

Scripting Tutorials 2012

Trolling scripts with tags

9/3/2012
Kirby_422

Table of Contents

Does this really need sub-categories?	1
Double Jump	1
Plasma Pistol Battery Charge (OpenSauce)	2
Boosting (OpenSauce)	3

Does this really need sub-categories?

Sometimes, there is certain things you cannot do with scripting alone. Factors that you cannot check. When these come into play, we need to troll the scripts into doing what we want them to, by using tags; **TROLL WILL ALL OF YOUR MIGHT!**

Double Jump

So, say we want a double jump. We have to think of a few things: What triggers this? Ie, what im I watching. What is going to perform this second jump? Where is this applicable? (MP/SP)

So, we'll quickly grab our basis. We need to check when someone presses jump; (player_action_test_jump), and way of detecting when you are on the ground. Since we are using a player_action_test, this means we're targeting SP. The way we're gonna perform our second jump might as well be a damage_effect. So, now we think about the tags required for detecting the ground. For this, a simple vehicle shooting projectiles downwards works; the projectiles' max travel distance would be the same as its height from the ground, where ever you choose to attach this vehicle. The inpackt damage_effect would be about the same size. When it hits ground, this damage_effect kills the vehicle, and only the vehicle.

Now that we've gathered a billion scraps, and you've made your vehicle (You're doing that on yourself, I'm too lazy to make the tags for you guys. I only have some that are half setup for samus from a many years ago), it's time to get onto the script!

```
(script static unit player (unit (list_get (players) 0)))
(script continuous double_jump
(sleep_until (player_action_test_jump) 5) ;;Check for the jump button
(sleep 10) ;;We're gonna wait a bit to let them get away from the ground first.
(player_action_test_reset) ;;Let the game forget about the jump check.
(object_create_anew inair_detector_vehicle) ;;Create our detector object
(objects_attach (player) "body" inair_detector_vehicle "") ;;Attach
(sleep_until (or (> (unit_get_health inair_detector_vehicle) 0) (player_action_test_jump)) 1)
;;Wait until they are no longer inair, or they have pressed jump.
(if (player_action_test_jump) (begin ;;If the reason it advanced was because of the jump button,
(effect_new_on_object_marker "effects\doubleJump" (player) "left foot") ;;Effect leftfoot
(effect_new_on_object_marker "effects\doubleJump" (player) "right foot") ;;Effect rightfoot
))
(sleep_until (> (unit_get_health inair_detector_vehicle) 0) 1) ;;Wait until they are on ground
;;if they weren't already
(player_action_test_reset) ;;Forget about the jump button being pressed.
) ;;End of script, repeat back to the top.
```

Plasma Pistol Battery Charge (OpenSauce)

Now that my cat has stopped trying to smash every dish in the house, lets continue! Other halo games apparently have this charge stuff, where your battery lowers constantly when you've charged your weapon.

So, we want something to check if they are charged, we want to alter battery (Also known as age). To check if its charged, we don't have any commands. We could go similar to the last example, and have the plasma pistol kill something when charged, or effect this things shield, and have it drain battery at a speed relative to the remaining shield. For this example, however, we're gonna rely on animations and scenery. We'll use an animation to drag the scenery in closer, and when its within a certain distance, that will be considered charging. Now, for the battery, default halo has nothing that can do this. We'll just (`weapon_data_get_real <object> <string>`) and (`weapon_data_set_real <object> <string> <real>`), but then you see the next thing. This requires an object, where will we get that? OS again. (`unit_data_get_object <object> <string>`) With this, we can ask for the players weapon. But then, how do we know what weapon in their inventory is the plasma pistol? (`unit_data_get_integer <unit> <string>`) to ask for `current_weapon_index`. Now, we ask if any of these have conflicts with MP... The nature of it should work fairly fine in MP, but the nature of the scripts. `weapon_data_set_real` will not function in multiplayer, it is set to not even run if the engine is set to multiplayer. So, with that, we know we want SP sadly.

```
(global object held_pp none)
(script static unit p0 (unit (list_get (players) 0)))
(script continuous charge_drainer
  (sleep_until (unit_has_weapon_readied (p0) "weapons\plasma pistol\plasma pistol_H3") 1)
  ;;Wait until the player has the modified weapon (with the animation)
  (if (= (unit_data_get_integer (p0) "current_weapon_index") 0)
    (set held_pp (unit_data_get_object (p0) "weapon0"))
  ;;If the weapon is 0, save a hook to that weapon in the global held_pp
  (if (= (unit_data_get_integer (p0) "current_weapon_index") 1)
    (set held_pp (unit_data_get_object (p0) "weapon1"))))
  ;;If its weapon 1, hook that one instead
  (objects_attach held_pp "attach_01" pp_check_01 "")
  (objects_attach held_pp "attach_02" pp_check_02 "")
  ;;Attach whatever you're using to detect. I'm using two scenery to detect each other.
  ;;Its perfectly fine to detect between the scenery and the gun if you want.
  (objects_detach held_pp pp_check_01)
  (objects_detach held_pp pp_check_02)
  ;;Detach them, as objects_attach'd items position will always return where they were attached at,
  ;;and this will allow them to show their position correctly.
  (if (and (< (objects_distance_to_object pp_check_02 pp_check_01) 0.01) ;;If they are together
    (< (weapon_data_get_real held_pp age) 1)) ;;And the plasma pistol has battery to lose.
    (weapon_data_set_real held_pp age (+ (weapon_data_get_real held_pp age) 0.01))
```

```
;;Then decrease it by 1 unit (1.00 = 0% remaining, 0.50 = 50%, 0.00 = 100% remaining)
)
)
```

Boosting (OpenSauce)

What about if we want a generic script, that profiles a vehicle basic on its tags? How would you go about figure out what a vehicle is by its tags? Simply check where the seat is located in the vehicle. Have OpenSauce give you the distance between your seat and the vehicles core. Now, you're probably asking why this would matter if you've named the vehicle; I ask you this, what about if you didn't name that vehicle, that everything was learned merely by where its seat was? OS can return a vehicle that a player is offered to enter. With this, every player has their own global variable, that stores the vehicle they are driving. You could have 100 vehicles, but would only need to care about the 16 that players are beside.

So, for this, we would need to build a test map to gather facts about each vehicle. Build a map with named whatever you want to add. Enter that vehicle's driver seat, (objects_distance_to_object (players) YOURVEHICLE). record the numbers, try moving around a bit, they'll shift ever so slightly. get a good idea of the number.

For now, let's just set one up for one player.

```
(global unit current_vehicle (unit (list_get (players) 0))) ;;Global to store a vehicle
(script static unit player0 (unit (list_get (players) 0)))
(global boolean can_boost false) ;;Based on whether we'll let that ID boost.
(global short vehicle_id -1) ;;Each vehicle has its own ID number.
(script static void identify_vehicle
.....
;;-1 - unset
;;0 - Unknown (Won't boost with this)
;;1 - Ghost
;;2 - warthog
;;3 - scorpion
;;4 - banshee
;;5 - wraith
.....
(sleep 30) ;wait for insured results; To early, and the player might not be in place.
(if
  (and ;;If the player is within the bounds we've researched (give a bit of leanway)
    (> (objects_distance_to_object (players) current_vehicle) 0.4369)
    (< (objects_distance_to_object (players) current_vehicle) 0.4371)
  )
  (set vehicle_id 1) ;;Then set the ID to the ID of that vehicle.
  (if
    (and
```

```

    (> (objects_distance_to_object (players) current_vehicle) 0.2219)
    (< (objects_distance_to_object (players) current_vehicle) 0.2221)
  )
  (set vehicle_id 2)
  (if
  (and
    (> (objects_distance_to_object (players) current_vehicle) 0.2121)
    (< (objects_distance_to_object (players) current_vehicle) 0.2123)
  )
  (set vehicle_id 3)
  (if
  (and
    (> (objects_distance_to_object (players) current_vehicle) 0.2044)
    (< (objects_distance_to_object (players) current_vehicle) 0.2046)
  )
  (set vehicle_id 4)
  (if
  (and
    (> (objects_distance_to_object (players) current_vehicle) 0.3367)
    (< (objects_distance_to_object (players) current_vehicle) 0.3369)
  )
  (set vehicle_id 5)
  (set vehicle_id 0) ;;If it cannot figure out, ID 0, Unknown.
  ))))
  ))))
)
(script continuous select_vehicle
  (sleep_until
  (and
    (!= (objects_distance_to_object (players) (player_data_get_object 0 nearest_object)) -1)
    ;;If the object is within scanable range.
    (= (unit_data_get_real (unit (player_data_get_object 0 nearest_object)) driver_power) 0)
    ;;And it's not being driven (Since we dont care about other vehicles in this case)
  )
  1)
  (set current_vehicle (unit (player_data_get_object 0 nearest_object))) ;;Then hook into that vehicle.
  )
)
(script continuous vehicle_boost_check
  (sleep_until
  (and
    (>= (unit_data_get_real current_vehicle driver_power) 0.5) ;;If it's being driven
    (< (objects_distance_to_object (players) current_vehicle) 1) ;;And still near the player.
  )
  )
  5)
  (identify_vehicle) ;;ID the vehicle.
  (sleep 1)
  (if (!= vehicle_id 0) (set can_boost true)) ;;If it isn't an unknown vehicle, allow it to boost.
  (sleep_until

```

```

(not (and ;;Wait until the conditions are no longer met (Power, or near)
(>= (unit_data_get_real current_vehicle driver_power) 1)
(< (objects_distance_to_object (players) current_vehicle) 1)
))
5)
(set can_boost false) ;;Prevent boosting.
(unit_set_desired_flashlight_state current_vehicle false) ;;Turn off flashlight.
(sleep 5)
(set vehicle_id -1) ;;Set ID to unset.
)
(script continuous vehicle_boost_preform
  (sleep_until (and can_boost (unit_get_current_flashlight_state current_vehicle)) 1)
  ;;Wait until they can boost, and have their flashlight on.
  (if (or (= vehicle_id 1) (= vehicle_id 4) (= vehicle_id 5)) ;;If they have unlimited boost, or
    (< (+ (abs_real (object_data_get_real current_vehicle transitional_velocity x))
      (abs_real (object_data_get_real current_vehicle transitional_velocity y))) 0.4)
    ;;Are within a certain speed bounds for limited vehicles.
    ) (begin ;;Then boost!
      (object_data_set_real current_vehicle transitional_velocity x
        (+ (object_data_get_real current_vehicle transitional_velocity x)
          (* 0.02 (object_data_get_real current_vehicle forward x))))
      (object_data_set_real current_vehicle transitional_velocity y
        (+ (object_data_get_real current_vehicle transitional_velocity y)
          (* 0.02 (object_data_get_real current_vehicle forward y))))

      (if (= vehicle_id 4) (object_data_set_real current_vehicle transitional_velocity z 0))
      ;;For banshees, we want to effect their vertical. For now, we'll just prevent them from vertical.
      ;;We could do the same as the X and Y, except with 'up' instead of forward, etc
    ))
)
)

```

If you wonder why this one feels more rushed, blame my cat. My fluffy, adorable, cat who feasts on human flesh and blood, because he can only eat the freshest meat right off a living person (he won't eat normal meat) Isn't my cat so cute?

Kirby_422