

1. Setup & Integration

To use GWorld v2 in your plugin, you must add it as a dependency.

1. Maven Dependency

Add the `core` module from GWorld to your `pom.xml`. Since the API is provided by the server at runtime, we use the `provided` scope.

```
<dependency>
  <groupId>de.gilljan</groupId>
  <artifactId>gworld-core</artifactId>
  <version>2.0.1-SNAPSHOT</version>
  <scope>provided</scope>
</dependency>
```

We use the GitHub Maven Package Registry. To use it, please refer to the GitHub documentation: <https://docs.github.com/en/packages/working-with-a-github-packages-registry/working-with-the-apache-maven-registry#installing-a-package>

2. plugin.yml

To ensure that your plugin can access GWorld, it must be loaded *after* GWorld. To do this, add the entry to your `plugin.yml`:

```
depend: [GWorld]
# Alternatively, if GWorld is optional:
# softdepend: [GWorld]
```

3. Access to the API

The main entry point is the `GWorldAPI` interface. You can obtain the instance using the static method of the main class.

```
import de.gilljan.gworld.GWorld;
import de.gilljan.gworld.api.GWorldAPI;
import de.gilljan.gworld.api.IWorldManager;

public class MyPlugin extends JavaPlugin {

    private IWorldManager worldManager;

    @Override
    public void onEnable() {
        if (Bukkit.getPluginManager().isPluginEnabled("GWorld")) {
            // Retrieve API instance
            GWorldAPI api = GWorld.getInstance();

            // Load the manager for worlds
            this.worldManager = api.getWorldManager();
        }
    }
}
```

Revision #5

Created 2026-02-09 19:01:34 UTC by Gilljan (Jannik)

Updated 2026-02-27 22:34:23 UTC by Gilljan (Jannik)