Multiple Squad and Random Objectives Tutorial

This tutorial demonstrates how the Framework handles multiple squads (either player or AI controlled), objective randomization, and a dynamic interface for changing loadouts.

To complete this tutorial you will modify a mission script. The map elements needed for the tutorial are supplied.

Preparing to Create the Mission

Copy the MultiSquadObjectives.Altis subdirectory to:

> <MyDocuments>\Arma 3 - Other Profiles\<MyProfile>\missions\

Copy the Zen_FrameworkFunctions directory from the Shell.Stratis directory to:

> <MyDocuments>\Arma 3 - Other Profiles\<MyProfile>\missions\
> MultiSquadObjectives.Altis\

Review the Map

Open Arma and then the Editor. Select Altis Island and Continue.

Choose the load function and then select MultiSquadObjectives.

East of Stavros two playable squads are ‘parked’. The primary squad will be inserted by boat and the other squad inserted by helicopter.

The secondary squad’s landing zone and the ‘loadout’ ammo box are marked with icons.

No changes are required to the map.

Updating the Initialization Script

Open the *init.sqf* file in

> <MyDocuments>\Arma 3 - Other Profiles\<MyProfile>\missions\
> MultiSquadObjectives.Altis\

Immediately above the lines

> *// All clients stop executing here, do not delete this line*
> *if (!isServer) exitWith {};*

Add a mission briefing statement:

```
Player creatediaryRecord["Diary", ["Multi-Squad Co-op Tutorial", "Two
BLUFOR squads are tasked with the goal of eliminating a recently
arrived enemy unit. Local friendly units have assisted with the
stockpiling of various advanced weapons. After the enemy vehicle is
```

```
clearly identified, choose a loadout and eliminate the
vehicle.<br/>"]];
```

At the point labeled ‘Enter the mission code here’ enter these statements:

Add to the Arma III action list for all playable characters the ability to share ammo and re-pack magazines:

```
0 = [[(group X11),(group Y11)]] call Zen_AddGiveMagazine;
0 = [[(group X11),(group Y11)]] call Zen_AddRepackMagazines;
```

Set the time and the weather:

```
0 = [["date", 18, 8, 10, 3, 2035]] spawn Zen_SetWeather;
0 = [["overcast",0.6]] spawn Zen_SetWeather;
```

Insert squad ‘X’ by boat:

```
_Xsquad_boat_SpawnPos = [_PlayerPosition,
[100,200],[],2,[0,0],[120,160]] call Zen_FindGroundPosition;
_Xsquad_boat = [_Xsquad_boat_SpawnPos, "b_boat_transport_01_f"] call
Zen_SpawnBoat;
0 = [_Xsquad_boat, ["XSquadBeachhead", _Xsquad_boat_SpawnPos], (group
X11), "normal"] spawn Zen_OrderInsertion;
0 = [(group X11), _Xsquad_boat] call Zen_MoveInVehicle;
```

Insert squad ‘Y’ by helicopter:

```
_Ysquad_heli_SpawnPos = ["YSquadBeachhead",
[600,800],[],[2],[0,0],[90,120]] call Zen_FindGroundPosition;
_Ysquad_helicopter = [_Ysquad_heli_SpawnPos, "b_heli_light_01_f", 80]
call Zen_SpawnHelicopter;
0 = [_Ysquad_helicopter, ["YSquadBeachhead", _Ysquad_heli_SpawnPos],
(group Y11), "normal", 80] spawn Zen_OrderInsertion;
0 = [(group Y11), _Ysquad_helicopter] call Zen_MoveInVehicle;
```

Prepare the ammo box with possible load outs:

```
0 = [BLUFORLoadouts,["AA Specialist","AT Rifleman","AT
Specialist","Grenadier"],-1] call Zen_AddLoadoutDialog;
```

Give the squads time to land and get to the ammo box:

```
sleep 120;
```

Create an array with four OPFOR vehicles and choose one at random:

```
_vehicleArray =
["O_APC_Tracked_02_cannon_F","O_MBT_02_cannon_F","O_Heli_Attack_02_F",
"O_Heli_Light_02_F"];
_selectedVehicle = [_vehicleArray] call Zen_ArrayGetRandom;
```

Create a custom objective using the randomly choose vehicle:

```
_yourObjective = ["OPFORPatrolOne", [(group X11),(group Y11)], east,
"Custom", "Eliminate", _selectedVehicle] call Zen_CreateObjective;
```

Get the reference to the vehicle and populate it with a crew:

```
_objectiveVehicle = ((_yourObjective select 0) select 0);
0 = [_objectiveVehicle, east] call Zen_SpawnVehicleCrew;
```

Order the OPFOR vehicle to patrol an area:

```
if (_objectiveVehicle isKindOf "Air")
      then {[_objectiveVehicle, "OPFORPatrolOne"] spawn
      Zen_OrderAircraftPatrol;}
else
{[_objectiveVehicle, "OPFORPatrolOne"] spawn Zen_OrderVehiclePatrol;};
```

Test if either squad X or Y is NOT controlled by player. If controlled by Arma III engine then give it some simple instructions and then give two units in the squad a Framework loadout:

```
if (!isplayer X11) then{

      0 = [(group X11), _objectiveVehicle,[50,200],[0,360],"normal"]
      spawn Zen_OrderInfantryPatrol;

      _returnArray = [[(group X11)],"name",west] call
      Zen_TrackInfantry;

      if (_objectiveVehicle isKindOf "Air")
      then {[[X11,X12], "AA Specialist"] call Zen_GiveLoadoutBlufor;}
      else {[[X11,X12], "AT Specialist"] call Zen_GiveLoadoutBlufor;};
};

if (!isplayer Y11) then{

      0 = [(group Y11), _objectiveVehicle,[50,200],[0,360],"normal"]
      spawn Zen_OrderInfantryPatrol;

      _returnArray = [[(group Y11)],"name",west] call
      Zen_TrackInfantry;

      if (_objectiveVehicle isKindOf "Air")
      then {[[Y11,Y12], "AA Specialist"] call Zen_GiveLoadoutBlufor;}
      else {[[Y11,Y12], "AT Specialist"] call Zen_GiveLoadoutBlufor;};
};
```

Loop until objective is complete and then exit the mission:

```
waituntil { sleep 5; [(_yourObjective select 1)] call
Zen_AreTasksComplete };

endMission "end1"
```

Play the Mission.

To play this mission from inside the editor select ‘Preview’.

If you want to play this as co-op then save/export as multi-player mission and then play as multi-player mission.

Post-Mortem

If you played the mission here’s what you should have seen:

- A briefing
- One squad inserted by boat, one by helicopter
- A single objective created
- An ammo box with an action menu option to change load out
- After destroying the OPFOR vehicle the mission ended.

Technical Corner

This tutorial is the first ‘pure’ co-op mission. But it shows how to put code in your mission script so that you missions you’ve worked so hard to create can also be played solo.

Boat Insertions

AI controlled units will tread water until give patrol instructions.

Make this mission more realistic by adding this line just before the two minute wait:

```
if (!isplayer X11) then{
      0 = [(group X11),BLUFORLoadouts,[5,20],[0,360],"normal"] spawn
      Zen_OrderInfantryPatrol;};

if (!isplayer Y11) then{
      0 = [(group X11),BLUFORLoadouts,[5,20],[0,360],"normal"] spawn
      Zen_OrderInfantryPatrol;};
```

Loadout Options

The array of loadout names are the same as are used in the loadout assignment functions (see *Assassination Mission* tutorial).

Change the loadout of a member of your squad by selecting the unit and then the action menu (‘6’). The loadout option should be in the action list.

Random Selection

The Framework Pick Random function selects one element randomly from an array; even arrays that have a mix of data types (e.g., strings, integers or references to icon or area markers).

Framework Return Values

This script contains the statement:

```
_objectiveVehicle = ((_yourObjective select 0) select 0);
```

The variable *_your Objective* is an array with two elements. *‘_yourObjective select 0’* means to select the first element in the array. This element is itself an array. So select the first element of this array with the command ‘*select 0’*.

Mixing Player and AI Controlled Groups

The Framework functionality is focused on the easy creation of co-op missions. But of course, it’s not always possible to get enough players to lead every group in a player designed mission.

The Framework supports the ability to create and complete objectives regardless of how many groups and units are ‘filled’ by human players.

By using functions such as *Zen_OrderInfantryPatrol* and *Zen_GiveLoadoutXXXXXX* it is possible to cause AI units to behave realistically.

So go ahead and design those four squad missions and don’t worry about how many players join (or don’t). With a few commands the AI controlled groups can still contribute to completing the missions.