# Weld Joint on parent/child Objects with Bodies

Using objects with children that also have bodies can be tricky if they are required to stay fixed together.

When something collides with the parent or child, they will break apart. This is the expected behaviour. But it may not be the effect you want.

In this tutorial we'll cover two things:

1. Having an object with at least one child, and having a projectile hit both objects, which will separate them.
2. Introducing a weld joint to keep the parent and child together in the case of collisions.

Our test object will be a crude ship that is in two parts. The main part will be the parent, the child will be the front section of the ship. A projectile is fired at the ship. If the projectile hits either part, the ship will separate.

Here are our two ship sprites, and a projectile.



First, we'll start with setting the physics, and the gravity and turning on the bodies for debug:

```
[Physics]
Gravity        = (0, 0, 0)
ShowDebug  = true
```

Begin with the main ship config. It will need Object, Graphic, and Body sections:

```
[Ship]
Graphic          = ShipGraphic
Rotation    = 90
Position    = (0, 0, 0.5)
Body        = ShipBody

[ShipGraphic]
Texture        = ship.png
Pivot          = center
Smoothing      = true
```

```
[ShipBody]
Dynamic        = true
PartList   = ShipBodyPart
LinearDamping  = 1
AngularDamping = 2
FixedRotation  = false

[ShipBodyPart]
Type       = box
Solid      = true
SelfFlags  = ship
CheckMask  = projectile
Friction   = 1.2
```

Next, the child part of the ship which is the front extra cockpit section:

```
[ShipChild]
Graphic     = ShipConeGraphic
Body       = ShipChildBody
Position    = (0, -32, 0)

[ShipConeGraphic]
Texture       = ship-cone.png
Pivot         = center
Smoothing     = true

[ShipChildBody]
Dynamic        = true
PartList   = ShipChildBodyPart

[ShipChildBodyPart]
Type       = box
Solid      = true
SelfFlags  = shipchild
CheckMask  = projectile
```

Let's add this child object to the main ship as a child.

```
[Ship]
Graphic         = ShipGraphic
Rotation   = 90
Position   = (0, 0, 0.5)
ChildList  = ShipChild
Body       = ShipBody
```

Next step is to define the projectile that will towards the ship from the bottom right of the screen and strike the ship:

```
[Projectile]
Graphic     = ProjectileGraphic
Position    = (200, 200, 0.5)
Body        = ProjectileBody
Speed       = (-400, -500, 0)

[ProjectileGraphic]
Texture       = projectile.png
Pivot         = center
Smoothing     = true

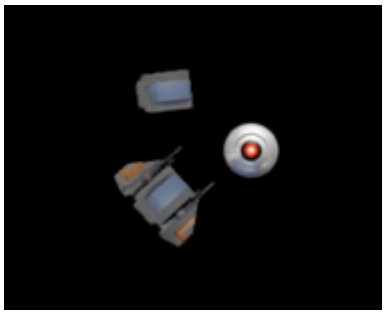
[ProjectileBody]
Dynamic        = true
PartList   = ProjectileBodyPart
LinearDamping  = 1
AngularDamping = 2
FixedRotation  = false

[ProjectileBodyPart]
Type       = sphere
Solid      = true
SelfFlags  = projectile
CheckMask  = ship # shipchild
Friction   = 1.2
Density        = 20
```

The code in the Init() function to create the objects:

```
  orxObject_CreateFromConfig("Ship");
  orxObject_CreateFromConfig("Projectile");
```

Let's run that.



*(The image above shows the projectile hitting the ship, but with physics debug off for clarity)*

The projectile strikes between the main part of the ship and the child behind, sending both parts off in different directions. Normally this would not occur, but because a body has been added to the child, the two won't stay together if it collides with something.

This may be the effect you are after. If you have a multi-part object, you may wish the parts to break off on contact immediately.

However, if you want the whole ship to stay together during a collision, then you need to use a joint.

In our case we'll use a weld joint which welds two objects together. Or more correctly, welds a child (with a body) to it's parent:

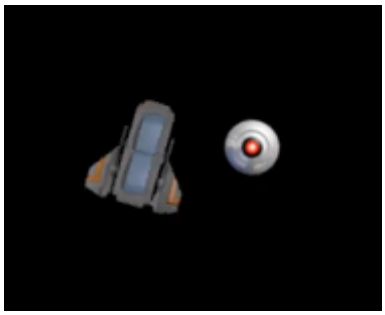
```
[WeldJoint]
Type        = weld
ParentAnchor    = (0, -32, 0)
ChildAnchor     = (0, 0, 0)
Collide         = false
```

Note the Anchor positions. These override any child position that might be defined on the child object.

And in the ship object, add the WeldJoint to a ChildJointList. Where there is an entry in ChildList, there is a matching entry in ChildJointList:

```
[Ship]
Graphic          = ShipGraphic
Rotation    = 90
Position    = (0, 0, 0.5)
ChildList   = ShipChild
ChildJointList = WeldJoint
Body        = ShipBody
```

Run again, and the projectile will first strike between the main body of the ship and the child section of the ship, and will send the whole spinning. But the ship stays together as a whole unit.



# Troubleshooting

Avoid using:

```
Density = 0
```

in a child BodyPart. The effect will be to remove rotation when a projectile hits any of the bodies.

From:
https://www.orx-project.org/wiki/ - **Orx Learning**

Permanent link:
**https://www.orx-project.org/wiki/en/tutorials/physics/weld_joint_on_objects_with_bodies**

Last update: **2025/09/30 17:26 (14 months ago)**