Obsidian API Tutorial #1 – Setup – All versions

Let's get started

Welcome to the first Obsidian API tutorial! This video version of this tutorial can be found here (https://youtu.be/oUIHq4EnafM).

Prerequisites

To start with, you'll need a basic mod setup. I'm talking main mod class, Client Proxy, Common Proxy, and a single entity (Entity, Model and Render classes). There are plenty of tutorials out there on how to get to this point, so I'm not going to cover it. I suggest having some animations made already, it'll help speed up the process. Also, it helps if you're competent with Java and Eclipse (which is the IDE that I'll be using).

Adding the API

You'll need two downloads – the deobfuscated API jar and the sources jar. Get the latest versions here (http://www.dabigjoe.com/obsidian-suite/downloads/). You need to add the deobfuscated API jar into the libs folder in the root directory of your development environment (the one containing the eclipse folder, the src folder etc.). The libs folder might not exist, just create it if you need to. A quick project refresh in Eclipse (F5) should make the libs folder appear in the Package Explorer, and it should now contain the deobfuscated API jar.

You can see the deobfuscated API in Eclipse, but your software can't yet. You need to tell Eclipse that you've added a new dependency, which you can do by opening up a terminal in the root directory and running

gradlew --refresh-dependencies eclipse
That'll tell Eclipse we're using the API, so it won't complain when we try to start using it.

The First Line of Code

Let's check everything has worked so far. Open up your main mod file, and add the following to your FMLInitializationEvent method:

AnimationNetworkHandler.init();
That should use an Obsidian API import (Ctrl+Shift+O to quickly add imports). The animation network handler automatically sorts out animation packets for client/server stuff. If you can do this without getting any errors, you've successfully added the API, well done.

The Obsidian Event Handler

There's a final thing we need in the main mod file, and that's the event handler for animations. Once you've registered this you can forget about it – it works in the background. I like having a separate method for registering my event handlers:

@EventHandler
public void load(FMLInitializationEvent event) {
    MinecraftForge.EVENT_BUS.register(new ObsidianEventHandler());
}
If you already have a method like this then great, just add the line in the body rather than the method.

That's the main mod file completely set up! I've included a copy of what mine looks like at the bottom of the tutorial.

API Sources Jar

I suggest watching the video for this bit, however if you prefer text then read on. This part of the tutorial begins here (https://youtu.be/oUIHq4EnafM?t=5m1s).

The keen beans among you might have noticed you can't view the source of the API yet. That's where the sources jar we downloaded earlier comes in. Stick the sources jar in a sources folder in the root directory, as you did with the deobfuscated jar but in sources rather than libs. Right click on the project in the Package Explorer -> Build Path -> Configure Build Path. Select the Libraries tab in the window that opens, and scroll down till your find the API jar. Expand it, select the source attachment (should be none), click edit and then browse to the sources jar in the sources folder. Apply this, do a refresh, and try opening an API source file (Ctrl+Shift+T to open any class file, then type something like ObsidianEventHandler to find an API class).

That's it!

Congratulations, you completed the hardest part of using the API! It's all easy from here on in. Next we're going to look at how we use resources created by the animator.

Edited files

Main mod file

@Mod(modid = "ApiTutorial")
public class ModAPITutorial {

    @Mod.Instance("ApiTutorial")
    public static ModAPITutorial instance;

    @SidedProxy(serverSide = "apiTutorial.CommonProxy", clientSide = "apiTutorial.ClientProxy")

```
    public static CommonProxy proxy;

    @Mod.EventHandler
    public void init(FMLInitializationEvent event)
    {
        instance = this;
        proxy.init();
        AnimationNetworkHandler.init();
    }

    @EventHandler
    public void load(FMLInitializationEvent event) {
        MinecraftForge.EVENT_BUS.register(new ObsidianEventHandler());
    }
}
```

 Edit (http://www.dabigjoe.com/wp-admin/post.php?post=444&action=edit)

LEAVE A REPLY

Logged in as dabigjoe (http://www.dabigjoe.com/wp-admin/profile.php). Log out? (http://www.dabigjoe.com/wp-login.php?
action=logout&redirect_to=http%3A%2F%2Fwww.dabigjoe.com%2Fobsidian-suite%2Ftutorials%2F1-setup%2F&_wpnonce=6e8ffaa668)

Comment

POST COMMENT