

## Documentation

# Controlled Spawn & Waypoints Randomizr (CSWR)

Last update: v7.0

[What is this](#)

[Video Tutorials](#)

[If you need an SQF editor](#)

[Run the script for a first look](#)

[Install the script in my mission ★★★](#)

[Set the sides to spawn](#)

[Set who the enemy is](#)

[Set the weaponry and the side size](#)

[Spawn-points for side](#)

[Spawn-points > Spawn Delay methods](#)

[Destinations \(waypoints\)](#)

[Destinations > Hold-marker](#)

[Destinations > Watch-marker](#)

[Destinations > Occupy-marker](#)

[Behavior for each group](#)

[Customizing the loadouts](#)

[Electronic Warfare Resources](#)

[Debug Mode: investigating what is happening](#)

[Fixing: trying to edit the unit's loadout](#)

[Fixing: making CSWR lighter for servers](#)

[Fixing: helicopters over ship floors \(NOT AVAILABLE, WIP\)](#)

[Fixing: units are not in a crouch when inside a ruin](#)

[How CSWR works technically \(advanced knowledge\)](#)

[Contribute to the CSWR script](#)

[Author](#)

## What is this

**CSWR** is an *Arma 3* script that allows the Mission Editor (you) to spawn AI units and vehicles (ground, water, and air, including air paratroop for soldiers and vehicles) and makes those units move randomly to waypoints forever in life, where spawn-points and waypoints are easily pre-defined by the Mission Editor through *Eden* marker's positions. CSWR accepts side loadout customization, including ethnicity, voice language, and additional customizations for snipers, marines, heavy ground crewmen, and paratroops. CSWR almost doesn't change any original *Arma* AI behavior, saving server performance and *Arma 3* integrity.

### Creation concept:

Bring life to the mission through non-stop units' movements with some level of unpredictability without losing control of server performance and what AI units can do.

### What to expect:

- No dependencies from other mods or scripts;
- Run flawlessly for multiplayer and single-player missions;
- Manually set which markers the side can use as spawn-points;
- Create unlimited and different types of spawn-points:
  - Infantry Spawn-points;
  - Marines Spawn-points; **NEW**
  - Ground vehicle spawn-points;
  - Nautical and amphibious spawn-points; **NEW**
  - Helicopter spawn-points;
  - Air Paratroop for vehicle spawn-points;
  - Air Paratroop for people spawn-points;
- Spawn-points can be triggered by:
  - Mission starts;
  - Trigger activation;
  - Timer;
  - Target status;
- There is no re-spawn. Death is death for units and vehicles spawned by CSWR;
- Create unlimited and different types of destinations:
  - Move Restrict;
  - Move Public;
  - Move Any;

- Nautical Move Restrict; **NEW**
  - Nautical Move Public; **NEW**
  - Nautical Move Any; **NEW**
  - Move Watch;
  - Move Hold;
  - Move Occupy;
- Set which vehicles a side will use (including helicopters and boats);
- Set which units/weapons (regardless of the faction or side) a side will use;
- Set how large is each group-type of a side;
- Customize the loadout of different roles of the same side:
  - Infantry;
  - Snipers;
  - Paratroopers;
  - Heavy Ground Crewmen;
  - Marines; **NEW**
- Set those segments of a side can use Night-vision-Goggles:
  - Only Infantry;
  - Only Snipers;
  - Only Paratroops;
  - Only Marines; **NEW**
  - All of them;
  - No one;
  - Set if a side has Flashlights as an alternative for Night-visions;
- Deeper customization:
  - Side ethnicity; **NEW**
    - Automatic faces selection;
    - Automatic language selection;
  - Language manual customization; **NEW**
- Set what behavior each group and vehicle start the game:
  - Safe;
  - Aware;
  - Combat;
  - Stealth;
  - Chaos;
- Set if all vehicles and units spawned by CSWR should be editable by Zeus;
- Set if the CSWR should wait for another script load first on the server;
- Debugging: friendly error handling;

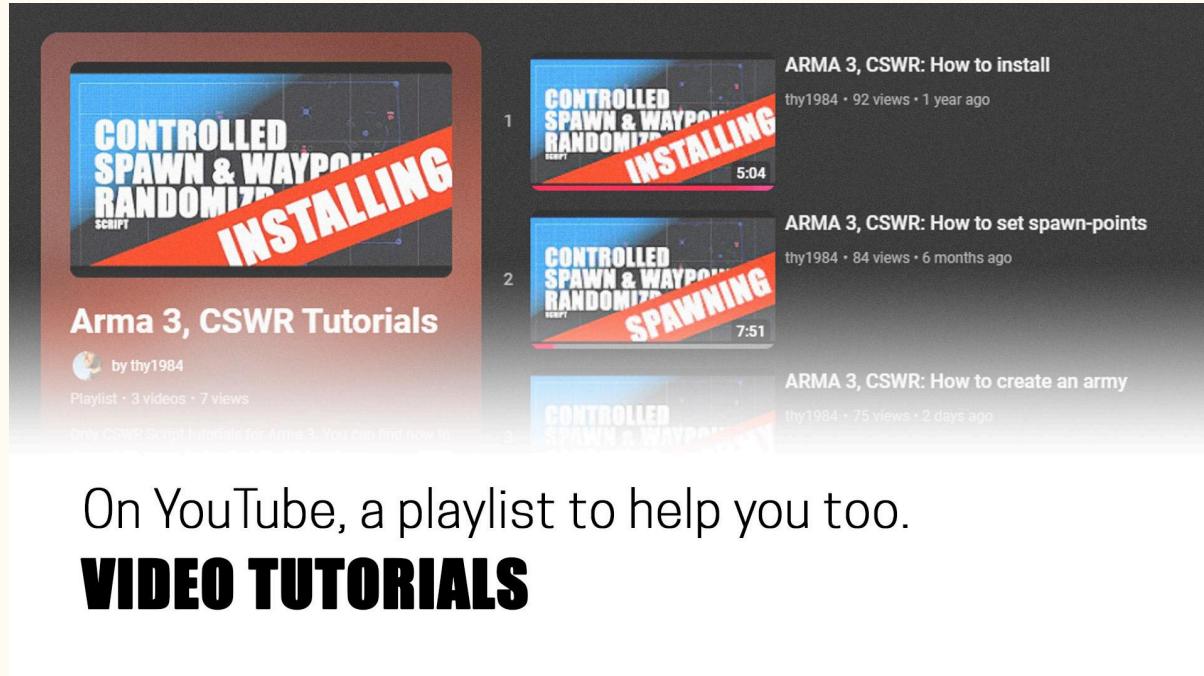
- Debugging: hint monitor to control some AI numbers;
- Debugging: full documentation available.

**CSWR supports content from:**

- *Arma 3*;
- *Expansion Apex*;
- *DLC Malden*;
- *DLC Laws of War*;
- *DLC Marksmen*;
- *DLC Helicopters*;
- *DLC Tanks*;
- *DLC Contact*;
- *DLC Zeus*;
- *DLC Karts*;
- *CDLC Western Sahara*;
- *CDLC Reaction Forces*;
- *CDLC Expeditionary Forces*;
- *CDLC Global Mobilization*;
- *CDLC CSLA Iron Curtain*;
- *CDLC Spearhead 1944*;
- *CDLC S.O.G. Prairie Fire*;
- Mod *RHS*;
- Mod *CUP*;
- And probably whatever you want...

## Video Tutorials

<https://www.youtube.com/playlist?list=PL9C3CUvV0NhM0HspcB9ajPNBj9ILy3ozN>



On YouTube, a playlist to help you too.

## **VIDEO TUTORIALS**

## If you need an SQF editor

Sure, I'm using *Visual Studio Code* with this customs specific for *Arma 3*:

<https://forums.bohemia.net/forums/topic/239960-vs-code-tutorial-how-to-config-vs-code-for-arma-3-2023/>

If you need something simpler:

<https://notepad-plus-plus.org/>, install it and, when you open some script file, go to Notepad++ main menu, “Language” and select “C” as file language. That's it.

## Run the script for a first look

1. Go to <https://steamcommunity.com/sharedfiles/filedetails/?id=2740912514>
2. Subscribe and wait for Steam to download it;
3. Open *Arma 3*, go to Multiplayer > Server browser > Host server > click Host Server button;
4. Select “Stratis” map and, after that, select “CSWR (...);
5. Play.

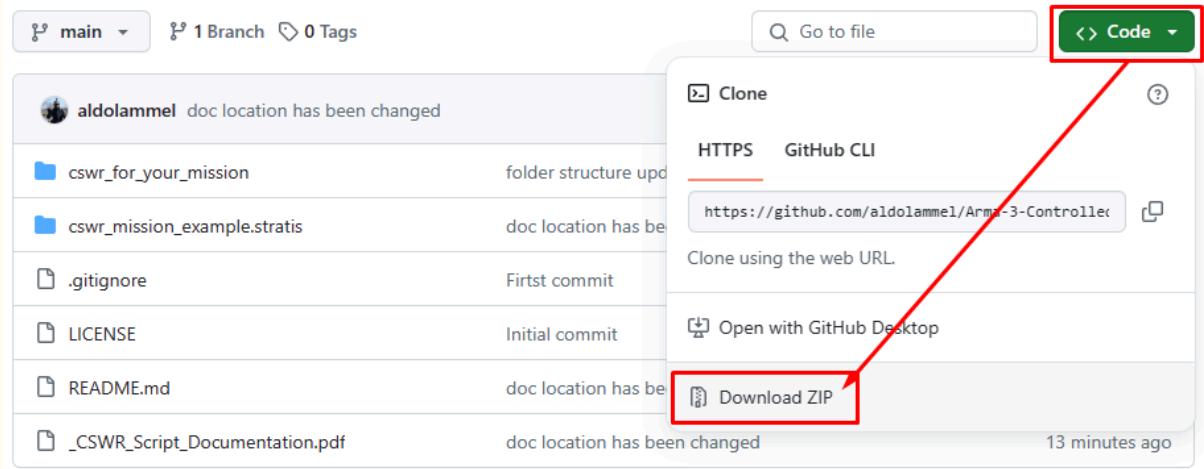
**Important:** if you want to apply this script in your own missions, [check this out](#).

## Install the script in my mission ★★★

1. Go to:

<https://github.com/aldolammel/Arma-3-Controlled-Spawn-And-Waypoints-Randomizr-Script>

2. Download the zip and open it;



3. In zip, get in the "cswr\_for\_your\_mission" folder;
4. Copy all "cswr\_for\_your\_mission" content to your mission folder root:  
DRIVE:\Users\yourName\Documents\Arma 3\yourProfile\missions\yourMission\
5. **WARNING:** in the mission folder root, if you already have a "description.ext" file, don't replace the current file with the new one. Just add the code below in the current file:

```
class cfgFunctions
{
    // CSWR: CONTROLLED SPAWN AND WAYPOINTS RANDOMIZR
    #include "CSWRandomizr\THY_CSWR_functions.hpp"
};
```

6. **WARNING:** still in the mission folder, if you already have an "init.sqf" file, don't replace the current file with the new one. Just add the code below in the current file:

```
// CSWR > HIDE THE SCRIPT MARKERS:
if (!CSWR_isOn || !CSWR_isOnDebug) then {if !isMultiplayer then {uiSleep 1};{private
    _mkr = toUpper _x; private _mkrChecking = _mkr splitString CSWR_spacer; if
    (_mkrChecking find CSWR_prefix isEqualTo -1) then {_x setMarkerAlpha 0}}
    forEach allMapMarkers};
```

7. Now, [let's create your custom side\(s\)!](#)

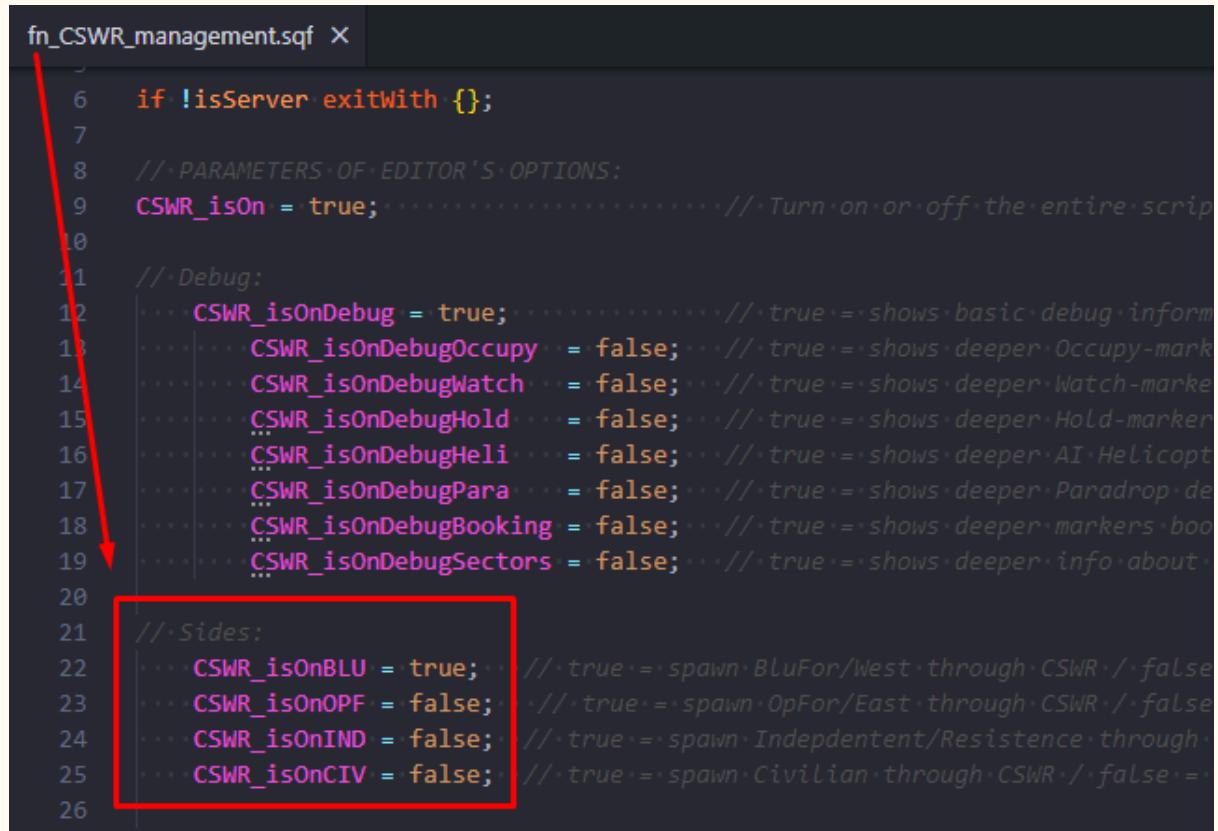


## Set the sides to spawn

In *Arma 3* there are 4 sides available: Blufor, Opfor, Independent, and Civilian. All of them are built by groups that are composed of units (soldiers/people). With CSWR you will spawn dynamically one or more sides. It's up to you. Let's do it, by opening the file:

\CSWRandomizer\fn\_CSWR\_management.sqf

Set as *true* the side you want to spawn through CSWR:

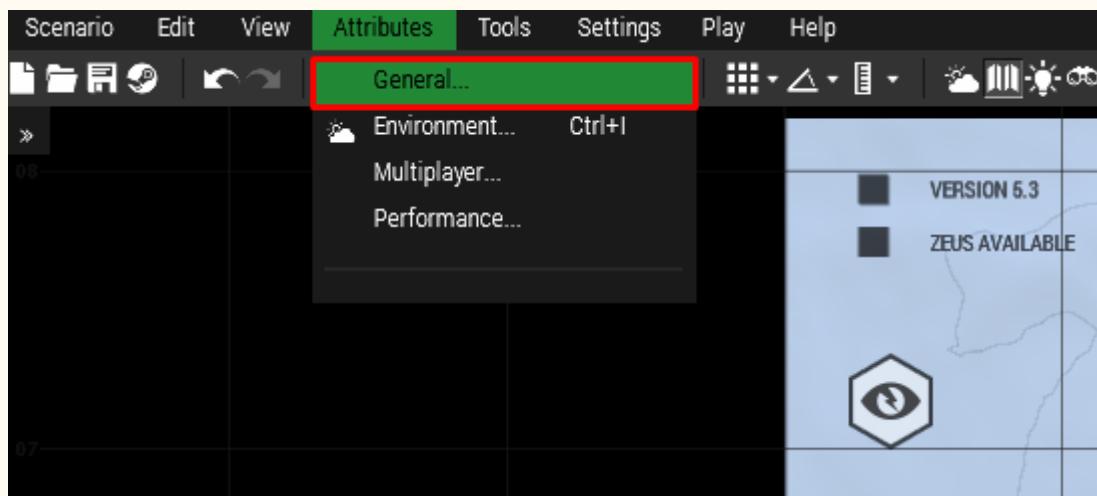


```
fn_CSWR_management.sqf X
1
2
3
4
5
6   if !isServer exitWith {};
7
8   //PARAMETERS OF EDITOR'S OPTIONS:
9   CSWR.isOn = true; //Turn on or off the entire script
10
11  //Debug:
12  CSWR.isOnDebug = true; //true == shows basic debug info
13  CSWR.isOnDebugOccupy = false; //true == shows deeper Occupy-marker
14  CSWR.isOnDebugWatch = false; //true == shows deeper Watch-marker
15  CSWR.isOnDebugHold = false; //true == shows deeper Hold-marker
16  CSWR.isOnDebugHeli = false; //true == shows deeper AI Helicopter
17  CSWR.isOnDebugPara = false; //true == shows deeper Paradrop/de
18  CSWR.isOnDebugBooking = false; //true == shows deeper markers boo
19  CSWR.isOnDebugSectors = false; //true == shows deeper info about
20
21  //Sides:
22  CSWR.isOnBLU = true; //true == spawn Blufor/West through CSWR //false
23  CSWR.isOnOPF = false; //true == spawn OpFor/East through CSWR //false
24  CSWR.isOnIND = false; //true == spawn Independent/Resistance through
25  CSWR.isOnCIV = false; //true == spawn Civilian through CSWR //false
26
```

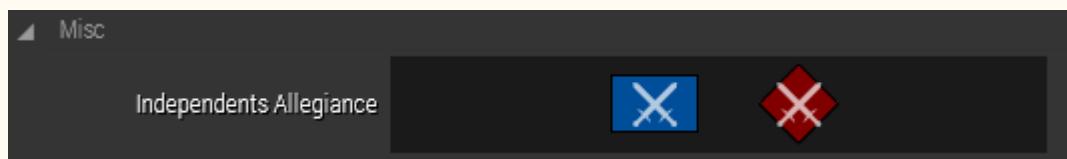
## Set who the enemy is

CSWR works with what you have defined through *Eden Editor*:

1. On *Eden's* main menu, go to Attributes > General;



2. In the “General” window, look for “Misc”:

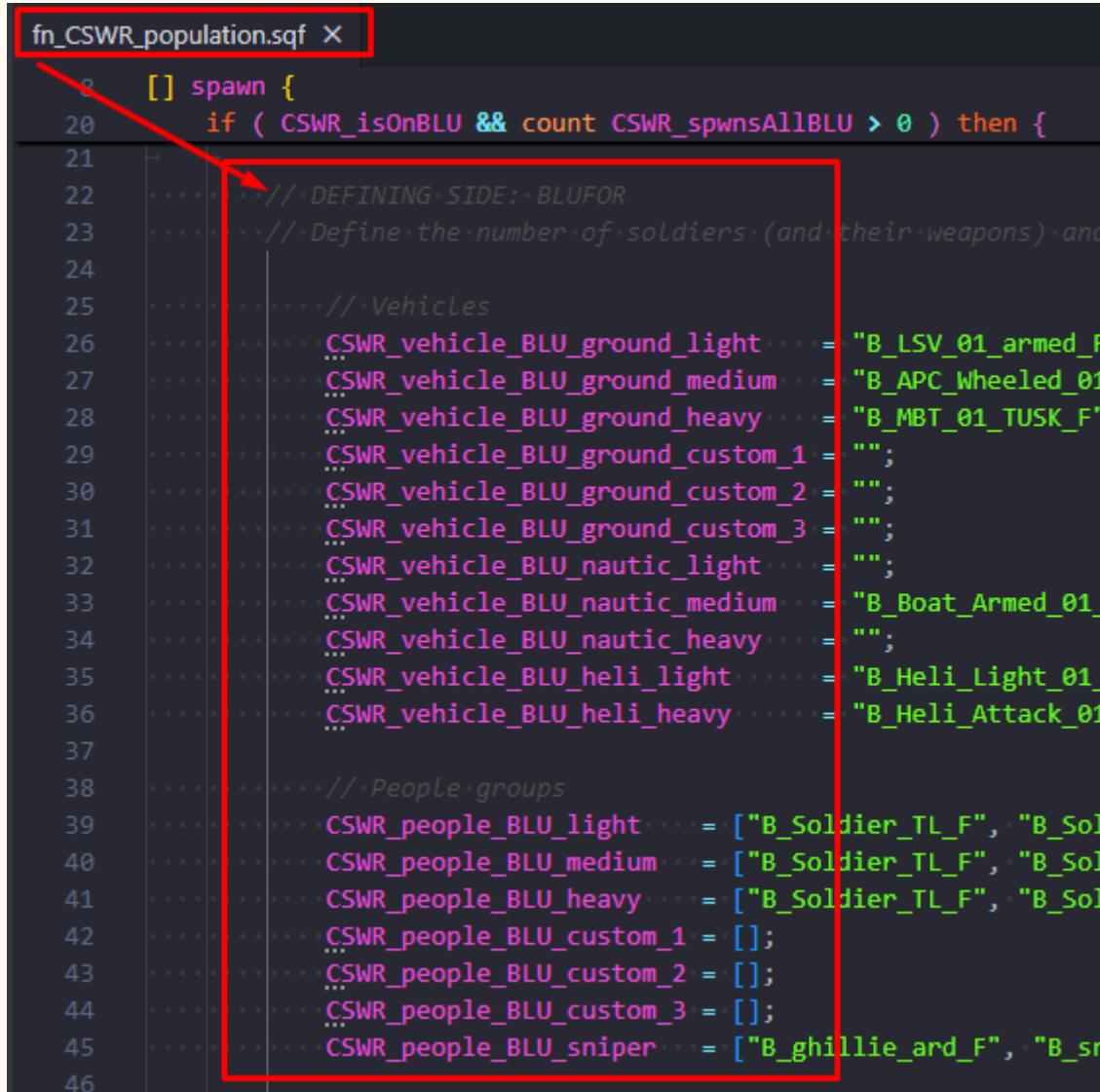


3. As BluFor and OpFor are always enemies of each other, you only need to define who is the Independent enemy, by clicking over the side flag.

**Important:** about Arma 3 logic, Civilian doesn't fear Independent and their allies.

## Set the weaponry and the side size

Now, let's request CSWR to spawn the [side allowed to do that](#). In the [fn\\_CSWR\\_population.sqf](#) file, search the section "DEFINING SIDE: BLUFOR", for example:

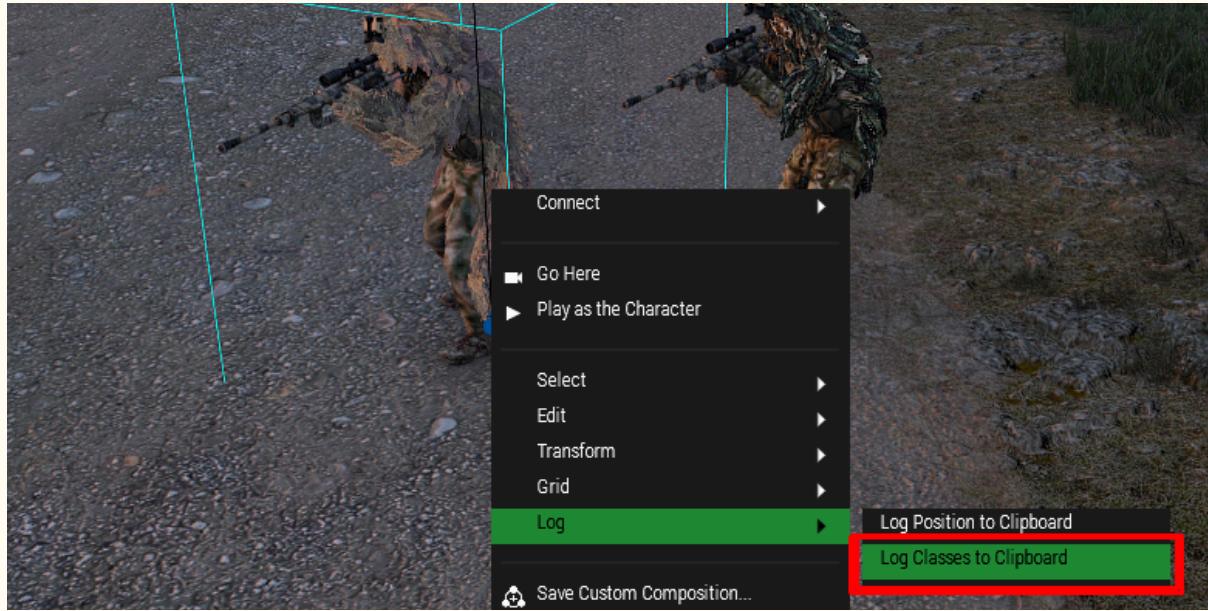


```
fn_CSWR_population.sqf X

18 [ ] spawn {
19     if ( CSWR_isOnBLU && count CSWR_spwnsAllBLU > 0 ) then {
20
21         //DEFINING SIDE: BLUFOR
22         //Define the number of soldiers (and their weapons) and
23         //Vehicles
24         CSWR_vehicle_BLU_ground_light = "B_LSV_01_armed_F"
25         CSWR_vehicle_BLU_ground_medium = "B(APC_Wheeled_01"
26         CSWR_vehicle_BLU_ground_heavy = "B(MBT_01_TUSK_F"
27         CSWR_vehicle_BLU_ground_custom_1 = "";
28         CSWR_vehicle_BLU_ground_custom_2 = "";
29         CSWR_vehicle_BLU_ground_custom_3 = "";
30         CSWR_vehicle_BLU_nautic_light = "";
31         CSWR_vehicle_BLU_nautic_medium = "B_Boat_Armed_01"
32         CSWR_vehicle_BLU_nautic_heavy = "";
33         CSWR_vehicle_BLU_heli_light = "B_Heli_Light_01"
34         CSWR_vehicle_BLU_heli_heavy = "B_Heli_Attack_01"
35
36         //People groups
37         CSWR_people_BLU_light = [ "B_Soldier_TL_F", "B_Sol
38         CSWR_people_BLU_medium = [ "B_Soldier_TL_F", "B_Sol
39         CSWR_people_BLU_heavy = [ "B_Soldier_TL_F", "B_Sol
40         CSWR_people_BLU_custom_1 = [];
41         CSWR_people_BLU_custom_2 = [];
42         CSWR_people_BLU_custom_3 = [];
43         CSWR_people_BLU_sniper = [ "B_ghillie_ard_F", "B_sr
```

### Step 1/3: select the soldiers' weaponry

Crucial: on *Eden*, place somewhere those units you want to use as *BluFor* soldiers. Don't worry about the side, nationality, language, faces, or uniforms. Just select those units that originally have the weaponry you wanna see in CSWR *BluFor* army. You can use mods if you want.

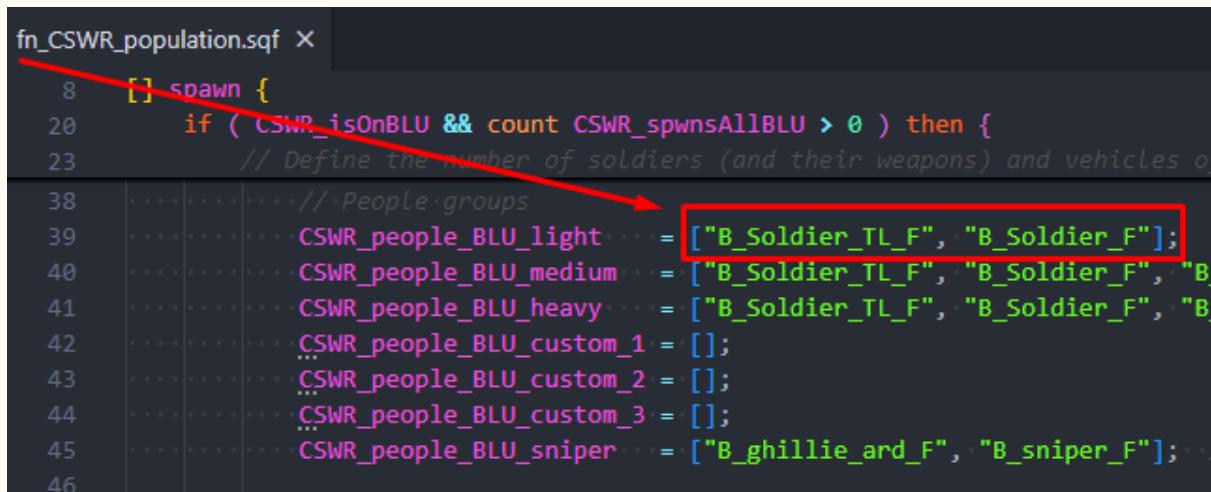


Collecting the classname of a soldier through *Eden Editor*. Next, use CTRL+V [on your script editor](#).

Later you will see: [how to customize your entire army loadouts](#).

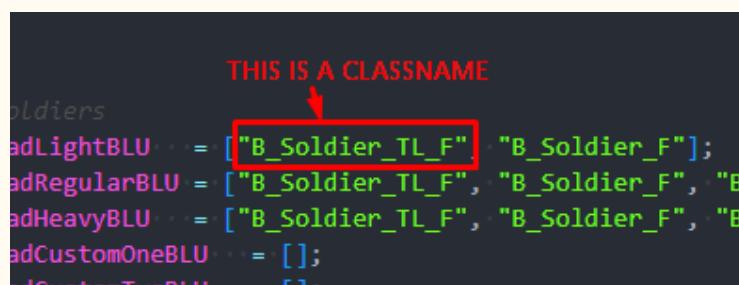
## Step 2/3: increase or not the size of each group-type

Now let's build your group types through the `fn_CSWR_population.sqf` file:



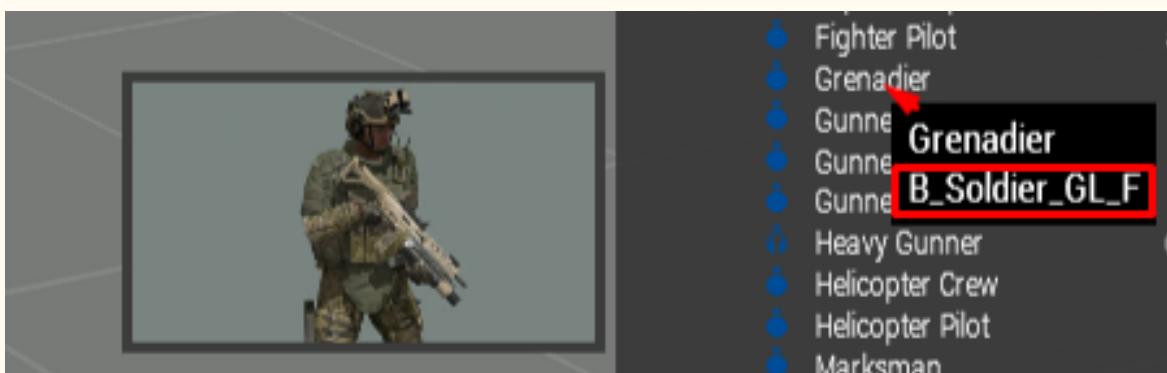
```
fn_CSWR_population.sqf X
8 [ ] spawn {
20     if ( CSWR_isOnBLU && count CSWR_spwnsAllBLU > 0 ) then {
23         // Define the number of soldiers (and their weapons) and vehicles o
38         ... // People groups
39         CSWR_people_BLU_light = ["B_Soldier_TL_F", "B_Soldier_F"];
40         CSWR_people_BLU_medium = ["B_Soldier_TL_F", "B_Soldier_F", "B_
41         CSWR_people_BLU_heavy = ["B_Soldier_TL_F", "B_Soldier_F", "B_
42         ... CSWR_people_BLU_custom_1 = [];
43         ... CSWR_people_BLU_custom_2 = [];
44         ... CSWR_people_BLU_custom_3 = [];
45         CSWR_people_BLU_sniper = ["B_ghillie_ard_F", "B_sniper_F"];
46 }
```

People groups supports many types of soldiers in a single group. Up to you to customize the type of soldier you want for each group type. Those green names are “classnames” and they are unique from *Arma 3*, *Arma 3 DLC*, *CDLC*, and *Arma 3* mods like *RHS* and *CUP*. Each type of soldier on each side has its own classname.



```
THIS IS A CLASSNAME
...
adLightBLU = ["B_Soldier_TL_F", "B_Soldier_F"];
adRegularBLU = ["B_Soldier_TL_F", "B_Soldier_F", "B_
adHeavyBLU = ["B_Soldier_TL_F", "B_Soldier_F", "B_
adCustomOneBLU = [];
adCustomTwoBLU = [];
```

On *Eden Editor*, drag and drop the unit or vehicle you want to use on CSWR, copying the classname and pasting that into the group you’re building like the image above. Here is another example of figuring out the classname of something through the `Assets` menu:



**Important 1:** it's okay if you use any unit classname from mods, or even from units of other sides. CSWR knows what you would like to do and it will handle that.

**Important 2:** never use vehicle classnames in people groups, as well as the opposite.

Example of what not to do:

```
→ // Soldiers-groups  
→ CSWR_team_BLU_light = ["B_Soldier_TL_F", "B_Soldier_F", "B_G_Offroad_01_armed_F"];
```

Arma 3 doesn't work well when a single group is composed of infantry and vehicles together.

### Step 3/3: define how many groups will spawn

Each row represents a group you want to spawn. You will see the vehicle rows separated from the Soldier groups, but the logic is the same.

```
fn_CSWR_population.sqf ×  
69 // People-groups  
70 // Guideline-for-you:[Side;[Spawns;CSWR_spawnsForPeopleBLU,CSWR_spawnsParadropBLU];Group-type;Group-formation:_form_BLU;  
[Destination;_move_ANY,_move_PUBLIC,_move_RESTRICTED,_move_OCCUPY,_move_WATCH,_move_HOLD];[Spawn-delay;timer,trigger-act  
71 [BLUFOR,[CSWR_spawnsForPeopleBLU,"A"],CSWR_people_BLU_light,_form_BLU_1,_be_COMBAT,[_move_RESTRICTED,"A"],[]]call THY_fn  
72 [BLUFOR,[CSWR_spawnsForPeopleBLU,"A"],CSWR_people_BLU_medium,_form_BLU_2,_be_COMBAT,[_move_RESTRICTED,"A"],[]]call THY_fn  
73 [BLUFOR,[CSWR_spawnsForPeopleBLU,"A"],CSWR_people_BLU_heavy,_form_BLU_1,_be_COMBAT,[_move_RESTRICTED,"A"],[]]call THY_fn  
74 [BLUFOR,[CSWR_spawnsForPeopleBLU,"A"],CSWR_people_BLU_sniper,_form_BLU_1,_be_COMBAT,[_move_RESTRICTED,"A"],[]]call THY_fn  
75 [BLUFOR,[CSWR_spawnsForPeopleBLU,"A"],CSWR_people_BLU_light,_form_BLU_2,_be_COMBAT,[_move_RESTRICTED,"A"],[]]call THY_fn  
76 [BLUFOR,[CSWR_spawnsForPeopleBLU,"B"],CSWR_people_BLU_medium,_form_BLU_1,_be_COMBAT,[_move_RESTRICTED,"A"],[]]call THY_fn  
77 [BLUFOR,[CSWR_spawnsForPeopleBLU,"B"],CSWR_people_BLU_medium,_form_BLU_1,_be_STEALTH,[_move_RESTRICTED,"B"],[]]call THY_fn  
78 [BLUFOR,[CSWR_spawnsForPeopleBLU,"B"],CSWR_people_BLU_heavy,_form_BLU_2,_be_COMBAT,[_move_RESTRICTED,"B"],[]]call THY_fn  
79 [BLUFOR,[CSWR_spawnsForPeopleBLU,"B"],CSWR_people_BLU_sniper,_form_BLU_2,_be_COMBAT,[_move_RESTRICTED,"B"],[]]call THY_fn  
80 [BLUFOR,[CSWR_spawnsParadropBLU,"A"],CSWR_people_BLU_light,_form_BLU_2,_be_AWARE,[_move_RESTRICTED,"B"],[]]call THY_fn  
81 [BLUFOR,[CSWR_spawnsParadropBLU,"A"],CSWR_people_BLU_medium,_form_BLU_1,_be_STEALTH,[_move_RESTRICTED,"B"],[]]call THY_fn  
82 [BLUFOR,[CSWR_spawnsParadropBLU,"A"],CSWR_people_BLU_heavy,_form_BLU_2,_be_CHAOS,[_move_RESTRICTED,"B"],[]]call THY_fn  
83 [BLUFOR,[CSWR_spawnsParadropBLU,"A"],CSWR_people_BLU_sniper,_form_BLU_2,_be_SAFE,[_move_RESTRICTED,"B"],[]]call THY_fn
```

Change (or not) the group type in each row:

```
fn_CSWR_population.sqf ×  
69 // People-groups  
70 // Guideline-for-you:[Side;[Spawns;CSWR_spawnsForPeopleBLU,CSWR_spawnsParadropBLU];Group-type;Group-formation:_form_BLU;  
[Destination;_move_ANY,_move_PUBLIC,_move_RESTRICTED,_move_OCCUPY,_move_WATCH,_move_HOLD];[Spawn-delay;timer,trigger-act  
71 [BLUFOR,[CSWR_spawnsForPeopleBLU,"A"],CSWR_people_BLU_light,_form_BLU_1,_be_COMBAT,[_move_RESTRICTED,"A"],[]]call THY_fn  
72 [BLUFOR,[CSWR_spawnsForPeopleBLU,"A"],CSWR_people_BLU_medium,_form_BLU_2,_be_COMBAT,[_move_RESTRICTED,"A"],[]]call THY_fn  
73 [BLUFOR,[CSWR_spawnsForPeopleBLU,"A"],CSWR_people_BLU_heavy,_form_BLU_1,_be_COMBAT,[_move_RESTRICTED,"A"],[]]call THY_fn  
74 [BLUFOR,[CSWR_spawnsForPeopleBLU,"A"],CSWR_people_BLU_sniper,_form_BLU_1,_be_COMBAT,[_move_RESTRICTED,"A"],[]]call THY_fn  
75 [BLUFOR,[CSWR_spawnsForPeopleBLU,"B"],CSWR_people_BLU_light,_form_BLU_2,_be_AWARE,[_move_RESTRICTED,"B"],[]]call THY_fn  
76 [BLUFOR,[CSWR_spawnsForPeopleBLU,"B"],CSWR_people_BLU_medium,_form_BLU_1,_be_STEALTH,[_move_RESTRICTED,"B"],[]]call THY_fn  
77 [BLUFOR,[CSWR_spawnsForPeopleBLU,"B"],CSWR_people_BLU_heavy,_form_BLU_2,_be_CHAOS,[_move_RESTRICTED,"B"],[]]call THY_fn  
78 [BLUFOR,[CSWR_spawnsForPeopleBLU,"B"],CSWR_people_BLU_sniper,_form_BLU_2,_be_SAFE,[_move_RESTRICTED,"B"],[]]call THY_fn
```

Below, the options you have:

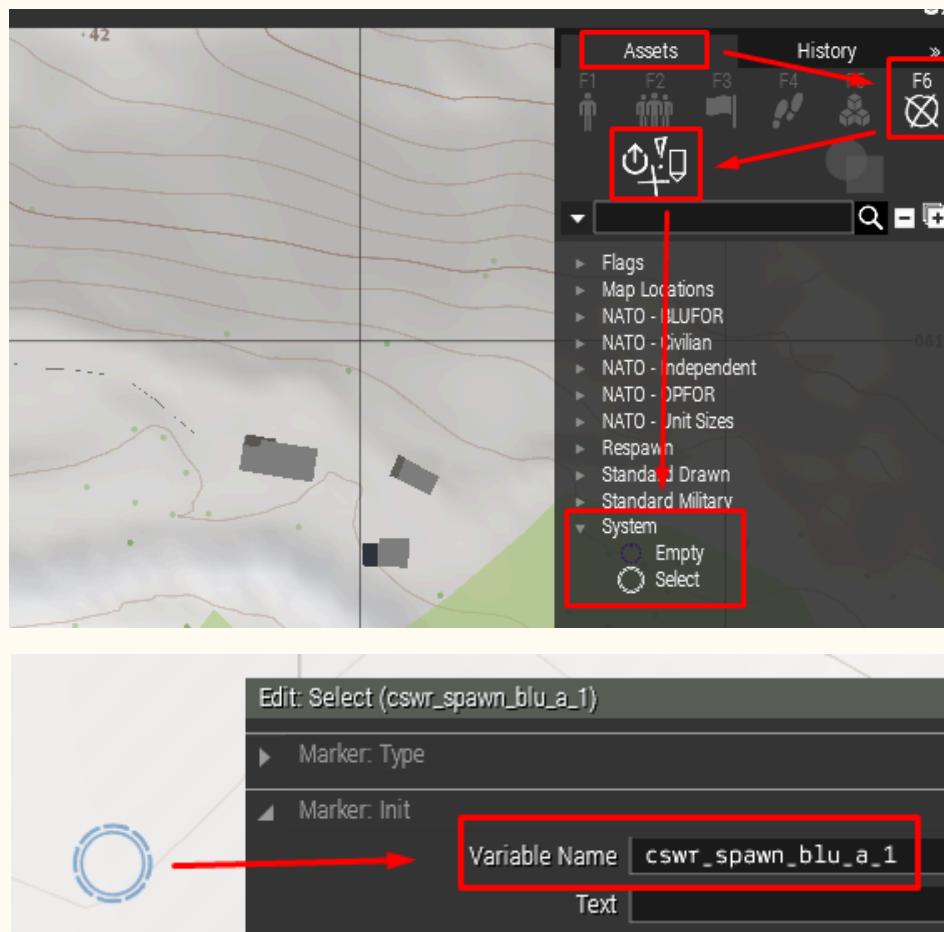
<b>Group-type column:</b> XXX = Side tag (blu, opf, ind, civ)	<b>Vehicle-type column:</b> XXX = Side tag (blu, opf, ind, civ)
CSWR_people_XXX_light	CSWR_vehicle_XXX_ground_light
CSWR_people_XXX_medium	CSWR_vehicle_XXX_ground_medium
CSWR_people_XXX_heavy	CSWR_vehicle_XXX_ground_heavy
CSWR_people_XXX_custom_1 CSWR_people_XXX_custom_2 CSWR_people_XXX_custom_3	CSWR_vehicle_XXX_ground_custom_1 CSWR_vehicle_XXX_ground_custom_2 CSWR_vehicle_XXX_ground_custom_3
CSWR_people_XXX_sniper	CSWR_vehicle_XXX_nautic_light CSWR_vehicle_XXX_nautic_medium CSWR_vehicle_XXX_nautic_heavy CSWR_vehicle_XXX_heli_light CSWR_vehicle_XXX_heli_heavy

**Important:** be careful to not abuse the number of rows. [It impacts server performance directly.](#)

## Spawn-points for side

Now, let's tell CSWR where the *BluFor* AI should spawn on the map. Define the spawn-points on *Eden Editor*, dropping a "Select" marker on the map and name it based on the example below.

To find the "Select" marker on *Eden*: through the "Assets" menu, click on "Markers" (F6) and then "System" on the category list. There, you find the "Select" marker.



Above, is a spawn-point for the BLU side. This is the marker attribute on *Eden Editor*.



Once you set the spawn-points in your mission, you're ready to rename each of them with one of the options below:

ON EDEN EDITOR	
What can spawn over there:	Marker variable-name for Spawns: XXX = Side tag (blu, opf, ind, civ) / A = any sector letter
<b>People groups</b> can spawn <b>Vehicles</b> cannot spawn	<b>Marker variable-name for Spawns:</b> XXX = Side tag (blu, opf, ind, civ) / A = any sector letter <ul style="list-style-type: none"> <li>• cswr_spawn_XXX_A_1</li> </ul>
<b>People groups</b> cannot spawn <b>Ground Vehicles</b> can spawn <b>Amphibious Vehicles</b> can spawn <b>Nautical Vehicles</b> can spawn <b>Helicopters</b> cannot spawn	<ul style="list-style-type: none"> <li>• cswr_spawnveh_XXX_A_1</li> </ul>
<b>People groups</b> cannot spawn <b>Ground Vehicles</b> cannot spawn <b>Amphibious Vehicles</b> cannot spawn <b>Nautical Vehicles</b> cannot spawn <b>Helicopters</b> can spawn	<ul style="list-style-type: none"> <li>• cswr_spawnheli_XXX_A_1</li> </ul>
<b>People groups</b> can spawn <b>Ground Vehicles</b> can spawn <b>Amphibious Vehicles</b> can spawn <b>Nautical Vehicles</b> cannot spawn <b>Helicopters</b> cannot spawn <b>Vehicles for Civilian</b> cannot spawn	<ul style="list-style-type: none"> <li>• cswr_spawnparadrop_XXX_A_1</li> </ul>

## ON POPULATION FILE

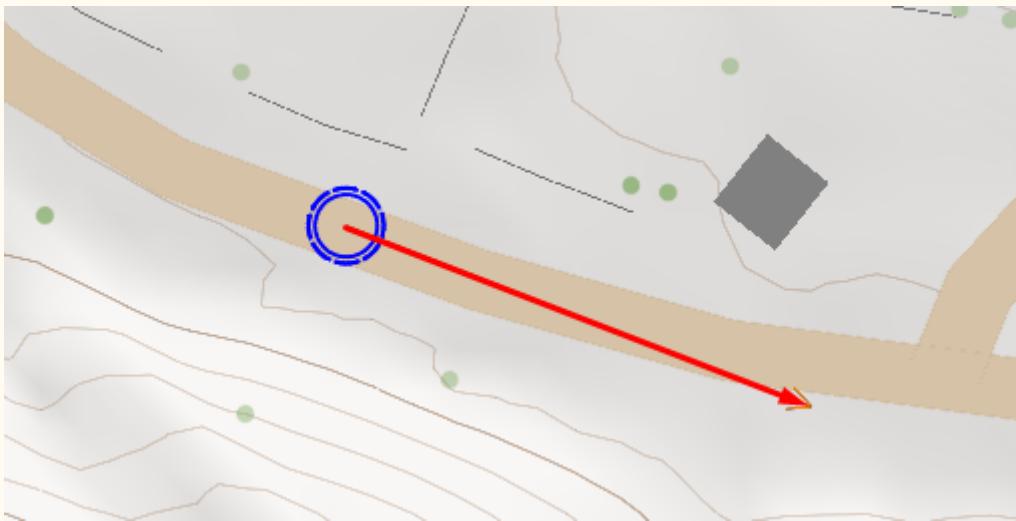
Group-type column: XXX = Side tag (blu, opf, ind, civ)	Spawn-point type column: XXX = Side tag
CSWR_people_XXX_light	<ul style="list-style-type: none"> <li>• CSWR_spawnsForPeopleXXX</li> <li>• CSWR_spawnsParadropXXX</li> </ul>
CSWR_people_XXX_medium	
CSWR_people_XXX_heavy	
CSWR_people_XXX_custom_1 CSWR_people_XXX_custom_2 CSWR_people_XXX_custom_3	<p>Groups of people are allowed to spawn on the ground, water, and air. Those groups CSWR decide to spawn in a spawn-point in water, the group automatically will use the marines loadout. If the group spawns on land, the loadout used will be the infantry. The group that spawns in the paradrop spawn-point will use the paratrooper loadout.</p>
CSWR_people_XXX_sniper	
CSWR_vehicle_XXX_ground_light	<ul style="list-style-type: none"> <li>• CSWR_spawnsForVehicleXXX</li> <li>• CSWR_spawnsParadropXXX</li> </ul>
CSWR_vehicle_XXX_ground_medium	
CSWR_vehicle_XXX_ground_heavy	
CSWR_vehicle_XXX_ground_custom_1 CSWR_vehicle_XXX_ground_custom_2 CSWR_vehicle_XXX_ground_custom_3	<p>Ground vehicles (with their crew) are allowed to spawn on land or in air with paradrop. Amphibious vehicles (with their crew) are allowed to spawn including in water when the spawn-point is placed by the editor in water.</p> <p>(*) Vehicles for CIV side CANNOT be spawned by paradrop.</p>
CSWR_vehicle_XXX_nautic_light	<ul style="list-style-type: none"> <li>• CSWR_spawnsForVehicleXXX</li> </ul>
CSWR_vehicle_XXX_nautic_medium	<p>For nautical vehicles finding a spawn to use is needed to set a spawn for vehicles in water, otherwise, the nautical vehicles won't be spawned.</p>
CSWR_vehicle_XXX_nautic_heavy	
CSWR_vehicle_XXX_heli_light	<ul style="list-style-type: none"> <li>• CSWR_spawnsForHelicopterBLU</li> </ul>
CSWR_vehicle_XXX_heli_heavy	<p>Helicopters (with their crew) are allowed to spawn landed if the spawn-point is on land, and they will spawn already hovering if the spawn-point is in water.</p>

Of course, you might add or remove spawn-points as much as you want since you don't forget to [make sure the side is true](#) on the `fn_CSWR_management.sqf` file.

## Set the spawn direction

Get used to setting the spawn direction even for people groups it doesn't make any difference but for vehicles it's important:

1. Toggle the map to 2D view, pressing "M";
2. Pressing Left-Shift, hold down the main mouse button over the marker;
3. Adjust the direction the vehicles should be facing right after the spawn.



## Set the spawn sector

To sectorize spawns of groups and vehicles is super easy. If you want to make, for example, only 2 groups of all your BLU army spawn in a specific region, you need to use sectors in some of your spawn-point markers.

```
cswr_spawn_blu_a_1
cswr_spawn_blu_a_2
cswr_spawn_blu_a_3
```

Above, 3 people spawn for those BLU units that want to spawn only in some of those options in Sector "A". But "where is sector A?" You define it by placing the markers on the map.

```
[BLUFOR, [CSWR_spawnsForPeopleBLU, "A"], CSWR_people_BLU_light, _form_
[BLUFOR, [CSWR_spawnsForPeopleBLU, "A"], CSWR_people_BLU_medium, _form_
[BLUFOR, [CSWR_spawnsForPeopleBLU, "B"], CSWR_people_BLU_heavy, _form_
[BLUFOR, [CSWR_spawnsForPeopleBLU, "B"], CSWR_people_BLU_sniper, _form_
```

In this case above, the light-group will spawn only through those sectorized spawn-points with “A” letter. You have the entire English alphabet to use as sector letters if you need them.

**Important:** it's not possible to set more than one sector to a group/vehicle.

## Spawn-points > Spawn Delay methods

If you want to make the groups and/or vehicles of one or more sides spawn lately in the mission, you can use 3 spawn delay methods to trigger those late spawns.

In the `fn_CSWR_population.sqf` file, through the *Spawning side section*, the last brackets “[ ]” of each group or vehicle are responsible for setting which method (or methods) the group will execute in the game.

```
light, _form_BLU_1, _be_COMBAT, [_move_RESTRICTED, "A"], [] call THY_fnc_CSWR_add_group  
medium, _form_BLU_2, _be_COMBAT, [_move_RESTRICTED, "A"], [] call THY_fnc_CSWR_add_group  
heavy, _form_BLU_1, _be_COMBAT, [_move_RESTRICTED, "A"], [] call THY_fnc_CSWR_add_group  
niper, _form_BLU_1, _be_COMBAT, [_move_RESTRICTED, "A"], [10, trigger_1] call THY_fnc_CSWR_add_group  
light, _form_BLU_2, _be_COMBAT, [_move_RESTRICTED, "A"], [1] call THY_fnc_CSWR_add_group  
medium, _form_BLU_1, _be_COMBAT, [_move_RESTRICTED, "A"], [5] call THY_fnc_CSWR_add_group  
heavy, _form_BLU_2, _be_COMBAT, [_move_RESTRICTED, "B"], [] call THY_fnc_CSWR_add_group  
niper, _form_BLU_2, _be_COMBAT, [_move_RESTRICTED, "B"], [] call THY_fnc_CSWR_add_group  
light, _form_BLU_2, _be_AWARE, [_move_RESTRICTED, "B"], [] call THY_fnc_CSWR_add_group
```

Editing the Spawn Delay of BLU side in `fn_CSWR_population.sqf` file.

So in that file, use these options (“Spawn Delay” column below) if you want to set some delay for some groups/vehicles:

Method:	Description:	In Population file <b>Spawn Delay:</b> possibilities in each group/vehicle line
Timer	Define a unique integer or float number (in minutes) that represents the time for the group to execute its spawn later in the mission.  This time begins to count not when the mission gets started but when the group is requested to be created by CSWR instead. By default, BLU side is read in ~3 seconds after the mission starts. If you were using all sides in CSWR, for example, OPF, IND, and CIV would depend how many side rows each side has in the population file.	[10]
Trigger Activation	Define one or more triggers on <i>Eden Editor</i> that, when activated, allow the group to execute its spawn later in the mission.  Trigger names can be anything as long as they match the <i>Population</i> file and	[my_trigger_1] Or [my_trigger_1, my_trigger_2]

	the trigger names configured by you on <i>Eden</i> .	
<b>Target Status</b>	<p>Define one or more objects (soldiers/people/building) on <i>Eden Editor</i> that, when killed or destroyed, allowed the group to execute its spawn later in the mission.</p> <p>Object names can be anything as long as they match the <i>Population</i> file and the object names configured by you on <i>Eden</i>.</p>	<p>[target_1] or [target_1, target_2, target_3]</p>
<b>Timer + Trigger</b>	Mix of methods. The first one reached allows the spawn immediately.	[10, my_trigger_1] or [10, my_trigger_1, my_trigger_2]
<b>Timer + Target</b>	Mix of methods. The first one reached allows the spawn immediately.	[5, target_1] or [5, target_1, target_2]
<b>Timer + Trigger + Target</b>	Mix of methods. The first one reached allows the spawn immediately.	[90, my_trigger_1, target_1] or [90, my_trigger_1, target_1, target_2]
<b>Trigger + Target</b>	Mix of methods. The first one reached allows the spawn immediately.	[trigger_1, my_trigger_1]
<b>No Spawn delay</b>	Set the group/vehicle to spawn right after the mission starts. Default!	[ ] or [0]

Important: for groups and vehicles with more than one Spawn Delay method, it's important to know that the first method reached to allow the spawn will ignore the other methods configured for that group, allowing the group to spawn immediately.

## But I want to delay all units

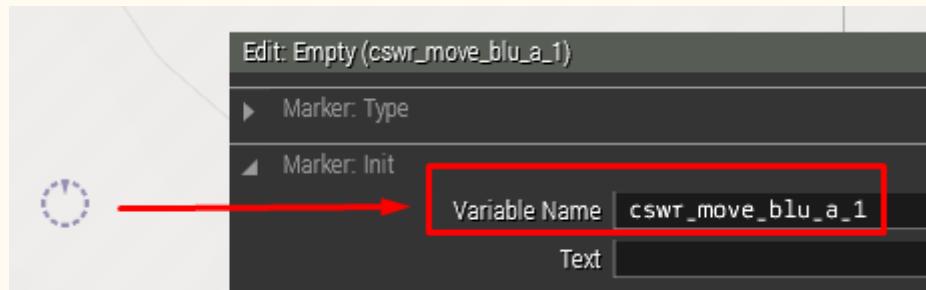
In this case, you can use the *CSWR\_wait* to delay when the script will be triggered by the server in-game. After this time the CSWR will initiate its functions normally.

```
fn_CSWR_management.sqf M X
29    ... CSWR_isElectroWarForOPF = true; //true
30    ... CSWR_isElectroWarForIND = true; //true
31    //Others:
32    ... CSWR_isEditableByZeus = true; //true
33    //Server:
34    ... CSWR_serverMaxFPS = 50.0; //Be advised:
35    ... CSWR_serverMinFPS = 20.0; //Be advised:
36    ... CSWR_wait = 60; //If you need to make CS
37
```

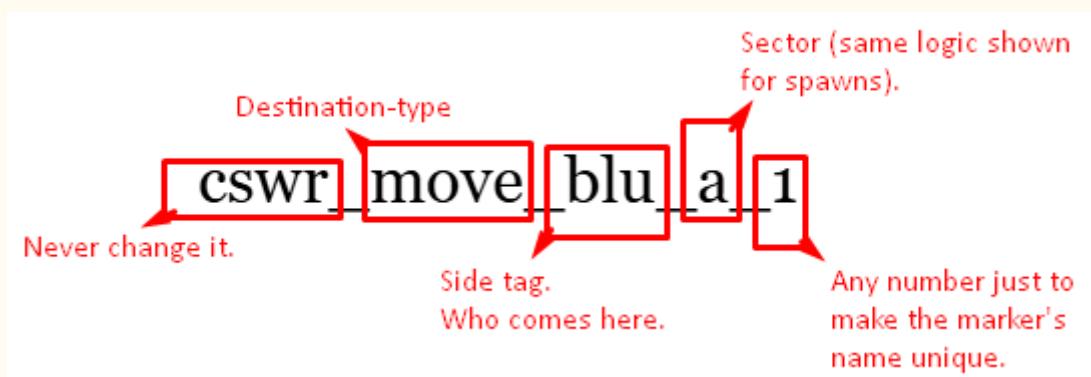
In the example above, CSWR would start its functions after 60 **seconds** (not minutes here) later once the mission gets started.

## Destinations (waypoints)

In CSWR, there's almost no end-point, which means after the group spawns, it will move through the map directly to the pre-defined destinations (waypoints), randomly, and can take a break just a while before going to the next same destination type.



An example of a destination marker on *Eden Editor*.



Below, there are all kinds of destinations available for you name it on *Eden Editor* markers:

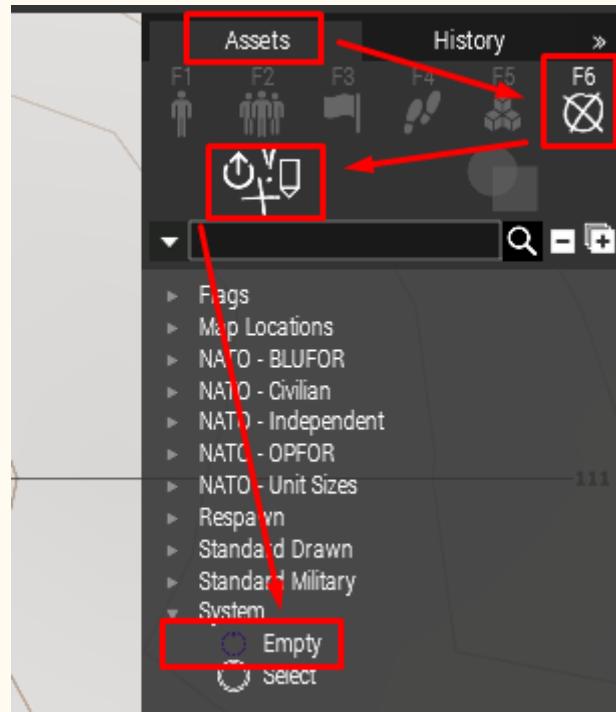
ON EDEN		
Destination types:	Description:	Destination marker names: XXX = Side tag (blu, opf, ind, civ) 1 = Any number / A = any sector letter
Move markers	Move destination is the simplest marker. A group of soldiers or a lone vehicle using it, after spawning, will go to one of them randomly, spend a while, and move on to the next marker of the same type and same sector. There are 2 types of move markers: public and restricted by side.	<ul style="list-style-type: none"><li>• cswr_move_public_A_1 <i>Public-destination is used by everyone assigned to use that, including civilians.</i></li><li>• cswr_move_XXX_A_1 <i>Restricted-destination (it's when a move marker has a side tag declared) is used only by the side assigned in the marker name.</i></li></ul> <p><b>Civilians</b> cannot use this.</p>

<p><b><u>Hold markers</u></b></p>	<p>Hold destination marker makes the AI holding the position for a long time. The hold marker was designed mainly for tracked vehicles (tank, IFV) that, after the spawn, will go to the hold marker and execute the hold-maneuver <u>almost exactly over the marker position and heading for the hold marker direction</u> (configured on Eden). Soldiers using hold destinations always will avoid being over the roads/streets and never in the hold-marker center position (reserved for tracked vehicles). After spawning, the hold-groups will go to one of them randomly, spend a long time there, and then move on to the next same type marker using the same sector letter.</p> <p><b>It's advised to not use hold markers over the roads/streets to avoid vehicle crashes, nor on very rough terrain.</b></p>	<ul style="list-style-type: none"> <li>• cswr_hold_XXX_A_1</li> </ul> <p><i>Hold-destination is used only by the side assigned in the marker name.</i></p> <p><b>Nautical vehicles</b> cannot use this. <b>Helicopters</b> cannot use this.</p>
<p><b><u>Watch markers</u></b></p>	<p>Watch destination is a marker designed exclusively for soldiers of <u>sniper groups with one or two units at maximum</u>. Once the spawning runs, the sniper group assigned with a watch destination will go around the marker (1000m by default), looking for high spots with some vision of the marker position. After the group's arrival, the sniper group will stay there (never changing the location) until the mission ends.</p>	<ul style="list-style-type: none"> <li>• cswr_watch_XXX_A_1</li> </ul> <p><i>Watch-destination is used only by the side assigned in the marker name.</i></p> <p><b>Vehicles</b> cannot use this. <b>Civilians</b> cannot use this. <b>Non-sniper-group</b> cannot use this.</p>
<p><b><u>Occupy markers</u></b></p>	<p>Occupy destination is a marker where a group of soldiers (<u>at a max of six units</u>) will get inside one of all available buildings inside a range (of 200m by default) from the marker center. Once inside the building, the group will stay there</p>	<ul style="list-style-type: none"> <li>• cswr Occupy_XXX_A_1</li> </ul> <p><i>Occupy destination used by Blufor groups assigned to use that.</i></p> <p><b>Vehicles</b> cannot use this.</p>

	<u>maybe</u> for a long time, and if there's an enemy around, the group will fight from inside and not leave their covered position.	
--	--	--

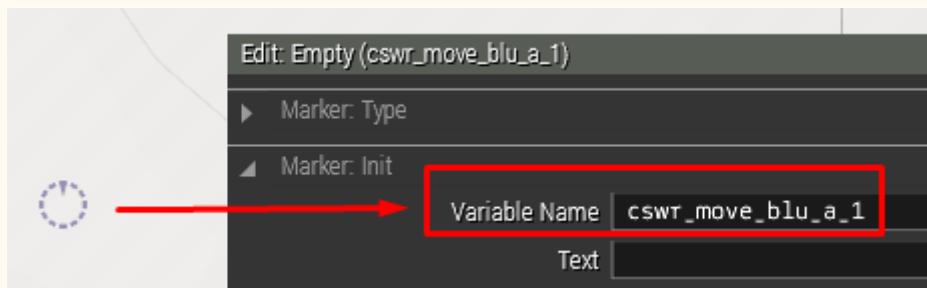
## Step 1/2: including destinations on *Eden Editor*

On *Eden Editor*, drop an “Empty” marker on the map, rename it with one of those examples from the table above, and with *Copy & Paste*, add how many markers (destinations of that type) your mission asks because the numeral of the markers name will increase automatically:



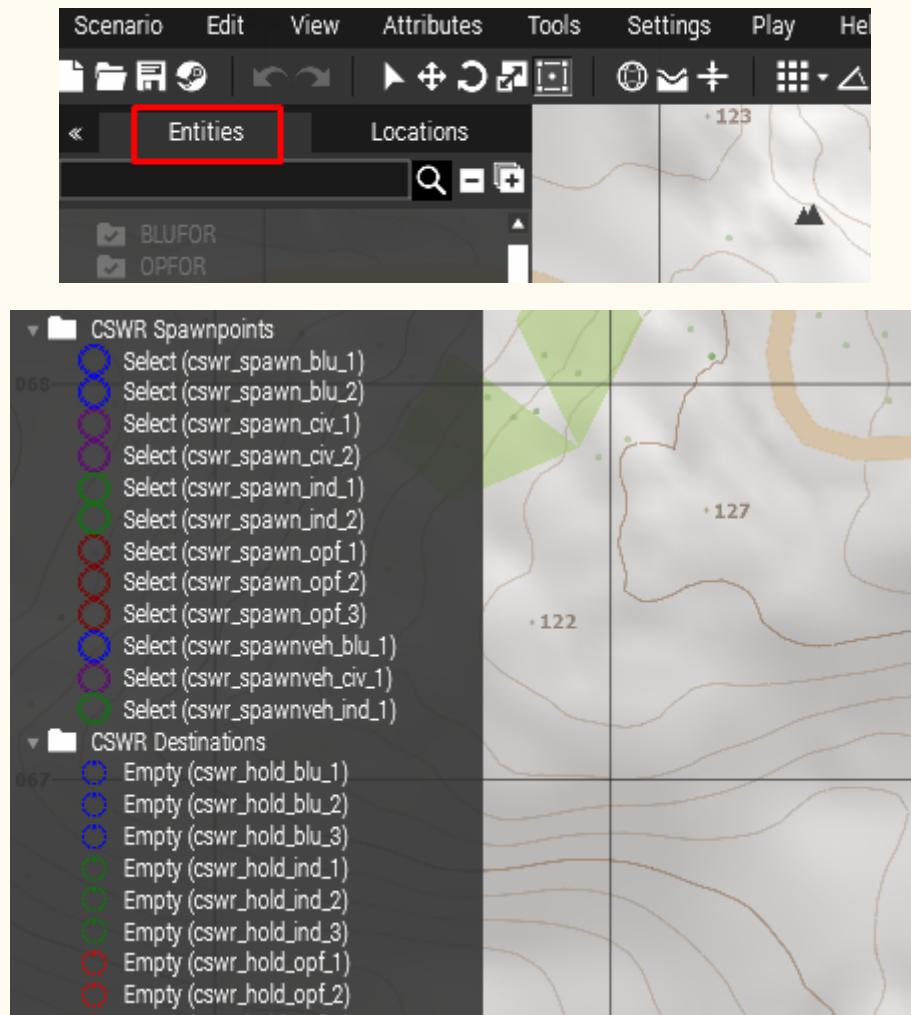
To find in *Eden Editor* “Empty” marker, in the Assets menu, click on “Markers” (F6) and, after that, “System” on the category list. There, you will find out “Empty” marker.

Drag and drop the marker where you want troops and vehicles going through. Soon you will understand each section of the marker’s name structure.



An example of a destination marker and its attributes on *Eden Editor*.

A tip for you: use the Entities menu to organize what markers are spawn and what of them are destinations:



## Step 2/2: define where each spawned group will move

Open the file: `fn_CSWR_population.sqf`

```
fn_CSWR_population.sqf X
67
68 // Soldier groups
69 // [Side; [Spawns; [CSWR_spwnsBLU, CSWR_spwnsVehicles, CSWR_spwnsParadropBLU]; Group type; Group formation]
70
71 [BLUFOR, [CSWR_spwnsBLU], CSWR_group_BLU_sniper, _form_BLU_1, _be_COMBAT, [_move_WATCH], []] call THY_fn
72 [BLUFOR, [CSWR_spwnsBLU, "A"], CSWR_group_BLU_light, _form_BLU_2, _be_SAFE, [_move_RESTRICTED, "A"], []
73 [BLUFOR, [CSWR_spwnsBLU], CSWR_group_BLU_light, _form_BLU_2, _be_AWARE, [_move_ANY], []] call THY_fn
74 [BLUFOR, [CSWR_spwnsBLU], CSWR_group_BLU_light, _form_BLU_1, _be_COMBAT, [_move_ANY], []] call THY_fn
75 [BLUFOR, [CSWR_spwnsBLU], CSWR_group_BLU_light, _form_BLU_1, _be_COMBAT, [_move_ANY], []] call THY_fn
76 [BLUFOR, [CSWR_spwnsBLU], CSWR_group_BLU_light, _form_BLU_1, _be_SAFE, [_move_ANY], []] call THY_fn
```

A red arrow points from the text above to the line of code at index 71, specifically highlighting the `_move_WATCH` parameter.

Above, each line is a Blufor group of soldiers. In the column before the last one, you define with type of destinations that the group must be assigned. Below, are the options (Destination types column):

ON POPULATION FILE:		
Only for debug <b>Dest. classes:</b>	<b>Destination types:</b>	<b>Description:</b>
Common destinations	_move_ANY	<p>Group and vehicle assigned for this go to any <u>common</u> destination marker, including restricted move markers from other sides, no matter whether different sectors. The only restriction about _move_ANY is: that if the vehicle is a nautical one, the vehicle will consider only nautical moves to go, and the same happens with ground vehicles, including the amphibious ones that NEVER will choose a nautical move to go.</p> <p>It's recommended if you want to provoke clashes as fast as possible between two or more sides spawned by CSWR.</p> <p><b>Sectorizing</b> is not allowed.  <b>Civilians</b> cannot use this.</p>
	_move_PUBLIC	<p>Group assigned for this goes to where anyone can go.</p> <p>Highly recommended to be used mainly for civilian side.</p>
	_move_RESTRICTED	<p>Group assigned for this goes only to restricted destination markers of their own side.</p> <p><b>Civilian</b> cannot use this.</p>
	<u>move_OCCUPY</u>	<p>Group assigned for this goes only to occupy destination markers of their own side.</p> <p><b>Vehicles</b> cannot use this.</p>
	<u>move_WATCH</u>	<p>Group assigned for this goes only to one watch destination marker of their own side.</p> <p><b>Vehicles</b> cannot use this.  <b>Civilian</b> cannot use this.  <b>Non-sniper-groups</b> cannot use this.</p>
	<u>move_HOLD</u>	<p>Group assigned for this goes only to hold-destination markers of their own side.</p> <p>Highly recommended to be used mainly for tracked-vehicles as MTBs and IFVs.</p> <p><b>Nautical vehicles</b> cannot use this.  <b>Helicopters</b> cannot use this.</p>

```
U_light, _form_BLU_2, _be_COMBAT, [_move_OCCUPY, "A"], []]
```

Example of a destination type 'occupy' in sector "A".

Can the groups move free, without a pre-defined destination?

No. In CSWR the focus is the editor has control of the areas where the troops should move through with a good level of unpredictability.

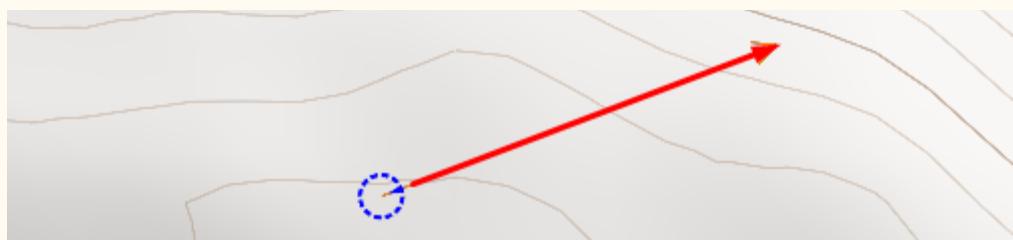
## Destinations > Hold-marker

**Helicopters** cannot use this.

**Nautical vehicles** cannot use this.

Hold-markers make units, mainly tracked vehicles, hold in that position and stay there for long periods, meanwhile, tracked vehicles always will take place in the center of the marker and face a specific direction configured by the Mission Editor. From all destinations available, only hold-markers are able to consider the *Eden* marker's direction:

1. Toggle the map to 2D view, pressing “M”;
2. Pressing Left-Shift, hold down the main mouse button over the marker;
3. Adjust the direction the tracked-vehicles must face right after their arrival.



Above, an example of a hold-marker facing to the northeast on *Eden Editor*.



Above, (when Debug is true through Zeus view) the same example in-game a tank holds that position.



Above, in-game the same example.

**Important:** groups of soldiers/people and non-tracked-vehicles don't care about the destination marker direction.

## Destinations > Watch-marker

**Vehicles** cannot use this.

**Civilians** cannot use this.

**Non-sniper-groups** cannot use this.

The watch-marker was designed specifically for Sniper groups (max of 2 members). When the group spawns, and is assigned to go to a watch destination, first the group will check on the map if there are highlands around the target zone (position of *Eden* watch-marker itself) and if there is a clear visual of the target zone. After that, the group will choose the best position possible to maintain that zone under watch for the whole game, never changing their position again.



Above, an example of a watch-marker (target zone) in *Eden Editor*.



Above, same example but in-game when Debug is `true`. That blue dot is the highland found around.



Above, sniper group behavior when with two members watching the zone.

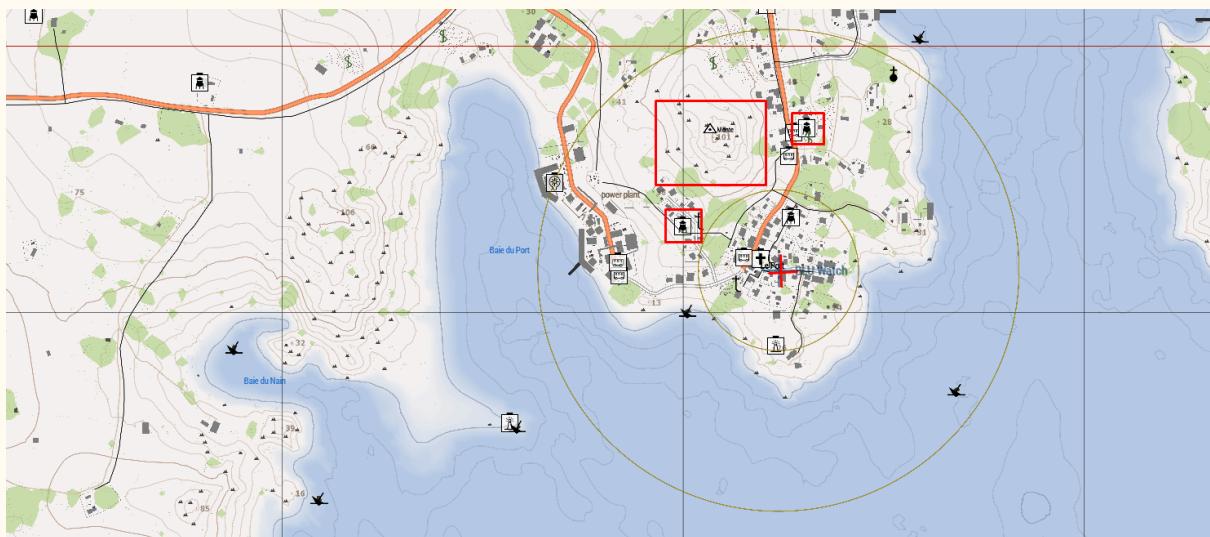
**Important 1:** if no good wild spot is found, the watcher-group will try to find an urban tower with a satisfactory view of the area to be watched.



Above, the watcher group in a tower, watching the zone right after opening the windows.

All acceptable towers that the watcher group will consider are configured as classnames' list called `CSWR_acceptableTowersForWatch` in `fn_CSWR_management.sqf` file.

**Important 2:** originally, the watcher group will look for spots between the range of 200 to 600m, ignoring all wild and urban spots out of that. To force the watcher-group to stay in urban areas only, you could decrease the maximum range value by looking for the `CSWR_watchMkrRange` variable in the `fn_CSWR_management.sqf` file.



Above, red squares are the potential watcher group positions, never in the small circle, never out of the big one.

**Important 3:** even if you set acceptable towers inside the correct range from the target zone if the tower is not high enough compared to the target zone, the watcher group will ignore that building.

## Destinations > Occupy-marker

**Vehicles** cannot use this.

The occupy-marker is exclusive for groups of soldiers/people (with a max of 6 members). As smaller the group is, easier to find a good building to cover all group members inside. When the group spawns, and is assigned to go to an occupied destination, first the group will check on the map the options around the occupy-marker and will go there, staying there for a long time before changing to another building. If the enemy is spotted, the group doesn't leave the building until the tread has been neutralized.



Above, an example of an occupy-marker in *Eden Editor*.



Above, through Zeus' view, the group members spread out inside the chosen building.

**Important:** if no good buildings are found or all buildings around the occupy-markers are destroyed, the group will stand still in its current position forever. To avoid this behavior at least right after the mission gets started, make sure you are setting the occupy-marker in urban areas or places with one or more buildings around. By default, CSWR will search those places 200m in all directions from the occupy-marker selected by the group. To change this value, look for the `CSWR_occupyMkrRange` variable in `fn_CSWR_management.sqf`.

## Behavior for each group

CSWR allows the Mission Editor to define the behavior of each group/vehicle spawned.

For example, go to the “SPAWNNG: BLUFOR” section and check it out:

The screenshot shows a portion of a mission editor script titled "fn\_CSWR\_population.sqf". The code is written in a dialect of SQF, used in Arma 3 missions. It defines several groups under the "BLUFOR" side, each with specific spawn parameters and behavior types. A red box highlights the "be\_XXX" entries in the last few rows of the list, which correspond to the examples in the table below. An arrow points from the text "check it out:" to this highlighted area.

```
fn_CSWR_population.sqf
...
69 ...// People groups
70 ...// Guideline for you: [Side; [Spawns; CSWR_spawnsForPeopleBLU; CSWR_spawnsParadropBLU]; Group
    [Destination; _move_ANY, _move_PUBLIC, _move_RESTRICTED, _move_OCCUPY, _move_WATCH, _move_HOLD
71 ...
72 ...[BLUFOR, [CSWR_spawnsForPeopleBLU, "A"], CSWR_people_BLU_light, _form_BLU_1, _be_SAFE, [_move_
73 ...[BLUFOR, [CSWR_spawnsForPeopleBLU, "A"], CSWR_people_BLU_medium, _form_BLU_2, _be_AWARE, [_move_
74 ...[BLUFOR, [CSWR_spawnsForPeopleBLU, "B"], CSWR_people_BLU_heavy, _form_BLU_1, _be_STEALTH, [_move_
75 ...[BLUFOR, [CSWR_spawnsForPeopleBLU, "B"], CSWR_people_BLU_sniper, _form_BLU_1, _be_AWARE, [_move_
76 ...[BLUFOR, [CSWR_spawnsForPeopleBLU, "B"], CSWR_people_BLU_light, _form_BLU_2, _be_COMBAT, [_move_
77 ...[BLUFOR, [CSWR_spawnsForPeopleBLU, "B"], CSWR_people_BLU_medium, _form_BLU_1, _be_COMBAT, [_move_
78 ...[BLUFOR, [CSWR_spawnsForPeopleBLU, "B"], CSWR_people_BLU_heavy, _form_BLU_2, _be_COMBAT, [_move_
79 ...[BLUFOR, [CSWR_spawnsForPeopleBLU, "B"], CSWR_people_BLU_sniper, _form_BLU_2, _be_COMBAT, [_move_
80 ...[BLUFOR, [CSWR_spawnsParadropBLU, "A"], CSWR_people_BLU_light, _form_BLU_2, _be_AWARE, [_move_
81 ...[BLUFOR, [CSWR_spawnsParadropBLU, "A"], CSWR_people_BLU_medium, _form_BLU_1, _be_STEALTH, [_move_
82 ...[BLUFOR, [CSWR_spawnsParadropBLU, "A"], CSWR_people_BLU_heavy, _form_BLU_2, _be_CHAOS, [_move_
83 ...[BLUFOR, [CSWR_spawnsParadropBLU, "A"], CSWR_people_BLU_sniper, _form_BLU_2, _be_SAFE, [_move_
84
```

Above, each row represents a BLU group that will spawn, and each group has its personality configured for all game long. Below are the options (Behavior types column) you might use for one or more groups, up to you:

In Population file Behavior types:	Description:
_be_SAFE	<ul style="list-style-type: none"><li><b>Group</b> walking with the guns low, looking around, and feeling safe.</li><li><b>Vehicle</b> slow and on roads. If a tank or another vehicle with turret, the crew out into the hatch.</li></ul>
_be_AWARE	<ul style="list-style-type: none"><li><b>Group</b> walking but with guns ready, looking around and concerned.</li><li><b>Vehicle</b> slow and on roads, but concerned with the crew inside when a tank or another vehicle with a turret.</li></ul>
_be_COMBAT	<ul style="list-style-type: none"><li><b>Group</b> in combat mode, running, ready to fight and keep the formation.</li><li><b>Vehicle</b> in combat mode, running, ready to fight, leaving the roads to take shortcuts.</li></ul>
_be_STEALTH	<ul style="list-style-type: none"><li><b>Group</b> in combat mode, but much more in a cautious manner, moving always along the cover.</li><li><b>Vehicle</b> in combat mode, turning the lights off. Even if it's a heavy vehicle, it'll avoid knocking down trees.</li></ul>
_be_CHAOS	<ul style="list-style-type: none"><li><b>Group</b> full speed, ready to fight and lose the formation.</li><li><b>Vehicle</b> full speed, rides everywhere, firing at will.</li></ul>

## Customizing the loadouts

**To know:** a soldier's loadout is all the gears, weapons, and equipment a soldier carries for a mission.

You can deeply customize the **loadout** by side, setting specific stuff by roles that you will see later. And the better: you can use stuff from mods easily too since the mod is loaded correctly in your server.

The easiest part of the customization you find in *fn\_CSWR\_management.sqf* file:

```
fn_CSWR_management.sqf 9 X
27
28 // For all sides:
29   ... CSWR_isBackpackForAllByFoot = false; // true == all units by foot (including
30   ... CSWR_isVestForAll ..... = false; // true == all units (including CIV) wi
31
32 // Settings by Side:
33   ... // Blu
34   ... CSWR_facesFromRegionBLU = ""; // Options: "Asia", "Africa", "Euro
35   ... CSWR_languageBLU ..... = ""; // Options: "UsEnglish", "GbEnglish
36   ... CSWR_canNvgInfantryBLU = false; // true == BLU infantry (including L
37   ... CSWR_canNvgMarinesBLU = false; // true == BLU marines receive Night
38   ... CSWR_canNvgParatroopsBLU = false; // true == BLU paratroops receive Ni
39   ... CSWR_canNvgSnipersBLU = false; // true == BLU snipers receive Night
40   ... CSWR_nvgDeviceBLU ..... = "NVGoggles"; // Set the NightVision classnam
41   ... CSWR_canFlashlightBLU ..... = true; // true == BLU units with no NVG get
42   ... CSWR_isForcedFlashlBLU ..... = false; // true == try to force all BLU stays
43   ... CSWR_flashlightDeviceBLU = "acc_flashlight"; // Set the flashlight clas
44   ... CSWR_watcherAccuracyBLU = "V"; // Group's accuracy executing Watcher
45   ... CSWR_isElectroWarForBLU = true; // true == vehicles of BLU will use E
46   ... // Opf
47   ... CSWR_facesFromRegionOPF = ""; // Options: "Asia", "Africa", "Euro
```

Example of *BluFor* basic customization.

Before you jump into more loadout customizations, get to know some points that help you later:

- A group-type is a group of people (soldiers) OR a vehicle with its crewmen;
- The same group-type can perform different roles but just one role during the game;
- The *CSWR* loadout customization works based on **role**, not group-type only;
- Roles are defined based on **where** the group-type spawns;
- Customizing only the Infantry, other roles will automatically follow that customization.

Look at the table below: the left column is the group-types you ask *CSWR* to spawn through the *fn\_CSWR\_population.sqf* file. The right column tells what you can customize for each group-type by role:

In Population file: <b>Group-Type by Role</b> XXX = Side tag (blu, opf, ind, civ)	In Loadout file: <b>You can customize:</b> Mandatory = cannot be removed
<b>Role &gt; Infantry</b> <ul style="list-style-type: none"> <li>• CSWR_people_XXX_light</li> <li>• CSWR_people_XXX_medium</li> <li>• CSWR_people_XXX_heavy</li> <li>• CSWR_people_XXX_custom_1</li> <li>• CSWR_people_XXX_custom_2</li> <li>• CSWR_people_XXX_custom_3</li> <li>• CSWR_people_XXX_sniper</li> <li>• CSWR_vehicle_XXX_ground_light</li> <li>• CSWR_vehicle_XXX_ground_medium</li> <li>• CSWR_vehicle_XXX_ground_heavy</li> </ul> <p><i>CSWR automatically recognizes as an Infantry role those group-types spawning on land.</i></p>	<b>You can customize:</b> <ul style="list-style-type: none"> <li>• <b>Helmet</b> customizable</li> <li>• <b>Uniform</b> customizable</li> <li>• <b>Vest</b> customizable</li> <li>• <b>Backpack</b> customizable</li> <li>• <b>Goggles</b> customizable</li> <li>• <b>NVG</b> on/off</li> <li>• <b>Flashlight</b> on/off</li> <li>• <b>Items</b> customizable</li> </ul> <p><i>All other loadouts inherit part of the infantry loadout.</i></p> <p><i>If not customized, the original unit loadout is used and other loadouts won't inherit anything.</i></p>
<b>Role &gt; Infantry &gt; Heavy Crew</b> <ul style="list-style-type: none"> <li>• CSWR_vehicle_XXX_ground_light</li> <li>• CSWR_vehicle_XXX_ground_medium</li> <li>• CSWR_vehicle_XXX_ground_heavy</li> </ul> <p><i>CSWR automatically recognizes as an Infantry Heavy Crew role those group-types of ground vehicle spawning on land and are using a Tank, APC, or IFV, no matter if the group-type has "light" or "medium" in the name.</i></p>	<ul style="list-style-type: none"> <li>• <b>Helmet</b> customizable *</li> <li>• <b>Goggles</b> customizable *</li> <li>• <b>Vest</b> customizable *</li> </ul> <p><i>(*) If not customized, inherit from Infantry.</i></p> <ul style="list-style-type: none"> <li>• <b>Uniform</b> inherited from Infantry</li> <li>• <b>Flashlight</b> inherited from Infantry</li> <li>• <b>Items</b> inherited from Infantry</li> </ul> <ul style="list-style-type: none"> <li>• <b>Backpack</b> not available</li> <li>• <b>NVG</b> not available (only if vehicle provides)</li> </ul>

### Role > Infantry > Sniper groups

- CSWR\_people\_XXX\_sniper

CSWR (*manually*) recognizes as an *Infantry Sniper Group* role that specific sniper group-type spawned on land.

- **Rifle** customizable
- **Rifle setup** customizable
- **Uniform** customizable \*
- **Helmet** customizable \*
- **Goggles** customizable \*
- **Backpack** customizable \*
- **NVG** on/off
- **Vest** customizable (mandatory) \*
- **Binoculars** customizable (mandatory)

(\*) If not customized, inherit from Infantry.

- **Items** inherited from Infantry

- **Pistol** no-customizable (mandatory)
- **Flashlight** not available

### Role > Marines

- CSWR\_people\_XXX\_light
- CSWR\_people\_XXX\_medium
- CSWR\_people\_XXX\_heavy
- CSWR\_people\_XXX\_custom\_1
- CSWR\_people\_XXX\_custom\_2
- CSWR\_people\_XXX\_custom\_3
- CSWR\_people\_XXX\_sniper
- CSWR\_vehicle\_XXX\_nautic\_light
- CSWR\_vehicle\_XXX\_nautic\_medium
- CSWR\_vehicle\_XXX\_nautic\_heavy

CSWR automatically recognizes as a Marine role those group-types that spawn in water.

- **Uniform** customizable \*
- **Helmet** customizable \*
- **Goggles** customizable \*
- **NVG** on/off
- **Vest** customizable \*
- **Backpack** customizable \*

(\*) If not customized, inherit from Infantry.

- **Flashlight** inherited from Infantry
- **Items** inherited from Infantry

### Role > Marines > Heavy Crew

- CSWR\_vehicle\_XXX\_ground\_light
- CSWR\_vehicle\_XXX\_ground\_medium
- CSWR\_vehicle\_XXX\_ground\_heavy

CSWR automatically recognizes as a Marine Heavy Crew role those group-types of amphibious vehicle spawning in water and are using a Tank, APC, or IFV, no matter if the group-type has "light" or "medium" in the name.

- **Helmet** inherited from Inf. H. Crew
- **Goggles** inherited from Inf. H. Crew
- **Vest** inherited from Inf. H. Crew
- **Uniform** inherited from Marines
- **Flashlight** inherited from Infantry
- **Items** inherited from Infantry
- **Backpack** not available
- **NVG** not available (or if vehicle provides)

### Role > Marines > Sniper groups

- CSWR\_people\_XXX\_sniper

CSWR (*manually*) recognizes as a Marine Sniper Group role that specific sniper group-type spawned in water.

- **Rifle** inherited from Inf. Sniper
- **Rifle setup** inherited from Inf. Sniper
- **NVG** inherited from Inf. Sniper
- **Binoc.** inherited from Inf. Sniper (mandat)
- **Uniform** inherited from Marines
- **Helmet** inherited from Marines
- **Goggles** inherited from Marines
- **Backpack** inherited from Marines
- **Vest** inherited from Marines (mandatory)

- **Items** inherited from Infantry
- **Pistol** no-customizable (mandatory)
- **Flashlight** not available

## Role > Paratroops

- CSWR\_people\_XXX\_light
- CSWR\_people\_XXX\_medium
- CSWR\_people\_XXX\_heavy
- CSWR\_people\_XXX\_custom\_1
- CSWR\_people\_XXX\_custom\_2
- CSWR\_people\_XXX\_custom\_3
- CSWR\_people\_XXX\_sniper
- CSWR\_vehicle\_XXX\_ground\_light
- CSWR\_vehicle\_XXX\_ground\_medium
- CSWR\_vehicle\_XXX\_ground\_heavy

CSWR automatically recognizes as a Paratroop role those group-types that spawn by paradrop (air).

- **Helmet** customizable \*
- **Uniform** customizable \*
- **NVG** on/off
- **Vest** customizable (mandatory) \*
- **Goggles** customizable \*

(\*) If not customized, inherit from Infantry.

- **Flashlight** inherited from Infantry
- **Items** inherited from Infantry
- **Backpack** not available yet (WIP)

## Role > Paratroops > Heavy Crew

- CSWR\_vehicle\_XXX\_ground\_light
- CSWR\_vehicle\_XXX\_ground\_medium
- CSWR\_vehicle\_XXX\_ground\_heavy

CSWR automatically recognizes as a Paratroop Heavy Crew role those group-types of ground vehicle spawning by paradrop (air), using a Tank, APC, or IFV, no matter if the group-type has "light" or "medium" in the name.

- **Helmet** inherited from Inf. H. Crew
- **Goggles** inherited from Inf. H. Crew
- **Vest** inherited from Inf. H. Crew
- **Uniform** inherited from Paratroops
- **Flashlight** inherited from Infantry
- **Items** inherited from Infantry
- **Backpack** not available
- **NVG** not available (or if vehicle provides)

## Role > Paratroops > Sniper group

- CSWR\_people\_XXX\_sniper

CSWR (manually) recognizes as a Paratroop Sniper Group role that specific sniper group-type spawned by paradrop (air).

- **Rifle** inherited from Inf. Sniper
- **Rifle setup** inherited from Inf. Sniper
- **NVG\*** inherited from Inf. Sniper
- **Pistol** inherited from Inf. Sniper
- **Vest** inherited from Paratroops
- **Uniform** inherited from Paratroops
- **Helmet** inherited from Paratroops
- **Goggles** inherited from Paratroops
- **Items** inherited from Infantry
- **Backpack** not available yet (WIP)
- **Flashlight** not available

## Role > Air > Helicopter Crew

- CSWR\_vehicle\_XXX\_heli\_light

- **Uniform** inherited from Infantry
- **Items** inherited from Infantry
- **Helmet** no-customizable
- **Goggles** no-customizable

- CSWR\_vehicle\_XXX\_heli\_heavy
- NVG not available (or if helmet has one)
- Flashlight not available

CSWR automatically recognizes as a Helicopter Crew role those group-types of helicopter, no matter where it's spawning (land or water).

Now you're prepared for the advanced customization. Open the file: [fn\\_CSWR\\_loadout.sqf](#)

```
//BLU-INFANTRY-UNITS:  
//Loadout-replacement//empty("")·results·no·change.  
[  
    "",.....//infantry·uniform  
    "H_Boniehat_tan",.....//infantry·helmet  
    "",.....//infantry·goggles  
    "V_Chestrig_khk",.....//infantry·vest·(one)  
    "B_Carryall_cbr",.....//infantry·backpack  
  
    _unit, _grpType, _grpSpec, _tag] call THY_fnc_CSWR_L  
  
//BLU-PARATROOP-UNITS:  
//Loadout-replacement//empty("")·results·no·change.  
[  
    "U_B_CombatUniform_mcam_tshirt",.....//Paratrooper  
    "H_HelmetB_light_desert",.....//Paratrooper  
    "G_Lowprofile",.....//Paratrooper  
    "V_HarnessO_brn",.....//Paratrooper  
  
    _unit, _grpType, _grpSpec, _tag] call THY_fnc_CSWR_L  
  
//BLU-INF.-HEAVY-CREW-UNITS:  
//Loadout-replacement//empty("")·results·no·change.  
[  
    "H_HelmetCrew_I",.....//heavy·crew·helmet  
    "G_Balaclava_lowprofile",.....//heavy·crew·goggles  
    "V_Chestrig_khk",.....//heavy·crew·vest  
  
    _unit, _grpType, _grpSpec, _tag] call THY_fnc_CSWR_L  
  
//BLU-INF.-SNIPER-GROUP-UNITS:  
//Loadout-replacement//empty("")·results·no·change.  
[  
    "U_B_FullGhillie_sard",.....//sniper·uniform  
    "",.....//sniper·helmet  
    "",.....//sniper·goggles  
    "",.....//sniper·vest·(cannot  
    "",.....//sniper·backpack  
    "srifle_EBR_F",.....//sniper·rifle·(cannot  
    "20Rnd_762x51_Mag",.....//sniper·rifle·magazin  
    "optic_SOS",.....//sniper·rifle·sight/  
    "",.....//sniper·rifle·rail  
    "muzzle_snrs_B",.....//sniper·rifle·muzzle  
    "",.....//sniper·rifle·bipod  
    "Rangefinder",.....//sniper·binoculars/(  
  
    _unit, _grpType, _grpSpec, _tag] call THY_fnc_CSWR_L
```

Keep in mind you are able to easily customize almost\* everything, including uniforms, vests, backpacks, and their contents. (\*) Weaponry is strongly advised to use only the weapons that the original unit used. [That already was mentioned](#). Nowadays, only Sniper groups have total support from CSWR to change the main rifle easily.

For example, you can remove all GPS devices (or any other items easily) from each CSWR spawned Blufor unit only, if you want. Or just exchange the standard vest from all Opfor units, without losing their items inside the original vest. With CSWR you're allowed to customize all unit loadouts spawned dynamically through the CSWR script.

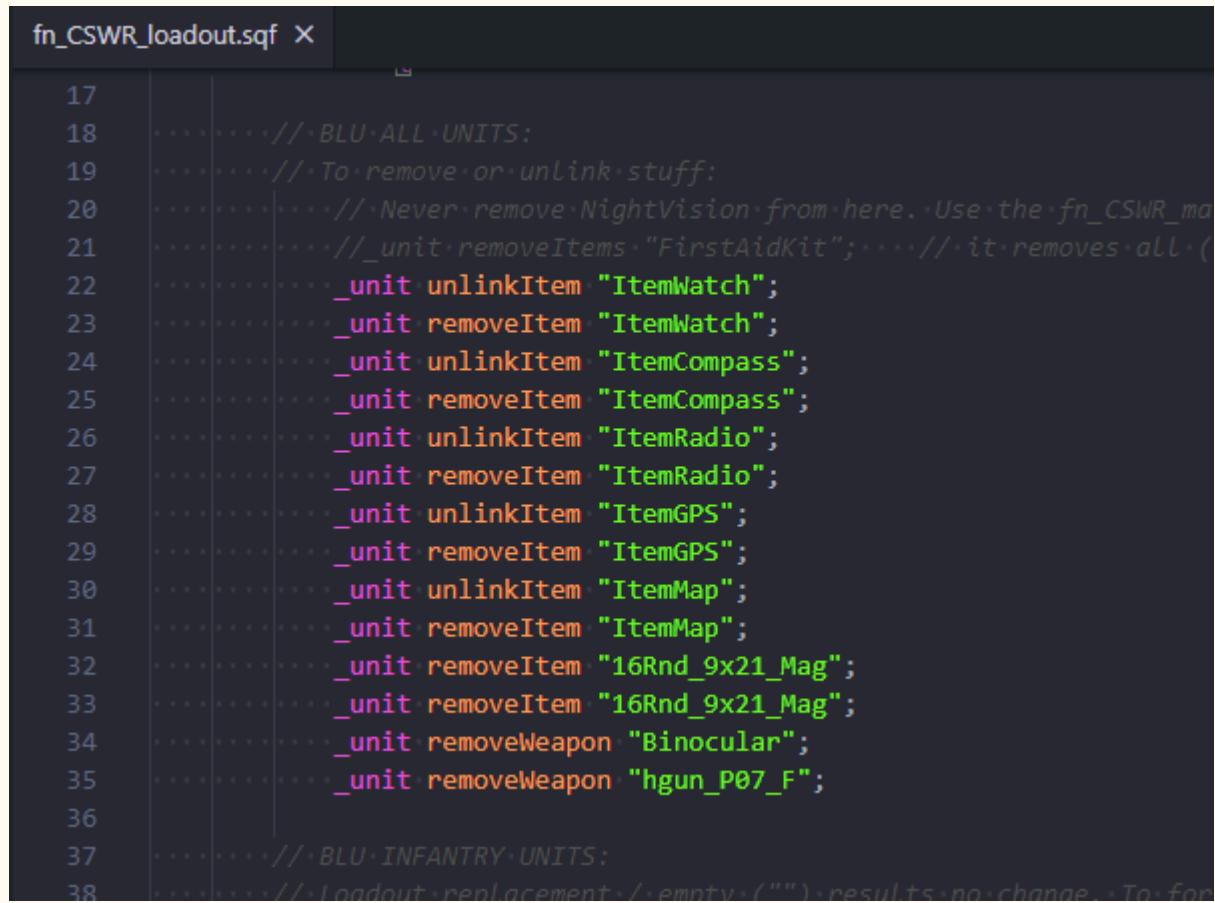
**Critical:** keep in mind if you change the primary weapon, for example, you would remove specialized weapons from specific soldier classes such as machine-gunners

and marksmen, adding the same primary gun for everyone on the side if you force their addition through CSWR.

Use loadout editing wisely, remembering that anything you add or remove as items will impact ALL units on that side. Then open the *fn\_CSWR\_loadout.sqf* file and let's customize your army:

## Step 1/4: removing and unlinking items

Here you will remove (and sometimes unlink and, after that, remove) items unnecessary for all units by a side:



The screenshot shows a code editor window with the title "fn\_CSWR\_loadout.sqf". The code is written in Arma 3's sqf language. It contains two main sections: "BLU\_ALL\_UNITS:" and "BLU\_INFANTRY\_UNITS:". Both sections contain commands to remove or unlink specific items from unit inventories. The items listed include "FirstAidKit", "ItemWatch", "ItemCompass", "ItemRadio", "ItemGPS", "ItemMap", "16Rnd\_9x21\_Mag", and "Binocular". The "Binocular" command is notable as it is a weapon item.

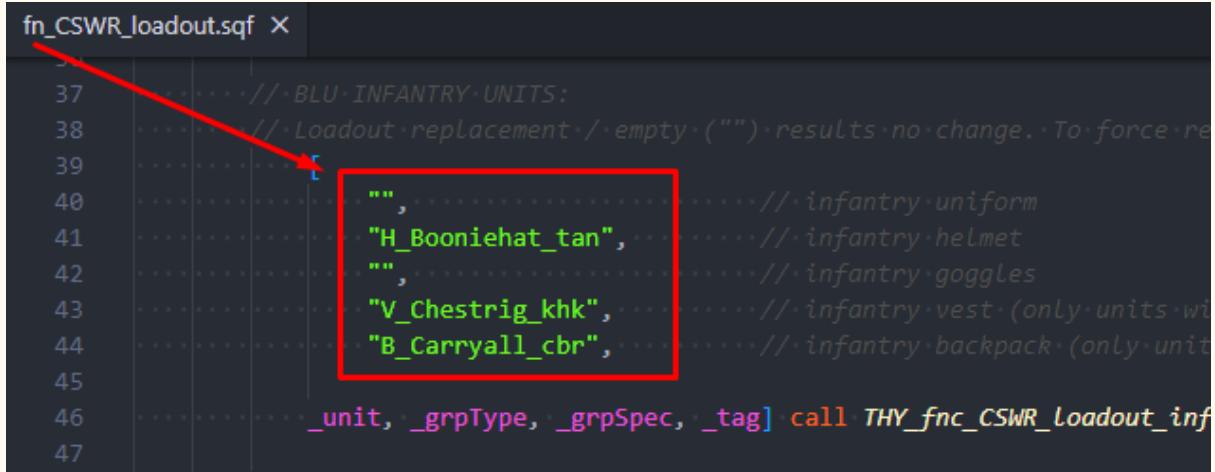
```
fn_CSWR_loadout.sqf
17
18     // BLU_ALL_UNITS:
19     // To remove or unlink stuff:
20     // Never remove NightVision from here. Use the fn_CSWR_ma
21     // _unit removeItems "FirstAidKit"; // it removes all (
22     _unit unlinkItem "ItemWatch";
23     _unit removeItem "ItemWatch";
24     _unit unlinkItem "ItemCompass";
25     _unit removeItem "ItemCompass";
26     _unit unlinkItem "ItemRadio";
27     _unit removeItem "ItemRadio";
28     _unit unlinkItem "ItemGPS";
29     _unit removeItem "ItemGPS";
30     _unit unlinkItem "ItemMap";
31     _unit removeItem "ItemMap";
32     _unit removeItem "16Rnd_9x21_Mag";
33     _unit removeItem "16Rnd_9x21_Mag";
34     _unit removeWeapon "Binocular";
35     _unit removeWeapon "hgun_P07_F";
36
37     // BLU_INFANTRY_UNITS:
38     // Loadout replacement // empty ("") results no change. To for
```

For Arma 3, "Binocular" is a weapon and not an item, so be careful with the commands to remove things. Always test your customs, mainly when you're removing and adding things in unit inventories.

**Important** if you remove the radio, it means group leaders won't have radios and are not able to exchange information with each other in long distances. It's an Arma 3 dynamic, not from CSWR.

## Step 2/4: customizing the infantry

Looking just at the green words on the image below, when you leave an empty field ("") it means that you don't want to change the [original gear of the unit you've set previously through the population file](#). When you want to force the gear removal, you can use "REMOVED" in uppercase. And of course, if you desire to change, for example, the uniform of the infantry, type into the uniform quotes the uniform classname desired.

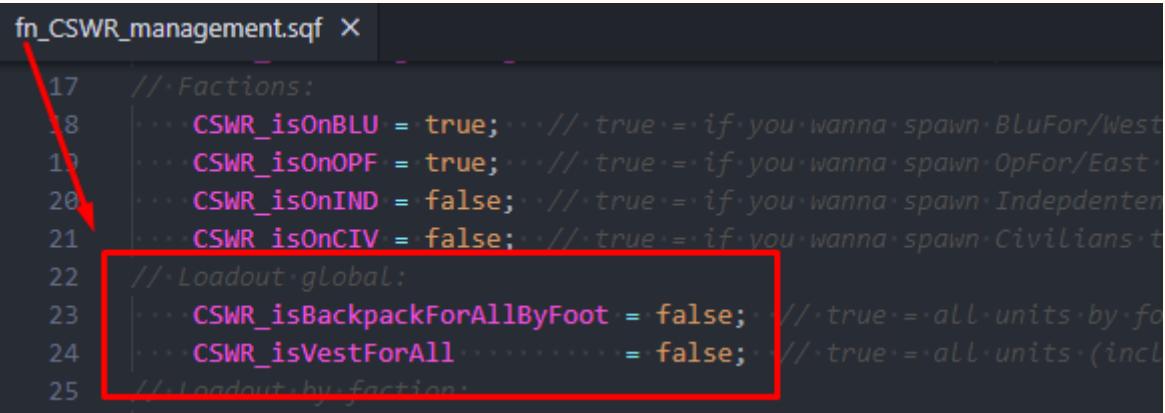


```
fn_CSWR_loadout.sqf X
35
36
37     ... // BLU INFANTRY UNITS:
38     ... // Loadout replacement ./empty("") results no change. To force re
39
40     "",           ... // infantry uniform
41     "H_Booniehat_tan", ... // infantry helmet
42     "",           ... // infantry goggles
43     "V_Chestrig_khk", ... // infantry vest (only units wi
44     "B_Carryall_cbr", ... // infantry backpack (only unit
45
46     _unit, _grpType, _grpSpec, _tag] call THY_fnc_CSWR_Loadout_inf
47
```

## Step 3/4: understanding the uniform, the vest, and the backpack

In *Arma 3*, the uniform, vest, and backpack are items' containers for each soldier. If you set an item for a unit, if you check their inventory in-game, there will be the item, maybe inside the uniform, backpack, or vest. If the unit has no uniform, vest, or backpack, the unit won't be able to store the item. When you set a new vest for a side, CSWR will scan all items inside the current vest, save them, change the vest to the new one, and, finally, transfer all original items to the new vest. The same happens when the uniform and backpack are replaced, for example, preserving the original unit weapons magazines.

**Important:** Notice that not every soldier has a backpack or vest. CSWR will respect that logic but if you want to force every unit to get a backpack and/or a vest, you can do it through the file *fn\_CSWR\_management.sqf*, setting as true the options:



```
fn_CSWR_management.sqf X

17 // Factions:
18   CSWR_isOnBLU = true; // true = if you wanna spawn BluFor/West
19   CSWR_isOnOPF = true; // true = if you wanna spawn OpFor/East
20   CSWR_isOnIND = false; // true = if you wanna spawn Independent
21   CSWR_isOnCIV = false; // true = if you wanna spawn Civilians
22 // Loadout global:
23   CSWR_isBackpackForAllByFoot = false; // true = all units by foot
24   CSWR_isVestForAll = false; // true = all units (incl
25 // Loadout by faction:
```

The option `CSWR_isBackpackForAllByFoot` when `true` sets the pre-defined new backpacks for all soldiers of all army sides, except paratroopers and heavy crewmen.

#### Step 4/4: adding and linking new items

To finish, the last section of loadout customization is the area to add things. No doubt, this is the section that I use the least in my customizations. Generally, I just remove stuff from units, but when customizing the civilian side for example, I love to add the cellphone item to each unit just for fun. Be aware for some items like GPS that is important to “addItem” first and, right after “linkItem” too. Otherwise, if you just add the GPS, the item will be in the unit’s inventory but not in use.

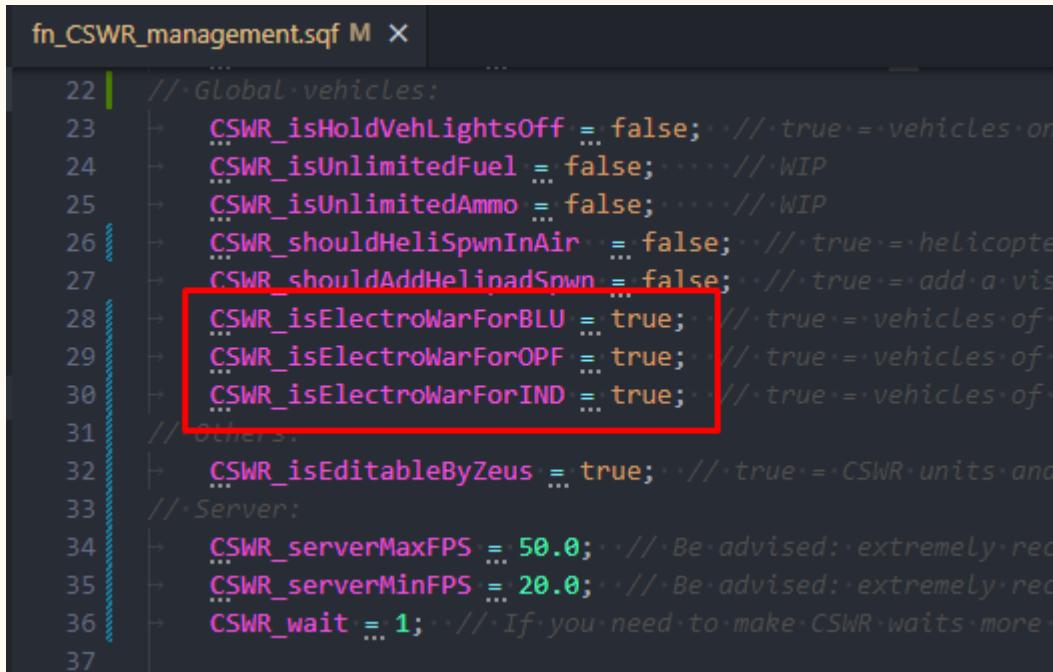
```
    // BLU ALL UNITS:
    // To add or link stuff:
    // Never add NightVision from here. Use the fn_CSWR_management.sqf to set this.
    _unit addItem "FirstAidKit"; // create just one bandage in unit inventory.
    _unit addItem "FirstAidKit"; // create another bandage in unit inventory.
};
```

**Critical:** about adding lots of things, always take care not to exceed the load limit of the chosen uniform, vest, and backpack. If no more space in their inventory, items start to disappear.

**Important:** if something is wrong in your loadout changing tests or you want to learn how to see the original unit loadout code, check this out.

## Electronic Warfare Resources

You can turn on or off the electronic warfare resources for vehicles by each side. For example, if you want to make BLU a Brazilian Army against a poor OPF Argentine militia, you can turn *true* the electronic war systems for BLU, and turn it *false* for OPF in *fn\_CSWR\_management.sqf* file:



```
fn_CSWR_management.sqf M X
22 // Global vehicles:
23     CSWR_isHoldVehLightsOff = false; // vehicles on
24     CSWR_isUnlimitedFuel = false; // WIP
25     CSWR_isUnlimitedAmmo = false; // WIP
26     CSWR_shouldHeliSpwnInAir = false; // true = helicopter
27     CSWR_shouldAddHelipadSpwn = false; // true = add a vis
28     CSWR_isElectroWarForBLU = true; // true = vehicles of
29     CSWR_isElectroWarForOPF = true; // true = vehicles of
30     CSWR_isElectroWarForIND = true; // true = vehicles of
31 // Others:
32     CSWR_isEditableByZeus = true; // true = CSWR units and
33 // Server:
34     CSWR_serverMaxFPS = 50.0; // Be advised: extremely rec
35     CSWR_serverMinFPS = 20.0; // Be advised: extremely rec
36     CSWR_wait = 1; // If you need to make CSWR waits more
37
```

What happens when *true*?

- Side vehicles are capable of reporting their own positions;
- Side vehicles are capable of receiving targets;
- Side vehicles are capable of reporting targets;

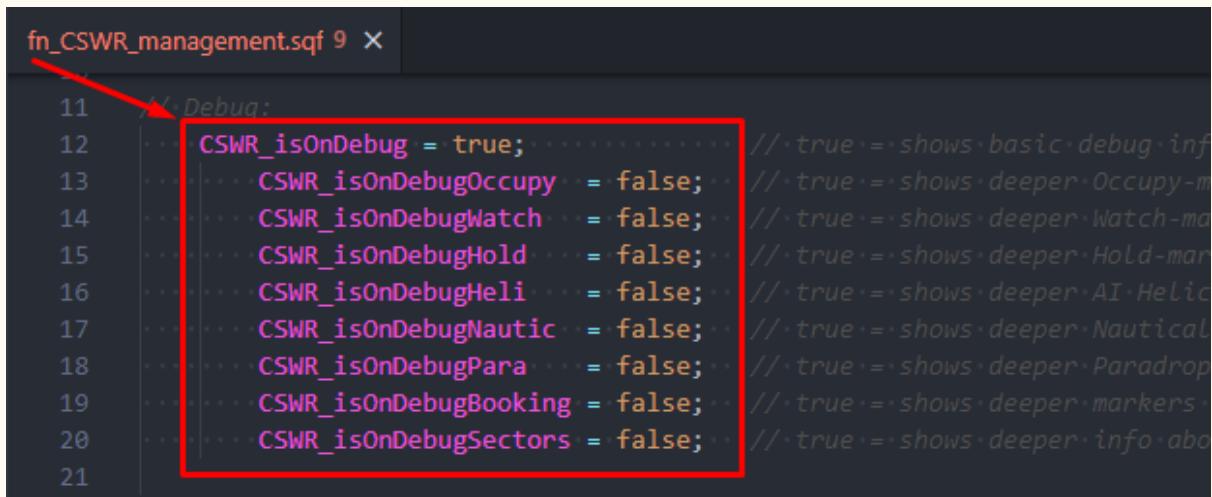
**Important:** to improve the game experience, all helicopters always will receive targets from their allies, even the Mission Editor turned off electronic warfare for a side.

## Debug Mode: investigating what is happening

First of all, make sure you are NOT running another script debug together because it could break the data presentation on your screen. Work with only one script debug mode ‘true’ at a time.

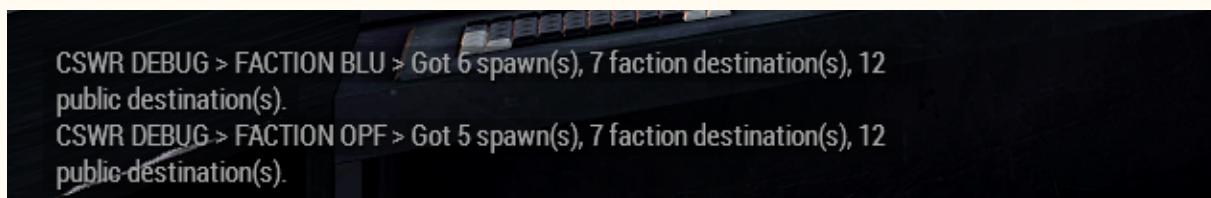
That said, if you’re facing something weird or unexpected or even just to check the CSWR AI behaviors with the settings you have set, it’s recommended to turn `Debug Mode true`. This resource will describe a lot of actions the script is doing and, if you turn *true* on the Debug extensions (for Hold, Watch, Occupy, and Helicopters) you have powerful information on the screen.

To see deeper information about Occupy movements, for example, it’s needed to turn to *true* `CSWR_isOnDebugOccupy` as well as maintain `CSWR_isOnDebugGlobal` activated.

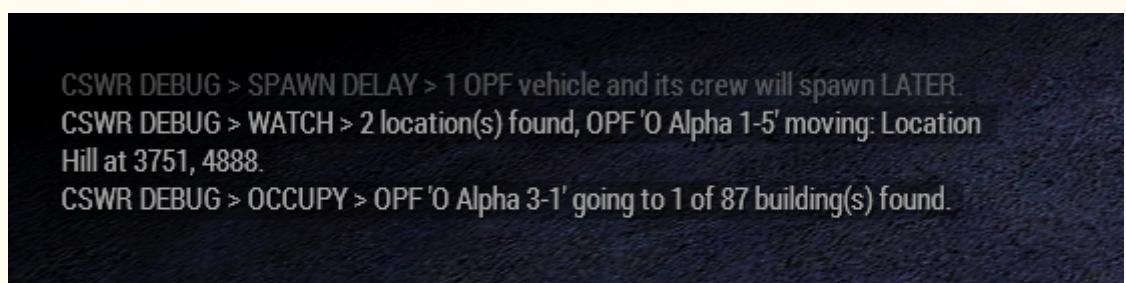


```
fn_CSWR_management.sqf 9 X
11 // Debug:
12 CSWR_isOnDebug = true;           // true == shows basic debug info
13 CSWR_isOnDebugOccupy = false;    // true == shows deeper Occupy-movements
14 CSWR_isOnDebugWatch = false;     // true == shows deeper Watch-movements
15 CSWR_isOnDebugHold = false;      // true == shows deeper Hold-managements
16 CSWR_isOnDebugHeli = false;      // true == shows deeper AI Helicopter movements
17 CSWR_isOnDebugNautic = false;    // true == shows deeper Nautical movements
18 CSWR_isOnDebugPara = false;      // true == shows deeper Paradrop movements
19 CSWR_isOnDebugBooking = false;   // true == shows deeper markers
20 CSWR_isOnDebugSectors = false;   // true == shows deeper info about sectors
21
```

The header of `fn_CSWR_management.sqf` file.



An example of CSWR debug messages right after the mission starts with debugging as `true`.



CSWR DEBUG > SPAWN DELAY > A OPF vehicle was granted TO SPAWN by  
TIMER (it was 120 secs).

CSWR DEBUG > BLU 'B Alpha 2-1' helicopter returning to base!

CSWR DEBUG > OPF 'O Bravo 1-1' helicopter is TAKING OFF!  
CSWR DEBUG > HOLD > BLU 'B Alpha 1-6' tracked-vehicle hold [Desired:  
152.665° | Executed: 152.963°].

**Important:** don't forget to turn the (at least) `CSWR_isOnDebugGlobal false` to save  
the server performance, ignoring lots of debug code lines that, when your mission is  
ready, would be unnecessary still to be running.

## Fixing: trying to edit the unit's loadout

If you are facing a hard time editing your side's loadouts, maybe you are trying to remove a thing that doesn't exist or calling weird commands. Let's investigate:

- 1) On *Eden Editor*, drag and drop on map one unit of the original side you are trying to make changes;
- 2) After that, go to unit Attributes and click "Edit Loadout" > "BI Virtual Arsenal";
- 3) Now, just click on "Export" button;



- 4) Open your Notepad and CTRL+V;

5) Now, check carefully if you are editing the same items the original unit brings.

## Fixing: making CSWR lighter for servers

First of all: the number of AI you ask to CSWR spawn in the mission is the CSWR feature with the highest impact on server performance.

### Which number of AI is good enough

Remember that in `fn_CSWR_population.sqf` pretty often each new line is a group of units. If one side you have 20 lines, maybe you are asking CSWR to spawn 60 units or even much more, all depends on [how many units each one of the group-types](#) you've built. In 2023 my game computer was able to manage around 300 CSWR AI's without huge performance drops. Arma 3 servers with less than 20 FPS will impact the performance of all its players. Make some tests and, if you really know what you're doing, you can edit some numbers in `fn_CSWR_management.sqf` to help CSWR recalc stuff:

```
fn_CSWR_management.sqf M X
33 // Server:
34 CSWR_serverMaxFPS = 50.0; // Be advised: extremely recommended do
35 CSWR_serverMinFPS = 20.0; // Be advised: extremely recommended do
36 CSWR_wait = 1; // If you need to make CSWR waits more for other sides
37
```

### Sides CSWR will spawn

It doesn't matter for performance if the script will spawn one or all sides. The most important will be how many AI units you are dropping in the mission.

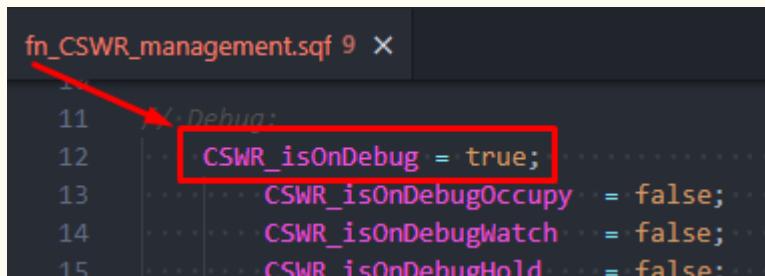
### Turn off the Zeus

If your mission won't use Zeus power-ups, just turn it off to give a good breathe to the server.

```
fn_CSWR_management.sqf 9 X
102
103 // Others:
104 CSWR_isEditableByZeus = true;
105 //CSWR_reportToRadioAllies = false;
106 //CSWR_spawnHeliOnShipFloor = 25;
107
```

## Turn off the Debug

Debug should be used only during the mission creation. After that, turn it off. Debug demands a lot of CPU calculations that are useless for the real game.



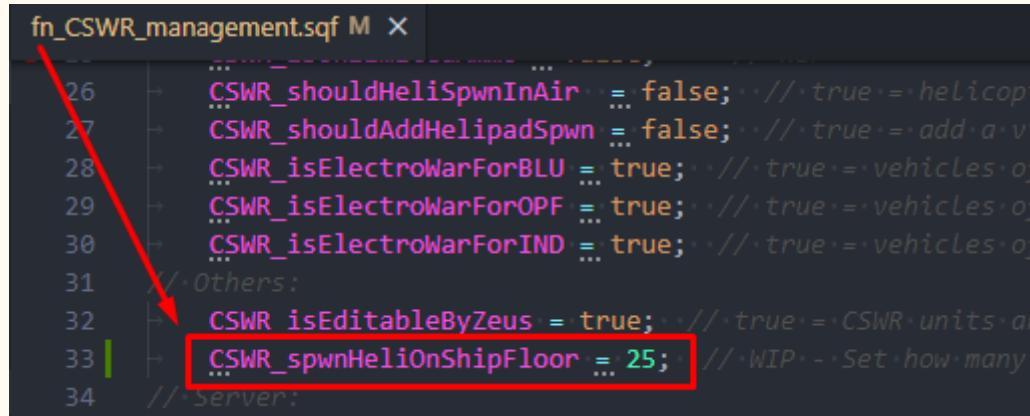
```
fn_CSWR_management.sqf 9 X
1
2
3
4
5
6
7
8
9
10
11    A\Debug;
12        CSWR_isOnDebug = true;
13        CSWR_isOnDebugOccupy = false;
14        CSWR_isOnDebugWatch = false;
15        CSWR_isOnDebugHold = false;
```

## Using the Dynamic Simulation

All AI created by CSWR doesn't use Dynamic Simulation. Because of the CSWR creation concept, the script doesn't want to work with [DynamicSimulation](#) enabled. Of course, you can add manually how many units and objects using that feature through the *Eden Editor* or another script, CSWR won't care about it.

## Fixing: helicopters over ship floors (NOT AVAILABLE, WIP)

I'm working to make this easier but for now, if you want to set spawns for helicopters over an asset floating in water (like a ship) you will need to set the height between the water surface and the asset's solid floor:



```
fn_CSWR_management.sqf M X
26     CSWR_shouldHeliSpwnInAir = false; // true == helicop
27     CSWR_shouldAddHelipadSpwn = false; // true == add a v
28     CSWR_isElectroWarForBLU = true; // true == vehicles o
29     CSWR_isElectroWarForOPF = true; // true == vehicles o
30     CSWR_isElectroWarForIND = true; // true == vehicles o
31     /> Others:
32     CSWR_isEditableByZeus = true; // true == CSWR_units.a
33     CSWR_spwnHeliOnShipFloor = 25; // WIP -- Set how many
34 // Server:
```



The solid floor of the *USS Freedom* is 25 meters above sea level.



The solid floor of the *USS Liberty* is 9 meters above sea level.

**Critical:** extremely recommended you have more spawns dedicated to helicopters by the number of helicopters you want to spawn. This temporary solution is fragile and can easily provoke explosions if more than one helicopter chooses the same spawn-point at the same time.

## Fixing: units are not in a crouch when inside a ruin

It is because of one of these two factors: 1) the ruin is not listed as a building to be ignored by CSWR, and 2) it's not listed as an acceptable ruin to be used by CSWR units doing occupy movement.



Above, it's how the group in occupy stays in an unknown ruin.

Below, in an acceptable ruin.



If you want to allow the units to use a ruin as a place to be (in a crouch in these cases), add the ruin classname in the `CSWR_occupyAcceptableRuins` list. Otherwise, add the classname in `CSWR_occupyIgnoredBuildings`.

Both lists are in the `fn_CSWR_management.sqf` file.

## How CSWR works technically (advanced knowledge)

All the action descriptions below were briefed to make them readable easily, not representing the entire feature flow but those most important actions.

### Destination > Common > Move

1. Once CSWR starts, it reads and saves all move-destinations (markers) in the move-destinations-library, classifying them as move-markers-on-land and move-markers-in-water;
2. The vehicle or group spawns and waits for its `_move_RESTRICTED`, `_move_PUBLIC` or `_move_ANY` destination;
3. CSWR sets the group/vehicle behavior chosen by the editor through `fn_CSWR_population.sqf` (this is a suggestion to AI but it decides based on the environment around them);
4. CSWR doesn't interfere with Arma's decisions during the group/vehicle journey until the move-marker position;
  - a. Except if a helicopter with low ammo or low fuel or too damaged or the crew is wounded, it returns to the closest side helipad (include those invisible) used as spawn to fix its situation and fly back to the front after that;
    - i. If the helicopters spawned in water (hovering) and there's no any spawn for helicopters on land, the helicopters won't execute RTB and will fight until death.
5. Once the group/vehicle arrives in position, maybe it stays for a short time, maybe not, but:
  - a. If a group of people: Arma decides;
  - b. If a ground vehicle: Arma decides;
  - c. If a nautical vehicle: Arma decides;
  - d. If a helicopter: 66% chances to just pass by the area; 33% to execute a "Search and Destroy" for a while;
6. While remaining in the move-position:
  - a. If a group of people: Arma decides;
  - b. If a ground vehicle: Arma decides;
  - c. If a nautical vehicle: Arma decides;
  - d. If a helicopter: never stays in a position, except if it's engaging;
7. After a short time (editable via `fn_CSWR_management.sqf`) taking a break, the group or vehicle looks for another move-destination in the same sector. From the step 3, repeat everything.

## Destination > Specialized > Hold

1. Once CSWR starts, it reads and saves all hold-destinations (markers) in the hold-destinations-library, classifying them only as hold-markers-on-land;
2. CSWR removes all small rocks and bushes too close to the hold-markers center;
3. The vehicle or group spawns and waits for its *\_move\_HOLD* destination;
4. CSWR sets the group/vehicle behavior chosen by the editor through *fn\_CSWR\_population.sqf* (this is a suggestion to AI but it decides based on the environment around it);
5. The script understands if it's a group of people or a vehicle;
  - a. If a group of people:
    - i. It will choose a peripheral position around the hold-marker, but never over a road or too rough terrain or water and not too close to another object, (*WIP* and never be in a peripheral position in front of the center-direction where maybe an armed tracked vehicle can park with its main gun pointed to the horizon in front);
    - ii. The group's direction when in the hold position is random;
  - b. If a wheeled vehicle:
    - i. The same as a group of people;
  - c. If a tracked vehicle (priority):
    - i. CSWR searches for a free hold-destination not booked by another tracked vehicle:
      1. If some hold-destination is free, the tracked-vehicle books that position and goes there to hold as accurately as possible on the center of the hold-destination;
      2. If no free hold-destination available, the tracked vehicle takes a peripheral position in a booked hold-destination;
6. CSWR doesn't interfere with Arma's decisions during the group/vehicle journey until the hold-marker position;
7. Once the group/vehicle arrives in position:
  - a. If a group of people: the group changes its formation to a diamond, no matter the group's behavior;
  - b. If a wheeled vehicle: parks and faces the same direction that it has arrived;
  - c. If a tracked-vehicle: is lifted 1 meter, rotated in the desired direction, and then released to remain in the position and direction requested by the editor via *Eden*, avoiding inaccuracies and collisions with rough terrains;
8. While remaining in the hold-position:
  - a. If a group of people: nothing but Arma's decisions until CSWR orders a new position;
  - b. If a wheeled vehicle: nothing but Arma's decisions until CSWR *orders* a new position;
  - c. If tracked vehicle:
    - i. Nothing but Arma's decisions until CSWR *orders* a new position;

- ii. If the vehicle is neutralized and the crew survives, CSWR joins the crew into the first side infantry close enough to them to see and force the survivors to perform the infantry role since;
- 9. After a long time (editable via *fn\_CSWR\_management.sqf*) on hold, the group or vehicle looks for another hold-destination available and in the same sector, unbooking the current marker and booking the next one. From the step 4, repeat everything.

Destination > Specialized > Watch

1. Soon.

Destination > Specialized > Occupy

1. Soon.

## Contribute to the CSWR script

### Discussion on Bohemia Forums

<https://forums.bohemia.net/forums/topic/237504-release-controlled-spawn-and-waypoints-randomizr/>

### Changelog on GitHub

<https://github.com/aldolammel/Arma-3-Controlled-Spawn-And-Waypoints-Randomizr-Script#changelog>

## Author

Based in Porto Alegre, Brazil



thy [@aldolammel](#)

### If you care

Just give a like for CSWR on Workshop to spread the word :)

<https://steamcommunity.com/sharedfiles/filedetails/?l=swedish&id=2740912514>