SCRIPTING

Repositorio de Scripts

- Comandos de ejecución
- Script "tormenta" particulas (Arena, nieve, etc...)

Comandos de ejecución

call

- Ejecuta instrucciones en el stack actual
- Espera a que el código termine
- Es lo mismo que escribir el código en el mismo sitio pero permite separar el código en distintos ficheros

execVM

- Compila y añade un script .sqf a la cola del scheduler
- El script no se ejecuta inmediatamente

spawn

- Añade instrucciones a la cola del scheduler
- Las instrucciones no se ejecutan inmediatamente

https://community.bistudio.com/wiki/Code_Optimisation

Script "tormenta" particulas (Arena, nieve, etc...)

Script para simular partículas. Se puede generar tormentas de arena, nieve, etc... No afecta al rendimiento del server .

dustStorm.sqf

```
// Variable global para controlar la tormenta (Inicia ACTIVADA al entrar)
if (isNil "Mission_Sandstorm_Active") then {
    Mission_Sandstorm_Active = true;
};

// --- SERVIDOR: Control del clima global ---
if (isServer) then {
    [] spawn {
        while {true} do {
            waitUntil { Mission_Sandstorm_Active };

            // Transición a clima de tormenta (Inmediato/Rápido al inicio)
            0 setOvercast 0.25;
            0 setRain 0;
            0 setFog [0.10, 0.01, 10];
            0 setGusts 0.8;
            setWind [15, 15, true];
            forceWeatherChange;
            waitUntil { !Mission_Sandstorm_Active };

            // Restaurar clima despejado y DETENER EL VIENTO POR COMPLETO
            0 setOvercast 0;
            0 setFog 0;
            0 setGusts 0;
            setWind [0, 0, true]; // Fuerza viento en 0
            forceWeatherChange;
        };
    };
};
if (!hasInterface) exitWith {};
// --- CLIENTE: Control de efectos visuales y de sonido ---
[] spawn {
    scriptName "Dynamic_Sandstorm_Handler";
    while {true} do {
        // Esperar a que la tormenta esté activa Y el jugador haya cargado completamente
        waitUntil { Mission_Sandstorm_Active && {!isNull player} && {alive player} };
        // 1. Crear Corrección de Color
        private _ppColor = PPeffectCreate ["ColorCorrections", 1500];
        _ppColor ppeffectEnable true;
        _ppColor ppeffectAdjust [
            1.20, 1.22, -0.03,
            [0.85, 0.60, 0.35, 0.025],
            [0.85, 0.55, 0.28, 0.40],
            [0.2, 0.12, 0.08, 0.0]
        ];
        _ppColor ppeffectCommit 1; // Transición rápida de 1s al entrar
        // 2. Crear Emisor de Partículas
        private _emitter = createVehicleLocal ["#particlesource", getPosATL player, [], 0, "NONE"];
        _emitter setParticleParams [
            ["\A3\data_f\ParticleEffects\Universal\Universal", 16, 12, 8, 0], "", "Billboard",
            1, 8, [0, 0, 0], [10, 10, 0], 1, 1.275, 1, 0,
            [5, 15, 25],
            [[0.6, 0.45, 0.25, 0.05], [0.5, 0.4, 0.2, 0.12], [0.6, 0.45, 0.25, 0]],
            [1000], 1, 0, "", "", player
        ];
        _emitter setParticleRandom [3, [20, 20, 5], [5, 5, 2], 0, 0.25, [0, 0, 0, 0.05], 0, 0];
        _emitter setParticleCircle [0, [0, 0, 0]];
        _emitter setDropInterval 0.025;
        _emitter attachTo [player, [0, 0, 0]];
// 3. Bucle de Audio (Optimizado con minutos y control de fallos)
        private _audioScript = [player] spawn {
            params ["_targetPlayer"];
```

******ENTRADA GENERADA CON IA********

Script de Tormenta de Arena Dinámica

Este script añade una tormenta de arena inmersiva a tu misión, gestionando de forma independiente la carga del servidor (clima) y la de los clientes (efectos visuales y de sonido).

? ¿Cómo ponerlo en marcha? (Instalación rápida)

El código está optimizado para funcionar en un **único archivo**, ya que incluye comprobaciones internas (`isServer` y `hasInterface`) para saber qué ejecutar en cada máquina.

1. **Crea el archivo del script:**
   - En la carpeta scripts de la raíz de tu misión, crea un archivo de texto y nómbralo `tormenta.sqf`.
   - Pega todo el código proporcionado dentro de este archivo.
2. **Ejecuta el script al inicio de la misión:**
   - Abre o crea el archivo `init.sqf` en la carpeta de tu misión.
   - Añade la siguiente línea para llamar al script al arrancar:
   - Fragmento de código

```
[] execVM "scripts\tormenta.sqf";
```

3. **Requisito de Audio (Importante):**
   - El script intenta reproducir un sonido llamado `"Sandstorm_Wind"`. Necesitarás tener este archivo de audio (normalmente un `.ogg` de unos 18 segundos) definido en tu archivo `description.ext` bajo la clase `CfgSounds`.
   -

```
class CfgSounds
{
    sounds[] = {};
    class Sandstorm_Wind
    {
```

```
        name = "Sandstorm_Wind";
        sound[] = {"sound\sandstorm.ogg", 10, 1.0}; // Volumen base y tono
        titles[] = {};
    };
};
```

?? Explicación del Código: ¿Qué hace cada parte?

El script se divide en tres bloques principales que se ejecutan simultáneamente en un mismo archivo:

- **Control de la Variable Global:**
  - Define la variable `Mission_Sandstorm_Active`.
  - Por defecto, el script inicia con la tormenta **ACTIVADA** (`true`) al entrar a la misión.
- **Control del Servidor (`isServer`):**
  - Se encarga de modificar el clima global para todos los jugadores.
  - Cuando se activa, ajusta de forma rápida la nubosidad (0.25), elimina la lluvia, añade niebla espesa y aumenta las ráfagas y la fuerza del viento.
  - Cuando se desactiva, restaura el clima despejado y detiene el viento por completo.
- **Control del Cliente (`hasInterface`):**
  - Solo se ejecuta si el jugador tiene interfaz y ha cargado completamente (está vivo).
  - **Efectos Visuales:** Crea un filtro de corrección de color con tonos anaranjados/marrones y genera un emisor de partículas (`#particlesource`) atado al jugador para simular la arena volando.
  - **Efectos de Sonido:** Ejecuta un bucle de audio que varía el volumen y el tono del sonido `"Sandstorm_Wind"`. Incluye un sistema de pausas dinámicas que alterna entre pausas normales (1.5 a 3.5 minutos) y momentos de "calma chicha" (3.5 a 6 minutos) para no saturar al jugador.
  - **Limpieza:** Si el jugador muere o la tormenta se desactiva, el script elimina las partículas, detiene el bucle de sonido y desvanece el filtro de color suavemente durante 2 segundos.

? ¿Cómo encender y apagar la tormenta en medio de la partida?

Como la tormenta depende de una única variable global, puedes controlarla fácilmente desde un **Activador (Trigger)** en el editor, un script de Zeus, o al completar una tarea.

Si sigues los pasos tal y como están, el script arrancará al arrancar la partida.

- **Para apagar la tormenta:**
  Fragmento de código

```
Mission_Sandstorm_Active = false; publicVariable "Mission_Sandstorm_Active";
```

- **Para volver a encenderla:**
  Fragmento de código

```
Mission_Sandstorm_Active = true; publicVariable "Mission_Sandstorm_Active";
```

NOTA: *El script esta hecho con los pocos conocimiento que tengo de programación, mucha IA y prueba y error. Posiblemente haya soluciones mejores. Lo que se ha comprobado es que funciona bien y no afecta aparentemente al servidor, que responde prácticamente igual con 20 jugadores, con el scripts y sin el.*