

Converting a 5CP map into a PC5 Map

Step 0: Preparation

Get a working version of the map you want to convert

If you have a .vmf file of it, this should be a given

If you only have a .bsp you need to decompile it and fix up any decompilation errors, mainly area portals

Rename it to `pc5_yourmapname`

Step 1: Rename Stuff

Next up, all control point related objects need to be adjusted.

All **team_control_point** entities and all **cap_point_base** models need to be renamed and **trigger_capture_areas** need their associated control point field updated.

Location	Type	Name / Associated Control Point Field
RED's Last	trigger_capture_area	cp_red1
	team_control_point	cp_red1
	cap_point_base.mdl	cp_red1_prop
RED's 2nd	trigger_capture_area	cp_red2
	team_control_point	cp_red2zw
	cap_point_base.mdl	cp_red2_prop
Mid	trigger_capture_area	cp_mid
	team_control_point	cp_mid
	cap_point_base.mdl	cp_mid_prop
BLU's 2nd	trigger_capture_area	cp_blu2
	team_control_point	cp_blu2
	cap_point_base.mdl	cp_blu2_prop

BLU's 1st	z	cp_blu1
	team_control_point	cp_blu1
	cap_point_base.mdl	cp_blu1_prop

Step 2: Update Stuff

If the **info_observer_point** entities in your map have been set to associate with a control point, make sure the field is updated to reflect the new **team_control_point** entity names we set in step 2.

Same goes for **info_player_teamspawn**, update the name of the associated control point as shown in step 2 in order for them to work properly.

Step 3: Copy-Paste Stuff

Open the newest **pc_winlogic.vmf** file in Hammer.

Disable all visgroups but the "pc_logic" one and copy all elements into the map you want to convert. Place them somewhere easy to find, ideally around the central control point.

Step 4: Delete Stuff

Find any existing **tcp_master** and **tf_round_timer** entities and check their output tabs. If there are no *working* outputs, delete them.

(In some maps, you may need to remove additional entities if they exist, namely **math_counter**, **game_round_win**, **game_end**, **game_text**)

If a **tf_gamerules** entity already exists, check it's input tab if it changes *respawnwavetimes*. If so, go to the entity that the input originates from (in this case usually a **logic_auto**) and change the output so that the target entity field says: "**pc_gamerules**".

Then delete the old **tf_gamerules** entity as well as any broken outputs of the **logic_auto**, if you have one.

Step 5: More Copy-Pasting

(enable all visgroups)

For each **trigger_capture_area** do the following:

- Update the associated control point field (if not done in step 2)
- Copy all outputs from the outputs tab in the **pc_winlogic_x.vmf** and paste them into the output tab for the corresponding **trigger_capture_area** on your map

For example:

1. Go to the outputs tab of the central control points **trigger_capture_area** in `pc5_winlogic.vmf`
2. Copy all the outputs
3. Go to the outputs tab of the central control points **trigger_capture_area** in `yourmapname.vmf`
4. Paste the copied outputs

There will be a bunch of outputs that don't longer work because of the changes we did in step 2. Delete those but keep any working outputs (unless they are duplicates).

Step 6: Compiling & Testing

The map is technically done, compile your map, go in-game, generate a navmesh and play against bots for a while to ensure everything is working correctly.

DONE!