# Load an existing world safely with MyWorlds

MyWorlds manages Minecraft worlds, travel and world-specific settings. When importing an existing world, the environment and generator matter as much as the folder contents. Loading it under the wrong rules can produce mismatched new terrain.

## Choose one manager for the task

Check the MyWorlds release against the server's exact Minecraft and Java versions. Obtain its required BKCommonLib dependency from the maintainer's official links and use a supported combination. Do not treat the presence of another world manager as proof that importing its configuration will be automatic and risk-free.

Record every plugin currently controlling world loading, portals, spawn and inventories. Plan which one will own each feature. Avoid running two competing managers against the same new world while trying to decide which one works better.

## Back up and identify the source world

Stop the server and take a completed backup of all worlds, player data and manager configuration. Preserve the source world separately before uploading a copy. Note its Minecraft version, environment and any generator plugin or datapack it needs. Keep external database backups aligned where player state depends on them.

Read backup guidance and world upload guidance. Check the release's current world layout before copying files; a newer dimension layout should not be forced into an older tutorial's folder structure.

## Inspect before loading

1. Install the compatible MyWorlds and BKCommonLib files using the plugin installation guide . Start once and inspect the earliest dependency errors.
2. Upload only the copied source data to the appropriate location while the server is stopped. Do not replace the production world.
3. Start and use `/world list` to inspect available worlds.
4. Check `/world listgenerators` if the world depends on a generator. Confirm that the correct plugin is enabled before proceeding.
5. Use the installed build's `/world load` syntax with the correct environment and generator for a first import. A previously known world may retain this metadata; an unknown one needs explicit care.

Commands shown with a slash are for in-game use. Console commands omit it. A load error is a reason to stop and inspect the source details, not to run world-repair or deletion commands against the only copy.

# Verify travel, persistence and inventory behaviour

Visit a safe known location and inspect terrain, spawn and borders. Avoid exploring ungenerated areas until the generator is confirmed. Check `/world info` for the loaded world's settings and test a normal restart to confirm the intended load behaviour persists.

Test with an ordinary player. World permissions, game mode and inventory policy are separate settings; loading a creative world does not automatically make item transfer safe. MyWorlds has inventory features, but enabling or changing those on a populated server requires a separate backed-up test plan. Do not enable a second inventory manager for the same state.

If a world behaves unexpectedly, retain both the original source and current test files. Send support its source version, generator, environment, MyWorlds/BKCommonLib versions and first error. Further reading: managing worlds and the official feature documentation.

Looking for another plugin? Browse the Minecraft plugin guide directory.

---

Revision #2
Created 2026-11-09 08:57:16 UTC by CloudExa Hosting Admin
Updated 2026-11-09 08:57:17 UTC by CloudExa Hosting Admin