

Changing the default application configuration file for orx/Scroll

The format for bootstrapping the starting config file is slightly different to a regular Orx application as [shown here](#).

When an orx/Scroll game or application starts, the matching config data file is expected to reside in the same folder as the executable, and to have the same name as the executable.

To change this, add a Bootstrap prototype to your Scroll header:

```
class MyGame : public Scroll<MyGame>
{
private:
    virtual orxSTATUS Bootstrap() const;
    //Other function prototypes here
};
```

Note the use of const above.

In your Scroll class, simply drop in the Bootstrap function:

```
orxSTATUS MyGame::Bootstrap() const
{
    // Add "../data/config" to the list of locations that config files can
    be loaded from
    orxResource_AddStorage(orxCONFIG_KZ_RESOURCE_GROUP, "../data/config",
orxFALSE);

    // Loads a specific config file
    orxConfig_Load("my-game-config.ini");

    return orxSTATUS_FAILURE;
}
```

Scroll will automatically pick up the Bootstrap function if it has been provided.

The return status is set to `orxSTATUS_FAILURE`. This effectively stops Orx from continuing to look for a default config, after the Bootstrap function has completed. If `orxSTATUS_SUCCESS` is returned, Orx will still try and load the default config located next to the executable

Last update: 2017/12/19 en:tutorials:changing_default_config_file_for_scroll https://orx-project.org/wiki/en/tutorials/changing_default_config_file_for_scroll?rev=1513648420
05:53 (2 years ago)

Your Scroll application will now call its root .ini file from some other location relative to the .exe.

From:
<https://orx-project.org/wiki/> - **Orx Learning**

Permanent link:
https://orx-project.org/wiki/en/tutorials/changing_default_config_file_for_scroll?rev=1513648420

Last update: **2017/12/19 05:53 (2 years ago)**

