

# BIEN DEBUTER AVEC REALMCRAFTER STANDARD

Voici un petit tutoriel pour vous apprendre à vous familiariser avec les outils de RealmCrafter et vous permettre de commencer avec les bases.

1. Project Manager p. 2-3
2. Le Game Editor (GE) p.4
3. Optimisation
4. Onglet Média, importation dans RC p.6
5. Création d'un terrain dans RC
6. Importation d'un terrain créé avec un autre logiciel p.9
7. Importer un personnage animé p.10
8. Les Gubbins
9. Les PNJ
10. Les Factions
11. Le Combat
12. Les Scripts
13. Trucs et Astuces
14. Glossaire
15. Comparatif version Standard et Pro p.25

#### 1. Le Project Manager

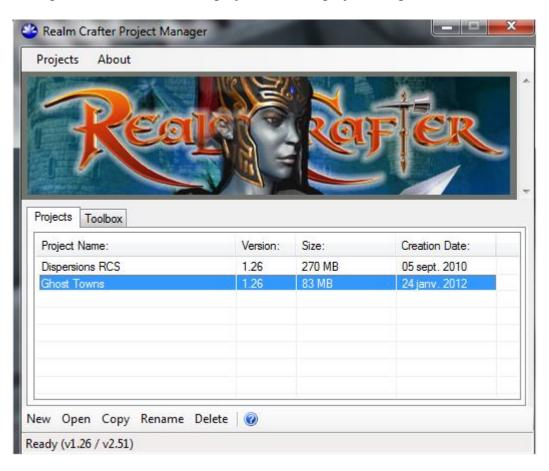
#### 1. Onglet **Projects**

C'est ici que seront listés vos différents projets, le bouton **New** vous proposera d'en créer un nouveau avec la version RC de votre choix (si vous avez les 2).

Double clic sur votre projet ouvrira le Game Editor (GE).

Le Game Editor (GE), c'est l'outil qui caractérise RealmCrafter, il permet de créer, d'agencer d'organiser, et de définir les paramètres de votre jeu. Tous les éléments terminés se combineront ici pour donner vie à votre jeu.

Vous pouvez créer un nouveau projet, ouvrir un projet, le copier, le renommer, ou le supprimer.



Tout en bas à gauche figure la ou les versions que vous possédez de RC, ici par exemple, il s'agit de RCS en 1.26 et RCP en 2.51

#### 2. Onglet Toolbox

A droite, de l'onglet **Projects** vous avez **Toolbox** 

ce second onglet permet de tester un projet et il liste également tous les logiciels fournis gratuitement avec RC ou achetés en plus.

Vous devriez avoir tous les outils ci-dessous par défaut (à l'exception de <u>RC Mega Terrain</u> qui s'achète et que j'ai mis en surbrillance en bleu en bas). Si ce n'est pas le cas, téléchargez et installez **RC Missing Tools**, et désignez-lui votre dossier RealmCrafter:

http://cachua.free.fr/RC\_Missing\_Tools\_Fix.exe



**Game Editor** = pour ouvrir le logiciel d'édition

**Test Server (Normal)** = lance le serveur (3 fenêtres s'ouvrent, cliquer sur **Unlock Updates server** pour démarrer le serveur)

**Test Server (MySql)** = lance le serveur Sql (personne n'a pu dire s'il fonctionnait correctement) ?

**Test client** = lance le client qui permet de tester/jouer

**Script editor** = ouvre le logiciel d'édition de script

**Architect** = Permet d'organiser l'agencement de vos maps et objets

**Rock Editor** = permet de générer des rochers

**Terrain Editor** = permet de créer des maps

**Tree Editor** = permet de créer des arbres

**Gubbin Tool** = ouvre le logiciel de customisation des persos, ajout d'accessoires

**Script Converter** = permet de convertir les anciens scripts en nouveaux (avant 1.20)

(RC Mega Terrain (PAYANT, non fourni par défaut) = permet de créer des maps plus grandes, plus détaillées mais plus lourdes, grosse chute de FPS)

#### 3. Test d'un projet

Assurez-vous d'avoir sélectionné votre projet en cliquant dessus dans l'onglet **Projects**, puis dans **Toolbox**:

#### \_ Lancez **Test Server** (normal)

3 fenêtres s'ouvrent: \_ **Account:** Pour voir les comptes des joueurs connectés (possibilité de les transformer en GM, les bannir etc...)

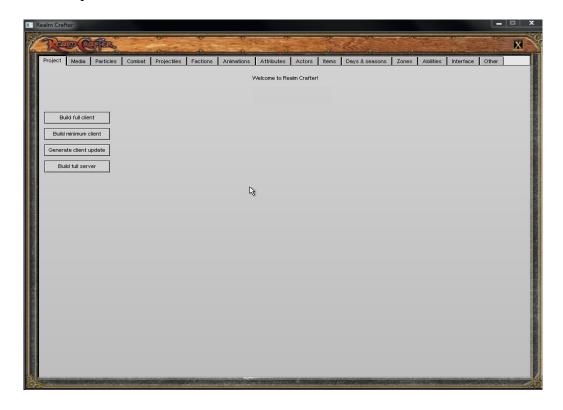
**\_ Game:** Pour poster des messages ingame, voir les joueurs connectés dans chaque zone et voir la fenêtre de Chat

\_ Updates: Pour lancer le serveur, cliquez sur Unlock Updates server

\_ Lancez **Test Client:** Connectez-vous avec les identifiants par défaut et testez votre jeu.

## 2. La GE (GAME Engine ou Game Editor)

C'est le logiciel d'édition, il se présente comme une fenêtre composée de plusieurs Onglets, qui renvoient à des éléments précis de la création.



Le 1er onglet **PROJECT** servira à compiler notre projet et ses éventuelles mises à jour.

L'onglet **MEDIA** permet d'importer nos médias (3d, 2d, musiques et sons)

L'onglet **PARTICLES** permet de créer des particules (pour les attaques, soins etc..)

L'onglet **COMBAT** permet de définir le type de combat (4 possibilités) et le type de dégâts

L'onglet **PROJECTILES** permet de créer un projectile, sort, boule de feu etc..

L'onglet **FACTIONS** permet de définir l'affinité entre chaque race/classe

L'onglet **ANIMATIONS** permet d'attribuer les clefs d'animations à une action pour chaque perso

L'onglet **ATTRIBUTES** permet de définir les attributs par défaut de chaque perso

L'onglet **ACTORS** permet de créer/définir son personnage et ses attributs de départ

L'onglet **ITEMS** permet de créer/définir un objet personnage et ses attributs.

L'onglet **DAYS & SEASONS** permet de définir le temps en jeu, la saison, les rayons solaires

L'onglet **ZONES** permet d'organiser les maps créées au préalable

L'onglet **ABILITIES** permet de créer de nouveaux skills via les scripts qui les déclenchent

L'onglet **INTERFACE** permet de modifier l'organisation de l'interface affichée en jeu

L'onglet **OTHER** regroupe diverses options incasables (collision, Bulles de Chat, Monnaie etc..)

#### 3. Optimisation

Avant de commencer quoi que ce soit, il faut comprendre les limites de <u>RealmCrafter Standard:</u>
RealmCrafter a été conçu en **BlitzBasic**, langage de programmation BASIC et moteur graphique DirectX 7 :

\_ les spécificités de DirectX7 sont simples: Les textures doivent être carrées et multiple de 2. donc 512 x 512, 256 x 256, 128 x 128, 64 x 64 etc...

Une texture 256 x 512 consomme autant de VRAM qu'une 512 x 512, car le .b3D la réajuste en carré en se basant sur le côté le plus grand...

**Evitez les textures en 1024 x 1024!** (elles consomment 4Mb de VRAM!)

512 x 512 maximum, Soit 1Mb de VRAM donc 4 X moins lourde, l'idéal en 256 x 256.

La taille des textures fait ralentir le client, l'utilisation des textures a un impact de 10-30% sur les FPS.

\_ Travaillez avec des textures en format .dds (DXT3), c'est le format le moins lourd.

#### \_ Modéliser les personnages avec 2000 faces maximum.

-50 factions

RC a aussi des limites par zone que vous devez connaitre:

- -500 items
- -150 triggers
- -250 waypoints
- -100 portails
- -100 points de spawn
- -50 actors maximum par zone (conseillé pour éviter le lag)

#### Les collisions doivent être calculées au minimum.

<u>Conseil perso:</u> Cochez *Disable actor-> Actor collisions* dans l'onglet Other de la GE. ça signifie que vos personnages passeront au travers les uns des autres en jeu.

## \_ BlitzBasic supporte très peu de clef d'animation!

La documentation officielle de RC conseille **20 bones** pour les persos jouables.

La taille du fichier *Actor* est responsable du "loading lag" (chaque fois qu'un perso entre sur une map, RC lag) **Trop de bones animés = Updates plus lourde pour le serveur donc réponses plus longues = Lag en jeu!** 

Sauvegardez votre Actor en .B3D et assurez-vous que le poids du fichier soit inférieur à 200 ko!

Le perso fournit avec RC met presque 3 secondes à charger, il pèse 1,10mo!

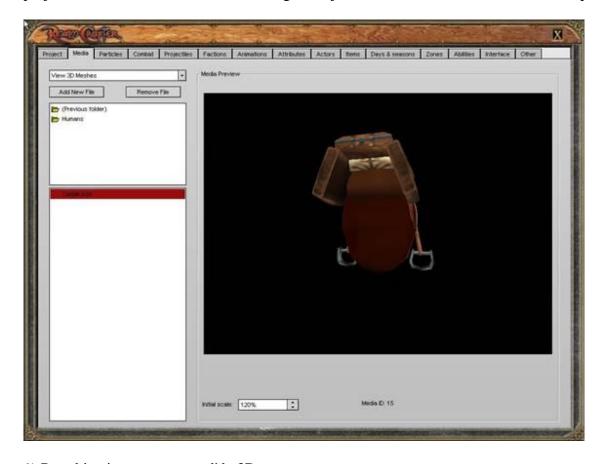
Imaginez avec 10 persos sur la map... A l'écran Création de personnage, cliquez sur le perso,

le temps qu'il mettra à apparaître sera le même en jeu, donc si c'est trop long => lag ingame assuré...

Perso RC par défaut	Mon perso
1308 faces (sans armure)	1600 faces
2 textures 512x512, Head et Body, en .png	512x512 body et 256x256 Head, en .dds
32 bones (5 bones inutiles, normalement 27)	20 bones
32 Séquences animées	30 Séquences animées
<b>1539 Frames au total</b> (32 seq de 48 Frames)	<b>1020 Frames au total</b> (30 seq de 34 Frames)
1.10mo en .B3D (1.93mo encrypté en .eb3d)	188ko en .B3D (un fichier .x fait le double)
Temps de chargement = 3 Secondes !!!	Temps de chargement = 0 !

## 4. Onglet Média, importation dans RC

L'onglet le plus important est **l'onglet Média**. C'est à partir de lui que l'on charge tous les éléments de notre projet dans RealmCrafter afin de les intégrer en jeu, modèle 3D, textures, sons et musiques:



#### 1) Pour bien importer un modèle 3D:

\_ Pensez à créer un dossier avec **votre modèle 3D** <u>ET</u> ses textures (dans le même dossier!) et importer tout le dossier. Valable pour les persos, les objets, les bâtiments etc...

\_ Si vous chargez un personnage, il doit être obligatoirement accompagné de 2 textures (fictives, elles peuvent être complètement blanches par exemple) nommées **Head.bmp** et **Body.bmp**, elles doivent être affectées à la tête et au corps du perso <u>avant de les importer</u> (sinon ça ne sert à rien). ça permet à RC de différencier la tête du corps et donc de modifier ces textures plus facilement en jeu.

Dans l'onglet MEDIA, Chargez votre modèle 3D dans "View 3D Meshes" ADD A NEW FILE.

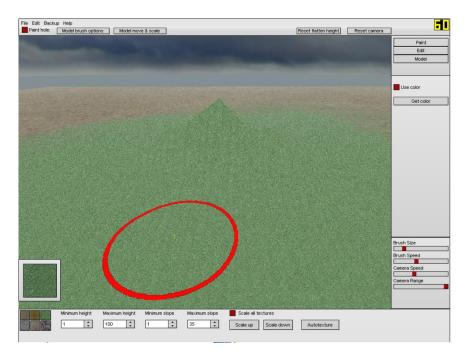
Ensuite dans "View TEXTURES" chargez les vraies textures une par une.

#### 2) Les erreurs:

- \_ si les textures ne sont pas dans le même dossier que le mesh
- \_ si le perso n'est pas accompagné au minimum de 2 textures : <u>Head.bmp</u> et <u>Body.bmp</u>
- \_ si le perso n'a pas de squelette et que vous cliquez sur **Animated** (il doit <u>au moins</u> avoir un bone Head pour être considéré comme jouable)
- \_ Ne jamais modifier ou supprimer de dossier dans votre répertoire RealmCrafter, vous feriez planter le programme.(Memory Access Violation = MAV) Sa structure ne doit pas changer !

## 5. Création d'un terrain dans RC

Ouvrir RCTE fournit avec RC, vous le trouverez dans le **PROJECT MANAGER**, dans la **Toolbox**, Terrain Editor. Le curseur jaune correspond à la taille d'un perso en jeu.



#### 1. Commandes et interface

- \_ Pour se déplacer, maintenir ESPACE + click gauche ou click droit (Comme dans l'onglet Zone de la GE)
- \_ Pour agir sur le terrain, appliquer un effet, une texture etc... = Click droit
- \_ Pour modifier la taille du Terrain, vous trouverez l'option "**Edit**" dans la barre du haut: SCALE UP et DOWN, ou + et également pour modifier la taille du terrain.
- 3 boutons dans la colonne à droite:

**PAINT**= pour poser les textures. Pour modifier la teinte d'une texture, cocher USE COLOR, et choisissez la teinte en cliquant sur GET COLOR. Pour effacer la teinte sur votre texture, choisissez simplement la teinte blanche.

Pour modifier la texture cliquez sur une des 6 cases en bas à gauche, puis cliquez sur la fenêtre du dessus où la texture sélectionnée apparait en plus gros, une fenêtre s'ouvre, cliquez sur la texture de votre choix. peignez avec le click droit.

Si le terrain a été agrandi, la texture sera floue, ajustez alors sa taille avec les boutons Scale UP et Scale Down en bas à droite, cochez SCALE ALL TEXTURES, si vous souhaitez toutes les ajuster en même temps.

**EDIT**= permet de modifier le terrain RAISE/LOWER land pour les hauteur, Pull land, tire le terrain, smooth l'arrondi, flatten, le 2nd curseur bleu définit la hauteur du terrain et l'aplati.

**MODEL**= permet de déposer ses modèles 3D sur la map pour tester, et de modifier leur taille.

**Brush Size**= définit la zone d'action de votre curseur

**Brush Speed**= définit la vitesse d'action (la force)

Camera Speed= définit la vitesse de déplacement de votre caméra sur la map

Camera Range= définit la visibilité maximale de votre camera (mettez la à fond c'est plus pratique pour bosser.

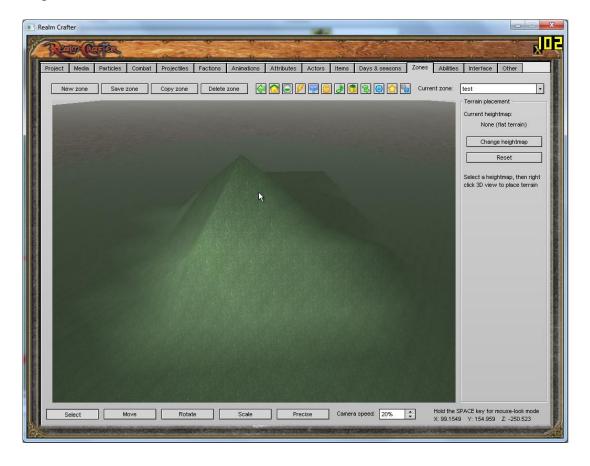
#### 2. Exportation et sauvegarde

File --> **EXPORT**, à **ENCRYPT TERRAIN** choisissez YES si vous souhaitez l'encrypter pour le protéger, à **KEEP NATIVE TERRAIN**, choisissez YES pour conserver le terrain avec les textures d'origine et sans les modèles.

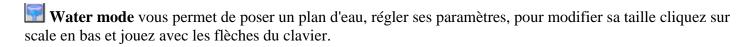
ATTENTION File -> Save n'exporte pas la map dans la GE! mais la sauvegarde simplement pour RCTE

#### 3. Onglet **Zones**

Dans la GE, vous trouverez votre map dans l'onglet **ZONES**, dans le petit panneau déroulant "nom de votre map".



- Environnment Options vous permet de placer le ciel, la visibilité maximale en jeu, de régler le climat etc...
- Portal Mode vous permet de placer un portail de départ pour votre perso, nommez-le "debut" par exemple, puis cliquez droit sur la map pour le placer, régler sa position avec le bouton PRECISE
- Scenery Mode, les scenery sont les objets, cette option vous permet de loader vos batiments, arbres et autres décors sur la map avec le click droit (pensez à les charger dans l'onglet MEDIA avant!)
- Emitters Mode, les particles emitters sont les effets de particules, click droit pour les poser sur la map



- Collision box mode pour placer des murs invisibles pour bloquer l'accès à certaines zones
- Sound zone mode pour placer des zones de sons et de musiques
- Trigger mode pour placer des déclencheurs de script
- Waypoint Mode pour placer des points de déplacements de vos PNJ et mobs, les temps de pause, les temps de re-spawn en cas de mort etc...
- Other options pour les options diverses dont l'image de chargement de la map
- Terrain mode pour importer un terrain créé depuis un autre logiciel, on ne peut importer que le heightmap (plan de la map en noir et blanc), pas la map elle-même.

## 6. Importation d'un terrain créé avec un autre logiciel

Dans onglet MEDIA (Textures), on charge d'abord le heightmap,

Dans l'onglet **ZONES**, sélectionnez **NEW ZONE**, puis l'option **Terrain Mode**, cliquez sur **Change Heightmap** et faites un click droit au milieu de la fenêtre pour y placer la map.

La map apparait toute plate, modifier ses hauteurs, sa position, sa rotation avec le bouton **PRECISE** en bas, **Scale Y** pour modifier les hauteurs et lui donner du relief.

**CHANGE COLOUR MAP**= vous permet de placer une seule texture de couleur pour votre terrain, sinon la map est blanche (c'est le problème avec les heightmaps provenant d'autres éditeurs)

**CHANGE DETAIL MAP**= vous permet de rajouter une texture de détail pour enrichir et diversifier votre colour map de base.

**Detail map scale**= augmente la quantité de détail de votre terrain 100 maxi (plus lourd)

Maximum triangles= augmente la qualité visuelle de votre terrain 50 000 maxi (plus lourd)

**Enable Morphing** et **Shading** contribue à améliorer la qualité avec effet d'ombres (toujours plus lourd si vous l'activer).

**SAVE ZONE** pour sauvegarder vos travaux.

## 7. Importer un personnage animé

#### Rappel des erreurs:

- \_ si les textures ne sont pas dans le même dossier que le mesh.
- \_ si le perso n'est pas accompagné de 2 textures <u>Head.bmp</u> et <u>Body.bmp</u>
- \_ si le perso a plus de 37 bones! (20 bones conseillés voir ci-dessous)
- <u>Sans squelette ou sans bone Head</u>, le personnage sera non jouable! (case grisée)

Pour les persos animés, il faut <u>obligatoirement et au minimum</u> un bone nommé HEAD pour la tête.

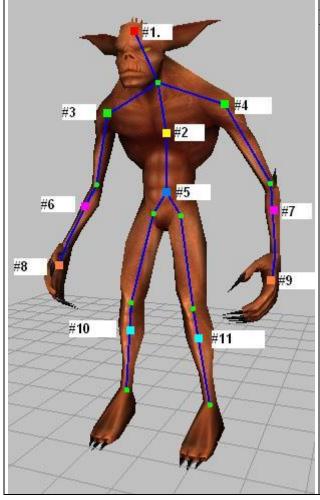
Sans ça, RealmCrafter considérera que c'est un objet statique et donc non jouable.

Dans l'onglet ACTORS la case ACTOR IS PLAYABLE sera grisée.

Donc avant d'importer votre personnage, assurez-vous donc qu'il dispose bien d'un squelette (renommer le bone de la tête en HEAD) et d'un set d'animation.

Vous allez devoir renommer les bones de cette façon afin de pouvoir accueillir des **Gubbins**, pour y placer des parties d'armures ou des armes, voici la liste des noms de bones que RCS reconnait. Vous pouvez le faire via des logiciels de modélisation 3D comme Blender, ou des logiciels d'animation gratuits comme PaceMaker ou FragMotion:

Voici un squelette-type a reproduire et utiliser pour RealmCrafter (avec, ici, la position des gubbins)



# 1. Head (Obligatoire)

- 2. Chest
- 3. R\_Shoulder
- 4. L\_Shoulder
  - 5. Pelvis
- 6. R\_Forearm
- 7. L\_Forearm
  - 8. R\_Hand
  - 9. L\_Hand
  - 10. R\_Shin
  - 11. L\_Shin

#### 1. Onglet Actors

#### Cliquez sur **NEW ACTOR**:

*Description* = Actor Race, renommez-la, et définissez sa faction, mettez Traders pour commencer (cf. plus bas le principe des factions)

*General* = Dans Start Area, choisissez votre map de départ, en face écrivez le nom de votre portail de départ, ici "debut".

Maintenant Cochez **ACTOR IS PLAYABLE** pour que votre perso soit disponible dans votre écran de sélection au début du jeu.

Appearance = MALE BODY MESH -> chargez votre modèle 3D du perso en cliquant sur CHANGE. ensuite chargez les textures en cliquant sur MALE FACE TEXTURE et MALE BODY TEXTURE, vous pouvez en définir 5 différentes qui seront disponibles au début du jeu.

*Attributes* = Cliquez sur chacun des attributs et réglez-les toutes à 10 au lieu de 0, sinon le perso ne bougera pas si la vitesse reste 0, sera mort si Health est à 0 etc...

*Preview* = Pour visualiser votre perso prêt pour son intégration en jeu, si une texture est inversée ou un truc cloche.

Pour sauvegarder, cliquez sur SAVE ACTORS

#### 2. Onglet **Animations**

Seulement si vous avez créé vos propres persos et animations! Si votre personnage est animé il faut maintenant définir ses animations dans l'onglet ANIMATIONS. cliquez sur NEW ANIMATION SETS, sélectionnez le New set et renommez-le "humain" par exemple.

Cliquez sur l'action qui vous intéresse Walk (marcher) de 1 à 28 par exemple, ensuite Run (courir) et Idle (debout) et inscrivez les frames de départ et de fin de vos animations. Modifiez également la vitesse de vos animations en réglant le % en dessous dans Animation speed.

Pour sauvegarder, cliquez sur SAVE ANIMATIONS SETS

On retourne très rapidement dans l'onglet **ACTORS** et dans: *General* = en bas dans **Male Animation Set**, choisissez "Humain"

Pour sauvegarder, cliquez sur SAVE ACTORS

Il est maintenant prêt à aller gambader sur votre map.

## 8. Les Gubbins

De mémoire, il arrive que le Gubbin editor fasse apparaître le mesh si minuscule qu'on a l'impression qu'il n'a pas été chargé.

De même, effectuer une rotation sur le mesh avait tendance à le déformer.

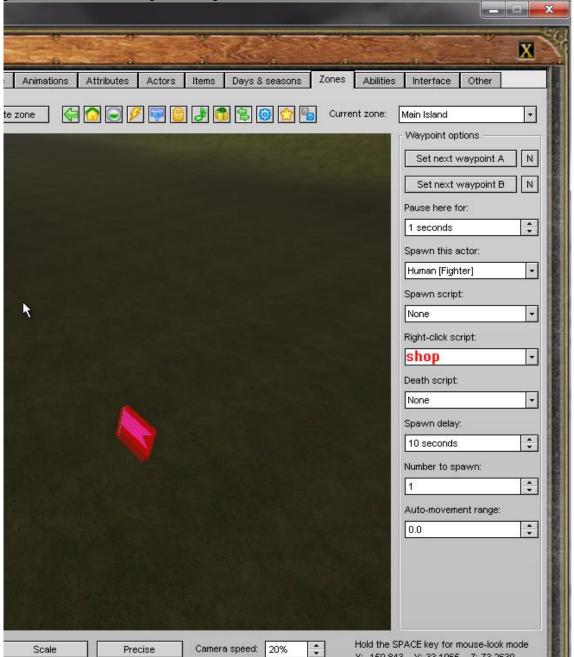
L'astuce consistait alors à importer le mesh dans RC avec un bone, pour empêcher les déformations de skin.

## 9. Les PNJ

Dans RealmCrafter, les PNJ se positionnent sur la map via des Waypoints.

Symbolisés par de petits drapeaux, ils définissent la position d'un actor sur une map, et ils peuvent également

permettre de définir le parcours que fera cet acteur.



1. Onglet **Zones** 

Chargez une map.

Cliquez sur Waypoint Mode, et cliquez droit sur la map pour déposer <u>un drapeau</u> (point de spawning = là où le PNJ apparaitra).

#### Spawn this Actor:

On spawne un Human [Fighter] par défaut.

 $\Delta$  Quand vous spawnez un actor, assurez-vous que le drapeau soit toujours légèrement au dessus du niveau du sol, sinon il risque de passer à travers le sol et tomber dans le vide, et vous penserez qu'il ne spawne pas du tout sur la map.

Vous pourrez plus tard définir:

Pause here for:
Spawn script:
Right-Click Script:
Death Script:
Spawn delay:
Number to Spawn:
Auto-movement Range:
2. Option PVP
(Si vous souhaitez que la zone soit destinée aux combats Joueur contre Joueur, PVP)
Dans <b>Other Options</b> , cochez <b>PVP enabled</b> , pour rendre possible le combat entre joueur sur cette map. Pensez-vous à le cocher ou à le décocher en fonction de vos maps. Sauvegardez la zone.
3. Onglet Factions (voir chapitre suivant) On clique sur Humans dans la liste de gauche
Adjust this faction's rating with: on choisit aussi Humans dans la liste pour définir le comportement des humains entre eux
This faction's rating with that faction is: 0% (statut neutre qui nous permet d'attaquer quand même)
Sauvegardez la zone et testez en jeu.
4. Test en jeu
Dans le Project Manager de RC, cliquez d'abord sur votre projet puis dans l'onglet Toolbox:
_ Lancez <b>Test Server (normal)</b> Dans la fenêtre <b>Updates</b> , lancez le serveur en cliquant sur <b>Unlock Updates server</b>
_ Lancez Test Client

## 10. Les Factions

Les factions permettent de gérer la diplomatie et l'affinité des races entres elles.

#### dans l'onglet FACTIONS

Vous changez les paramètres de factions, avec les faction ratings qui vont de -100 a 100.

-100 > 25: hostile 25 > 60: ami

60 > 100 : allié ( aide le personnage attaqué en cas de script avec 'actorcallforhelp' )

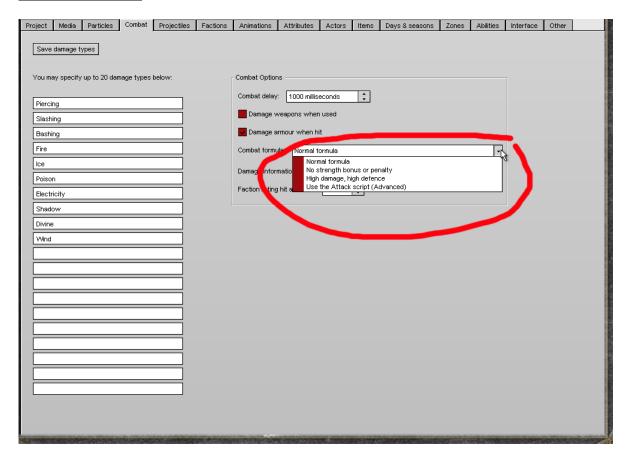
s'il a 100 avec le joueur et qu'il est attaqué, il ne réagira pas quelle que soit son mode de combat.

s'il a **-100** avec le joueur et qu'il est en 'Always attack' il prendra l'initiative d'attaquer dès que vous êtes a une distance inférieure a son paramètre d'agressivité.

si le PNJ à **-100** avec le joueur mais qu'il est en '**Defensive**' il n'attaquera que si il est attaqué. Il faudra aussi peut-être scripter sa réaction: le faire attaquer et appeler a l'aide.

Il y a 3 fonctions a connaitre : SetActorAIState(IA, 3) SetActorTarget(IA, attaquant) ActorCallForHelp(IA)

## 11. Le combat :



RealmCrafter propose 4 options pour le combat:

- \_Normal Formula
- \_ No strength bonus or penalty
- \_ High damage, high defence

Les 3éres sont déjà hard codées dans le programme, il n'y a qu'à les sélectionner pour qu'elle fonctionne. La dernière nécessite que vous scriptiez votre propre système de combat.

\_Use the Attack Script (Advanced)

Gardez à l'esprit que que le code est toujours plus rapide que le script (en calcul/exécution).

#### Definition des variables utilisées dans le code:

armor - Stat de l'armure totale, de toutes les armures portées par l'actor attaqué

weapon - Stat d'attaque de l'arme utilisée par l'attaquant

strength - Stat de force de l'attaquant

toughness - stat de résistance de l'actor attaqué

damage - dégâts infligés à l'actor attaqué

rand (x, y) - un nombre entre x et y

/ - divisé par

#### 1. Normal Formula

L'idée basique derrière cette formule:

"Si vous êtes assez fort pour utiliser votre arme, vous recevez un bonus, et si vous êtes trop faible, vous recevez une pénalité."

L'inconvénient, c'est que si vous continuez à augmenter votre stat force, une fois que vous avez le bonus, le bonus n'augmente pas. Donc les dégâts sont essentiellement basés sur votre arme plus ou moins un ensemble de bonus/pénalité, en fonction de la comparaison arme/force:

Si l'attaquant n'est pas équipé d'une arme ou que l'arme fait des dégâts en dessous de 1%

Si l'attaquant est équipé d'une arme qui ne fait pas de dégâts inférieur à 1%, **ET** que sa force est inférieure à celle de son arme

```
\equiv damage = (weapon - rand(5, 8)) - (armor + (toughness / 8))
```

Si l'attaquant est équipé d'une arme qui ne fait pas de dégâts inférieur à 1%, **ET** que sa force est supérieur à celle de son arme

Si l'attaquant est équipé d'une arme qui ne fait pas de dégâts inférieur à 1%, **ET** que sa force est égale à celle de son arme

#### 2. No Strength Bonus or penalty (Sans Bonus, ni Pénalité de force)

Même chose que la normal formula, mais sans bonus, ni pénalité de force:

Si l'attaquant n'est pas équipé d'une arme ou que l'arme fait des dégâts en dessous de 1%

```
damage = ( (strength / 8) + rand (-5, 5) ) - (armor + (toughness / 8) )
```

Si l'attaquant est équipé d'une arme qui ne fait pas de dégâts inférieur à 1%

```
damage = weapon - (armor + (toughness / 8))
```

#### 3. High Damage, high defence (Gros dégâts, grosse défense)

Celle-ci est différente des 2 autres, Force et Résistance joue un rôle plus dominant:

"Les dégâts augmentent en même temps que la stat force augmente".

Dans le code source de RC Standard, elle est appelée "Multiplied Formula" (formule multiplicatrice) Si l'attaquant n'est pas équipé d'une arme ou que l'arme fait des dégâts en dessous de 1%

```
damage = (strength + rand (-10, 10)) - (armor * toughness))
```

Si l'attaquant est équipé d'une arme qui ne fait pas de dégâts inférieur à 1%,

```
    damage = (weapon * strength) - (armor * toughness)
```

#### Autre bases communes aux 3 formules

- Les valeurs de Resistance sont additionnées ou soustraites des dégâts, ça dépendra de la résistance de l'actor attaqué au type de dégâts. Note: C'est légèrement différent de la "multiplied formula", mais ça a essentiellement le même effet.
- Chaque attaque à 90% chance de toucher. Si une attaque échoue, le joueur attaqué jouera l'animation "parry" (Défense).
- Il y a 10% chance de faire un critical hit, ce qui génère des dégats x2. En réalité il y a 10% chance de 90% chance de le faire, soit 9% de chance de faire un critical hit.
- Si les dégâts sont inférieurs à 1, alors dégâts = 1

Beaucoup de personne choisisse l'option du combat scripté (**Use the Attack Script**), mais il y a un avantage de performance en utilisant l'une des 3 formules hard codées, vu que le code est plus rapide que le script.

Aussi pour les utilisateurs des sources, ce guide peut être utilisé pour les aider à choisir la formule qui ressemble le plus au système de combat qu'ils ont en tête, alors vous pourrez modifier les sources pour faire absolument tout ce que vous voulez, comme une alternative au combat scripté.

Les options hard codé du combat que RC propose sont souvent plus délaissé au profit du combat scripté. L'avantage d'utiliser le combat hard codé c'est qu'il est plus rapide et plus efficace côté serveur que le combat scripté.

## 12. Les Scripts:

Dans RealmCrafter, une partie des mécanismes de jeu les plus basiques est déjà hard codée. Pour le reste, et pour les systèmes plus spécifiques ou propres à votre jeu, il faudra avoir recours au scripting.

Depuis la version 1.25, RealmCrafter utilise le **BVM** (**Brisk Virtual Machine**) en langage de script. C'est un langage très simple qui demande de la logique et de la vigilance (anglais et un peu de maths):

------

Prenons un exemple avec ce script de Heal (récupération de points de vie):

```
Using "RC_Core.rcm"
; Date/Time: 14:49:57 on 17/04/2012
; By Giuliani on GIULIANI-PC

Function Main()

Player = Actor()

AnimateActor(Player, "Default attack", 0.05, 1)
CreateEmitter(Player, "HealthSpell", 29, 1000)
Max = MaxAttribute(Player, "Health")
SetAttribute(Player, "Health", Max)

Return

End Function
```

#### Maintenant détaillons ensemble ce script

```
Using "RC Core.rcm"
                                          <= cette partie ne sert plus à rien depuis RC 1.25
; Date/Time: 14:49:57 on 17/04/2012
                                          <= Point virgule ";"en début de ligne, il permet de rajouter
; By Giuliani on GIULIANI-PC
                                          <= des commentaire non pris en compte par le code
Function Main()
                                          <= Fonction principale obligatoire
 Player = Actor()
                                          <= on commence d'abord par définir les variables
                                          <= ici Player va désigner le joueur principal ( Actor() )
       AnimateActor(Player, "Default attack", 0.05, 1) <= On fait jouer l'animation du joueur
        CreateEmitter(Player, "HealthSpell", 29, 1000) <= On fait jouer l'émitter de particule
        Max = MaxAttribute(Player, "Health") <= On définit l'attribut maximum du joueur, HP
       SetAttribute(Player, "Health", Max)
                                                     <= On déclenche l'attribut max
```

Return

End Function

#### 2. Les variables

Afin de réaliser nos scripts, nous aurons à utiliser des variables:

Par exemple, dans un RPG, pour définir des points de vie, on peut dire HP=4, ceci est une variable. Les variables peuvent êtres des nombres entiers, à virgule ou des chaînes de caractères (texte)

```
1) Les INT (integer= nombres entiers) symbole % 12 -18 5000
```

# 2) Les **FLOAT** (floating-points= nombres à virgule) symbole #

3.14 -19.234 10.0

## 2) Les **STRING** (texte) symbole \$

```
"Hello, world!"
"1234"
"I ate 10 hamburgers."
```

Les variables (et leur valeur) ont une durée de vie limitée, elles ne durent que jusqu'à la fin d'un script, elles sont ensuite détruites et recyclées.

Vous pouvez utiliser le même nom de variable dans différents scripts, mais chaque variable sera différente de celle d'un autre script et pourra avoir une valeur différente.

Il existe un moyen de conserver leur valeur, d'un script à un autre, en utilisant les Global.

#### 3. Sauvegarde

Les scripts se sauvegardent au format .rsl Ils sont à placer dans ce dossier:

\Projects\[Votre Projet]\Data\Server Data\Scripts

#### 4. Vérification

Pour vérifier la validité d'un script rapidement, on lance le serveur de jeu et on le ferme aussitôt, puis dans:

\Solstar Games\Realm Crafter 1\Projects\[Votre Projet]\\\Data\Logs\]

#### On ouvre **Server Log.txt**

S'il y a une erreur dans votre script, quelque chose comme ça devrait s'afficher:

```
[22:40:06] Loaded 22 scripts...
[22:40:06] Script log: Line 24 of file marchand2.rsl: parse error
```

**Parse error** signifie que vous avez fait une faute de frappe (orthographe, oubli de virgule, de parenthèse etc...) à la ligne 24 de votre script "marchand2.rsl"

Donc, ce n'est même pas la peine d'essayer votre script en jeu, il ne se lancera pas.

**Invalid Stream Handle** signifie que vous essayez d'accéder à un fichier externe (texte, base donnée etc..) qui n'existe pas ou qui n'a pas été créé.

Recherchez l'erreur, corrigez-la et relancez le serveur pour écrire un nouveau Server Log

$\Delta$ Attention aux Majuscules/Minuscules, le BVM est sensible à la casse.
$\Delta$ <b>Attention</b> aux oublis de ponctuation (parenthèses, virgule, guillemets etc)
$\Delta$ <b>Attention</b> ne pas utiliser de pourcentage puisque % appelle la variable INT (nombre entier).
$\Delta$ <b>Attention</b> il doit y avoir autant de <b>ENDIF</b> que de <b>IF</b> .
$\Delta$ <b>Attention</b> n'utilisez jamais de chiffre dans le nom d'une fonction ex: Function Trade2 ()
$\Delta$ <b>Attention</b> n'utilisez jamais de chiffre dans le nom d'une variable ex: Player3 = ContextActor ()
5. Lecture d'un fichier externe (texte, base de donnée etc)
Avec RCS, vous pouvez gérer des données, listes, tableaux et autres depuis un fichier texte externe. (Parce que le serveur MySql ne fonctionne pas).
Le fichier externe que vous créerez pour pouvoir stocker vos données sera à placer dans le dossier, au dessus du dossier script:
Data\Server Data\Script Files
Si vous ne créez pas de fichier, vous aurez l'erreur Invalid Stream Handle et le serveur plantera.
Vous pourrez ensuite piloter et avoir accès à ces fichiers, en scriptant le fichier In-game Commands.rsl
pour ce faire vous devrez utiliserez ces commandes:  _ OpenFile()  _ ReadFile()  _ WriteFile()  _ CloseFile()  _ DeleteFile() etc

Je vous invite à jeter un coup d'œil aux commandes de scripts section File Commands

Assurez-vous que ces fichiers soient ouverts pour un temps très court. Un système peut seulement laisser ouvert quelques fichiers en même temps. Aussi si 2 personnes tentent d'écrire en même temps dans le même fichier, vous risquez de perdre les données. Ce système est plus risqué que MySql, mais comme le serveur ne fonctionne toujours pas...

## 13. Trucs et astuces :

#### Pour obtenir de la transparence sur une texture:

- \_ Avec <u>un système de masque</u>, devant le nom de la texture on place **M**\_ (on colore en noir 0,0,0 la partie qu'on veut rendre invisible)
- \_ avec <u>une couche Alpha</u>, devant le nom de la texture on place A\_ (on définit la partie masquée avec une couche alpha, sinon tout sera semi-transparent)

Les Masques sont moins gourmands que les couches alpha pour le GPU (Processeur Graphique) mais laissent des contours noirs et captent mal la lumière.

Pensez également à optimiser la texture grâce aux différents formats:

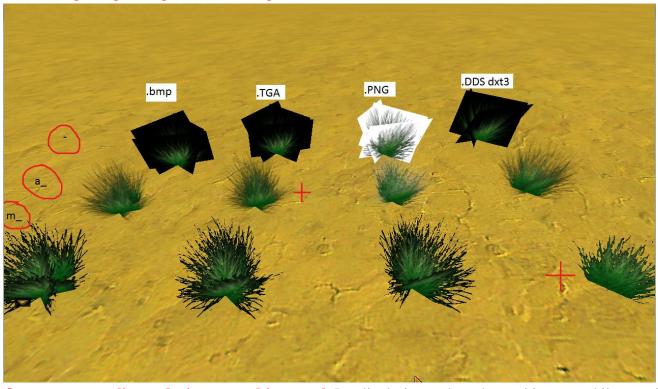
Après quelques tests le .TGA est le meilleur avec une couche alpha (plus de détail) et moins lourd qu'un .BMP Le .DDS est le mieux optimisé (dxt3) pour tout, très léger, mais il perd en netteté, peut-être avec un format plus lourd (dxt5?)

#### Evitez:

.PNG = pas mal mais contours blancs sur de l'alpha

.BMP = trop lourd

.JPG = Trop compressé, pas de couche alpha



On ne peut pas cliquer droit sur un objet au sol. Le clic droit est réservé aux objets quand ils sont dans votre inventaire.

Si vous voulez que vos joueurs puissent cliquer droit sur des objets de votre monde, alors vous devrez les charger en tant qu'**Actor**.

On ne peut pas faire grand chose avec des objets scenery (statiques). A part des éléments de décor. Si vous voulez qu'un objet bouge, etc, il faudra donc en faire un Actor. ça peut paraître étrange d'avoir à définir un objet statique en tant qu'actor, mais c'est comme ça que RC fonctionne.

#### Pour faire un actor:

- Le mesh de l'actor doit être chargé en tant que "animated" quand on l'importe dans l'onglet "media" de la GUE.
- Il doit avoir au moins un bone, nommé "Head".
- Il doit avoir au moins une animation, avec au moins un frame. ce qui signifie que ça nécessite un second bone (sinon, l'animation ne fera pas bouger le mesh).
- Il doit avoir un "Animation Set" dans l'onglet de la GE. Vous devez copier un set existant (i.e. "Human [Fighter]") sinon ça ne marchera pas. et conserver toutes les autres animations, même si vous n'en avez pas besoin. faites les juste commencer et finir à zéro.
- -Les attributs Health (points de vie) et Speed (vitesse) doivent être supérieurs à zéro. Sinon votre actor mourra des qu'il apparaitra et restera immobile!

L'outil Gubbin peut faire apparaître un objet si petit, qu'on croirait qu'il n'est pas là (i.e. une épée dans la main d'un joueur). Je pense que ça arrive quand on fait faire une rotation à l'objet dans le Gubbin Tool. Apparemment, en lui attribuant un bone, ça empêche la déformation du mesh. Pour une épée, placez un bone dans le manche.

Dans l'onglet Zone, sous-section Music, Le "Repeat time:" est en seconde, et ça correspond à la durée de votre chanson, pas au temps entre chaque chanson. je vous le dis parce qu'en fonction des mesures, il se peut que vous n'entendiez rien (si votre chanson commence avec un silence).

Toujours vérifiez votre fichier "server log.txt" pour trouver les erreurs de script (et autres): \Solstar Games\Realm Crafter 1\Projects\{Votre projet}\Data\Logs\Server Log.txt

#### **MEMORY ACCESS VIOLATION ou MAV:**

- \_ sont soit causée par une modification ou un re-nommage ou une suppression de fichier dans RC (mesh 3d, objet, personnage, images, textures, sons, musique etc...)
- \_ soit par une erreur de script (parfois absente du Server Log).

## 14. Glossaire:

Il est également plus judicieux de vous familiariser avec certains termes:

**RCS** = Version Standard de RealmCrafter

**RCP** = Version Professionnelle de RC

**GE** = Game Engine, c'est l'éditeur de projet. C'est à partir de ce dernier que tout se crée et s'intègre dans votre projet (modèles 3D, maps, textures, sons etc..). Il est commun aux 2 versions de RealmCrafter et fonctionne quasiment toujours de la même façon, on y charge d'abord ses propres medias avant de commencer à créer/modifier.

**Bones / Joints** = Articulation d'un squelette nécessaire à l'animation et à l'intégration de ce dernier dans RC. RC nécessite **obligatoirement** un bone nommé **HEAD** pour la tête

**Gubbins** = Une fois renommés, ces bones permettent d'accueillir des Gubbins, ceux sont par exemple des parties d'armures ou des armes que l'on ira fixer sur un bone en particulier (épaulettes, casques sur la tête, épée à la main etc...)

**RCTE** = RealmCrafter Terrain Editor, l'éditeur de terrain fournit gratuitement avec chacune des versions de RC et qui permet de créer des maps.

**MAV** = erreur qui survient quand vous avez modifié/renommé/supprimé quelque chose que vous n'auriez pas du dans RC ou quand il y a une erreur dans un de vos scripts.

Formats d'exportation/importation:

Pour importer <u>un modèle 3D</u> => en <u>.B3D (idéal)</u> ou en .X (ou en .3DS Sans Animation)

Pour importer une texture => en .DDS (idéal) ou en .PNG (ou en TGA, JPG, BMP Déconseillés)

Pour importer un son/musique => en .OGG (idéal), en .mp3 (ou en .WAV Déconseillé)

## 15. Petit comparatif des 2 versions de RealmCrafter:

	RC STANDARD	RC PROFESSIONNEL
Version stable	1.26 (bêta 1.30 buggée)	2.53
Prix	\$100 (~70€)	\$165 <del>(\$300)</del> (~116€)
Langue	Anglais	Anglais
Moteur 3D	DirectX 7	DirectX 9
Configuration requise	Pentium4 1.3Ghz	Core2 Duo ~3Ghz
	512mo	2 Go
	Geforce 5500 / Radeon 9800	Geforce 8000 / Radeon HD 3000 (Shader model 2.0)
Serveur	Codé en BlitzBasic, script en Brisk BVM	Recodé en C++, script en C#
Estimation accueil	Environ 100 personnes (serveur dédié)	+ de 1000 personnes ? (serveur dédié)
Sources	Disponibles gratuitement	Disponibles gratuitement
Bon à savoir	Créer un jeu complet est possible avec la version 1.26, une version bêta 1.30 existe également, mais elle est buggée.	RCP nécessite plus de connaissance en programmation, le langage de script étant passé en C#.
	Les sources sont disponibles et permettent de tout éditer mais il faut	Dernières versions buggées mais les sources sont désormais disponibles.
	posséder les logiciels BlitzBasic pour modifier le client (~100€) et BlitzPlus	Plus d'équipes aux commandes de RCP
	(~100€) pour modifier le serveur.	
	Plus d'équipes aux commandes de RCS, mais la communauté continue d'améliorer le programme.	

Voila qui devrait vous aider à débuter, si vous avez d'autres soucis, n'hésitez pas à demander de l'aide sur le forum.

Bon courage!