

Dynmap

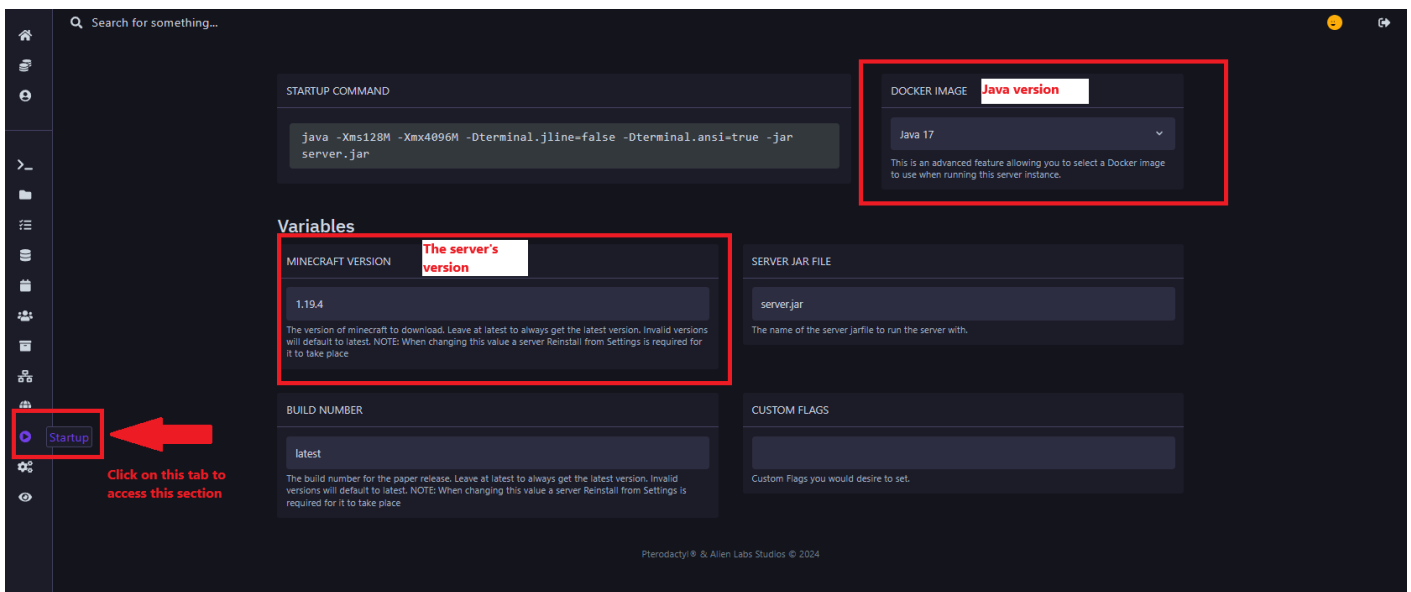
- [Installation and usage](#)
- [Changing Storage Type \(Optimization\)](#)
- [Changing the resolution of the map \(Optimization\)](#)

Installation and usage

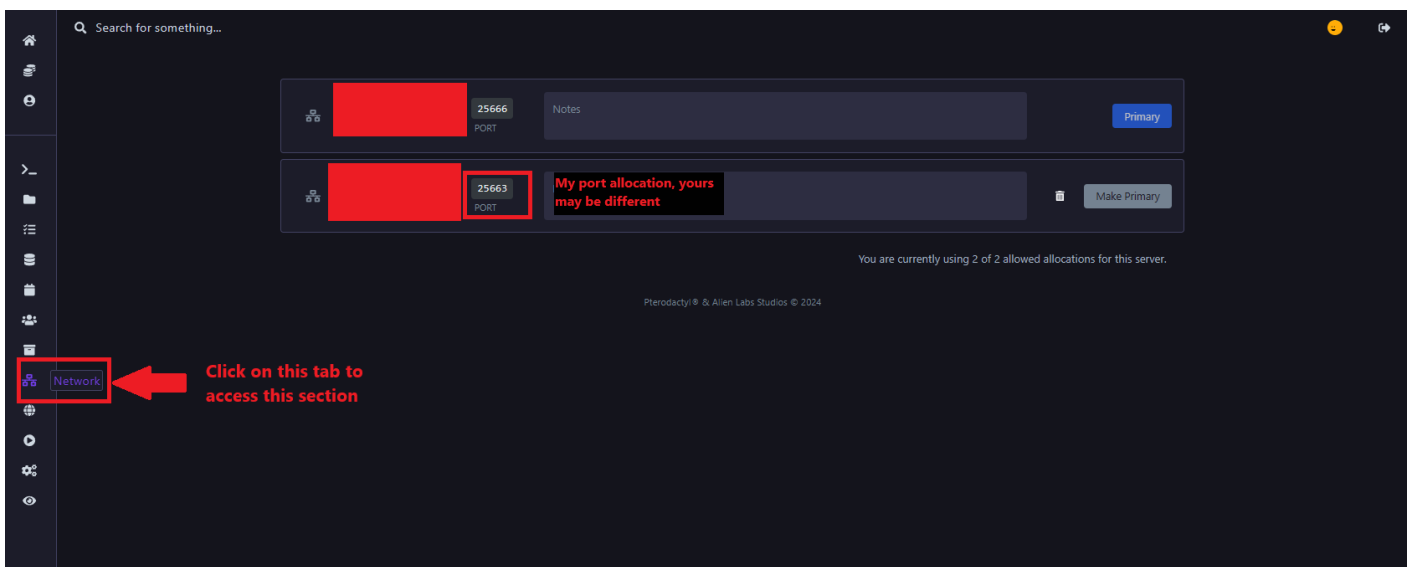
Prerequisites

Spigot/Paper 1.10.2+ (This guide makes use of 1.19.4)

Java 11+ (This guide makes use of Java 17+)



Additional port allocation (In this guide the port is 25663)

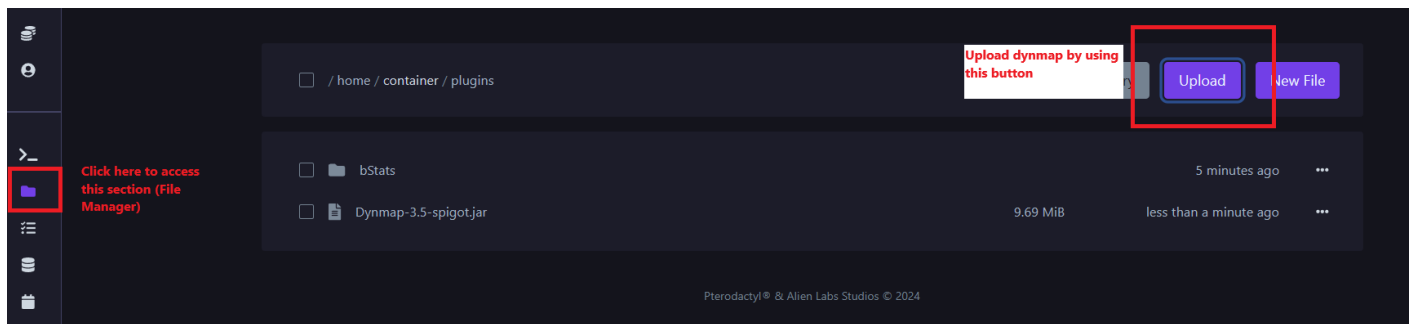


Free storage (a lot)

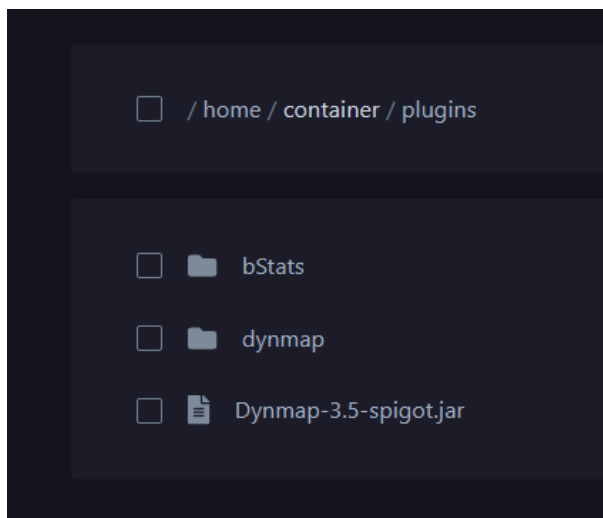
Installation

Download [dynmap](#)

Upload it in the `plugins` folder, from the File Manager.

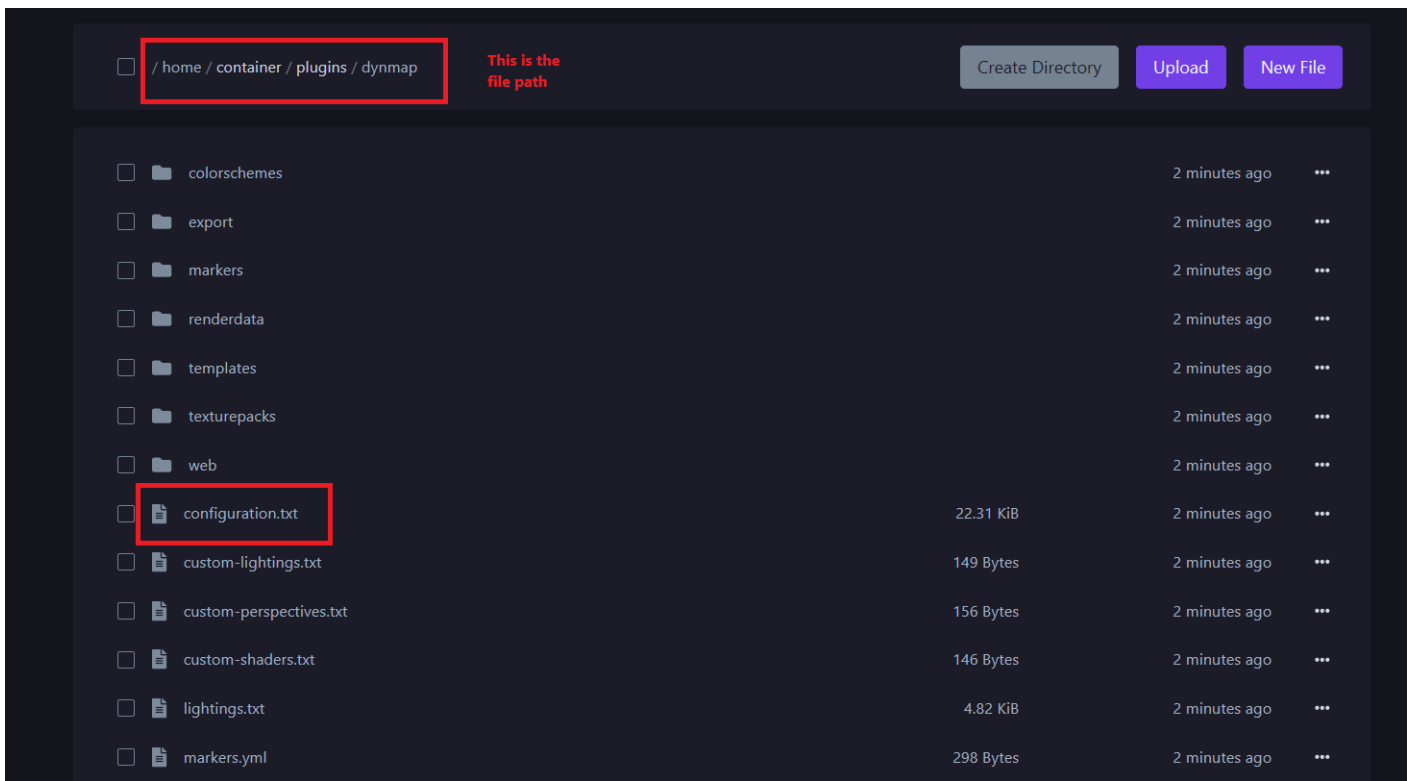


(Re)start the server to load the plugin. You'll know it loaded successfully if it generated a folder named "dynmap"

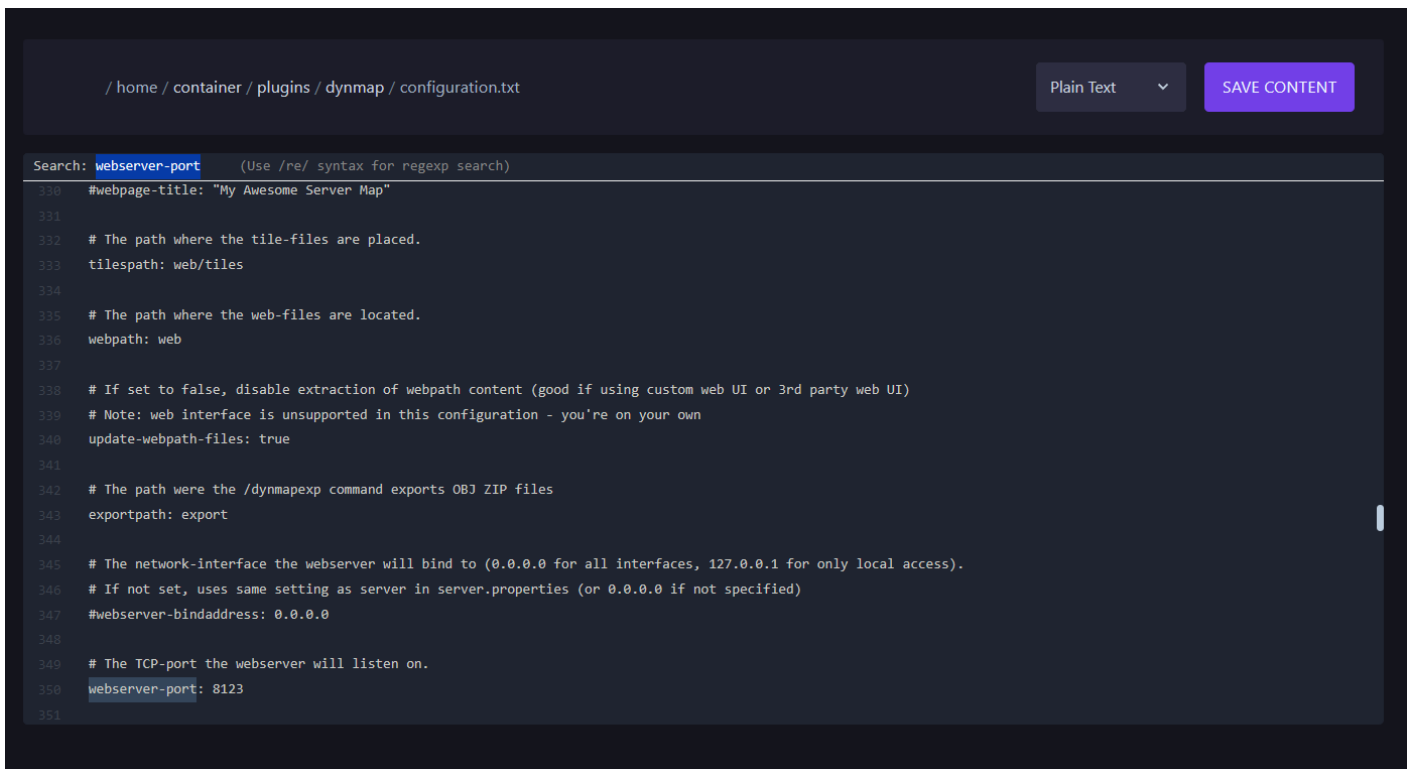


Web Server

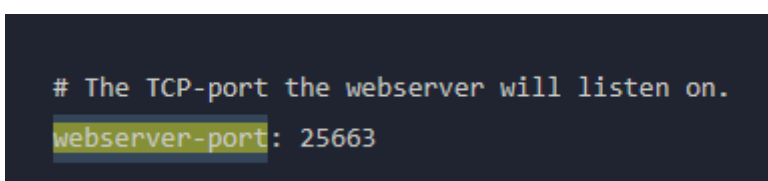
In the `Configuration.txt`, located at `plugins/dynmap/Configuration.txt` :



Find (use CTRL + F shortcut) `webserver-port` field within the file.



Change the port's value to your additional port allocation's value.



Restart the server to apply the changes

Changing Storage Type (Optimization)

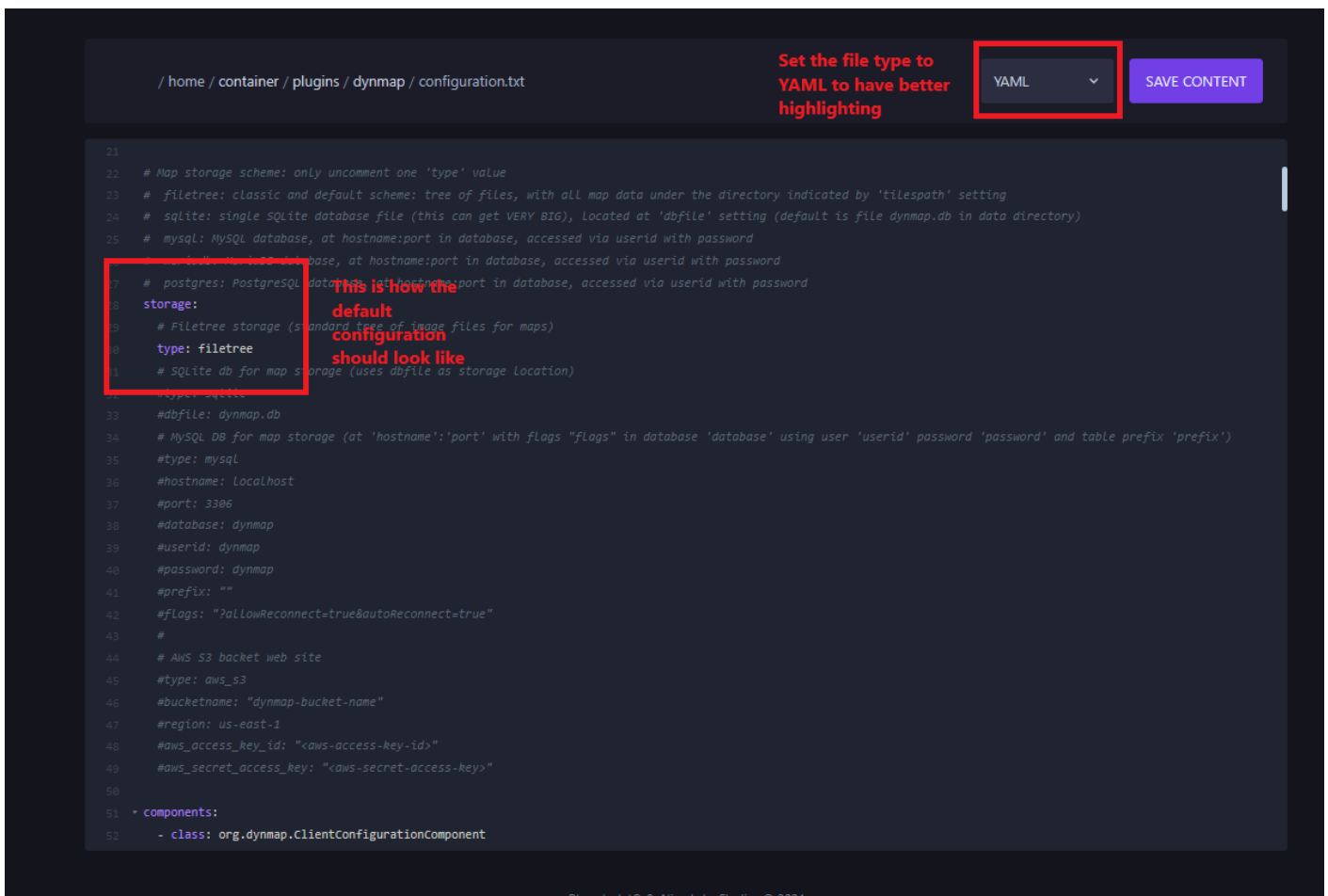
What is this chapter about?

In this chapter I'll help you optimize Dynmap. The default configuration is not the best, it can be improved and that's what we are doing.

Changing Storage Provider

By default the storage type is set to `filetree` (As shown by the following image).

The configuration is located at `plugins/dynmap/configuration.txt`.



SQLite

We would suggest you to switch over to SQLite if you were using the default `filetree` storage.

/ home / container / plugins / dynmap / configuration.txt

YAML

SAVE CONTENT

```
21
22 # Map storage scheme: only uncomment one 'type' value
23 # filetree: classic and default scheme: tree of files, with all map data under the directory indicated by 'tilespath' setting
24 # sqlite: single SQLite database file (this can get VERY BIG), located at 'dbfile' setting (default is file dynmap.db in data directory)
25 # mysql: MySQL database, at hostname:port in database, accessed via userid with password
26 # mariadb: MariaDB database, at hostname:port in database, accessed via userid with password
27 # postgres: PostgreSQL database, at hostname:port in database, accessed via userid with password
28 * storage:
29 # Filetree storage (standard tree of image files for maps)
30 #type: filetree &lt;- DONT FORGET TO COMMENT THIS OUT
31 # SQLite db for map storage (uses dbfile as storage location)
32 type: sqlite
33 dbfile: dynmap.db
34 # MySQL DB for map storage (at 'hostname': 'port' with flags "flags" in database 'database' using user 'userid' password 'password' and table prefix 'prefix')
35 #type: mysql
36 #hostname: localhost
37 #port: 3306
38 #database: dynmap
39 #userid: dynmap
40 #password: dynmap
41 #prefix: ""
42 #flags: "?allowReconnect=true&autoReconnect=true"
43 #
```

Pterodactyl® & Alien Labs Studios © 2025

- Change `type` to `sqlite` (case non-sensitive)
- Make sure `type` and `dbfile` are the only enabled settings (to comment lines out, put a `#` in front of the line.
- `dbfile` allows you to customize the database's name. Doesn't have any performance impact. **The file's extension must be `.db`.**
- When using **Fabric/Forge** you'll need to make use of [Kosmolot's SQLite mod](#).

MySQL

Another option is `MySQL` we don't really suggest you to use this for DynMap due to network latency and other external factors

```

21
22 # Map storage scheme: only uncomment one 'type' value
23 # filetree: classic and default scheme: tree of files, with all map data under the directory indicated by 'tilesPath' setting
24 # sqlite: single SQLite database file (this can get VERY BIG), located at 'dbfile' setting (default is file dynmap.db in data directory)
25 # mysql: MySQL database, at hostname:port in database, accessed via userid with password
26 # mariadb: MariaDB database, at hostname:port in database, accessed via userid with password
27 # postgres: PostgreSQL database, at hostname:port in database, accessed via userid with password
28 + storage:
29 # Filetree storage (standard tree of image files for maps)
30 #type: filetree <- DONT FORGET TO COMMENT THIS OUT
31 # SQLite db for map storage (uses dbfile as storage location)
32 #type: sqlite
33 #dbfile: dynmap.db
34 # MySQL DB for map storage (at 'hostname':port with flags "<flags>" in database 'database' using user 'userid' password 'password' and table prefix 'prefix')
35 type: mysql
36 hostname: localhost
37 port: 3306
38 database: dynmap
39 userid: dynmap
40 password: dynmap
41 prefix: "" # Can add MySQL database for multiple plugins
42 flags: ">allowReconnect=true&autoReconnect=true"
43 #
44 # AWS S3 bucket web site
45 # Do not modify the "flags" if you don't know what you are doing
46 #bucketname: "dynmap-bucket-name"
47 #region: us-east-1
48 #aws_access_key_id: "<aws-access-key-id>"
49 #aws_secret_access_key: "<aws-secret-access-key>"
50
51 + components:
52 - class: org.dynmap.ClientConfigurationComponent

```

- Change `type` to `mysql` (case non-sensitive)
- Change the other highlighted values to the ones generated. [Refer to this guide.](#)
- Now save the file and restart the server to have the new storage type applied.

Changing the resolution of the map (Optimization)

Possible values	Description
<code>vlowres</code>	Uses the HDMaP renderer with view from the SE with the "vlowres" resolution (2 pixels per block edge)
<code>lowres</code>	Uses the HDMaP renderer with view from the SE with the "lowres" resolution (4 pixels per block edge)
<code>hires</code>	Uses the HDMaP renderer with view from the SE with the "hires" resolution (16 pixels per block edge)
<code>low_boost_hi</code>	Uses the HDMaP renderer with view from the SE with the "lowres" resolution (4 pixels per block edge), with boosted tiles rendered at "hires" (16 pixels per block edge)
<code>hi_boost_vhi</code>	Uses the HDMaP renderer with view from the SE with the "hires" resolution (16 pixels per block edge), with resolution boosted tiles rendered at 'vhires' resolution (32 pixels perblock edge)
<code>hi_boost_xhi</code>	Uses the HDMaP renderer with view from the SE with the "hires" resolution (16 pixels per block edge), with resolution boosted tiles rendered at 'xhires' resolution (64 pixels perblock edge)

Brief Explanation

These values allow you to either save storage (**vlowres** saves the most) or have a high quality map (**hi_boost_xhi** provides the best quality/resolution for the map).

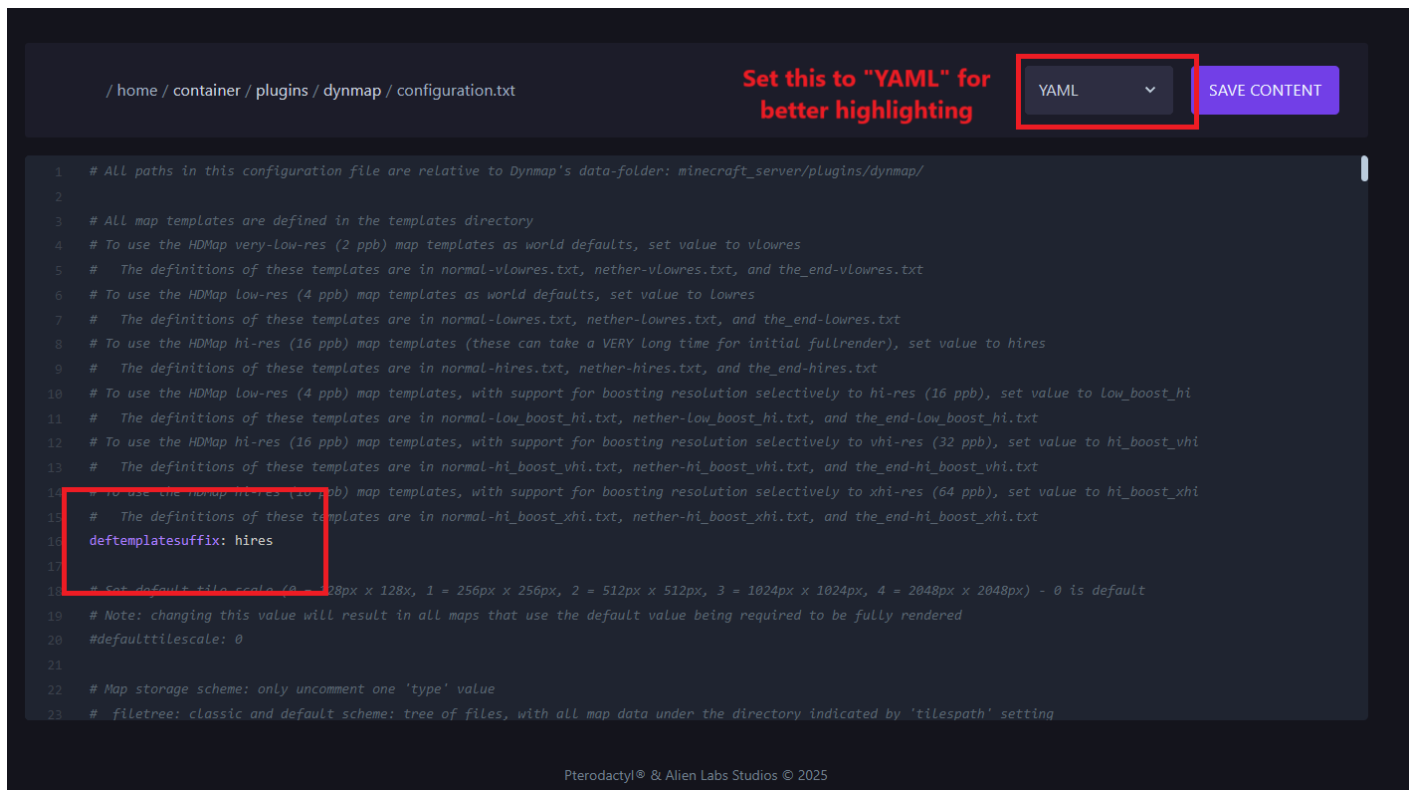
Which option should I choose?

The default value is **hires**. That probably works for most server owners. But for those who lack storage, they may be prompted to pick a "storage saver" option such as **vlowres** or **lowres**.

Additionally High Resolution maps take a lot of time to render (and an exorbitant amount of storage).

Where and how to change the resolution

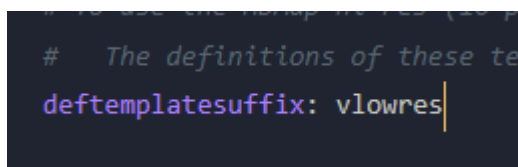
Through the use of the File Manager, reach the following path: `/home/container/plugins/dynmap` (NOTE: This is the path for Spigot/Paper servers. These are all folders), where you'll find a text file named `configuration.txt`. Open it.



Now change `deftemplatesuffix` with one of the values given at the beginning of this article. Save the file with the `Save content` button and restart the server through the console.

Examples

Very Low Resolution:



Low Resolution:

