

Don Rudi's



FAC Script

Introduction

The FAC Script was created to provide mission makers with an easy to use tool to create fun FAC missions.

The basic idea is, the player flies around on any map, looking for ground targets – be it ground units or map objects. He can then request an air strike on this target, by selecting an available AI bomber group, the type of ordnance he wants on the target, the amount of ordnance and optionally an attack heading.

He then marks the target with a rocket and the selected AI bombers will attack this mark.

How to request a strike

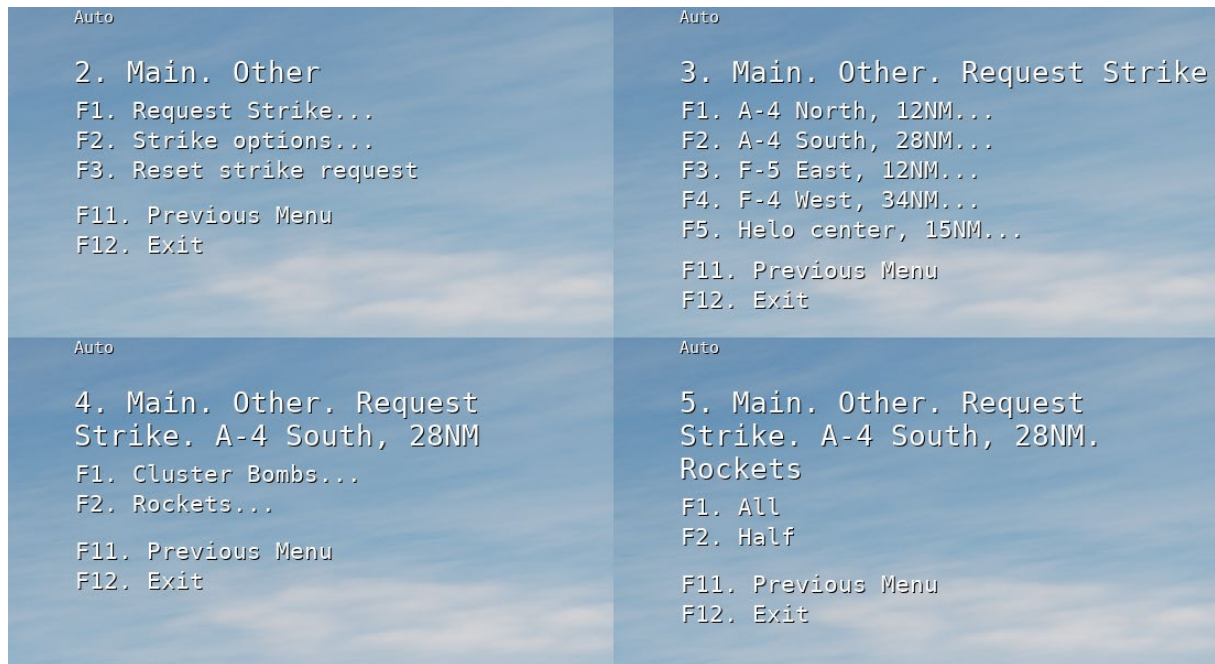
If the player identified a valid target (AI ground units or a map object), he needs to do the following steps:

- Open communication menu
- Select F10 – Other
- Select “Request strike”
- Select one of the available strike groups
- Select the type of ordnance he wants to be dropped on the target.
(currently iron bombs, cluster bombs or rockets can be selected.)
- Select the amount of ordnance to be dropped.
(For bombs: Single, pair, half, all. For Rockets: half, all)

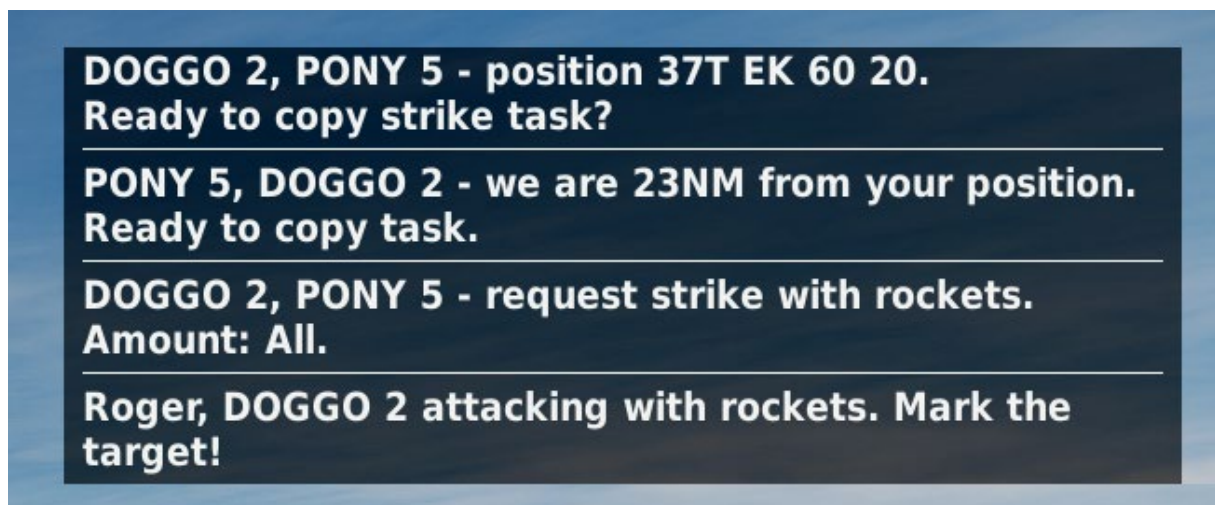
Once these selections were made, the dialogue between FAC and striker will start and the task is set.

The dialogue will now prompt the player to fire a marker rocket at his desired target, which the attackers will engage with the selected options from strike menu.

As an option the player can select an attack heading, from the F10 – Other -> Strike options menu.



The strike request menu



Dialogue between FAC and AI bombers when a task is created

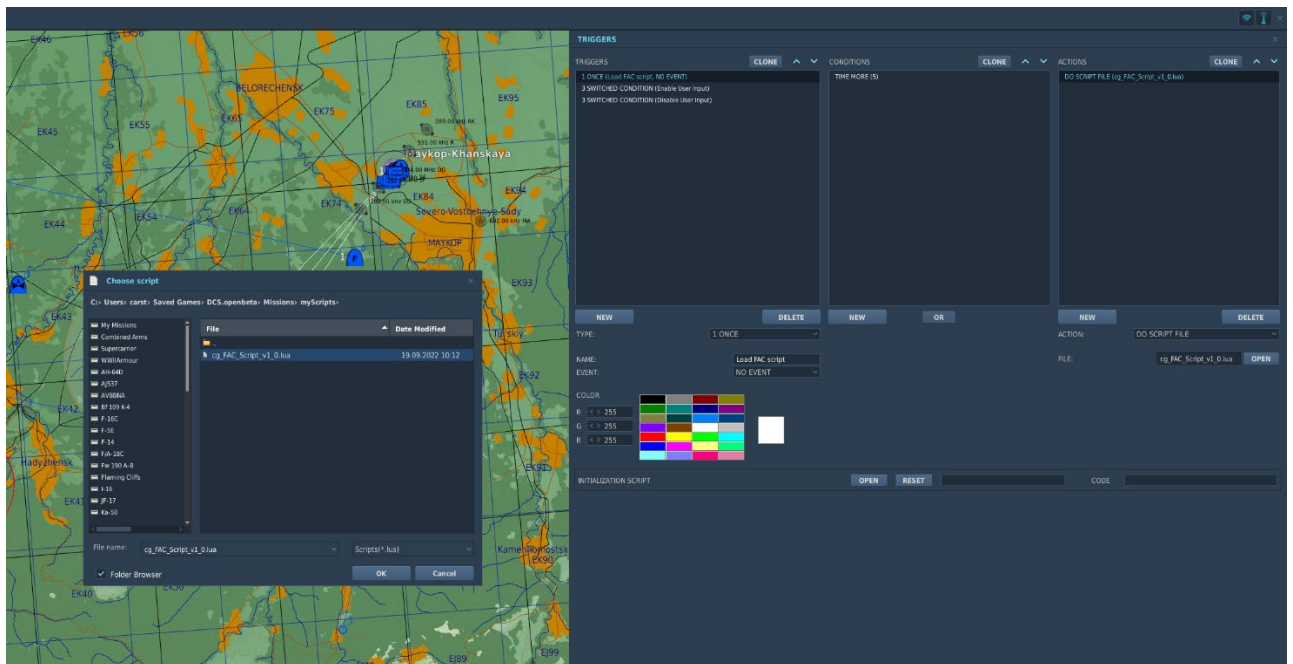
Building the script into a mission

The script needs only one trigger in the mission editor in order to run:

TRIGGER: **ONCE** (best select a name like "load script")

CONDITION: **TIME MORE (x)** with *x* being seconds after mission start. As usually a lot happens at mission start, best load a script after a little while.

ACTION: **DO SCRIPT FILE** and browse to your downloaded `cg_FAC_Script_v1_0.lua`.



Adding player and AI aircraft to the script

Player

The script has two player related variables, the Unit name and the callsign.

```
30 local PLAYER = Unit.getByName('PLAYER')
31 local myCallsign = "PONY 5"
```

The value 'PLAYER' in the brackets needs to be identical with the player's unit name in the mission editor.

The entry for myCallsign can be freely selected. It is not connected to the mission editor and is only used for the on screen messages.

Any aircraft or helicopter can be used, that can fire rockets.

AI Strikers

AI strikers **group name** must follow the naming convention **BOMBER1**, **BOMBER2**, **BOMBER3**, asf.

Set the **TASK** of the AI group to **GROUND ATTACK**.

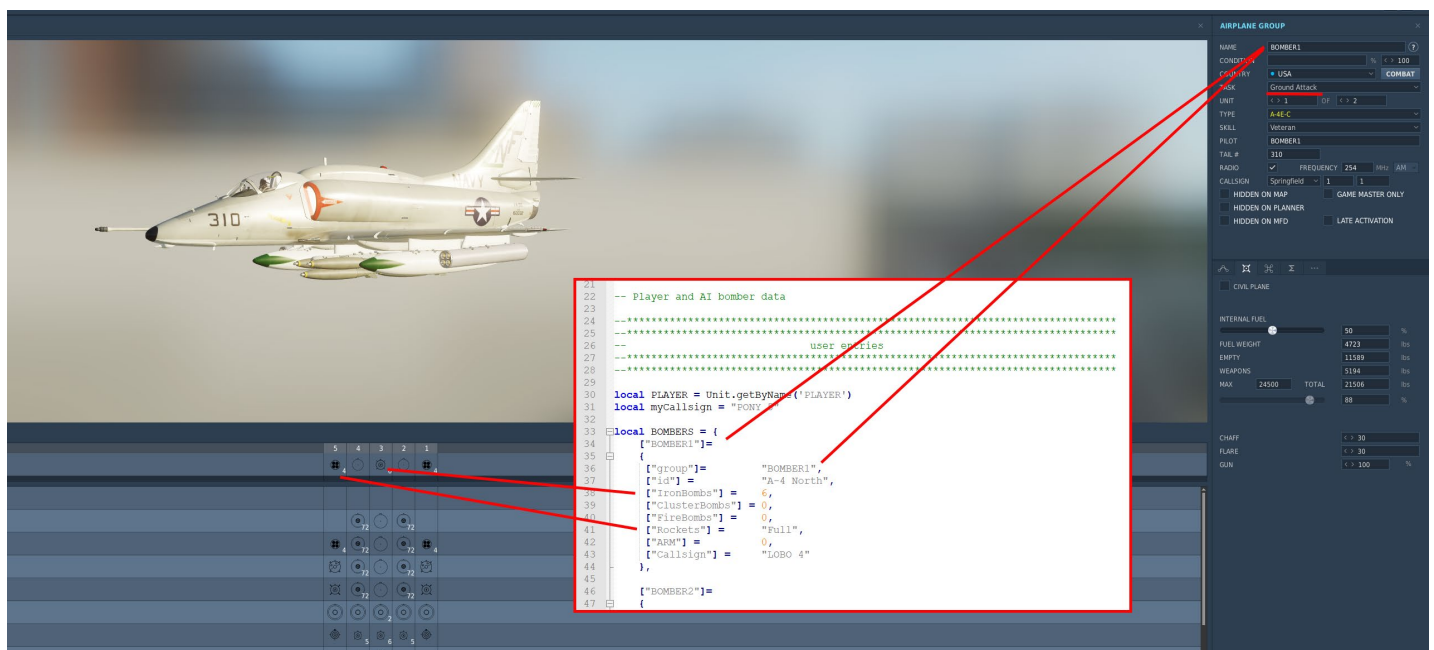
As loadouts you can select:

- Iron bombs (Mk8x series, etc)
- Cluster bombs 8CBUX series, etc)
- Rockets (best use heavy rockets like ZUNI, as light rockets sometimes are not recognized for the ground attack task)*

Use identical loadouts within a group!

Note: Fire bombs (Mk77 series) are supported by the script, but their release is coded incorrectly in DCS, so the AI will release them overhead the target and not in a way to hit the target.

Select any AI aircraft that supports the task of ground attack.



*) There only seems to be a problem, when using light/Hydra rockets on mod aircraft like the A-4, with standard ED aircraft or regular modules they seem to work fine. However, as I cannot guarantee for them, use them at your own risk.

Adding AI striker details to the script

```
53 ["BOMBER1"]=
54 {
55   ["group"]="BOMBER1",
56   ["id"]="A-4 North",
57   ["IronBombs"] = 6,
58   ["ClusterBombs"] = 0,
59   ["FireBombs"] = 0,
60   ["Rockets"] = "Full",
61   ["ARM"] = 0,
62   ["Callsign"] = "LOBO 4",
63   ["orbitWP"] = 1
64 },
```

["BOMBER1"] needs to be the Group name as set in the mission editor.

["group"] should be the same as the Group name.

["id"] Is the name displayed in the F10 menu. Can be chosen freely.

["IronBombs"] enter the number of iron bombs the lead aircraft carries.

["ClusterBombs"] enter the number of cluster bombs the lead aircraft carries.

["FireBombs"] see above, but not recommended to be used.

["Rockets"] if the aircraft has rockets enter "Full", if it has no rockets enter "nil"

["ARM"] currently not used

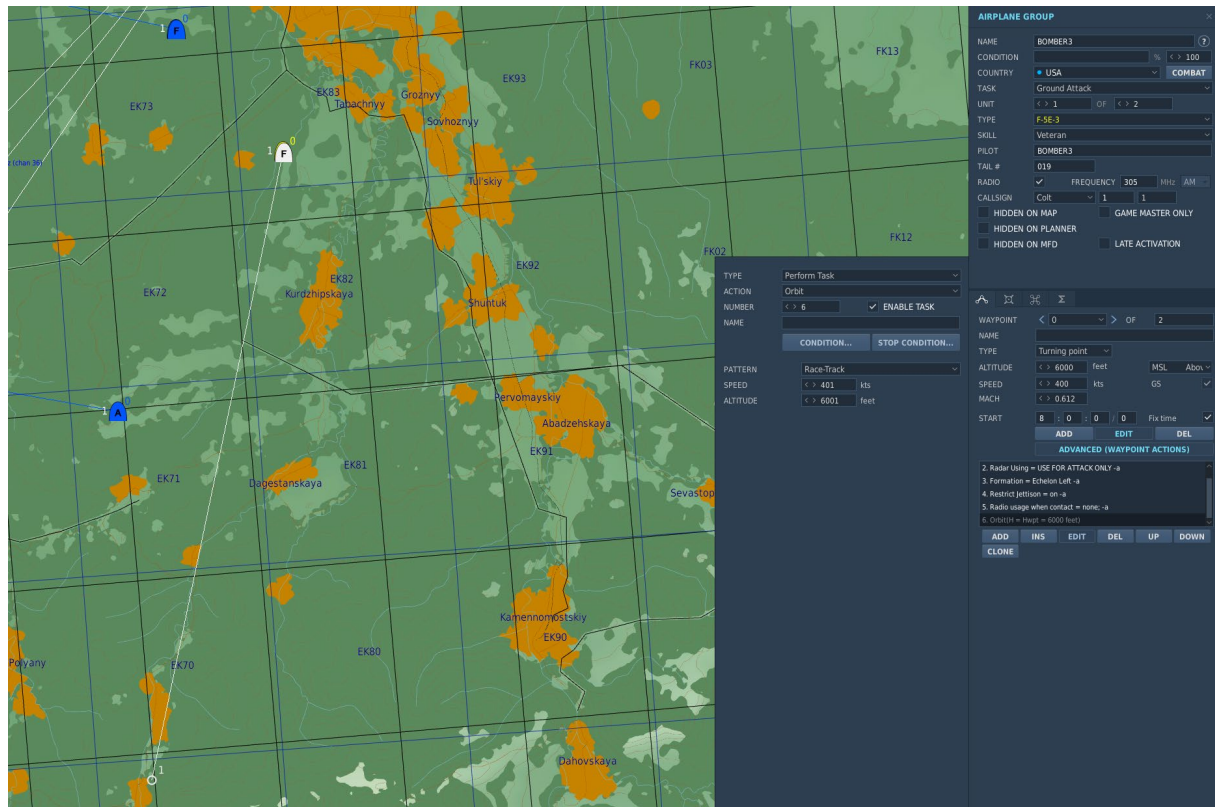
["Callsign"] used for on screen messages, choose freely.

["orbitWP"] is the point at which the group has the advanced waypoint action **ORBIT**. This point is needed to send the strike group back to its holding pattern after a strike. Note: in the script this is the waypoint number in sequence, thus 1st, 2nd, 3rd, etc. In the editor the same waypoint might have a different name/number. For instance, air starts begin with waypoint 0 – as this is the 1st waypoint, please enter a 1 in the script.

Copy and paste the above block, incase more AI groups are needed.

AI strikers flight routes

Basically the AI strikers can be placed anywhere on the map. However, it is recommended to put them into a holding pattern, from which you can request them and which they will automatically rejoin after they fulfilled your task.



Select a waypoint and add an **ADVANCED WAYPOINT ACTION** and select **PERFORM TASK** and then **ORBIT**. Altitude will be set automatically, set the speed as you wish.

After a strike, the group will always return to their initial orbiting waypoint automatically.

Outlook

The script will be enhanced over time, as both new features arrive in DCS or the free OV-10 mod or simply the author's lua skills increase.

Updates

Version 1.4 (30.09.2022):

- Nonexistent ai attackers lead to nil script error: fixed
- Added 2 mile call for attackers, so the FAC doesn't need to stare at his smoke for the whole time waiting for the attackers to arrive
- Reworked the reset strike request logic to prevent hiccups

Imprint

cg_FAC_Script_v1_4.miz

version 1.4

09/2022

By Don Rudi

Enjoy!