Zeusday Mission Build Process

Step 1: Preparation (curator)

- Download zeusday template: Latest Zeusday Template
- Extract the ZIP to your MP Missions Folder, this is generally located in one of the following locations:
  - `C:/Users/(LOGIN NAME)/Documents/Arma 3 - Other Profiles/(PROFILE NAME)/mpmissions`
  - `C:/Users/(LOGIN NAME)/Documents/Arma 3/mpmissions`
  - If the profile folder exists, but the mpmissions is missing, just create it
- You should now have a folder called ZeusTemplate.Altis in your mpmissions folder

Step 2: Mission Building (curator)

- Start up Arma, ensuring you have either no mods, or at most, any of the following whitelisted mods enabled:
  - ZEC - Zeus and Eden Templates / Building Compositions
  - 3den Enhanced
- Mods such as Achilles / CBA / Blastcore alter the required mods lists for clients, and in some cases, such as CBA make changes to all the init fields on save of units/groups which can be an utter pain to remove
- Build your mission
  - See the "Editing FAQ" below regarding altering of assets, or adding assets

Step 3: Prep Release (curator)

Ensure Mission Parameters in Eden

- From the Attributes Menu, select General
  - Ensure "Title" is set appropriately, this is shown on the map and mission selection menu

Edit: template.ini

This file contains information that controls the packaging of your mission within the templating framework

It is essential you update this accordingly with a minimum of author and mission name

Edit: settings.sqf

This file is called in the "preInit" stage, and controls various elements of the YAINA framework that are useful for zeusdays.

This includes disabling of real_weather, vehicle abandonment among other things. Whilst we feel we have sane defaults, feel free to alter for it to be suitable for your mission.

Step 4: Request Mission (curator)

- Right click on our zeus-template.Altis folder and select "Add to Zip"
- Send that ZIP to any member of the Server Team for them to deploy it

Step 5: Test Mission (admin)

- Deploy the zip to the test server
- Drag the zip onto: `EU1/Batch Files/Build Zeusday PBO.bat`
- Ensure successful output and that the ZeusDayName is in the server¥mpmissions folder
- Connect to test server
- Issue !mission ZeusDayName
- Ensure mission loads as intended

Step 6: Deploy Mission (admin)

- Deploy the zip to the live server
- Drag the zip to: EU1/Utils/build-zeusday-pbo.bat
- Ensure successful output and that the ZeusDayName is in the server¥mpmissions folder

Step 7: Start ZeusDay (admin lv.3)

- Once step 6 has been completed, an admin level 3 can start the Zeusday by issuing !mission

Editing FAQ

3den Enhanced

Handy helpers for 3den enhanced scenario editing:

Units / Groups:

- Right click on a placed group or unit, then click attributes:
- Object: States - Stance: Force AI into a specific stance to prevent them going prone at first sign of contact and become inivsible in the high grass.
- Object: Disable AI Features - Path: Locks the selected unit in place, making them basically a garrisoned unit, but you get more freedom about placement.

I suggest you use this method to garrison your objectives. If you place down about 5-6 different unit types, set them all to stand up, and disable pathing, then you can simply copy and paste them around the buildings.

Waypoints:

- Click on a group - Shift+Right Click: Creates a waypoint.
- Right click on a placed Waypoint, then click attributes: The last waypoint in a series, change it to a "Cycle" waypoint (so it loops the patrol pattern). All other waypoints keep as either "Move", or if you want AI to be more agressive "Seek and Destroy".

Objects / Terrain:

- Right click on a placed object, then click attributes:
- Object: Special States - Enable Damage: Disable damage on walls and gates you don't want players to ram / destroy with vehicles.
- Right dropdown menu - Modules - Environment - Edit Terrain Object: Lets you edit buildings, lock and open doors, set damage on it and other things.
- Right dropdown menu - Modules - Environment - Hide Terrain Objects: Lets you clear a space of any obstructions such as buildings, rocks, trees and so on. Helpful for building a custom structure inside a city.

Various other tips

- Top Toolbar - Toggle vertical mode and surface snapping: Helpful if you want to sink a certain object into the ground, without snapping to the terrain. Experiment with which settings fit for what.

Editing images

- All textures are in the "data" folder
- feel free to create PAA files and replace them to your own needs
- In order to maintain file size, if you remove references to a image, please delete it from the data folder

Delete a vehicle at base

- Go for it...however, I would recommend against the removing the following vehicles:
  - MERT chopper
  - Repair Bobcat

Add a new vehicle

- Create the vehicle you wish to provide
- If respawn is enabled, it will respawn in this position once clear
- Set the weapon loadout / inventory loadout of the vehicle
- Edit the "init" field as follows
  - Limit drivers to certain traits ? (air assets, by default limit to pilots if not explicitly set)
    - Set a global variable on the asset (PRIOR to vehicle init)

      ```
      this setVariable ["YAINA_VEH_Drivers", ["HQ", "MERT", "PILOT",
      "UAV"], true];
      ```

      - This overrides the defaults, so setting this on a Heli with just MERT will mean that pilots cannot get in
  - Initialize the watcher
    - For a choppper

```
[this, false, 10, 1000] call YAINA_VEH_fnc_initHeli;
```

      - This means:
        - Setup the chopper
          - There isn't really anything different between a chopper and a jet or ground vehicle from the perspective of the handler, however, by default, our choppers have the "Set Explosives" option for pilots, so this function could be used on a blackfish without concern
        - does not have keys
        - respawns in 10 seconds
          - A value of -1 means do not respawn, it is useful for destructable assets whilst maintaining keys, driver restrictions etc.
        - is considered abandoned if no players within 1000m (2000 with abandonment)
          - This setting is ignored if abandonment is globally disabled via settings.sqf
          - A value of 0 means no abandonment (e.g UAVs)
    - For UAVs

```
[this, false, 10, 0] call YAINA_VEH_fnc_initVehicle;
```

      - UAVs must have an abandonment distance of 0 otherwise they'll be despawning whilst folks fly
      - the arguments are as per the chopper above
    - For any other vehicle

```
[this, true, 10, 3000] call YAINA_VEH_fnc_initVehicle;
```

      - the arguments are as per the chopper above
  - Full Example:
    - A Medical HEMMT, restricted to mert, with keys, and no respawn, and abandoned after 500m would be:

```
this setVariable ["YAINA_VEH_Drivers", ["MERT"], true];
[this, true, -1, 500] call YAINA_VEH_fnc_initVehicle;
```

  - Even more advanced options
    - See Functions/YAINA/Vehicle/fn_initVehicle.sqf
    - Examples can be found Functions/YAINA/init/fn_serverAltis.sqf

        - With this you have full control over every element of the vehicle setup
            - Want fully custom loadouts that aren't available in 3den ? use the init code
            - Want to have a MERT taru loadout? that too is done using the init code
        - Custom driver verification checks (e.g: the HEMMTT Ammo Truck that only allows group leaders to drive) by defining getInHandler

Can I bring in additional functions and scripts ?

This is a much more advanced topic and not something the every day curator will need access to, but for some days, we might want to add additional functionality above and beyond the base YAINA configuration, and this is how we go about it.

In order to test all these, it would be far more sensible to download the YAINA code base and run it (from within 3den is fine) before separating out your functions into just the template.

Functions:

- Edit Functions¥CfgFunctions.hpp, you may use includes if you wish to logically separate your code base
- It is possible to override exitsing YAINA functions, but would recommend you don't

Scripts:

- Anything in the "Scripts" folder will be force-merged into the client deliverable, if files already exist in the base mission they will be overwritten.
- These will need a helper function to execute them, see above

Description.ext:

- Some additions, such as UI elements require includes in description.ext, the contents of description.append.ext is **appended** to description.ext
    - Do not use for CfgFunctions blocks, use the existing functionality outlined above
- You are responsible for resolving any UI Element IDC/IDX conflicts, so maybe best to test these as part of an entire mission and then bring into the