Simple Persistent Ground Groups (SPGG)

SPGG is a script that uses "lfs" and "io" to save ground & Naval groups/units in missions to a file. Mission design is important to avoid the same units replicating every restart. See example mission files for one of many ways to design your mission.

The script support most of the CTLD (Complete Transport and Logistics Deployment - https://forum.dcs.world/topic/120045-complete-transport-and-logistics-deployment-ctld/) functionality. Some code is reused from CTLD to make it compatible with CTLD.

It is not recommended to use this script if you are not aware of the risks of using "lfs" and "io" in DCS lua scripts. Scripts can access your filesystem if you enable LFS! This functionality is off by default in DCS to protect the players from malicious code.
Use at your own risk!

Features:
- Save all ground units on the map in intervals or by a trigger you want.
- Saves backup files with ingame timestamps.
- Support most of CTLD functionality. Saves CompleteAASystems (Can repair SAM sites made by CTLD after mission restart)[1]
- **MIST** is now not required to run the script. (See SPGG.lua : spgg.useMIST)
- FLAG and LUA variable for tracking if the mission is loaded in a persistent state.
- Reuse Group and Unit Id's.

Does not support :
- AI tasking or routes

[1] You can build a samsite, restart the mission/server and repair it with the number of units alive when SPGG saved. If units are dead when SPGG saves the SAM site and you restart the server. You can only repair the units that were saved alive.

Example:
BUK site has 3 launchers, 1 CP and 1 Radar.
If 1 Launcher is killed during the mission, you can repair the BUK site with all the Launchers.
But if you do not repair it and the mission/server restart with that 1 launcher dead, it will only have 2 Launchers after this point.

For mission setup example see the included .miz file (SPGG_Example_v0xx_type.miz)

TABLE OF CONTENT

Settings in SPGG.lua

- **spgg.defaultDrive** : Save directory! (**Please see the examples in SPGG.lua and .miz files**)
- **spgg.saveFilename** : Save File location.
- **spgg.backupSaveDirName** : What subfolder to save backup saves.
- **spgg.enableBackupSaves** : Enable or disable backup saves - Values true or false.
- **spgg.Savetime** : How many minutes between the scheduled saving of units runs.
- **spgg.ReuseGroupNames** : Uses the original group names from the save file - Values true or false.
- **spgg.ReuseUnitNames** : Uses the original unit names from the save file - Values true or false.
- **spgg.ReuseID** : Uses the original unit and group id from the save file - Values true or false.
- **spgg.saveOnlyActiveGroups** : Used to exclude groups that are deactivated - Values true or false.
- **spgg.showEnvinfo** : Shows information in dcs.log about events - Values true or false.
- **env.setErrorMessageBoxEnabled** : Regular DCS error dialog that will pause dcs - Values true or false.

- **spgg.useMIST** : Use MIST for groupID and userID on spawning units - Values true or false
If you do not use MIST this must be value: false

- **spgg.excludeGroupNameTbl** : All groups names beginning with strings in this array will be excluded from saveing.
- **spgg.includeStaticObjectTypeTbl**: All Static Objects with type names in this array will be included when saveing.

How to setup DCS with SPGG:

Step 1 - Removing LFS and IO sanitation

It is not recommended to use this script if you are not aware of the risks of using "lfs" and "io" in DCS lua scripts. Scripts can access your filesystem if you enable LFS! This functionality is off by default in DCS to protect the players from malicious code.
Use at your own risk!

Enable "**lfs**" and "**io**" in DCS Game Folder ->
disk:\DCS World OpenBeta\Scripts\MissionScripting.lua

Change from:
```
do
   sanitizeModule('os')
   sanitizeModule('io')
   sanitizeModule('lfs')
  _G['require'] = nil
  _G['loadlib'] = nil
   _G['package'] = nil
End
```

Change to:
```
do
   sanitizeModule('os')
   --sanitizeModule('io')
   --sanitizeModule('lfs')
  _G['require'] = nil
  _G['loadlib'] = nil
   _G['package'] = nil
end
```

Step 2 - Where to install SPGG

**Installing SPGG has changed from previous versions (v020 and below) !**

Copy the content of this zip file directly to your DCS "Scripts" folder. This folder is in your "DCS profile folder. (If you have a different profile location put it there)

The default location Open Beta: %userprofile%\Saved Games\DCS.openbeta\Scripts\
The default location Stable: %userprofile%\Saved Games\DCS\Scripts\

Please use the default location if you are not an advanced user!

How the folder should look like:
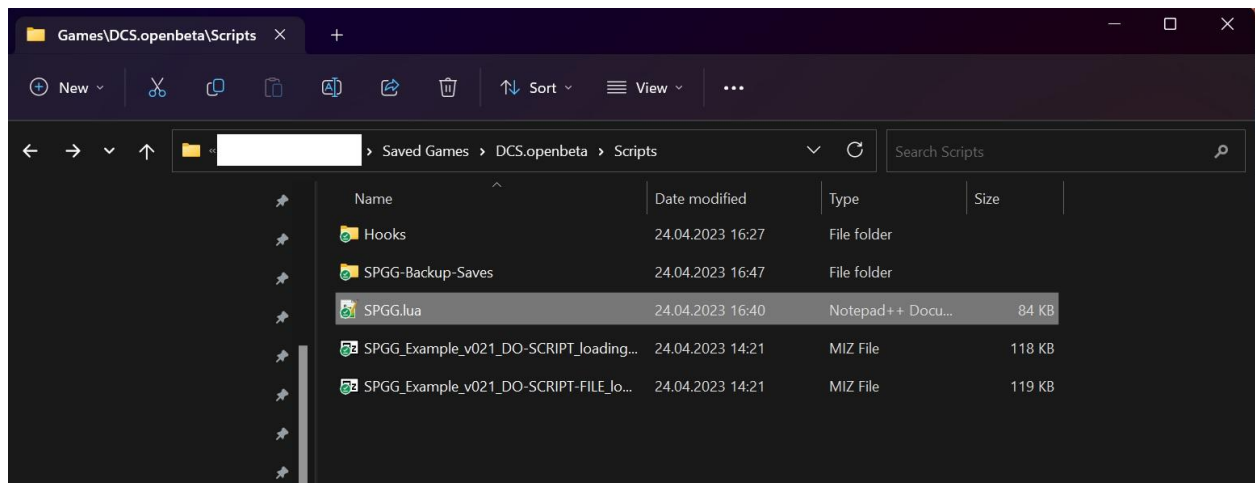
If you are changing the path to a different location than your DCS profile path, please change
SPGG.lua to the location you want. **(Se Example 1 and 2 below)**
See line : spgg.defaultDrive = lfs.writedir() .. [[Scripts\]]

Example 1:
If you change the last part of this script line to "SPGG\" like this :
local spgg.defaultDrive= lfs.writedir() .. [[Scripts\SPGG\]]

The script will try and find the script files in the path: "<your DCS profilepath>\Scripts\SPGG\"
The code "lfs.writedir()" will always get the current DCS profile your server or game is using.
If you run multiple instances of DCS servers, you do not need to change this line! It will find the path to the server's profile!

Example 2:
If you remove "lfs.writedir()" from the script line like this:
local spgg.defaultDrive= [[C:\myscripts\SPGG\]]

The script will look in "C:\myscripts\SPGG2\" for the script files.

Step 3 - How to Load SPGG in to your mission

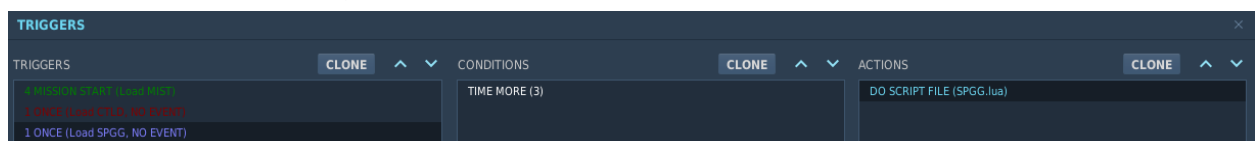## Option 1 (EASY) : Importing the script file SPGG.lua in to the mission

Important: If you change any setting in SPGG.lua, you need to add the file to your mission again!

Add Trigger :
"**ONCE**" -> "**TIME MORE (5)**" -> "**DO SCRIPT FILE**"
Select "**SPGG.lua**" with the "**Open**" button.

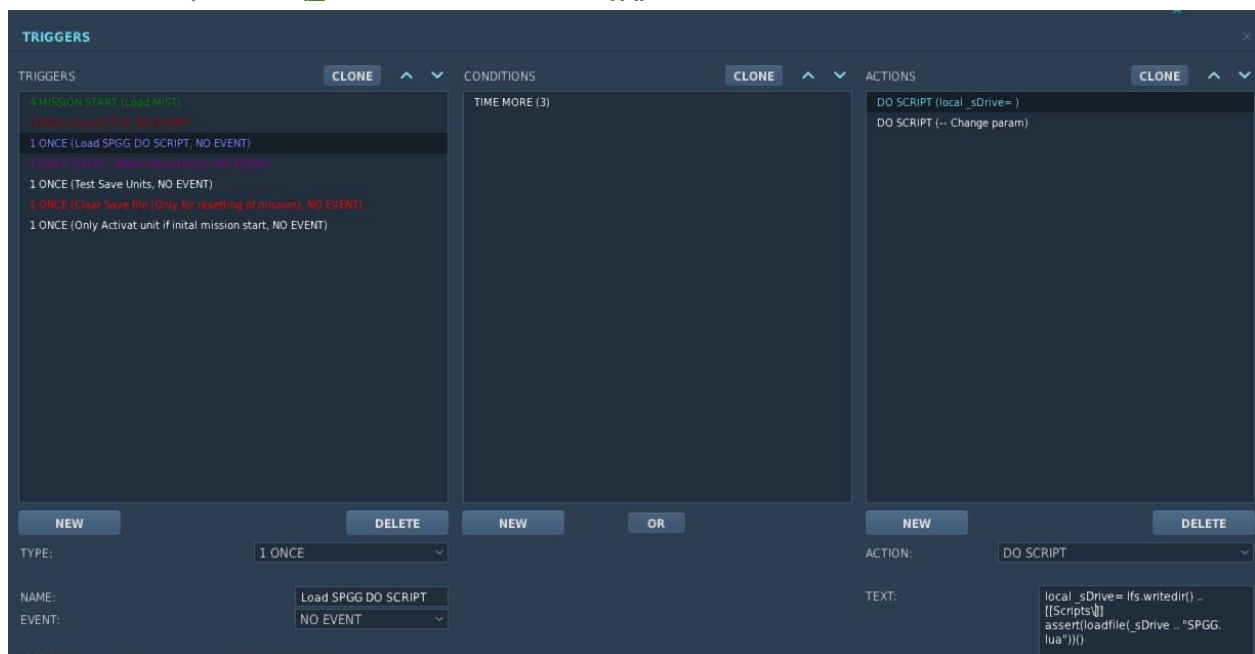If **MIST** and **CTLD** is used, load SPGG later then MIST and CTLD.

Option 2 (ADVANCED): Load the SPGG.lua script directly from the folder

Add Trigger :
"**ONCE**" -> "**TIME MORE (3)**" -> "**DO SCRIPT**"
 Using this code (if you install SPGG a in different folder, change the folder path):

```
local _sDrive= lfs.writedir() .. [[Scripts\]]
assert(loadfile(_sDrive .. "SPGG.lua"))()
```

Step 4 - How to SPAWN units in your mission

Add Trigger :
"**ONCE**" -> "**TIME MORE (8)**" -> "**DO SCRIPT**"
Set the "TIME MORE" after the loading of SPGG.lua into the mission!

Copy the functions under into "**DO SCRIPT**" :
spgg.spawnBlueGroundGroup()
spgg.spawnRedGroundGroup()
spgg.spawnNeutralGroundGroup()
spgg.spawnBlueStaticObject()
spgg.spawnRedStaticObject()
spgg.spawnNeutralStaticObject()
spgg.spawnBlueSeaGroup()
spgg.spawnRedSeaGroup()
spgg.spawnNeutralSeaGroup()
spgg.findAndAddSamSystems()
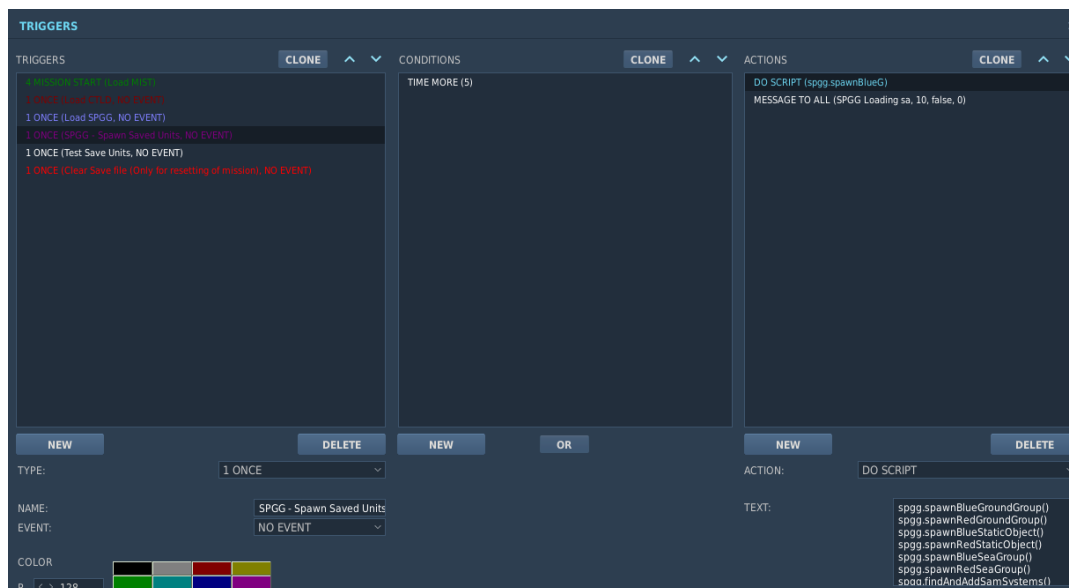
If you example only want Blue units to spawn back, you can change this to:
spgg.spawnBlueGroundGroup()
spgg.spawnBlueStaticObject()
spgg.spawnBlueSeaGroup()
spgg.findAndAddSamSystems()

If you example only want Blue Ground units to spawn back, you can change this to:
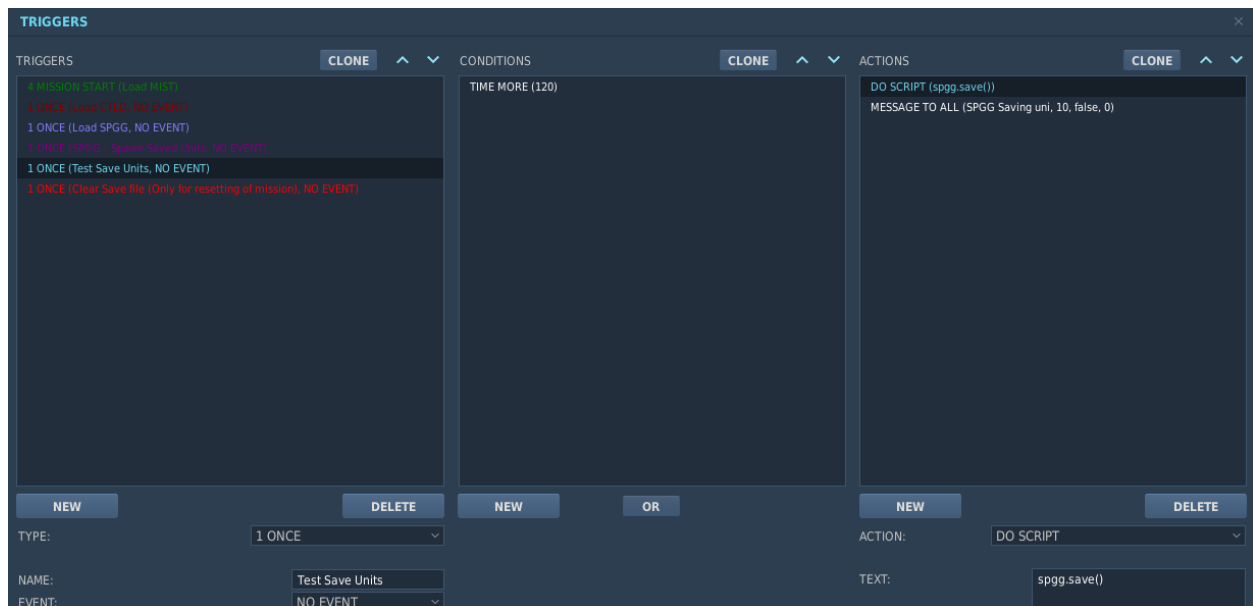**spgg.spawnBlueGroundGroup()**
spgg.findAndAddSamSystems()

Step 5 - Saving groups

The ground groups are saved every 60 min by default. It can be changed from "**SPGG.lua**" by changing :
spgg.Savetime = 60

If you want to save it by other means as well just run a "**DO SCRIPT**" trigger with the function :
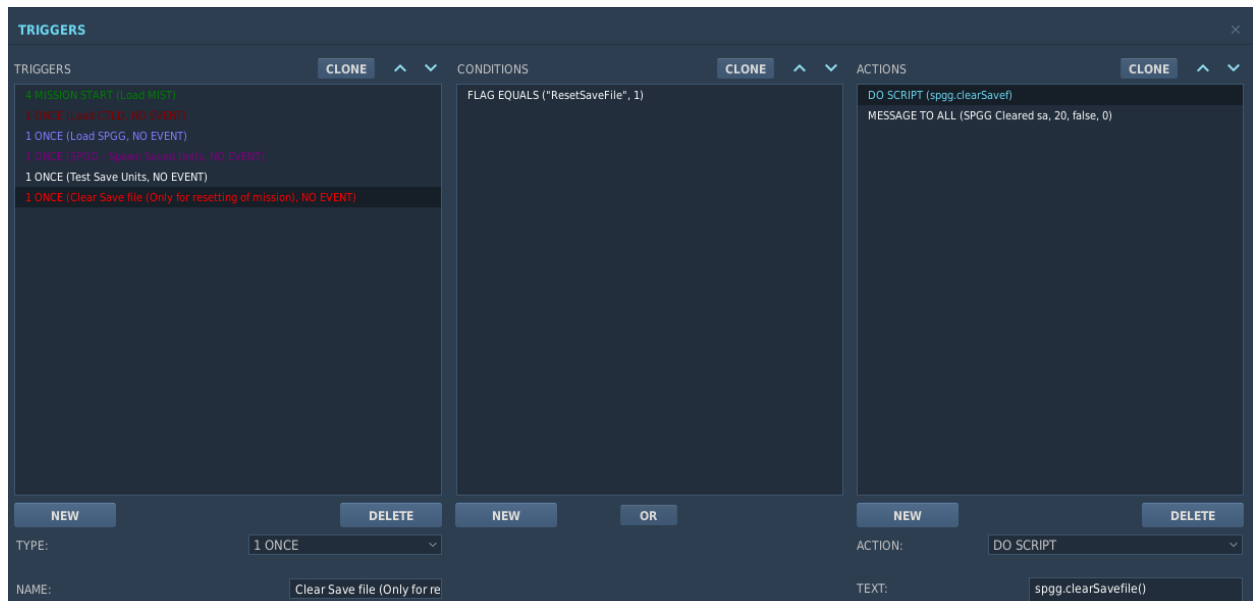**spgg.save()**

Step 5 - How to clear the save file

To clear the save file and stop the save scheduler/loop, run a "DO SCRIPT" with the function :
spgg.clearSavefile()

You can also stop the mission and delete "all" the content of the save file or just delete it.

(Used to Reset mission to initial state after mission is complete or you want to reset the mission)
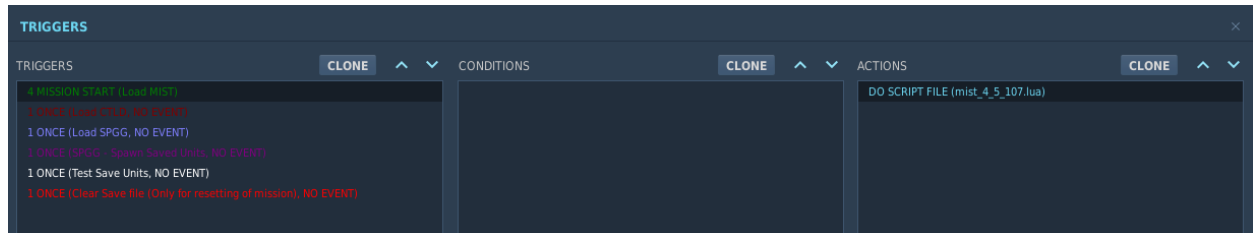

For mission setup example see the included .miz file (SPGG_Example_v0xx_type.miz)

(Optional) How to load MIST in to your Mission

Script: https://forum.dcs.world/topic/120045-complete-transport-and-logistics-deployment-ctld/

Load MIST into your mission with the "**MISSION START**" trigger.

(Optional) How to load CTLD in to your Mission

Script: https://forum.dcs.world/topic/82178-mission-scripting-tools-mist-enhancing-mission-scripting-lua/

If you are using CTLD, Load CTLD "before" the SPGG script and "after" MIST.
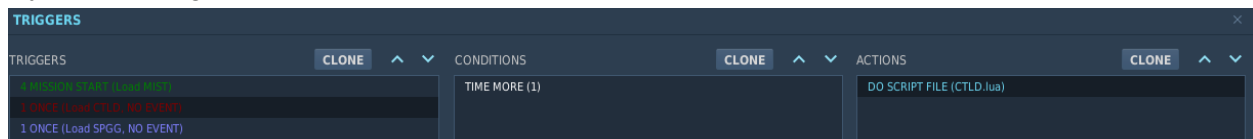
(Optional) Change CTLD to use MIST Group/Unit ID

If you are using CTLD (**MIST version**) it is recommended that you open the "CTLD.lua" file and change some lines to avoid GroupID and UnitID duplicates:
(It is recommended to also use spgg.useMIST = true in SPGG.lua if you change this!)

Change line 1501 and 1509 from :

```
ctld.nextUnitId = 1;
ctld.getNextUnitId = function()
        ctld.nextUnitId = ctld.nextUnitId + 1

        return ctld.nextUnitId
end

ctld.nextGroupId = 1;

ctld.getNextGroupId = function()
        ctld.nextGroupId = ctld.nextGroupId + 1

        return ctld.nextGroupId
```

Change line 1501 and 1509 to :

```
ctld.nextUnitId = 1;
ctld.getNextUnitId = function()
        ctld.nextUnitId = mist.getNextUnitId()

        return ctld.nextUnitId
end

ctld.nextGroupId = 1;

ctld.getNextGroupId = function()
        ctld.nextGroupId = mist.getNextGroupId()

        return ctld.nextGroupId
```

Troubleshooting

Check Dcs.log file under path:
 %userprofile%\Saved Games\DCS.openbeta\Logs\

1. **Error messages:**
   [string "local _sDrive= lfs.writedir() .. [[Scripts\]]..."]:1: attempt to index global 'lfs' (a nil value)

   Cause: LUA's "LFS" and/or "IO" module is sanitized in DCS.

   Solution:
   See "Step 1 - Removing LFS and IO sanitation"

2. **Error messages:**
   can't open 'C:\Users\<username>\Saved Games\DCS.OpenBeta\Scripts\SPGG\SPGG_Load_vxxx.lua'

   can't open 'C:\Users\<username>\Saved Games\DCS.OpenBeta\Scripts\SPGG\SPGG_Save_vxxx.lua'

   can't open 'C:\Users\<username>\Saved Games\DCS.OpenBeta\Scripts\SPGG\SPGG_savefile.lua'

   Cause: Folder or files does not exist in the path given in SPGG.lua

   Solution:
   Check SPGG.lua for the line (Can be different if you changed it):
   spgg.defaultDrive = lfs.writedir() .. [[Scripts\SPGG\]]

   Check if this path is the same as where your files are located!
   Examples of how to change this are included above the line.

   Filenames if changed also need to be the same in SPGG.lua and the folder:
   spgg.saveFunctionsFilename = "SPGG_Save_v020.lua"
   spgg.loadFunctionsFilename = "SPGG_load_v020.lua"
   spgg.saveFilename = "SPGG_savefile.lua"

   See "Step 2" & "Step 3" or example .miz files provided!