

# Make it real!

Daniel Monedero Tórtola

NLS

Isabel Crespo Pérez

# Índice de contenidos

Palabras clave.....	3
MIR.....	3
BLENDER.....	3
PYTHON.....	3
Introducción.....	4
Idea Principal.....	4
Requerimientos.....	5
Hardware (recomendados):.....	5
Software:.....	5
Motivación.....	6
Desarrollo del Trabajo.....	7
FASE 1.....	7
FASE 2.....	7
FASE 3.....	7
Interfaz Gráfica.....	8
Bibliografía y enlaces.....	10

## **Palabras clave**

MIR, Make, it, real, Blender, 3D, Diseño, Informática, Plugin, Python, GNU

### ***MIR***

Make It Real, entre los diseñadores de imagen sintética la tendencia actual es la representación basada en el fotorrealismo, copiar el mundo que les rodea de la forma más exacta posible, para ello emplean múltiples técnicas para simular comportamientos reales, aplicar propiedades físicas a los objetos, simular el comportamiento de las luces, los movimientos, en muchas ocasiones esto se traduce en una belleza irreal, ya que se elimina de las escenas cualquier imperfección.

### ***BLENDER***

Programa de edición de imágenes sintéticas 3D, una de sus principales características es que es gratuito y de libre distribución. Tiene muchos aspectos que se pueden mejorar pero como todo hay que dejar pasar el tiempo y confiar en todos los profesionales que están detrás del telón mejorándolo día a día. Se ha utilizado en Spiderman 2 y Stuart Little.

### ***PYTHON***

Lenguaje de programación que al igual que 3D interactúa con Blender, dando la posibilidad a los editores de crear objetos a través de la programación, su sintaxis es muy sencilla, y ese a sido el principal motivo para utilizarlo en este trabajo.

# Introducción

## *Idea Principal*

El objetivo de MIR es crear un plugin para Blender que le dote de un sistema sencillo con el que poder crear deformaciones tanto a nivel de malla como de piel.

Para la programación de dicho plugin (incluida la GUI) se hará uso de python y de la API de Blender, el desarrollo se compone de dos partes: el propio blender en el que se probarán los scripts generados y herramientas externas como Vim o Emacs para ayudar al desarrollador a crear scripts en python. La interfaz del plugin se creará también con dichos editores de código ya que las herramientas existentes de desarrollo gráfico de GUIs con python no son compatibles con la API de blender; de esta forma, se creará un gestor de deformaciones, unas deformaciones base y unas clases base con las que cualquiera podrá añadir nuevas deformaciones. Cada deformación estará formada de, por una parte la interfaz y por otro lado las funciones que, recogiendo los datos que el usuario haya introducido mediante la interfaz, genere las deformaciones.

El proyecto se completará con un manual de usuario-memoria (en el que se describirá el proceso de creación, las bases para el uso de Blender y, más en profundidad, el uso de MIR; así como un manual del desarrollador en el que se explicará a nivel de funciones y de API lo necesario para comprender el funcionamiento del plugin y cómo añadir nuevos defectos al mismo.

## Requerimientos

### *Hardware (recomendados):*

- CPU a 300 Mhz.
- 128 MB de memoria ram.
- 20 MB de espacio libre en disco duro.
- Pantalla a 1024 x 768 pixels y 16 bits de color.
- Ratón de 3 botones.
- Tarjeta gráfica compatible con Open GL con 8 MB de memoria.

### *Software:*

- Sistema operativo windows 98, ME, 2000, XP, Mac OS X 10.2, 10.3, 10.4, Linux 2.2.5 i386, Linux 2.3.2 PPC, FreeBSD 5.3 i386, Irix 6.5 mips3 o Solaris 2.8 sparc y Blender 2.3 o superior.

## Motivación

Una de las fases del proceso de creación de imágenes vectoriales es el retoque del objeto para dotarle de realismo, el resultado final de nuestra creación depende mucho de esta fase, a menudo suele resultar tedioso y requiere mucho tiempo.

Dentro de estos retoques podemos citar las deformaciones, estas pueden ser a nivel interno aplicando modificadores a la malla del objeto o externas, trabajando sobre la textura, la capa exterior y visible.

Para nosotros era importante intentar minimizar el tiempo necesario para crear estos efectos, facilitar su edición a través de la creación de un script de fácil utilización, así como reducir el coste de computo (renderizado).

## **Desarrollo del Trabajo**

### ***FASE 1***

El desconocimiento de python y blender, nos hizo dedicar un tiempo considerable a la búsqueda de información y aprendizaje de las herramientas necesarias. Se consultaron varios documentos sobre el lenguaje de programación, también recurrimos a distintos foros.

### ***FASE 2***

Después de entender las limitaciones y las posibilidades reales se adaptó la idea inicial, ya que en un principio el trabajo se pretendía ampliar e incluir en el script funcionalidad para editar pelo. También se diseñó la IGU, y la arquitectura de la aplicación.

### ***FASE 3***

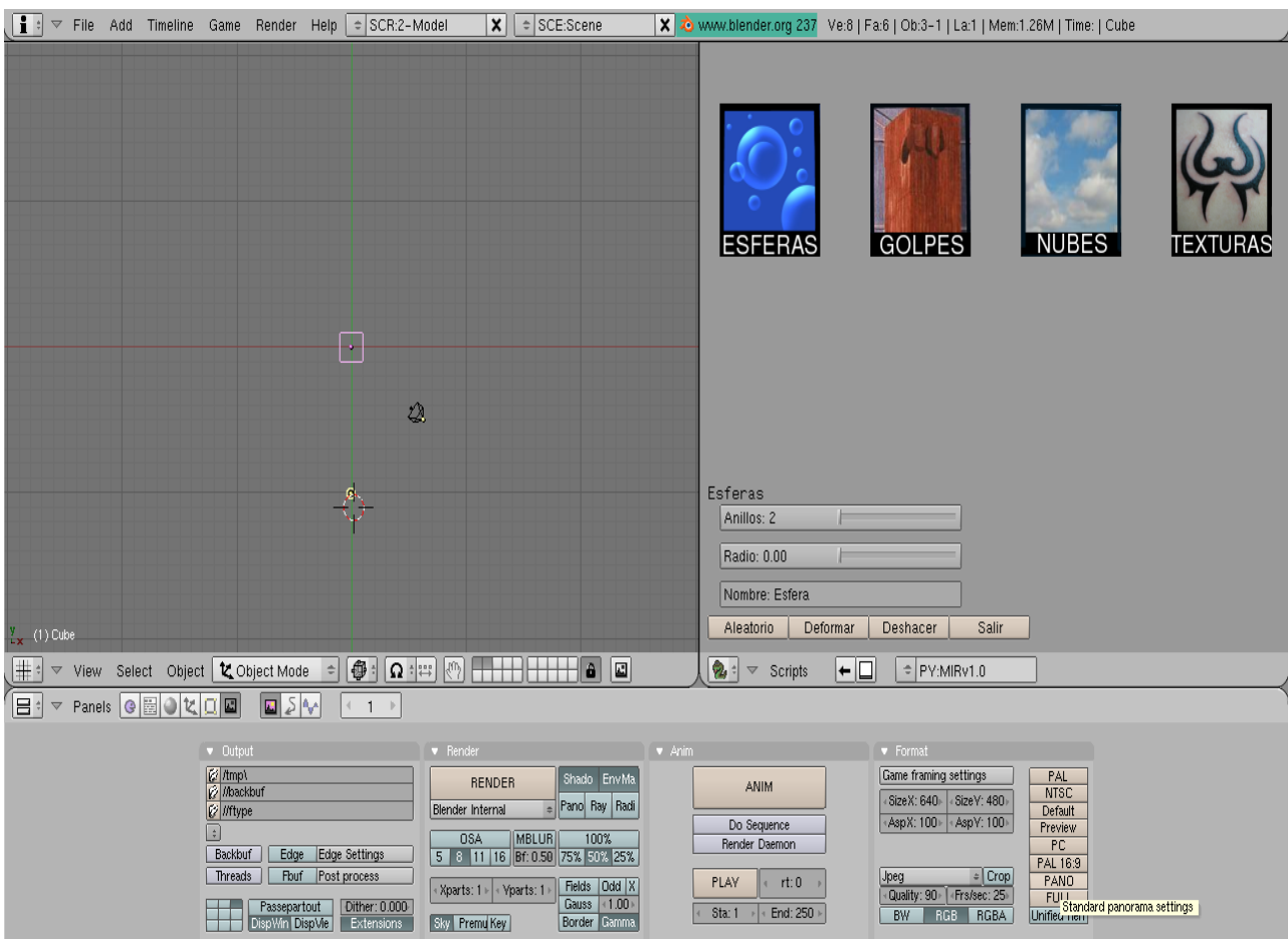
Desarrollo de la interfaz gráfica y el programa.

## Interfaz Gráfica

La idea es crear una interfaz amigable, la curva de aprendizaje para familiarizarse con el sript es mínima para cualquier usuario.

Lo primero que hay que hacer es escoger el tipo de deformación que se quiere aplicar, tenemos la posibilidad de a partir de un número variable de parámetros modificar el resultado final, deshacer cambios antes de aplicar, o dejar que sea el propio programa el que aplique parámetros de forma aleatoria.

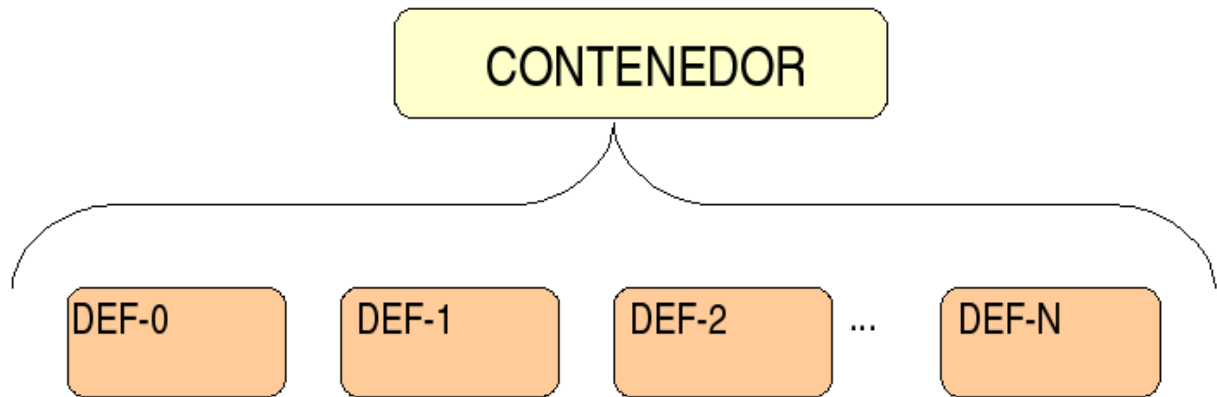
Las deformaciones se aplicaran directamente sobre la malla (esferas, golpes) o sobre la textura (nubes, texturas).





Interfaz gráfica, interactuando con blender, vista principal en la que se pueden observar las deformaciones existentes, así como los grados de libertad o parámetros variables de la deformación esferas.

Arquitectura de la aplicación.



## Bibliografía y enlaces

- The official Blender 2.3 guide
- <http://www.elysiun.com>
- <http://www.blender3d.org/documentation/>
- <http://www.blender.org/modules/documentation/237PythonDoc/index.html>
- <http://blender.guanajuato.net>
- <http://www.programacion.com/tutorial/python>
- <http://www.dedalo-3d.com>