# Spawn dinamico randomizzato veicoli

## Configurazione

### btc\_rhs\_global\_variables.sqf

All’interno vengono definite tutte le variabili globali utili al CORE per ricavare gli array di classi partendo da dati in input.

Le variabili contenute tra i commenti:

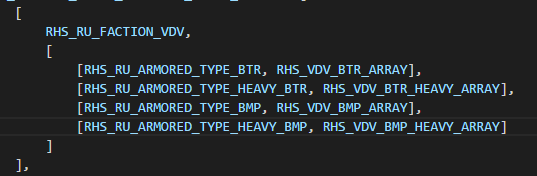
* “/\*\*\*\*\* DEFINING VEHICLES TYPES \*\*\*\*/” e “/\*\*\*\*\* END DEFINING VEHICLES TYPES \*\*\*\*/” : vengono usate sia per la configurazione delle associazioni (che vediamo in seguito) sia come input per le funzioni;
* “\*\*\*\*\* DEFINING FACTIONS PER SIDE \*\*\*\*\*/” e “/\*\*\*\*\* END DEFINING FACTIONS PER SIDE \*\*\*\*\*/”: vengono usate sia per la configurazione delle associazioni (che vediamo in seguito) sia come input per le funzioni;
* “/\*\*\*\*\* DEFINING ARRAY OF CLASSES NAMES FOR VEHICLES \*\*\*\*\*/” e “/\*\*\*\*\* END DEFINING ARRAY OF CLASSES NAMES FOR VEHICLES \*\*\*\*\*/”: vengono usate per reperire le classi dei veicoli dalle associazioni (che vediamo in seguito).
* “/\*\*\*\*\* DEFINING ARRAYS OF VEHICLES TYPES AND FACTIONS PER SIDE \*\*\*\*\*/” e “/\*\*\*\*\* END DEFINING ARRAYS OF VEHICLES TYPES AND FACTIONS PER SIDE \*\*\*\*\*/”: vengono usate per le associazioni o per ciclare tra i tipi e le fazioni.

### btc\_rhs\_relationships.sqf

All’interno troviamo le associazioni tra le varie variabili globali precedentemente configurate



Questa variabile globale è l’associazione tra fazioni, tipi di veicoli e array di classes names dei veicoli si presenta in forma di array di array.



Questa è la configurazione standard di una fazione. Si presenta come array e si spiega in questo modo:

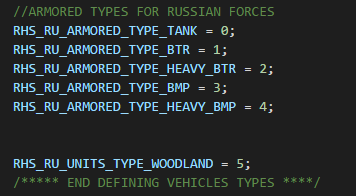
La fazione “RHS\_RU\_FACTION\_VDV” contiene 4 tipologie di veicoli (RHS\_RU\_ARMORED\_TYPE\_BTR, RHS\_RU\_ARMORED\_TYPE\_HEAVY\_BTR, RHS\_RU\_ARMORED\_TYPE\_BMP, RHS\_RU\_ARMORED\_TYPE\_HEAVY\_BMP) ai quali sono associati determinati array di classi (RHS\_VDV\_BTR\_ARRAY, RHS\_VDV\_BTR\_HEAVY\_ARRAY, RHS\_VDV\_BTR\_HEAVY\_ARRAY, RHS\_VDV\_BMP\_HEAVY\_ARRAY).

Nel caso in cui si voglia aggiungere una tipoligia di veicolo per la suddetta fazione basta definire nel file “btc\_rhs\_global\_variables.sqf” le variabili necessarie e creare la relazione tra le stesse nel file “btc\_rhs\_relationships.sqf”

Example.

Supponiamo di voler aggiungere lo spawn di unita woodland per la fazione VDV procederemo in questo modo:

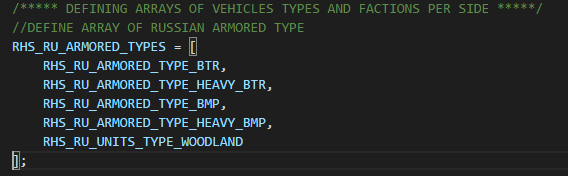
aggiunta tipo veicolo in “btc\_rhs\_global\_variables.sqf”:



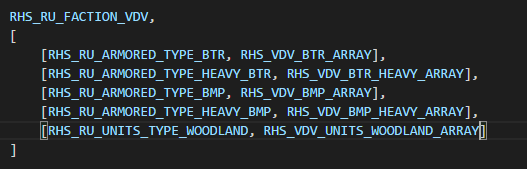
Aggiunta array di classi in “btc\_rhs\_global\_variables.sqf”:



Aggiunta di tipo nell’array delle tipologie di veicoi in “btc\_rhs\_global\_variables.sqf”:



Aggiunta della relazione in “btc\_rhs\_relationships.sqf”:



In questo modo non occorre cambiare alcunchè nel codice sorgente e richiamando la funzione per lo spawn mettendo in input “RHS\_RU\_FACTION\_VDV”(fazione), “RHS\_RU\_UNITS\_TYPE\_WOODLAND” verranno create le unità randomicamente.

## Funzioni principali

### BTC\_fnc\_CreateRandomArmored

Prende in ingresso quattro variabilii:

* Quantità dei veicoli (intero);
* Fazione del veicolo (variabile globale a scelta tra quelle definite in questo documento)
* Tipo del corazzato (variabile globale a scelta tra quelle definite in questo documento)
* Nomi dei marker (array di stringhe contenente il nome dei marker in mappa)
* Raggio di spawn dei veicoli (intero)

Dati gli input sopra definiti seleziona l’array corrispondente per la fazione e il tipo di veicolo in ingresso (associazione definita nella configurazione);

Per ogni veicolo da creare esegue:

* Chiamata per recuperare l’array di classi dei veicoli corrispondente alla fazione ed il tipo di veicolo in ingresso (associazione definita nell’apposito file di configurazione);
* Recupera randomicamente il nome del marker tra quelli forniti in ingresso;
* Recupera randomicamente la classe del veicolo dall’array precedentemente recuperato;
* Recupera la posizione del marker selezionato;
* Recupera una posizione random in un raggio, definito in input, partendo dal punto del marker selezionato;
* Crea il veicolo nella posizione recuperata;
* Crea la crew del veicolo;
* Aggiunge il veicolo creato ad un array;
* Ritorna l’array di veicoli creati;

## BTC\_fnc\_CreateRandomEmptyArmored

Prende in ingresso quattro variabilii:

* Quantità dei veicoli (intero);
* Fazione del veicolo (variabile globale a scelta tra quelle definite in questo documento)
* Tipo del corazzato (variabile globale a scelta tra quelle definite in questo documento)
* Nomi dei marker (array di stringhe contenente il nome dei marker in mappa)
* Raggio di spawn dei veicoli (intero)

Dati gli input sopra definiti seleziona l’array corrispondente per la fazione e il tipo di veicolo in ingresso (associazione definita nella configurazione);

Per ogni veicolo da creare esegue:

* Chiamata per recuperare l’array di classi dei veicoli corrispondente alla fazione ed il tipo di veicolo in ingresso (associazione definita nell’apposito file di configurazione);
* Recupera randomicamente il nome del marker tra quelli forniti in ingresso;
* Recupera randomicamente la classe del veicolo dall’array precedentemente recuperato;
* Recupera la posizione del marker selezionato;
* Recupera una posizione random in un raggio, definito in input, partendo dal punto del marker selezionato;
* Crea il veicolo nella posizione recuperata;
* Aggiunge il veicolo creato ad un array;
* Ritorna l’array di veicoli creati;