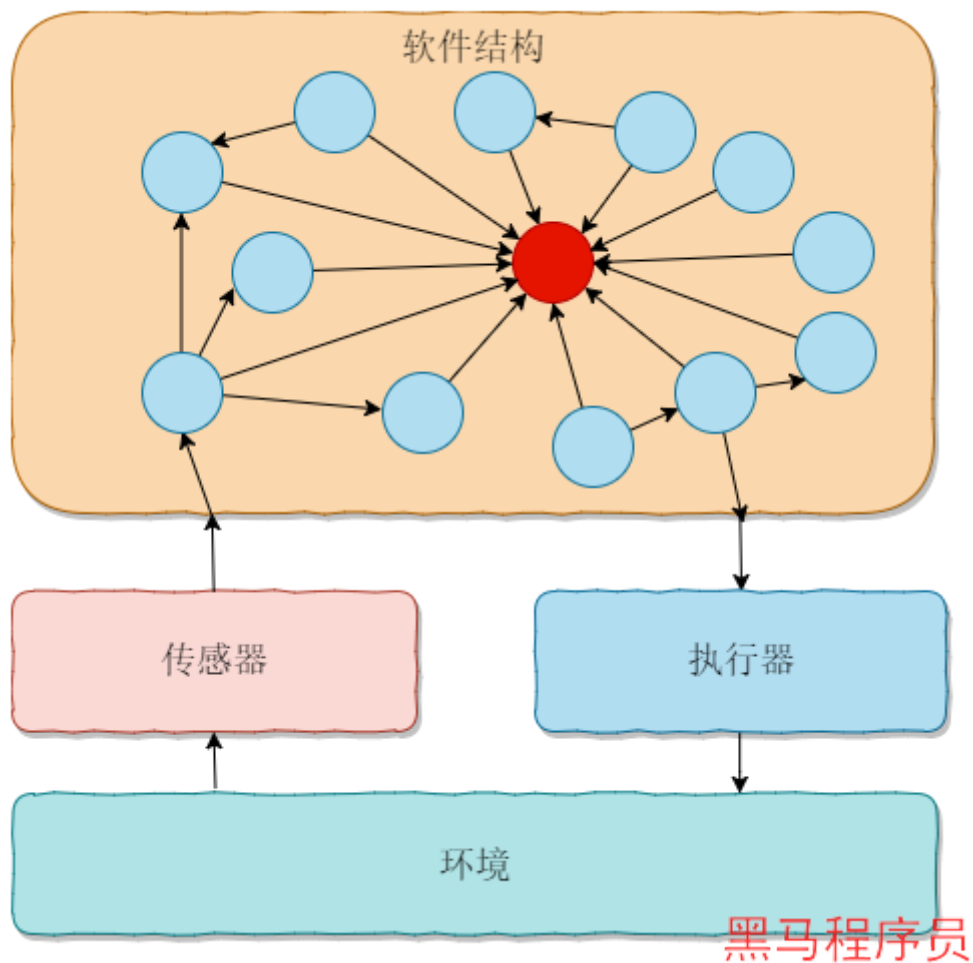


Topic通讯概念

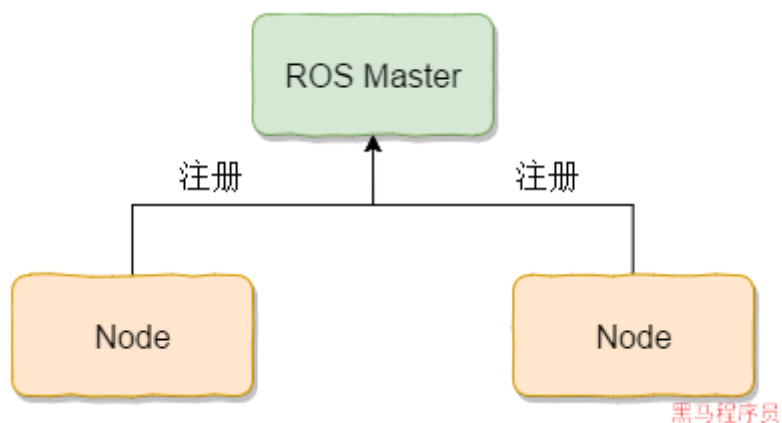
ROS要解决的问题



- ros整个框架下，我们开发大量的节点
- 节点间需要有数据的交互
- 数据该如何交互，这个是ROS要解决的问题，也是核心问题。

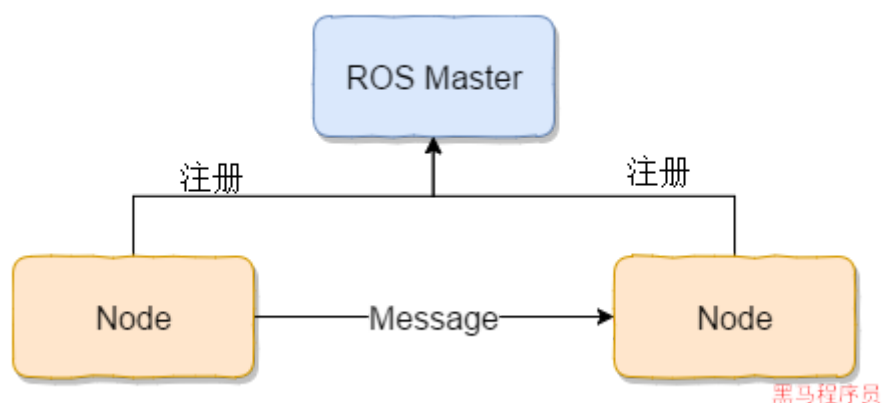
一些思考

ROS整个工程启动后的一个结构现状如图：



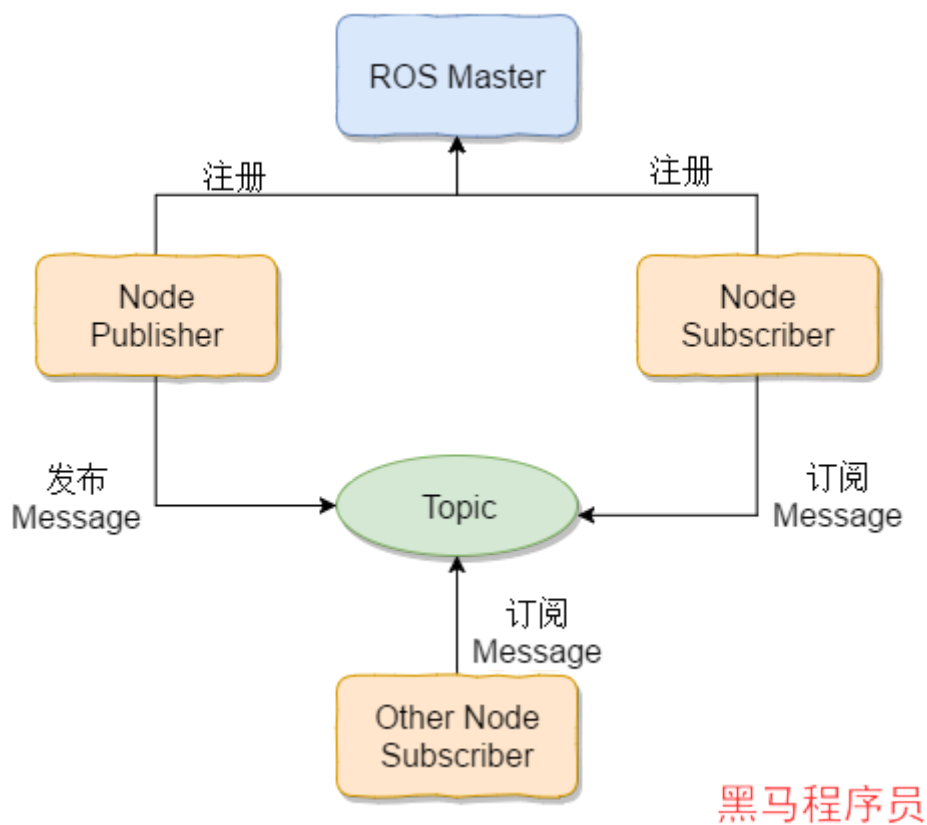
多个Node节点都需要到ROS Master进行注册。

每个Node完成自己的功能逻辑。有的时候Node和Node间需要有数据的传递，这个时候ROS提供了一种数据通讯机制。



通讯原理图

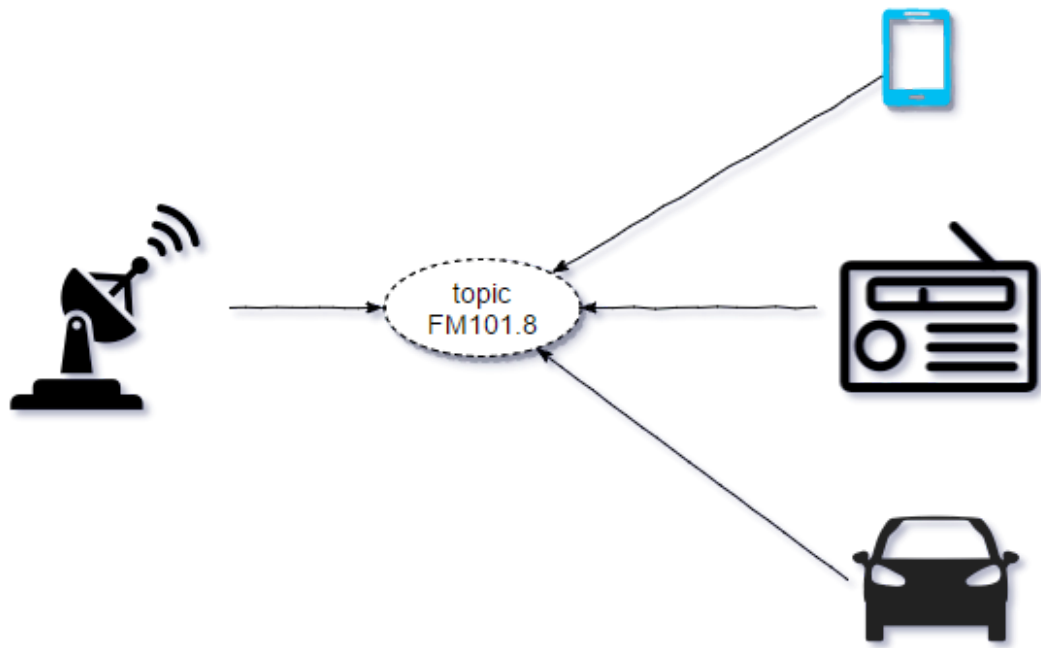
ROS 在设计Node间通讯机制的时候，考虑的还是比较周详的，设计了Topic通讯机制，如下图：



Node间进行通讯，其中发送消息的一方，ROS将其定义为 **Publisher(发布者)**，将接收消息的一方定义为 **Subscriber(订阅者)**。

考虑到消息需要广泛传播，ROS没有将其设计为点对点的单一传递，而是由 **Publisher** 将信息发布到 **Topic(主题)** 中，想要获得消息的任何一方都可以到这个 **Topic** 中去取数据。

理解



黑马程序员

我们可以将这种通讯方式用现实中的例子做类比。

例如：广播站和收音机。

- 广播站好比是 `publisher`
- 收音机好比是 `subscriber`
- 收听频段好比是 `topic` 主题
- 广播站不停的往外广播消息，不关心是否有谁去接收
- 多个收音机可以去 同一个频段收听广播，收听到的内容是相同的