

IISE Transactions L^AT_EX Template

John Doe ^a and Jane Roe ^b

^a Department, University, City, Country

^b Department, University, City, Country

Abstract



Keywords: *IISE Transactions*; L^AT_EX; Manuscript format; Taylor & Francis.

Contents

1	Documentation conventions	7
2	Introduction	8
2.1	Destiny	8
2.2	Response	8
2.3	Operation	9
2.4	Arguments	9
2.4.1	Character	9
2.4.2	Integer	9
2.4.3	Boolean	9
2.4.4	Float	10
2.4.5	String	10
2.4.6	Array	10
2.4.7	File	11
2.4.8	Server type	12
2.4.9	Block	12
2.4.10	Item	13
3	Server manager petition	14
3.1	Start server operation	14
3.1.1	Maps	16
3.1.2	Plugins	17
3.1.3	Server version	18
3.1.4	Config files	18
3.2	Server started notification	18
3.3	Error notification	19
4	Server petition	20
4.1	Server petition group	20
4.2	Server petition operation	21
4.3	Base operations	21

4.3.1	Server stop operation	22
4.3.2	Server stopped notification	22
4.3.3	Server started notification	22
4.3.4	Whitelist player operation	23
4.3.5	OP player operation	23
4.3.6	Error notification	24
4.4	Performance operations	24
4.5	WorldGuard operations	24
4.6	Residence operations	24
5	? petition	25
5.1	<i>Subsection heading 3.1</i>	25
5.1.1	<i>Sub-subsection heading 3.1.1</i>	25
5.2	<i>Subsection heading 3.2</i>	25
5.3	<i>Subsection heading 3.3</i>	25
6	Revision history	26
A	Blocks	27
A.1	Material modifiers	27
A.1.1	Unused modifiers	27
A.1.2	Age	29
A.1.3	Attachment	30
A.1.4	Axis	30
A.1.5	Berries	32
A.1.6	Bites	33
A.1.7	Candles	33
A.1.8	Charges	33
A.1.9	Conditional	33
A.1.10	Delay	33
A.1.11	Down	34
A.1.12	North, South, East and West	34

A.1.13 Up	36
A.1.14 Eggs	37
A.1.15 Extended	37
A.1.16 Waterlogged	45
A.2 Material modifiers concatenation	54
References	57

Figures

2.1	Packet structure	8
2.2	True packet with the LSB at 1	9
2.3	True packet with all bits at 1	10
2.4	Structure of a String	10
2.5	Example of a string array	11
2.6	File structure	12
2.7	Structure of a Block	12
3.1	Server manager petition structure	14
3.2	Start server petition structure	15
3.3	Start server response structure	16
3.4	Start server error response structure	16
3.5	Usual plugin structure	17
3.6	File plugin structure	18
3.7	Server started notification structure	19
3.8	Error notification structure	19
4.1	Server petition structure	20
4.2	Implemented group response structure	21
4.3	Stop server operation structure	22
4.4	Server stopped response structure	22
4.5	Server started response structure	23
4.6	Whitelist player operation structure	23
4.7	OP player operation structure	23
A.1a	Modifier concatenation	55
A.1b	Integer modifier concatenation	56

Tables

2.1	DST bits meaning	8
2.2	Block enum	13
4.1	Extended types	21
6.1	Revision history	26
A.1	Unused Spigot BlockData's modifiers	28
A.2	Unused Spigot BlockData's modifiers on certain blocks	29
A.3	Ageable materials	30
A.4	Attachable materials	30
A.5	Orientable materials	32
A.6	Orientable materials	36
A.7	Materials with up option	37
A.8	Waterlogged materials	54

1 Documentation conventions

..

abbreviations

2 Introduction

explicar los distintos protocolos que se hablaron a continuacion

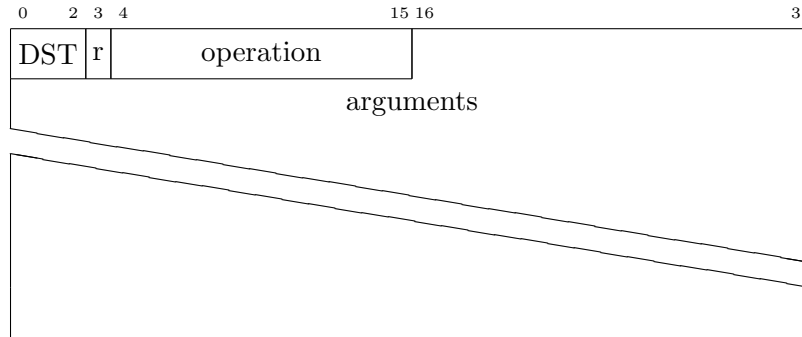


Figure 2.1: Packet structure

2.1 Destiny

explain

reference to the interconnected blocks

DST[2]	DST[1]	DST[0]	Destination
0	0	0	ServerManagerPetition
0	0	1	ServerPetition
0	1	0	ClientConnectorPetition
0	1	1	ClientPetition
1	X	X	<i>Reserved</i>

Table 2.1: DST bits meaning

2.2 Response

Some of the petitions have return objects. Those petitions will return to the sender (Tester-Connector) with the same code, but with a '1' on the Response parameter. In that case, the parameter Destiny now means 'Origin'.

Some petitions have async "returns" (for example: examples). Those will be sent using petitions without return's operations (so, petitions without a mirror petition with a '1' as Response), marked as responses (Response bit at '1').

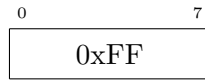


Figure 2.3: True packet with all bits at 1

2.4.4 Float

Floats are 4-bytes floating-point numbers. They are represented following the IEEE 754¹.

2.4.5 String

Strings are arrays of characters. Refer to the respective subsections for more information.

2.4.6 Array

Arrays are a set of n elements of the same type.

The structure is a 2-byte **first (0..7) MSB, then (8..15) LSB** integer (representing the number of elements, n), followed by n elements of the same type. As a note here, by representing the size with a 2-byte integer the maximum number of elements per array is 65,535.

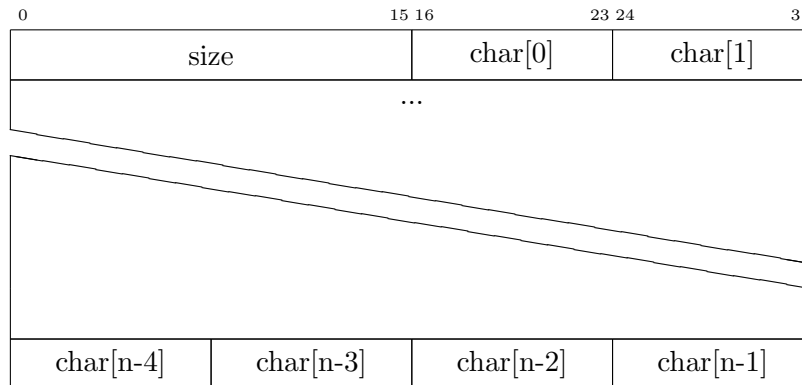


Figure 2.4: Structure of a String

Arrays can be multidimensional, holding n arrays of the same type. It's worth mentioning that they don't have to be arrays of the same length, as can be seen in Figure 2.5, Example of a string array.

¹This standard should be used by C, Java and Python. **cite?**

0		15 16		23 24		31	
2 [number of arrays]			5 [str[0]’s length]				
h		e		l		l	
o		6 [str[1]’s length]				w	
o		r		l		d	
!		next type					

Figure 2.5: Example of a string array

2.4.7 File

Similar to the Array, a File is a name (String), followed by a group of bytes.

The problem here is that if we stick with the Array structure, the maximum size of a file will be around 8kB. To solve this, the File structure implements some kind of 'extended array', that extends the 'size' parameter to 32 bits. That way, the file size restriction by protocol definition² is 4GB.

²Besides defining here what's allowed, remember that this packet will be inside a TCP payload **definition?**. This means that the maximum file size will be probably redefined by the machine's TCP firewalls.

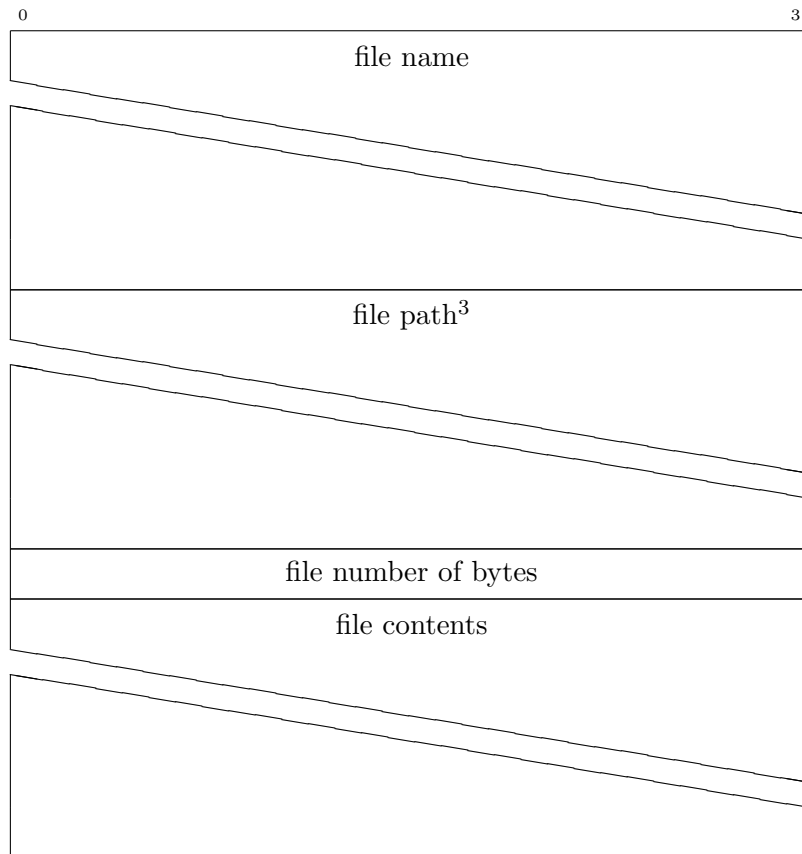


Figure 2.6: File structure

2.4.8 Server type

The Server type specifies the Minecraft server.

As a standard, we only support Spigot (*Spigot* (n.d.)) and Paper (*PaperMC* (n.d.)), but for scalability reasons this parameter is a String specifying the server type.

2.4.9 Block

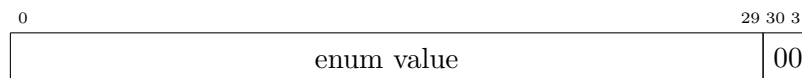


Figure 2.7: Structure of a Block

³The path must be relative, and you can't go outside the Server directory (using '../'). Both " and './' means the root of the Server directory.

unsigned 4-bytes integer. 2MSB forced at 00 (01, 10 and 11 reserved for Complex/Basic Blocks (if made)), others as Enum value

Enum value	Block name	First Minecraft version
0	AIR	1.8

Table 2.2: Block enum

2.4.10 Item

...

3 Server manager petition

...

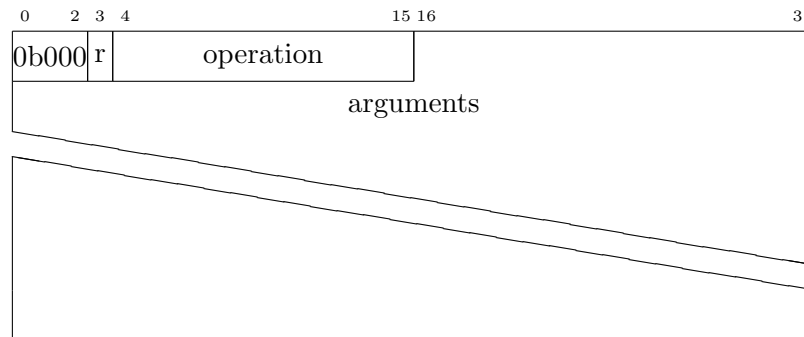


Figure 3.1: Server manager petition structure

Table of operations

You don't have to implement the NOP operation in this destiny block because the timeout happens inside the Server petition block. That is, if you don't call operations (or send NOPs) to the Server petition for a long time, the server will stop, and because the server stopped the Server manager will close the established connection.

3.1 Start server operation

...

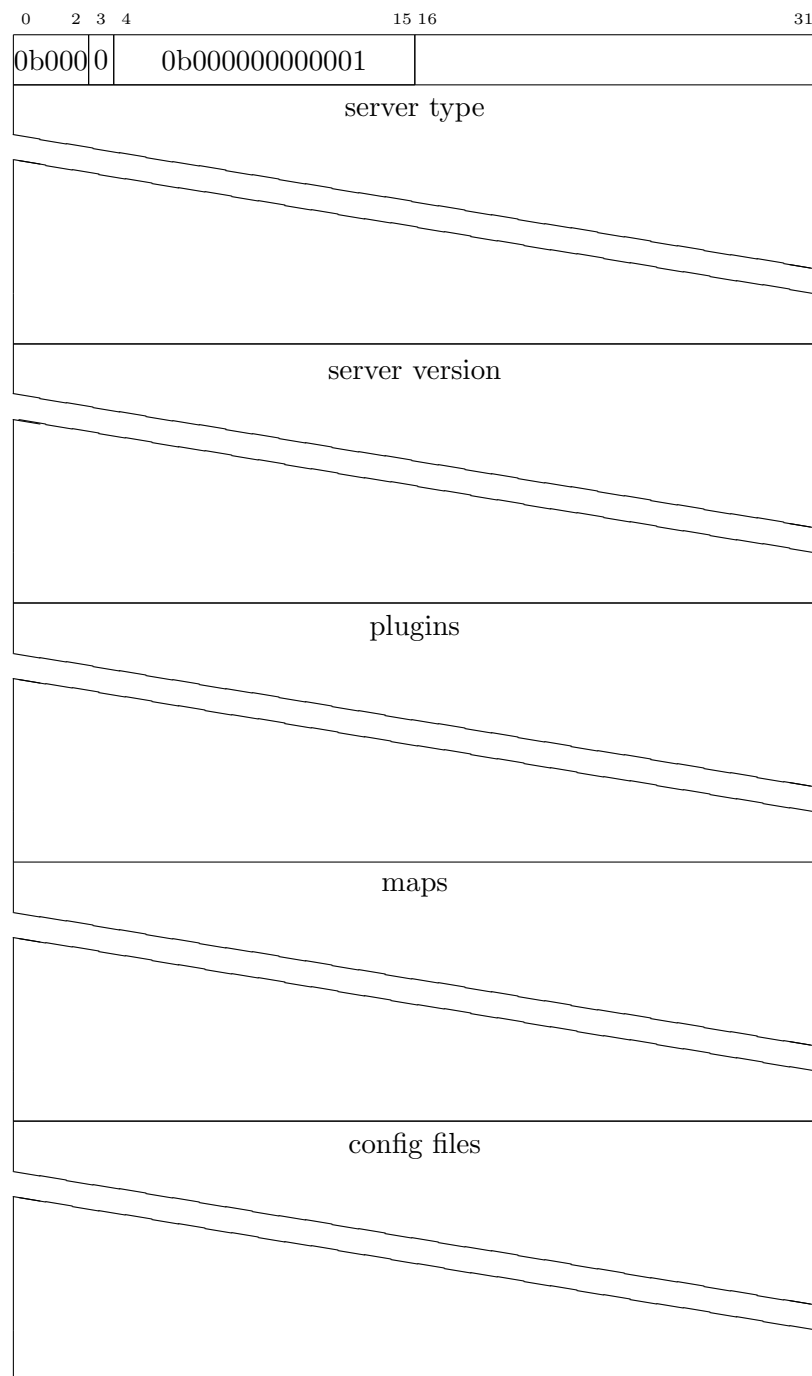


Figure 3.2: Start server petition structure

Once a 'start server' request is received the program should create a server with the specified arguments, and return its IP:Port (for example, '127.0.0.1:25565', a 15-characters string; see Figure 3.3, Start server response structure). The IP to send the Server Petitions is the same, but the next port (IP:<port+1>).

If it's not possible to create it (for example: one argument is invalid, the user sent a plugin when it's specified that only Usual Plugins are allowed **explain**, or there's no free servers of that type), then an empty IP is returned (see Figure 3.4, Start server error response structure).

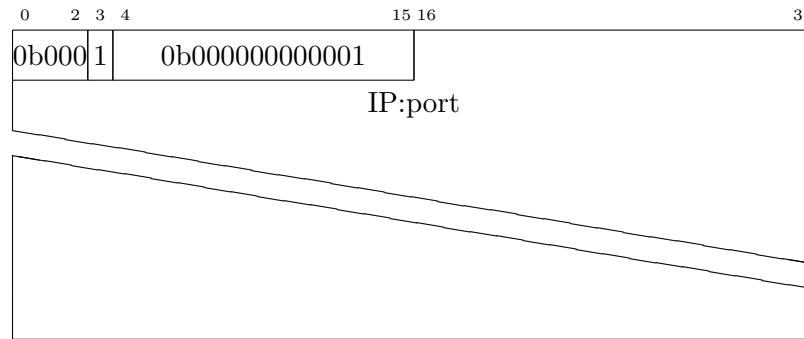


Figure 3.3: Start server response structure

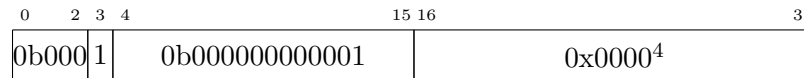


Figure 3.4: Start server error response structure

3.1.1 Maps

Array of maps (worlds; Map[]). To have more information about arrays check the subsection 2.4.6, Array.

About the Map type, Minecraft is divided on different worlds (*World - Minecraft Wiki* (n.d.)). By default there's only three, but with some plugins this number can increase.

In order to properly test some plugins, there may be needed some kind of known place. To avoid overusing the Set block operation **link** you can send using this argument your(s) world(s).

Map in more detail

⁴Being the argument an array, the first 2 bytes specifies its size. As we must return an empty array, the argument should be exactly 16 zeroes.

3.1.2 Plugins

Array of plugins (Plugin[]). To have more information check the subsection 2.4.6, Array.

About the Plugin type, there's three types of plugins:

1. Usual plugins

The Usual plugins are plugins that you expect everyone to have for being extremely common, like WorldGuard (*WorldGuard* (n.d.)), or to allow the user to test plugins with Premium plugins⁵ dependencies. This allows both security and performance.

Something to highlight is the fact that, as mentioned in the operation Allows non usual plugins [reference](#), some ServerManager will only allow plugins that are already in the machine.

As can be seen in the Figure 3.5, Usual plugin structure, the first argument (that specifies the Plugin type) is 0x00.

The plugin version is optional, and can't be specified in the parameter *name*. If no version is provided (an empty string) then the Server Manager will pick the plugin with the highest version that is compatible with the desired server version.

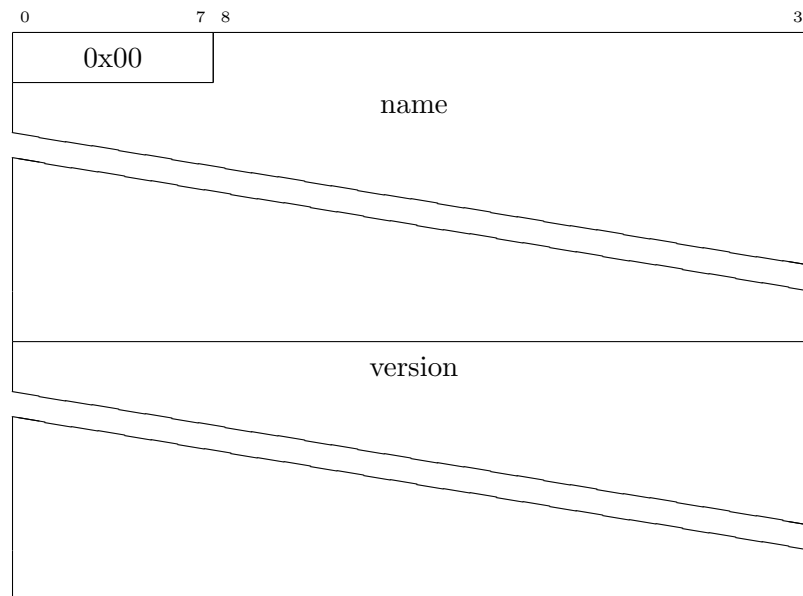


Figure 3.5: Usual plugin structure

⁵Premium plugins are paid plugins. For that reason, only the purchaser can download them (so you can't send a link to the plugin), and sending them through the internet via file upload may not be legal, so the plugin must be already downloaded in the machine.

2. Uploaded plugins

The Uploaded plugins are plugins available in some website, thus can be sent through an URL.

structure?

3. File plugins

File plugins are plugins that are non-usual and aren't uploaded in any website, so they must be sent as a file.

As can be seen in the Figure 3.6, File plugin structure, the first argument (that specifies the Plugin type) is 0x02.

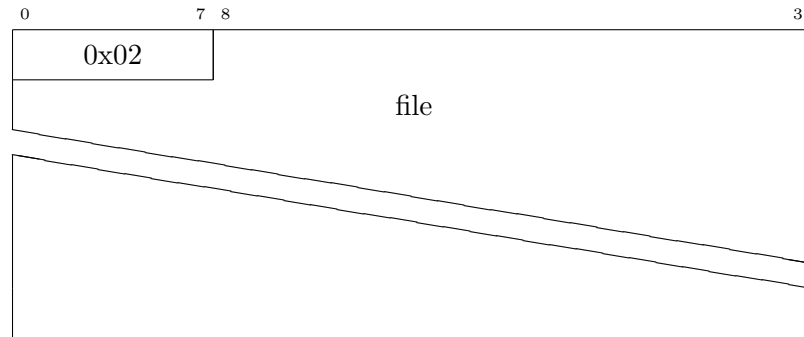


Figure 3.6: File plugin structure

mixed plugin types example?

3.1.3 Server version

String specifying the server type's version. For example, '1.12.2'.

3.1.4 Config files

...

3.2 Server started notification

After a Start server operation the server will start. Due to the unpredictable amount of time that the server takes to start up you'll receive a Server started notification once the server socket is available.

You may notice that there's another Server started notification under the Server petition section. That notification goes to the ServerManager ref?, while this goes to the Tester ref?. Also, the Server one have a token that is only shared between Server and the ServerManager, and the Tester doesn't have to know it too.

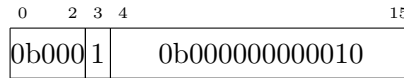


Figure 3.7: Server started notification structure

3.3 Error notification

...

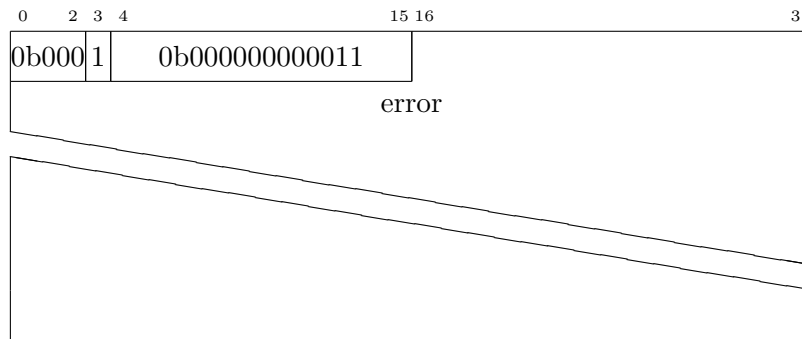


Figure 3.8: Error notification structure

4 Server petition

...

The server petitions are a bit different from the rest. The server petitions are designed in a way that everyone have some common operations, and then you can add some others optionally (and even non-standard ones). We'll define this 'set of operations' as groups.

For that reason, the operation field (defined on the Figure 2.1, Packet structure) becomes the group, and then the operation is defined on the next 2 bytes, as shown in the Figure 4.1, Server petition structure.

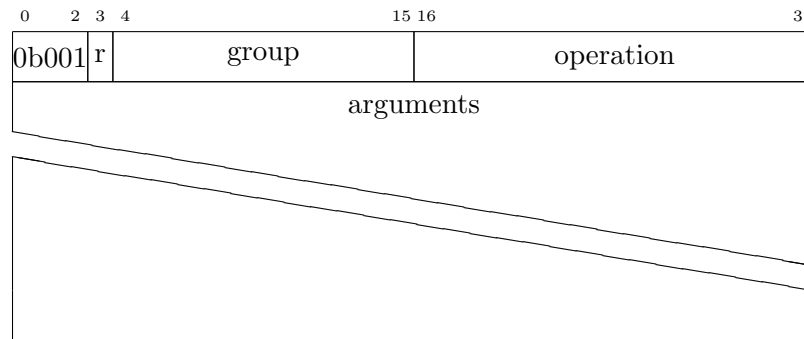


Figure 4.1: Server petition structure

4.1 Server petition group

The group tells which kind of petitions we're talking about.

The MSB **abbreviation?** tells if the group is one of the standards, thus must be followed by specification, or if it's non-standard, so the petition can be whatever the user want it to be. This is useful if you want to implement a petition not followed by the standard, or if the petition only makes sense in your personal environment.

The 0b00000000000001 group represents the 'base group'. This group implements some basic operations, and must be implemented. All the others are optional.

type[15]	type[14..4]	Extended type
0	0b000000000000	NOP ⁶
0	0b000000000001	Base operations
0	0b000000000010	Performance operations
0	0b000000000011	WorldGuard operations
0	0b000000000100	Residence operations
1	XXXXXXXXXXXX	Reserved for internal use

4.3.1 Server stop operation

...

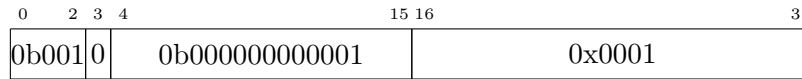


Figure 4.3: Stop server operation structure

4.3.2 Server stopped notification

... response to...

To have more information about the *server id* parameter check the Subsection 4.3.3, Server started notification.

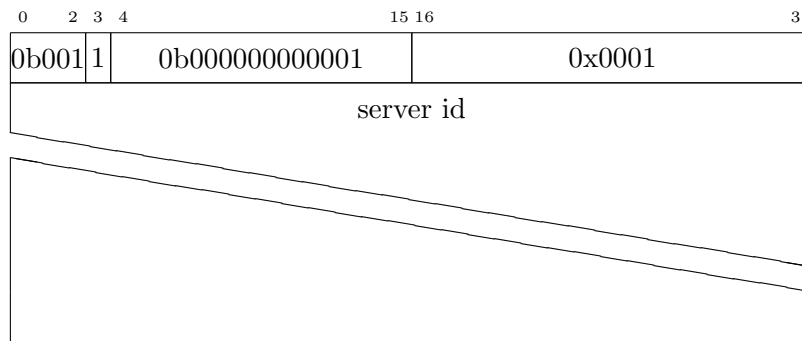


Figure 4.4: Server stopped response structure

4.3.3 Server started notification

This notification is sent to the Server Manager [ref?](#), as a response for the Start server operation, thus not really a response of a Server's operation.

As one IP can have multiple servers, a string that identifies the server must be sent with the response. This argument can be whatever you want (for example, <server ip>:<server port> will be unique), but must be shared between both the Server Manager and the Server. For security reasons [cite IP spoofing or similar](#) (because the Tester [ref?](#) also knows the server's IP and port) a hash function is encouraged to be used.

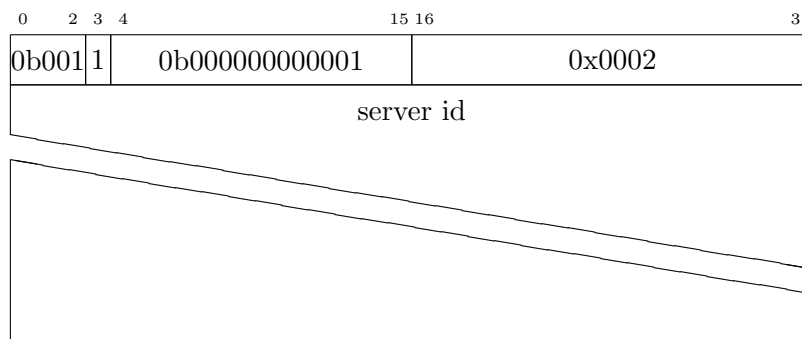


Figure 4.5: Server started response structure

4.3.4 Whitelist player operation

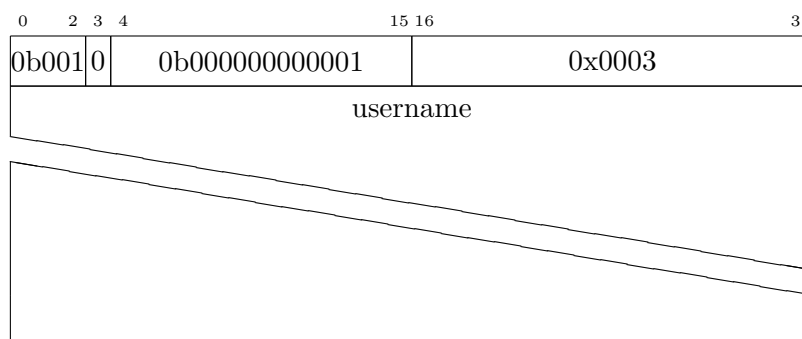


Figure 4.6: Whitelist player operation structure

4.3.5 OP player operation

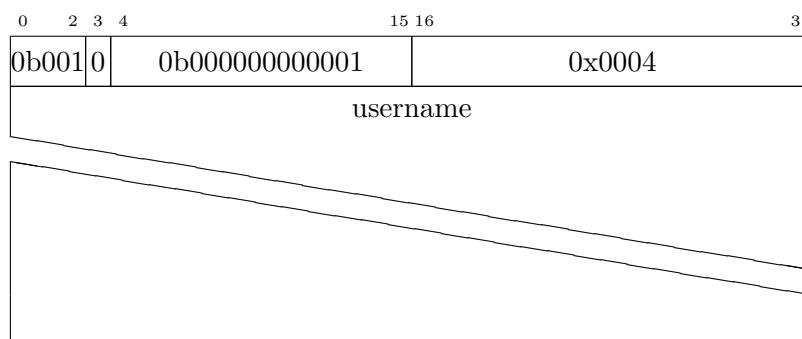


Figure 4.7: OP player operation structure

4.3.6 Error notification



4.4 Performance operations



4.5 WorldGuard operations



4.6 Residence operations



5 ? petition

First-level headings should be in bold.

5.1 *Subsection heading 3.1*

Second-level headings should be in bold italics.

5.1.1 *Sub-subsection heading 3.1.1*

Third-level headings should be in italics.

5.2 *Subsection heading 3.2*

5.3 *Subsection heading 3.3*

6 Revision history

Date	Revision	Changes
date	1	Initial release.

Table 6.1: Revision history

A Blocks

To generate the blocks enum Spigot 1.19 was used. That means that all the block names *should* be the exact same as ?.

A.1 Material modifiers

There's one downside on using Spigot's Material: it doesn't describes perfectly the block. In some aspects it will, for example, distinguish between wood types, but it won't differentiate between a wooden stair and a wooden stair with water.

That's why there's some prefixes and suffixes (that will be discussed in the following subsections) surrounding the original Spigot name, to make every possible Minecraft block combination appear in the block enum. Just to clarify, all the block modifiers has also been extracted from Spigot (all ?'s subinterfaces).

A.1.1 Unused modifiers

There's some Spigot modifiers that beside existing it won't be imported because there aren't a distinguished block in their own. You can find those in Figure A.1, Unused Spigot BlockData's modifiers.

Modifier name	Reason for discarding
has_bottle_X	Inventory dependent
has_record	Inventory dependent
enabled	Adjacent redstone dependent
triggered	Adjacent redstone dependent
instrument	Bottom-block dependent
occupied	Entity dependent
persistent	Admin block
unstable	Admin block
distance	Block dependent
stage	Same block
short	Tick dependent

Modifier name	Reason for discarding
attached	Block dependent
disarmed	Block dependent
power	Block/event dependent
tilt	Entity dependent
can_summon	Admin block
shrieking	Entity dependent
bloom	Admin block
bottom	Bottom-block dependent
powered	Admin block / block dependent
has_book	Inventory dependent
sculk_sensor_phase	Admin block
signal_fire	Bottom-block dependent
north=tall	Top-block dependent
south=tall	Top-block dependent
east=tall	Top-block dependent
west=tall	Top-block dependent

Table A.1: Unused Spigot BlockData’s modifiers

In addition to this, some modifiers applied to certain blocks doesn’t change the block itself. Those are mentioned in Figure A.2, Unused Spigot BlockData’s modifiers on certain blocks.

Block name	Modifier name
CAVE_VINES	age
CACTUS	age
FIRE	age
KELP	age
SUGAR_CANE	age
MANGROVE_PROPAGULE	age
TWISTING_VINES	age

Block name	Modifier name
WEeping_VINES	age
ANDESITE_WALL	up
BLACKSTONE_WALL	up
BRICK_WALL	up
COBBLED_DEEPSLATE_WALL	up
COBBLESTONE_WALL	up
DEEPSLATE_BRICK_WALL	up
DEEPSLATE_TILE_WALL	up
DIORITE_WALL	up
END_STONE_BRICK_WALL	up
GRANITE_WALL	up
MOSSY_COBBLESTONE_WALL	up
MOSSY_STONE_BRICK_WALL	up
MUD_BRICK_WALL	up
NETHER_BRICK_WALL	up
POLISHED_BLACKSTONE_BRICK_WALL	up
POLISHED_BLACKSTONE_WALL	up
POLISHED_DEEPSLATE_WALL	up
PRISMARINE_WALL	up
RED_NETHER_BRICK_WALL	up
RED_SANDSTONE_WALL	up
SANDSTONE_WALL	up
STONE_BRICK_WALL	up

Table A.2: Unused Spigot BlockData’s modifiers on certain blocks

A.1.2 Age

Represents the different growth stages that a crop-like block can go through.

Defaults to 0.

Material	Age range
BEETROOTS	0-3
BAMBOO	0-1
CARROTS	0-7
CHORUS_FLOWER	0/5 ⁸
COCOA	0-2
FROSTED_ICE	0-3
MELON_STEM	0-7
NETHER_WART	0-3
POTATOES	0-7
PUMPKIN_STEM	0-7
SWEET_BERRY_BUSH	0-3
WHEAT	0-7

Table A.3: Ageable materials

A.1.3 Attachment

Denotes how the bell is attached to its block.

Defaults to floor.

Material	Options
BELL	ceiling/double_wall/floor/single_wall

Table A.4: Attachable materials

A.1.4 Axis

Represents the axis along whilst this block is oriented.

Except for NETHER_PORTAL (which defaults to x), it defaults to y.

⁸The block is the same from age 0 to 4, and it changes in age 5. That's why age=5 is considered as age=1, and age=0-4 as age=0, as you may notice in Figure A.1, Modifier concatenation.

Material	Age range
NETHER_PORTAL	x/z
ACACIA_LOG	x/y/z
ACACIA_WOOD	x/y/z
BASALT	x/y/z
BIRCH_LOG	x/y/z
BIRCH_WOOD	x/y/z
BONE_BLOCK	x/y/z
CHAIN	x/y/z
CRIMSON_HYPHAE	x/y/z
CRIMSON_STEM	x/y/z
DARK_OAK_LOG	x/y/z
DARK_OAK_WOOD	x/y/z
DEEPSLATE	x/y/z
HAY_BLOCK	x/y/z
INFESTED_DEEPSLATE	x/y/z
JUNGLE_LOG	x/y/z
JUNGLE_WOOD	x/y/z
MANGROVE_LOG	x/y/z
MANGROVE_WOOD	x/y/z
MUDDY_MANGROVE_ROOTS	x/y/z
OAK_LOG	x/y/z
OAK_WOOD	x/y/z
OCHRE_FROGLIGHT	x/y/z
PEARLESCENT_FROGLIGHT	x/y/z
POLISHED_BASALT	x/y/z
PURPUR_PILLAR	x/y/z
QUARTZ_PILLAR	x/y/z
SPRUCE_LOG	x/y/z
SPRUCE_WOOD	x/y/z
STRIPPED_ACACIA_LOG	x/y/z

Material	Age range
STRIPPED_ACACIA_WOOD	x/y/z
STRIPPED_BIRCH_LOG	x/y/z
STRIPPED_BIRCH_WOOD	x/y/z
STRIPPED_CRIMSON_HYPHAE	x/y/z
STRIPPED_CRIMSON_STEM	x/y/z
STRIPPED_DARK_OAK_LOG	x/y/z
STRIPPED_DARK_OAK_WOOD	x/y/z
STRIPPED_JUNGLE_LOG	x/y/z
STRIPPED_JUNGLE_WOOD	x/y/z
STRIPPED_MANGROVE_LOG	x/y/z
STRIPPED_MANGROVE_WOOD	x/y/z
STRIPPED_OAK_LOG	x/y/z
STRIPPED_OAK_WOOD	x/y/z
STRIPPED_SPRUCE_LOG	x/y/z
STRIPPED_SPRUCE_WOOD	x/y/z
STRIPPED_WARPED_HYPHAE	x/y/z
STRIPPED_WARPED_STEM	x/y/z
VERDANT_FROGLIGHT	x/y/z
WARPED_HYPHAE	x/y/z
WARPED_STEM	x/y/z

Table A.5: Orientable materials

A.1.5 Berries

Indicates whether the block has berries.

Defaults to false.

CAVE_VINES true/false CAVE_VINES_PLANT true/false

A.1.6 Bites

Represents the amount of bites which have been taken from this slice of cake.

Defaults to 0.

CAKE 0-6

A.1.7 Candles

Represents the number of candles which are present.

Defaults to 1.

BLACK_CANDLE 1-4 BLUE_CANDLE 1-4 BROWN_CANDLE 1-4 CANDLE 1-4 CYAN_CANDLE
1-4 GRAY_CANDLE 1-4 GREEN_CANDLE 1-4 LIGHT_BLUE_CANDLE 1-4 LIGHT_GRAY_CANDLE
1-4 LIME_CANDLE 1-4 MAGENTA_CANDLE 1-4 ORANGE_CANDLE 1-4 PINK_CANDLE
1-4 PURPLE_CANDLE 1-4 RED_CANDLE 1-4 WHITE_CANDLE 1-4 YELLOW_CANDLE
1-4

A.1.8 Charges

Represents the amount of times the anchor may still be used.

Defaults to 0.

RESPAWN_ANCHOR 0-4

A.1.9 Conditional

Denotes whether this command block is conditional or not.

Defaults to false.

CHAIN_COMMAND_BLOCK true/false COMMAND_BLOCK true/false REPEATING_COMMAND_BLOCK
true/false

A.1.10 Delay

Propagation delay of a repeater.

Defaults to 1.

REPEATER 1-4

A.1.11 Down

Set which faces of the block textures are displayed on.

Except for BROWN_MUSHROOM_BLOCK, MUSHROOM_STEM and RED_MUSHROOM_BLOCK (which defaults to true), it defaults to false.

CHORUS_PLANT true/false GLOW_LICHEN true/false SCULK_VEIN true/false BROWN_MUSHROOM true/false MUSHROOM_STEM true/false RED_MUSHROOM_BLOCK true/false

A.1.12 North, South, East and West

Set which faces of the block textures are displayed on.

As the *tall* option is unused (check Table A.1, Unused Spigot BlockData's modifiers), *none* and *low* will be considered as *false* and *true*, respectively.

Material	Options (default on bold)
ACACIA_FENCE	true/ false
BIRCH_FENCE	true/ false
BLACK_STAINED_GLASS_PANE	true/ false
BLUE_STAINED_GLASS_PANE	true/ false
BROWN_STAINED_GLASS_PANE	true/ false
CHORUS_PLANT	true/ false
CRIMSON_FENCE	true/ false
CYAN_STAINED_GLASS_PANE	true/ false
DARK_OAK_FENCE	true/ false
FIRE	true/ false
GLASS_PANE	true/ false
GLOW_LICHEN	true/ false
GRAY_STAINED_GLASS_PANE	true/ false
GREEN_STAINED_GLASS_PANE	true/ false
IRON_BARS	true/ false
JUNGLE_FENCE	true/ false
LIGHT_BLUE_STAINED_GLASS_PANE	true/ false
LIGHT_GRAY_STAINED_GLASS_PANE	true/ false

Material	Options (default on bold)
LIME_STAINED_GLASS_PANE	true/ false
MAGENTA_STAINED_GLASS_PANE	true/ false
MANGROVE_FENCE	true/ false
NETHER_BRICK_FENCE	true/ false
OAK_FENCE	true/ false
ORANGE_STAINED_GLASS_PANE	true/ false
PINK_STAINED_GLASS_PANE	true/ false
PURPLE_STAINED_GLASS_PANE	true/ false
RED_STAINED_GLASS_PANE	true/ false
SCULK_VEIN	true/ false
SPRUCE_FENCE	true/ false
TRIPWIRE	true/ false
VINE	true/ false
WARPED_FENCE	true/ false
WHITE_STAINED_GLASS_PANE	true/ false
YELLOW_STAINED_GLASS_PANE	true/ false
BROWN_MUSHROOM_BLOCK	true /false
MUSHROOM_STEM	true /false
RED_MUSHROOM_BLOCK	true /false
ANDESITE_WALL	none /low/tall
BLACKSTONE_WALL	none /low/tall
BRICK_WALL	none /low/tall
COBBLED_DEEPSLATE_WALL	none /low/tall
COBBLESTONE_WALL	none /low/tall
DEEPSLATE_BRICK_WALL	none /low/tall
DEEPSLATE_TILE_WALL	none /low/tall
DIORITE_WALL	none /low/tall
END_STONE_BRICK_WALL	none /low/tall
GRANITE_WALL	none /low/tall
MOSSY_COBBLESTONE_WALL	none /low/tall

Material	Options (default on bold)
MOSSY_STONE_BRICK_WALL	none /low/tall
MUD_BRICK_WALL	none /low/tall
NETHER_BRICK_WALL	none /low/tall
POLISHED_BLACKSTONE_BRICK_WALL	none /low/tall
POLISHED_BLACKSTONE_WALL	none /low/tall
POLISHED_DEEPSLATE_WALL	none /low/tall
PRISMARINE_WALL	none /low/tall
REDSTONE_WIRE	none /low/tall
RED_NETHER_BRICK_WALL	none /low/tall
RED_SANDSTONE_WALL	none /low/tall
SANDSTONE_WALL	none /low/tall
STONE_BRICK_WALL	none /low/tall

Table A.6: Orientable materials

A.1.13 Up

Set which faces of the block textures are displayed on.

Except for CHORUS_PLANT, FIRE, GLOW_LICHEN, SCULK_VEIN and VINE (which defaults to false), it defaults to true.

Material	Options
CHORUS_PLANT	true/false
FIRE	true/false
GLOW_LICHEN	true/false
SCULK_VEIN	true/false
VINE	true/false
BROWN_MUSHROOM_BLOCK	true/false
MUSHROOM_STEM	true/false
RED_MUSHROOM_BLOCK	true/false

Material	Options
----------	---------

Table A.7: Materials with up option

A.1.14 Eggs

Defaults to 1.

TURTLE_EGG 1-4

A.1.15 Extended

Defaults to false.

extended=false (PISTON) extended=false (STICKY_PISTON)

eye=false (END_PORTAL_FRAME)

face=wall(ACACIA_BUTTON)face = wall(BIRCH_BUTTON)face = wall(CRIMSON_BUTTON)face =
wall(DARK_OAK_BUTTON)face = wall(GRINDSTONE)face = wall(JUNGLE_BUTTON)face =
wall(LEVER)face = wall(MANGROVE_BUTTON)face = wall(OAK_BUTTON)face =
wall(POLISHED_BLACKSTONE_BUTTON)face = wall(SPRUCE_BUTTON)face =
wall(STONE_BUTTON)face = wall(WARPED_BUTTON)

facing=down (HOPPER) facing=north(ACACIA_BUTTON)facing = north(ACACIA_DOOR)facing =
north(ACACIA_FENCE_GATE)facing = north(ACACIA_STAIRS)facing = north(ACACIA_TRAPDOOR)
north(ACACIA_WALL_SIGN)facing = north(ANDESITE_STAIRS)facing = north(ANVIL)facing =
north(ATTACHED_MELON_STEM)facing = north(ATTACHED_PUMPKIN_STEM)facing =
north(BARREL)facing = north(BEEHIVE)facing = north(BEE_NEST)facing =
north(BELL)facing = north(BIG_DRIPLEAF)facing = north(BIG_DRIPLEAF_STEM)facing =
north(BIRCH_BUTTON)facing = north(BIRCH_DOOR)facing = north(BIRCH_FENCE_GATE)facing =
north(BIRCH_STAIRS)facing = north(BIRCH_TRAPDOOR)facing = north(BIRCH_WALL_SIGN)facing =
north(BLACKSTONE_STAIRS)facing = north(BLACK_BED)facing = north(BLACK_GLAZED_TERRACOTTA)
north(BLACK_WALL_BANNER)facing = north(BLAST_FURNACE)facing = north(BLUE_BED)facing =
north(BLUE_GLAZED_TERRACOTTA)facing = north(BLUE_WALL_BANNER)facing =
north(BRAIN_CORAL_WALL_FAN)facing = north(BRICK_STAIRS)facing = north(BROWN_BED)facing =
north(BROWN_GLAZED_TERRACOTTA)facing = north(BROWN_WALL_BANNER)facing =

$north(BUBBLE_{C}ORAL_{W}ALL_{F}AN)_{facing} = north(CAMPFIRE)_{facing} = north(CARVED_{P}UMP_{K}IN)_{facing}$
 $north(CHAIN_{C}OMMAND_{B}LOCK)_{facing} = north(CHEST)_{facing} = north(CHIPPED_{A}NVIL)_{facing}$
 $north(COBBLED_{D}EEPSLATE_{S}TAIRS)_{facing} = north(COBBLESTONE_{S}TAIRS)_{facing} =$
 $north(COCOA)_{facing} = north(COMMAND_{B}LOCK)_{facing} = north(COMPARATOR)_{facing} =$
 $north(CREEPER_{W}ALL_{H}EAD)_{facing} = north(CRIMSON_{B}UTTON)_{facing} = north(CRIMSON_{D}OOR)_{facing}$
 $north(CRIMSON_{F}ENCE_{G}ATE)_{facing} = north(CRIMSON_{S}TAIRS)_{facing} = north(CRIMSON_{T}RAPDOOR)_{facing}$
 $north(CRIMSON_{W}ALL_{S}IGN)_{facing} = north(CUT_{C}OPPER_{S}TAIRS)_{facing} = north(CYAN_{B}ED)_{facing}$
 $north(CYAN_{G}LAZED_{T}ERRACOTTA)_{facing} = north(CYAN_{W}ALL_{B}ANNER)_{facing} =$
 $north(DAMAGED_{A}NVIL)_{facing} = north(DARK_{O}AK_{B}UTTON)_{facing} = north(DARK_{O}AK_{D}OOR)_{facing}$
 $north(DARK_{O}AK_{F}ENCE_{G}ATE)_{facing} = north(DARK_{O}AK_{S}TAIRS)_{facing} = north(DARK_{O}AK_{T}RAPDOOR)_{facing}$
 $north(DARK_{O}AK_{W}ALL_{S}IGN)_{facing} = north(DARK_{P}RISMARINE_{S}TAIRS)_{facing} =$
 $north(DEAD_{B}RAIN_{C}ORAL_{W}ALL_{F}AN)_{facing} = north(DEAD_{B}UBBLE_{C}ORAL_{W}ALL_{F}AN)_{facing} =$
 $north(DEAD_{F}IRE_{C}ORAL_{W}ALL_{F}AN)_{facing} = north(DEAD_{H}ORN_{C}ORAL_{W}ALL_{F}AN)_{facing} =$
 $north(DEAD_{T}UBE_{C}ORAL_{W}ALL_{F}AN)_{facing} = north(DEEPSLATE_{B}RICK_{S}TAIRS)_{facing} =$
 $north(DEEPSLATE_{T}ILE_{S}TAIRS)_{facing} = north(DIORITE_{S}TAIRS)_{facing} = north(DISPENSER)_{facing}$
 $north(DRAGON_{W}ALL_{H}EAD)_{facing} = north(DROPPER)_{facing} = north(ENDER_{C}HEST)_{facing} =$
 $north(END_{P}ORTAL_{F}RAME)_{facing} = north(END_{S}TONE_{B}RICK_{S}TAIRS)_{facing} =$
 $north(EXPOSED_{C}UT_{C}OPPER_{S}TAIRS)_{facing} = north(FIRE_{C}ORAL_{W}ALL_{F}AN)_{facing} =$
 $north(FURNACE)_{facing} = north(GRANITE_{S}TAIRS)_{facing} = north(GRAY_{B}ED)_{facing} =$
 $north(GRAY_{G}LAZED_{T}ERRACOTTA)_{facing} = north(GRAY_{W}ALL_{B}ANNER)_{facing} =$
 $north(GREEN_{B}ED)_{facing} = north(GREEN_{G}LAZED_{T}ERRACOTTA)_{facing} = north(GREEN_{W}ALL_{B}ANNER)_{facing}$
 $north(GRINDSTONE)_{facing} = north(HORN_{C}ORAL_{W}ALL_{F}AN)_{facing} = north(IRON_{D}OOR)_{facing} =$
 $north(IRON_{T}RAPDOOR)_{facing} = north(JACK_{O}LANTERN)_{facing} = north(JUNGLE_{B}UTTON)_{facing}$
 $north(JUNGLE_{D}OOR)_{facing} = north(JUNGLE_{F}ENCE_{G}ATE)_{facing} = north(JUNGLE_{S}TAIRS)_{facing}$
 $north(JUNGLE_{T}RAPDOOR)_{facing} = north(JUNGLE_{W}ALL_{S}IGN)_{facing} = north(LADDER)_{facing} =$
 $north(LECTERN)_{facing} = north(LEVER)_{facing} = north(LIGHT_{B}LUE_{B}ED)_{facing} =$
 $north(LIGHT_{B}LUE_{G}LAZED_{T}ERRACOTTA)_{facing} = north(LIGHT_{B}LUE_{W}ALL_{B}ANNER)_{facing} =$
 $north(LIGHT_{G}RAY_{B}ED)_{facing} = north(LIGHT_{G}RAY_{G}LAZED_{T}ERRACOTTA)_{facing} =$
 $north(LIGHT_{G}RAY_{W}ALL_{B}ANNER)_{facing} = north(LIME_{B}ED)_{facing} = north(LIME_{G}LAZED_{T}ERRACOTTA)_{facing}$
 $north(LIME_{W}ALL_{B}ANNER)_{facing} = north(LOOM)_{facing} = north(MAGENTA_{B}ED)_{facing} =$
 $north(MAGENTA_{G}LAZED_{T}ERRACOTTA)_{facing} = north(MAGENTA_{W}ALL_{B}ANNER)_{facing} =$

$north(MANGROVE_BUTTON) facing = north(MANGROVE_DOOOR) facing = north(MANGROVE_FENCE)$
 $north(MANGROVE_STAIRS) facing = north(MANGROVE_TRAPDOOR) facing = north(MANGROVE_TUR)$
 $north(MOSSY_COBBLESTONE_STAIRS) facing = north(MOSSYSTONE_BRICK_STAIRS) facing =$
 $north(MOVING_PISTON) facing = north(MUD_BRICK_STAIRS) facing = north(NETHER_BRICK_STAIRS)$
 $north(OAK_BUTTON) facing = north(OAK_DOOOR) facing = north(OAK_FENCE_GATE) facing =$
 $north(OAK_STAIRS) facing = north(OAK_TRAPDOOR) facing = north(OAK_WALL_SIGN) facing =$
 $north(ORANGE_BED) facing = north(ORANGE_GLAZED_TERRACOTTA) facing =$
 $north(ORANGE_WALL_BANNER) facing = north(OXIDIZED_CUT_COPPER_STAIRS) facing =$
 $north(PINK_BED) facing = north(PINK_GLAZED_TERRACOTTA) facing = north(PINK_WALL_BANNER)$
 $north(PISTON) facing = north(PISTON_HEAD) facing = north(PLAYER_WALL_HEAD) facing =$
 $north(POLISHED_ANDESITE_STAIRS) facing = north(POLISHED_BLACKSTONE_BRICK_STAIRS) facing =$
 $north(POLISHED_BLACKSTONE_BUTTON) facing = north(POLISHED_BLACKSTONE_STAIRS) facing =$
 $north(POLISHED_DEEPSLATE_STAIRS) facing = north(POLISHED_DIORITE_STAIRS) facing =$
 $north(POLISHED_GRANITE_STAIRS) facing = north(PRISMARINE_BRICK_STAIRS) facing =$
 $north(PRISMARINE_STAIRS) facing = north(PURPLE_BED) facing = north(PURPLE_GLAZED_TERRACOTTA)$
 $north(PURPLE_WALL_BANNER) facing = north(PURPUR_STAIRS) facing = north(QUARTZ_STAIRS)$
 $north(REDSTONE_WALL_TORCH) facing = north(RED_BED) facing = north(RED_GLAZED_TERRACOTTA)$
 $north(RED_NETHER_BRICK_STAIRS) facing = north(RED_SANDSTONE_STAIRS) facing =$
 $north(RED_WALL_BANNER) facing = north(REPEATER) facing = north(REPEATING_COMMAND_BLOCK)$
 $north(SANDSTONE_STAIRS) facing = north(SKELETON_WALL_SKULL) facing =$
 $north(SMALL_DRIPLEAF) facing = north(SMOKER) facing = north(SMOOTH_QUARTZ_STAIRS) facing =$
 $north(SMOOTH_RED_SANDSTONE_STAIRS) facing = north(SMOOTH_SANDSTONE_STAIRS) facing =$
 $north(SOUL_CAMPFIRE) facing = north(SOUL_WALL_TORCH) facing = north(SPRUCE_BUTTON) facing =$
 $north(SPRUCE_DOOOR) facing = north(SPRUCE_FENCE_GATE) facing = north(SPRUCE_STAIRS) facing =$
 $north(SPRUCE_TRAPDOOR) facing = north(SPRUCE_WALL_SIGN) facing = north(STICKY_PISTON) facing =$
 $north(STONECUTTER) facing = north(STONE_BRICK_STAIRS) facing = north(STONE_BUTTON) facing =$
 $north(STONE_STAIRS) facing = north(TRAPPED_CHEST) facing = north(TRIPWIRE_HOOK) facing =$
 $north(TUBE_CORAL_WALL_FAN) facing = north(WALL_TORCH) facing = north(WARPED_BUTTON) facing =$
 $north(WARPED_DOOOR) facing = north(WARPED_FENCE_GATE) facing = north(WARPED_STAIRS) facing =$
 $north(WARPED_TRAPDOOR) facing = north(WARPED_WALL_SIGN) facing = north(WAXED_CUT_COPPER)$
 $north(WAXED_EXPOSED_CUT_COPPER_STAIRS) facing = north(WAXED_OXIDIZED_CUT_COPPER_STAIRS)$

$north(WAXED_W EATHERED_C UT_C OPPER_S TAIRS) facing = north(WEATHERED_C UT_C OPPER_S TAIRS) facing = north(WHITE_B ED) facing = north(WHITE_G LAZED_T ERRACOTTA) facing = north(WHITE_W ALL_B ANNER) facing = north(WITHER_S KELETON_W ALL_S KULL) facing = north(YELLOW_B ED) facing = north(YELLOW_G LAZED_T ERRACOTTA) facing = north(YELLOW_W ALL_B ANNER) facing = north(ZOMBIE_W ALL_H EAD) facing = south(OBSERVER) facing = up(AMETHYST_C LUSTER) facing = up(BLACK_S HULKER_B OX) facing = up(BLUE_S HULKER_B OX) facing = up(BROWN_S HULKER_B OX) facing = up(CYAN_S HULKER_B OX) facing = up(END_R OD) facing = up(GRAY_S HULKER_B OX) facing = up(GREEN_S HULKER_B OX) facing = up(LARGE_A METHYST_B UD) facing = up(LIGHTNING_R OD) facing = up(LIGHT_B LUE_S HULKER_B OX) facing = up(LIGHT_G RAY_S HULKER_B OX) facing = up(LIME_S HULKER_B OX) facing = up(MAGENTA_S HULKER_B OX) facing = up(MEDIUM_A METHYST_B UD) facing = up(ORANGE_S HULKER_B OX) facing = up(PINK_S HULKER_B OX) facing = up(PURPLE_S HULKER_B OX) facing = up(RED_S HULKER_B OX) facing = up(SHULKER_B OX) facing = up(SMALL_A METHYST_B UD) facing = up(WHITE_S HULKER_B OX) facing = up(YELLOW_S HULKER_B OX)$

$half=bottom(ACACIA_S TAIRS) half = bottom(ACACIA_T RAPDOOR) half = bottom(ANDESITE_S TAIRS) half = bottom(BIRCH_S TAIRS) half = bottom(BIRCH_T RAPDOOR) half = bottom(BLACKSTONE_S TAIRS) half = bottom(BRICK_S TAIRS) half = bottom(COBBLED_D EEP_S LATE_S TAIRS) half = bottom(COBBLESTONE_S TAIRS) half = bottom(CRIMSON_S TAIRS) half = bottom(CRIMSON_T RAPDOOR) half = bottom(CUT_C OPPER_S TAIRS) half = bottom(DARK_O AK_S TAIRS) half = bottom(DARK_O AK_T RAPDOOR) half = bottom(DARK_P RISMARINE_S TAIRS) half = bottom(DEEPSLATE_B RICK_S TAIRS) half = bottom(DEEPSLATE_T ILE_S TAIRS) half = bottom(DIORITE_S TAIRS) half = bottom(END_S TONE_B RICK_S TAIRS) half = bottom(EXPOSED_C UT_C OPPER_S TAIRS) half = bottom(GRANITE_S TAIRS) half = bottom(IRON_T RAPDOOR) half = bottom(JUNGLE_S TAIRS) half = bottom(JUNGLE_T RAPDOOR) half = bottom(MANGROVE_S TAIRS) half = bottom(MANGROVE_T RAPDOOR) half = bottom(MOSSY_C OBBLESTONE_S TAIRS) half = bottom(MOSSY_S TONE_B RICK_S TAIRS) half = bottom(MUD_B RICK_S TAIRS) half = bottom(NETHER_B RICK_S TAIRS) half = bottom(OAK_S TAIRS) half = bottom(OAK_T RAPDOOR) half = bottom(OXIDIZED_C UT_C OPPER_S TAIRS) half = bottom(POLISHED_A NDESITE_S TAIRS) half = bottom(POLISHED_B LACKSTONE_B RICK_S TAIRS) half = bottom(POLISHED_B LACKSTONE_S TAIRS) half = bottom(POLISHED_D EEP_S LATE_S TAIRS) half = bottom(POLISHED_D IORITE_S TAIRS) half = bottom(POLISHED_G RANITE_S TAIRS) half = bottom(PRISMARINE_B RICK_S TAIRS) half = bottom(PRISMARINE_S TAIRS) half = bottom(PURPUR_S TAIRS) half = bottom(QUARTZ_S TAIRS) half = bottom(RED_N ETHER_B RICK_S TAIRS) half = bottom(RED_S ANDSTONE_S TAIRS) half = bottom(SANDSTONE_S TAIRS) half = bottom(SMOOTH_Q UARTZ_S TAIRS)$

bottom(SMOOTH_RED_SANDSTONE_STAIRS)half = bottom(SMOOTH_SANDSTONE_STAIRS)half =
bottom(SPRUCE_STAIRS)half = bottom(SPRUCE_TRAPDOOR)half = bottom(STONE_BRICK_STAIRS
bottom(STONE_STAIRS)half = bottom(WARPED_STAIRS)half = bottom(WARPED_TRAPDOOR)half
bottom(WAXED_CUT_COPPER_STAIRS)half = bottom(WAXED_EXPOSED_CUT_COPPER_STAIRS)half
bottom(WAXED_OXIDIZED_CUT_COPPER_STAIRS)half = bottom(WAXED_WEATHERED_CUT_COPP
bottom(WEATHERED_CUT_COPPER_STAIRS)half = lower(ACACIA_DOOR)half =
lower(BIRCH_DOOR)half = lower(CRIMSON_DOOR)half = lower(DARK_OAK_DOOR)half =
lower(IRON_DOOR)half = lower(JUNGLE_DOOR)half = lower(LARGE_FERN)half =
lower(LILAC)half = lower(MANGROVE_DOOR)half = lower(OAK_DOOR)half =
lower(PEONY)half = lower(ROSE_BUSH)half = lower(SMALL_DRIPLEAF)half =
lower(SPRUCE_DOOR)half = lower(SUNFLOWER)half = lower(TALL_GRASS)half =
lower(TALL_SEAGRASS)half = lower(WARPED_DOOR)
 hanging=false (LANTERN) hanging=false (MANGROVE_PROPAGULE)hanging =
 false(SOUL_LANTERN)
 hatch=0 (TURTLE_EGG)
 hinge=left (ACACIA_DOOR)hinge = left(BIRCH_DOOR)hinge = left(CRIMSON_DOOR)hinge =
 left(DARK_OAK_DOOR)hinge = left(IRON_DOOR)hinge = left(JUNGLE_DOOR)hinge =
 left(MANGROVE_DOOR)hinge = left(OAK_DOOR)hinge = left(SPRUCE_DOOR)hinge =
 left(WARPED_DOOR)
 honey_level = 0(BEEHIVE)honey_level = 0(BEE_NEST)
 in_wall = false(ACACIA_FENCE_GATE)in_wall = false(BIRCH_FENCE_GATE)in_wall =
 false(CRIMSON_FENCE_GATE)in_wall = false(DARK_OAK_FENCE_GATE)in_wall =
 false(JUNGLE_FENCE_GATE)in_wall = false(MANGROVE_FENCE_GATE)in_wall =
 false(OAK_FENCE_GATE)in_wall = false(SPRUCE_FENCE_GATE)in_wall = false(WARPED_FENCE_GA
 inverted=false (DAYLIGHT_DETECTOR)
 layers=1 (SNOW)
 leaves=none (BAMBOO)
 level=0 (COMPOSTER) level=0 (LAVA) level=0 (WATER) level=1 (POWDER_SNOW_CAULDRON)leve
 1(WATER_CAULDRON)
 lit=false (BLACK_CANDLE)lit = false(BLACK_CANDLE_CAKE)lit = false(BLAST_FURNACE)lit =
 false(BLUE_CANDLE)lit = false(BLUE_CANDLE_CAKE)lit = false(BROWN_CANDLE)lit =

false(BROWN_CANDLE_AKE)lit = false(CANDLE)lit = false(CANDLE_AKE)lit =
false(CYAN_CANDLE)lit = false(CYAN_CANDLE_AKE)lit = false(DEEPSLATE_REDSTONE_ORE)
false(FURNACE)lit = false(GRAY_CANDLE)lit = false(GRAY_CANDLE_AKE)lit =
false(GREEN_CANDLE)lit = false(GREEN_CANDLE_AKE)lit = false(LIGHT_BLUE_CANDLE)lit =
false(LIGHT_BLUE_CANDLE_AKE)lit = false(LIGHT_GRAY_CANDLE)lit = false(LIGHT_GRAY_CAND
false(LIME_CANDLE)lit = false(LIME_CANDLE_AKE)lit = false(MAGENTA_CANDLE)lit =
false(MAGENTA_CANDLE_AKE)lit = false(ORANGE_CANDLE)lit = false(ORANGE_CANDLE_AK
false(PINK_CANDLE)lit = false(PINK_CANDLE_AKE)lit = false(PURPLE_CANDLE)lit =
false(PURPLE_CANDLE_AKE)lit = false(REDSTONE_LAMP)lit = false(REDSTONE_ORE)lit =
false(RED_CANDLE)lit = false(RED_CANDLE_AKE)lit = false(SMOKER)lit =
false(WHITE_CANDLE)lit = false(WHITE_CANDLE_AKE)lit = false(YELLOW_CANDLE)lit =
false(YELLOW_CANDLE_AKE)lit = true(CAMPFIRE)lit = true(REDSTONE_TORCH)lit =
true(REDSTONE_WALL_TORCH)lit = true(SOUL_CAMPFIRE)
 locked=false (REPEATER)
 mode=compare (COMPARATOR) mode=load (STRUCTURE_BLOCK)
 moisture=0 (FARMLAND)
 note=0 (NOTE_BLOCK)
 open=false (ACACIA_DOOR)open = false(ACACIA_FENCE_GATE)open = false(ACACIA_TRAPDOOR)
false(BARREL)open = false(BIRCH_DOOR)open = false(BIRCH_FENCE_GATE)open =
false(BIRCH_TRAPDOOR)open = false(CRIMSON_DOOR)open = false(CRIMSON_FENCE_GATE)op
false(CRIMSON_TRAPDOOR)open = false(DARK_OAK_DOOR)open = false(DARK_OAK_FENCE_GATE
false(DARK_OAK_TRAPDOOR)open = false(IRON_DOOR)open = false(IRON_TRAPDOOR)open =
false(JUNGLE_DOOR)open = false(JUNGLE_FENCE_GATE)open = false(JUNGLE_TRAPDOOR)open
false(MANGROVE_DOOR)open = false(MANGROVE_FENCE_GATE)open = false(MANGROVE_TRA
false(OAK_DOOR)open = false(OAK_FENCE_GATE)open = false(OAK_TRAPDOOR)open =
false(SPRUCE_DOOR)open = false(SPRUCE_FENCE_GATE)open = false(SPRUCE_TRAPDOOR)open
false(WARPED_DOOR)open = false(WARPED_FENCE_GATE)open = false(WARPED_TRAPDOOR)
 orientation=north_up(JIGSAW)
 part=foot (BLACK_BED)part = foot(BLUE_BED)part = foot(BROWN_BED)part =
 foot(CYAN_BED)part = foot(GRAY_BED)part = foot(GREEN_BED)part = foot(LIGHT_BLUE_BED)part
 foot(LIGHT_GRAY_BED)part = foot(LIME_BED)part = foot(MAGENTA_BED)part =

$foot(ORANGE_{BED})part = foot(PINK_{BED})part = foot(PURPLE_{BED})part = foot(RED_{BED})part =$
 $foot(WHITE_{BED})part = foot(YELLOW_{BED})$

$pickles=1 (SEA_{PICKLE})$

$rotation=0 (ACACIA_{SIGN})rotation = 0(BIRCH_{SIGN})rotation = 0(BLACK_{BANNNER})rotation =$
 $0(BLUE_{BANNNER})rotation = 0(BROWN_{BANNNER})rotation = 0(CREEPER_{HEAD})rotation =$
 $0(CRIMSON_{SIGN})rotation = 0(CYAN_{BANNNER})rotation = 0(DARK_{OAK}_{SIGN})rotation =$
 $0(DRAGON_{HEAD})rotation = 0(GRAY_{BANNNER})rotation = 0(GREEN_{BANNNER})rotation =$
 $0(JUNGLE_{SIGN})rotation = 0(LIGHT_{BLUE}_{BANNNER})rotation = 0(LIGHT_{GRAY}_{BANNNER})rotation =$
 $0(LIME_{BANNNER})rotation = 0(MAGENTA_{BANNNER})rotation = 0(MANGROVE_{SIGN})rotation =$
 $0(OAK_{SIGN})rotation = 0(ORANGE_{BANNNER})rotation = 0(PINK_{BANNNER})rotation =$
 $0(PLAYER_{HEAD})rotation = 0(PURPLE_{BANNNER})rotation = 0(RED_{BANNNER})rotation =$
 $0(SKELETON_{SKULL})rotation = 0(SPRUCE_{SIGN})rotation = 0(WARPED_{SIGN})rotation =$
 $0(WHITE_{BANNNER})rotation = 0(WITHER_{SKULL})rotation = 0(YELLOW_{BANNNER})rotation =$
 $0(ZOMBIE_{HEAD})$

$shape=north_south(ACTIVATOR_{RAIL})shape = north_south(DETECTOR_{RAIL})shape =$
 $north_south(POWERED_{RAIL})shape = north_south(RAIL)shape = straight(ACACIA_{STAIRS})shape =$
 $straight(ANDESITE_{STAIRS})shape = straight(BIRCH_{STAIRS})shape = straight(BLACKSTONE_{STAIRS})shape =$
 $straight(BRICK_{STAIRS})shape = straight(COBBLED_{DEEPSLATE}_{STAIRS})shape =$
 $straight(COBBLESTONE_{STAIRS})shape = straight(CRIMSON_{STAIRS})shape = straight(CUT_{COPPER}_{STAIRS})shape =$
 $straight(DARK_{OAK}_{STAIRS})shape = straight(DARK_{PRISMARINE}_{STAIRS})shape =$
 $straight(DEEPSLATE_{BRICK}_{STAIRS})shape = straight(DEEPSLATE_{TILE}_{STAIRS})shape =$
 $straight(DIORITE_{STAIRS})shape = straight(ENDSTONE_{BRICK}_{STAIRS})shape =$
 $straight(EXPOSED_{CUT}_{COPPER}_{STAIRS})shape = straight(GRANITE_{STAIRS})shape =$
 $straight(JUNGLE_{STAIRS})shape = straight(MANGROVE_{STAIRS})shape = straight(MOSSY_{COBBLE}_{STAIRS})shape =$
 $straight(MOSSY_{STONE}_{BRICK}_{STAIRS})shape = straight(MUD_{BRICK}_{STAIRS})shape =$
 $straight(NETHER_{BRICK}_{STAIRS})shape = straight(OAK_{STAIRS})shape = straight(OXIDIZED_{CUT}_{COPPER}_{STAIRS})shape =$
 $straight(POLISHED_{ANDESITE}_{STAIRS})shape = straight(POLISHED_{BLACKSTONE}_{BRICK}_{STAIRS})shape =$
 $straight(POLISHED_{BLACKSTONE}_{STAIRS})shape = straight(POLISHED_{DEEPSLATE}_{STAIRS})shape =$
 $straight(POLISHED_{DIORITE}_{STAIRS})shape = straight(POLISHED_{GRANITE}_{STAIRS})shape =$
 $straight(PRISMARINE_{BRICK}_{STAIRS})shape = straight(PRISMARINE_{STAIRS})shape =$
 $straight(PURPUR_{STAIRS})shape = straight(QUARTZ_{STAIRS})shape = straight(RED_{NETHER}_{BRICK}_{STAIRS})shape =$

straight(RED_SANDSTONE_STAIRS)*shape* = *straight*(SANDSTONE_STAIRS)*shape* =
straight(SMOOTH_QU ARTZ_STAIRS)*shape* = *straight*(SMOOTH_RED_SANDSTONE_STAIRS)*shape* =
straight(SMOOTH_SANDSTONE_STAIRS)*shape* = *straight*(SPRUCE_STAIRS)*shape* =
straight(STONE_BRICK_STAIRS)*shape* = *straight*(STONE_STAIRS)*shape* = *straight*(WARPED_STAIRS)*shape* =
straight(WAXED_CUT_COPPER_STAIRS)*shape* = *straight*(WAXED_EXPOSED_CUT_COPPER_STAIRS)*shape* =
straight(WAXED_OXIDIZED_CUT_COPPER_STAIRS)*shape* = *straight*(WAXED_WEATHERED_CUT_COPPER_STAIRS)*shape* =
straight(WEATHERED_CUT_COPPER_STAIRS)*shape* =
snowy=false (GRASS_BLOCK)*snowy* = false(MYCELIUM)*snowy* = false(PODZOL)
thickness=tip (POINTED_DRIPSTONE)
type=bottom (ACACIA_SLAB)*type* = bottom(ANDESITE_SLAB)*type* = bottom(BIRCH_SLAB)*type* =
bottom(BLACKSTONE_SLAB)*type* = bottom(BRICK_SLAB)*type* = bottom(COBBLED_DEEPSLATE_SLAB)*type* =
bottom(COBBLESTONE_SLAB)*type* = bottom(CRIMSON_SLAB)*type* = bottom(CUT_COPPER_SLAB)*type* =
bottom(CUT_RED_SANDSTONE_SLAB)*type* = bottom(CUT_SANDSTONE_SLAB)*type* =
bottom(DARK_OAK_SLAB)*type* = bottom(DARK_PRISMARINE_SLAB)*type* = bottom(DEEPSLATE_BRICK_SLAB)*type* =
bottom(DEEPSLATE_TILE_SLAB)*type* = bottom(DIORITE_SLAB)*type* = bottom(END_SSTONE_BRICK_SLAB)*type* =
bottom(EXPOSED_CUT_COPPER_SLAB)*type* = bottom(GRANITE_SLAB)*type* = bottom(JUNGLE_SLAB)*type* =
bottom(MANGROVE_SLAB)*type* = bottom(MOSSY_COBBLESTONE_SLAB)*type* = bottom(MOSSY_STOILET)*type* =
bottom(MUD_BRICK_SLAB)*type* = bottom(NETHER_BRICK_SLAB)*type* = bottom(OAK_SLAB)*type* =
bottom(OXIDIZED_CUT_COPPER_SLAB)*type* = bottom(PETRIFIED_OAK_SLAB)*type* =
bottom(POLISHED_ANDESITE_SLAB)*type* = bottom(POLISHED_BLACKSTONE_BRICK_SLAB)*type* =
bottom(POLISHED_BLACKSTONE_SLAB)*type* = bottom(POLISHED_DEEPSLATE_SLAB)*type* =
bottom(POLISHED_DIORITE_SLAB)*type* = bottom(POLISHED_GRANITE_SLAB)*type* =
bottom(PRISMARINE_BRICK_SLAB)*type* = bottom(PRISMARINE_SLAB)*type* = bottom(PURPUR_SLAB)*type* =
bottom(QUARTZ_SLAB)*type* = bottom(RED_NETHER_BRICK_SLAB)*type* = bottom(RED_SANDSTONE_SLAB)*type* =
bottom(SANDSTONE_SLAB)*type* = bottom(SMOOTH_QU ARTZ_SLAB)*type* = bottom(SMOOTH_RED_SANDSTONE_SLAB)*type* =
bottom(SMOOTH_SANDSTONE_SLAB)*type* = bottom(SMOOTH_SSTONE_SLAB)*type* =
bottom(SPRUCE_SLAB)*type* = bottom(STONE_BRICK_SLAB)*type* = bottom(STONE_SLAB)*type* =
bottom(WARPED_SLAB)*type* = bottom(WAXED_CUT_COPPER_SLAB)*type* = bottom(WAXED_EXPOSED_CUT_COPPER_SLAB)*type* =
bottom(WAXED_OXIDIZED_CUT_COPPER_SLAB)*type* = bottom(WAXED_WEATHERED_CUT_COPPER_SLAB)*type* =
bottom(WEATHERED_CUT_COPPER_SLAB)*type* = normal(MOVING_PISTON)*type* =
normal(PISTON_HEAD)*type* = single(CHEST)*type* = single(TRAPPED_CHEST)

$\text{vertical}_{direction} = \text{up}(\text{POINTED}_D\text{RIPSTONE})$

A.1.16 Waterlogged

Denotes whether this block has fluid in it.

Besides underwater blocks⁹ (which defaults to true), it defaults to false. All the possible options are true or false.

Material	Aquatic block ¹⁰
ACACIA_FENCE	X
ACACIA_LEAVES	X
ACACIA_SIGN	X
ACACIA_SLAB	X
ACACIA_STAIRS	X
ACACIA_TRAPDOOR	X
ACACIA_WALL_SIGN	X
ACTIVATOR_RAIL	X
AMETHYST_CLUSTER	X
ANDESITE_SLAB	X
ANDESITE_STAIRS	X
ANDESITE_WALL	X
AZALEA_LEAVES	X
BIG_DRIPLEAF	X
BIG_DRIPLEAF_STEM	X
BIRCH_FENCE	X

⁹BRAIN_CORAL, BRAIN_CORAL_FAN, BRAIN_CORAL_WALL_FAN, BUBBLE_CORAL, BUBBLE_CORAL_FAN, BUBBLE_CORAL_WALL_FAN, CONDUIT, DEAD_BRAIN_CORAL, DEAD_BRAIN_CORAL_FAN, DEAD_BRAIN_CORAL_WALL_FAN, DEAD_BUBBLE_CORAL, DEAD_BUBBLE_CORAL_FAN, DEAD_BUBBLE_CORAL_WALL_FAN, DEAD_FIRE_CORAL, DEAD_FIRE_CORAL_FAN, DEAD_FIRE_CORAL_WALL_FAN, DEAD_HORN_CORAL, DEAD_HORN_CORAL_FAN, DEAD_HORN_CORAL_WALL_FAN, DEAD_TUBE_CORAL, DEAD_TUBE_CORAL_FAN, DEAD_TUBE_CORAL_WALL_FAN, FIRE_CORAL, FIRE_CORAL_FAN, FIRE_CORAL_WALL_FAN, HORN_CORAL, HORN_CORAL_FAN, HORN_CORAL_WALL_FAN, SEA_PICKLE, TUBE_CORAL, TUBE_CORAL_FAN and TUBE_CORAL_WALL_FAN

Material	Aquatic block ¹⁰
BIRCH_LEAVES	✗
BIRCH_SIGN	✗
BIRCH_SLAB	✗
BIRCH_STAIRS	✗
BIRCH_TRAPDOOR	✗
BIRCH_WALL_SIGN	✗
BLACKSTONE_SLAB	✗
BLACKSTONE_STAIRS	✗
BLACKSTONE_WALL	✗
BLACK_CANDLE	✗
BLACK_STAINED_GLASS_PANE	✗
BLUE_CANDLE	✗
BLUE_STAINED_GLASS_PANE	✗
BRICK_SLAB	✗
BRICK_STAIRS	✗
BRICK_WALL	✗
BROWN_CANDLE	✗
BROWN_STAINED_GLASS_PANE	✗
CAMPFIRE	✗
CANDLE	✗
CHAIN	✗
CHEST	✗
COBBLED_DEEPSLATE_SLAB	✗
COBBLED_DEEPSLATE_STAIRS	✗
COBBLED_DEEPSLATE_WALL	✗
COBBLESTONE_SLAB	✗
COBBLESTONE_STAIRS	✗
COBBLESTONE_WALL	✗
CRIMSON_FENCE	✗
CRIMSON_SIGN	✗

Material	Aquatic block ¹⁰
CRIMSON_SLAB	✗
CRIMSON_STAIRS	✗
CRIMSON_TRAPDOOR	✗
CRIMSON_WALL_SIGN	✗
CUT_COPPER_SLAB	✗
CUT_COPPER_STAIRS	✗
CUT_RED_SANDSTONE_SLAB	✗
CUT_SANDSTONE_SLAB	✗
CYAN_CANDLE	✗
CYAN_STAINED_GLASS_PANE	✗
DARK_OAK_FENCE	✗
DARK_OAK_LEAVES	✗
DARK_OAK_SIGN	✗
DARK_OAK_SLAB	✗
DARK_OAK_STAIRS	✗
DARK_OAK_TRAPDOOR	✗
DARK_OAK_WALL_SIGN	✗
DARK_PRISMARINE_SLAB	✗
DARK_PRISMARINE_STAIRS	✗
DEEPSLATE_BRICK_SLAB	✗
DEEPSLATE_BRICK_STAIRS	✗
DEEPSLATE_BRICK_WALL	✗
DEEPSLATE_TILE_SLAB	✗
DEEPSLATE_TILE_STAIRS	✗
DEEPSLATE_TILE_WALL	✗
DETECTOR_RAIL	✗
DIORITE_SLAB	✗
DIORITE_STAIRS	✗
DIORITE_WALL	✗
ENDER_CHEST	✗

Material	Aquatic block ¹⁰
END_STONE_BRICK_SLAB	✗
END_STONE_BRICK_STAIRS	✗
END_STONE_BRICK_WALL	✗
EXPOSED_CUT_COPPER_SLAB	✗
EXPOSED_CUT_COPPER_STAIRS	✗
FLOWERING_AZALEA_LEAVES	✗
GLASS_PANE	✗
GLOW_LICHEN	✗
GRANITE_SLAB	✗
GRANITE_STAIRS	✗
GRANITE_WALL	✗
GRAY_CANDLE	✗
GRAY_STAINED_GLASS_PANE	✗
GREEN_CANDLE	✗
GREEN_STAINED_GLASS_PANE	✗
HANGING_ROOTS	✗
IRON_BARS	✗
IRON_TRAPDOOR	✗
JUNGLE_FENCE	✗
JUNGLE_LEAVES	✗
JUNGLE_SIGN	✗
JUNGLE_SLAB	✗
JUNGLE_STAIRS	✗
JUNGLE_TRAPDOOR	✗
JUNGLE_WALL_SIGN	✗
LADDER	✗
LANTERN	✗
LARGE_AMETHYST_BUD	✗
LIGHTNING_ROD	✗
LIGHT_BLUE_CANDLE	✗

Material	Aquatic block ¹⁰
LIGHT_BLUE_STAINED_GLASS_PANE	✗
LIGHT_GRAY_CANDLE	✗
LIGHT_GRAY_STAINED_GLASS_PANE	✗
LIME_CANDLE	✗
LIME_STAINED_GLASS_PANE	✗
MAGENTA_CANDLE	✗
MAGENTA_STAINED_GLASS_PANE	✗
MANGROVE_FENCE	✗
MANGROVE_LEAVES	✗
MANGROVE_PROPAGULE	✗
MANGROVE_ROOTS	✗
MANGROVE_SIGN	✗
MANGROVE_SLAB	✗
MANGROVE_STAIRS	✗
MANGROVE_TRAPDOOR	✗
MANGROVE_WALL_SIGN	✗
MEDIUM_AMETHYST_BUD	✗
MOSSY_COBBLESTONE_SLAB	✗
MOSSY_COBBLESTONE_STAIRS	✗
MOSSY_COBBLESTONE_WALL	✗
MOSSY_STONE_BRICK_SLAB	✗
MOSSY_STONE_BRICK_STAIRS	✗
MOSSY_STONE_BRICK_WALL	✗
MUD_BRICK_SLAB	✗
MUD_BRICK_STAIRS	✗
MUD_BRICK_WALL	✗
NETHER_BRICK_FENCE	✗
NETHER_BRICK_SLAB	✗
NETHER_BRICK_STAIRS	✗
NETHER_BRICK_WALL	✗

Material	Aquatic block ¹⁰
OAK_FENCE	✗
OAK_LEAVES	✗
OAK_SIGN	✗
OAK_SLAB	✗
OAK_STAIRS	✗
OAK_TRAPDOOR	✗
OAK_WALL_SIGN	✗
ORANGE_CANDLE	✗
ORANGE_STAINED_GLASS_PANE	✗
OXIDIZED_CUT_COPPER_SLAB	✗
OXIDIZED_CUT_COPPER_STAIRS	✗
PETRIFIED_OAK_SLAB	✗
PINK_CANDLE	✗
PINK_STAINED_GLASS_PANE	✗
POINTED_DRIPSTONE	✗
POLISHED_ANDESITE_SLAB	✗
POLISHED_ANDESITE_STAIRS	✗
POLISHED_BLACKSTONE_BRICK_SLAB	✗
POLISHED_BLACKSTONE_BRICK_STAIRS	✗
POLISHED_BLACKSTONE_BRICK_WALL	✗
POLISHED_BLACKSTONE_SLAB	✗
POLISHED_BLACKSTONE_STAIRS	✗
POLISHED_BLACKSTONE_WALL	✗
POLISHED_DEEPSLATE_SLAB	✗
POLISHED_DEEPSLATE_STAIRS	✗
POLISHED_DEEPSLATE_WALL	✗
POLISHED_DIORITE_SLAB	✗
POLISHED_DIORITE_STAIRS	✗
POLISHED_GRANITE_SLAB	✗
POLISHED_GRANITE_STAIRS	✗

Material	Aquatic block ¹⁰
POWERED_RAIL	✗
PRISMARINE_BRICK_SLAB	✗
PRISMARINE_BRICK_STAIRS	✗
PRISMARINE_SLAB	✗
PRISMARINE_STAIRS	✗
PRISMARINE_WALL	✗
PURPLE_CANDLE	✗
PURPLE_STAINED_GLASS_PANE	✗
PURPUR_SLAB	✗
PURPUR_STAIRS	✗
QUARTZ_SLAB	✗
QUARTZ_STAIRS	✗
RAIL	✗
RED_CANDLE	✗
RED_NETHER_BRICK_SLAB	✗
RED_NETHER_BRICK_STAIRS	✗
RED_NETHER_BRICK_WALL	✗
RED_SANDSTONE_SLAB	✗
RED_SANDSTONE_STAIRS	✗
RED_SANDSTONE_WALL	✗
RED_STAINED_GLASS_PANE	✗
SANDSTONE_SLAB	✗
SANDSTONE_STAIRS	✗
SANDSTONE_WALL	✗
SCAFFOLDING	✗
SCULK_SENSOR	✗
SCULK_SHRIEKER	✗
SCULK_VEIN	✗
SMALL_AMETHYST_BUD	✗
SMALL_DRIPLEAF	✗

Material	Aquatic block ¹⁰
SMOOTH_QUARTZ_SLAB	✗
SMOOTH_QUARTZ_STAIRS	✗
SMOOTH_RED_SANDSTONE_SLAB	✗
SMOOTH_RED_SANDSTONE_STAIRS	✗
SMOOTH_SANDSTONE_SLAB	✗
SMOOTH_SANDSTONE_STAIRS	✗
SMOOTH_STONE_SLAB	✗
SOUL_CAMPFIRE	✗
SOUL_LANTERN	✗
SPRUCE_FENCE	✗
SPRUCE_LEAVES	✗
SPRUCE_SIGN	✗
SPRUCE_SLAB	✗
SPRUCE_STAIRS	✗
SPRUCE_TRAPDOOR	✗
SPRUCE_WALL_SIGN	✗
STONE_BRICK_SLAB	✗
STONE_BRICK_STAIRS	✗
STONE_BRICK_WALL	✗
STONE_SLAB	✗
STONE_STAIRS	✗
TRAPPED_CHEST	✗
WARPED_FENCE	✗
WARPED_SIGN	✗
WARPED_SLAB	✗
WARPED_STAIRS	✗
WARPED_TRAPDOOR	✗
WARPED_WALL_SIGN	✗
WAXED_CUT_COPPER_SLAB	✗
WAXED_CUT_COPPER_STAIRS	✗

Material	Aquatic block ¹⁰
WAXED_EXPOSED_CUT_COPPER_SLAB	✗
WAXED_EXPOSED_CUT_COPPER_STAIRS	✗
WAXED_OXIDIZED_CUT_COPPER_SLAB	✗
WAXED_OXIDIZED_CUT_COPPER_STAIRS	✗
WAXED_WEATHERED_CUT_COPPER_SLAB	✗
WAXED_WEATHERED_CUT_COPPER_STAIRS	✗
WEATHERED_CUT_COPPER_SLAB	✗
WEATHERED_CUT_COPPER_STAIRS	✗
WHITE_CANDLE	✗
WHITE_STAINED_GLASS_PANE	✗
YELLOW_CANDLE	✗
YELLOW_STAINED_GLASS_PANE	✗
BRAIN_CORAL	✓
BRAIN_CORAL_FAN	✓
BRAIN_CORAL_WALL_FAN	✓
BUBBLE_CORAL	✓
BUBBLE_CORAL_FAN	✓
BUBBLE_CORAL_WALL_FAN	✓
CONDUIT	✓
DEAD_BRAIN_CORAL	✓
DEAD_BRAIN_CORAL_FAN	✓
DEAD_BRAIN_CORAL_WALL_FAN	✓
DEAD_BUBBLE_CORAL	✓
DEAD_BUBBLE_CORAL_FAN	✓
DEAD_BUBBLE_CORAL_WALL_FAN	✓
DEAD_FIRE_CORAL	✓
DEAD_FIRE_CORAL_FAN	✓
DEAD_FIRE_CORAL_WALL_FAN	✓
DEAD_HORN_CORAL	✓
DEAD_HORN_CORAL_FAN	✓

Material	Aquatic block ¹⁰
DEAD_HORN_CORAL_WALL_FAN	✓
DEAD_TUBE_CORAL	✓
DEAD_TUBE_CORAL_FAN	✓
DEAD_TUBE_CORAL_WALL_FAN	✓
FIRE_CORAL	✓
FIRE_CORAL_FAN	✓
FIRE_CORAL_WALL_FAN	✓
HORN_CORAL	✓
HORN_CORAL_FAN	✓
HORN_CORAL_WALL_FAN	✓
SEA_PICKLE	✓
TUBE_CORAL	✓
TUBE_CORAL_FAN	✓
TUBE_CORAL_WALL_FAN	✓

Table A.8: Waterlogged materials

A.2 Material modifiers concatenation

... (how to join modifiers)

If a material doesn't have the attribute that the diagram is checking it will assume that the attribute value is the default one (0 or false, in most of the cases), resulting in ignoring that property.

¹⁰If it's an underwater block (defaults to false).

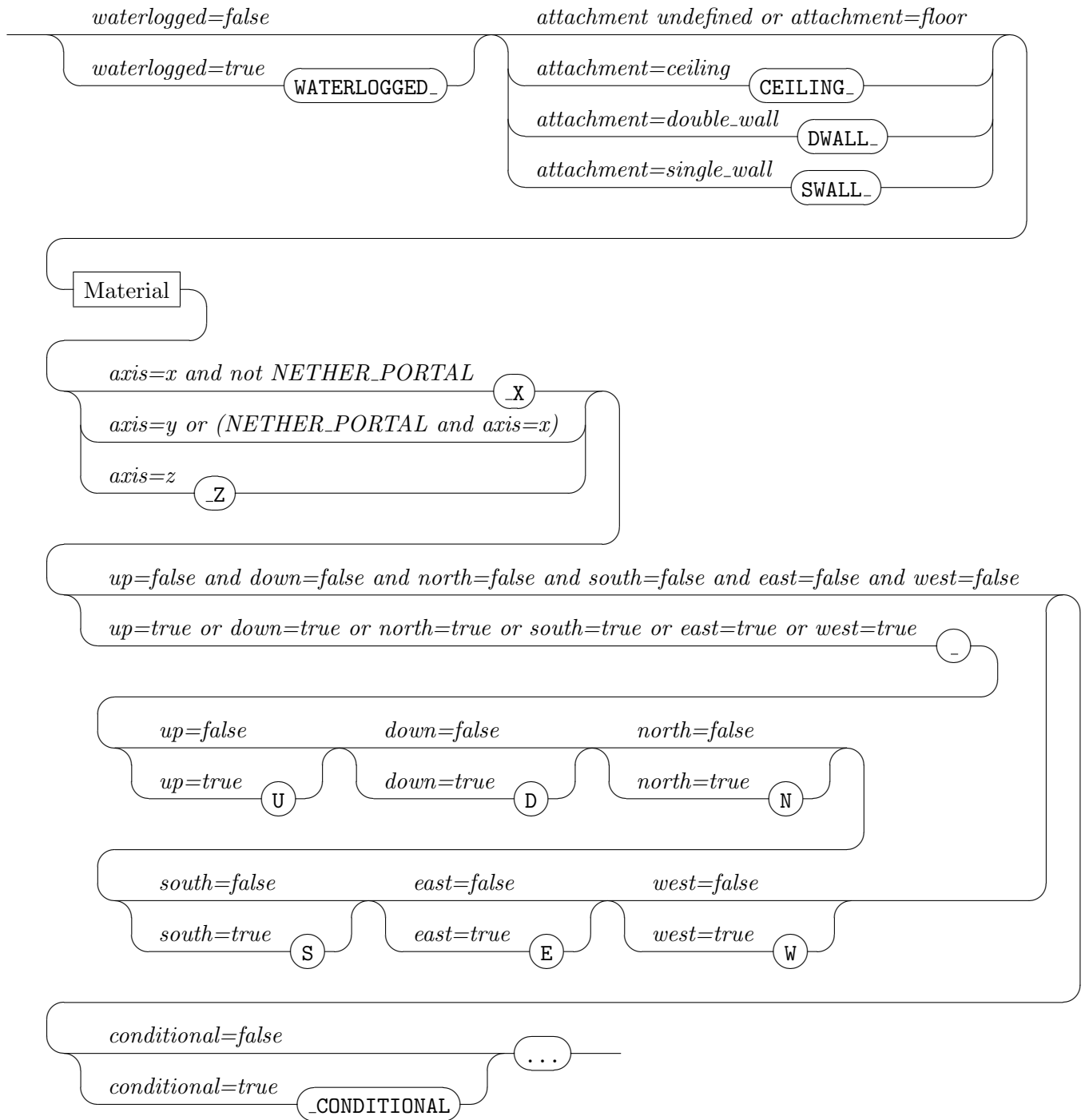


Figure A.1a: Modifier concatenation

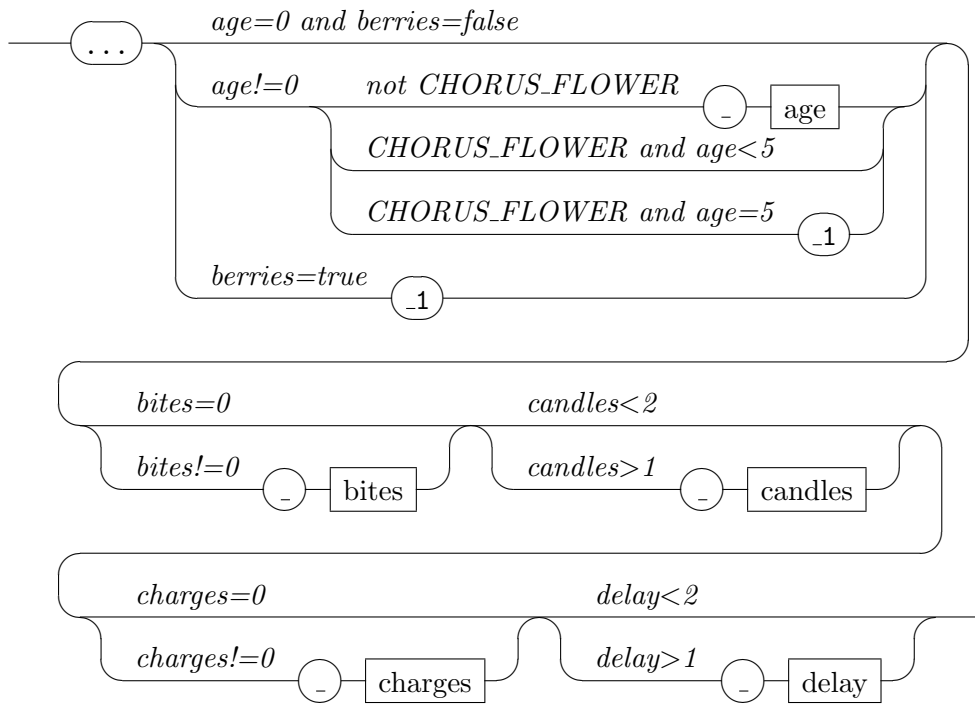


Figure A.1b: Integer modifier concatenation

References

Papermc. (n.d.). Retrieved from <https://papermc.io/>

Spigot. (n.d.). Retrieved from <https://www.spigotmc.org/>

Worldguard. (n.d.). Retrieved from <https://dev.bukkit.org/projects/worldguard>

World - minecraft wiki. (n.d.). Retrieved from <https://minecraft.fandom.com/wiki/>

World