

## ¿QUÉ ES ROS?

ROS, o *Robotics Operating System* en inglés, es una herramienta de programación que permite conectar distintos componentes o programas, de manera que compartan información. Está compuesto por nodos o procesos, *topics* y mensajes.

Tiene su origen en la Universidad de Stanford, hace 9 años, y desde entonces se ha convertido en un estándar que utilizan muchos investigadores como herramienta para descubrir nuevos avances en el campo de la robótica.

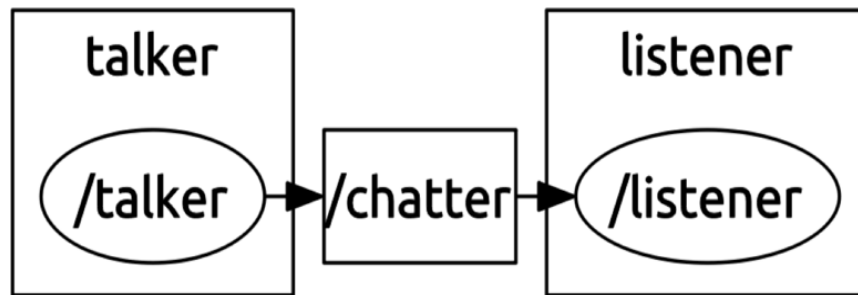


## ¿CÓMO FUNCIONA ROS?

ROS funciona de la siguiente manera muy similar a un chat. Una persona escribe mensajes y los demás los reciben, a la vez que envían los suyos propios. A parte, puedes tener un chat de diferentes temáticas y que varias personas estén conectadas a uno o varios a la vez.

Haciendo la similitud entre ROS y el chat, los procesos son las personas y los *topics* o temas son los chats y sus diferentes temáticas. De esta manera, si un programa quiere saber un dato, por ejemplo, la temperatura ambiente, se conectará al *topic* de temperatura y le empezarán a llegar datos.

Por otro lado, habrá un programa conectado a un sensor de temperatura que publicará los datos que vaya obteniendo. Estos datos de temperatura son los mensajes

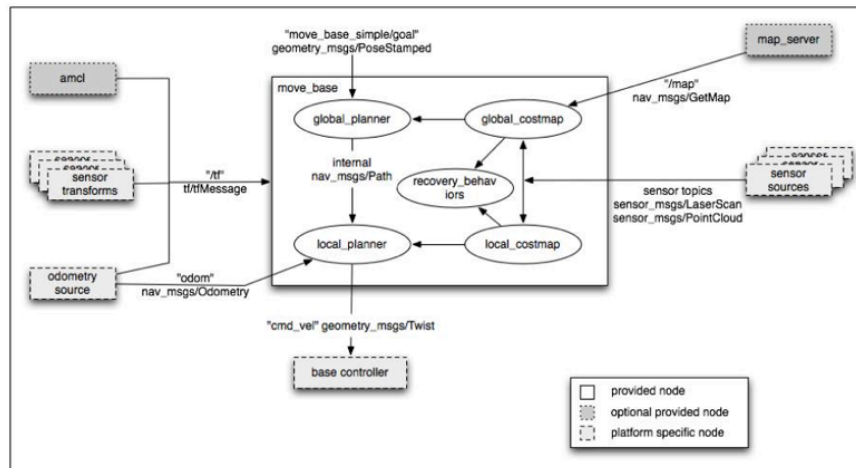


Estructura sencilla con un publicador (talker), un *topic*, y un subscriptor (listener)

## ¿QUÉ APORTA A ROBDOS TEAM?

Con ROS conectamos los distintos componentes software que tenemos, de esta manera habrá varios nodos que publiquen datos de los sensores, cada uno del suyo, y otros que recojan esa información, procesándola, y, si hace falta, publicando esta nueva información y haciéndola disponible para otros procesos.

Mediante esta cadena de nodos y *topics*, se construye la estructura de programas que soporta y define el comportamiento de WASABI.



Ejemplo de estructura compleja donde se construye el mapa del entorno con un robot genérico