

## How to make a ship capable of launching Shuttles: By Dragon\_UK

You will first need the following:

- 1/ MPE (Model Property Editor- That Comes with the SDK)
- 2/ Latest Launching Framework

---

There are two things which need to be done:

- A- Add in the Shuttle Bay to a ship of your choice. (Using MPE)
- B- Add the Shuttles to the Ship (Using an external script)

Note: IF THE SHIP ALREADY HAS A “**Shuttle Bay OEP**” AND “**Shuttle Bay**” THEN YOU CAN SKIP PART A.

The first thing you need to do is open up the ship you want to have shuttles with MPE. I won't go into detail about how to use MPE; I'm hoping you already know how to add in new properties.

### **Part A- Create the Shuttle Bay**

1/Add in a new **Object Emitter**, call it “**Shuttle Bay OEP**”, then chose where you want it etc. and make the object emitter a *shuttle* from the drop down menu.

2/Add in a new **Hull** property and name it “**Shuttle Bay**”, this is not exactly a hull, its just used to illustrate where the shuttle bay is, your best off making it in the same location as the above property. Modify the properties so that it isn't *critical* or *primary*, and that it is *targetable*.

The ship is now set up to become a carrier.

### **Part 2 - Add in the shuttles**

1/ Rename the example script to the ship you are using. (Make sure it is the name of the ship script, not the hardpoint)

Refer to the screenshot to complete the next section.

2/ Rename the words with the **RED** line under them to the ship script (NOT THE HARDPOINT). If you were doing this for the Norway Class, you would write “**norway**” in this place.

3/ Rename the parts with the **BLUE** line under them to the **shuttle** name you want to use, the stock shuttle is called “**Shuttle**”, but the example includes the **PeragrineF1** (providing you have it, if you don't then delete that line, or change "**PeragrineF1**" to the word "**Shuttle**", this will just mean that there are more standard shuttles.

4/ The parts with the **YELLOW** line under them correspond to the amount of THAT TYPE of shuttle you have specified, e.g. 5 = Five **PeragrineF1**'s available, and 12 = Twelve **Shuttles** available.

That's basically it; if you want another type of shuttle to be included, then just copy the line with the **GREEN** arrow next to it and paste it below, then redo 2,3 & 4.

```

# Permission to redistribute this code as part of any other packaging requires
# the explicit permission of the author in advance.
#####

# A simple example of how to define a custom Carrier with a specified
# compliment of vessels
Carrier = __import__( "ftb.Carrier")
class Dauntless( Carrier.Carrier):
    def __init__( self, pShip):
        Carrier.Carrier.__init__( self, pShip)
        # The script name should be the name of the "ship" script, not the
        # hardpoint (otherwise, you'll crash your BC and this is bad)
        LauncherGroup = __import__( "ftb.LauncherGroup")
        group = LauncherGroup.LauncherGroup()

        LauncherManager = __import__( "ftb.LauncherManager")
        launcher = LauncherManager.GetLauncher( "Shuttle Bay", pShip)
        group.AddLauncher( "Shuttle Bay", launcher)
        #launcher.AddLaunchable( "PeragrineF1", "ftb.friendlyAI", 5)
        launcher.AddLaunchable( "Shuttle", "ftb.friendlyAI", 12)

        #launcher = LauncherManager.GetLauncher( "Shuttle Bay 2", pShip)
        #group.AddLauncher( "Shuttle Bay 2", launcher)
        #launcher.AddLaunchable( "Shuttle", "ftb.friendlyAI", 2)

        #launcher = LauncherManager.GetLauncher( "Shuttle Bay 3", pShip)
        #group.AddLauncher( "Shuttle Bay 3", launcher)
        #launcher.AddLaunchable( "Shuttle", "ftb.friendlyAI", 6)

        self.AddLauncher( "Group 1", group)

        # Play with this feature if you dare... MUHAHAHAHAHAHAHAHAHAHAHAHA!!!!
        #group.SetLaunchMode( LauncherGroup.ALL)

    # Define how much Shuttles we can carry maximal (Return Shuttles script)
    def GetMaxShuttles(self):
        return 50
    # Don't use the following Tractors to Dock:
    def IgnoreTractors(self):
        return ["Aft Tractor 2", "Forward Tractor 1", "Forward Tractor 2"]

# "Dauntless" is the "ShipProperty" name of the ship to be registered as
# defined in the Hardpoints PY file for your ship
ShipManager = __import__( "ftb.ShipManager")
ShipManager.RegisterShipClass( "Dauntless", Dauntless)

```

Everything should work fine providing you have done things correctly. Only change the parts that are referred to in this guide in order to avoid errors.