

Making Exploding Objects For CFS2 With Gmax

INTRODUCTION

You will have to have knowledge of the workings of cfs2 scenery, textures, dp files etc. as this is beyond the scope of this tutorial.

Tool box required

You would need to use the FS2002 SDK for GMAX makemdl File version 8.0.0.20403.

You will also need the FSModelExp.dle from FS2002 PRO **Please note this dle from FS2002 pro is pay-ware.**

There is a work around. You would need to use the FS2002 SDK for GMAX <http://www.fsdeveloper.com/forum/downloads.php?do=file&id=95>

and FS2004 SDK for GMAX <http://www.fsdeveloper.com/forum/downloads.php?do=file&id=32>

This is to save a x file so you can compile with makemdl File version 8.0.0.20403.

You will need Raw2sca from <http://webpages.charter.net/ludowr/>

You will need to use a tool called [MDLCommander](#)

Lets set-up your Gamepacks.

This is the way I do it.

FS2004

Run fs2004_sdk_gmax_setup that you just downloaded and install to same folder.

Take the gamepacks and put into your Gmax root folder, place short-cut to desktop, or place it in a short-cut folder on your desktop. Next go to control panel and uninstall and keep the set up launcher for backup. **PLEASE NOTE YOU DO NOT NEED THE GMAX AND PLUGIN CONFIGURATION FILE THAT IS OUTSIDE THE GAMEPACKS FOLDER.**

FS2002

Extract files to same folder as the zip file you downloaded. Make a folder and call it FS2002, inside this folder make a sub folder called plugins and place makemdl into it, then place the Microsoft Word Documents into the FS2002 folder for reference.

Now go to the FS2004 folder and open, then go to the docs folder and open. Take out the fs2002 short-cut and place on your desktop, or place it in a short-cut folder on your desktop. Now place the Gmax and plugin configuration files from FS2004 docs into the FS2002 folder. Place this folder into your Gamepacks folder.

Installing MDLCommander

Before you can use MDLCommander, it needs to be installed of course. This is explained nicely in the documentation, but as it often seems to give problems, we will describe the process here as well. To install MDLCommander you need to take the following steps. Go the plugins sub-folder of the Fs2004 gamepacks folder Rename the existing makemdl.exe to mkmdl.exe

Rename the existing makemdl.parts.xml to mkmdl.parts.xml

Copy the file mdlcommander.exe from the MDLCommader package you downloaded to the plugins folder

Rename mdlcommander.exe to makemdl.exe

This will ensure that GMax now calls MDLCommander instead of MAKEMDL directly when you export your object. Which gives you the option to do some special tricks. You will also notice that you do now get a GUI again during the export (just like the Fs2002 gamepack always did), where you can select the MakeMDL settings.

Configuring MDLCommander

We first need to tell MDLCommander what we want it to do, so therefore you need to make a little configuration file. Unfortunately this configuration has to be set for each object you are going to export, as the file should have the same name as the MDL file you are saving. So assume that you are export a MDL object called myhouse.mdl, in that case you need to create a MDLCommander configuration file called myhouse.par in the folder you are exporting to.

This PAR file is just a plain ASCII text file, so you can easily create it with notepad. When you save the file, be sure that Windows does not give it the extension .par.txt, which can happen if you have hidden file extensions in Windows. To save the X file during the export, you need to put the following text in the PAR file:

```
+savexfile
```

After you exported your object, you should now see that the X file that MakeMDL used has been saved in the folder you exported to. So you are now ready to use it in whatever way you want, how to do that is outside the scope of this article.

Final remark

Just a final remark this article describes how to use MDLCommander with the Fs2004 GMax gamepack. You can also use it with the Fs2002 gamepack, in that case you don't have to rename the file makemdl.parts.xml, as it does not exist in the Fs2002 gamepack. For the FsX gamepack MDLCommander does not work.

Retrieved from [http://www.fsdeveloper.com/wiki/index.php?](http://www.fsdeveloper.com/wiki/index.php?title=X_file_saving_with_MDLCommander)

[title=X_file_saving_with_MDLCommander](http://www.fsdeveloper.com/wiki/index.php?title=X_file_saving_with_MDLCommander)

Lets get started

The os I am using is windows xp Home.

Go to Control Panel.

Go to Appearance and Themes.

Go to folder options,click on File Types.

Go New,create a new file extension **.Par**

Setting up your raw2sca Folder

Make a new folder and name it new raw2sca and place into

it, **BGLC_9,raw2sca,scasm,raw2sca_object.bat**

Rename **raw2sca_object.bat** to
Myobject.bat

MAKE A SCA SUITABLE FOR COMPILING LIBRARYS

I think it is better to use no underscores in the file name,better file names are written in the .sca file.

To do this copy the writing from Header to EndObj.and paste to notepad and save as **Sca Template.txt** as a template,and place in your new raw2sca folder. Save a copy as **Myobject.sca**

**Header(1 N90:00:00.00 S89:60:00.00
E180:00:00.00 W179:60:00.00)**

**LatRange(S89:60:00.00
N90:00:00.00)**

Set(BUF 1024)

Set(areamx 600)

Set(LINBUF 600)

**ObjID(4D9EF5D2 1ACE4BBE
A6E5C777 D9B9858F)**

LibObj(
PWR 0
SIZE xx
SCALE 0.5
TYPE 0
NAME "My_object")

IfVarAnd(:Myobject_Next 0024 0040)

Jump(:Myobject_Main)
:Myobject_Next
Jump32(:Myobject_regular_object)

:Myobject_Main

:Myobject_damaged_object
Include(Myobject-d_0.sca)

:Myobject_regular_object
Include(Myobject_0.sca)

EndObj

Make a .par file

In notepad write

Myobject.par

+savexfile

Save as **Par Template.txt** you will need this file as a template to make other **.par** files.

and then save as **Myobject.par** and place it into the folder you are going to use for saving your X file.

NEXT

In Par Template.txt **change to**

Myobject-d.par

+savexfile

and save as **Myobject-d.par** and place it into the folder you are going to use for saving your X file.

Start up your new FS2004 gmax.

Load model, go to export, in file name save as **Myobject** save as type Flightsim Scenery Object (MDL) press save.

In Model Type select SceneryFS9 and press Start, when done press Exit.

Your **X File** is now saved.

Load damage model, go to export in file name save as **Myobject-d** save as type Flightsim Scenery Object (MDL) press save.

In Model Type select SceneryFS9 and press Start, when done press Exit. Your

X File is now saved

Go to FS2002 Gamepacks plugin folder

Double click makemdl.

Go to options and tick Keep Files and click start.

Go to Input File and Browse to the folder you save your **X** files in and open.

Go to Output File and Browse to the folder you want to save your **.ASM** files in then save.

Go to Model Type and select Scenery. Ignore Heading, Distance, Image complexity.

You will then have to put in any valid Latitude and Longitude, don't worry we will not need this later.

Click start and exit.

Do the above for **both** x files. You should end up with

Myobject_0.asm and **Myobject-d_0.asm**.

You do not need the Myobject.asm, Myobject-d.asm or the two bgls that was compiled.

Compiling your .asm to

This is the way I do it.

Copy your Myobject_0.ASM and the Myobject-d_0.ASM to the new raw2sca folder you set up.

Open your Myobject.sca

Use the GuidMaker to generate a new guid number like this

4D9EF5D21ACE4BBEA6E5C777D9B9858F and then copy and paste this number to ObjID in the sca and put in gaps between the letters and numbers like this 4D9EF5D2 1ACE4BBE A6E5C777 D9B9858F. (EVERY EIGHT) save and close GuidMaker.

EVERY OBJECT SHOULD HAVE ITS OWN UNIQUE GUID

NUMBER.

In Scale put in your scale. (In gmax 1.0 will be 0.5 in the sca.)

In Name put the name you want to use for you Dps.

Open the Myobject_0.asm and find Radius in modelling units = xxx

Copy this number and paste to **size** in your sca file.

**Header(1 N90:00:00.00 S89:60:00.00
E180:00:00.00 W179:60:00.00)**

**LatRange(S89:60:00.00
N90:00:00.00)**

Set(BUF 1024)

Set(areamx 600)

Set(LINBUF 600)

**" WHEN COMPILING LIBRARYS AND
THEY ARE LARGE AND YOU GET AN
LINEBUFFER OVERFLOW YOU CAN
INCREASE THE
BUF,areamx,LINBUF.**

**ObjID(4D9EF5D2 1ACE4BBE
A6E5C777 D9B9858F)**

LibObj(

PWR 0 "WE DONT USE"

SIZE XX

SCALE 0.5

TYPE 0 "WE DONT USE"

NAME "My_object")

IfVarAnd(:Myobject_Next 0024 0040
)

Jump(:Myobject_Main)

:Myobject_Next

Jump32(
:Myobject_regular_object)

:Myobject_Main

:Myobject_damaged_object

Include(Myobject-d_0.sca)

:Myobject_regular_object

Include(Myobject_0.sca)

EndObj

Edit the Myobject.bat so it looks like this

scasm Myobject.sca

Now drag the **Myobject_0.asm** and drop it onto
BGLC_9

Now drag the **Myobject-d_0.asm** and drop it onto
BGLC_9

Now we have two bgl's one named **Myobject_0.bgl** and the
other **Myobject-d_0.bgl**

Now drag the **Myobject_0.bgl** and drop it onto
raw2sca

Now drag the **Myobject-d_0.bgl** and drop it onto
raw2sca

Double click on **Myobject.bat** and a bgl will be compiled called
Myobject.bgl

This is the bgl for your mission builder all you have to do now is make the dp file
Add to mdlnames, Add to descrip.dat, and info screenshots.

Lets make a library

Follow tutorial as above and follow the sca format below.

**Header(1 N90:00:00.00 S89:60:00.00
E180:00:00.00 W179:60:00.00)**

LatRange(S89:60:00.00
N90:00:00.00)

Set(BUF 1024)
Set(areamx 600)
Set(LINBUF 600)

ObjID(4D9EF5D2 1ACE4BBE
A6E5C777 D9B9858F)

LibObj(
PWR 0
SIZE xx
SCALE 0.5
TYPE 0
NAME "My_object")

IfVarAnd(:Myobject_Next 0024 0040
)

Jump(:Myobject_Main)

:Myobject_Next

```
Jump32(  
:Myobject_regular_object )  
  
:Myobject_Main  
  
:Myobject_damaged_object  
Include( Myobject-d_0.sca )  
  
:Myobject_regular_object  
Include( Myobject_0.sca )
```

```
EndObj  
ObjID( 4D9EF5D2 1ACE4BBE  
A6E5C777 D9B9858F ) "Reguid this  
area only"
```

```
LibObj(  
PWR 0  
SIZE xxx  
SCALE 0.5  
TYPE 0  
NAME "My_object1" )
```

```
IfVarAnd( :My_object1_Next 0024  
0040 )
```

```
Jump( :Myobject1_Main )
```

```
:Myobject1_Next
```

```
Jump32(  
:Myobject1_regular_object )
```

```
:Myobject1_Main
```

```
:Myobject1_damaged_object
```

```
Include( Myobject1-d_0.sca )
```

```
:Myobject1_regular_object
```

```
Include( Myobject1_0.sca )
```

```
EndObj
```

=====

=====

HOPE THIS TUTORIAL HELPS.

Please leave feedback.

Robert John