ROS.NET Unity for Multiplatform Applications

Jordan Allspaw
Carlos Ibarra Lopez
Eric McCann
What is ROS.NET?

• A series of C# projects that allow a MANAGED .NET application to communicate traditional ROS nodes
Standard ROS Features

- Publishers & Subscribers
- Parameters
- Services
- Custom Messages
Tools/Samples

• Compressed Image View
• Point Cloud and LaserScan Visualization
• URDF / Robot Model
• Dynamic Reconfigure
• TF Tools
ROS.NET Syntax

```csharp
class SamplePublisher : MonoBehaviour {
    public RosCore rosmaster;
    private Publisher<String> pub;

    void Start () {
        NodeHandle nh = rosmaster.getNodeHandle();
        pub = nh.advertise<Message<String>>("/chatter", 10);
    }

    void Update () {
        Message<String> msg = new Message<String>();
        msg.data = "Welcome to ROSCON!";
        pub.publish(msg);
    }
}
```
Why use Unity (and Windows)

- Some devices only support Windows
- Support for Windows, Linux, Mac OS, Android, IOS
- Quickly utilize 3D graphic capabilities
Sample Usage with HTC Vive
Thanks!

https://github.com/uml-robotics/ROS.NET

https://github.com/uml-robotics/ROS.NET_Unity