



## II- Le INI, le texte comme maître du jeu

### 1/ Introduction au INI

Que vous l'aimiez ou non, le INI sera indispensable à la création d'un mod. Que vous vouliez ajouter un pouvoir à Aragorn, ou simplement changer son modèle, vous devrez impérativement passer par le INI, donc autant s'y mettre tout de suite.

Mais tout d'abord, pour ceux que ça intéresse, autant savoir ce qu'est réellement un fichier .ini. Il s'agit d'un fichier de configuration, utilisé afin de déterminer simplement des configuration, sans avoir à les intégrer dans le code source, ce qui rendrait peu aisé leur modification rapide. De plus, les fichiers INI permettent à l'utilisateur averti de les modifier à sa guise.

Sachez tout d'abord que dans un fichier INI, les textes suivant un point virgule ( ; ) ne seront pas pris en compte dans la lecture du fichier par le moteur de jeu. Il est donc facile d'insérer des commentaires, pour se souvenir de ce qu'on a voulu faire en mettant tel ou tel pièce de code. Il suffit de passer à la ligne pour que le moteur de jeu lise les commentaires suivants :

```
;un fichier INI basique  
<votre code>  
;fin du fichier INI
```

Aussi, prenez l'habitude de ne plus parler de héros ou de bâtiments, mais d'objets.

Les fichiers INI sont utilisés aussi par Windows, mais je vais dans ce tutorial seulement détailler l'usage des fichiers INI pour BFME.

### 2/ architecture du fichier INI.big

Après avoir extrait votre fichier INI.big à l'aide de FinalBig, déplacez le dossier ou vous avez stocké les fichiers extraits, afin qu'il soit dans

```
<votre disque dur>\Program Files\EA GAMES\La Bataille pour la Terre du  
Milieu(tm)\data
```

Il s'agit du répertoire par défaut, ajustez par vous même si vous avez installé votre jeu ailleurs.

Maintenant, contemplons les multiples fichiers INI devant vous, un beau bazar

...

Je vais ici vous détailler rapidement le rôle de chacun des fichiers (j'ai **surligné** les plus importants, et *mis en italique* les moins importants)

a) La racine

- *AIData* : aucun ! Les valeurs pour l'IA ont été transférés vers Default\AIData\ini
- *AmbientStream* : les sons d'ambiance sont codés ici
- *Animation2D* : c'est ici que les textures sont animés. Je n'ai personnellement jamais modifié ce fichier, et je ne pense pas que vous aurez à le modifier non plus
- **Armor** : enfin un fichier intéressant ! C'est ici que sont stockés tous les types d'armures, utilisés dans les codes d'unités, et les réactions de ces armures face aux différents types de dégâts.
- **AttributeModifier** : principalement utilisé par les sorts et pouvoirs, ce fichier concentre toutes les caractéristiques des modifications d'états pour les unités. Si vous voulez qu'un sort modifie les caractéristiques d'un objet pendant un certain temps (ou indéfiniment même), alors vous intéressez vous à ce fichier
- *AudioSettings* : c'est ici que les paramètres audios sont stockés. Vous n'aurez sans doute pas à utiliser ce fichier
- *Bannerui* : je n'ai jamais joué avec ce fichier, mais je crois qu'il commande les jolies bannières de renfort utilisées en campagne
- *Campaign* : fichiers pour la campagne, ce fichier me semble inutilisé, vu la nature des valeurs se rapprochant fortement d'un certain Command and Conquer : Generals ...
- **CommandButton** : autre fichier extrêmement important, il contient absolument tous les boutons utilisés dans le jeu, du petit bouton Avancer à l'invocation de l'Armée des Morts. C'est donc ici que vous coderez tous vos nouveaux boutons
- *CommandMap* : fichier vide (il était utilisé avant que CommandMapDebug soit mis en place)
- *CommandMapDebug* : utilisé lors de la version bêta de BFME, afin de permettre de tester assez rapidement les choses voulues. On pourra remarquer des valeurs telles « DebugCommandMap DEMO\_ADD\_CASH » qui sont assez suggestives sur leurs rôles. Elles ont toutefois été retirées du code source final du jeu, donc sont inutilisables.
- **CommandSet** : Le commandset, c'est l'ensemble des boutons qu'une unité a. On regroupe donc la désignation de tous les boutons sous un seul code, le commandset, qu'on ajoutera au code de l'objet. Voici un exemple de CommandSet : PHOTO

- *Controlbarresizer* : aucune idée ... je dirais que ça à voir avec l'interface du jeu
- *Controlbarscheme* : utilisé pour coder l'interface du palantir.
- *Crate* : utilisé pour les objets qui apparaissent lors de la destruction d'une cave (les caisses d'or)
- *Credits* : c'est ici qu'est codé le générique et les remerciements, vous pouvez vous y ajoutez en suivant la procédure indiquée au début, dans les commentaires
- *DamageFX* : c'est là que sont présents tous les effets spéciaux liés au différents dégâts (une épée ne fait pas le même son qu'une explosion). Vous n'aurez probablement pas à modifier ce fichier
- *DrawGroupInfo* : aucune idée
- *Emotions* : le titre détaille plutôt bien le rôle de ce fichier; vous n'aurez probablement pas besoin de l'éditer
- *Environment* : utilisé pour coder le ciel et les divers éléments de l'environnement.
- *EVA* : rien à voir avec le prénom féminin, le système EVA est système d'annonce générale. Si Aragorn meurt, ou si vous perdez une base, une voix vous l'annoncera en jeu, et c'est codé ici. Si vous voulez ajoutez des EVA pour votre nouveau héros, pensez bien qu'il vous faudra ajouter des sons correspondants
- **ExperienceLevels** : on ajoutera ici tous les niveaux d'XP des nouveaux objets, ainsi que les détails de ces différents niveaux d'XP (expérience requise, bonus, etc ... )
- *FontSettings* : les polices d'écritures utilisés par les textes en jeu.
- *FontSubstitution* : une extension pour FontSettings.ini
- *FXList* : un FX (effet spécial) désigne un ensemble d'effets aussi variés que le son, la luminosité, les couleurs, les EVA utilisés, l'effet de mouvement de caméra etc ... Pour simplifier le coding des FX, ils sont rassemblés sous un seul nom, auquel on pourra se référencer. Ce nom est codé dans FXList.ini . Il s'agit d'un domaine assez complexe, ne vous attendez pas à réaliser des FX incroyables en 5 minutes.
- *FXParticleSystem(custom)* : le code pour les effets lumineux des FX, admirez la complexité ...
- **Gamedata** : c'est dans ce fichier que sont stockés toutes les données concernant chaque objet, chaque paramètre. On y rencontre des valeurs aussi variées que la vie, ou les dégâts que la limite de population.
- *Gamedatadebug* : on avait du boire un petit coup pour fêter la fin du

développement de BFME, et le fichier qui avait servi pour la bêta s'est insinueusement glissé dans le jeu final. Il est inutilisable

- *GameLOD(presets)* : les paramètres pour le niveau de détails choisi par l'utilisateur. Vous n'aurez sans doute aucun besoin de modifier ce fichier (aller modifier votre PC si vous n'êtes pas satisfaits)
- *Housecolor* : diverses textures d'objets, utilisés pour changer la housecolor (si votre couleur en escarmouche est le vert, l'unité aura des teintes vertes pour la distinguer)
- *InGameUI* : UI signifie User Interface. Il s'agit du code pour la barre de commande.
- *LargeGroupAudio* : code pour gérer les sons et les voix dans le cas de grands groupes, afin de recréer l'effet de (dés)ordre
- *LightPoints* : aucune idée
- *LivingWorld* : code général pour la carte de La Terre du Milieu en Campagne
- *LivingWorldLogic* : fichier vide ...
- *LivingWorldRegions* : codes pour les différentes régions de la carte (les films diffusés, les modèles utilisés ...)
- **Locomotor** : définit les actions que l'objet peut faire, de sa vitesse, à son milieu de prédilection (terre, air ... ) en passant par la vitesse quand il tourne.
- *MemoryPools* : aucune idée
- *MiscAudio* : divers codes de sons, ainsi que quelques paramètres audios.
- *Mouse* : les différents curseurs de la souris, changeant selon la situation
- *Multiplayer* : quelques paramètres pour le jeu en multijoueur, rien d'intéressant pour nous.
- *Music* : le nom détaille bien le rôle du fichier. Vous l'utiliserez si vous souhaitez ajoutez des musiques en jeu.
- **ObjectCreationList** : Liste de création d'objets. Voilà j'ai (quasiment) tout dit sur ce fichier où vous ajouterez vos codes pour les invocations d'objets.
- *OptionRegistry* : je ne voit aucun intérêt à ce fichier, car ses valeurs sont reprises par Gamedata et GameLOD si je ne m'abuse
- *ParticleSystem* : voir FXParticleSystem

- *Pathfinder* : un fichier entier pour une valeur, qui ne sera pas intéressante de modifier, un joli gâchis pour la clarté
- *PlayerAITypes* : définit les fichiers IA à utiliser, pour pouvoir les utiliser par la suite dans *PlayerTemplate*. Ajouter une IA pour une nouvelle faction est TRES fastidieux, alors n'espérez pas modifier ce fichier avant un petit moment
- **PlayerTemplate** : code principal pour les factions, il rassemble les diverses données codées ici et là, tel l'IA.
- *Rank* : je crois que ça a à voir avec les niveaux obtenus en multijoueur, mais les commentaires en début de fichier me font douter ...
- *Roads* : code pour les chemins, les ponts etc ...
- **Science** : fichier qui rassemble une partie des codes pour l'arbre des sorts.
- *ShellMenuScheme* : ce fichier code pour l'animation du menu principal, mais ne semble pas complet
- *SoundEffects* : tous les effets sonores, du fracas d'une épée contre un bouclier à l'explosion d'une mine. Facile à modifier, moins facile de trouver des jolis sons utiles
- **SpecialPower** : contient tous les ENUMS des différents pouvoirs spéciaux. Un ENUM est la définition du pouvoir, ce qu'il fait principalement. On ne peut PAS créer son propre ENUM, donc vous devrez ruser quand il s'agira de faire un pouvoir ne ressemblant à aucun existant. Après les paramètres du pouvoir s'en suivent dans le code concerné. Notez que l'on peut aussi modifier le temps de recharge du pouvoir.
- *Speech* : tous les discours et autres dialogues.
- *Terrain* : les textures des différents terrains, pour les mappers.
- **Upgrade** : quand vous souhaitez qu'un objet GAGNE une capacité, il vous faudra coder une upgrade ( « amélioration ») Vous allez abuser de ce fichier, et espérer ne pas atteindre le nombre maximal d'upgrades (je ne l'ai jamais atteint)
- *VictorySystem* : le système de victoire (oui je ne sais pas à quoi sert ce fichier)
- *Video* : le code pour les vidéos (comme pour les fichiers audios, on donne un nom au fichier pour le désigner dans le code). Le format .vp6 utilisé n'ayant pas été brisé, vous ne pouvez rien faire de ce côté (pour le moment)
- *Voice* : quelques autres discours et cris

- *Water* : le code pour l'eau, l'animation de l'eau, et les effets selon le moment de la journée
- *WaterTextures* : suite de Water.ini, avec les textures utilisées par l'eau
- **Weapon** : Si vous voulez armer votre objet, pour le défendre dans les dangereuses contrées de la Terre du Milieu, il vous faudra lui coder ses armes ici, avec des données aussi variées que la portée, le type de dégâts, la possibilité de manquer la cible etc ...
- *Weather* : des données sur la météo. Rien de génial, sauf si vous aimez la neige (changez « IsSnowing = no » en « IsSnowing = yes »)
- *WebPages* : lien vers les pages internet, avec des raccourcis pour être utilisés dans le code
- *WindowTransition* : les animations lors du passage d'un menu à un autre.
- *Campaigns* : tous les fichiers INI contrôlant les paramètres utilisés en campagne
- *Default* : les fichiers « types » pour chaque type de fichier INI, Ils sont assez sympa à regarder pour apprendre à facilement créer un bouton ou voir tous les paramètres pour les armes.
- *MappedImages* : tous les fichiers contrôlant les ButtonImages et les portraits, Ils permettent au moteur de jeu de savoir à quelle partie de l'image correspond le ButtonImage/Portrait
- *Objects/Cinematics* : tous les Objects utilisés dans les cinématiques, et que vous n'utilisez pas en jeu. En effet, les cinématiques de BFME sont faites en temps réelles, et non en images de synthèse
- *Objects/Civilian* : les objets utilisés pour les unités/batiments hors des 4 factions. Regardez bien le fichier CivilianBuildings.ini, on s'en servira plus tard
- *Objects/Demotest* : les objets utilisés lors de l'alpha interne du jeu chez EALA; tout est obsolète je crois
- **Objects/Evilfaction** : tous les objets liés directement aux factions du mal. Le fichier que vous utiliserez le plus sera probablement evilfactionunits.ini. On pourra remarquer qu'étrangement, les fichiers des hordes sont situés à part
- **Objects/Goodfactions** : les objets des factions du bien
- **Objects/Hordes** : les hordes des factions du mal, aucune idée pourquoi les développeurs les ont séparés, mais bon ...

- *Objects/Nature* : le titre parle de lui même
- *Objects/Neutral* : les objets utilisés sur les maps comme unités neutres, ou bien les Objects inclassables (Shelob, le Balrog ...)
- *Objects/Obsolete* : des objets qui ne sont plus utilisés, mais gardés pour l'équipe des cinématiques
- *Objects/System* : les objets utilisés par le système de jeu, donc ceux du Livre des Sorts entre autres.

Et voilà, nous avons vu TOUS les fichiers INI. Dans le prochain tutorial, nous apprendrons quelques manipulations basiques pour que vous puissiez commencer à impressionner vos amis ;-)