Scripting Tutorials 2013

Passing variables between maps

8/22/2013
Kirby_422

Table of Contents

What does this require, and Why, or When, should I use it?

To store data across maps is actually quite easy. Player settings are designed to stay loaded as long as the game is running, and luckily, the game still is designed the same way as it was for Xbox; We have 4 players settings to do whatever we want. More so, all player settings that scripts can tweak, are designed as arrays. This means, we can overwrite any data in the game with our settings if we so choose. This is not a good idea, but it is valid.

Why would you need this? Why wouldn't you want to send data between maps? You can check what weapons the player has at the end of a SP map, then give them that weapon in the next map, or transfer their health over. Making pokemon maps, what trainer wants to lose all their money, items, and pokemon when they move town to town? You can also include special gametype enhancements, that you can toggle rather than having it the default (IE, toggle something in ui.map, and then when you start the MP map, it is a AI variant instead)

The commands

The get commands, obviously recover the data. Set saves it. The short field is for the index you are looking for. 0-3 are allocated ingame for player data, however you can enter stuff like 100 if you want, it's just not guaranteed that it will not exception, or stay, since those are other parts of the game. Negative numbers tend to distort the graphics ingame like erasing fonts, altering render orders, etc.

Yaw and Pitch can store any decimal numbers. Digital/Gamepad forward/strafe can store decimals between 0 and 1. Digital/Mouse Yaw/Pitch can store decimals between ~0 and ~100. Mouse forward/strafe any decimal, etc. Test in-game to ensure the field you are using can correctly store the data you want to store.

```
(get_yaw_rate <short>)
(get_pitch_rate <short>)
(get_digital_forward_throttle <short>)
(get_digital_strafe_throttle <short>)
(get_digital_yaw_increment <short>)
(get_digital_pitch_increment <short>)
(get_mouse_forward_threshold <short>)
(get_mouse_strafe_threshold <short>)
(get_mouse_yaw_scale <short>)
(get_mouse_pitch_scale <short>)
(get_gamepad_forward_threshold <short>)
(get_gamepad_strafe_threshold <short>)
(get_gamepad_yaw_scale <short>)
(get_gamepad_pitch_scale <short>)
```

```
(set_yaw_rate <short> <real>)
(set_pitch_rate <short> <real>)
(set_digital_forward_throttle <short> <real>)
(set_digital_strafe_throttle <short> <real>)
(set_digital_yaw_increment <short> <real>)
(set_digital_pitch_increment <short> <real>)
(set_mouse_forward_threshold <short> <real>)
(set_mouse_strafe_threshold <short> <real>)
(set_mouse_yaw_scale <short> <real>)
(set_mouse_pitch_scale <short> <real>)
(set_gamepad_forward_threshold <short> <real>)
(set_gamepad_strafe_threshold <short> <real>)
(set_gamepad_yaw_scale <short> <real>)
(set_gamepad_pitch_scale <short> <real>)
```

Storing Weapons and Health example

Since most people probably do not have exact uses in mind, we'll do a general thing that anyone making their own SP campaign could use. We'll use the Digital Forward Throttle to store our data, since we don't need much data, and since it can only store from 1 to 0, its optimal. Since we only need 3 values, Health, Primary Weapon, and Secondary Weapon, we'll use indexes 2, 3, leaving 0 alone so we don't need to bother resetting it after loading the map, and 1 in case the research into controller indexes ever enables proper coop. Please note, that if you ever change index 0, please change it back as soon as the transaction is complete, otherwise it will have effects on the players controls.

So, we need to choose values from 0.000000 to 1.000000 for each weapon; Since that is so many possible combinations, we'll store both weapon in one field, and health in its own. We first must decide numbers to associate each weapon with.
00 pistol, 01 assault rifle, 02 shotgun, 03 sniper rifle, 04 flamethrower, 05 rocket launcher, 06 plasma pistol, 07 plasma rifle, 08 fuel rod, 09 needler, 10 energy sword, and then add any custom weapons you have afterwards.

```
(script static unit p0 (unit (list_get (players) 0)))
(script static void backup_data
        (set_digital_forward_throttle 2 0)
        ;;Initialize it to 0, incase for whatever reason they are holding a non-supported weapons
        ;;IE, they bump possessed a hunter or something.
        (cond ;;Figure out what weapon is readied, and set the value according to our previous plans
                ((unit_has_weapon_readied (p0) "weapons\pistol\pistol") (set_digital_forward_throttle 2 0.000000))
                ((unit_has_weapon_readied (p0) "weapons\assault rifle\assault rifle") (set_digital_forward_throttle 2 0.000001))
                ((unit_has_weapon_readied (p0) "weapons\shotgun\shotgun") (set_digital_forward_throttle 2 0.000002))
                ((unit_has_weapon_readied (p0) "weapons\sniper rifle\sniper rifle") (set_digital_forward_throttle 2 0.000003))
                ((unit_has_weapon_readied (p0) "weapons\flamethrower\flamethrower") (set_digital_forward_throttle 2
0.000004))
                ((unit_has_weapon_readied (p0) "weapons\rocket launcher\rocket launcher") (set_digital_forward_throttle 2
0.000005))
                ((unit_has_weapon_readied (p0) "weapons\plasma pistol\plasma pistol") (set_digital_forward_throttle 2
0.000006))
```

```
                    ((unit_has_weapon_readied (p0) "weapons\plasma rifle\plasma rifle") (set_digital_forward_throttle 2 0.000007))
                    ((unit_has_weapon_readied (p0) "weapons\plasma_cannon\plasma_cannon") (set_digital_forward_throttle 2
0.000008))
                    ((unit_has_weapon_readied (p0) "weapons\needler\needler") (set_digital_forward_throttle 2 0.000009))
                    ((unit_has_weapon_readied (p0) "weapons\energy sword\energy sword") (set_digital_forward_throttle 2
0.000010))
            )
        (cond ;;Figure out what weapon is possessed, but not readied, and add the value to what we previously set.
            ((and (unit_has_weapon (p0) "weapons\pistol\pistol") (not (unit_has_weapon_readied (p0)
"weapons\pistol\pistol")))
                        (+ 0 0)) ;;+0 is no change, no need to further process.
            ((and (unit_has_weapon (p0) "weapons\assault rifle\assault rifle") (not (unit_has_weapon_readied (p0)
"weapons\assault rifle\assault rifle")))
                        (set_digital_forward_throttle 2 (+ (get_digital_forward_throttle 2) 0.0001)))
            ((and (unit_has_weapon (p0) "weapons\shotgun\shotgun") (not (unit_has_weapon_readied (p0)
"weapons\shotgun\shotgun")))
                        (set_digital_forward_throttle 2 (+ (get_digital_forward_throttle 2) 0.0002)))
            ((and (unit_has_weapon (p0) "weapons\sniper rifle\sniper rifle") (not (unit_has_weapon_readied (p0)
"weapons\sniper rifle\sniper rifle")))
                        (set_digital_forward_throttle 2 (+ (get_digital_forward_throttle 2) 0.0003)))
            ((and (unit_has_weapon (p0) "weapons\flamethrower\flamethrower") (not (unit_has_weapon_readied (p0)
"weapons\flamethrower\flamethrower")))
                        (set_digital_forward_throttle 2 (+ (get_digital_forward_throttle 2) 0.0004)))
            ((and (unit_has_weapon (p0) "weapons\rocket launcher\rocket launcher") (not (unit_has_weapon_readied (p0)
"weapons\rocket launcher\rocket launcher")))
                        (set_digital_forward_throttle 2 (+ (get_digital_forward_throttle 2) 0.0005)))
            ((and (unit_has_weapon (p0) "weapons\plasma pistol\plasma pistol") (not (unit_has_weapon_readied (p0)
"weapons\plasma pistol\plasma pistol")))
                        (set_digital_forward_throttle 2 (+ (get_digital_forward_throttle 2) 0.0006)))
            ((and (unit_has_weapon (p0) "weapons\plasma rifle\plasma rifle") (not (unit_has_weapon_readied (p0)
"weapons\plasma rifle\plasma rifle")))
                        (set_digital_forward_throttle 2 (+ (get_digital_forward_throttle 2) 0.0007)))
            ((and (unit_has_weapon (p0) "weapons\plasma_cannon\plasma_cannon") (not (unit_has_weapon_readied (p0)
"weapons\plasma_cannon\plasma_cannon")))
                        (set_digital_forward_throttle 2 (+ (get_digital_forward_throttle 2) 0.0008)))
            ((and (unit_has_weapon (p0) "weapons\needler\needler") (not (unit_has_weapon_readied (p0)
"weapons\needler\needler")))
                        (set_digital_forward_throttle 2 (+ (get_digital_forward_throttle 2) 0.0009)))
            ((and (unit_has_weapon (p0) "weapons\energy sword\energy sword") (not (unit_has_weapon_readied (p0)
"weapons\energy sword\energy sword")))
                        (set_digital_forward_throttle 2 (+ (get_digital_forward_throttle 2) 0.0010)))
        )
        (set_digital_forward_throttle 3 (unit_get_health (p0))) ;;Store health in field 3. Halo already returns health as a value, 1 to 0.
        ;;;;;;;;;;;;;;;;;;;;
        ;; OS ONLY ADDITIONS (Pass grenades on)
        (set_digital_forward_throttle 2 (+ (get_digital_forward_throttle 2)
                                            (\ (unit_data_get_integer (p0) "total_grenade_count[plasma]") 10)
                                            (\ (unit_data_get_integer (p0) "total_grenade_count[frag]") 100)
        ))
        ;;The digits now stand, as 0.[PlasmaGrenade][FragGrenade][Weap2][Weap2][Weap1][Weap1] for our 6 decimals stored.
        ;;You can store 20 possible weapons (0-19) per slot, and 9 grenades per slot.
        ;;END OS ADDITIONS
        ;;;;;;;;;;;;;;;;;;;;
)

(global real raw_data 0)
(global short loaded_data 0)
(script static void load_data
        (set raw_data (get_digital_forward_throttle 2))
        (if (and (!= raw_data 1) (!= raw_data 0)) (begin ;;Check if anything is stored.
                    ;;;;;;;;;;;;;;;;
                    ;; OS ADDITIONS
```

```
(set loaded_data (* raw_data 10)) ;;Load data without decimals
(set raw_data (- (* 10 raw_data) loaded_data)) ;;Clear loaded data from raw data to prevent future confusion.
(unit_data_set_integer (p0) "total_grenade_count[plasma]" loaded_data)

(set loaded_data (* raw_data 10)) ;;Load data without decimals
(set raw_data (- (* 10 raw_data) loaded_data)) ;;Clear loaded data from raw data to prevent future confusion.
(unit_data_set_integer (p0) "total_grenade_count[frag]" loaded_data)
;;
;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;
;;DO THIS IF YOU DIDN'T USE THE OS ADD-ON, and DO NOT USE IT IF YOU DID USE THE OS ADD-ON.
(set raw_data (* 100 raw_data))
;;
;;;;;;;;;;;;;;;
;;;;;;;;;;;;;;;;;;

(set loaded_data (* raw_data 100)) ;;Load data without decimals
(set raw_data (- (* 100 raw_data) loaded_data)) ;;Clear loaded data from raw data to prevent future confusion.
(cond
        ((= load_data 0) (player_add_equipment (p0) "hp_start_prof" 0))
        ((= load_data 1) (player_add_equipment (p0) "ar_start_prof" 0))
        ((= load_data 2) (player_add_equipment (p0) "sg_start_prof" 0))
        ((= load_data 3) (player_add_equipment (p0) "sr_start_prof" 0))
        ((= load_data 4) (player_add_equipment (p0) "ft_start_prof" 0))
        ((= load_data 5) (player_add_equipment (p0) "rl_start_prof" 0))
        ((= load_data 6) (player_add_equipment (p0) "pp_start_prof" 0))
        ((= load_data 7) (player_add_equipment (p0) "pr_start_prof" 0))
        ((= load_data 8) (player_add_equipment (p0) "pc_start_prof" 0))
        ((= load_data 9) (player_add_equipment (p0) "ne_start_prof" 0))
        ((= load_data 10) (player_add_equipment (p0) "eb_start_prof" 0))
)
(set loaded_data (* raw_data 100)) ;;Load data without decimals
(cond
        ((= load_data 0) (player_add_equipment (p0) "hp_start_prof" 0))
        ((= load_data 1) (player_add_equipment (p0) "ar_start_prof" 0))
        ((= load_data 2) (player_add_equipment (p0) "sg_start_prof" 0))
        ((= load_data 3) (player_add_equipment (p0) "sr_start_prof" 0))
        ((= load_data 4) (player_add_equipment (p0) "ft_start_prof" 0))
        ((= load_data 5) (player_add_equipment (p0) "rl_start_prof" 0))
        ((= load_data 6) (player_add_equipment (p0) "pp_start_prof" 0))
        ((= load_data 7) (player_add_equipment (p0) "pr_start_prof" 0))
        ((= load_data 8) (player_add_equipment (p0) "pc_start_prof" 0))
        ((= load_data 9) (player_add_equipment (p0) "ne_start_prof" 0))
        ((= load_data 10) (player_add_equipment (p0) "eb_start_prof" 0))
)
) (player_add_equipment (p0) "default_start_prof" 1)) ;;If no data was retrieved or empty, load defaults.
(if (!= (get_digital_forward_throttle 3) 0.0)
        (unit_set_current_vitality (p0) (max 1 (* (get_digital_forward_throttle 3) 75)) 75) )
;;ensure this doesnt kill them, by picking the higher number between 1, and the restored number * 75)
)
```

Make sure if you're using OS, you remove the '**(set raw_data (* 100 raw_data))**' in the load data, and if you are not using OS, you remove every **OS ADDITION** from the script before compiling.

Kirby4President2013

Kirby_422