

Fuji Manual

sakurawald

August 30, 2024
Version v1.4.9

Contents

1	Concept	1
1.1	Configuration File	1
1.1.1	Definition	1
1.1.2	Types	1
1.2	Module	2
1.2.1	Definition	2
1.2.2	Properties	2
1.2.3	Module Path	2
1.2.4	How to enable/disable a module	3
1.3	Job	3
1.3.1	Definition	3
1.3.2	Cron Expression as Trigger Rule	3
1.4	Regex	3
1.4.1	Definition	3
1.4.2	Reference	3
2	Permission	4
2.1	Definition	4
2.2	Types	4
2.3	What is the permission system used by fuji?	5
2.3.1	Explanation	5
2.3.2	Set a string permission for a command.	5
2.4	Reference	6
3	Meta	7
3.1	Definition	7
3.2	Example	7
4	Configuration	8
4.1	Main-Control File	8
4.1.1	common	8
4.1.1.1	quartz	8
4.1.1.1.1	logger_level	8
4.1.1.2	backup	8

CONTENTS	2
----------	---

4.1.1.2.1	max_slots	8
4.1.1.2.2	skip	8
4.1.1.3	language	9
4.1.1.3.1	default_language	9
4.2	Module-Control File	10
5	Module	11
5.1	config	12
5.1.1	Purpose	12
5.1.2	Command	12
5.1.2.1	/fuji reload	12
5.2	language	13
5.2.1	Purpose	13
5.2.2	Feature	13
5.2.3	Difference	13
5.3	chat	14
5.3.1	Purpose	14
5.3.2	Feature	14
5.3.3	Command	15
5.3.3.1	/chat	15
5.3.4	Placeholder	15
5.3.5	Configuration	15
5.3.5.1	format	15
5.3.5.1.1	rewrite	15
5.3.5.1.2	mention_player	15
5.3.5.1.2.1	sound	15
5.3.5.1.2.2	volume	15
5.3.5.1.2.3	pitch	15
5.3.5.1.2.4	repeat_count	15
5.3.5.1.2.5	interval_ms	15
5.3.5.1.3	spy	16
5.3.6	Sub-Module	16
5.3.6.1	display	16
5.3.6.1.1	Placeholder	16
5.3.6.1.2	Configuration	16
5.3.6.1.2.1	expiration_duration_s	16
5.3.6.2	history	16
5.3.6.2.1	Configuration	16
5.3.6.2.1.1	buffer_size	16
5.3.7	Reference	16
5.4	placeholder	17
5.4.1	Purpose	17
5.4.2	Placeholder	17
5.5	motd	19
5.5.1	Purpose	19
5.6	nametag	20

5.6.1	Purpose	20
5.6.2	Configuration	20
5.6.2.1	update_cron	20
5.6.2.2	style	20
5.6.2.3	render	20
5.6.3	Example	20
5.6.4	Reference	21
5.7	tab_list	22
5.7.1	Purpose	22
5.7.2	Configuration	22
5.7.2.1	update_cron	22
5.7.2.2	style	22
5.7.3	Sub-Module	22
5.7.3.1	sort	22
5.7.3.1.1	Purpose	22
5.7.3.1.2	How it works	22
5.7.3.1.3	Example	22
5.7.3.1.4	Sub-Module	23
5.7.3.1.4.1	sync_game_profile	23
5.7.3.2	faker	23
5.7.3.2.1	Purpose	23
5.8	tpa	24
5.8.1	Purpose	24
5.8.2	Command	24
5.8.3	Configuration	24
5.8.3.1	timeout	24
5.9	back	25
5.9.1	Purpose	25
5.9.2	Command	25
5.9.3	Configuration	25
5.9.3.1	ignore_distance	25
5.10	home	26
5.10.1	Purpose	26
5.10.2	Command	26
5.10.3	Meta	26
5.11	pvp	27
5.11.1	Purpose	27
5.11.2	Command	27
5.12	afk	28
5.12.1	Purpose	28
5.12.2	Command	28
5.12.3	Configuration	28
5.12.3.1	format	28
5.12.3.2	afk_checker	28
5.12.3.2.1	cron	28
5.12.3.3	afk_event	28

5.12.3.4	effect	28
5.12.3.4.1	invulnerable	28
5.12.3.4.2	targetable	28
5.12.3.4.3	moveable	28
5.13	rtp	29
5.13.1	Purpose	29
5.13.2	Feature	29
5.13.3	Command	29
5.13.3.1	/rtp	29
5.13.3.1.1	Argument	29
5.13.4	Configuration	29
5.13.4.1	setup	29
5.14	works	30
5.14.1	Purpose	30
5.14.2	Concept	30
5.14.3	Command	30
5.14.4	Configuration	30
5.14.4.1	sample_time_ms	30
5.14.4.2	sample_distance_limit	30
5.14.4.3	sample_counter_top_n	31
5.15	deathlog	32
5.15.1	Purpose	32
5.15.2	Command	32
5.15.3	Example	32
5.16	functional	33
5.16.1	Purpose	33
5.16.2	Sub-Module	33
5.16.2.1	/workbench	33
5.16.2.2	/enchantment	33
5.16.2.3	/grindstone	33
5.16.2.4	/stonecutter	33
5.16.2.5	/anvil	33
5.16.2.6	/cartography	33
5.16.2.7	/enderchest	33
5.16.2.8	/smithing	33
5.16.2.9	/loom	33
5.17	system_message	34
5.17.1	Purpose	34
5.18	cleaner	35
5.18.1	Purpose	35
5.18.2	Command	35
5.18.2.1	/cleaner clean	35
5.18.3	Configuration	35
5.18.3.1	cron	35
5.18.3.2	key2age	35
5.18.3.3	ignore	36

5.18.3.3.1	ignoreItemEntity	36
5.18.3.3.2	ignoreLivingEntity	36
5.18.3.3.3	ignoreNamedEntity	36
5.18.3.3.4	ignoreEntityWithVehicle	36
5.18.3.3.5	ignoreEntityWithPassengers	36
5.18.3.3.6	ignoreGlowingEntity	36
5.18.3.3.7	ignoreLeashedEntity	36
5.19	command_scheduler	37
5.19.1	Purpose	37
5.19.2	Command	37
5.19.2.1	/command-scheduler trigger	37
5.20	command_permission	38
5.20.1	Purpose	38
5.20.2	How it works	38
5.20.3	Example	38
5.21	command_rewrite	39
5.21.1	Purpose	39
5.22	command_alias	40
5.22.1	Purpose	40
5.23	command_attachment	41
5.23.1	Purpose	41
5.23.2	Command	41
5.23.2.1	/command-attachment	41
5.23.3	Example	41
5.24	command_interactive	42
5.24.1	Purpose	42
5.24.2	Example	42
5.25	command_warmup	43
5.25.1	Purpose	43
5.26	command_cooldown	44
5.26.1	Purpose	44
5.27	command_toolbox	45
5.27.1	Purpose	45
5.27.2	Sub-Module	45
5.27.2.1	/bed	45
5.27.2.2	/extinguish	45
5.27.2.3	/feed	45
5.27.2.4	/fly	45
5.27.2.5	/god	45
5.27.2.6	/hat	45
5.27.2.7	/sit	45
5.27.2.8	/heal	45
5.27.2.9	/lore	45
5.27.2.10	/more	46
5.27.2.11	/ping	46
5.27.2.12	/realname	46

5.27.2.13	/nickname	46
5.27.2.14	/repair	46
5.27.2.15	/reply	46
5.27.2.16	/seen	46
5.27.2.17	/suicide	46
5.27.2.18	/top	46
5.27.2.19	/send_message	46
5.27.2.20	/send_broadcast	46
5.27.2.21	/send_actionbar	46
5.27.2.22	/send_title	46
5.27.2.23	/send_toast	46
5.27.2.24	/trashcan	47
5.27.2.25	/tppos	47
5.27.2.26	/warp	47
5.27.2.27	/burn	47
5.27.2.28	/help_op	47
5.27.2.29	/near	47
5.27.2.30	/jump	47
5.27.2.31	/compass	47
5.28	command_spy	48
5.28.1	Purpose	48
5.29	commandEvent	49
5.29.1	Purpose	49
5.29.2	Example	49
5.30	world	50
5.30.1	Purpose	50
5.30.2	Concept	50
5.30.3	Configuration	50
5.30.3.1	blacklist	50
5.30.4	Example	50
5.31	teleport_warmup	52
5.31.1	Purpose	52
5.31.2	Configuration	52
5.31.2.1	warmup_second	52
5.31.2.2	interrupt_distance	52
5.31.2.3	dimension	52
5.32	top_chunks	53
5.32.1	Purpose	53
5.32.2	Command	53
5.32.2.1	/chunks	53
5.32.3	Configuration	53
5.32.3.1	top	53
5.32.3.2	nearest_distance	53
5.32.3.3	hide_location	53
5.32.3.4	type2score	53
5.33	skin	54

5.33.1	Purpose	54
5.33.2	Command	54
5.33.2.1	/skin	54
5.33.3	Configuration	54
5.33.3.1	default_skin	54
5.33.3.2	random_skins	54
5.34	world_downloader	55
5.34.1	Purpose	55
5.34.2	Configuration	55
5.34.2.1	url_format	55
5.34.2.2	port	55
5.34.2.3	bytes_per_second_limit	55
5.34.2.4	context_cache_size	55
5.35	whitelist	56
5.35.1	Purpose	56
5.36	head	57
5.36.1	Purpose	57
5.36.2	Command	57
5.36.2.1	/head	57
5.37	profiler	58
5.37.1	Purpose	58
5.37.2	Command	58
5.37.2.1	/profiler	58
5.38	tester	59
5.38.1	Purpose	59
5.38.2	Command	59
5.38.2.1	/tester	59
5.39	multiplier	60
5.39.1	Purpose	60
5.39.2	Supported Numeric Types	60
5.39.3	Example	60
5.40	disabler	61
5.40.1	Purpose	61
5.40.2	Sub-Module	61
5.40.2.1	chat_speed_disabler	61
5.40.2.2	move_speed_disabler	61
5.40.2.3	max_player_disabler	61
5.41	anti_build	62
5.41.1	Purpose	62
5.41.2	Anti Types	62
5.41.3	Exapmle	62
5.42	color	63
5.42.1	Purpose	63
5.42.2	Sub-Module	63
5.42.2.1	Sign	63
5.42.2.2	Anvil	63

5.43	kit	64
5.43.1	Purpose	64
5.43.2	Concept	64
5.43.3	Command	64
5.43.3.1	/kit	64
5.44	temp_ban	65
5.44.1	Purpose	65
5.44.2	Command	65
5.44.2.1	/temp-ban	65
5.44.3	Example	65
5.45	command_meta	66
5.45.1	Purpose	66
5.45.2	Sub-Module	66
5.45.2.1	run	66
5.45.2.1.1	Purpose	66
5.45.2.1.2	Command	66
5.45.2.1.2.1	/run	66
5.45.2.1.3	Example	66
5.45.2.2	for_each	67
5.45.2.2.1	Purpose	67
5.45.2.2.2	Command	67
5.45.2.2.2.1	/for_each	67
5.45.2.2.3	Example	67
5.45.2.3	chain	67
5.45.2.3.1	Purpose	67
5.45.2.3.2	Command	67
5.45.2.3.2.1	/chain	68
5.45.2.3.3	Example	68
5.45.2.4	delay	68
5.45.2.4.1	Purpose	68
5.45.2.4.2	Command	68
5.45.2.4.2.1	/delay	68
5.45.2.4.3	Example	68
5.45.2.5	json	68
5.45.2.5.1	Purpose	68
5.45.2.5.2	Command	68
5.45.2.5.3	/json	69
5.45.2.5.4	Exaxmple	69
5.45.2.5.5	Reference	69
5.45.2.6	attachment	69
5.45.2.6.1	Purpose	69
5.45.2.6.2	Command	69
5.45.2.6.2.1	/attachment	69
5.45.2.6.3	Example	69
5.45.2.7	shell	70
5.45.2.7.1	Purpose	70

5.45.2.7.2	Example	70
5.46	gameplay	71
5.46.1	Sub-Module	71
5.46.1.1	multi_obsidian_platform	71
5.46.1.1.1	Purpose	71
5.46.1.1.2	Configuration	71
5.46.1.1.2.1	factor	71
5.46.1.2	carpet	71
5.46.1.2.1	Sub-Module	71
5.46.1.2.1.1	fake_player_manager	71
5.46.1.2.1.2	better_info	72
6	Q&A	73
6.1	Where is the configuration files?	73
6.2	What is .json file?	73
6.3	How can I edit a configuration file?	73
6.4	What is .dat file?	74
6.5	How to update fuji to a new version?	74
6.5.1	Step 1. Backup the data	74
6.5.2	Step 2. Test the new version in your test-environment	74
6.5.3	Step 3. Apply the changes to production-environment	74
6.6	Fuji conflits with some of my mods.	74
6.7	How can I report bugs or suggest new features?	74
7	Transformer	75
7.1	Command Transformer	75
7.2	Command Generator	76
8	Development	77
8.1	Setup the development environment	77
9	Suggestion	78
9.1	Suggestion on server-side mods	78
9.1.1	Explanation	78
9.1.2	Server-side mode list	79

Chapter 1

Concept

1.1 Configuration File

1.1.1 Definition

All files inside `config/fuji` directory are named `configuration file`

1.1.2 Types

Note: The types of configuration files

1. Control File A `control file` is used to control behaviours.

1.1. Main-Control File The `main-control file` refers to the `config/fuji/config.json` file, which is used to enable/disable a module.

1.2. Module-Control File Some modules will have their own control file, which is used to control the behaviour of the module.

2. User-Data File User-data files are used to store the data generated by the user.

1.2 Module

1.2.1 Definition

A `module` is used to provide a specific purpose.

Example: The purpose of modules

1. `ChatModule` provides chat-format customization.
2. `TpaModule` provides `/tpa` command.

1.2.2 Properties

The properties of a module are as follows:

1. **Can be disabled** You can disable any module completely in `main-control` file by setting the value of its `enable` key to `false`.
2. **Can work standalone** The codes of a module is self-contained, there is no reference to other modules.

1.2.3 Module Path

A `module` is identified by a unique `module path`.

Example: What a module-path looks like?

The module path of the module `tpa` is `tpa`.
The module path of the module `history` whose parent module is `chat`, is `chat.history`.
You will see a list of `enabled modules` identified by their `module path` at the server-startup process.

A `module` can have `sub-module`. The relationship between `parent-module` and `sub-module` is relative, and there is nothing special about `sub-module`.

1.2.4 How to enable/disable a module

You can enable/disable a module in `config/fuji/config.json` by setting the value of its `enable` key to `true/false`

A module will be enabled if the following conditions are met:

1. The required dependency mods are installed.
2. Its `parnet-module` is `enabled`.
3. The value of its `enable` key is set to `true`.

Example: How to enable a sub-module

To make the module `chat.display` enabled, you need to enable `chat` module first.

1.3 Job

1.3.1 Definition

A job is some things will be done repeatedly.

1.3.2 Cron Expression as Trigger Rule

A language named `cron` language is used to define when a job should be triggered.

Tip: Don't write cron expression by hand. Use generator!

A cron expression looks like `0 * * ? * *`, which means trigger the job every minute.

You can use the generator to generate a cron expression: <https://www.freeformatter.com/cron-expression-generator-quartz.html>

1.4 Regex

1.4.1 Definition

Regex is a language used to define the pattern of strings.

1.4.2 Reference

1. <https://regexr.com/>
2. <https://regex101.com/>

Chapter 2

Permission

2.1 Definition

A `permission` is used to decide whether a `player` can do something or not.

2.2 Types

To make the discussion clearer, we define the types of `permission` as follows:

1. **level permission** A permission level is a non-negative number used in vanilla minecraft. The higher number means the higher authority.
2. **string permission** Usually, a `string permission` is introduced by a `permission plugin`, such as `luckperms`.

2.3 What is the permission system used by fuji?

2.3.1 Explanation

Fuji use the mojang's `vanilla permission system`, which is based on `level permission`.

As a convention, most of the commands registered by fuji, requires `level permission` to be 0 to use. Only a few of the commands require the `level permission` to be 4 to use.

2.3.2 Set a string permission for a command.

By default, fuji only use the `level permission` as the requirement of a command. However, if you want to use `string permission` for a command, you can use `command permission module`, which is used to override the requirement of an existing command.

Example: Allow players to use `/seed` command

The command `/seed` provided by mojang requires `level permission` to be 3 to use. If you want to allow players to use `/seed` command, but you don't want to grant `op` for them. Then in this situation, you can grant the `string permission` for them: `/lp group default permission set fuji.permission.seed true`, which means that: set the requirement of command `/seed` to string permission `fuji.permission.seed`.

Example: Dis-allow players to use `/list` command

The command `/list` provided by mojang required `level permission` to be 0 to use. If you want to dis-allow players to use `/list` command, but because this command requires no `string permission` to use, so it's impossible to ban it via `luckperms`. In this situation, you can grant a `string permission`: `/lp group default permission set fuji.permission.list false` for them, which means that: set the requirement of command `/list` to string permission `fuji.permission.list`.

Example: Unset the override of requirement of the command

To undo the operation in the first example, just issue `/lp group default permission unset fuji.permission.seed`

2.4 Reference

1. https://minecraft.fandom.com/wiki/Permission_level

Chapter 3

Meta

3.1 Definition

A meta is a key-value pair.

Note:

Note that meta is introduced by luckperms mod, there is no meta in vanilla minecraft.

3.2 Example

Example: Set a meta for a group.

The home module supports the meta `fuji.home.hone_limit`, which controls how many homes a player can create. To set the max homes limits to 3: `/lp group default meta set fuji.home.hone_limit 3`

Example: Query all metas for a group

Issue `/lp group default info`

Chapter 4

Configuration

4.1 Main-Control File

4.1.1 common

The `common` options inside `config/fuji/config.json` will influence `all` modules.

4.1.1.1 quartz

Fuji use `quartz` library as scheduler, all the `Job` are managed by quartz. Quartz library use a language called `cron language` to define when to trigger a job.

4.1.1.1.1 logger_level

The logger level for `quartz`. The logger level from high to low are: OFF, FATAL, ERROR, WARN, INFO, DEBUG, TRACE, ALL.

Example: Enable all logs for quartz

Set the value to `ALL` to display all the messages from quartz. It's recommended to set at least `WARN` level, to avoid `console spam`.

4.1.1.2 backup

Fuji will back up the `config/fuji` directory automatically before it loads any module.

4.1.1.2.1 max_slots

How many `backup` should we keep?

4.1.1.2.2 skip

The list of `path resolver` to skip in backup.

Insert `head` means skip the folder `config/fuji/head`.

4.1.1.3 language

4.1.1.3.1 default_language

The default language to use.

Fuji also supports multi-language based on player's client-side language if the server is able to do so.

You need to enable `language module` to let fuji respect client-side's language settings.

Also, if the server can't support client-side's language, it will fallback to the `default_language`

4.2 Module-Control File

You can read more about `module-control` file for each module in [Module](#)

Chapter 5

Module

5.1 config

5.1.1 Purpose

This module provides the root command `/fuji`, which includes some operations on fuji itself.

5.1.2 Command

5.1.2.1 `/fuji reload`

Reload all configuration files and all modules.

Note: Module itself can't be hot reloaded

After you **enable** or **disable** a module, you must **restart** your server.

5.2 language

5.2.1 Purpose

This module provides client-side multi-language support.

Note: What is client-side language?

When the client joins a server, it will send its client options, including the client-side language value. The server can send messages in language used by the client later.

5.2.2 Feature

1. **Client-Side Respect** The client-side language will be respected if possible.
2. **Lazy-load** Only load the necessary language into the memory.

5.2.3 Difference

Disabled: All the players use the `default_language`.

Enabled: Fuji will **try** to respect the player's client-side language, if the server-side supports.

5.3 chat

5.3.1 Purpose

This module provides chat system customization.

5.3.2 Feature

1. You can create your own **regex transformer** to replace **chat message** sent by players.
2. You can insert any **placeholder** like **%world:name%**
3. You can insert player's prefix and suffix. Just insert **%fuji:player_prefix%** and **%fuji:player_suffix%**.
4. You can insert **item**, **inv** and **ender** to display your item, inventory and ender-chest.
5. You can insert **Steve** to mention another player named Steve.
6. You can insert **pos** to show the position.
7. You can use **markdown language** to define simple format.
8. You can use **mini-message language** to define complex format.
9. Besides the **server chat format**, each player can also set their per-player chat format.
10. This module doesn't **cancel** the vanilla chat events, so it can work with other chat related mods.

Note: Set prefix and suffix for players

Luckperms is required to set **prefix** and **suffix**.

After you installed **luckperms** mod, just issue **/lp group default meta setprefix <yellow>[awesome]** to assign prefix.

Don't forget to change the format in chat module, and issue **/fuji reload**

Tip: Write complex style using mini-language

You can use **mini-language** to write complex style Text.

See more:

1. <https://docs.advntr.dev/minimessage/format.html>
2. <https://placeholders.pb4.eu/user/quicktext>

5.3.3 Command

5.3.3.1 /chat

5.3.4 Placeholder

1. `%fuji:pos%` player current position
2. `%fuji:player_prefix%` player prefix
3. `%fuji:player_suffix%` player suffix

5.3.5 Configuration

5.3.5.1 format

The server chat format for all players.

5.3.5.1.1 rewrite

The `regex language` list used to `rewrite` the player chat message.

You can use regex language to transform player's chat input (only chat message, no command usage).

5.3.5.1.2 mention_player

If you insert **Steve** in chat message, then the player named Steve will get audio mention.

5.3.5.1.2.1 sound

The type of `sound` used to notify the mentioned player.

Tip: Query all identifiers of sound

Issue the command `/playsound ...`

5.3.5.1.2.2 volume

5.3.5.1.2.3 pitch

5.3.5.1.2.4 repeat_count

The sound repeat count.

5.3.5.1.2.5 interval_ms

The interval between each repeat.

5.3.5.1.3 spy

Log chat information into server-console.

5.3.6 Sub-Module**5.3.6.1 display****5.3.6.1.1 Placeholder**

1. `%fuji:item%` display player's main-hand item.
2. `%fuji:inv%` display player's inventory.
3. `%fuji:ender%` display player's ender-chest.

5.3.6.1.2 Configuration**5.3.6.1.2.1 expiration_duration_s**

For each display data, how long should we save in the memory.

Note that if a player shares its inventory items, then fuji will save a copy of his inventory data in the memory.

5.3.6.2 history

New joined players can see the historical chat messages.

5.3.6.2.1 Configuration**5.3.6.2.1.1 buffer_size**

How many chat messages should we save, so that we can send for a new-joined player.

5.3.7 Reference

1. [Text Placeholder API - default placeholders](#)
2. [Luckperms - prefix, suffix and meta](#)

5.4 placeholder

5.4.1 Purpose

This module provides more `placeholder` for `Text Placeholder API` mod.

5.4.2 Placeholder

1. `%fuji:player_mined%` sum of mining blocks of a player
2. `%fuji:server_mined%` sum of mining blocks of a server
3. `%fuji:player_placed%` sum of placed blocks of a player
4. `%fuji:server_placed%` sum of placed blocks of a server
5. `%fuji:player_killed%` sum of killed entities of a player
6. `%fuji:server_killed%` sum of killed entities of a server
7. `%fuji:player_moved%` sum of moved distance of a player
8. `%fuji:server_moved%` sum of moved distance of a server
9. `%fuji:player_playtime%` sum of playtime of a player
10. `%fuji:server_playtime%` sum of playtime of a server
11. `%fuji:health_bar%` the health bar of a player
12. `%fuji:rorate hello%` rotate the string **hello**
13. `%fuji:has_permission%` check luckperms permission
14. `%fuji:has_meta%` get luckperms meta
15. `%fuji:random_player%` get a random online player
16. `%fuji:random 1 5%` get a random number from 1 to 5
17. `%fuji:escape%` escape a placeholder form the parser.
An optional numebr argument is used as the levels to escape.
18. `%fuji:date%` get current date.
An optional string argument is used to set the **date formatter**, for example, `%fuji:date HH:MM%`.
See also: <https://docs.oracle.com/javase/8/docs/api/java/text/SimpleDateFormat.html>

Tip: Use placeholder in language file

It's allowed to write placeholders in language file.

Note: Some other mods that provide more placeholders

<https://placeholders.pb4.eu/user/mod-placeholders/>

5.5 motd

5.5.1 Purpose

This module provides motd customization.

5.6 nametag

5.6.1 Purpose

This module provides nametag customization.

5.6.2 Configuration

5.6.2.1 update_cron

The cron used for the job to **update** the properties of **display entity**.

5.6.2.2 style

Define the style for the nametag **display entity**.

5.6.2.3 render

Define the nametag render behaviour.

Note: The explanation of each field in style and render

You can refer to the minecraft wiki about **display entity**: <https://minecraft.wiki/w/Display>

5.6.3 Example

Example: Set background to blue color

```
"background": -16776961
```

Example: Set half transparency

```
"text_opacity": 128
```

Example: Scale the size of text into double

```
"scale": {  
  "x": 2.0,  
  "y": 2.0,  
  "z": 2.0  
},
```

5.6.4 Reference

1. <https://minecraft.wiki/w/Display>

5.7 `tab_list`

5.7.1 Purpose

This module provides tab list customization.

5.7.2 Configuration

5.7.2.1 `update_cron`

The cron used for the job to update the tab list.

5.7.2.2 `style`

The style for tab list.

5.7.3 Sub-Module

5.7.3.1 `sort`

5.7.3.1.1 Purpose

If enable this module, the `player names` in `tab list` will be sorted by `weight`. The default weight is 0, the range of weight is [0, 675], which means you can set at most 676 sort groups.

5.7.3.1.2 How it works

The tab list sort method is client-side decided. So the workaround is to send virtual-player entry to the client-side, and hide the real player in client-side's tablist.

In this case, the client-side will find that, all command target selector will display the virtual-player.

And you can see the virtual-player in client-side's Player Reporting UI.

Note: The virtual player have no performance issue

The virtual-player is just an entry listed in tab list, when the client ask the server tab list, the server lie with the virtual-player list.

There is not a real player entity in the server side, so no extra performance problem.

The sync method is event-based, and cached, so the performance is good.

5.7.3.1.3 Example

Example: Set a weight for a group

Issue the command `/lp group default meta set fuji.tab_list.sort.weight 1` After you set a new weight, you should Issue the command `/fuji reload` or re-connect to refresh the tab-list.

5.7.3.1.4 Sub-Module**5.7.3.1.4.1 sync_game_profile**

Whether to copy the game profile from real-player to virtual-player.

5.7.3.2 faker**5.7.3.2.1 Purpose**

This module is used to send random fake data to client.

5.8 tpa

5.8.1 Purpose

This module provides teleport request for players.

5.8.2 Command

1. /tpa
2. /tpahere
3. /tpaaccept
4. /tpadeny

5.8.3 Configuration

5.8.3.1 timeout

Tpa request expiration duration.unit is second

5.9 back

5.9.1 Purpose

This module allows player to teleport back to **last teleport point** or **death point**.

5.9.2 Command

1. `/back`

5.9.3 Configuration

5.9.3.1 ignore_distance

If the player's teleportation destination is close enough, we ignore this teleportation.

5.10 home

5.10.1 Purpose

This module allows players to create their home.

5.10.2 Command

1. `/home`

5.10.3 Meta

1. `fuji.home.home_limit` The home number per player limit.

5.11 pvp

5.11.1 Purpose

This module provide pvp state toggle.

5.11.2 Command

1. /pvp

5.12 afk

5.12.1 Purpose

This module provides afk detection, afk event, afk name customization and afk effects.

5.12.2 Command

1. /afk

5.12.3 Configuration

5.12.3.1 format

The tab-name format when a player is afk

5.12.3.2 afk_checker

5.12.3.2.1 cron

The cron to define how the afk_checker is triggered.

5.12.3.3 afk_event

Execute commands on afk events.

Example: Kick a player if he enters afk state

```
"on_enter_afk": [  
  "send-broadcast <gold>Player %player:name% is now afk",  
  "kick %player:name% You are kicked beacause on afk."  
]
```

5.12.3.4 effect

Afk effects are applied if a player enters afk state.

5.12.3.4.1 invulnerable

Immune to all damage?

5.12.3.4.2 targetable

Can be targeted by a hostile entity ?

5.12.3.4.3 moveable

Can be moved if in afk state ?

5.13 rtp

5.13.1 Purpose

Provides random teleportation.

5.13.2 Feature

1. Per dimension configurable.
2. Ignore fluid blocks, such as water and lava.
3. Ignore powered snow.

5.13.3 Command

5.13.3.1 /rtp

5.13.3.1.1 Argument

1. **-dimension** target dimension

Tip: Improve the performance on rtp

1. It's highly recommended to pre-gen the world chunks. To gen a new chunk during rtp requires about 2 10 seconds. If a chunk is pre-gen, then it will be fast.
2. It's highly recommended to pre-gen the world chunks. To gen a new chunk during rtp requires about 2 10 seconds. If a chunk is pre-gen, then it will be fast.

5.13.4 Configuration

5.13.4.1 setup

Teleport setup per dimension. Dimensions that are not in the list will be disabled to rtp.

5.14 works

5.14.1 Purpose

This module provides a bill-board gui for players to show their **works**.

5.14.2 Concept

A **work** is a **teleporation point**.

The types of work are as follows:

1. **Non-Production-Work** the project don't produce any resource (e.g. bone, string, coal).
2. **Production-Work** the project produce some resource.

Tip: The main difference between non-production work and production-work

For a production-work, fuji provides the **production sample** to count the **hopper** and **minecart-hopper**

Tip: About the production counter

1. You can use the production counter provided by production work to sample the output.
2. This module works with carpet-fabric's hopper counter. You can use both of them at the same time.
3. The hopper counter provided by this module will not destroy the item.

5.14.3 Command

1. `/works`

5.14.4 Configuration

5.14.4.1 `sample_time_ms`

For a production-work, how long should we sample it?

5.14.4.2 `sample_distance_limit`

For a production-work, how large the radius should we considered as the work's production.

5.14.4.3 sample_counter_top_n

For a production-work, we only display the topN output items.

5.15 deathlog

5.15.1 Purpose

This module logs the inventory on player death.

5.15.2 Command

1. `/deathlog`

5.15.3 Example

Example: Query the logs for a player

Issue the command `/deathlog view Steve`

Example: Restore a death log for a player

Issue the command `/deathlog restore Steve 0 Steve`

Tip: The death log index number is clickable

You can click the number to quickly restore the logged inventory to your inventory.

5.16 functional

5.16.1 Purpose

This module allows players to open a virtual-gui for functional-block.

5.16.2 Sub-Module

5.16.2.1 /workbench

5.16.2.2 /enchantment

5.16.2.3 /grindstone

5.16.2.4 /stonecutter

5.16.2.5 /anvil

5.16.2.6 /cartography

5.16.2.7 /enderchest

5.16.2.8 /smithing

5.16.2.9 /loom

5.17 system_message

5.17.1 Purpose

This module provides **system message** customization.

Note: What is called system message?

All the messages inside the mojang's official language file are called system message, such as: [en-us.json for minecraft 1.21](#)

Here are some messages that are system message:

1. player join and leave server message
2. player advancement message
3. player death message
4. player command feedback
5. player whitelist message

Note: Server-side message and Client-side message

It should be noted that, not all messages inside mojang's official language file are server-side message. If a message is client-side message, which means that the message is only used in client-side, then the module will have no effects on this message.

5.18 cleaner

5.18.1 Purpose

This module provides the entity cleaner to remove specified entities automatically.

Note: Only use this module to clean some edge-case entity

Since the vanilla minecraft also has a cleaner to remove the item stack in the ground, so it's recommended to only use this module to clean some weak-loading entities, like: the sand item stack ...

5.18.2 Command

5.18.2.1 `/cleaner clean`

Note: The cleaner will keep silent if cleans nothing

If the cleaner cleans nothing, then it will keep silent. (Which means you will not see any message in console, or in-game chat)

5.18.3 Configuration

5.18.3.1 `cron`

The cron used to define the job to trigger `/cleaner clean`.

5.18.3.2 `key2age`

The key is translatable key, which you can query in [en_us.json language file in minecraft 1.21](#).

The translatable key of entity starts with `entity.minecraft`.

The translatable key of item starts with `item.minecraft` and `block.minecraft`.

The age is the existence time of the entity, the unit of age is game tick, which means $20 \text{ age} = 20 \text{ ticks} = 1 \text{ second}$.

The cleaner will only remove the entities whose translatable key equals key, and age greater equals the defined age, and the entity must not in the ignore list.

Example: Clean the sand-block entity lives longer than 60sec

```
"block.minecraft.sand": 1200.
```

5.18.3.3 ignore

Entities match the ignore list will not be cleaned.
The cleaner will always ignore the following types:

1. player
2. any block attached entity (e.g. leash_knot)
3. any vehicle entity (e.g. minecart, boat ...)

5.18.3.3.1 ignoreItemEntity

Should we ignore all item entity.

5.18.3.3.2 ignoreLivingEntity

Should we ignore all living entity.

If you want the cleaner to remove monster or animals, you should enable this option.

5.18.3.3.3 ignoreNamedEntity

Should we ignore named entity.(With name tag, or name changed by anvil.)

5.18.3.3.4 ignoreEntityWithVehicle

Like entity riding in some other entity, e.g. minecraft, pig or spider

5.18.3.3.5 ignoreEntityWithPassengers

5.18.3.3.6 ignoreGlowingEntity

5.18.3.3.7 ignoreLeashedEntity

Tip: See what is cleaned in cleaner broadcast.

Hover your mouse on the cleaner broadcast, you can see what has been removed.

5.19 command_scheduler

5.19.1 Purpose

This module allows you to define jobs using cron express to run commands.

5.19.2 Command

5.19.2.1 /command-scheduler trigger

Immediately trigger a job.

5.20 command_permission

5.20.1 Purpose

This module provides the customization of **the requirement of any commands**.

5.20.2 How it works

The vanilla minecraft use a command system called brigadier. All the commands are registered, parsed and executed by brigadier. In this system, all commands are build into a tree structure. For example, like the command `/gamemode creative Steve` is composed by 3 command node:

1. **literal command node** `gamemode`
2. **argument command node** a valid gamemode
3. **argument command node** a valid player

And the command node path stands the tree node path. For `/gamemode creative Steve`, the path is ["gamemode", "gamemode", "target"]. Also, each command node has its requirement, which is a condition to check if the command user can use the command node. This module can override the requirement of a command node into a permission with prefix `fuji.permission;command_node_path;`.

Tip: Query all the registered command path

You can query a command path using `/lp group default permission set fuji.permission...` way.

5.20.3 Example

Example: Allow everyone to use `/op` command

Issue the command `/lp group default permission set fuji.permission.op true`

5.21 `command_rewrite`

5.21.1 Purpose

This module allows you to use regex language to rewrite the command line a player issued.

5.22 `command_alias`

5.22.1 Purpose

This module allows you to define command alias, which redirect to the existing command node.

Note: A command node is identified by path

See also: <https://minecraft.fandom.com/wiki/Commands>

5.23 command_attachment

5.23.1 Purpose

This module allows you to attach commands into itemstack.

5.23.2 Command

5.23.2.1 /command-attachment

Note: The same item-stack shares the-same instance

If you hold stick * 64 in your mainhand, then all the sticks share the same attached commands.

5.23.3 Example

Example: Create a magic-stick which will heal player when click.

Hold a 'stick' item in your main hand. Issue the command **/command-attachment attach-one heal**

Example: Create a magic-stick giving you diamond when left click (with use limit 3 times) and giving you gold when right click (with use limit 5 times).

Hold a 'stick' item in your main hand.

Issue the command **/command-attachment attach-one -maxUseTimes 3 -interactType LEFT give %player:name% minecraft:diamond 1**

Issue the command **/command-attachment attach-one -maxUseTimes 5 -interactType RIGHT give %player:name% minecraft:gold_ingot 1**

Example: Query the attached commands in the mainhand item

Issue the command **/command-attachment attach-one -maxUseTimes 3 -destroyItem false give %player:name% minecraft:apple 1**

5.24 command_interactive

5.24.1 Purpose

This module allows you to write commands in **sign block**.

5.24.2 Example

Example: Basic usage

```
/say hi %player:name%  
line 2 empty  
line 3 empty  
line 4 empty
```

Example: Add a prefix description text

```
prefix /say first  
/say the second  
/say hi %player:name%  
/say the last command
```

Example: Concat commands between lines

```
prefix /say this is  
the first /say and the  
second  
line 4 empty
```

Tip:

1. You need to press shift + right click to edit an interactive sign
2. The command is executed as the player.

5.25 `command_warmup`

5.25.1 Purpose

This module adds a warmup countdown before command usage.

Note: The difference between command warmup and command cooldown

command warmup is before command usage, while command cooldown is after command usage.

5.26 `command_cooldown`

5.26.1 Purpose

This module adds a cooldown countdown after the command usage.

5.27 command_toolbox

5.27.1 Purpose

This module provides some simple commands. (misc commands)

5.27.2 Sub-Module

5.27.2.1 /bed

5.27.2.2 /extinguish

5.27.2.3 /feed

5.27.2.4 /fly

5.27.2.5 /god

5.27.2.6 /hat

5.27.2.7 /sit

5.27.2.8 /heal

5.27.2.9 /lore

Example:

Issue the command /lore set ;rainbow;the first
line;newline; ;bold; ;green;the second

5.27.2.10 /more
5.27.2.11 /ping
5.27.2.12 /realname
5.27.2.13 /nickname
5.27.2.14 /repair
5.27.2.15 /reply
5.27.2.16 /seen
5.27.2.17 /suicide
5.27.2.18 /top
5.27.2.19 /send_message
5.27.2.20 /send_broadcast
5.27.2.21 /send_actionbar
5.27.2.22 /send_title
5.27.2.23 /send_toast

Example:

```
Issue the command /send-title Steve -mainTitle "§rain-  
bow¿Hello" -subTitle "§blue¿World" -fadeInTicks 60 -  
stayTicks 60 -fadeOutTicks 60
```

Example:

```
Issue the command /foreach send-title %player:name% -  
mainTitle "§rainbow¿Hello %player:name%"
```


5.27.2.24 /trashcan

5.27.2.25 /tppos

5.27.2.26 /warp

5.27.2.27 /burn

5.27.2.28 /help_op

5.27.2.29 /near

5.27.2.30 /jump

5.27.2.31 /compass

5.28 `command_spy`

5.28.1 Purpose

Log command issue into console.

5.29 command_event

5.29.1 Purpose

Execute commands on specific events.

5.29.2 Example

Example: Welcome the new-bie player

```
"on_player_first_joined": {  
  "command_list": [  
    "send-broadcast <light_purple>Welcome new player %player:name% to join us!",  
    "kit give %player:name% <kit-name>",  
    "execute as %player:name% run rtp",  
    "delay 10 spawnpoint %player:name%"  
  ]  
},
```

5.30 world

5.30.1 Purpose

Provides a unified world management.

5.30.2 Concept

What is the difference between world, dimension and dimension type ?

Well, in the early stage of minecraft, it only support single-dimension, which means 1 world only contains 1 dimension.

And now, 1 world can support multi dimension. Sometimes, you will see world and dimension means the same thing.

But clearer, we say: 1 world can contains 1 or more dimension, and each dimension has its dimension type.

Usually, you can say a mod adds extra dimension type and create extra dimension with that dimension type instead of extra world

See also: https://minecraft.wiki/w/Dimension_definition

See also: https://minecraft.wiki/w/Dimension_type

In vanilla minecraft, 1 world contains 3 dimensions (minecraft:overworld, minecraft:the_nether, minecraft:the_end). You can see the dimension of a world in world/level.dat file.

dimension type is used to create dimension, there are 4 dimension type in vanilla minecraft: minecraft:overworld, minecraft:overworld_caves, minecraft:the_nether and minecraft:the_end

In order to create extra dimensions of a dimension type, you need to at least exist one dimension of that dimension type.

Instead of writing data into the file world/level.dat, fuji will load the extra dimensions in game dynamically.

The file server.properties is used for the default world properties of extra dimensions

5.30.3 Configuration

5.30.3.1 blacklist

The dimensions in the blacklist will not be operated by this module. Use blacklist to avoid mis-operation.

5.30.4 Example

Example: Create an extra the_nether dimension

Issue	the	command	/world	create	my_nether
minecraft:the_nether					

Example: Delete the extra dimension

Issue the command `/world delete fuji:my_nether`

Example: Reset the extra dimension with random seed

Issue the command `/world reset fuji:my_nether`

Tip: Specify a seed for dimension

Issue the command `/world create my_nether minecraft:the_nether -seed 1234567890` Issue the command `/world reset fuji:my_nether -use-the-same-seed`

Tip: Make a resource-world that automatically reset every day

You can use command-scheduler module to execute `world reset` command automatically.

5.31 teleport_warmup

5.31.1 Purpose

This module adds a warmup cooldown before player-teleportation.

The teleportation will be cancelled if:

1. the player runs too far.
2. the player gets damage.
3. the player is in combat.

5.31.2 Configuration

5.31.2.1 warmup_second

The second to wait before the teleportation.

5.31.2.2 interrupt_distance

How far should we cancel the teleportation.

5.31.2.3 dimension

Only allowed in the following dimensions.

Warning: Dimensions that created by other mods may have special behaviour

Some other mods will add extra dimension (like, the mod the-bumblezone-fabric). Their dimension portal will work in a different way, so teleport warmup module may not compatibility with these mods. In the default options, we only allow teleport warmup works in the vanilla minecraft dimensions.

5.32 top_chunks

5.32.1 Purpose

This module compute a **laggy score** for all loaded chunks, and return the topN lagged chunks. Higher score means more lagged.

5.32.2 Command

5.32.2.1 /chunks

5.32.3 Configuration

5.32.3.1 top

The top chunks to show in /chunks command

5.32.3.2 nearest_distance

For a chunk, how much the radius used to search the nearest player around the chunk.

5.32.3.3 hide_location

Should we hide the chunk-position for a lagged-chunk? Hide chunk location to avoid grief or privacy purpose.

5.32.3.4 type2score

The dict to define how lagged a type(entity/entity_block) should be. For example: `this.put("entity.minecraft.zombie", 4);` means there are 15 zombies inside a chunk, then the chunk gets score $15 * 4 = 60$ Any other types not inside the dict used the score defined for type default

5.33 skin

5.33.1 Purpose

This module provides skin management.

5.33.2 Command

5.33.2.1 /skin

5.33.3 Configuration

5.33.3.1 default_skin

The default skin used for player who has no skin set.

5.33.3.2 random_skins

Random skin for fake-player, if you enable the local skin for fake-player.

5.34 world_downloader

5.34.1 Purpose

This module allows a player to download nearby chunks.

5.34.2 Configuration

5.34.2.1 url_format

The url format used to broadcast

5.34.2.2 port

5.34.2.3 bytes_per_second_limit

Max download speed limit for each connection.

5.34.2.4 context_cache_size

Max download request allowed in the memory at the same time.

5.35 whitelist

5.35.1 Purpose

This module makes the mojang vanilla whitelist system only compares the **username** and **ignore UUD**.

Warning: Only enable this module in offline-mode server

If you are hosting a online-mode server, you will never need to enable this module.

5.36 head

5.36.1 Purpose

This module allows players to buy heads from head-database.

5.36.2 Command

5.36.2.1 /head

5.37 profiler

5.37.1 Purpose

This module shows server health status, such as os, vm, cpu, ram, tps, mspt and gc.

Warning:

Enable this module requires spark mod installed.

5.37.2 Command

5.37.2.1 /profiler

5.38 tester

5.38.1 Purpose

This module is only used for development. If you are a developer, you can register new commands into this module for test-purpose.

5.38.2 Command

5.38.2.1 /tester

Warning:

You should not use this module at your production-environment, because it almost does nothing useful.

5.39 multiplier

5.39.1 Purpose

This module provides some **numeric multiplier**.

5.39.2 Supported Numeric Types

1. **damage** damage to player
2. **experience** experience a player gained

5.39.3 Example

Example: Double the damage from zombie to a player

Issue the command `/lp group default meta set fuji.multiplier.damage.minecraft:zombie 2`

Example: Cancel the fall damage

Issue the command `/lp group default meta set fuji.multiplier.damage.minecraft:fall 0`

Example: Double all damage to a player

Issue the command `/lp group default meta set fuji.multiplier.damage.all 2`

Example: Double all experience a player gained

Issue the command `/lp group default meta set fuji.multiplier.experience.all 2`

Example: Half all damage to a player

Issue the command `/lp group default meta set fuji.multiplier.damage.all 0.5`

5.40 disabler

5.40.1 Purpose

This module provides some **disabler** to disable checkers inside **vanilla minecraft**.

5.40.2 Sub-Module

5.40.2.1 chat_speed_disabler

Disable **Kicked for spamming**.

5.40.2.2 move_speed_disabler

Disable **moved too quickly** and **vehicle too quickly check**.

Warning: The movement anti-cheat inside vanilla minecraft is bad

Inside the vanilla minecraft server, there is a checker used to check if the player moves correctly. However, this checker usually makes wrong detection, and force setback the player, which makes the client-side gameplay feel lagged.

5.40.2.3 max_player_disabler

Disable the max players limit of the server.

5.41 anti_build

5.41.1 Purpose

This module allows you to ban the interaction with some item/block/entity.

5.41.2 Anti Types

1. break_block
2. place_block
3. interact_item
4. interact_block
5. interact_entity

Tip: Query identifiers

To query block identifier, use `/setblock ...` command.
 To query entity identifier, use `/summon ...` command.
 To query item identifier, use `/give ...` command.

5.41.3 Exapmle

Example: Ban TNT

add `minecraft:tnt` into `place_block` list

Example: Ban TNT but allow some players to use

Use `/lp user {player} permission set`
`fuji.anti_build.place_block.bypass.minecraft:tnt` to allow a player
 place the tnt.

5.42 color

5.42.1 Purpose

This module provides colorize for things.

5.42.2 Sub-Module

5.42.2.1 Sign

5.42.2.2 Anvil

5.43 kit

5.43.1 Purpose

This module allows you to make kits.

5.43.2 Concept

A **kit** is a set of itemstack.

5.43.3 Command

5.43.3.1 /kit

5.44 temp_ban

5.44.1 Purpose

This module provides temp-ban.

5.44.2 Command

5.44.2.1 /temp-ban

5.44.3 Example

Example:

Issue the command `/temp-ban player Steve 12h30m15s bad boy`

5.45 `command_meta`

5.45.1 Purpose

This module provides commands to operate on commands.

5.45.2 Sub-Module

5.45.2.1 `run`

5.45.2.1.1 Purpose

This module provides `/run` command, which can run a command with context.

5.45.2.1.2 Command

5.45.2.1.2.1 `/run`

5.45.2.1.3 Example

Example: Give random diamonds to all online players

Issue the command `/run as console give @a minecraft:diamond %fuji:random 8 32%`

Example: Give all online players random diamonds

Issue the command `/run as console foreach give %fuji:escape player:name% minecraft:diamond %fuji:escape fuji:random 8 32%`

Example: Execute a commands as a player

Issue the command `/run as player Steve back`

Example: Execute a commands as fake-op

Issue the command `/run as fake-op Steve give %player:name% minecraft:apple 1`

5.45.2.2 for_each

5.45.2.2.1 Purpose

This module provides /foreach command.

If a command is only targeted for single player, you can use foreach to apply it for all players

5.45.2.2.2 Command

5.45.2.2.2.1 /for_each

5.45.2.2.3 Example

Example:

Issue the command `/foreach say hello %player:name%`

Tip: Escape the placeholder properly

If you use foreach in scheduler module, then you should escape (Write phfuji:escape player:name instead of phplayer:name) the placeholder.

It's because the scheduler module will try to parse the placeholder, and you need to escape the placeholder, so that the placeholder can be parsed by foreach command.

Here is an example about escape the foreach command in scheduler command list: `/foreach give phfuji:escape player:name minecraft:diamond 16`

5.45.2.3 chain

5.45.2.3.1 Purpose

A chain command allows you to run another 2 commands, the first is any command, and the second is the chain command.

Tip: The return value of a command

In vanilla minecraft, the return value of command, are failed, pass and success.

5.45.2.3.2 Command

5.45.2.3.2.1 /chain

5.45.2.3.3 Example

Example: A nested chain.

Issue the command `/chain say 1 chain say 2 chain say 3`

Example: A breakable chain.

Issue the command `/chain bad command here chain say 2`

5.45.2.4 delay

5.45.2.4.1 Purpose

Delay command allows you to execute a command.

5.45.2.4.2 Command

5.45.2.4.2.1 /delay

5.45.2.4.3 Example

Example: A basic usage

Issue the command `/delay 3 say three seconds passed`

Example: A nested delay

`1/delay 1 delay 2 delay 3 say 6 seconds passed.`

5.45.2.5 json

5.45.2.5.1 Purpose

Provides a unified json editor.

5.45.2.5.2 Command

5.45.2.5.3 /json

5.45.2.5.4 Exaxmple

Example: Read a key

Issue the command `/json read "config/fuji/config.json" "$common.quartz.logger_level"`

Example: List keys

Issue the command `/json read "config/fuji/config.json" "$modules.keys()`

Example: Set a key

`/json write "config/fuji/config.json" "$common.quartz.logger_level"
NULL null`

5.45.2.5.5 Reference

1. <https://goessner.net/articles/JsonPath/>

5.45.2.6 attachment

5.45.2.6.1 Purpose

Provides a unified attachment facility, which can attach any data to any object.

5.45.2.6.2 Command

5.45.2.6.2.1 /attachment

5.45.2.6.3 Example

Example:

Issue the command `/attachment set news today hello world`

Example:

Issue the command `/attachment get news today`

5.45.2.7 shell**5.45.2.7.1 Purpose**

This module provides `/shell` command, which executes command line in your host shell.

Warning: This is a dangerous module

This module is a powerful and dangerous module, not recommended to enable it.

5.45.2.7.2 Example**Example: Create a file using placeholder**

Issue the command `/shell touch %player:name%.dangerous`

Example: Execute a program in the host os

Issue the command `/shell emacs`

Example: Possible to download a virus from Internet and execute it!

Issue the command `/shell ...`

5.46 gameplay

5.46.1 Sub-Module

5.46.1.1 multi_obsidian_platform

5.46.1.1.1 Purpose

This module makes the every **ender portal frame** generate its own **obsidian platform**.

makes every EnderPortal generate its own Obsidian Platform (Up to 128 in survival-mode).

You can even use creative-mode to build more Ender Portal and more ObsidianPlatform.

Please note that: all the obsidian-platform are vanilla-respect, which means they have the same chunk-layout and the same behaviour as vanilla obsidian-platform which locates in (100,50,0)

5.46.1.1.2 Configuration

5.46.1.1.2.1 factor

The coordination-conversion factor between overworld and the_end. In vanilla minecraft, the factor between overworld and the_nether is 8.

5.46.1.2 carpet

5.46.1.2.1 Sub-Module

5.46.1.2.1.1 fake_player_manager

Purpose

Enable this module requires carpet-fabric mod installed. This module provides some management for fake-player.

Command

/player who = query the owner of the fake-player.

/player renew = renew all of your fake-players.

Configuration

caps_limit_rule

How many fake-player can each player spawn (in different time)? The tuple means (day_of_week, minutes_of_the_day, max_fake_player_per_player). The range of day_of_week is [1,7]. The range of minutes_of_the_day is [0, 1440]. For example: (1, 0, 2) means if the days_of_week $i=1$, and minutes_of_the_day $i=0$, then the max_fake_player_per_player now is 2. Besides, you can add multi rules, the rules are checked from up to down. The first rule that matches current time will be used to decide the max_fake_player_per_player. You can issue

/player who to see the owner of the fake-player. Only the owner can operates the fake-player. (Op can bypass this limit)

renew_duration.ms

How long should we renew when a player issue command /player renew The command /player renew allows the player to manually renew all of his fake-player. If a fake-player don't gets renew, then it will expired and get killed. Use-case: to avoid some long-term alive fake-player.

transform_name

The rule to transform the name of fake-player. Use-case: add prefix or suffix for fake-player.

use_local_random_skins_for_fake_player

Should we use local skin for fake-player? Enable this can prevent fetching skins from mojang official server each time the fake-player is spawned. This is mainly used in some network situation if your network to mojang official server is bad.

5.46.1.2.1.2 better_info

Purpose

Add nbt query for /info block command. Add the command /info entity.

Chapter 6

Q&A

6.1 Where is the configuration files?

As a convention, all the files are placed in `config/fuji/` directory.

6.2 What is .json file?

A json file is a text file, whose name normally ends with `.json`.

6.3 How can I edit a configuration file?

To ensure the `readable` and `transparent`, most of the files are saved as `pure text format`. You can open them with a `text editor`.

Tip: Use a modern text editor.

Some files may have a large number of lines, so it's highly recommended to use a **modern** text editor, which can highlight symbols and reveal the structure of the file, such as:

1. Visual Studio Code
2. [Visual Studio Code - Web Online](#)
3. Vim
4. Emacs
5. Sublime Text

6.4 What is .dat file?

The file whose name ends with `.dat` are the vanilla minecraft NBT format file. To open such a file, you need to use a NBT Editor, such as NBTEditor.

6.5 How to update fuji to a new version?

6.5.1 Step 1. Backup the data

You should back up the `config/fuji` directory.

6.5.2 Step 2. Test the new version in your test-environment

Put the new version of fuji into `mods/` directory, start the server, and adjust the configuration to what you want.

Warning: Don't test new changes directly in your production-environment

It's highly recommended to setup a test-environment for a network-maintainer, so that you can test and tweak installed mods into what you want, and avoid un-expected situations.

6.5.3 Step 3. Apply the changes to production-environment

Now, it's ready to apply the changes to your production-environment.

6.6 Fuji conflits with some of my mods.

You can disable the conflicting module.If possible, create an issue at [fuji issue page](#), so that we can solve this later.

6.7 How can I report bugs or suggest new features?

You can create an issue at [fuji issue page](#)

Chapter 7

Transformer

7.1 Command Transformer

The following commands can be transformed:

1. `/blockcycling` = `/give player minecraft:debug_stick`
2. `/blockinfo` = `/data get block`
3. `/blocknbt` = `/data get block`
4. `/entityinfo` = `/data get entity`
5. `/entitynbt` = `/data get entity`
6. `/customrecipe` = **recipe editor**
7. `/exp` = `/experience`
8. `/gm`, `/gms`, `/gms` = `/gamemode`
9. `/findbiome` = `/locate biome`
10. `/tempfly` = `use luckperms temporal permission`
11. `/flyspeed` = `/attribute player minecraft:generic.flying_speed`
12. `/walkspeed` = `/attribute player minecraft:movement_speed`
13. `/maxhealth` = `/attribute player minecraft:generic.max_health`
14. `/groundclean` = `/kill @e[type=...]`
15. `/spawner` = `interact spawner block with spawn egg`
16. `/spawnmob` = `/summon`
17. `/shoot` = `/summon with motion`

18. `/smite, /thunder = /summon minecraft:lightning-bolt`
19. `/stats, /statsedit` = all stats files are located in `world/stats`
20. `/tree` = use `tree brush` from `world-edit` mod
21. `/unbreakable = /enchant player minecraft:unbreaking`
22. `/item_maker = /give`
23. `/replaceblock = /setblock`
24. chunks loading/unloading commands = These commands will not bring performance improvements.
25. `/note = /attachment get note`
26. `/...all = /foreach ...`
27. `/cuff` = use `anti-build` module with `string` permission
28. `/cheque` = use `command-attachment` module to make `magic item`

7.2 Command Generator

You can generate complex and powerful commands using:

1. <https://www.digminecraft.com/generators/>
2. <https://minecraft.tools/en/public-gallery.php>

Chapter 8

Development

8.1 Setup the development environment

1. Clone the source: `git clone https://github.com/sakurawald/fuji-fabric.git`
2. Change the working-directory: `cd fuji-fabric`
3. Compile the source: `./gradlew build`

Chapter 9

Suggestion

9.1 Suggestion on server-side mods

9.1.1 Explanation

Here is some mods that existing and recommended to use, which can make your life easier in fabric server-side crafting.

Note that fuji doesn't require these mods installed to work, and some of these mods have the same functionality as fuji.

If you want to taste something different, and if some of the mods provides a better experience than fuji, then you can just disable the module in fuji, and go use it.

Tip: Decide according to your situation

The suggestion is personal, you can make decision according to your preference.

9.1.2 Server-side mode list

1. [skin restorer](#)
2. [head index](#)
3. [carpet](#)
4. [ledger](#)
5. [styled chat](#)
6. [styled nickname](#)
7. [styled sidebar](#)
8. [styled player list](#)
9. [universal shop](#)
10. [image2map](#)
11. [inv view](#)
12. [universal graves](#)
13. [sit](#)
14. [lithium](#)
15. [krypton](#)
16. [luckperms](#)
17. [spark](#)
18. [enclosure](#)
19. [essential commands](#)
20. [world-edit](#)
21. [chunky](#)
22. [tab](#)
23. [blue map](#)
24. [mini motd](#)
25. [husk homes](#)