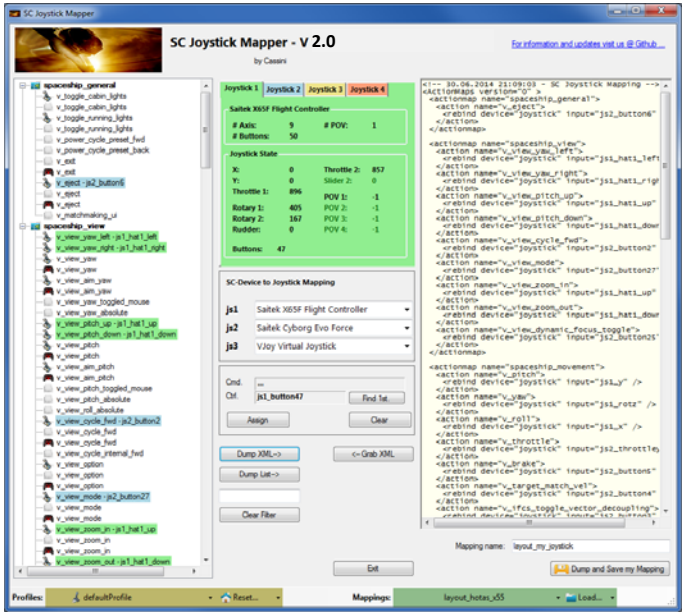
SC Joystick Mapper
Quick Reference Guide V 2.0

20140630 – Cassini

ChangeLog: see ReadMe.txt



Disclaimer:
Usual stuff – no warranty whatsoever..
Freeware – made for the SC community
Hope it helps and does not suck.
Have fun in the verse …

Updating from V 1.x to V 2.0:

- As the action list taken from the game assets (GameData.pak)
  You have to manually remove the ‘defaultProfile.xml’ file if it exists in the same folder as the program file.
  If the program finds it there it is taken before the one from the game (which is may be not what you wanted)

- You may however use this priority for any purpose
  i.e. place a defaultProfile.xml file in the program directory and it will be taken as action list

Workflow

- Connect the joystick devices to the PC
- Start from scratch or load an existing map from a file
- Make or refine mappings
- Save the new map to an XML file
- Use it in the game: e.g. pp_rebindkeys C:\maps\layout_my_joystick
- V 2.0: You may load and save the map directly from your game folders so next time you just use pp_rebindkeys layout_my_joystick

Note: the predefined actions are the ones found in the AC game default profile – it is likely that some of them will not work at all as the game is not finished. There is no proper description for which one does what – you may get help in SC Forums.

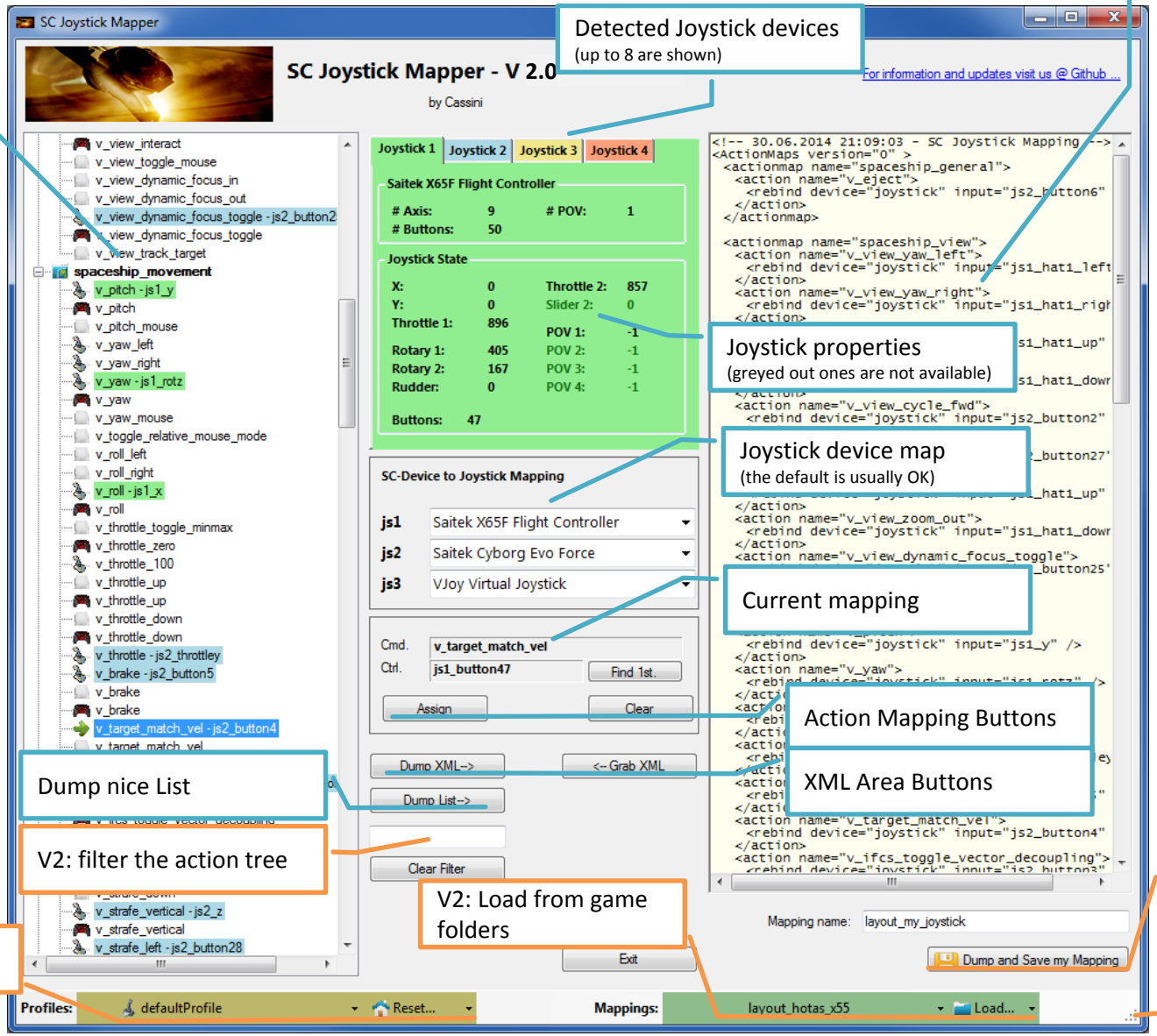
As I had my issues with missiles here a finding..

To reallocate the missile fire command you should map the following 2 actions to the same joystick button:

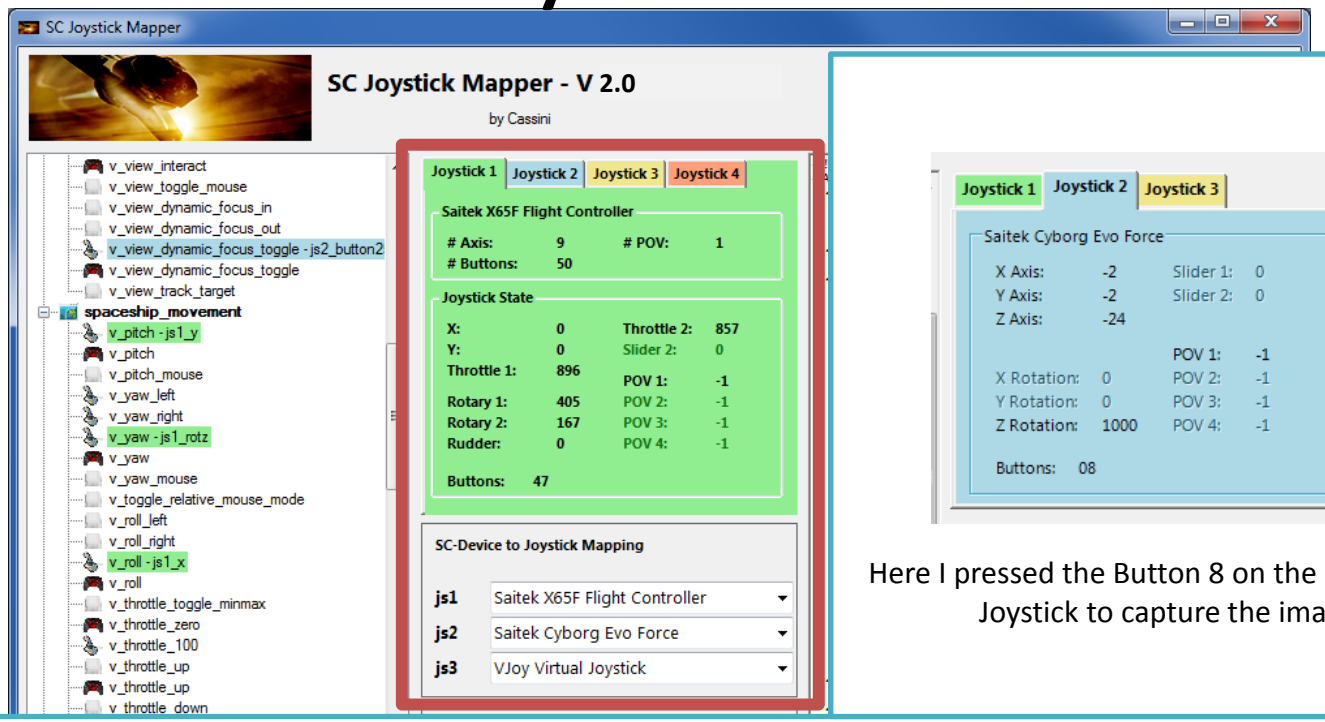
- *v_target_missile_lock_selected*
- *v_weapon_launch_missile*

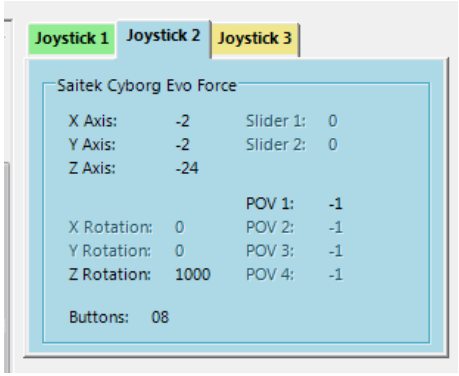
BTW: if you copy e.g. “pp_rebindkeys C:\maps\layout_my_joystick” from notepad you may use Ctrl-V to paste it in-game into the console – saves you some typing…

The GUI …


Action tree and mappings
XML dump of the mappings used
Detected Joystick devices
(up to 8 are shown)
SC Joystick Mapper
SC Joystick Mapper - V 2.0
by Cassini
For information and updates visit us @ Github ...
Joystick 1
Joystick 2
Joystick 3
Joystick 4
Saitek X65F Flight Controller
Axis: 9
POV: 1
Buttons: 50
Joystick State
X: 0
Y: 0
Throttle 1: 896
Rotary 1: 405
Rotary 2: 167
Rudder: 0
Throttle 2: 857
Slider 2: 0
POV 1: -1
POV 2: -1
POV 3: -1
POV 4: -1
Buttons: 47
Joystick properties
(greyed out ones are not available)
SC-Device to Joystick Mapping
js1 Saitek X65F Flight Controller
js2 Saitek Cyborg Evo Force
js3 VJoy Virtual Joystick
Joystick device map
(the default is usually OK)
Cmd. v_target_match_vel
Ctrl. js1_button47
Find 1st.
Assign
Clear
Current mapping
Action Mapping Buttons
Dump XML-->
<-- Grab XML
Dump List-->
XML Area Buttons
Dump nice List
V2: filter the action tree
Clear Filter
V2: Load from game folders
Exit
Mapping name: layout_my_joystick
Dump and Save my Mapping
V2: Save into game folders
V2: New Reset with options
V2: Resize the window
Profiles: defaultProfile
Reset...
Mappings: layout_hotas_x55
Load...
v_view_interact
v_view_toggle_mouse
v_view_dynamic_focus_in
v_view_dynamic_focus_out
v_view_dynamic_focus_toggle - js2_button2
v_view_dynamic_focus_toggle
v_view_track_target
spaceship_movement
v_pitch - js1_y
v_pitch
v_pitch_mouse
v_yaw_left
v_yaw_right
v_yaw - js1_rotz
v_yaw
v_yaw_mouse
v_toggle_relative_mouse_mode
v_roll_left
v_roll_right
v_roll - js1_x
v_roll
v_throttle_toggle_minmax
v_throttle_zero
v_throttle_100
v_throttle_up
v_throttle_up
v_throttle_down
v_throttle_down
v_throttle - js2_throttley
v_brake - js2_button5
v_brake
v_brake
v_target_match_vel - js2_button4
v_strafe_vertical - js2_z
v_strafe_vertical
v_strafe_left - js2_button28

The Joystick Area…





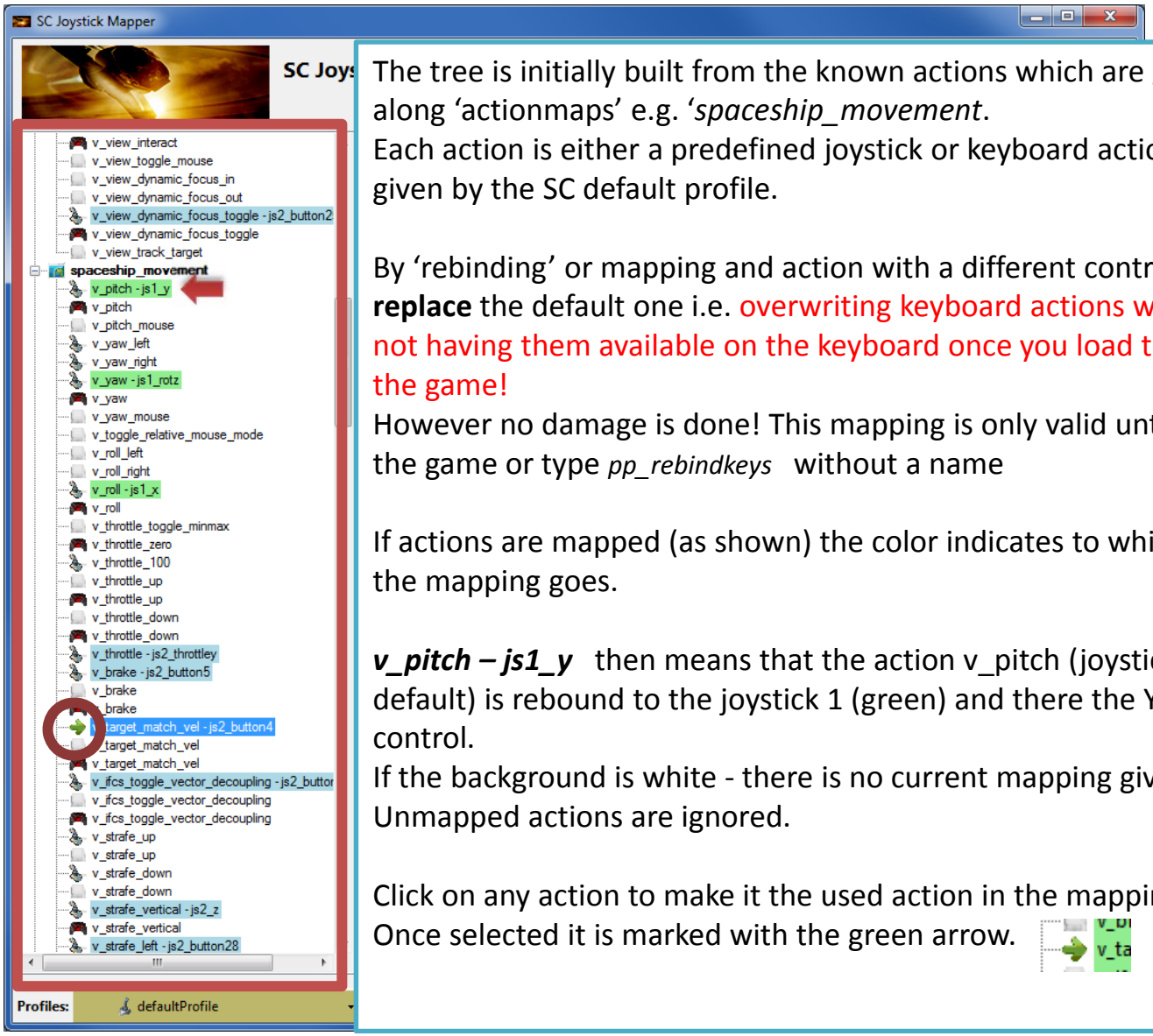
Here I pressed the Button 8 on the Cyborg Evo Joystick to capture the image

The tabs represent the joystick devices found connected to the PC also the number 1..8 shows the order the PC reports them which is crucial to the mapping as this will result in the js_1, js_2 .. Names used to build the command name.
The elements are the ones the joystick seems to support – greyed ones are not available for this device.

The *SC-Device to Joystick Mapping* can be used if the default assignment “Joystick 1 -> js_1” does not match what the CryEngine is using. – Usually the default should work. You may only remap js1..js3 - 4..8 will remain as detected.

Just hit any button, Axis and see how things are changing.

The Action Tree …



The tree is initially built from the known actions which are grouped along 'actionmaps' e.g. '*spaceship_movement*.
Each action is either a predefined joystick or keyboard action – this is given by the SC default profile.

By 'rebinding' or mapping and action with a different controls one does **replace** the default one i.e. overwriting keyboard actions will result in not having them available on the keyboard once you load the map in the game!
However no damage is done! This mapping is only valid until you exit the game or type *pp_rebindkeys* without a name

If actions are mapped (as shown) the color indicates to which joystick the mapping goes.

v_pitch – js1_y then means that the action v_pitch (joystick per default) is rebound to the joystick 1 (green) and there the Y-axis control.
If the background is white - there is no current mapping given.
Unmapped actions are ignored.

Click on any action to make it the used action in the mapping area.
Once selected it is marked with the green arrow.

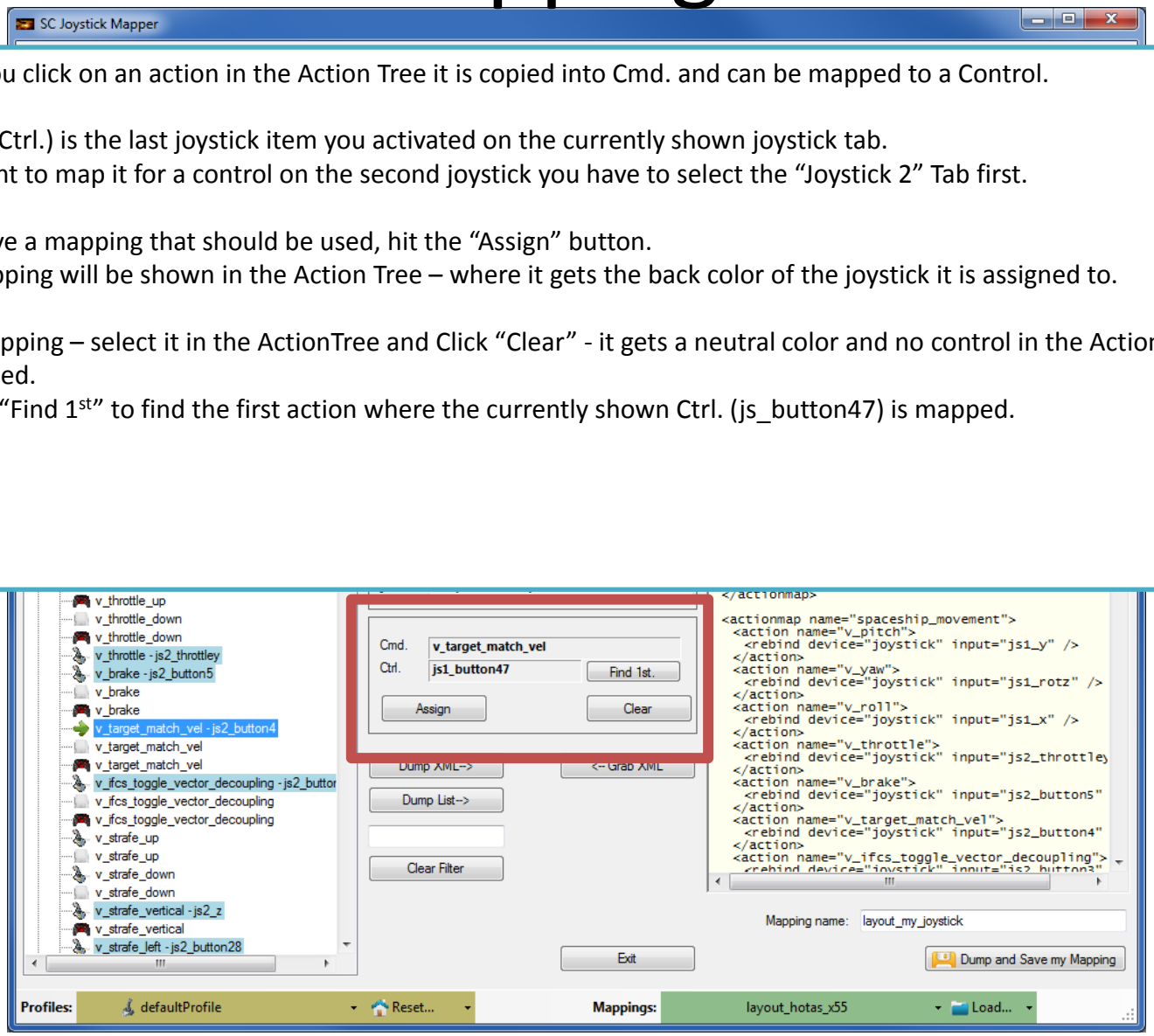
The Mapping Area…

Whenever you click on an action in the Action Tree it is copied into Cmd. and can be mapped to a Control.

The Control (Ctrl.) is the last joystick item you activated on the currently shown joystick tab.
I.e. if you want to map it for a control on the second joystick you have to select the “Joystick 2” Tab first.

Once you have a mapping that should be used, hit the “Assign” button.
The new mapping will be shown in the Action Tree – where it gets the back color of the joystick it is assigned to.

To clear a mapping – select it in the ActionTree and Click “Clear” - it gets a neutral color and no control in the ActionTree – it is now unmapped.
You may use “Find 1st” to find the first action where the currently shown Ctrl. (js_button47) is mapped.



The XML Area…

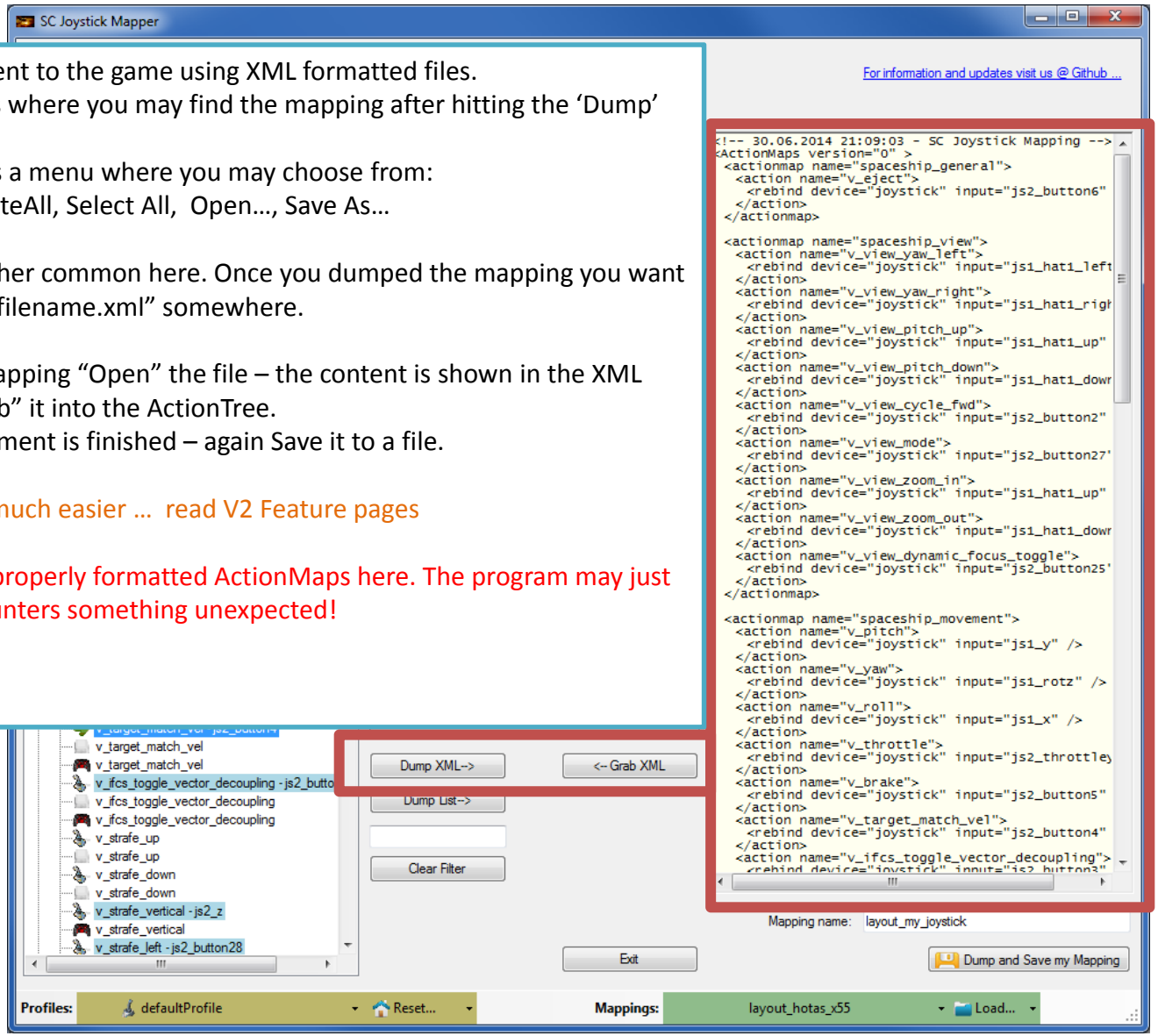
Mappings are sent to the game using XML formatted files.
The XML Area is where you may find the mapping after hitting the ‘Dump’ button.
Rightclick opens a menu where you may choose from:
Copy, Paste, PasteAll, Select All, Open…, Save As…

The usage is rather common here. Once you dumped the mapping you want to “Save” it as “filename.xml” somewhere.

To refine any mapping “Open” the file – the content is shown in the XML Area, then “Grab” it into the ActionTree.
Once the refinement is finished – again Save it to a file.

Load and Save much easier … read V2 Feature pages

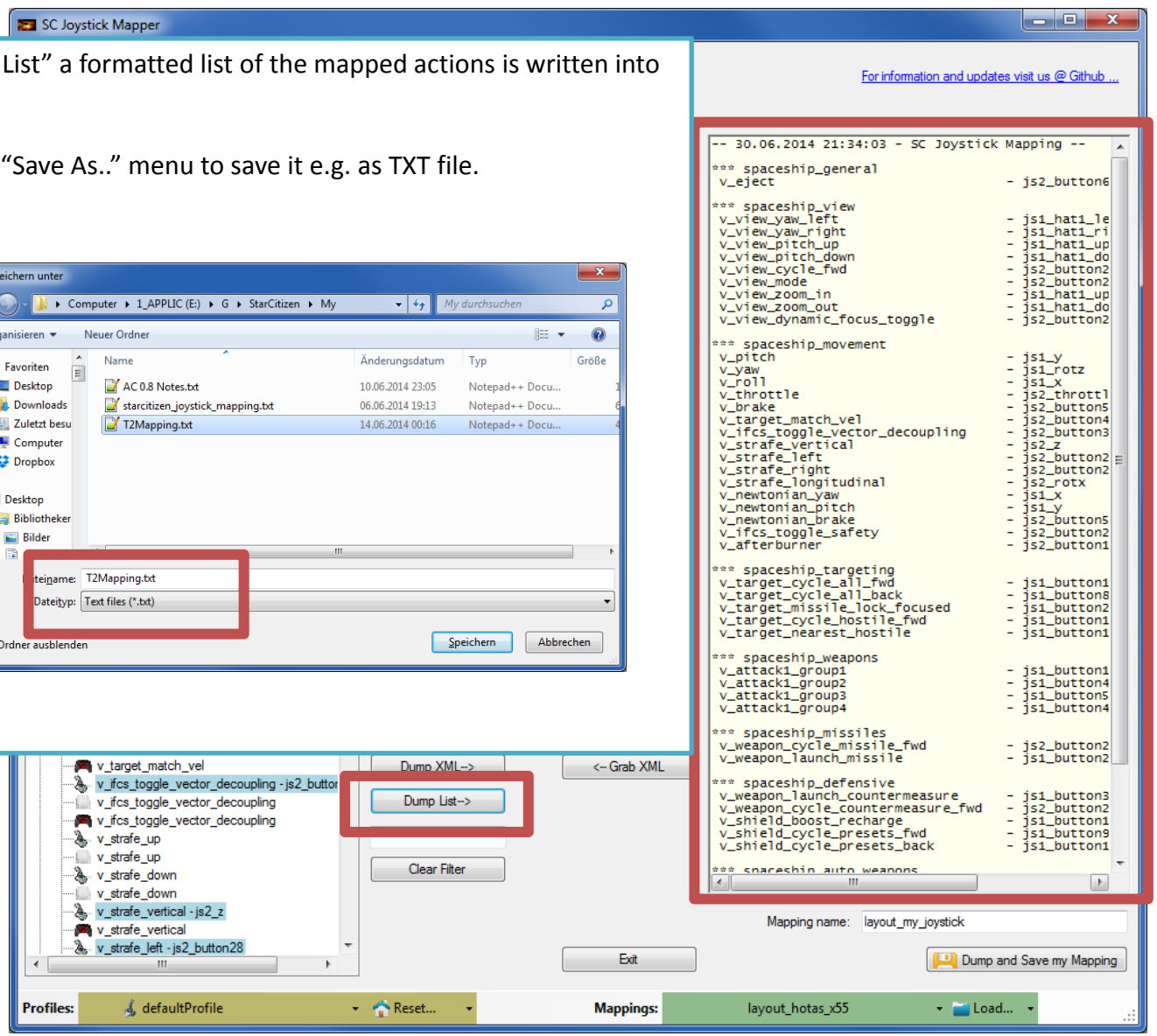
Note: only use properly formatted ActionMaps here. The program may just break if it encounters something unexpected!



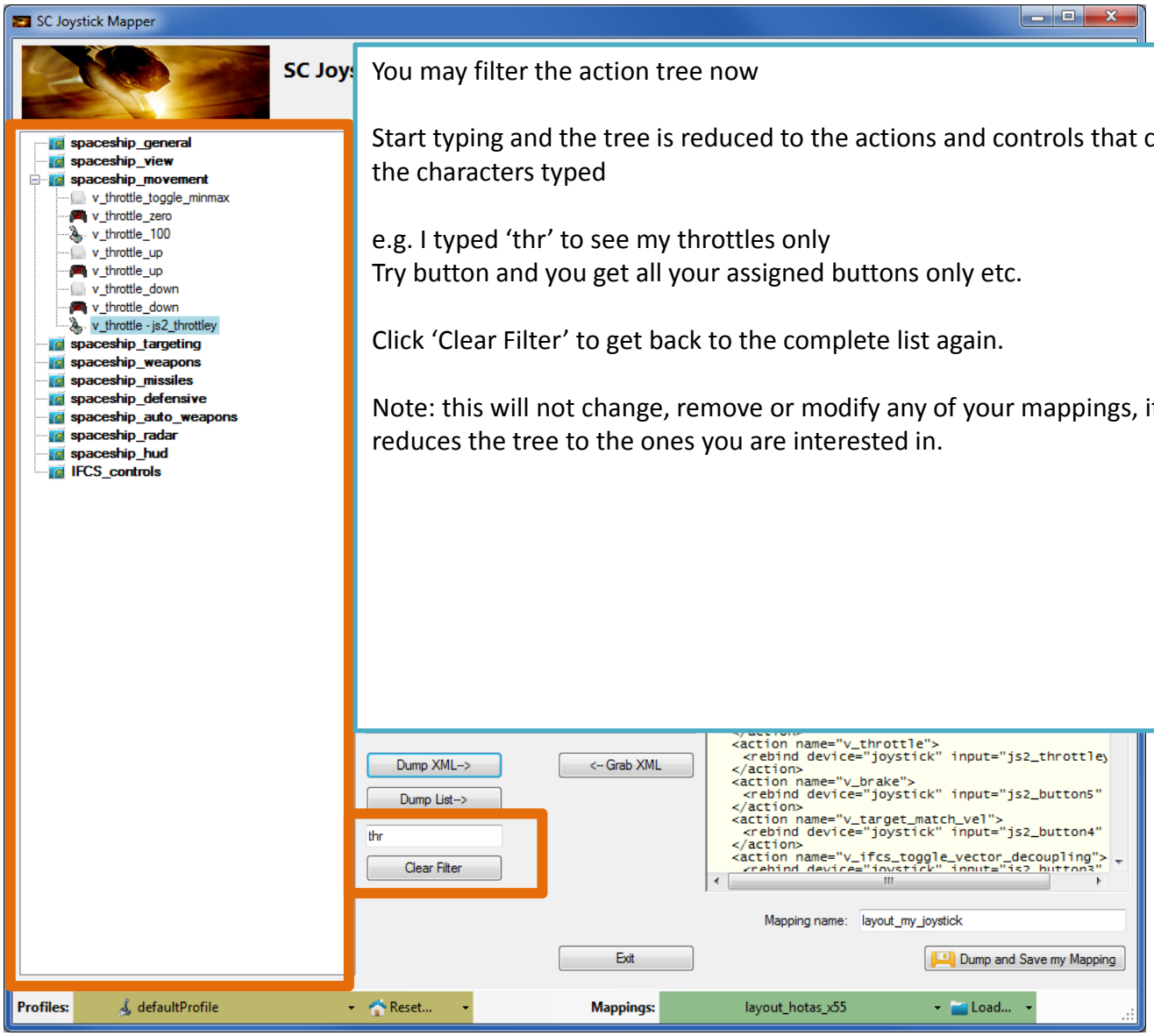
The XML Area…

If you hit “Dump List” a formatted list of the mapped actions is written into the XML area.

You may use the “Save As..” menu to save it e.g. as TXT file.



V2 – Features - 1



You may filter the action tree now

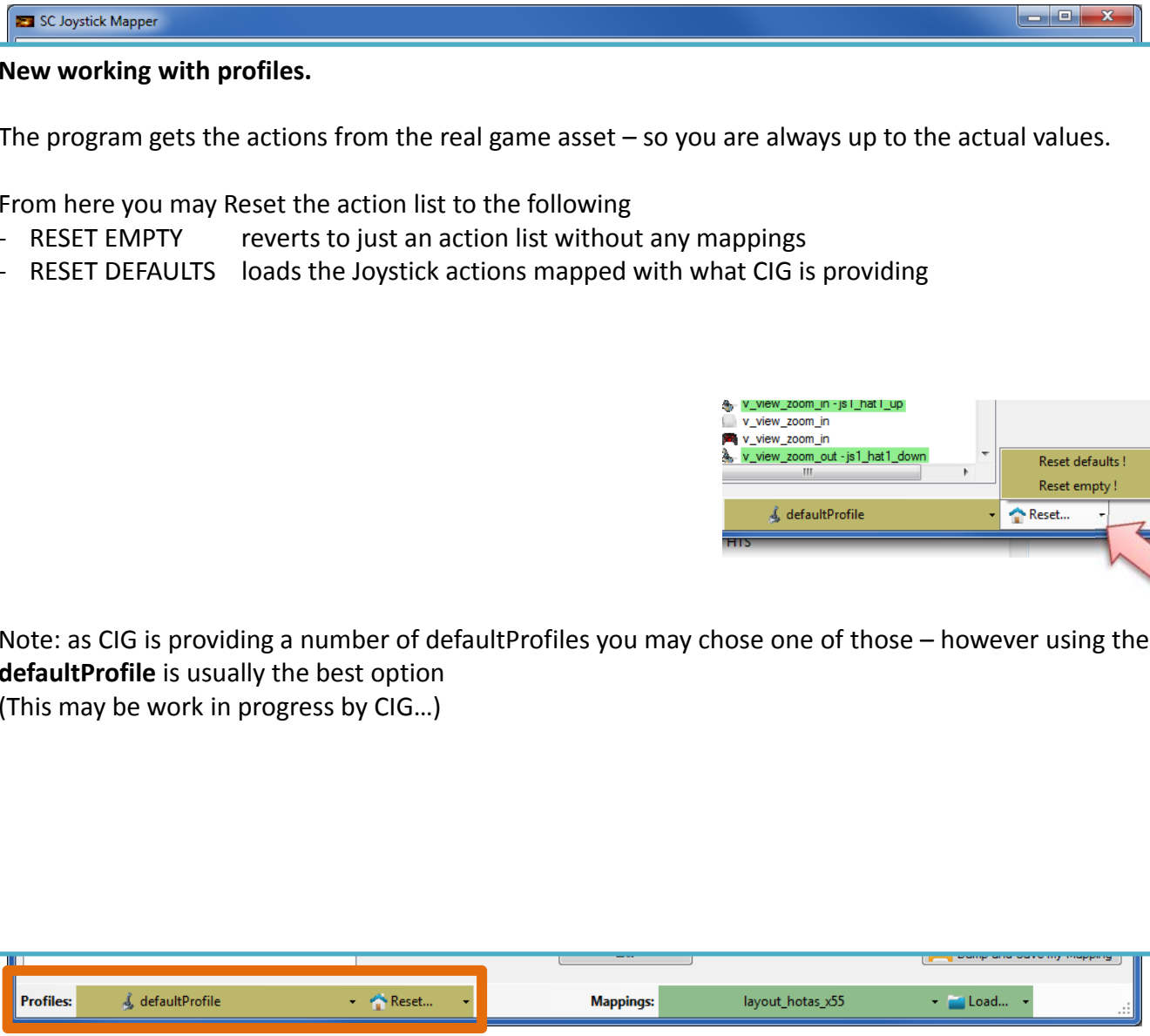
Start typing and the tree is reduced to the actions and controls that contain the characters typed

e.g. I typed ‘thr’ to see my throttles only
Try button and you get all your assigned buttons only etc.

Click ‘Clear Filter’ to get back to the complete list again.

Note: this will not change, remove or modify any of your mappings, it just reduces the tree to the ones you are interested in.

V2 – Features - 2

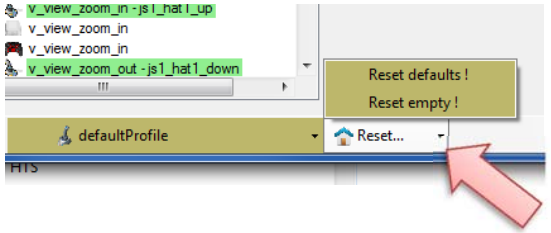


New working with profiles.

The program gets the actions from the real game asset – so you are always up to the actual values.

From here you may Reset the action list to the following

- RESET EMPTY reverts to just an action list without any mappings
- RESET DEFAULTS loads the Joystick actions mapped with what CIG is providing



Note: as CIG is providing a number of defaultProfiles you may chose one of those – however using the **defaultProfile** is usually the best option
(This may be work in progress by CIG…)

V2 – Features - 3

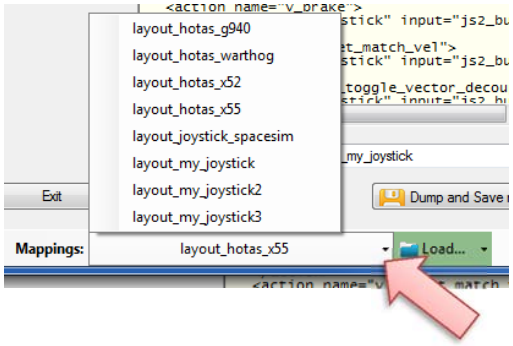
New working with actionmaps (Maps, Mapping etc..)

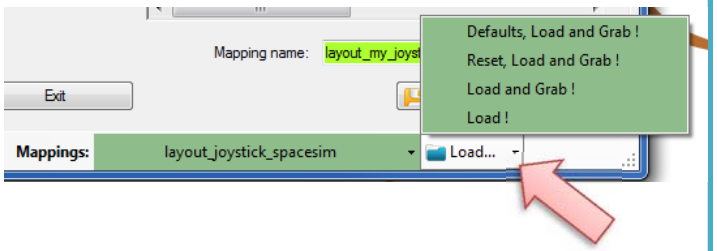
The program gets the actionsmaps from the real game asset – so you are always up to the actual values.
(…\StarCitizen\CitizenClient\Data\Controls\Mappings)

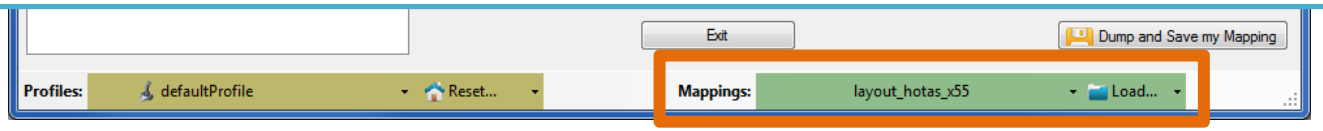
From here you may first chose a map, then ‘Load’ the actionmap – this will overwrite you XML window in any case

- LOAD loads the map into the XML window only
- LOAD and GRAB loads the map into the XML window and clicks Grab i.e. merges the existing mapping with the one loaded
- RESET, LOAD and GRAB first Reset (empty) the action list (all mappings cleared) then it loads and grabs the new map
- DEFAULT, LOAD and GRAB first Reset (defaults) the action list then it loads and grabs the new map and merges them with the defaults

See last page for some common workflows
And how to handle them easily







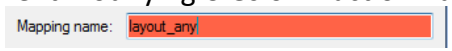
V2 – Features - 4

New working with your own actionmaps

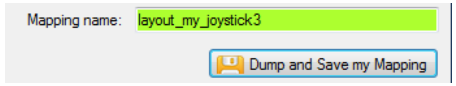
The program not only gets the actionsmaps from the real game asset – but also can save your maps there.
(…\StarCitizen\CitizenClient\Data\Controls\Mappings)

1. Type a name (limitations see note)
2. Hit the button – it will then Dump and Save your map into the game folder (well asking you to overwrite it if it exists)

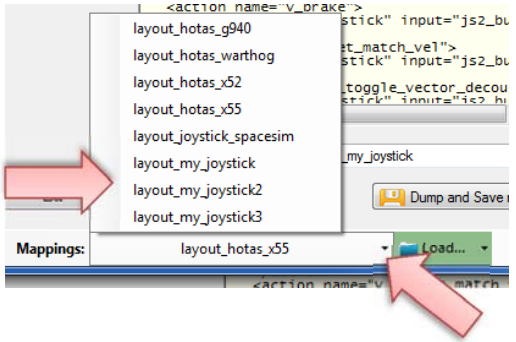
NOTE: your map name has always to start with 'layout_my_' to prevent modifying CIGs own actionmaps
Lowercase only, no spaces, tabs allowed else you see the red flag ..



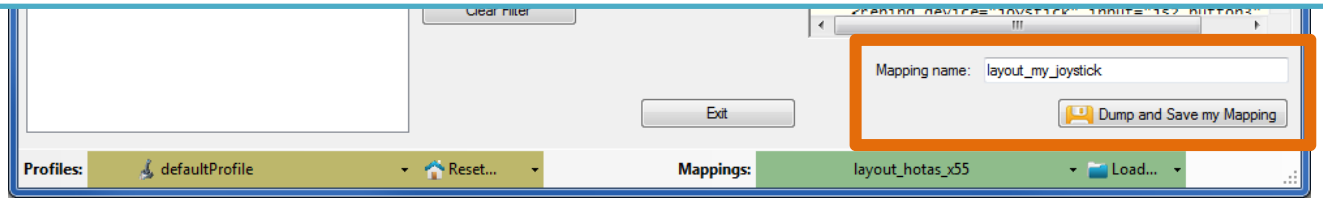
A successful Save will show the green flag



Your own maps will then show up like the game provided maps
`pp_rebindkeys layout_my_joystick` should load it into the game



Note: For your convenience each Save also makes a copy of into your personal "My Documents\SCJMapper" folder – no work is lost if there is an update that cleans the Mappings folder.



SCJMapper V 2 – Common Workflows