

Migration von v1-legacy zu v2

GWorld v2 verfügt über intelligente Migrations-Systeme, die den Übergang von älteren Versionen oder den Wechsel zwischen Speichermedien (Dateien zu Datenbank) so einfach wie möglich gestalten.

Du suchst nach einer Möglichkeit, von YAML zu MySQL zu wechseln? Dann folge bitte diesem Link:
[Wechsel von YAML \(File\) zu MySQL \(DB\)](#)

Migration von GWorld v1 zu v2

Wenn du von der ersten Version von GWorld auf v2 aktualisierst, erkennt das Plugin dies automatisch beim ersten Start.

Funktionsweise

- **Erkennung:** Das Plugin sucht nach einer existierenden `worlds.yml`, in der...
 - das Feld `ConfigVersion` fehlt
 - die Liste `LoadWorlds` existiert
 - die Liste `Worlds` existiert
- **Automatisches Backup:** Bevor Änderungen an den Datenbeständen vorgenommen werden, benennt GWorld die alte Datei in `worlds_legacy_backup.yml` um.
- **Daten-Konvertierung:** Der `LegacyMigrator` liest alle alten Welten-Definitionen aus und überführt sie in das neue, strukturierte **v2-Format**. Fehlende neue Felder werden mit Standardwerten (Defaults) aufgefüllt.
- **Seed-Reparatur:** Da in der v1 keine Seeds abgespeichert wurden, erkennt GWorld v2 fehlende Seeds beim ersten Laden der Welt automatisch und liest den korrekten Seed aus den Welt-Dateien aus und korrigiert den Eintrag.

Hinweis: Während der Migration wird die Konsole dich über den Fortschritt informieren. Sobald die Datei `worlds.yml` eine `ConfigVersion: 1` oder höher enthält, ist der Prozess abgeschlossen. Dies wird ebenfalls in der Konsole ausgegeben.

Nach einer erfolgreichen Migration starten GWorld v2 normal und alle Welten werden geladen

Sicherheitsmaßnahmen & Voraussetzungen

Um eine reibungslose Migration zu gewährleisten, beachte bitte folgende Punkte:

Voraussetzungen (siehe [Einführung](#))

- **Java-Version:** GWorld v2 benötigt zwingend mindestens **Java 21**, da moderne Sprachfeatures verwendet werden.

Checkliste

- ☐ Erstelle vor dem Upgrade ein manuelles Backup deines gesamten Plugin-Ordners.
- ☐ Prüfe nach der Migration mit dem Befehl `/gworlds`, ob alle Welten korrekt gelistet werden.
- ☐ Kontrolliere die Server-Konsole auf Fehlermeldungen während des ersten Starts.

Beispielhafte worlds.yml Konfigurationen

worlds.yml (v1)

worlds.yml (v2.0.1)

#Copyright (c) Gilljan 2020-2022. All rights reserved.

#Plugin written by Gilljan

LoadWorlds:

- world
- world_nether
- world_the_end

Worlds:

world:

generator: 'null'
type: normal
timeCycle: true
time: 6000
weatherCycle: true
weather: sun
pvp: true
mobs: true
animals: true
forcedGamemode: false
defaultGamemode: SURVIVAL
difficulty: NORMAL
randomTickSpeed: 3
announceAdvancements: true

world_nether:

generator: 'null'
type: nether
timeCycle: true
time: 6000
weatherCycle: true
weather: sun
pvp: true
mobs: true
animals: true
forcedGamemode: false
defaultGamemode: SURVIVAL
difficulty: NORMAL
randomTickSpeed: 3
announceAdvancements: true

world_the_end:

generator: 'null'
type: end

ConfigVersion: 1

Worlds:

world:

Environment: NORMAL
Alias: \$bMeine Welt
Type: NORMAL
Seed: 541091381232416907
AllowPvP: true
KeepSpawnInMemory: true
AnimalSpawning: true
DisabledAnimals:
- cow
MonsterSpawning: false
DisabledMonsters:
- spider
WeatherCycle: false
WeatherType: CLEAR
TimeCycle: false
Time: 6000
DefaultGamemode: false
GameMode: ADVENTURE
Difficulty: HARD
LoadOnStartup: true
RandomTickSpeed: 3
AnnounceAdvancements: true

world_nether:

Environment: NETHER
Alias: \$cDer Nether
Type: NORMAL
Seed: 541091381232416907
AllowPvP: true
KeepSpawnInMemory: true
AnimalSpawning: true
DisabledAnimals: []
MonsterSpawning: true
DisabledMonsters: []
WeatherCycle: true
WeatherType: CLEAR
TimeCycle: true
Time: 0
DefaultGamemode: false
GameMode: SURVIVAL

Diese Dokumentation bezieht sich auf GWorld Version 2.0.1.

Revision #8

Created 2026-02-01 10:52:56 UTC by Gilljan (Jannik)

Updated 2026-02-27 15:25:00 UTC by Gilljan (Jannik)