



2/ Premières Modifications

Après avoir vu dans la leçon précédente le rôle des nombreux fichiers INI, il est plus que temps de se lancer dans le premier TP. Nous allons ici faire quelques modifications très simples, mais toujours sympas à faire pour améliorer l'expérience de jeu.

a) gamedata.ini

Alors allons y, ouvrez gamedata.ini

Si vous vous rappelez bien, ce fichier contrôle la plupart des données du jeu. Cherchez, en cliquant sur ctrl + F, « Command »

Vous devriez normalement trouver un bout de code similaire :

```
;;; COMMAND POINTS FOR MULTIPLAY ;;;  
GoodCommandPointsMP2 = 200           ; 2 players  
EvilCommandPointsMP2 = 400  
  
GoodCommandPointsMP3 = 150           ; 3 players  
EvilCommandPointsMP3 = 300  
  
GoodCommandPointsMP4 = 150           ; 4 players  
EvilCommandPointsMP4 = 300  
  
GoodCommandPointsMP56 = 100         ; 5-6 players  
EvilCommandPointsMP56 = 200  
  
GoodCommandPointsMP78 = 80           ; 7-8 players  
EvilCommandPointsMP78 = 160
```

Comme les gars de chez EA sont très sympathique, ils nous ont laissé pleins de commentaires (remarquez les « ; ») dans le code pour nous simplifier la tâche. Donc comme le précisent les commentaires, on a ici les fameux points de commandements qui limitent le nombre d'unités disponibles en jeu. Si vous vous demandez pourquoi ils

sont copiés 5 fois, c'est parce qu'il faut les définir dans les différentes situations, selon le nombre de joueurs dans la partie.

Alors on va y aller, modifions les valeurs par ci par là. Essayez de prendre en compte votre PC pour ne pas le faire crasher. Le moteur de jeu SAGE ne crashera pas en premier, ça sera votre PC (je suis arrivé à 10 000 command points). Prenons par exemple :

```
;;; COMMAND POINTS FOR MULTIPLAY ;;;
GoodCommandPointsMP2 = 1200           ; 2 players
EvilCommandPointsMP2 = 1400

GoodCommandPointsMP3 = 1150           ; 3 players
EvilCommandPointsMP3 = 1300

GoodCommandPointsMP4 = 1150           ; 4 players
EvilCommandPointsMP4 = 1300

GoodCommandPointsMP56 = 1100          ; 5-6 players
EvilCommandPointsMP56 = 1200

GoodCommandPointsMP78 = 1080          ; 7-8 players
EvilCommandPointsMP78 = 1160
```

Voilà qui nous promet de belles batailles épiques =)

Notez que si vous voulez aussi modifier les points de commandements en campagne, il vous faudra éditer :

```
;;; CAMPAIGN BONUS CAPS
GoodCommandPointLimit = 300
EvilCommandPointLimit = 600
PowerLimit = 60
ResourceMultiplierLimit = 5.0
```

Voilà, une première petite modif sympa et facile à faire. On continue ?

Toujours dans Gamedata.ini, cherchez « Legolas » jusqu' à trouver :

```
;------Legolas-----
#define LEGOLAS_BUILD COST           3000
#define LEGOLAS_BUILD TIME          30
#define LEGOLAS_HEALTH              1000

#define LEGOLAS_LVL1_EXP_AWARD       70
#define LEGOLAS_LVL2_EXP_AWARD       80
#define LEGOLAS_LVL3_EXP_AWARD       90
#define LEGOLAS_LVL4_EXP_AWARD       100
#define LEGOLAS_LVL5_EXP_AWARD       120
#define LEGOLAS_LVL6_EXP_AWARD       150
#define LEGOLAS_LVL7_EXP_AWARD       180
#define LEGOLAS_LVL8_EXP_AWARD       210
```

```

#define LEGOLAS_LVL9_EXP_AWARD 250
#define LEGOLAS_LVL10_EXP_AWARD 300

#define LEGOLAS_LVL2_EXP_NEEDED 50 ;200
#define LEGOLAS_LVL3_EXP_NEEDED 100 ;400
#define LEGOLAS_LVL4_EXP_NEEDED 200 ;600
#define LEGOLAS_LVL5_EXP_NEEDED 300 ;800
#define LEGOLAS_LVL6_EXP_NEEDED 500 ;1000
#define LEGOLAS_LVL7_EXP_NEEDED 700 ;1200
#define LEGOLAS_LVL8_EXP_NEEDED 900 ;1400
#define LEGOLAS_LVL9_EXP_NEEDED 1100 ;1600
#define LEGOLAS_LVL10_EXP_NEEDED 1500 ;2000

#define LEGOLAS_PREATTACKDELAY 800
#define LEGOLAS_DELAYBETWEENSHOTS 0
#define LEGOLAS_FIRINGDURATION 0
#define LEGOLAS_RELOADTIME_MIN 1000
#define LEGOLAS_RELOADTIME_MAX 1000
#define LEGOLAS_BOW_RANGE 400
#define LEGOLAS_BOW_DAMAGE 60

#define LEGOLAS_HAWKSTRIKE_DAMAGE 200
#define LEGOLAS_HAWKSTRIKE_RANGE 410

#define LEGOLAS_ARROWSTORM_DAMAGE 80
#define LEGOLAS_ARROWSTORM_RANGE 400

#define LEGOLAS_KNIFE_PREATTACKDELAY 500
#define LEGOLAS_KNIFE_DELAYBETWEENSHOTS 1
#define LEGOLAS_KNIFE_FIRINGDURATION 500
#define LEGOLAS_KNIFE_DAMAGE

```

Vous devez sans doute vous demander ce que ce charabia veut dire. En fait, les plus matheux d'entre vous devraient comprendre assez rapidement.

Dans une équation du type : $3X + 1 = 2$, X est ce qu'on appelle en langage informatique une variable. X est une valeur, un nombre. Dans mon exemple, $X=1/3$. Mais dans le cas de Legolas, on utilise pas des X pour désigner les variables, car il y en a beaucoup trop. On utilise des variables composées de pleins de lettre, comme ça c'est plus facile à retrouver et à comprendre en lisant le code. Donc ici, prenons par exemple la variable LEGOLAS_BUILDCOST. Elle store le nombre 3000. En parallèle de Gamedata.ini, ouvrez Legolas.ini (object\goodfaction\units\rohan). Cherchez « LEGOLAS_BUILDCOST ». Vous allez normalement trouver le code suivant :

```

; ***DESIGN parameters ***
Side = Rohan
EditorSorting = UNIT
ThreatLevel = 5.0
ThingClass = CHARACTER_UNIT
BuildCost = LEGOLAS_BUILDCOST
BuildTime = LEGOLAS_BUILDTIME

```

Et c'est là qu'en plus de devoir être matheux, va falloir être un peu anglophone. **BUILDCOST** signifie « prix de construction ». Ainsi, cette variable auquel on fait

référence dans le code de l'object est définie dans gamedata.ini

Ca c'était pour les explications sur le format utilisé par EA, si vous n'avez rien compris c'est pas grave vous n'aurez qu'à modifier les valeurs dans gamedata.ini

Notez aussi qu'il n'est pas obligatoire d'utiliser des variables dans le code de l'object, on peut aussi mettre directement :

```
BuildCost      = 3000
```

Voilà, il ne vous reste plus qu'à vous armer d'un dictionnaire anglais et de trouver les variables que vous voulez modifier.

c) PlayerTemplate.ini, le code des factions

Vous pouvez maintenant fermer gamedata.ini, et ouvrir PlayerTemplate.ini
Son utilisation est comme vous le savez de gérer le code principal des 4 factions.

Prenons maintenant le bloc du Gondor

```
PlayerTemplate FactionGondor
```

```
Side          = Gondor
```

```
PlayableSide  = Yes
```

```
Evil          = No
```

```
StartMoney   = 0
```

```
MaxLevelMP   = 32
```

```
MaxLevelSP   = 78 ;80
```

```
PreferredColor = R:43 G:150 B:179
```

```
IntrinsicSciences = SCIENCE_GOOD
```

```
IntrinsicSciencesMP = SCIENCE_GONDOR
```

```
SpellBook    = GoodSpellBook
```

```
SpellBookMp  = GondorSpellBook
```

```
PurchaseScienceCommandSet = GoodSpellStoreCommandSet
```

```
PurchaseScienceCommandSetMP = GondorSpellStoreCommandSet
```

```
DisplayName  = INI:FactionGondor
```

```
DefaultPlayerAIType = GondorSkirmishAI
```

```
BeaconName   = MultiplayerBeacon
```

```
LightPointsUpSound = GondorLightPointsUp
```

```
ObjectiveAddedSound = Gui_MissionObjectiveNew
```

```
ObjectiveCompletedSound = Gui_MissionObjectiveCompleted
```

```
InitialUpgrades = Upgrade_GondorFaction
```

```
BuildableHeroesMP = GondorGandalf GondorBoromir GondorFaramir
```

```
RohanPippin
```

```
SpellStoreCurrentPowerLabel = APT:SpellStoreCurrentEvenstarPower
```

```
SpellStoreMaximumPowerLabel = APT:SpellStoreMaximumEvenstarPower
```

```
End
```

Ne vous étonnez pas si vous trouvez que mon code est plus petit, j'ai enlevé

tous les morceaux de codes qui n'étaient plus utilisés (ceux avec un « ; » devant)

Alors je ne vais pas me lancer dans une analyse précise de chaque ligne, on va garder cette tâche pour le tutorial sur créer une nouvelle faction =)

Intéressons nous plutôt à la ligne BuildableHeroesMP. Comme vous pouvez le remarquer, cette ligne est une suite de noms. Mais pourquoi taper GondorGandalf ? Il s'appelle juste Gandalf ! (Olorin pour les puristes de Tolkien). La réponse est que GondorGandalf est le nom d'object de Gandalf. Donc il va falloir que vous commenciez à farfouiller dans le dossier Object pour repérer les noms qui vous intéressent. Recherchons par exemple Gimli. Dans Gimli.ini, on peut voir dans les toutes premières lignes (et la première qui ne soit pas un commentaire) :

Object RohanGimli

Donc il vous suffira d'ajouter RohanGimli dans la ligne pour que ça ressemble à :

BuildableHeroesMP = GondorGandalf GondorBoromir GondorFaramir RohanPippin
RohanGimli

Vous pouvez maintenant mettre les héros existants dans la faction où vous le souhaitez =)

d) CommandSet.ini

Comme nous l'avons vu dans le précédent tutorial, les Commandsets sont les ensembles de boutons. J'ai remis l'image en bas de la page pour ceux qui auraient déjà oubliés. (voir Photo 1)

Alors donc nous allons commencer par lire un code de commandset, pour se donner une idée de comment ils fonctionnent. Notez que les commandsets des soldats que vous devrez modifier dans le cas des hordes est celui de la horde, et non de l'unité elle même (donc éditez CommandSet RohanPeasantHordeCommandSet et non CommandSet PeasantCommandSet) :

```
CommandSet RohanPeasantHordeCommandSet
  1 = Command_Guard
  2 = Command_Repair
  3 = Command_PurchaseUpgradeRohanForgedBlades
  4 = Command_PurchaseUpgradeRohanHeavyArmor
  5 = Command_PurchaseUpgradeRohanBasicTraining
    13 = Command_AttackMove
    14 = Command_Stop
    15 = Command_Guard
End
```

Alors comme vous pouvez le voir, on note tout d'abord Commandset <le nom du commandset>. Ainsi, le nom est une variable auquel on pourra faire référence dans le code de l'object. Mais ne vous affolez pas, je ne vais pas vous embêter avec ça, pour l'instant ...

Donc ensuite on a une série de numéro sous la forme X = <le commandbutton>

Ils sont présentés ainsi pour que le code puisse les interpréter. L'image 2 vous donnera une idée de l'interprétation d'un commandbutton par le moteur de jeu.

Donc on observe une architecture similaire à :

1
2
3
[palantir]
4
5
6

7- 15 sont cachés, mais on peut les utiliser pour des boutons avec des raccourcis.

Voilà, donc maintenant vous pouvez modifier les commandsets selon vos désirs, mais avec toutefois deux limites, que je vous apprendrais à contourner par la suite.

Vous ne pouvez pas modifier les commandsets des héros pour leur rajouter des pouvoirs, ce n'est pas aussi simple que ça, et aussi vous ne pouvez pas créer de nouveaux boutons, vu qu'on ne l'apprendra dans le prochain tutorial.

Voilà, vous avez assez de modifications à votre portée pour commencer à vous amuser pendant un petit moment.

Le prochain tutorial traitera de la création d'une unité, et sera sans doute lui aussi divisé en sous parties.

Bon modding !