



TRI Gaming

Mission Edit start guite

ミッションエディットスタートガイド

本ガイドでは初めてミッションエディットを行う方のために、エディット前の準備から実際にミッションを完成させるまでをサポートします。

ミッションを作ろうとされているあなたの頭の中には、完成させたい理想のミッションがあると思います。ですが少し待ってください。初めから複雑なミッションを作ろうとしても完成前に挫折してしまうでしょう。まずは他のミッションを参考に、シンプルなミッションを完成させてみましょう。そしてミッションの仕組みを理解できるようになれば、きっと理想のミッションが作れるようになるでしょう。

目次

1 事前の準備

1.1 PBOファイルの解体

2 サンプルミッションの確認

2.1 init.sqf

2.2 scriptsフォルダ

3 エディット

3.1 エディター画面

3.2 ミッションを作り変える

3.3 マップの変更

3.4 保存しよう！

3.5 ユニットを配置してみる

3.6 waypoint

3.7 オブジェクトの置き換え

3.8 タスクの書き換え

3.9 General設定

3.10 gear.sqf

3.11 arsenal.sqf

3.12 敵の変更

3.13 微調整

4 完成！

1 事前の準備

実際にミッションを作る前に、次のものを用意しましょう。

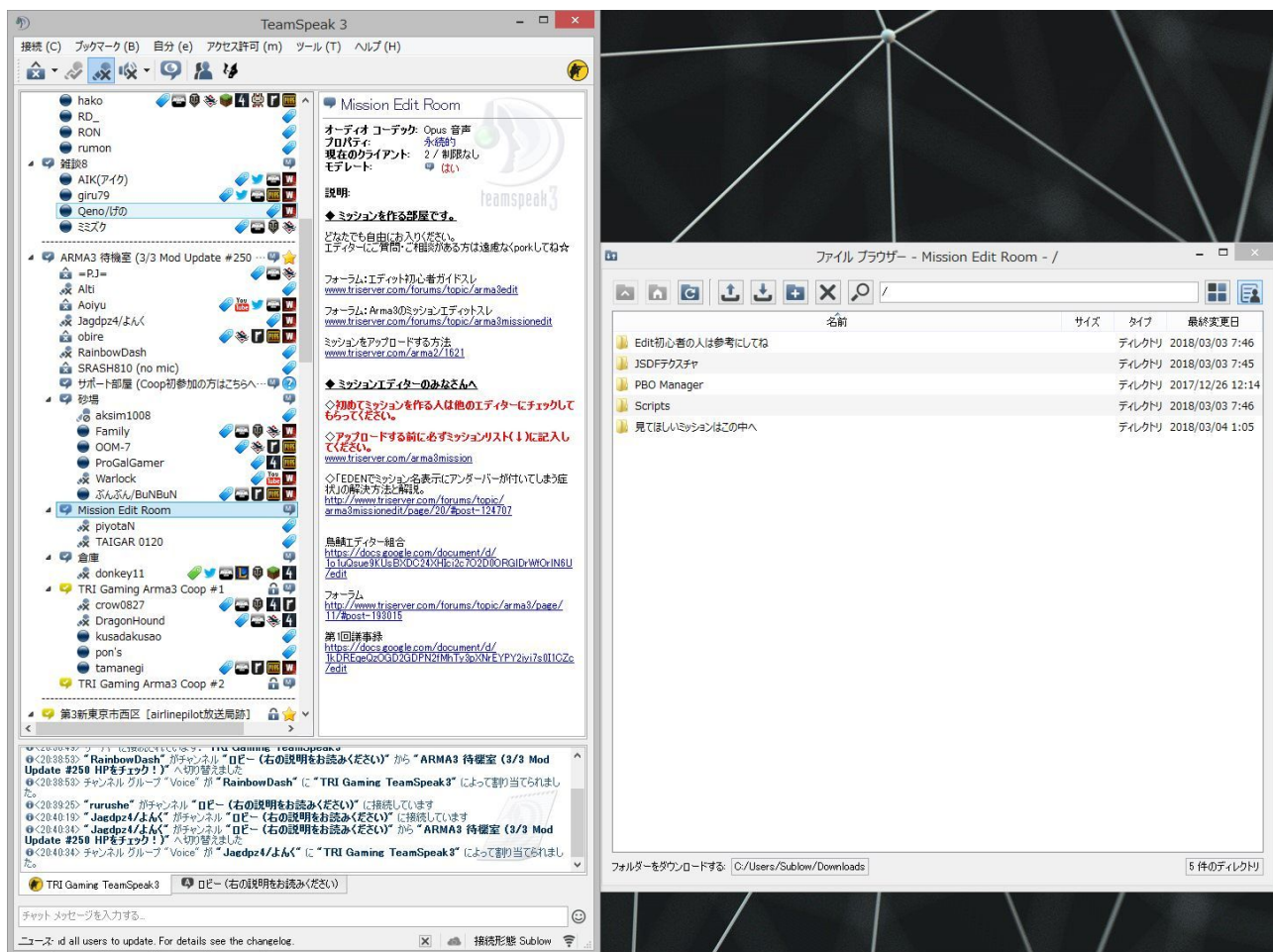
・PBOManager

Arma3で使われるミッションファイル形式であるPBOファイルを解体したりできるツールです。下記URLからDLできます。

<http://www.armaholic.com/page.php?id=16369>

・サンプルミッション

TS3のMission Edit Roomのファイルブラウザーにサンプルミッションがあります。これをDLしましょう。



TS3,ファイルブラウザー

Mission Edit Roomを右クリックするとファイルブラウザーを開けます。
『Edit初心者の人は参考にしてね』フォルダを開き、
『[[JP]_Co10_simple_1.Altis.pbo』をDLしてください。

1.1 PBOファイルの解体

サンプルミッションをDLしましたが、PBOファイルのままではエディターで開く事はできません。先にDLしたPBOManagerを使って解体しましょう。



右がPBOファイル、左が解体したもの

1.pboファイルを右クリック＞PBO Manager＞Extract toの一番下を押す。

2.同名のフォルダが生成されます。

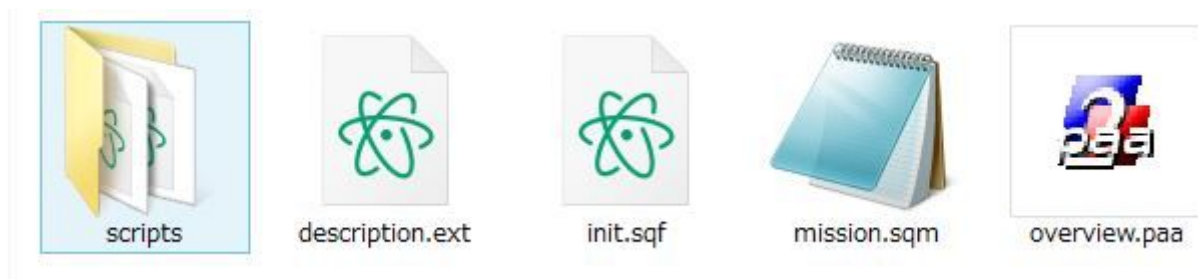
3.解凍したフォルダを下記のmissionsフォルダに移す。

C:\ユーザー\Name\ドキュメント\Arma 3 - Other Profiles\Name\missions

ミッションエディターで作成したミッションもここに保存されます。

2 サンプルミッションの確認

サンプルミッションのフォルダを開いてみましょう。下記のフォルダ、ファイルが入っているはずです。



解体されたフォルダの中身

scripts

ミッション中で呼び出されるスクリプトが入っています。

description.ext

タスク・ミッションの達成・失敗時の動作などが記述されています。

init.sqf

ミッション開始時に呼び出されるスクリプトです。タスクやブリーフィングの設定の他、開始時から呼び出したいスクリプトもここに記述します。

mission.sqm

ミッションエディタ内で作った内容が保存されてます。これを書き換えてしまうとミッショ

ンが開かなくなったりするので注意しましょう。

overview.paa

ミッション選択時や読み込み時に表示される画像です。

2.1 init.sqf

テキストエディタなどでinit.sqfを開いてみましょう。

```
1  if (!isDedicated) then {waitUntil {!isNull player && isPlayer player}};
2
3  enableRadio false;
4  tawvd_disablenone = true;
5  player addRating 10000;
6  enableSaving [false,false];
7  player setVariable ["ace_medical_medicClass", 1, true];
8
9  /* Task */
10 task1 = player createSimpleTask ["Destroy ammo cache"];
11 task1 setSimpleTaskDescription ["<marker name='icon1'>Ifestiona</marker>にある弾
  * 薬箱を破壊せよ。", "Destroy ammo cache", "Destroy ammo cache"];
12 player setCurrentTask task1;
13
14 /* Briefing */
15 player createDiaryRecord ["Diary", ["Medical", "ACE Medical System"]];
16 player createDiaryRecord ["Diary", ["Info", "ゲリラが秘匿した弾薬を破壊せよ。"]];
17
```

使用エディタはAtom

色々書いてあってよくわからないと思いますが、10行目以降を見てください。(テキストエディタは行数がでるものが便利です)

10行目から12行目はこういったタスクなのかを記述しています。弾薬箱の破壊ミッションだとわかりますね。15行目から16行目はブリーフィング内容の記述です。これらは共にゲーム内で直接確認できる部分です。後でこれらも書き換えてみましょう。

2.2 scriptsフォルダ

続いてscriptsフォルダを開いてみましょう。



scriptsフォルダの中身

ammo_box.sqf

弾薬箱の中身を設定しています。

arsenal.sqf

アーセナルの機能を持たせるスクリプトです。アーセナルで選択できる武器などをこれで設定できます。

ending.sqf

タスクの完了後、このスクリプトを呼び出す事でミッション終了としています。

gear.sqf

初期装備の設定をしています。味方にも敵にも使えます。

teleportToSix.sqf

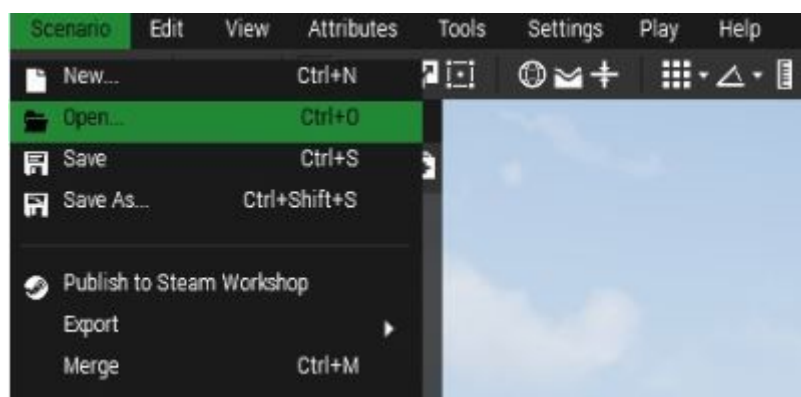
指揮官にテレポートするスクリプトです。

3 エディット

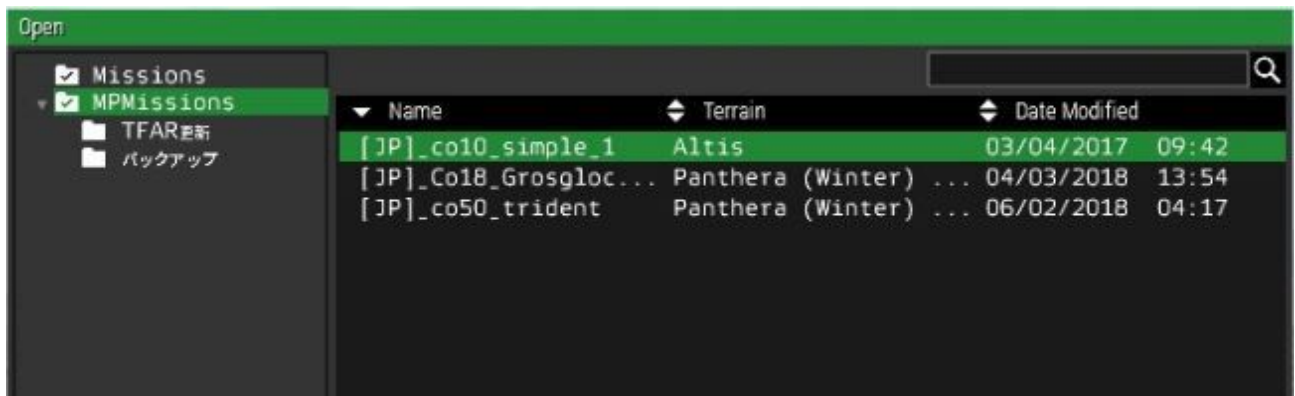
※ミッションを作る時は、サーバープリセット以外のMODを入れずにARMA3を起動してください。プリセット以外のMODが入っていた場合正常に動作しなくなります。

ARMA3を起動し、EDITORを選択してください。マップの選択画面になるので、とりあえずAltisを選択してください。(どこでもいいです)

エディター画面左上の『Scenario』からOpen>[JP]_co10_simple_1...を開きます。



エディター画面左上



1.1で保存場所を間違っていないければサンプルミッションがあるはずです

3.1 エディター画面



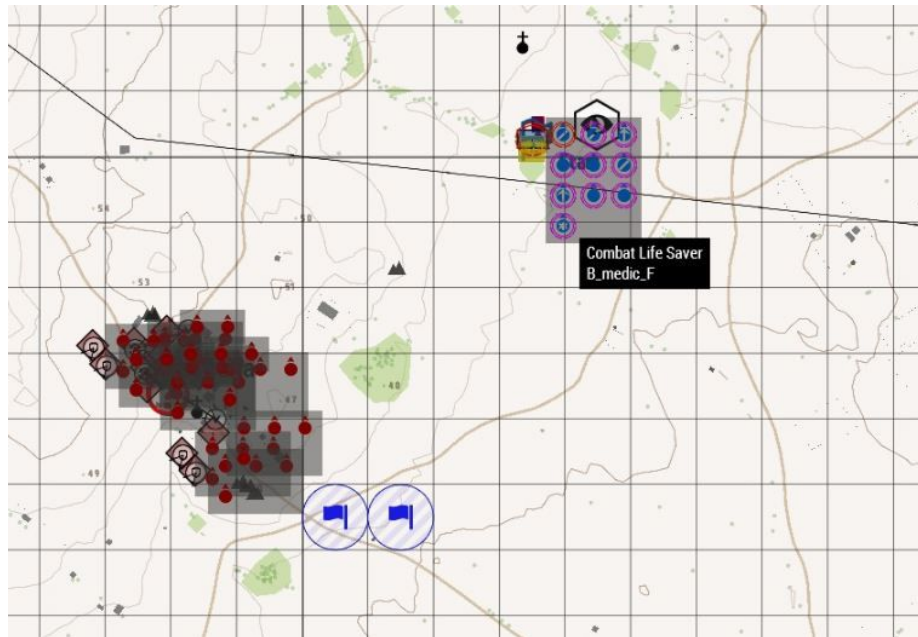
エディター内

正常にミッションフォルダが開けていれば上の画面になるはずです。画面左のカラムは配置済みのユニット・オブジェクトで、右のカラムから置きたいユニット・オブジェクトを選択して配置できます。ピンクもしくは赤い丸がついたユニットがミッションでプレイ可能なものです。敵側のCSATを見ると丸がついていないのがわかります。移動していくと、破壊対象の弾薬箱も確認できると思います。

3.2 ミッションを作り変える

このミッションは、AltisでCSATが守る弾薬箱をNATOが破壊するミッションです。今回はこれを土台に、Maldenでロシア兵が守る自走対空砲をアメリカ海兵隊が破壊するミッションに作り替えます。まずは、すでに配置されたオブジェクトやトリガー（青い旗のアイコン）をまるっとコピーして、Maldenに配置します。

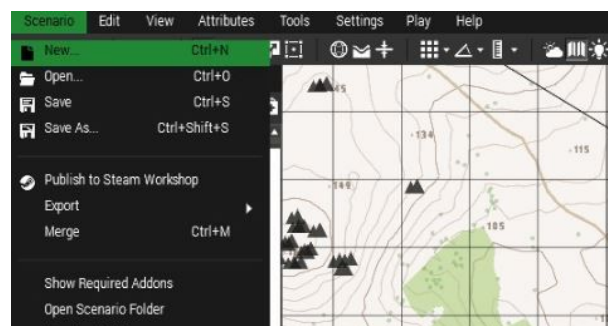
Mキーを押しマップ画面にしたら、全オブジェクトを選択してコピー(Ctrl+C)します。



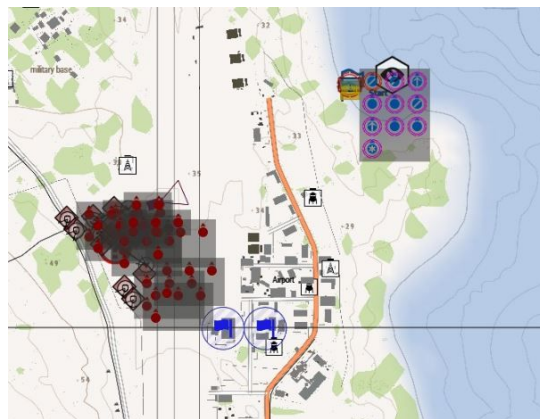
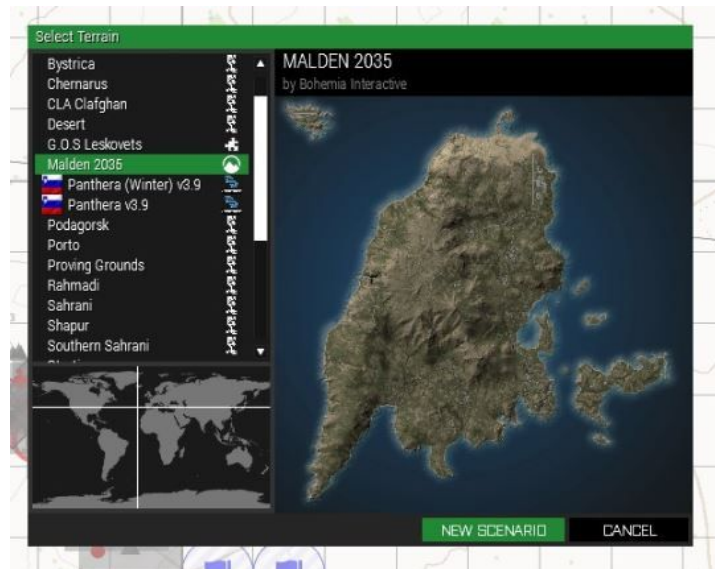
全部コピーしましょう

3.3 マップの変更

コピーしたらScenario>New>Maldenを選択してください。マップが読み込めたら適当な位置に貼り付けましょう。(Ctrl+V)



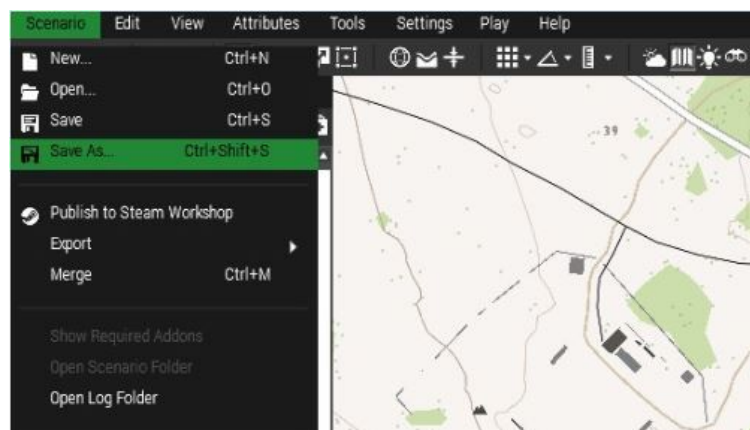
新しいミッションを作る時は『New』から

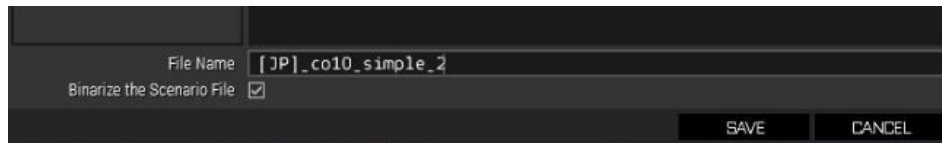


北西部の空港に置いてみました

3.4 保存しよう！

新しいマップへの貼り付けが終わったら、まず保存しましょう。





読みやすくてカッコいい名前を付けましょう

※保存する際はサーバーの命名規則に従ってください。

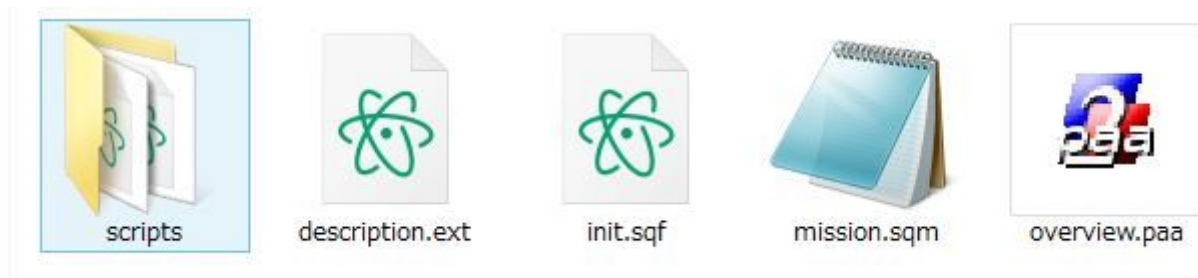
ここで付けた名前がPBOファイル名になります。

例)[JP]_Co10_simple_2

- ・ スペースではなく_(アンダーバー)で区切る
- ・ 10未満の場合は Co08 のように記す
- ・ バージョンなどは書かない(更新する際に上書きできるように)
- ・ 半角英数字のみ、記号は使わない
- ・ 大文字小文字は問いません

新しく名前を付けて保存したので、DLしたsimple_1ミッションとは別のフォルダができました。ここで注意したいのは、**scripts**フォルダや**init.sqf**などは引き継がれていないということです。なので、simple_1からsimple_2に**mission.spm以外**をコピーしましょう。

overview.paaは画像ファイルなので、今回のミッション用に別途用意しましょう。コピーしなくて大丈夫です。

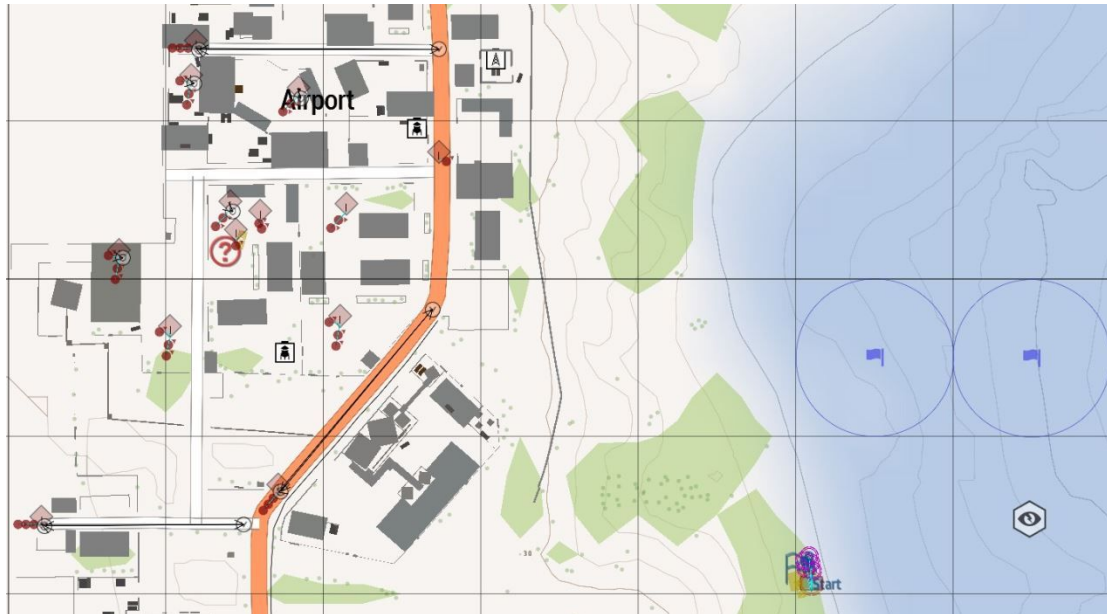


※mission.spmは入れちゃダメですよ！



3.5 ユニートを配置してみる

適当なユニットを選んでドラッグしてみてください。shiftを押しながらで方向転換、Altを押しながらで高さを変えられます。今回はMalden北西部の空港にあるを舞台としました。海側から進入する海兵隊に対し、破壊目標の外側にパトロールを配置し交戦が始まると敵が寄ってくるようになっています。



simple_2も開いてみてください

3.6 waypoint

敵ユニットを見ると黒い線が伸びて白い丸が出ていますよね？これがwaypointです。

よく使われるものに『MOVE』(ポイントまで移動)、『CYCLE』(行動を繰り返す)、『GUARD』(交戦が始まると駆け付ける)などがあります。

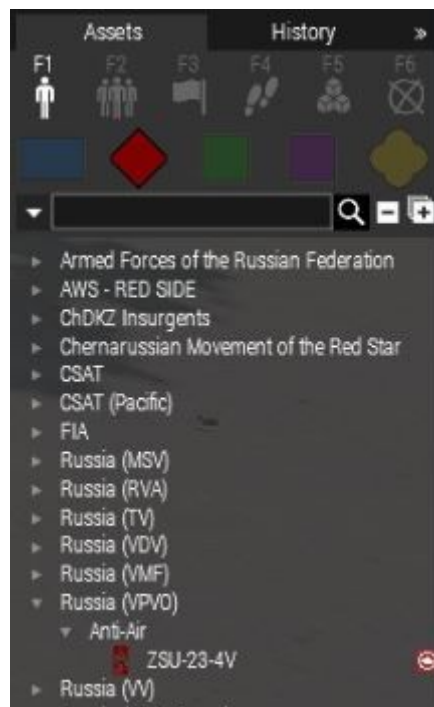
これらを組み合わせて、敵ユニットの動きをコントロールします。



画像のwaypointはGUARDです

3.7 オブジェクトの置き換え

この時点では破壊目標はまだ弾薬箱ですよね。自走対空砲に置き換えましょう。



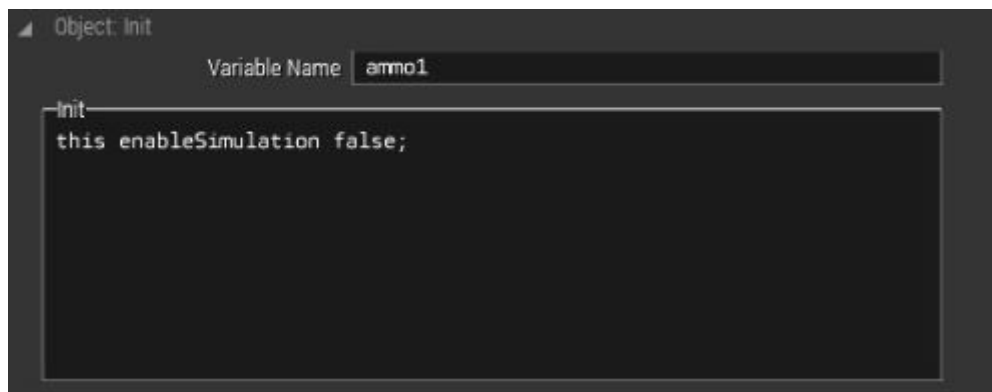
RHS製のものを選びました

右カラムの赤いマーク(OPFOR)からZSU-23-4Vを選び、置きたい場所をクリックします。今回はシンプルに誰も乗っていない車両にしました。Altキーを押しながら選択する事で、無人の車両を出すことができます。検索窓から検索もできますよ。



隣に並べました

次は弾薬箱をダブルクリックしてみてください。



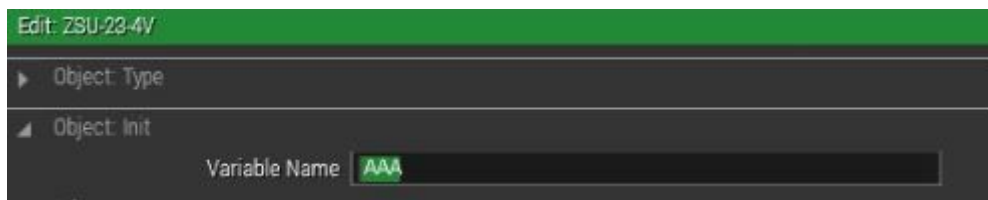
init欄は気にしないで大丈夫です

Variable Name欄を見てください。『ammo1』という名前がついていますね。
続いて並んだトリガー(青い旗のアイコン)の左側をダブルクリックしてみてください。



Trigger

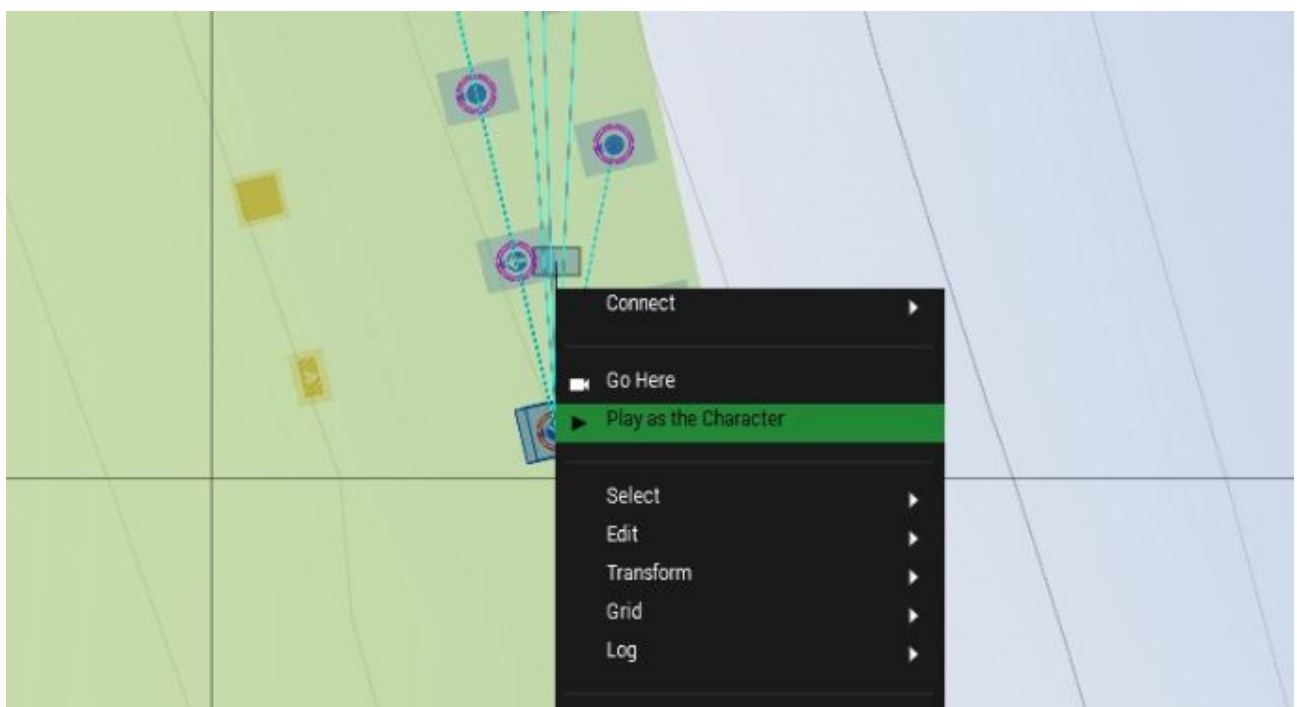
ここにも『ammo1』という名前がありますよね。このCondition欄には「ammo1が生きていない」という条件が記述されてあります。この条件が満たされた時、On Activation欄に記述された動作が発動します。『ammo1』を『AAA』と書き換えましょう。同時に、先ほど置いた自走対空砲のVariable Name欄にも『AAA』と記述します。



自走対空砲のVariable Name欄

さて、ここで再度保存しましょう。保存はこまめに！

意味の分からない英語を追いかけて少し疲れましたよね。息抜きに実際にテストプレイしてみましょう。



グループリーダーでプレイしてください

ユニットを右クリックすると上のようなメニューがでます。ここから『Play as the Character』を選択してください。エディット中はいつでもこの動作でテストプレイする事ができます。

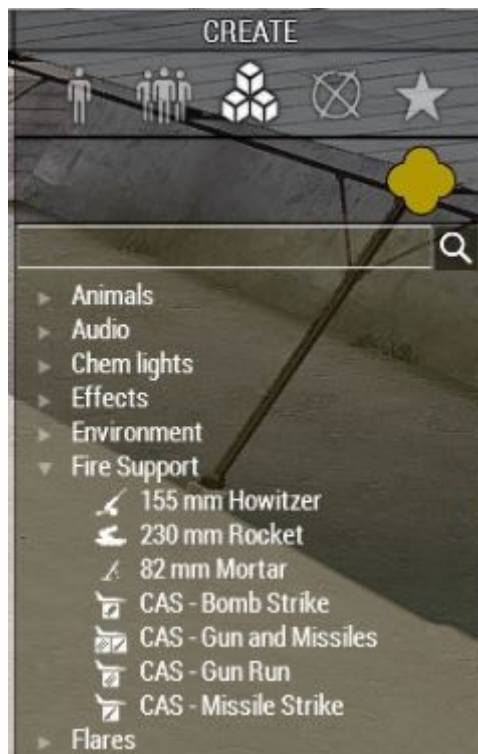
ゲーム内に入ったらShift+Yキーを押してください。Zeusモードになってミッション内を見渡せますか？Zeusモードにならない時はZeusモジュールがサンプルミッションからコピーできていないのかもしれません。確認してみてください。



Zeusモード

ZeusモードでAAAのところまで行きましょう。

右カラムから下画像のアイコンを選び、Fire Support>155mm Howitzerをクリックしてください。クリックした後にゲーム画面内をクリックすることで、その場に155mm砲弾を降らせる事ができます。



Zeus右カラム



クリックすると赤い丸が出てきます



破壊できました

タスクは達成できましたか？ちょっとよく見てください。



タスクは達成できたけど...

『DESTROY AMMO』になってますよね。ここも直さないといけませんね。

3.8 タスクの書き換え

マップを変えて、破壊目標を置き換えるところまでできました。ミッションの細部を修正していきましょう。『Play as the Character』でマップを開きタスクとブリーフィングを確認してください。弾薬箱の破壊のままになってますよね？

ミッションを保存したら、init.sqfを開いてください。

```
init.sqf
1  if (!isDedicated) then {waitUntil {!isNull player && isPlayer player}};
2
3  enableRadio false;
4  tawvd_disablenone = true;
5  player addRating 10000;
6  enableSaving [false,false];
7  player setVariable ["ace_medical_medicClass", 1, true];
8
9  /* Task */
10 task1 = player createSimpleTask ["Destroy ammo cache"];
11 task1 setSimpleTaskDescription ["<marker name='icon1'>Ifestiona</marker>にある弾薬箱を破壊せよ。",
  * "Destroy ammo cache", "Destroy ammo cache"];
12 player setCurrentTask task1;
13
14 /* Briefing */
15 player createDiaryRecord ["Diary", ["Medical", "ACE Medical System"]];
16 player createDiaryRecord ["Diary", ["Info", "ゲリラが秘匿した弾薬を破壊せよ。"]];
17
```

init.sqf

2.1で見た画面ですね。まず『弾薬箱』を『自走対空砲』に書き換えましょう。それから、10～11行目にある『Destroy ammo cache』(ammo1になってますね...)を『Destroy AAA』に書き換えましょう。

```
9  /* Task */
10 task1 = player createSimpleTask ["Destroy AAA"];
11 task1 setSimpleTaskDescription ["<marker name='icon1'>Ifestiona</marker>にある自走対空砲を破壊せよ
  * 。", "Destroy AAA", "Destroy AAA"];
12 player setCurrentTask task1;
```

9～12行目

次は16行目のブリーフィング内容を書き換えます。ミッションによって敵味方の状況なんかが書かれていたりしますよね。今回はシンプルにいきましょう。

```
14 /* Briefing */
15 player createDiaryRecord ["Diary", ["Medical", "ACE Medical System"]];
16 player createDiaryRecord ["Diary", ["Info", "空港に設置された自走対空砲を破壊せよ"]];
17
```

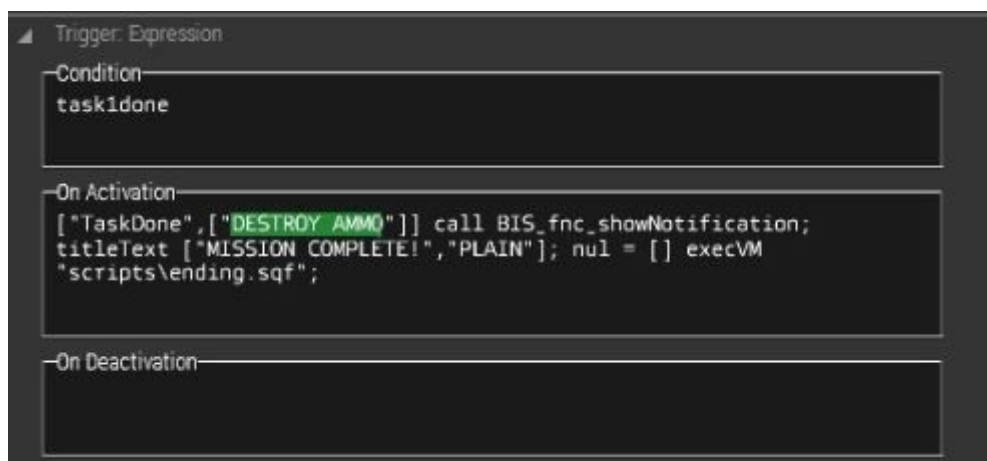
16行目

これでタスク、ブリーフィングの書き換えができました。またミッションエディターからゲーム内で確認してみましょう。



OKですね

3.7で宿題になっていたタスク達成の記述も直しましょう。
これはゲーム内のトリガーで管理されています。
並んでいるトリガーの右側をダブルクリックしてください。



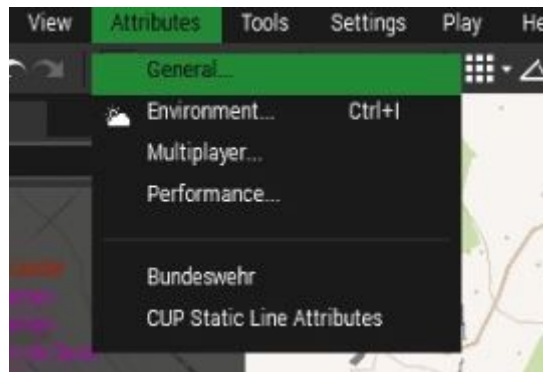
右側のトリガー

On Activation欄に『DESTROY AMMO』と記述がありますね。ここを『DESTROY AAA』に書き換えます。

ところでCondition欄の『task1done』に見覚えはありませんか？
これは左側のトリガーのOn activation欄で宣言されていたものです。
つまり全体の流れとして、**AAAが破壊(!alive AAA)**されると左側のトリガーが発動しtask1doneが宣言され、右側のトリガーが発動します。そして右側のトリガーのOn activation欄にある `nul = [] execVM "scripts\ending.sqf";` という記述でending.sqfが呼び出され、ミッション終了となります。

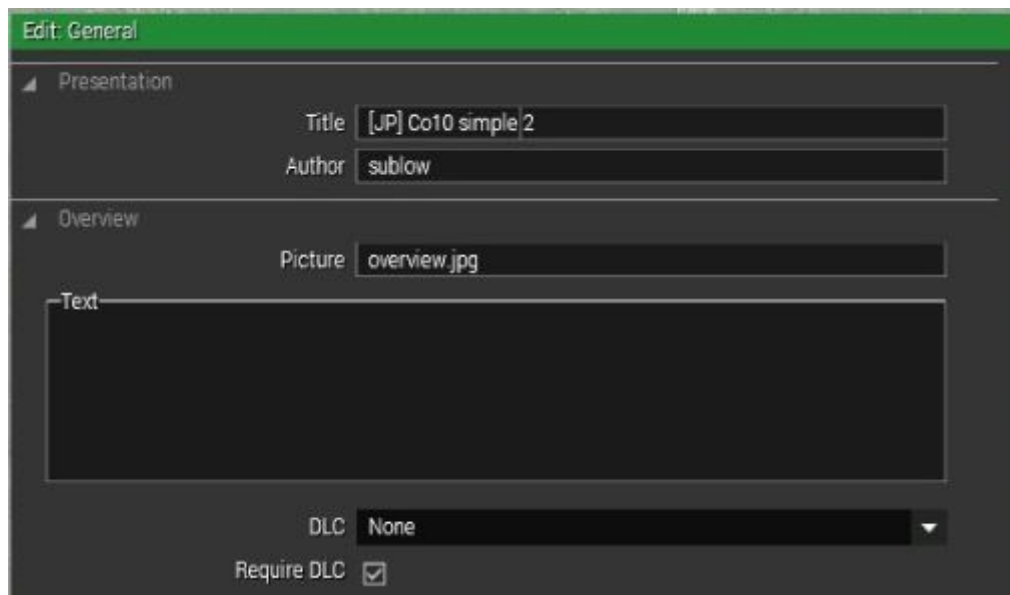
3.9 General設定

ミッションエディター上部のAttributes>Generalを開いてください。



General

下のような画面が開きましたね。ここでも名前をつけないといけません。



Titleをつけましょう

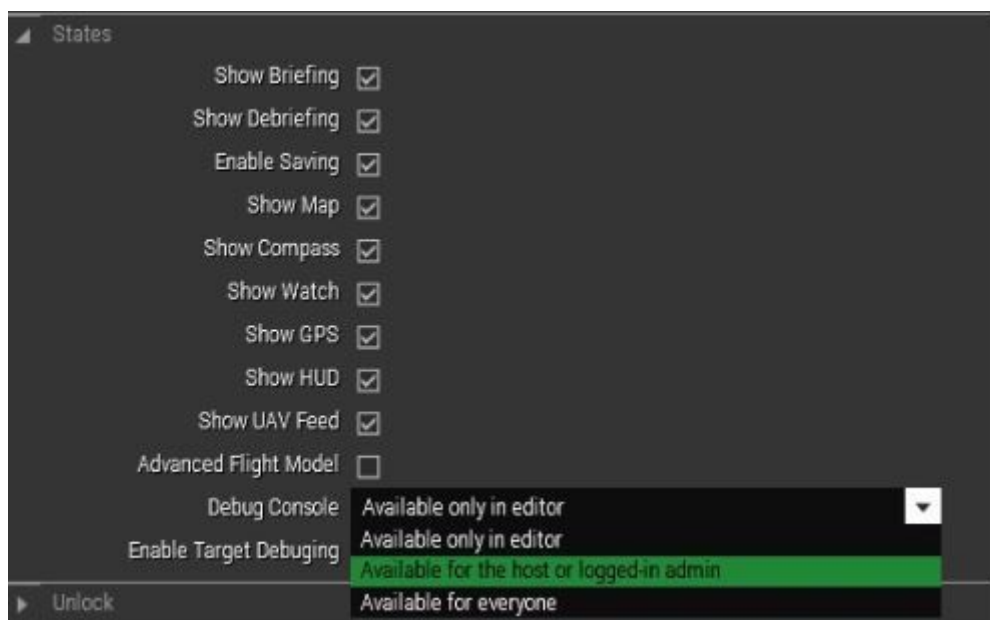
Generalの命名規則は次の通りです。

- ・ 半角スペースで区切る
- ・ 10未満の場合は Co08 のように記す
- ・ 半角英字のみ、日本語は不可です
- ・ 末尾にバージョンなどを書いても構いません

ここで付ける名前はミッション選択画面で表示されるものです。ミッションを保存する時の名前はPBOファイルの名前になります。これらは別のものとして扱われるので注意してください。

Picture欄には適当に名前をつけて、適当な画像に同じ名前をつけてミッションフォルダに入れておきましょう。推奨サイズは長辺1024pixelです。

General>StatesのDebug Consoleをアドミンも開けるようにしておいてください。

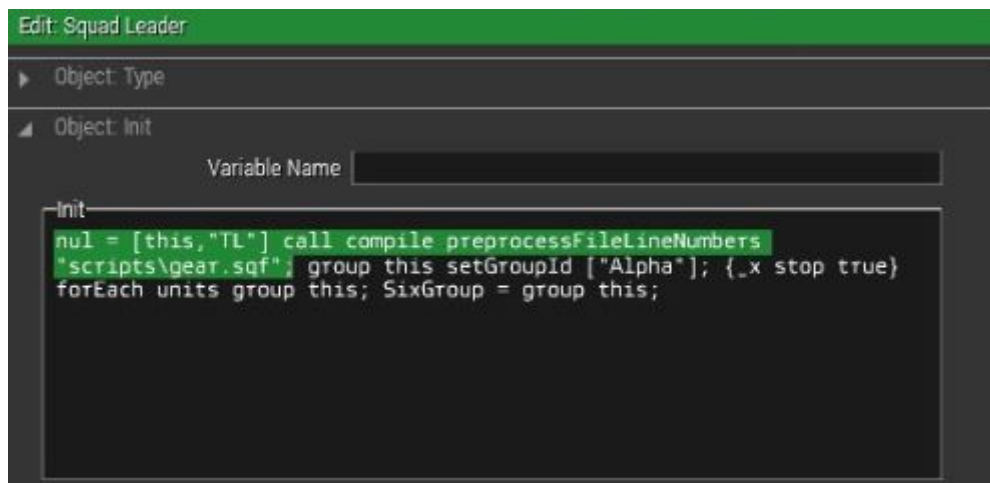


緑色のところを選択しておいてください

3.10 gear.sqf

ユニットの装備をスクリプトで管理するのが**gear.sqf**です。ゲーム内のロードアウト機能で装備を整えることもできますが、後々の変更や装備品の名前が変わった際などgear.sqfを使った方が有利な点があります。

NATOのユニットをダブルクリックしてみてください。



init欄

Init欄に `nul = [this, "TL"] call compile preprocessFileLineNumbers "scripts\gear.sqf";` という記述があります。これが`gear.sqf`を呼び出しています。ミッションを保存してscripts>`gear.sqf`を開いてください。

```
gear.sqf
1  /*
2  Gear setting script by trico
3  _nul = [this, "TL"] call compile preprocessFileLineNumbers "gear.sqf";
4  */
5  private ["_unit", "_type"];
6
7  _unit = _this select 0;
8  _type = _this select 1;
9
10 if(!local _unit) exitWith{};
11 waitUntil {!isNull _unit};
12
13 removeAllWeapons _unit;
14 removeAllAssignedItems _unit;
15 removeBackpack _unit;
16 removeVest _unit;
17 removeUniform _unit;
18 removeHeadGear _unit;
19 removeGoggles _unit;
20
21 _unit linkItem "ItemCompass";
22 _unit linkItem "ItemMap";
23 _unit linkItem "ItemWatch";
24
25 switch (_type) do {
26
27     case "TL":
28     {
29         _unit addWeapon "tf_anprc152";
30         _unit addWeapon "ACE_Vector";
31         _unit addWeapon "arifle_MX_F";
32         _unit addPrimaryWeaponItem "optic_Aco";
33         _unit addHeadgear "H_HelmetB";
34         _unit addBackpack "B_AssaultPack_rgr";
35         clearAllItemsFromBackpack _unit;
36         for "_i" from 1 to 2 do {_unit addItemToBackpack "DemoCharge_Remote_Mag";};
37         for "_i" from 1 to 1 do {_unit addItemToBackpack "ACE_Clacker";};
38         _unit addVest "V_PlateCarrier1_rgr";
39         for "_i" from 1 to 6 do {_unit addItemToVest "30Rnd_65x39_caseless_mag";};
40         for "_i" from 1 to 2 do {_unit addItemToVest "HandGrenade";};
41         for "_i" from 1 to 1 do {_unit addItemToVest "SmokeShell";};
42         for "_i" from 1 to 1 do {_unit addItemToVest "SmokeShellGreen";};
43         for "_i" from 1 to 1 do {_unit addItemToVest "SmokeShellRed";};
44     }
```

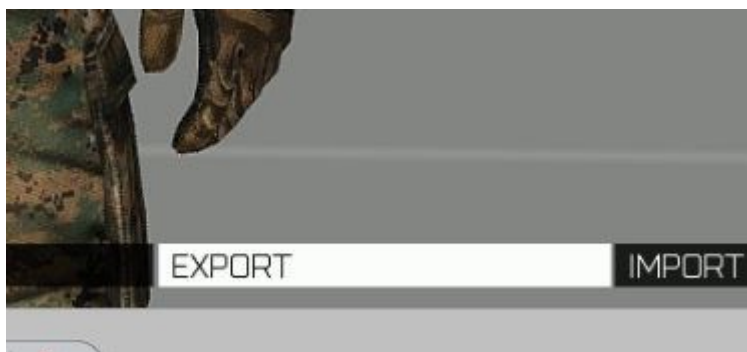
英語がいっぱい並んでいますが落ち着いてください。

27行目をみてください。 **case "TL":** と記述がありますね。 **"TL"** という名前が振られたユニットに28～51行目に記載された装備が割り当てられます。その下も同様にそれぞれの名前にそれぞれの装備が割り当てられています。

このgear.sqfでは“TL”,“AR”,“GR”,“RM”,“MD”と5つの装備が記述されています。
ユニットをクリックすると誰にどの装備が割り当てられているかわかるといいます。
例えば29行目の“tf_anprc152”は無線機のクラスネームです。全ての装備品にクラスネームが設定されていて、そのクラスネームをgear.sqfに書き込むことで装備を作っていきます。

クラスネームの調べ方

ミッションを保存してVIRTUAL ARSENALを起動してください。ここで好きな装備に着替えさせてください。そして下部のEXPORTを押しましょう。



VIRTUAL ARSENAL

これで今装備している装備品のクラスネームがクリップボードにコピーされます。
これを適当なメモ帳などに貼りつけましょう。

```
1  comment "Exported from Arsenal by Sublow";
2
3  comment "Remove existing items";
4  removeAllWeapons this;
5  removeAllItems this;
6  removeAllAssignedItems this;
7  removeUniform this;
8  removeVest this;
9  removeBackpack this;
10 removeHeadgear this;
11 removeGoggles this;
12
13 comment "Add containers";
14 this forceAddUniform "rhs_uniform_FROG01_wd";
15 for "_i" from 1 to 10 do {this addItemToUniform "ACE_fieldDressing";};
16 this addItemToUniform "ACE_EarPlugs";
17 for "_i" from 1 to 2 do {this addItemToUniform "ACE_epinephrine";};
18 for "_i" from 1 to 2 do {this addItemToUniform "ACE_morphine";};
19 for "_i" from 1 to 2 do {this addItemToUniform "ACE_tourniquet";};
20 this addVest "rhsusf_spc_teamleader";
21 for "_i" from 1 to 5 do {this addItemToVest "rhs_mag_30Rnd_556x45_M855A1_Stanag";};
22 this addItemToVest "SmokeShellGreen";
23 this addItemToVest "SmokeShell";
24 for "_i" from 1 to 2 do {this addItemToVest "HandGrenade";};
25 this addHeadgear "rhsusf_lwh_helmet_marpatwd_headset";
```

新規のファイルに貼り付け

英語が並んでいます但し落ち着いてください。
14行目からクラスネームが記載されています。

```
this forceAddUniform "rhs_uniform_FROG01_wd";
```

これはユニフォームのクラスネームです。
15行目ではユニフォームに入れたfieldDressing(包帯)のクラスネームが取得できます。
このようにかなりアナログな方法ですが、1点1点気長に取得していきましょう。

“TL”の装備を次のように書き換えました。(赤字が変更点)

```
case "TL":
{
    _unit addWeapon "tf_anprc152";
    _unit addWeapon "ACE_Vector";
    _unit addWeapon "rhs_weap_m4a1_blockII";
    _unit addPrimaryWeaponItem "rhsusf_acc_ACOG_USMC";
    _unit addHeadgear "rhsusf_lwh_helmet_marpatwd_headset";
    _unit addBackpack "rhsusf_falconii_coy";
    clearAllItemsFromBackpack _unit;
    for "_i" from 1 to 2 do {_unit addItemToBackpack "DemoCharge_Remote_Mag"};
    for "_i" from 1 to 1 do {_unit addItemToBackpack "ACE_Clacker"};
    _unit addVest "rhsusf_spc_teamleader";
    for "_i" from 1 to 6 do {_unit addItemToVest "rhs_mag_30Rnd_556x45_M855A1_Stanag"};
    for "_i" from 1 to 2 do {_unit addItemToVest "HandGrenade"};
    for "_i" from 1 to 1 do {_unit addItemToVest "SmokeShell"};
    for "_i" from 1 to 1 do {_unit addItemToVest "SmokeShellGreen"};
    for "_i" from 1 to 1 do {_unit addItemToVest "SmokeShellRed"};
    _unit forceAddUniform "rhs_uniform_FROG01_wd";
    for "_i" from 1 to 10 do {_unit addItemToUniform "ACE_fieldDressing"};
    for "_i" from 1 to 2 do {_unit addItemToUniform "ACE_morphine"};
    for "_i" from 1 to 2 do {_unit addItemToUniform "ACE_epinephrine"};
    for "_i" from 1 to 2 do {_unit addItemToUniform "ACE_tourniquet"};
    for "_i" from 1 to 1 do {_unit addItemToUniform "ACE_EarPlugs"};
    for "_i" from 1 to 1 do {_unit addItemToUniform "ACE_microDAGR"};
};
```

武器に合わせたマガジンを持たせるようにしましょう。
また、TLなのでVecter(双眼鏡)とmicroDAGR、そしてAAA破壊用の爆弾を持たせてあります。兵科によって持たせる装備を調整しましょう。

for “_i” from 1 to 2 do...というのはそのアイテムを何個持たせるかという設定です。例えば上の gear.sqfでいうと“rhs_mag_30Rnd_556x45_M855A1_Stanag”を6本持たせています。

末尾の;抜けに気を付けて全兵科仕上げましょう。



裸の人がいないかチェック

Edit Loadout

gear.sqfを使わずにユニットの装備を変更するのが**Edit Loadout**機能です。
エディタ内で装備を変更させたいユニットを右クリックし、**Edit Loadout**を選択します。



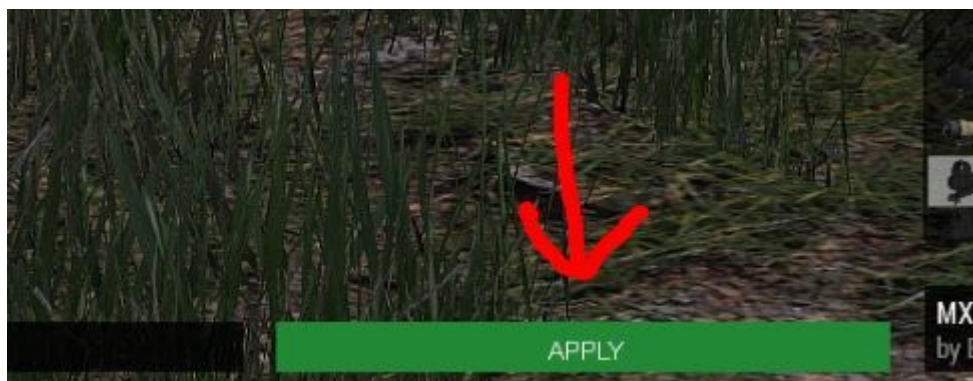
Edit Loadout

Edit Loadoutを開くと**ACE ARSENAL**画面になります。

ここで通常のアーセナル同様装備を変更できますが、まず**初めに全装備を外し全裸にしてください**。ポン置きのまま装備変更を進めた場合、ミッション中に意図していない装備が追加される可能性があります。

全裸で装備プリセットを保存しておくと呼び出すのに便利です。

装備が整ったら右下のAPPLYを押して完了です。



Escなどで閉じると装備が反映されません

クラスネームを拾う手間が無いので楽な反面、**クラスネームが変更になった際には全ユニットの装備を1点ずつ変更する**必要があります。クラスネームの変更はMODの更新で起こります。

また、装備品を見落としやすい(耳栓が無いなど)ので、注意してください。

銃と弾の組み合わせにも注意してください。

3.11 arsenal.sqf

全装備品が入ったアーセナルの設定は簡単なのですが、装備品を限定しようとする**とクラスネームの指定**が必要です。

scripts>**arsenal.sqf**を開いてみましょう。


```

arsenal.sqf
1  /*
2  Arsenal whitelist by trico
3  nul = [this] execVM "arsenal.sqf";
4  */
5  if (!isServer) exitWith {};
6
7  _box = _this select 0;
8
9  _aWeapons = [
10 /* Rifle */
11 "arifle_MX_F",
12 "arifle_MX_Black_F",
13 "arifle_MXC_F",
14 "arifle_MXC_Black_F",
15 "arifle_MX_GL_F",
16 "arifle_MX_GL_Black_F",
17 /* Machinegun */
18 "arifle_MX_SW_F",
19 "arifle_MX_SW_Black_F",
20 /* Pistol */
21 "hgun_ACPC2_F"
22 ];
23
24 _aMags = [
25 /* Magazine */
26 "30Rnd_65x39_caseless_mag",
27 "100Rnd_65x39_caseless_mag",
28 "9Rnd_45ACP_Mag",
29 /* Grenade */
30 "HandGrenade",
31 "SmokeShell",
32 "SmokeShellBlue",
33 "SmokeShellGreen",

```

arsenal.sqf

バニラの装備が入っています。必要なければ消してしまいましょう。

銃、弾、アイテム(光学機器)、服、ベスト、ヘルメットの欄に海兵隊装備を入れましょう。バックパック等はアーセナルで全員が装備できてもいいのか、ミッションによって判断しましょう。面倒であれば他のエディターが作ったgear.sqfやarsenal.sqfをそのまま流用すると楽です。ただし中身は必ず確認しておきましょう。

gear.sqf、arsenal.sqfともに問題があればミッション開始時にエラーが出るはずです。sqf何行目にエラーがあるか表示される事もあるので確認しておきましょう。

3.12 敵の変更

敵がCSATのままなのでロシア人に変更しましょう。

すでに置いてある敵と同数のユニットをだして、CSATを消しましょう。

敵が40mmグレネードを持っていたりすると1発で全員死んでしまうかもしれないので注意しましょう。サブレッサーを装備しているかどうかにも注意してください。



ロシア兵になりました

敵が交戦に呼び寄せられて破壊目標の場所に誰も残らない場合は、敵グループリーダーのinit欄に

[x disableAi "PATH"] forEach units group this;

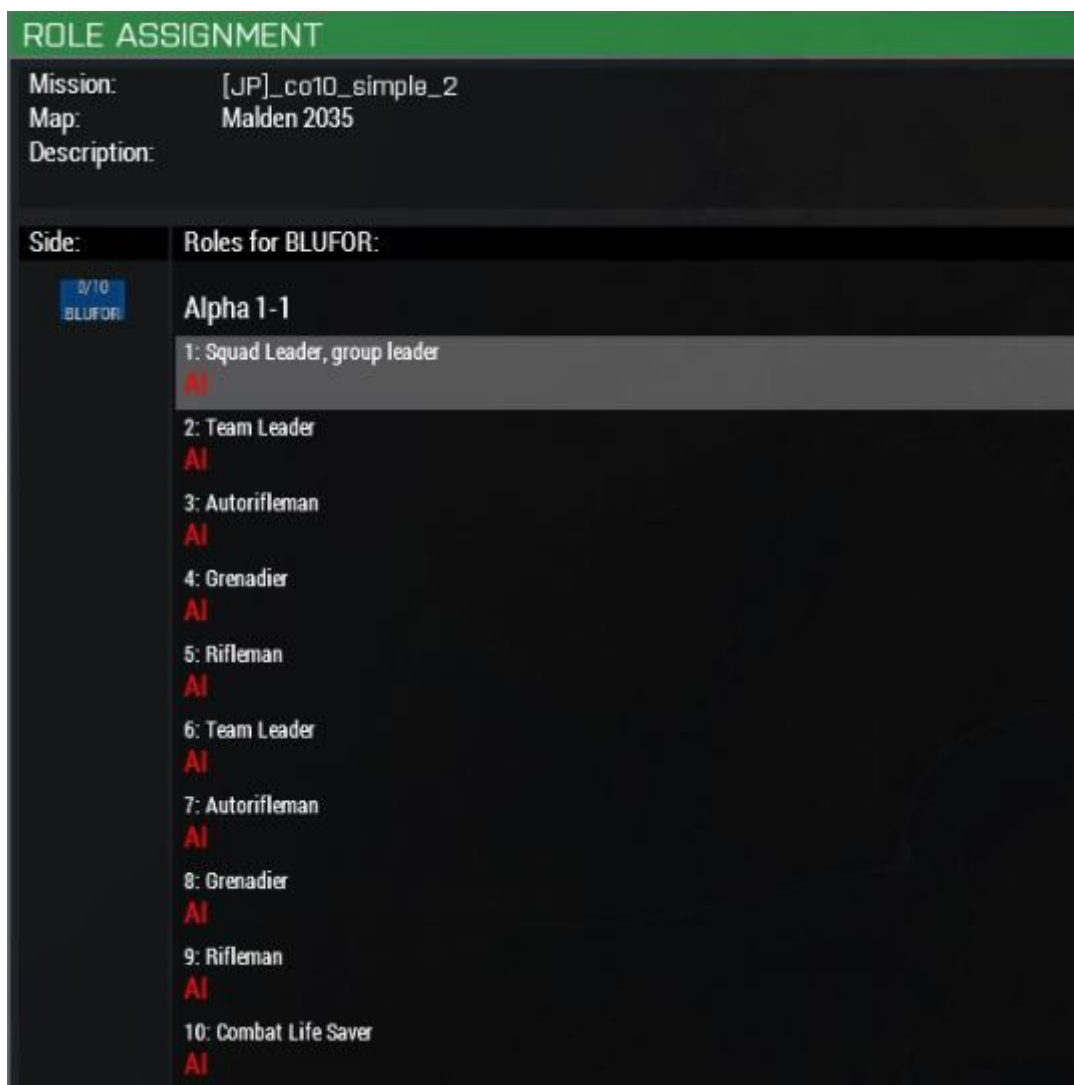
と書き込んでおくと、そのグループはその場から移動しなくなります。

3.13 微調整

ここまでくればほぼミッションとして形になっているはずです。

ところで海兵隊装備なのに編成が違うのが気になる人もいますよね。敵とのバランスを考えながら編成を変えてもいいかもしれません。

ここで注意したいのは、**ロールアサイン画面**での表示です。



Role Assignment画面

1～10までスロットがありますが、この表示順はミッションエディターで配置した順番になっています。なのでSLを最後に配置した場合はCombat Life Saverの下になってしまいます。配置順を間違ってしまった場合は先に配置したユニットを消すしかありません。

作戦時間も変更しましょう。

エディター上部のAttributes>Environmentから日時や天候の設定ができます。

NVがないのに作戦中に真っ暗になって困る事のないようにしてください。



Environment画面

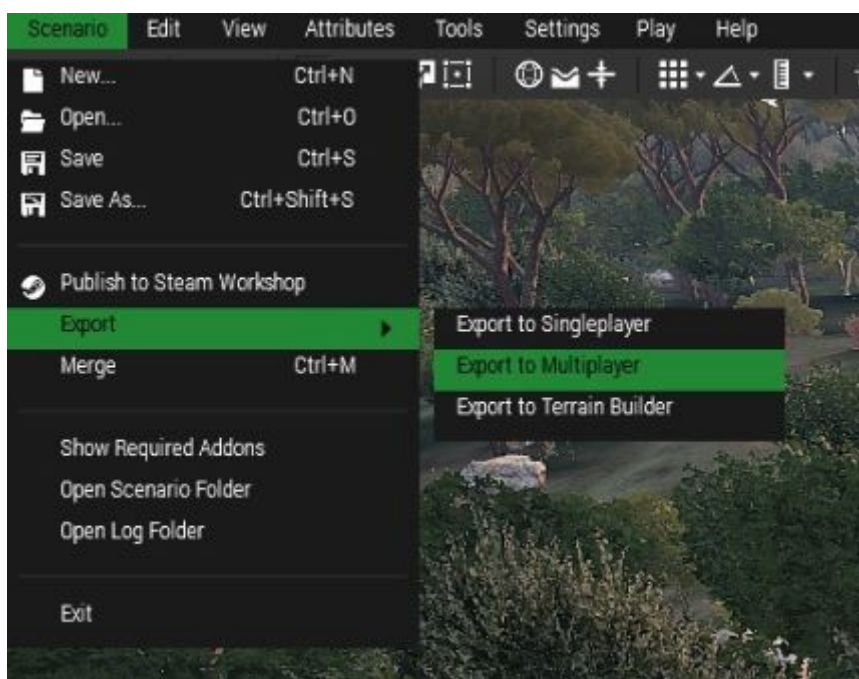
サブライボックスは必要なければ消してしまってもいいでしょう。普段のミッションだとLAWやAT4が入ってたりしますよね？しかし今回のミッションにランチャーがあると遠くからAAAを撃って終わってしまうかもしれません。

4 完成！

お疲れさまでした。

もう一度ミッションの名前が命名規則に沿っているか、タスクが達成できるか、各ユニットの持ち物やアーセナルの中身が問題ないか確認してみましょう。

TS3のミッションエディットルームにいき、誰かにミッションチェックしてもらい、**サーバーにアップロードしてもいいか必ず確認してください。**



Export

誰かにチェックしてもらう時、サーバーにアップロードする時はPBOファイルにする必要があります。ミッションエディター画面のScenario>Export>Export to MultiplayerでPBOファイルを作成できます。

PBOファイルの保存場所

steam\steamapps\common\arma3\MPMissions

Mission Edit Roomで見ってもらう時にアップロードする場所

TS3>Mission Edit Room>ファイルブラウザー>見てほしいミッションはこの中へ

ミッションをゼロから作るのは大変です。同じようなタスクのミッションを積極的に流用していきましょう。皆さんの作ったミッションが夜9時に回るのを楽しみにしています。

参考:プレイしたミッションの開き方

『このミッションどうやって作ってるんだろう?』というときはキャッシュにPBOファイルが保存されているので参考にしましょう。

プレイしたミッションの保存場所

C:\ユーザー\Name\AppData\Local\Arma 3\MPMissionsCache

著者:さぶろう

TS3:Sublow

ご意見ご質問は著者までどうぞ。

2018年3月22日

2018年4月2日改訂