Popcorn's ini Tutorial

Last update: November 30th 2006.

There are lot's of ways to save information into a file in The Games Factory 2 and Multimedia Fusion 2. This tutorial will teach you the basics of one of them.

The ini object

The ini object lets you write to and read from .ini files. Inifiles are organized text files. They are plain text, but set up in a way that's it's easy for applications such as TGF2/MMF2 to work with it. The good thing about ini files is that they give you very much control over its content. The bad thing is that they are very insecure. Because it is plain text, any person can alter it's information. Therefore, they are no use in applications where the information saved should not be accessed by the users, though unless make sure the file cannot be open. You can do that either by including it inside the exe file, or using an extension to encrypt and decrypt it.

How inifiles are organized

An inifile could look like this:

[Player 1]
Name = Yves
Lives: 3
Score = 6540

[Player 2]
Name = Francois
Lives: 2
Score = 8120

The text inside the brackets [] is the name of the group.
The group then contains items (Name, Lives, Score) which then again have values (Yves, 3 6540).

This file has 2 groups. Both groups have the same items, but have different values.

CONTENT

1.1 How to place
1.2 How to save a value from a counter to an **ini** file
1.3 How to save several values into the same group
1.4 How to load an **ini** value to a counter

1.1 How to place

To place an **ini** object in your game, simply create a new **ini** object from the new object window.
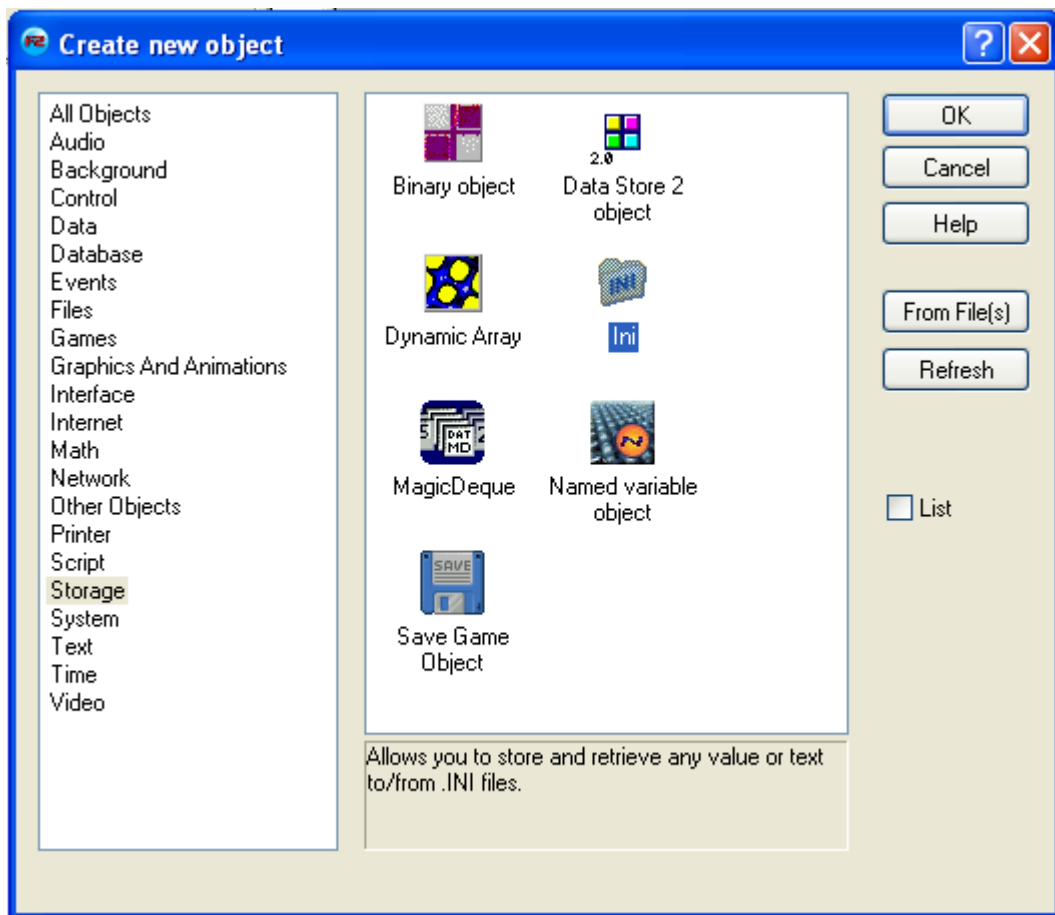
After you've placed it in the frame editor, doubleclick on it and give it a proper name. We want to create a file in the root directory of C called test.ini. Write the whole path for it as you see in the picture. The file test.ini will be made in the C root.
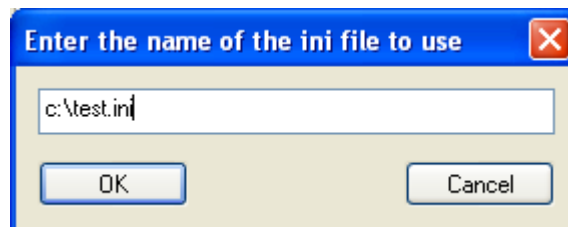

NOTE : The file wouldn't be created until we actually put some values in it from the event editor.

ALSO NOTE: In The Games Factory, which the very old version of this tutorial is referring to, the inifile would be made in the windows directory if no path was set. This is not the case with TGF2 / MMF2. If you do not write a path, but just the name of the file, the file will be created in the same folder that the game is running from. Since we haven't built the application into an exe, all files created at runtime will be created in a temporary folder. Therefore we just set a defined path for now.

1.2 How to save a value

First, you have to tell the program what **group** it shall read from / write to.

Make this event:
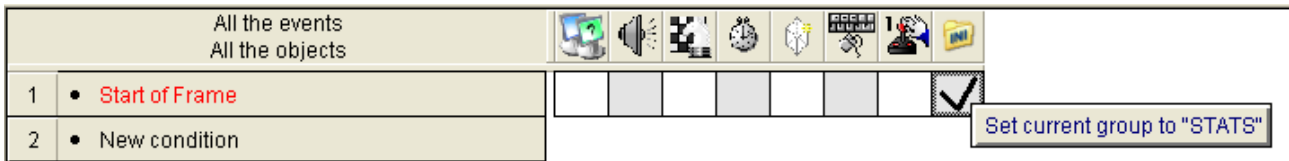
When you run the frame, the program sets the group to STATS. If this group is not allready in the **ini** file, it will make it. But not yet, we have to insert a value first, or MMF2 will not bother to create the file.

We will save a value. First, make a counter an give it a value. I give it the value 5.
Now, add an **action** to the **event** you allready created, so we can tell which item the value shall belong to:
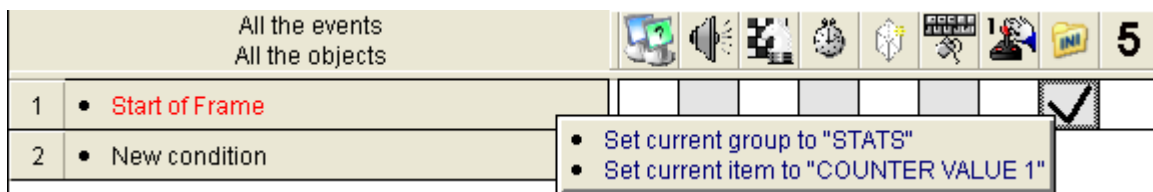
You can call the item everything you want. I called it "COUNTER VALUE 1".

Now we will save the value of the counter into the **ini** file.
Right click under the **ini** icon and select **set value**.
A new windows pops up. Here you can enter a value, but we wants to get the counter's value, so click the **edit** button, and then r**etrieve data from an object**. Now right click the Counter object and select c**urrent value**. Press **OK**.

Now the value of the counter will be saved into the **ini** file.
The whole e**vent** will looks like this.
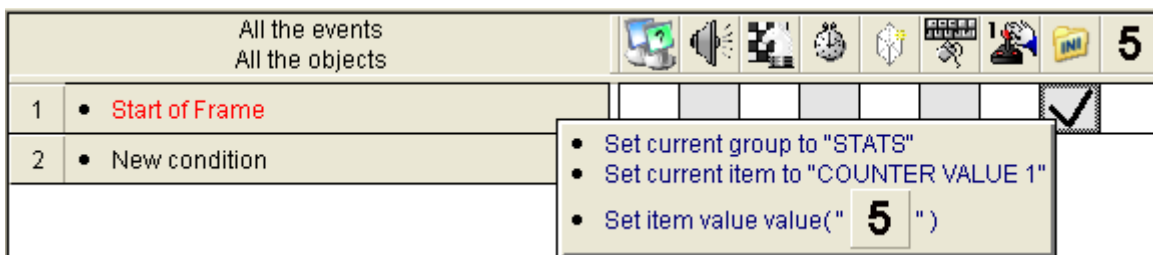
Run the level. Of course you wouldn't actually see anything, but still something have happend. If you followed the steps correctly, you should now see a file called test.ini in the c: directory. It should look like this:
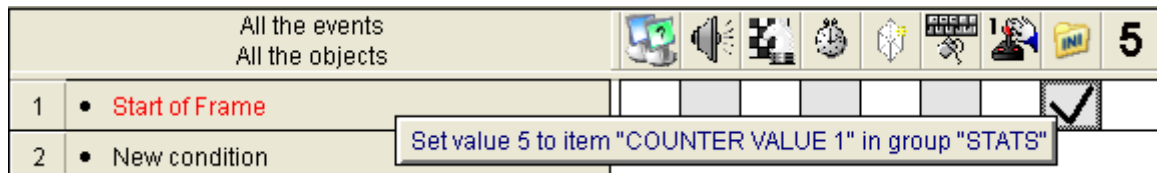
[STATS]
COUNTER VALUE 1=5

If this is the case, congratulations, you can now create an ini file and give it a group, an item and a value! If you can't find the file, you probably did something wrong earlier in the process.

You could also do this all in one step, like this:
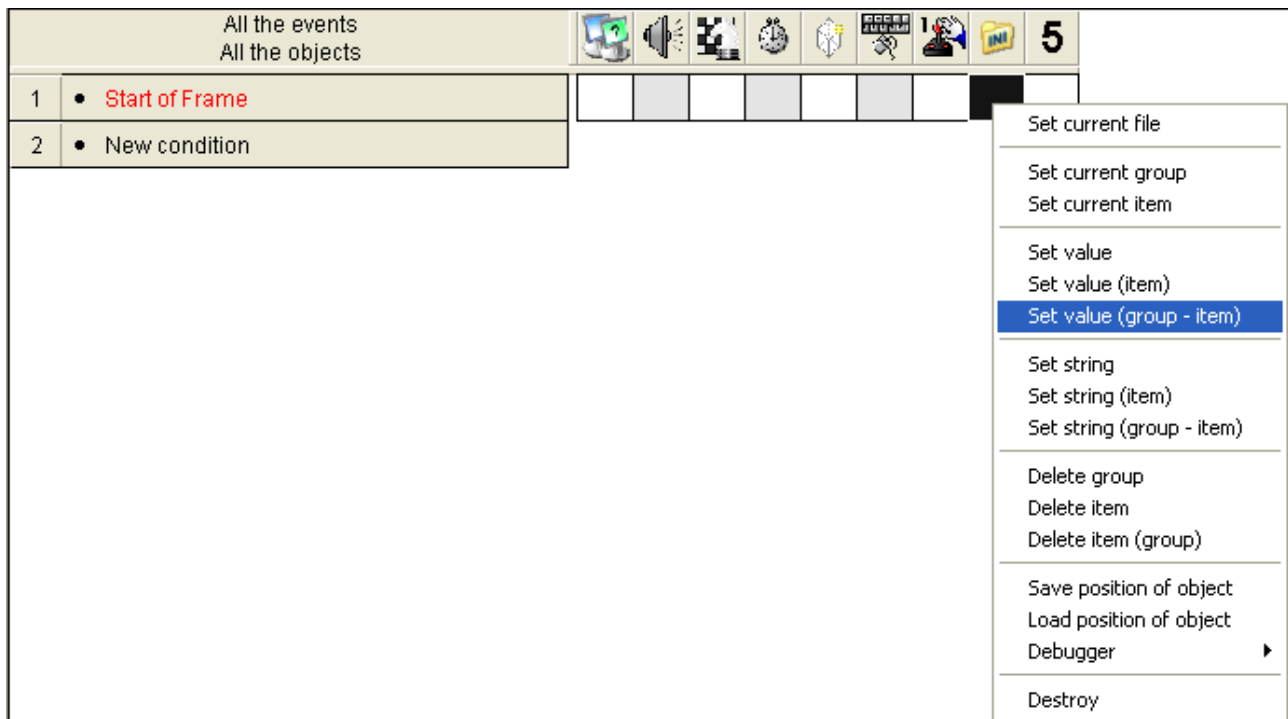
This way you can set the name of the group and item, and set the value, in one action.
It should look like this:

The inifile will look exactly the same. But this way the application will not keep the group or item open for later use, and you will have to open them again.

1.3 How to save several values into the same group

Now we will add one more value to the ini file. We will still use the same counter.
We also want to save it into the same **group**. But we want to do this when the player press the **spacebar** button.
So, make this **event:**
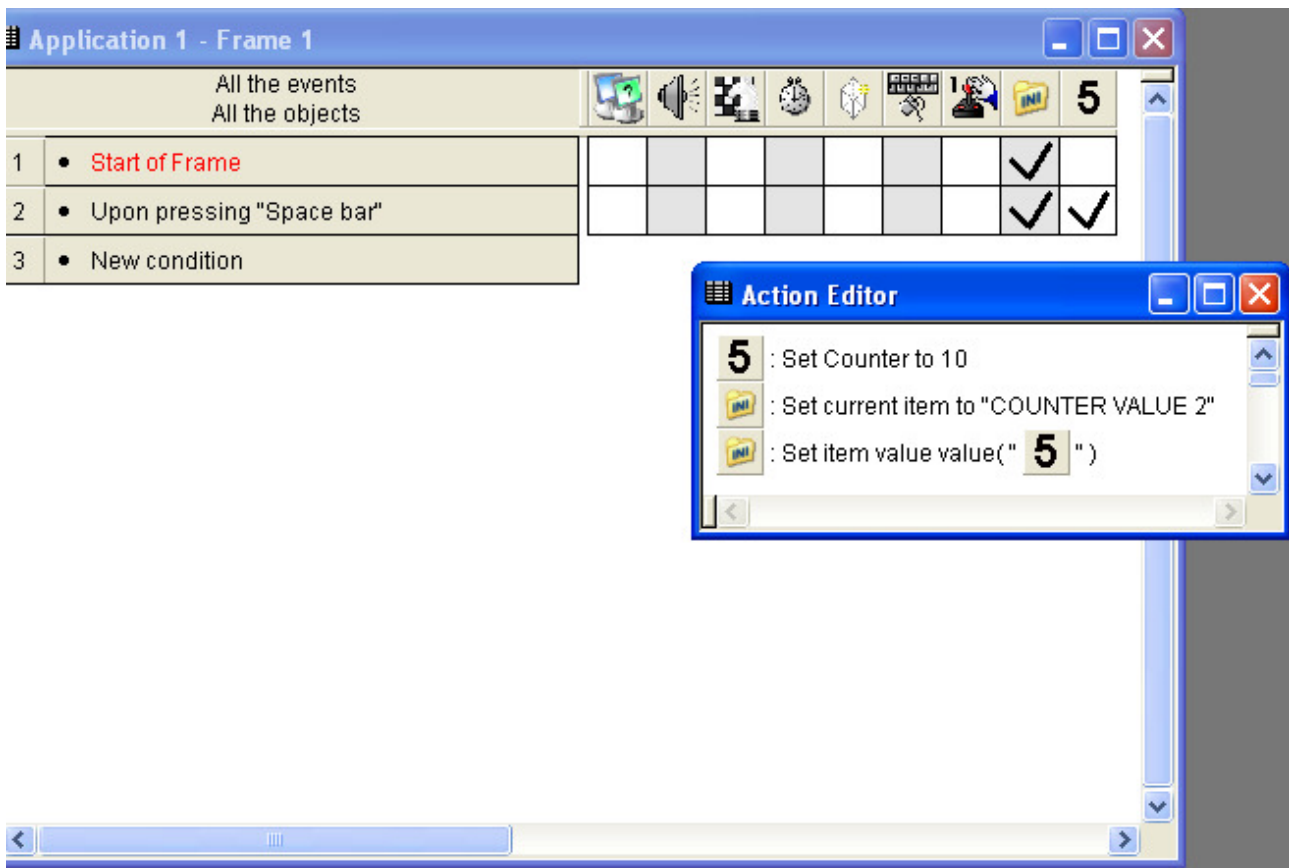
Note that the symbol 5 is the Counter we're using. It just happens to be shown as the number 5 because that's it's inital value.

We didn't name the group, because that is allready set. But we had to change the item since we wanted to save it to another spot of the ini file. If else, the first value would just have been overwritten.

The **ini** file will now look like this:

[STATS]
COUNTER VALUE 1=5
COUNTER VALUE 2=10

1.4 How to load a value from an ini

Ok, so now you know how to save values to an inifile. But that's no use if you can't load them back into MMF2. That's the next thing you will learn.

Make another application, or a new frame in the same application, and place a ini object as described in section 1.1, or just drag it from the first frame. You also need a counter, but this time you don't need to preset the counter's values, as they will be loaded from the **ini** file.

We will now make the event that will make the **ini** ready to read from.

Now, we will load the first **ini** value into "counter 1". To do this, we have to add a another action to the same event. Right click below the counter icon and select "Set counter". A window will pop up. Select **edit** and **Retrieve data from an object**. Now right click the **ini** icon and select **Get value**. Press **OK.**

It should look like this:
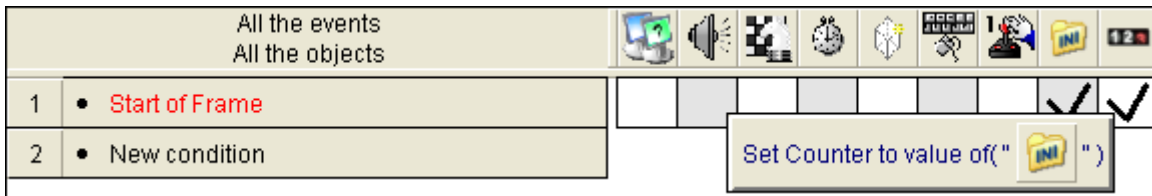
Run the frame and you will see that the counter will change its value to 5.
To **load** the other value we want to have another counter. So make one. Then you have to add a new **action** to the event:
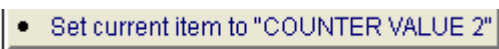
The **ini** file will now read from this item. So do the same for the other counter as you did for the first one.

Test the level and you will see that the first counter is 5 and the other one is 10.
Here are the events in the right order:

The order of the actions are extremely important, and you shall therefore be very careful when trying to copy such actoins. Often you have to type all in manually and can't just copy the them.

For any questions or comments regarding this tutorial please don't hesitate to e-mail me at ilikepopcorn@hotmail.com or kjetil.kjetil@gmail.com.