

func_vehicle

How to implement func_vehicle 0.1 in Copper 1.20

func_vehicle comes with 6 new .qc files, one of which is actually from copper, namely maths.qc. This means we need to skip adding the included maths.qc file, but do add the other 5 files into **progs.src** thusly:

```
...
maths.qc

phys_defs.qc
phys_matrix.qc
phys_exec.qc

func_vehicle.qc
func_vehicle_builtins.qc

utility.qc
subs.qc
...
```

Now we have the vehicle stuff included in the game, next we need to add interactions between the vehicle(s) and the rest.

First we want to add some functionality to **client.qc** for driving and entering/exiting the vehicle. In PutClientInServer, right below the setmodel stuff, add this:

```
...
    setmodel (self, "progs/player.mdl");
    modelindex_player = self.modelindex;

    setmodel (self, "progs/null.spr");
    modelindex_null = self.modelindex;

    stuffcmd(self, "chase_active 0\n");
    stuffcmd(self, "crosshair 1\n");

    setsize (self, VEC_HULL_MIN, VEC_HULL_MAX);
...
```

This is for allowing the player model to be invisible when seated, and resets the chasecam if a player stops playing while seated in a vehicle.

Still in **client.qc**, move down to `PlayerPostThink`, and add this under `CheckPowerups`:

```
...
    CheckPowerups ();

    TryUpdateVehicle();

    if (self.movetype != MOVETYPE_NOCLIP)
...

```

This handles the driving when seated in a vehicle.

When the player dies, they need to get tossed out from the vehicle. Open up **player.qc** and navigate to `PlayerDie` to add this:

```
...
void() PlayerDie =
{
    local float i;

    if (self.driving)
        Car_AttachToVehicle(self, self.driving); // will jump out from
vehicle immediately

    self.items = self.items - (self.items & IT_INVISIBILITY);
...

```

Next up we need to add a new impulse so players can enter and exit nearby vehicles. Open up **impulse.qc** and navigate to `ImpulseCommands` to add the following:

```
...
    case 12:
        CycleWeaponReverseCommand();
        break;
    case 50:
        Car_CheckAttach ();
        break;
/*
    // ai_nav
...

```

You've successfully added the basic functionality for entering, exiting and driving vehicles! Keep going if you also want to add the ability for monsters to detect you inside a vehicle and shoot at you.

In **ai.qc**, put this into visible:

```
...
    if (trace_inopen && trace_inwater)
        return FALSE;           // sight line crossed contents

    if (trace_fraction < 1 && trace_ent.owner.driving == targ)
        return TRUE;

    if (trace_fraction == 1)
...

```

This way monsters can detect players in cars.

Still in **ai.qc** we need to add the following to CheckClearAttack:

```
...
    if (trace_ent != self.enemy)
    {
        if (trace_ent.classname == "player")
            return TRUE;        // go ahead and hit a player
        if (trace_ent.owner.driving == self.enemy)
            return TRUE;

        return FALSE;          // don't have a clear shot
    }
...

```

This allows monsters to fight vehicles.

Another change in **ai.qc** is we need to remove the following from FilterTarget:

```
float(entity client) FilterTarget =
{
    if (client.flags & FL_NOTARGET || client.movetype == MOVETYPE_NOCLIP)
        return FALSE;
    if (client.health <= 0 || client.deadflag)
        return FALSE;
    if (client.customflags & (CFL_PLUNGE|CFL_LIMBO))
        return FALSE;
    if (range(client) == RANGE_TOOFAR)
        return FALSE;

    return TRUE;
}

```

This is because the player is using MOVETYPE_NOCLIP when inside a vehicle.

In **combat.qc**, navigate to **T_RadiusMultiDamage**, and change this part:

```
if (points > 0)
    T_RadiusDamageCheck(inflictor, attacker, head, points, multi);
```

Into this:

```
if (points > 0)
{
    vector dir = normalize(org - inflictor.origin);
    Phys_Force(head, 1000, dir * points, org, dir);
    T_RadiusDamageCheck(inflictor, attacker, head, points, multi);
}
```

Cars can now get pushed around by explosions.

If you want shambler lightning to do damage, open up **m_shambler.qc** and find the **CastLightning** function and change the following:

```
...
    traceline2(org, self.origin + dir * SHAM_LIGHTNING_RANGE, self,
TRACE_NOMONSTERS);

    ClearMultiDamage();
    LightningBeam (org, trace_endpos + dir*4, self, 10);
    ApplyMultiDamage();
}
...
```

And you're done!

/ Marko "Polo" Permanto