URDF: Adding simple geometry, part 1

Hi, let's continue with getting our factory simulation ready.

In the previous videos we've already removed most of the bins from the factory and we've also moved the bin we kept around to its final location.

As we'll be adding a second robot to the factory in a later video, we'll need somewhere to put it. If we'd put it directly on the floor, the robot would not be able to reach either the bin or the turtlebot, so we'll have to raise it slightly.

We'll do that by mounting it on something called a pedestal.

A pedestal is essentially just a box-shaped platform or column, so we'll use the support for box geometries in URDF to model it.

This will show us how we can use primitive shapes to add objects to our world - such as boxes, cylinders and spheres.

Primitive shapes can be very useful to quickly create simple objects in a URDF, as opposed to using a CAD program to create life-like models and then importing those.

As our pedestal does not need to be very detailed, we'll just add a box and give it the size and shape of the pedestal we need.

Just to remind ourselves, here's an overview of the factory we've built so far.

We have robot_1 near the conveyors and we already moved the bin we kept to the south, behind the pallet with the boxes on it.

Now we'll add the pedestal to the left of the bin, so robot_2 can be placed on-top of it.

So let's open up our text editor and add the pedestal!

We'll open our ‘h-r-double-u-ros.xacro’ file again and scroll all the way down to the bottom.

Here are the lines we changed to move the bin around.

Let's go past them and create some space.

We're going to have to add two things: a link -- this will describe our box - and a joint - which will give it a position in the factory world.

All this is xml, so we'll add the link with an opening and a closing tag and give it a name.

This name is important!

It must be unique, or the urdf loader will complain, and it should ideally immediately make the purpose of the link clear.

As it is right now, this link will not be visible in the simulation, as it has no shape associated with it and we also haven't connected it to the rest of the urdf tree.

To do that, we have to give it a visual element, which will be a box in this case.

For now we'll give the sides all the same length, but we'll fix that later.

Notice how there are no commas or semicolons to separate the numbers, it's just spaces. With the box defined, we'll fixate it in place in the factory world by creating a joint.

Just as with the link, we're giving this joint a clear and meaningful name, but here we'll also have to configure the TYPE of joint.

As we don't need the pedestal to move anywhere, we'll make it a fixed joint and set the name to 'pedestal_joint', as it is the joint that connects the pedestal to the world.

Remember: these names have to be unique or loading the urdf will fail.

With the link and joint in place, we are ready to take a look at what we've created.

First save the urdf, then switch to the terminal and start the visualization launch file.

So we've added the box, but it's far from usable as a pedestal right now.

It's not the right shape as it's far too big.

It doesn't have a color yet: as you can see it defaults to red.

The box is stuck halfway inside the floor.

And it's not in the correct location.

We've made quite some progress, but we're not done yet.

So far, we've defined the basic structure of the pedestal by adding a link and by connecting it to the rest of the factory world by adding a joint.

To give the pedestal an appearance, we've also defined the geometry for that link, which was a box.

But we'll still need to make some changes for the box to become a proper pedestal for our robot.

We have to give it the right dimensions. We have to choose a color for it.

The pedestal should be placed on the floor instead of halfway below it. And finally the pedestal should be located next to the bin.

Let's do all of that in the next video.