

LONG FORGOTTEN EARTH

PROJET

THE L.F.E PROJECT
TECHNICAL DATA

SOMMAIRE

1	FILE FORMATS.....	3
1.1	STATIC FILE FORMATS	3
1.1.1	Files level.xml.....	3
1.1.2	File Scene_xxx.xml	3
1.1.3	File Room_xxx.cfg.....	4
1.1.4	File Door_xxx.cfg	4
1.1.5	File Robot_xxx.cfg.....	4
1.1.6	File Animated_xxx.cfg	5
1.1.7	File button_xxx.cfg	5
1.1.8	File computer_xxx.cfg	5
1.1.9	File player.cfg.....	5
1.1.10	Waypoint Line files	5
1.1.11	File interface.xml	5
1.2	DYNAMIC FILE FORMATS	6
1.2.1	File Logbook_xxx.xml.....	6
1.2.2	File TheShip.xml	6
1.2.3	File player.xml	6
1.2.4	File Body_xxx.xml	6
1.2.5	File Hibernation.xml.....	7
1.2.6	File Rooms.xml	7
1.2.7	File Doors.xml	7
1.2.8	File computer_X.xml.....	7
1.3	CELESTIA FILES	7
1.4	MATERIALS SCRIPTS	7
1.5	GRADIENT FILES XXX.SVG	8
2	GAMES UNITS	9
2.1	WORLD UNITS	9
2.2	GAME SPEEDS	9
3	THE WEB SERVEUR	10
4	L.F.E. : THE TOOLS WE USE.....	11
4.1	PHOTOSHOP PLUGS-IN.....	11
4.2	CREATION DE TERRAIN	11
4.3	MAPPING DEFINITIONS.....	11
4.4	MISCELANEOUS	11
5	ANNEX: GAME ENGINES.....	12
5.1	GAME ENGINES	12
6	ANNEX: 3D TOOLS.....	13
7	ANNEX: TERRAIN TOOLS	14
8	ANNEX: TEXTURE, NORMAL MAP, SHADERS TOOLS	15
9	ANNEX : 3D OBJECTS LIBRARIES AND WEBSITES	16
10	ANNEX: C++ PROGRAMMING TOOLS	17
11	ANNEX: C++ PROGRAMMING WEBSITES	19
12	STRUCTURE DE DONNÉES	20
12.1	PLAYER.....	20
12.2	THE SHIP	20
12.3	VAISSEAU SPATIAL	20
12.4	PLANETE	20
12.5	ARCHIPELS D'ASTEROÏDES	21
12.6	ETOILE	21
13	ANNEX: FONTS	22
13.1	ANNEX: LES COULEURS	24
13.2	ANNEX: DOTSCENE DTD	24

1 File Formats

1.1 Static File Formats

Static files are located in the /media directory, and his subfolders.

1.1.1 Files level.xml

```
<level number= >
  <scene file="dotscene_file.xml" position="x y z" >comments</scene>
  <map>
    <link nodename="roomName" previous="roomName" />
    This tag allow to link 2 two rooms of the level, by the door(s) of the "nodename" room.
  </map>
</level>
```

1.1.2 File Scene_xxx.xml

Each level is described by an XML file with the standard "dotScene" format, generated by Blender3D.

```
<scene>
  <nodes>
    <node>
      Nodes at this level should all be rooms, (or the camera)
      All other nodes will be ignored.
    <node>
      Nodes at this level are: the door, the lights, and the objects of the room.
    <node>
      Nodes at this level are: the panel.
    </node>
  </node>
</nodes>
</scene>
```

Nodes "userdata"

STRING ANIMATION animation_name
 STRING DESCFILE description_file (*.cfg)
 INT RANGE value
 BOOL MAINLAMP
 STRING CLASS (see the list)

List of available classes:

simple
 cubemap
 envmap
 button
 door
 room
 light - - - ?
 computer/nav
 computer/maint

Rules for the Rooms =

- necessary userdata = STRING CLASS
- necessary userdata = STRING DESCFILE

Rules for the Doors=

- necessary userdata = STRING CLASS
- necessary userdata = STRING DESCFILE
- necessary userdata = STRING ANIMATION
- The door has only one child : the "panel"
- The panel has 2 materials (or more) : **x/door/inside** and **x/door/outside** (where **x** is the level number)

Rules for the Lights =

- Better do not use directional lights.
- Lights are limited to 5 per room
- Only spotlights can cast shadows
- Necessary userdata = INT RANGE.
- Optional userdata = BOOL MAINLAMP

Rules for the Computers =

- necessary userdata = STRING CLASS
- necessary userdata = STRING DESCFILE
- necessary material = computer/screen
- optional material = computer/keyboard
- optional material = computer/visu

Rules for the Lamps =

- necessary material = Lamps
- Note: Lamp is not the Light, but the glass around it.

Rules for the Animated objects =

- necessary userdata = STRING ANIMATION
- Optional userdata = STRING DESCFILE

Filenames Rules:

Scene_xxx.material
 Scene_xxx.xml
 Mesh= xxx.mesh
 Skeleton = xxx.skeleton

1.1.3 File Room_xxx.cfg

SoundFile= sound file with relative path (ie: ../media/sounds/beep.wav)
 HasWindows = "true | false"
 Panel= the material name for the panel (ie: Doorpanel/navroom)
 UniqName= not implemented yet
 LongName= not implemented yet

1.1.4 File Door_xxx.cfg

elevatorDoor= "true | false"
 Soundfile= sound file with relative path (ie: ../media/sounds/beep.wav)

1.1.5 File Robot_xxx.cfg

Meshfile=
 InterfaceFile=
 HelloAnimation=
 HelloSoundFile=
 ActivationAnimation=
 ActivationSoundFile=
 ShutdownSoundFile=
 EngineAnimation=
 EngineSoundFile=

```

Work1Animation=
Work1SoundFile=
NoUseAnimation=
Altitude=
Speed=

```

1.1.6 File Animated_xxx.cfg

```

Animation_type = "direct | reverse | pingpong "
Animation_loop = "true | false"
Animation_trigger = "OnClic | Always"
Focus_line="the text that will appear when the object get the focus"
Cast_shadow="true|false"

```

1.1.7 File button_xxx.cfg

```

meshFile=
IdleMaterial=
ActiveMaterial=
Action= this name will be used in the code button.cpp
Broken=true|false

```

1.1.8 File computer_xxx.cfg

```

Name =
InterfaceFile=
HelloSoundfile=
SparkSoundFile=
OperatingSoundFile=

```

1.1.9 File player.cfg

```

Name=
City=

```

1.1.10 Waypoint Line files

Filename is : wp_ship_n.xml

```

<line    number=      level=>
  <waypoint  x= y= z=    action= id= />
</line>

```

1.1.11 File interface.xml

```

<interface>
  <imageset
    file = "filename.imageset"
    name = "imageset name">

```

```

</imageset>

<title> The window title </title>

<mainWin x= y= w= h= alpha= frame="yes/no" background="yes/no" backgroundImage="imagename">
  <image x y z h >[@]name</image>
  <text x y z h alpha id= color= pen= align= background= frame= font= > [ @ ]text</text>
  <listbox x y z h alpha textid= > @choosemenu</listbox>
  <pushbutton x y z h alpha pen= font= call= > label</ pushbutton >
  < slider x y z h alpha range= linear="true/false" />
  < progressbar x y z h alpha > @value </ progressbar >
  < edit x y z h alpha background= pen= />
</mainWin>

<menuX colour=>
  <item id= text= call= submenu=>item_name</item>
</menuX>

</interface>

```

1.2 Dynamic File Formats

The content of dynamic files can change during the game. They are stored in /savedgame.

1.2.1 File Logbook_xxx.xml

```

<logbook      doc="title"      >
  <entry      title="entry title" > the entry text </entry>
</logboog>

```

1.2.2 File TheShip.xml

```

<ship>
  <reserves AirLevel= FoodLevel= />
  <statusE1 activity= temperature= />
  <statusE2 activity= temperature= />
  <environnement ambientTemperature= />
  <alert level= />
</ship>

```

to be completed

1.2.3 File player.xml

```

<player>
  <situation
    level=
    room=
    position "x y z" />
</player>

```

1.2.4 File Body_xxx.xml

```

<body mesh="meshfilename.mesh" >
  <status
    bones_ok=

```

```

        radia_ok=
        necro_ok=
        wound_ok=
        lastchance=
    />
</body>

```

1.2.5 File Hibernation.xml

```

<hibernation>
  <setting Initial_duration= Initial_date= Final_date= Subject= />
  <status broken="truefalse" locked= temperature= phase="operating|idle|starting|lending" />
</hibernation>

```

1.2.6 File Rooms.xml

```

< doors >
  <status name="RoomName" broken="truefalse" etr="EstimatedTimeToRepair" />
</ doors >

```

1.2.7 File Doors.xml

```

< doors >
  <status name="doorName" broken="truefalse" locked="truefalse" etr="EstimatedTimeToRepair" />
</ doors >

```

1.2.8 File computer_X.xml

```

    <status broken = " truefalse " />
    <spareparts listedbyName=" truefalse " >
      <spare qty= item= name = />
    </spareparts>

```

1.3 Celestia Files

Celestia use "Deep Space Catalog" files.

- binay format
- compressed format

Stars.dat = binary file = Coordinates are : Dec/Rad. **can be used by LFE**
Starname_d = stars correspondance name – number

Other files :

*.ssc = **solar system catalog** = can include asteroids, comets, far galaxies, clouds, spaceship, etc.

*.stc = **star catalog**

Stars.txt = stars positions (polar coodinates and relative magnitude)

1.4 Materials scripts

<< French doc to include >>

1.5 Gradient Files xxx.svg

Gradient files use the SVG format (Scalable Vector Graphics).

```
<svg>
  <defs>
    <lineargradient>
      <stop offset="value 0..1" style="colorvalue" />
    </lineargradient>
  </defs>
  <rect width= height= rx= ry= x= y= style="gradientname" />
</svg>
```

Syntax of the "style" attribute = TBC

2 Games Units

2.1 world units

1 wu = 0,1 m inside the spaceship. (to be adjusted)
 1 wu = 10 m in the space outside the ship.
 1 wu = ??? on the surface of planets.

2.2 Game speeds

Where	Engine	Speed	Unit	Distance	Average time for a standard action	Comments / Engine Mode
Very long distance	Plasma	0 à 1	C			Hibernation
Long distance	Jump	Instantané				Crystal. Need to calculate the good position.
Solar system	Plasma	0 à 1	C		Go from one planet to another in 1 to 10 minutes.	No Fuel needed. (Auto-refuel)
	Booster	1 à 3	C			
Orbiting	plasma	1.000.000	Km/h		Go into orbit in 10 minutes.	Planetary Mode
Landing on a planet					Land on a planet in 3 minutes	
Flying over a planet (spaceship)						
Walking on a planet		5 km/h				
Travelling on a planet (vehicle)		10 km/h				
Flying on a planet (vehicle)		50 km/h				

3 The Web Serveur

Update of the Galaxy Map (FTP or SQL ?)

Update of the Explorer's Encyclopedia

New planets by Explorers, or Dev Team (1 file per solar system). XML Format.

Explorer's Forum

Message to Dev. Team.

Patches:

Application updates

Meshes updates

Adventure scripts updates

When arriving in a solar system, the program check the file:

- locally on the PC
- on the web server
- else : create the file and upload it on the webserver.

4 L.F.E. : The tools we use

Fond étoilé / Espace	: universe3D / gliterrado
Nébuleuses de gaz	: Gliterrado (mieux que Universe)
Textures Planètes	: predefined pictures (Internet / books / lunarcell) + libnoise
Textures Etoiles	: solarcell / iris (?)
Fumées et nuages	: ?
Terrain heightmap	: lunarcell ou L3DT?
Terrain Texture	: T2 ou L3DT
Création de textures dynamiques	: texture editor (?)
Moteur Audio	: Fmod
Moteur graphique	: ogre
Moteur physique	: novodex ou openDE ou Newton ou TrueAxis ?
Moteur réseau	: Raknet ou openTNL?
Moteur scripting	: lua ?

4.1 Photoshop Plugs-in

Lunarcell	: Textures / bumpmap / lightmap / pour planètes non gazeuses
Solarcell	: Textures / bumpmap / lightmap / pour étoiles
Alphaworks	: Transformation du Noir en couche alpha (pour les effets de particules)
Night vision	: Effets pour les hologrammes
Gliterrado	: Fonds Espace (étoiles, nébuleuses, etc.). gratuit.
Universe3D	: Fonds Espace (étoiles, nébuleuses, etc.). démo.

4.2 Création de terrain

Map : 257x257 (chargement en 2 secondes).
513x513 (on a plus de finesse dans le décor, mais des effets de clipping et de LOD, et un temps de chargement de 6 secondes).

Texture :
256x256 pour les maps paginées.
1024x1024 pour une bonne finesse.
2048x2048 : bonne qualité, mais commence à ralentir le framerate.

4.3 Mapping definitions

Les différents types de mapping:

- **parallax map** (ou **offset map**) = permet des effets de rugosité, ou des effets de matière (tissus, pierre, etc)
- **Bumpmap** = permet des effets de reliefs. (image en niveau de gris). A été remplacé par les normal maps.
- **Normal map** = permet des effets de reliefs (layer Alpha en niveau de gris), et des informations sur les facettes du relief (les normales) accentuant ce relief (image RGB).
- **Lightmap** = permet de définir des zones sombres ou lumineuses (images en niveau de gris)
- **Heightmap** = image en niveau de gris. Permet de représenter un terrain. Chaque valeur de gris correspond à une altitude.
- **Mipmap** = technique qui consiste à avoir plusieurs images pour texturer un objet, et que l'on choisi en fonction de sa distance (et donc du niveau de détail nécessaire). Utilisé généralement avec des images au format DDS. Voir les outils sur le site de NVidia SDK.
- **SpecularMap** = xxxxx
- Description or reference of Cg scripts

4.4 Miscellaneous

see IWAR2 (sounds – mesh – textures)

Changing texture : see Texturemanager ::loadimage Texturemanager ::loaddrawData

Forum : « changing texture in runtime », « external texture source », « dynamic textures »

Bumpmapping : see Forum : « renderMonkey » « normal map »

5 Annex: Game Engines

5.1 Game Engines

A "Game Engine" includes generally: a 3D graphic engine, a sound engine, a physic engine, and a network engine.

- **Torque Engine**
100 \$
<http://www.garagegames.com>
- **Blitz3d**
100 \$. Blitz3D est un moteur de jeu de type "langage basic", avec quelques éléments de la programmation objet. (Apparemment, le successeur du DarkBasic). Très simple à mettre en œuvre, bien documenté, on arrive rapidement à un résultat. Cependant, les limitations du basic, le rendent impropre à des projets complexes ou volumineux.
<http://www.blitzbasic.com>
- **Cipher Engine**
100 \$
<http://www.cipherengine.com>
- **Crystal Space**
Free.
<http://crystal.sourceforge.net>
- **Nebula Device**
Free.
<http://www.radonlabs.de>
- **KJ-api**
<http://www.kjapi.de>
- **Axiom**
Free. Graphic Engine based on Ogre,. Coding = C# and Microsoft.Net.
<http://www.axiom3d.org>
- **Allegro**
Free.
<http://alleg.sourceforge.net>
- **Yake Engine**
Free. Game Engine composed by Ogre + OpenAL + OpenDE + Lua.
Interesting.
<http://yake.sourceforge.net>

6 Annex: 3D Tools

- **MYRIAD**
Polygon reduction (600 \$)
<http://www.myriadviewer.com/3ds.htm>
- **RATIONAL REDUCER**
Polygon reduction (6.000\$)
<http://www.taylortrade.com/sim/rationalreducer.htm>
- **POLYGON CRUNCHER**
Polygon reduction. Free.
<http://www.mootools.com/plugins/us/polygoncruncher/main.asp>
- **QEM LOSS**
Polygon reduction. Free.
<http://amber.rc.arizona.edu/lw/qemloss.html>
- **3D STUDIO MAX**
3D modeler. Many tutorials available.
<http://www.discreet.com/>
- **BLENDER 3D**
3D Modeler. Free. Complex.
<http://www.blender3d.com/>
- **TATTOO 3D**
Peinture sur mesh. Free.
<http://www.terabit.nildram.co.uk/tattoo/>
- **3D PHOTO BROWSER LIGHT**
3D Browser. Free.
<http://www.mootools.com/plugins/fr/main.asp>
- **GILE[S] : GLOBAL ILLUMINATION EDITOR**
40 €. Utilitaire pour texturer et ombrer des terrains au format 3DS. Grand choix de lumières et de textures.
<http://giles.thegamecreators.com>

7 Annex: Terrain Tools

- **TERRTEXGEN V2**

Le plus simple à utiliser, il y a juste à charger la heightmap et les textures que l'on veut utiliser. Il ne gère que 5 textures (une par niveau d'altitude) mais il utilise des transitions douces entre elles. On peut générer des textures jusqu'en 4096*4096 et il génère aussi les ombres de terrains. A noter qu'il est en évolution continue. Le plus: une Prévisualisation 3d du terrain. Simple, efficace. Gère le JPG. Il manque l'angle des lumières (pas de lumière rasante, par exemple), et les effets d'ombre sont trop faibles.

<http://www.delphigl.de/projects/terrtexgen.html>

- **TEXGEN**

Un peu plus complexe, car il gère plus de textures différentes, mais par contre ne permet pas une transition douce. Il n'est plus en développement continu, donc pas de nouvelle version en vue. Il gère aussi les ombres de terrain. Moyen.

<http://www.insanesoftware.de/FreeTools/index.html>

- **T2**

Le plus récent, il est apparemment le plus puissant, gérant même le slope, (permettant de changer la texture si le terrain est trop accidenté. Très utile pour avoir de la neige appliquée de façon réaliste). Il peut gérer des cartes paginées (tiles). **Utilisé par LFE.**

<http://www.petra.demon.co.uk/Games/texgen.html>

- **TERRAGEN**

Les images qu'il produit sont belles. On peut faire une image que l'on peut ainsi appliquer sans problème sur un maillage. (Pour générer la texture: Dans la fenêtre où l'on voit la preview, il y a un bouton "Camera Settings" en bas au milieu. Après en haut à droite, il faut cocher "orthographic" et cliquer sur "Auto Setup"). Enfin, on peut charger une heightmap existante ou bien en générer une (au format RAW, qui est une sorte de BMP). La version gratuite est limitée à 513x513 pour la heightmap, et à 1280x960 pour la texture (ce qui est insuffisant).

<http://www.planetside.co.uk/terrigen/>

- **CHRONOS**

Gère des scènes entières pour Ogre (fichiers material, mesh, etc). Nécessite visual.Net pour compiler. N'existe pas sous forme de binaire.

[Broken link](#)

- **HME**

Logiciel gratuit. Editeur de heightmap. Moyen.

<http://hme.sourceforge.net/>

- **GEOFRAC2000**

Générateur de terrain réalistes (heightmap et texture). 35\$

<http://www.geofrac2000.com>

- **GEOSCAPE3D**

Générateur de terrain réalistes (heightmap et texture). 65\$

<http://www.geoscape3D.net>

- **L3DT**

Gratuit. Semble très bien. Génère des heighmaps, normalmaps, lightmaps, à partir de procédures fractales. Comprend un visualisateur 3D.

<http://users.tpg.com.au/blake2/l3dt/>

8 Annex: Texture, Normal map, Shaders Tools

- **TEXSYNTH**

Très utile. Permet de convertir une magnifique texture mais qui malheureusement n'est pas utilisable en répétition, en une texture de type « tile ». Note: le calcul de conversion est assez long.

<http://www.bpeers.com/software/textsynth/>

- **DARKTREE**

Génération de textures procédurales. Outil commercial puissant. 400\$. Les textures peuvent ensuite être utilisées par un programme (jeu) grâce à des librairies Temps Réel.

<http://www.darksim.com/html/darktree25.html>

- **NVIDIA MELODY HOME**

Transforme des modèles 3D hi-polys en modèle 3D low-poly avec normal map. Gratuit.

<http://www.nvidia.com>

- **SKYPAINT**

Aide au dessin de ciels. Permet de générer une skybox (6 faces) à partir de quelques photos. 85\$.

<http://www.skypaint.com>

- **UNIVERSE 3D**

Création de décors spatiaux (étoiles planètes, nébuleuses). Existe aussi sous forme de plugin Photoshop. Démo gratuite pour les étoiles.

<http://www.diardsoftware.com>

- **TEXTURE EDITOR**

Bien. Génère des textures procédurales de type métal ou bois, cellules, ou planètes???. Contient une librairie en C++ qui peut être intégrée aux jeux pour la génération de textures en temps-réel. Gratuit.

<http://www.texture-editor.com>

- **IRIS**

Utilitaire de génération de textures répétitives (tiles), par combinaison de fractales, de sinusoïdales, décalage horizontal ou radial, et gradient de couleur. (utilisable pour les étoiles ?)

<http://www.cinegraphics.net/product.php?location=Iris>

9 Annex : 3D objects libraries and websites

Free models and textures	http://3Dup.com
Free plants models	http://3Dplants.com
Sounds	http://www.findsounds.com
Free models	http://www.3dcafe.com
Free models	http://avalon.viewpoint.com
Free models	http://www.turbosquid.com (login sphinkie)
Free models	http://www.the3Dstudio.com
Planets: JHT's Planetary	http://gw.marketingden.com/planets
Planets: Bjorn Jonsson	http://www.mmedia.is/%7Ebji
Nasa Solar System Simulator	http://samadhi.jpl.nasa.gov
Celestia textures	http://www.celestialvision.net

10 Annex: C++ programming Tools

- **OGRE**
Moteur graphique 3D.
<http://www.ogre3d.org>
- **OPCODE**
moteur physique. Gère uniquement les collisions.
<http://www.codercorner.com/Opcode.htm>
- **OPENDE**
Moteur physique. Peut être interfacé avec Ogre grâce aux librairies OGREODE ou OGREREFAPP.
<http://opende.sourceforge.net>
- **TOKAMAK**
Moteur physique. Peut être interfacé avec Ogre grâce à la librairie OGRETOK.
<http://www.tokamakphysics.com>
- **NOVODEX**
Moteur physique. Peut être interfacé avec Ogre grâce à la librairie NOGREDEX.
<http://www.novodex.com>
- **TRUE AXIS**
Moteur physique. Peut être interfacé avec Ogre.
<http://www.trueaxis.com>
- **NEWTON**
Moteur physique. Peut être interfacé avec Ogre grâce à la librairie OGREWTON.
<http://www.physicengine.com>
- **COLDET**
Détection des Collisions. Les meshes doivent être déclarés à l'API triangle par triangle.
<http://photoneffect.com/coldet/>
- **RACKNET**
Moteur réseau. Fiable. Couche réseau de haut niveau. Dédié aux jeux online et aux jeux multiplayer.
Semble simple.
<http://www.rakkarsoft.com>
- **OPENTNL (TORQUE NETWORK LIBRARY)**
Moteur réseau. Gratuit. API en C++. Robuste, sécurisé, facile à utiliser. Couche réseau de haut niveau.
<http://www.opentnl.org>
- **ENET**
Moteur réseau. Gratuit. API légère et simple. Robuste. Couche réseau de bas niveau.
<http://enet.cubik.org>
- **ZOÏD**
Moteur réseau. Réplication automatique des objets (*game objects*) et synchronisation de leur état sur une connexion réseau (sans renvoyer les données de façon récurrente). Couche réseau de haut niveau
Adapté pour les MMORG. API en C++.
<http://www.zoidcom.com>
- **LOOM**
Moteur IA. Complexe. Basé sur le LISP.
<http://www.isi.edu/isd/LOOM>
- **LUA**
Scripting language. Nécessite "LUAbind" et "Boost". Semble OK.
<http://www.lua.org>
- **NERVE BREAK**
Scripting language. Pas très répandu.
<http://nervebreak.sourceforge.net>
- **IRRLLICHT**
Moteur graphique et shader. Gratuit.
<http://irrlicht.sourceforge.net>

- **FMOD**
Librairie audio. Simple. Gère le mp3 et les sons 3D.
<http://www.fmod.org>
- **OPENAL**
Librairie audio. Un peu complexe. Ne gère pas le mp3.
<http://www.openal.org>
- **CRAZY EDDY**
GUI engine. Librairie "Graphic User Interface". Permet de créer des fenêtres, des boutons, des ascenseurs, etc. pour Ogre.
<http://crayzedsGui.sourceforge.net>
- **A* (A-STAR)**
Algorithme de "pathfinding" et de représentation de cartes. Il en existe d'autres.
<http://theory.stanford.edu/~amitp/GameProgramming/>
Une implémentation simple de cet algorithme est **MicroPather**:
<http://www.grinninglizard.com/projects.html>
- **LIB NOISE**
Gratuit. Librairie de génération de cartes, ou de planètes basée sur des algorithmes aléatoires.
<http://libnoise.sourceforge.net>
- **VISUAL C++ 6.0 SP5**
Le Service Pack 5 du Compilateur est nécessaire pour ogre.
<http://www.microsoft.com>

11 Annex: C++ programming websites

- **NOMAD**
<http://developer.nomadph.com>
- **OGRE HINTS & TIPS**
water effects : <http://www.enygmaarts.com/Ogre/Projects/Water/index.htm>
galaxy effects: <http://www.enygmaarts.com/Ogre/Projects/Galaxy/index.htm>
- **FORUM FMOD**
Login sphinx
<http://www.fmod.org/forum/>
- **FORUM OGRE-FR**
Login sphinx
<http://www.ogre-fr.net>
- **FORUM OGRE**
Login sphinkie
<http://www.ogre3d.org/phpBB2/>
- **OGRE: PAGES WIKI**
Login sphinkie
Encyclopédie Wiki, et Tutoriaux pour Ogre.
<http://ogre.digitalsentience.com/wiki/>
- **CE-GUI:**
Forum: sphinkie.

12 Structure de données

12.1 Player

Nom – ville
C.V. (texte)
Age
Nb planètes découvertes
Nom vaisseau
Position (imprécise) – quadran par ex.
Etat corporel (tete – buste – bras – jambes) (OK – irradié – nécrosé – contaminé)

Position précise
Amis / Ennemis
Réputation
Compétence pilotage
Criminel / Légal

12.2 The Ship

Propriétés de "Vaisseau spatial"

12.3 Vaisseau spatial

Nom
owner
Camp
Vecteur Vitesse (%c)
Masse (10 à 60 t)
Accélération
Position
Cargaison
Shield
Hull
Azimut

12.4 Planète

Caractéristiques physiques

Composition atmosphère (3 ou 4 éléments parmi : CO₂ – CO – N₂O – CH₄ – H₂O – O₂ – O₃ – N₂ – NH₃ - SO₂ – H₂SO₄)

- ⇒ couleur air
- ⇒ couleur nuages
- ⇒ couleur fog
- ⇒ densité atmosphère
- ⇒ (selon aussi couleur soleil)
- ⇒ composition des océans
- ⇒ couleur des océans

composition sol (voir tableau)

- ⇒ couleur sol
- ⇒ couleur poussière

rayon

masse

- ⇒ épaisseur atmosphère
- ⇒ présence de cratères
- ⇒ gravité
- ⇒ vitesse de libération

période de révolution (ou solitaire)

p_étoile

période de rotation

- ⇒ vents

nb Lunes / anneaux

⇒ présence vie (bactérie – marine – aérienne – terrestre – végétale – animale- colons)

distance étoile

⇒ température

⇒ présence eau

⇒ présence glace

⇒ présence végétation (jusqu'à quelle altitude)

⇒ présence océans liquides (pas forcément eau)

type de planète

⇒ composition sol Si-O2-X avec X =Al ou Mg ou Na ou Ca

⇒ densité

historique

Nom

Découvert par

Histoire

Commentaires

Villes (noms - position)

Gisements (noms – position – métal - Mine)

Fermes (noms - position)

Aéroports (noms - position)

Camp/race

Population (en millions d'habitants)

Organisation (sans – anarchie – oligarchie – théocratie – démocratie – féodale – nomade – ruines – centralisée – inconnue – liée a un camp)

Activité (sans - commerce – agriculture – mines – R&D – militaire – philosophie – élevage – industrie – technologie – arts – voyages – plaisirs – sports – jeux – pêche – chasse – artisanat- psychologie – inconnue)

Océan salé (?)

Caractéristiques de terrain

Coefficient de planitude

<= (selon atmosphère, vitesse rotation, vents)

Coefficient de régularité

⇒ îles

ressources

⇒ minerais (où, lequel, combien)

⇒ végétal

niveau océans

⇒

12.5 Archipels d'Astéroïdes

Nom

Position

Couleur dominante

Seed (pour le random)

Nombre d'astéroïdes

12.6 Etoile

Nom

Position

Magnitude absolue (de -8 à +50) (+6 = à peine perceptible)

Spectre (O-B-A-F-G-K-M-R-Z) => couleur / température

p_planets (max 5)

diamètre

masse

type (Trou noir, naine, Série Principale, SP variable, géante, supergéante, nébuleuse)

⇒ masse

⇒ magnitude

Composition (He – H – C – O – N – S)

Age

13 Annex: Fonts

Polices pour les menus et les titres



pour les gros titres des menus



pour les textes

Space Gothic : pour les textes

Space Frigate: lisible. Pour les petits textes.

RingBearer : pour le titre "LFE"



non (pas de minuscules)



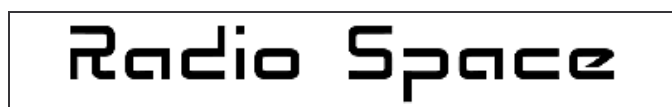
BN machine. Trop serré.

Pour les interfaces, et les computers:

Astro

Space Gothic

SFelectrome



pour les inscriptions murales:

UPPER CASE

A	B	C	D	E	F	G	H	I	J	K	L	M
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

LOWER CASE

A	B	C	D	E	F	G	H	I	J	K	L	M
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

DIGITS

1	2	3	4	5	6	7	8	9	0			
---	---	---	---	---	---	---	---	---	---	--	--	--

7thservice (large)

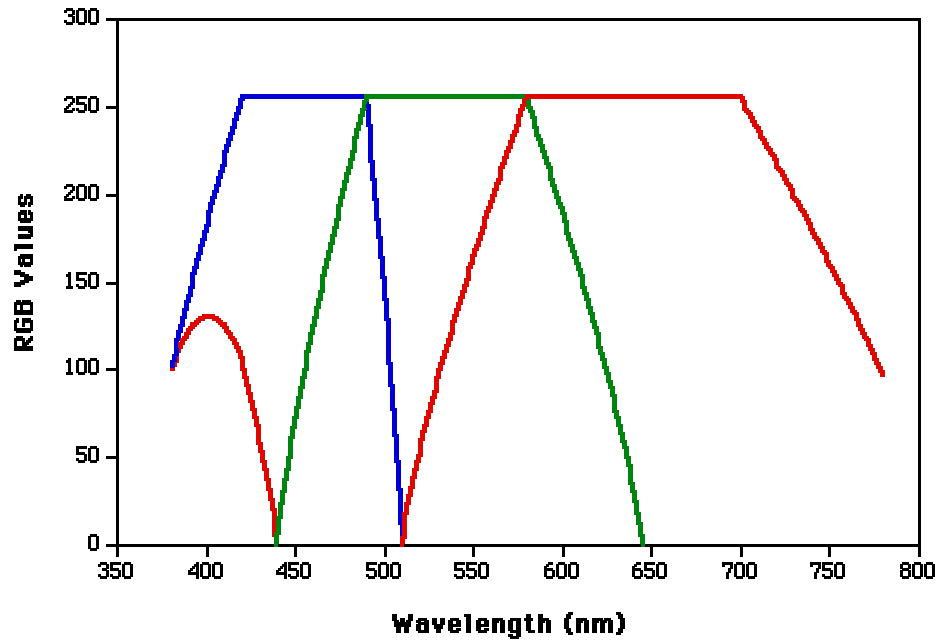
A		B		C		D	
---	--	---	--	---	--	---	--

Airlock Regular

N	U		S	C	H	O	O	L		M	I	L	I	T	I	A
---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	---	---

13.1 Annex: les couleurs

Algorithme permettant de convertir une longueur d'onde en un triplet RGB.



13.2 Annex: DotScene DTD

<<include file >>.