WatchWolf API Definition

WatchWolf Contributors

Keywords: WatchWolf; Minecraft plugin testing; Integration testing environment.

Contents

1 Documentation conventions **12**

1.1 Acronyms . . . 12

1.2 Glossary . . . 12

2 WatchWolf Introduction **13**

3 API Introduction **15**

3.1 Destiny . . . 15

3.2 Response . . . 15

3.3 Operation . . . 16

3.4 Arguments . . . 16

3.4.1 Character . . . 16

3.4.2 Boolean . . . 16

3.4.3 Double . . . 17

3.4.4 String . . . 17

3.4.5 Array . . . 17

3.4.6 File . . . 18

3.4.7 Server type . . . 19

3.4.8 Position . . . 19

3.4.9 Block . . . 20

3.4.10 Item . . . 22

3.4.11 Entity . . . 22

3.4.12 Container . . . 24

4 Server manager petition **26**

4.1 Start server operation . . . 26

4.1.1 Maps . . . 28

4.1.2 Plugins . . . 29

4.1.3 Server version . . . 30

4.1.4 Config files . . . 30

4.2 Server started notification . . . 30

4.3 Error notification 31
4.4 Get coverage operation 31
4.4.1 Get coverage request 31
4.4.2 Get coverage response 31

5 Server petition **33**
5.1 Server petition group 33
5.2 Server petition operation 34
5.3 Base operations 34
5.3.1 Server stop operation 35
5.3.2 Server stopped notification 35
5.3.3 Server started notification 35
5.3.4 Whitelist player operation 36
5.3.5 OP player operation 36
5.3.6 Set block operation 37
5.3.7 Get block operation 37
5.3.8 Get block response 37
5.3.9 Get player position operation 38
5.3.10 Get player position response 38
5.3.11 Give item to player operation 39
5.3.12 Run command operation 39
5.3.13 Get players operation 39
5.3.14 Get players response 40
5.3.15 Synchronize operation 40
5.3.16 Synchronize response 40
5.3.17 Tp player operation 40
5.3.18 Get player pitch operation 41
5.3.19 Get player pitch response 41
5.3.20 Get player yaw operation 42
5.3.21 Get player yaw response 42
5.3.22 Get player inventory operation 43
5.3.23 Get player inventory response 43

5.3.24 Get entities operation . . . 43
5.3.25 Get entities response . . . 44
5.3.26 Spawn entity operation . . . 44
5.3.27 Spawn entity response . . . 45
5.4 Performance operations . . . 45
5.5 WorldGuard operations . . . 45
5.6 Residence operations . . . 45

6 Clients manager petition 46
6.1 Start client operation . . . 46
6.2 Client started response operation . . . 46

7 Client petition 48
7.1 Stop client operation . . . 48
7.2 Send message operation . . . 48
7.3 Got message notification . . . 48
7.4 Run command operation . . . 49
7.5 Break block operation . . . 49
7.6 Equip item in hand operation . . . 50
7.7 Move at operation . . . 50
7.8 Look at operation . . . 51
7.9 Synchronize operation . . . 51
7.10 Synchronize response . . . 51
7.11 Hit operation . . . 51
7.12 Use operation . . . 52
7.13 Place block operation . . . 52
7.14 Hit entity operation . . . 52

8 Revision history 53

A Blocks 54
A.1 Unused material modifiers . . . 54
A.2 Material modifiers . . . 58

A.2.1 Age . . . 58
A.2.2 Attachment . . . 58
A.2.3 Axis . . . 58
A.2.4 Berries . . . 60
A.2.5 Bites . . . 61
A.2.6 Candles . . . 61
A.2.7 Charges . . . 62
A.2.8 Conditional . . . 62
A.2.9 Delay . . . 62
A.2.10 Down . . . 63
A.2.11 North, South, East and West . . . 63
A.2.12 Up . . . 65
A.2.13 Eggs . . . 66
A.2.14 Extended . . . 66
A.2.15 Eye . . . 66
A.2.16 Face . . . 67
A.2.17 Facing . . . 67
A.2.18 Half . . . 75
A.2.19 Hanging . . . 78
A.2.20 Hinge . . . 79
A.2.21 Honey level . . . 79
A.2.22 Inverted . . . 79
A.2.23 Layers . . . 80
A.2.24 Leaves . . . 80
A.2.25 Level . . . 80
A.2.26 Lit . . . 81
A.2.27 Locked . . . 83
A.2.28 Mode . . . 83
A.2.29 Note . . . 83
A.2.30 Open . . . 83
A.2.31 Orientation . . . 85

A.2.32 Part 85
A.2.33 Pickles 86
A.2.34 Powered 86
A.2.35 Rotation 88
A.2.36 Shape - Rails 89
A.2.37 Shape - Stairs 90
A.2.38 Type 94
A.2.39 Vertical direction 96
A.2.40 Waterlogged 96
A.3 Material modifiers aggregation 106
A.3.1 Direction aggregation 106
A.3.2 Axis aggregation 107
A.3.3 Age aggregation 108
A.3.4 Grouped materials aggregation 108
A.3.5 Stages aggregation 108
A.3.6 Parts aggregation 109

References **110**

Figures

2.1 Diagram representing WatchWolf's most important actuators 14
3.1 Packet structure 15
3.2 True packet with the LSB at 1 17
3.3 True packet with all bits at 1 17
3.4 Structure of a String 17
3.5 Example of a string array 18
3.6 File structure 19
3.7 Position structure 20
3.8 Structure of a Block 21
3.9 Structure of an Item 22
3.10 Structure of an Entity 23
3.11 Structure of a DroppedItem 24
3.12 Structure of a Container 25
4.1 Server manager petition structure 26
4.2 Start server petition structure 27
4.3 Start server response structure 28
4.4 Start server error response structure 28
4.5 Usual plugin structure 29
4.6 File plugin structure 30
4.7 Server started notification structure 31
4.8 Error notification structure 31
4.9 Get coverage operation structure 31
4.10 Get coverage response structure 32
5.1 Server petition structure 33
5.2 Implemented group response structure 34
5.3 Stop server operation structure 35
5.4 Server stopped response structure 35
5.5 Server started response structure 36
5.6 Whitelist player operation structure 36
5.7 OP player operation structure 36

5.8 Set block operation structure 37
5.9 Get block operation structure 37
5.10 Get block operation structure 38
5.11 Get player position operation structure 38
5.12 Get player position response structure 38
5.13 Give item to player operation structure 39
5.14 Run command operation structure 39
5.15 Get players operation structure 40
5.16 Get players response structure 40
5.17 Synchronize operation structure 40
5.18 Synchronize response structure 40
5.19 Tp player operation structure 41
5.20 Get player pitch operation structure 41
5.21 Get player pitch response structure 42
5.22 Get player yaw operation structure 42
5.23 Get player yaw response structure 42
5.24 Get player inventory operation structure 43
5.25 Get player inventory response structure 43
5.26 Get entities operation structure 44
5.27 Get entities response structure 44
5.28 Spawn entity operation structure 45
5.29 Spawn entity response structure 45
6.1 Clients manager petition structure 46
6.2 Start client petition structure 46
6.3 Client started petition response structure 47
7.1 Client petition structure 48
7.2 Stop client petition structure 48
7.3 Send message petition structure 48
7.4 Got message notification structure 49
7.5 Run command petition structure 49
7.6 Run command petition structure 50

7.7 Run command petition structure 50
7.8 Move at petition structure 50
7.9 Move at petition structure 51
7.10 Synchronize petition structure 51
7.11 Synchronize petition response structure 51
7.12 Hit petition structure 52
7.13 Use petition structure 52
7.14 Place block petition structure 52
7.15 Place block petition structure 52

Tables

3.1 DST bits meaning . . . 15
3.2 Block enum . . . 22
3.3 Item enum . . . 22
3.4 Entity enum . . . 23
3.5 Entity enum . . . 25
5.1 Extended types . . . 34
8.1 Revision history . . . 53
A.1 Unused Spigot BlockData's modifiers . . . 55
A.2 Unused Spigot BlockData's modifiers on certain blocks . . . 57
A.3 Ageable materials . . . 58
A.4 Attachable materials . . . 58
A.5 Orientable materials . . . 60
A.6 Materials with berries . . . 61
A.7 Cake . . . 61
A.8 Materials with candles . . . 62
A.9 Charged materials . . . 62
A.10 Conditionable materials . . . 62
A.11 Delayable materials . . . 63
A.12 Materials with down option . . . 63
A.13 Orientable materials . . . 65
A.14 Materials with up option . . . 66
A.15 Materials with eggs . . . 66
A.16 Extendable materials . . . 66
A.17 Materials with eye . . . 67
A.18 Directional materials . . . 67
A.19 Directional materials . . . 75
A.20 Two-blocks materials . . . 78
A.21 Hangable materials . . . 78
A.22 Doors . . . 79
A.23 Hives . . . 79

A.24 Invertible blocks . . . 80
A.25 Snow . . . 80
A.26 Bamboo . . . 80
A.27 Levelled materials . . . 81
A.28 Lightable materials . . . 82
A.29 Repeater . . . 83
A.30 Materials with mode . . . 83
A.31 Note block . . . 83
A.32 Openable materials . . . 84
A.33 Jigsaw orientations . . . 85
A.34 Jigsaw . . . 85
A.35 Openable materials . . . 86
A.36 Materials with pickles . . . 86
A.37 Powerabled materials . . . 88
A.38 Relation between rotation and where is looking . . . 88
A.39 Directional materials . . . 89
A.40 Rail directions . . . 90
A.41 Rails . . . 90
A.42 Stairs . . . 94
A.43 Type materials . . . 96
A.44 Dripstone . . . 96
A.45 Waterlogged materials . . . 105
A.46 Direction aggregation . . . 107
A.47 Axis aggregation . . . 107
A.48 Age aggregation . . . 108
A.49 Grouped materials aggregation . . . 108
A.50 Stages aggregation . . . 108
A.51 Parts aggregation . . . 109

1 Documentation conventions

1.1 Acronyms

MC Minecraft. 13

1.2 Glossary

async Asynchronous; returned non-sequentially. 16

IEEE 754 IEEE Standard for Floating-Point Arithmetic (IEEE 754) is a standard for computing floating-points operations. For more details about it check Rajaraman (2016). 17

Least significant bit The Least significant bit is the bit with the lowest index number. 7, 17

Most significant bit The Most significant bit is the bit with the higher index number. 33

Multidimensional array Array of arrays. 18

NOP No OPeration. It represents a valid operation that means 'do nothing'. 16

2 WatchWolf Introduction

WatchWolf is an integration testing environment for Minecraft plugins. It will validate that your plugin works using multiple real MC servers of different types and versions.

In order to achieve that, WatchWolf splits into 4 different programs, each one with one responsibility:

1. WatchWolf Tester

   WatchWolf Tester if the entry point to the WatchWolf environment.[1] It will orchestrate all the setup/stop process and run the user tests.

2. WatchWolf Servers Manager

   WatchWolf Servers Manager provides MC servers on-demand. It will start them and, after they have been closed, free the allocated resources.

3. WatchWolf Server

   WatchWolf Server is the actual MC server. It will contain the plugin to test and run the commands sent by WatchWolf Tester.

4. WatchWolf Clients Manager

   WatchWolf Clients Manager is the same as WatchWolf Servers Manager, but for clients. It will starts clients on-demand and connect them to the servers allocated by WatchWolf.

5. WatchWolf Client

   WatchWolf Client is a MC client, with the ability to connect to one server and interact with it.

You can see with more detail how the different programs relations on the Figure 2.1, Diagram representing WatchWolf's most important actuators.

[1] The WatchWolf environment is the combination of all the WatchWolf parts: Tester, Servers Manager, Server, Clients Manager and Client.

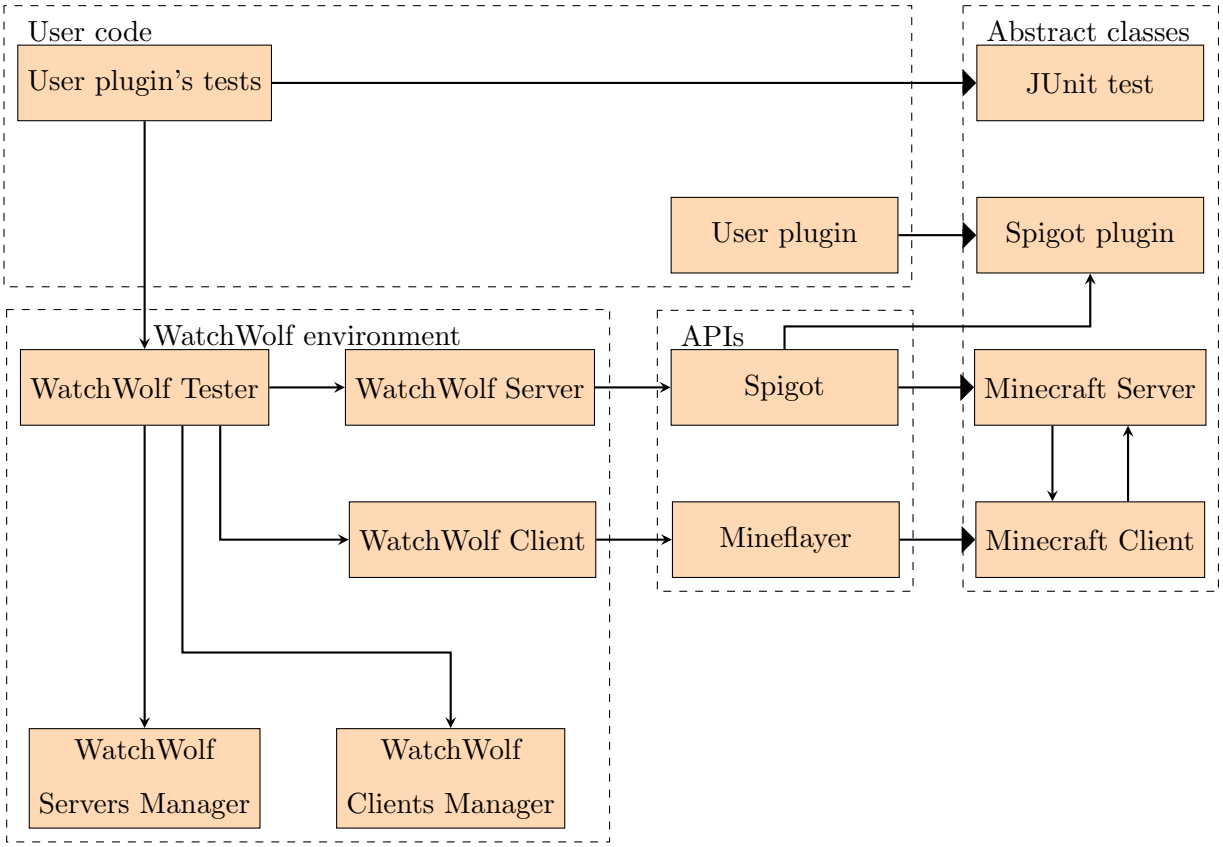

Figure 2.1: Diagram representing WatchWolf's most important actuators

3 API Introduction

In order to interact with the different WatchWolf modules, you'll have to follow the WatchWolf API: a series of supported operations in one program. All the packets sent & received will follow the structure shown in Figure 3.1, Packet structure.

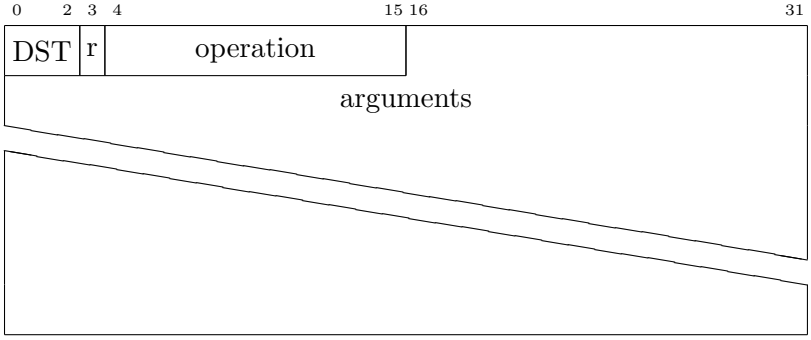

Figure 3.1: Packet structure

3.1 Destiny

The first argument (*DST*) will be the destiny of that packet. This will specify one of the 4 modules connected to WatchWolf Tester (for more information refer to Section 2, WatchWolf Introduction). Note that WatchWolf Tester itself is not present, as it will be indicated with the Response bit at 1. You can see the different *DST* values for each module om the Figure 3.1, DST bits meaning.

DST[2]	DST[1]	DST[0]	Destination
0	0	0	ServerManagerPetition
0	0	1	ServerPetition
0	1	0	ClientsManagerPetition
0	1	1	ClientPetition
1	X	X	*Reserved*

Table 3.1: DST bits meaning

3.2 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" (e.g. Error notification). Those will be sent directly marked as responses (Response bit at '1').

3.3 Operation

The Operation parameter specifies the desired request. Those change according to the Destiny, so they will be discussed in more detail in their respective sections.

The only exception is the all-zeroes operation (0b000000000000) which represents a NOP request. That way, if you need to perform a long test, you won't be kicked by inactivity[2] if you send this request every few minutes.

3.4 Arguments

The Arguments parameter specifies the arguments (if any) to the *Operation* request. Those change according to the Destiny, so the amount of arguments, and their types and order will be discussed in more detail in their respective sections.

Now there will be discussed the most common data types, so they will be independent of any programming language.

3.4.1 Character

Characters are sent as a 1-byte integer, representing its ASCII ref? value.

3.4.2 Boolean

Booleans are 1-bit element that represents *true* (0b1), or *false* (0b0).

For alignment define? reasons, booleans will be sent as 1-byte element. To avoid misunderstandings, let's define *false* as 0x00, and *true* as '*not define? false*'. That way, this two packets are valid *true* elements:

[2] This is a safety mechanism to avoid blocking a server to the same user forever. Besides being defined by the API it hasn't been implemented yet, and won't be until WatchWolf offers public servers.

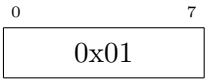

Figure 3.2: True packet with the LSB at 1

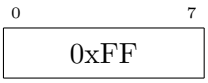

Figure 3.3: True packet with all bits at 1

3.4.3 Double

Doubles are 8-bytes floating-point numbers. They are represented following the IEEE 754[3].

3.4.4 String

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

3.4.5 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.

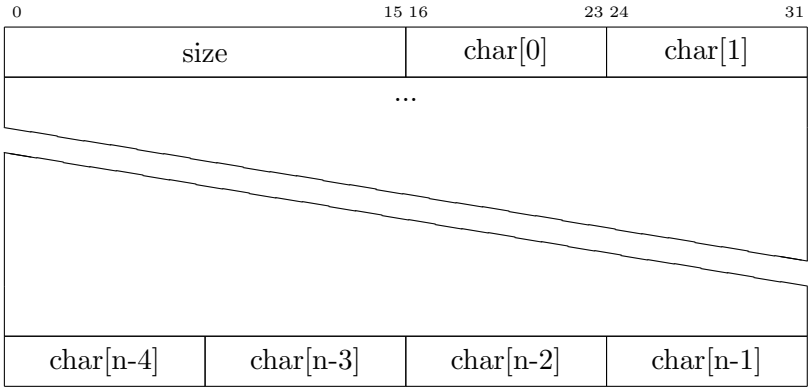

Figure 3.4: Structure of a String

[3]This standard should be used by C, Java and Python. cite?

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 3.5, Example of a string array.

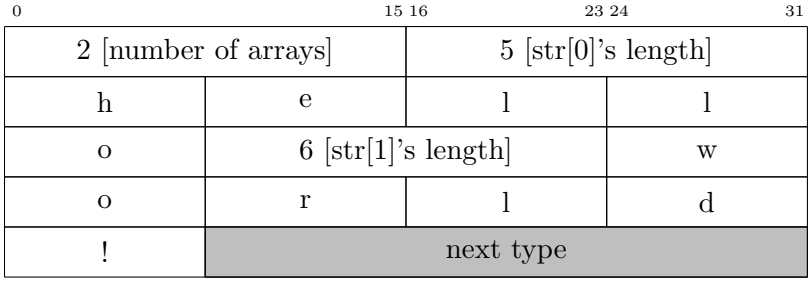

Figure 3.5: Example of a string array

3.4.6 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 definition[4] is 4GB.

[4]Besides 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.

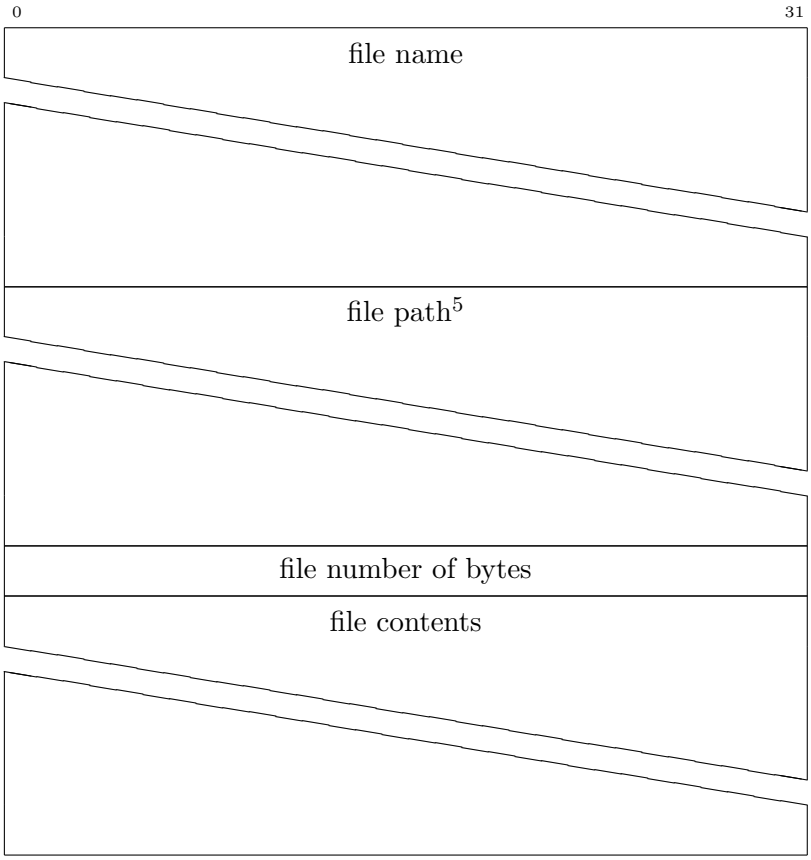

Figure 3.6: File structure

3.4.7 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.

3.4.8 Position

...

[5]The path must be relative, and you can't go outside the Server directory (using '../'). Both '' and './' means the root of the Server directory. adding '' compatilibity

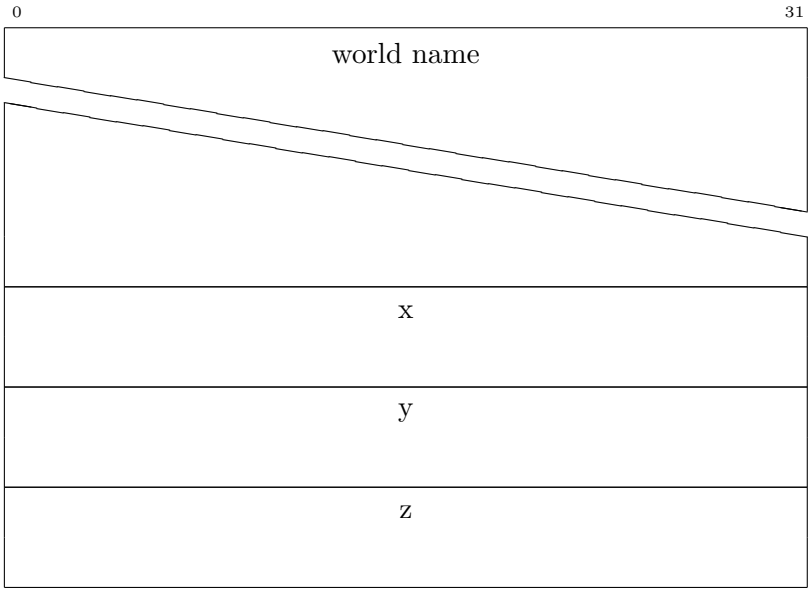

Figure 3.7: Position structure

3.4.9 Block

... 56 bytes

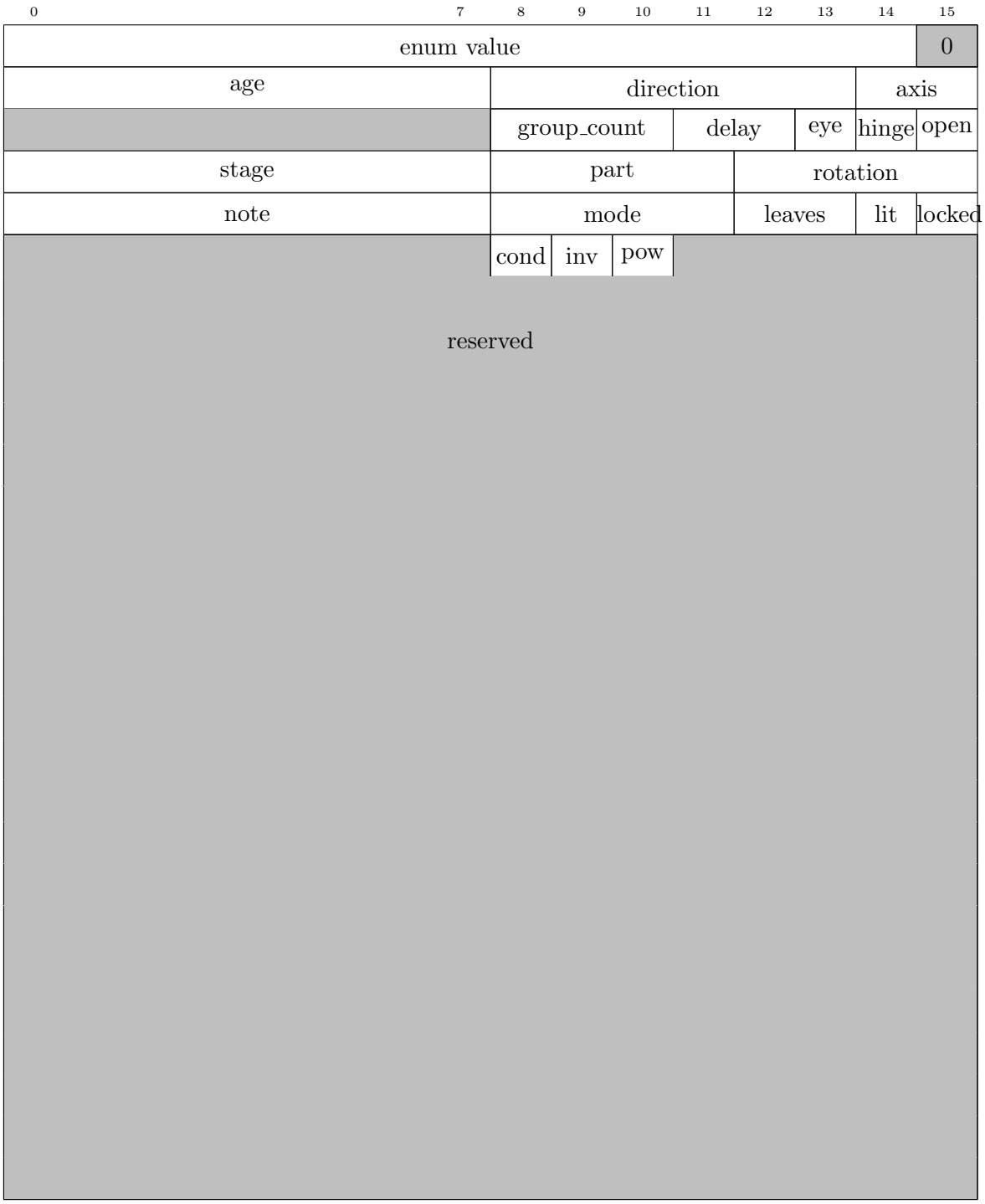

Figure 3.8: Structure of a Block

Enum value	Block name	First Minecraft version
0	AIR	1.8

Table 3.2: Block enum

3.4.10 Item

...

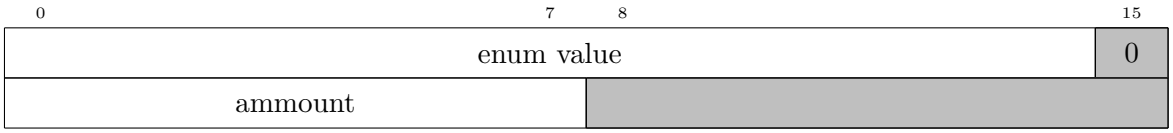

Figure 3.9: Structure of an Item

Enum value	Item name	First Minecraft version
0	AIR	1.8

Table 3.3: Item enum

3.4.11 Entity

...

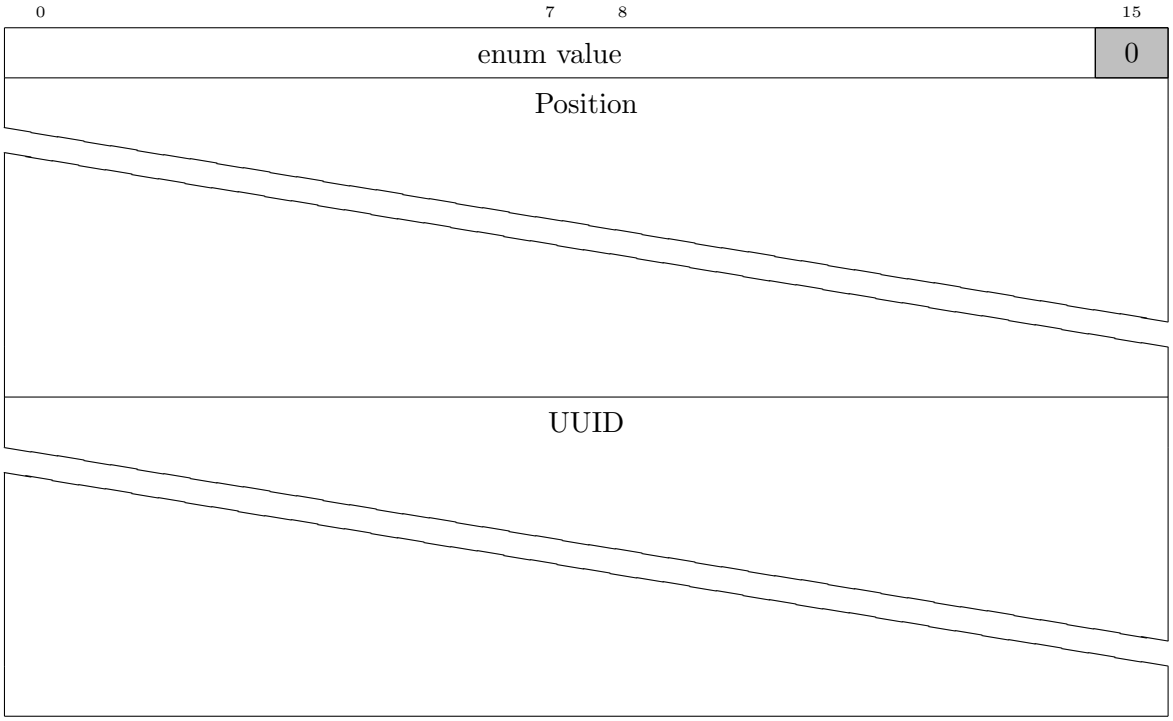

Figure 3.10: Structure of an Entity

Enum value	Entity type	First Minecraft version
0	DROPPED_ITEM	1.8

Table 3.4: Entity enum

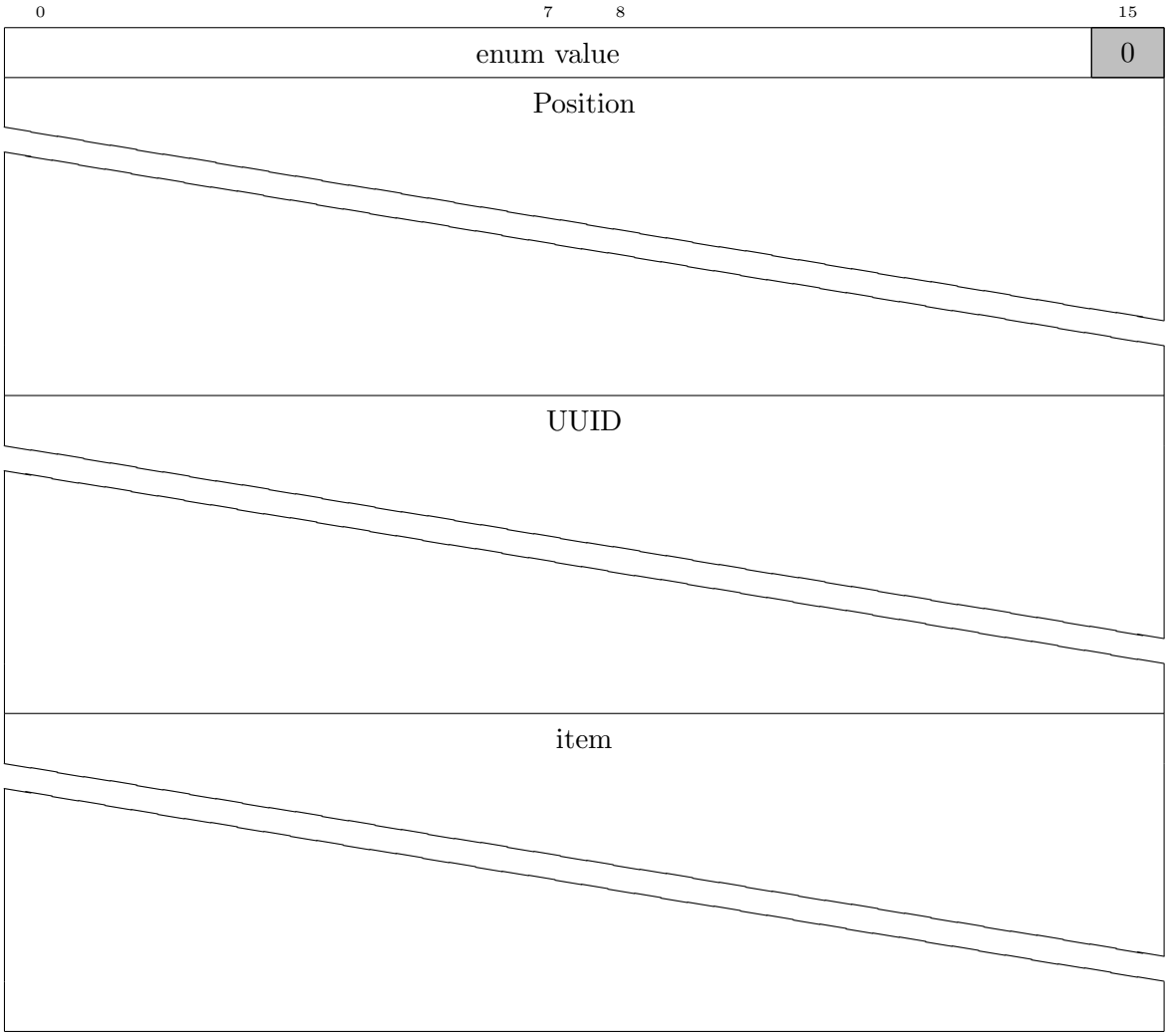

Figure 3.11: Structure of a DroppedItem

3.4.12 Container

...

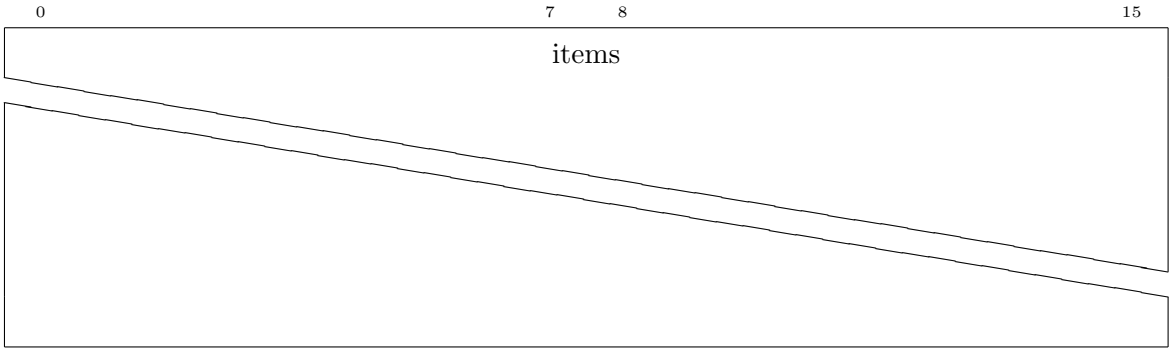

Figure 3.12: Structure of a Container

Enum value	Entity name	First Minecraft version
0	ZOMBIE	1.8

Table 3.5: Entity enum

4 Server manager petition

...

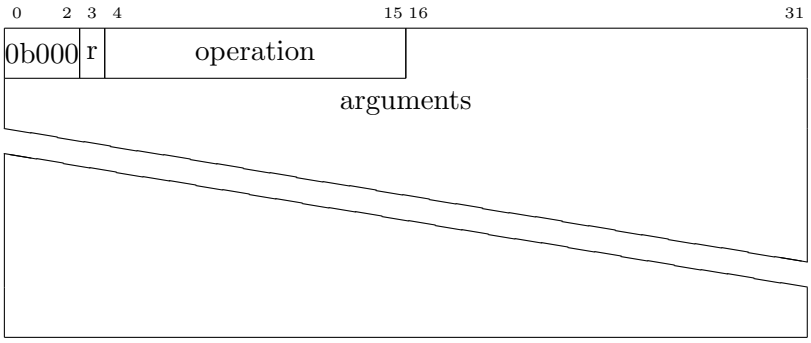

Figure 4.1: Server manager petition structure

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.

4.1 Start server operation

...

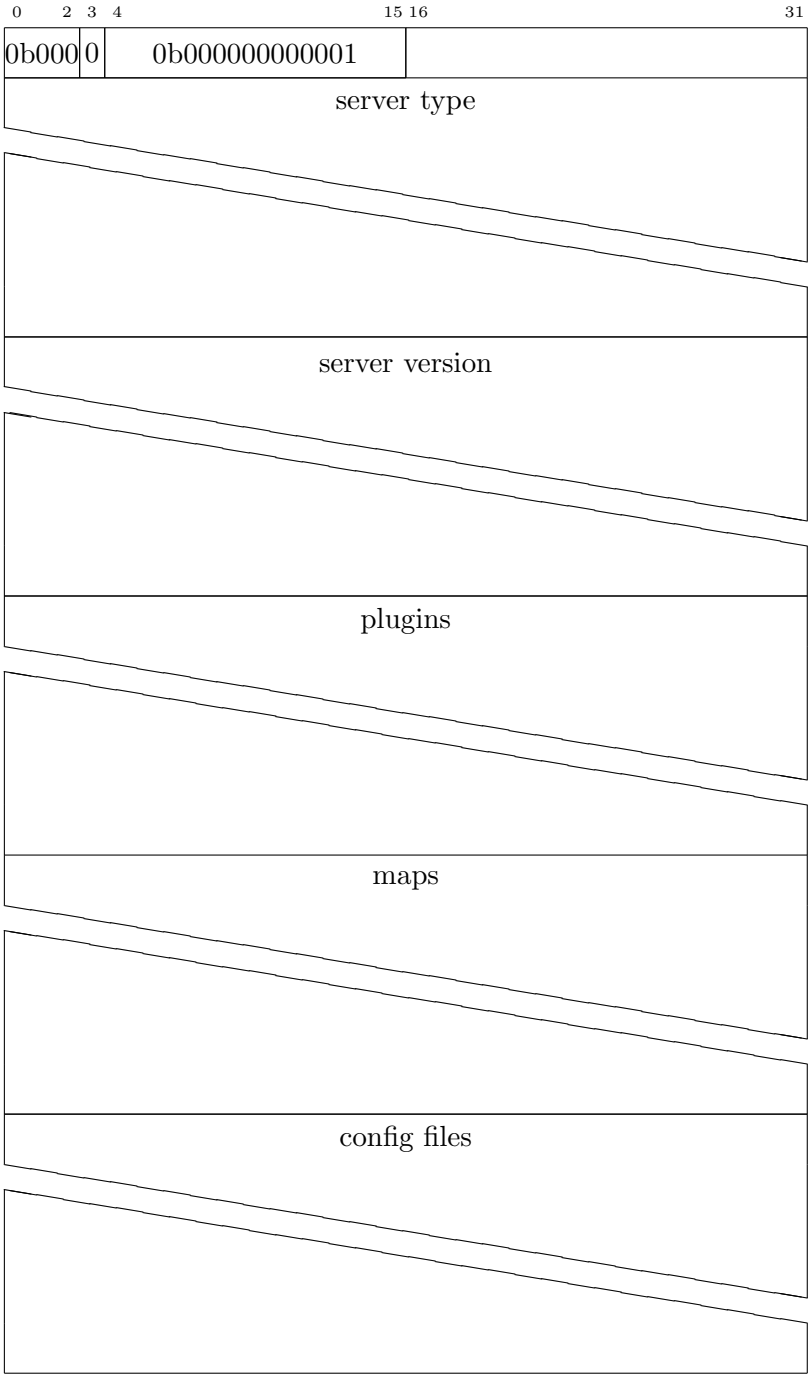

Figure 4.2: Start server petition structure

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 4.3, 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 4.4, Start server error response structure).

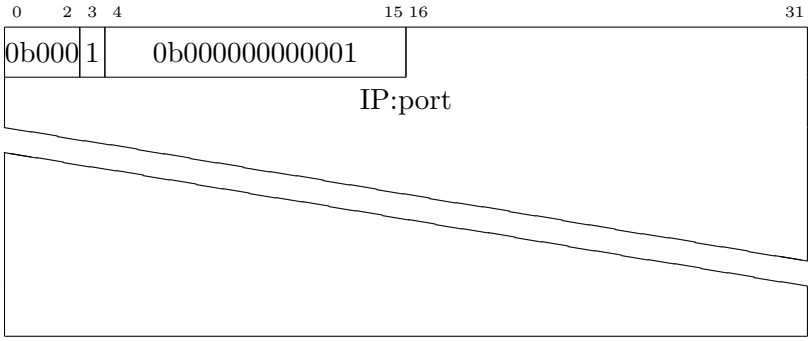

Figure 4.3: Start server response structure

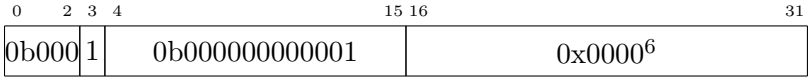

Figure 4.4: Start server error response structure

4.1.1 Maps

Array of maps (worlds; Map[]). To have more information about arrays check the subsection 3.4.5, 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

[6] Being 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.

4.1.2 Plugins

Array of plugins (Plugin[]). To have more information check the subsection 3.4.5, 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 plugins[7] 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 4.5, 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.

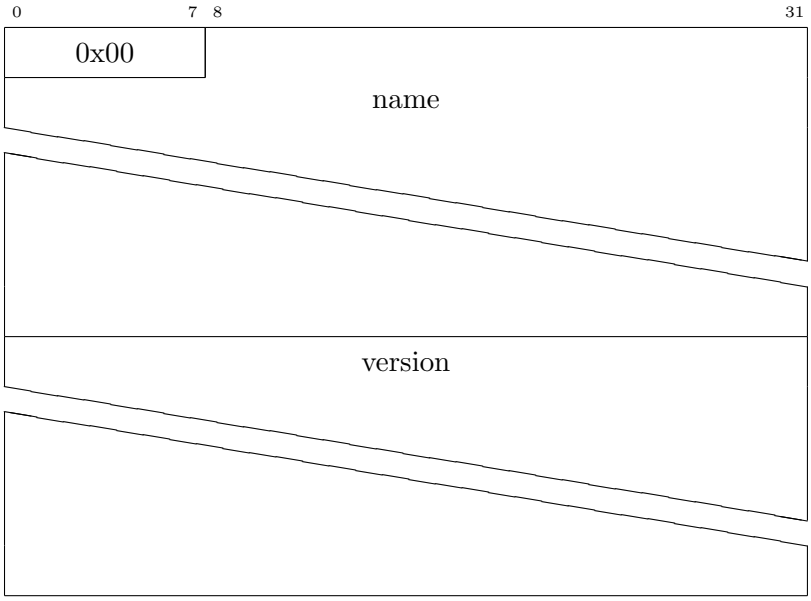

Figure 4.5: Usual plugin structure

[7]Premium 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 4.6, File plugin structure, the first argument (that specifies the Plugin type) is 0x02.

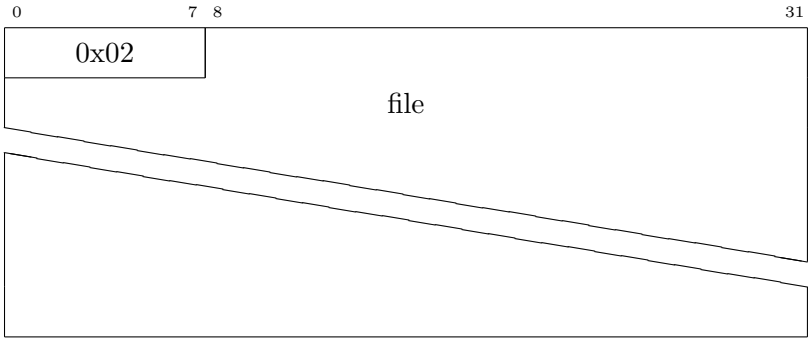

Figure 4.6: File plugin structure

mixed plugin types example?

4.1.3 Server version

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

4.1.4 Config files

...

4.2 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.

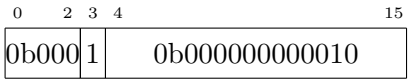

Figure 4.7: Server started notification structure

4.3 Error notification

...

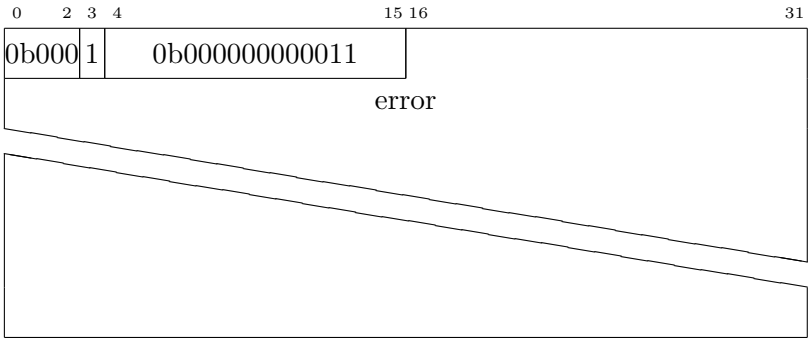

Figure 4.8: Error notification structure

4.4 Get coverage operation

... get JaCoCo binary file

4.4.1 Get coverage request

...

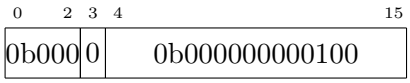

Figure 4.9: Get coverage operation structure

4.4.2 Get coverage response

...

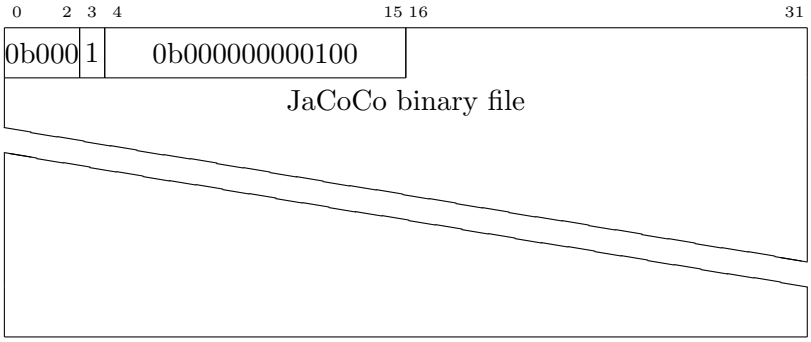

Figure 4.10: Get coverage response structure

5 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 3.1, Packet structure) becomes the group, and then the operation is defined on the next 2 bytes, as shown in the Figure 5.1, Server petition structure.

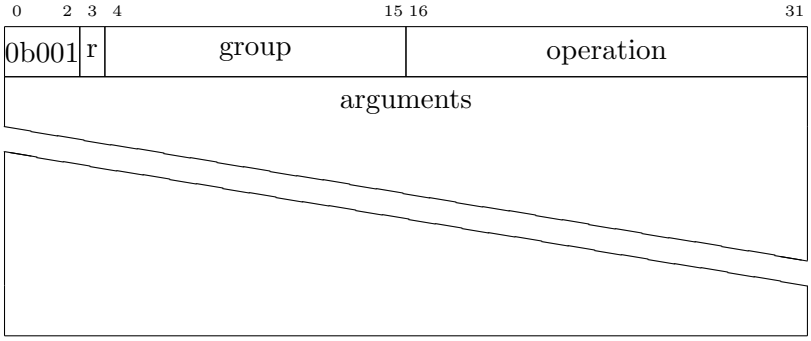

Figure 5.1: Server petition structure

5.1 Server petition group

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

The MSB 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 0b000000000001 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	0b00000000000	NOP[8]
0	0b00000000001	Base operations
0	0b00000000010	System operations
0	0b00000000011	Performance operations
0	0b00000010000	WorldGuard operations
0	0b00000010001	Residence operations
1	XXXXXXXXXXX	Reserved for internal use

Table 5.1: Extended types

If you've implemented an extended type and you believe that it makes sense to be part of the standard contact contact@watchwolf.dev to reserve one of the addresses.

5.2 Server petition operation

Like the parameter Operation, it specifies the desired request. For more information, refer to the subsection 3.3, Operation.

The only reserved operation is the all-zeroes operation (0x0000). It represents the question 'is this extended petition implemented?'. The server must response (with the response bit at 1) with *true* (group implemented on this machine) or *false* (unknown/unimplemented group), as it can be seen in Figure 5.2, Implemented group response structure.

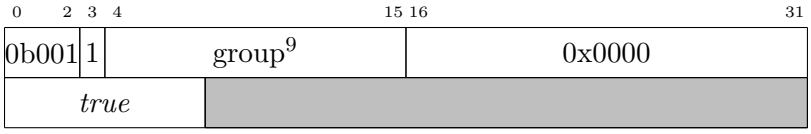

Figure 5.2: Implemented group response structure

5.3 Base operations

...

'is implemented' (all zeroes) optional

[8] As stated on the subsection 3.3, Operation, the all-zeroes operation represents a NOP request.

[9] except for groups 0b000000000000 and 0b000000000001

5.3.1 Server stop operation

...

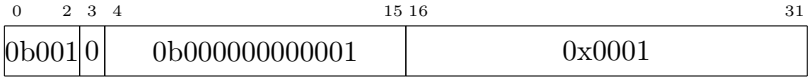

Figure 5.3: Stop server operation structure

5.3.2 Server stopped notification

... response to...

To have more information about the *server id* parameter check the Subsection 5.3.3, Server started notification.

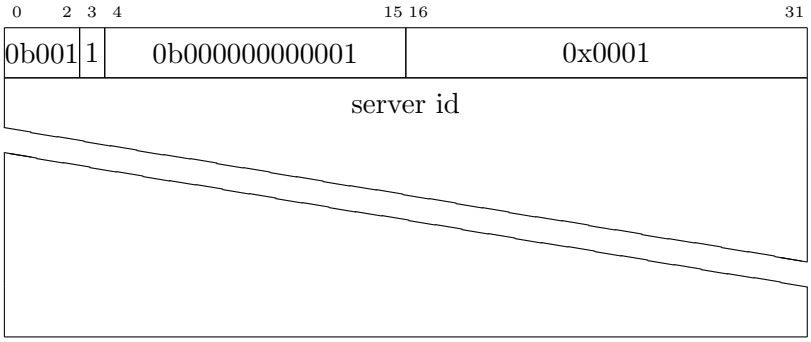

Figure 5.4: Server stopped response structure

5.3.3 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.

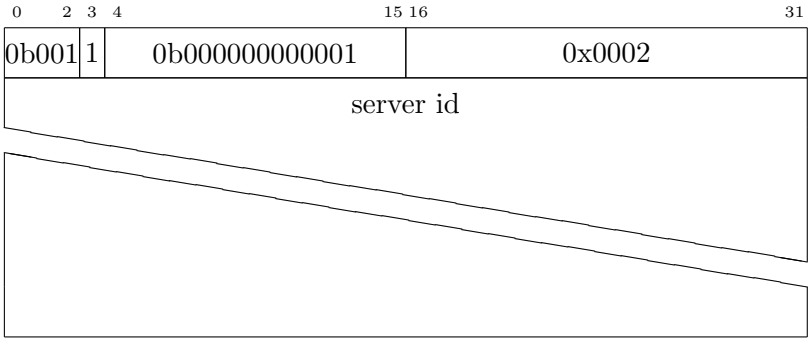

Figure 5.5: Server started response structure

5.3.4 Whitelist player operation

...

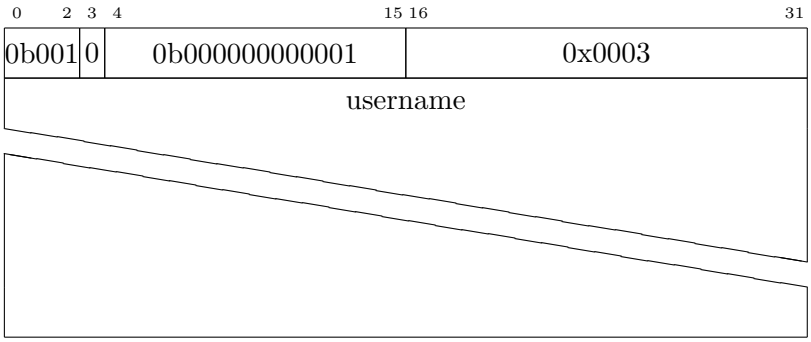

Figure 5.6: Whitelist player operation structure

5.3.5 OP player operation

...

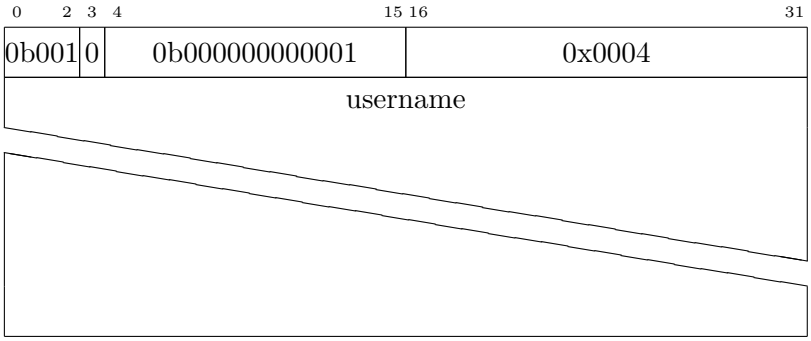

Figure 5.7: OP player operation structure

5.3.6 Set block operation

...

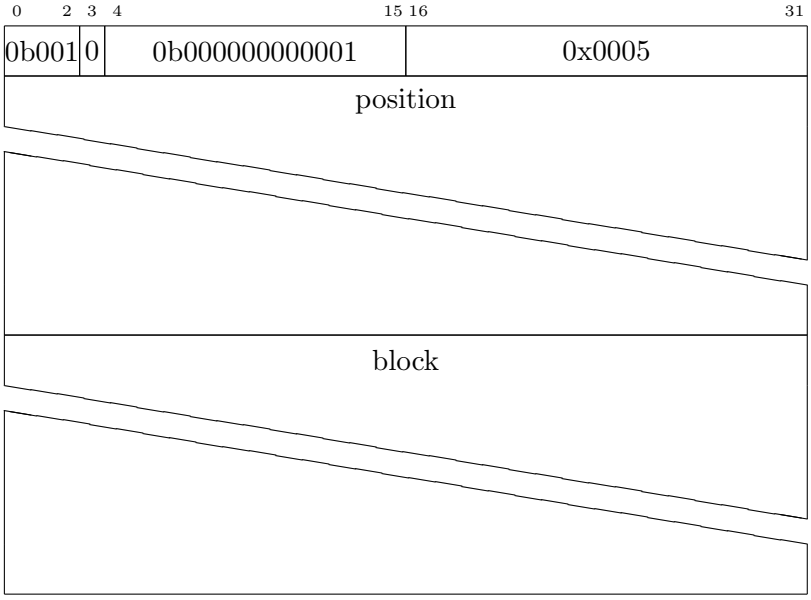

Figure 5.8: Set block operation structure

5.3.7 Get block operation

...

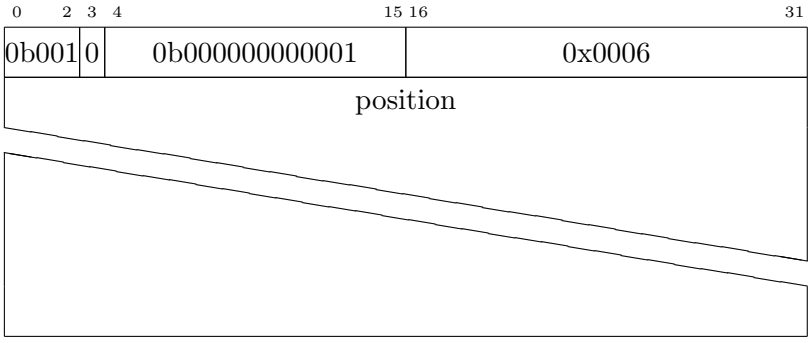

Figure 5.9: Get block operation structure

5.3.8 Get block response

...

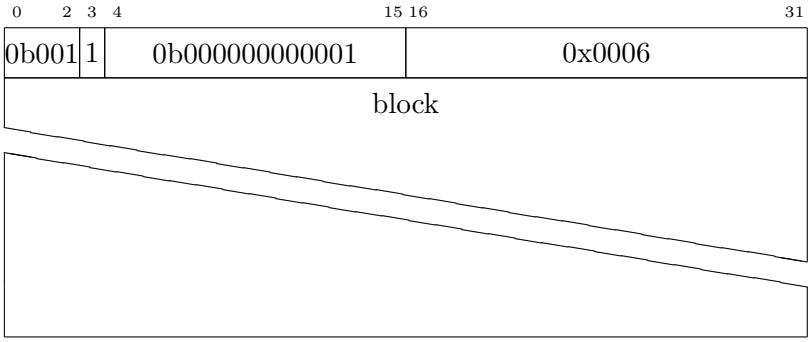

Figure 5.10: Get block operation structure

5.3.9 Get player position operation

...

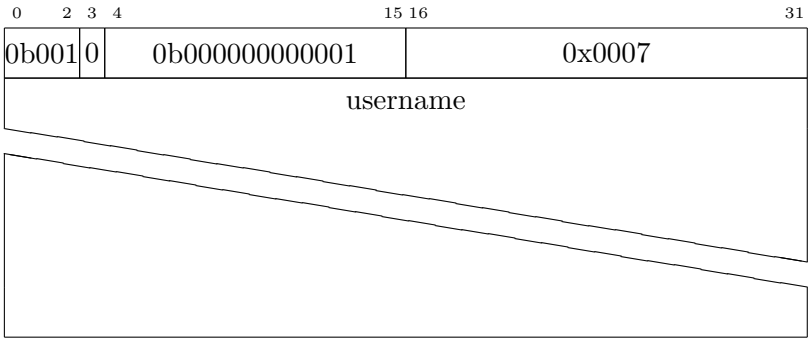

Figure 5.11: Get player position operation structure

5.3.10 Get player position response

...

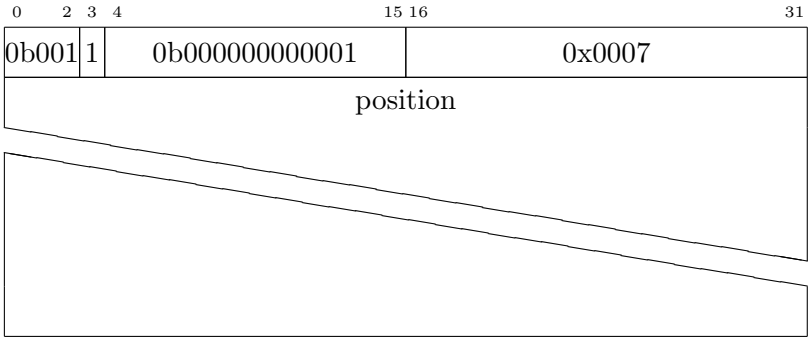

Figure 5.12: Get player position response structure

5.3.11 Give item to player operation

...

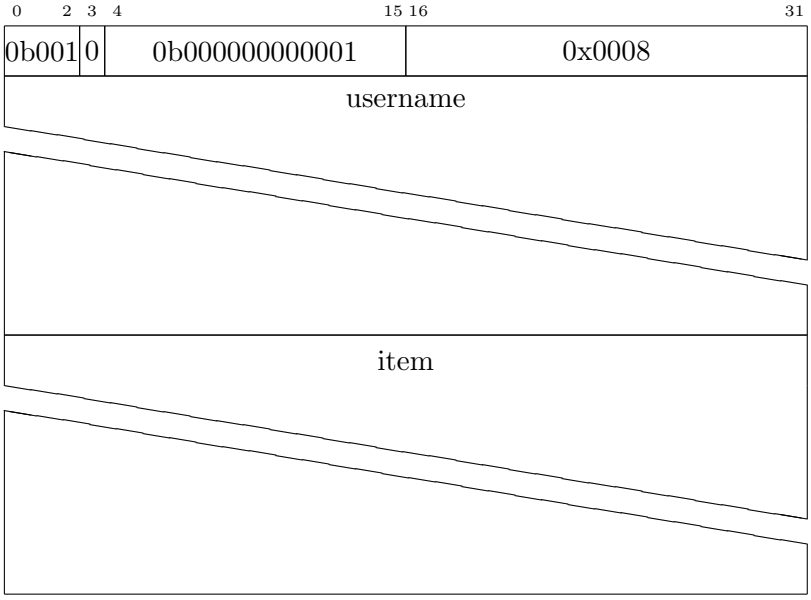

Figure 5.13: Give item to player operation structure

5.3.12 Run command operation

...

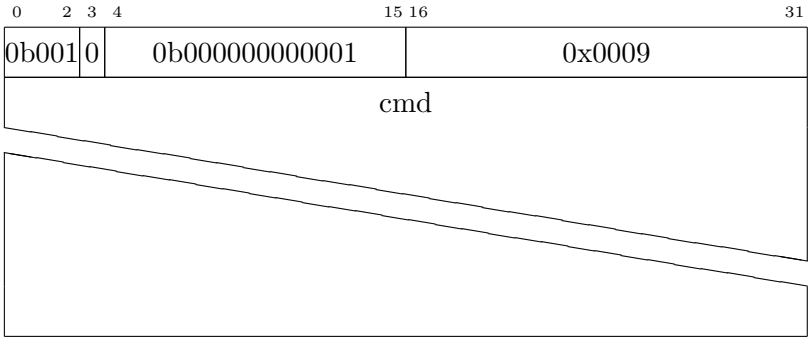

Figure 5.14: Run command operation structure

5.3.13 Get players operation

...

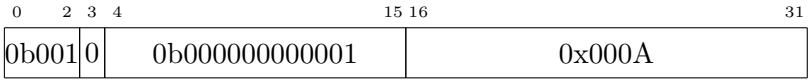

Figure 5.15: Get players operation structure

5.3.14 Get players response

...

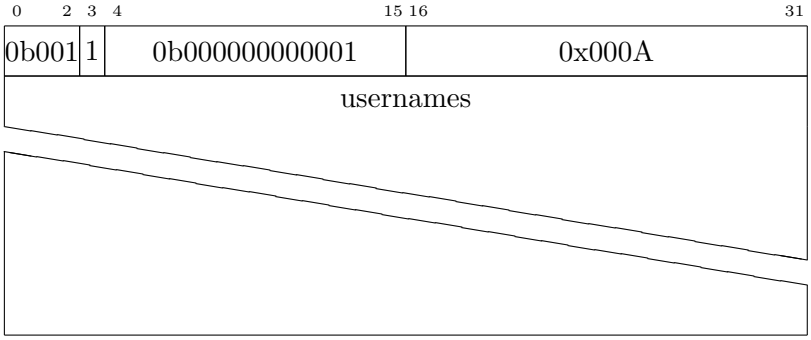

Figure 5.16: Get players response structure

5.3.15 Synchronize operation

...

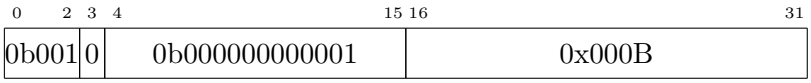

Figure 5.17: Synchronize operation structure

5.3.16 Synchronize response

...

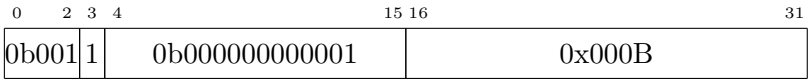

Figure 5.18: Synchronize response structure

5.3.17 Tp player operation

...

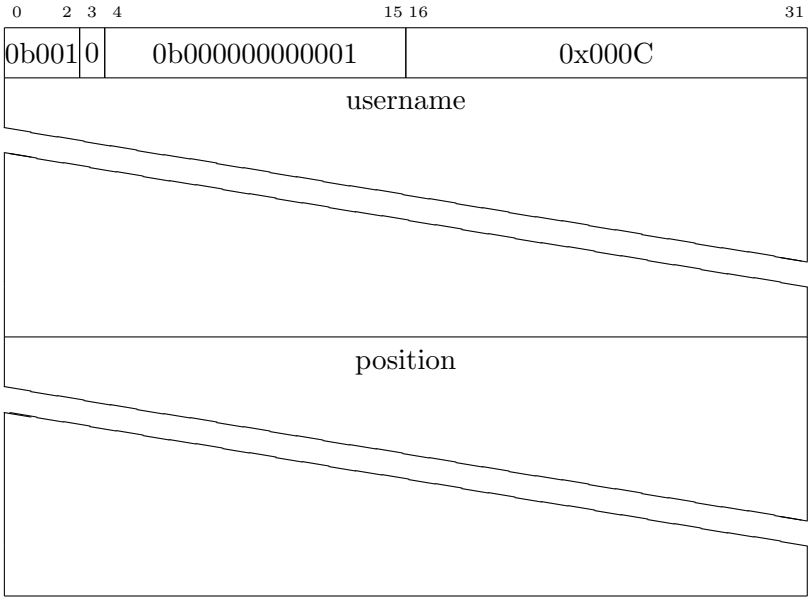

Figure 5.19: Tp player operation structure

5.3.18 Get player pitch operation

...

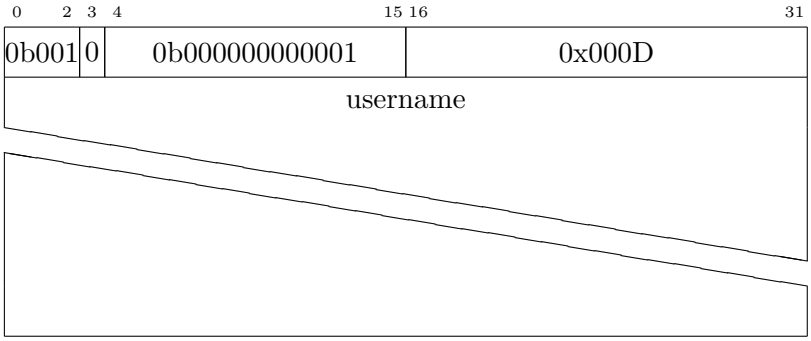

Figure 5.20: Get player pitch operation structure

5.3.19 Get player pitch response

...

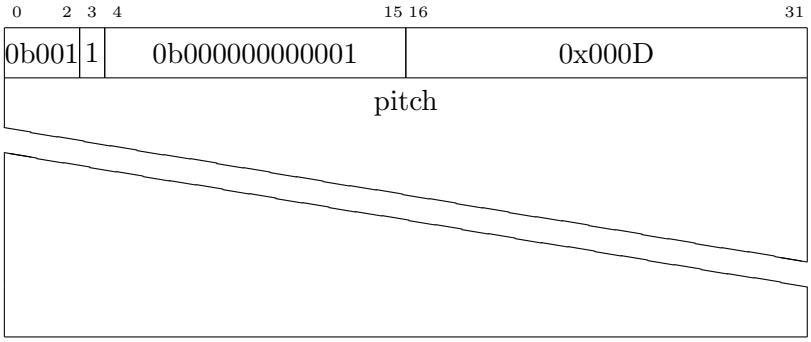

Figure 5.21: Get player pitch response structure

5.3.20 Get player yaw operation

...

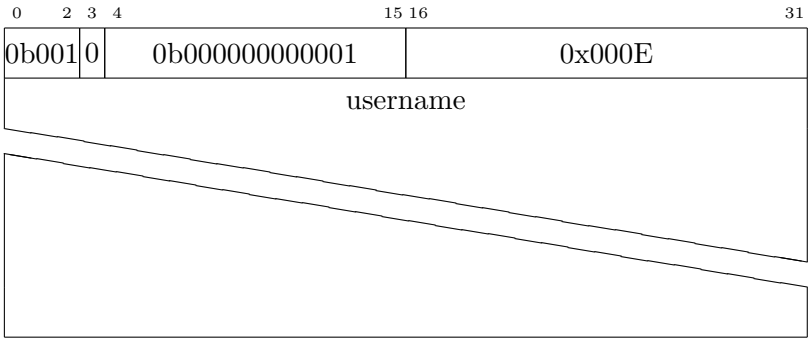

Figure 5.22: Get player yaw operation structure

5.3.21 Get player yaw response

...

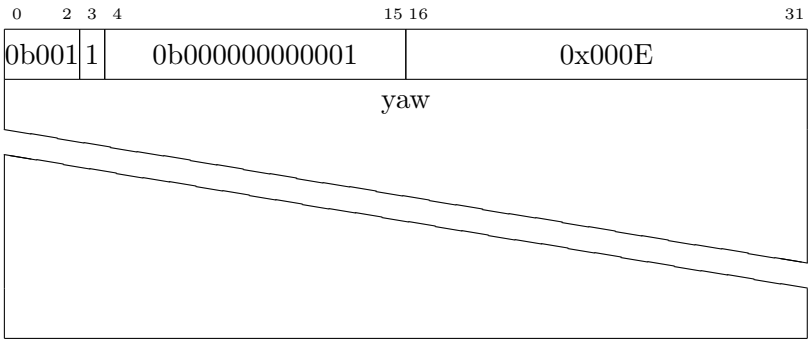

Figure 5.23: Get player yaw response structure

5.3.22 Get player inventory operation

...

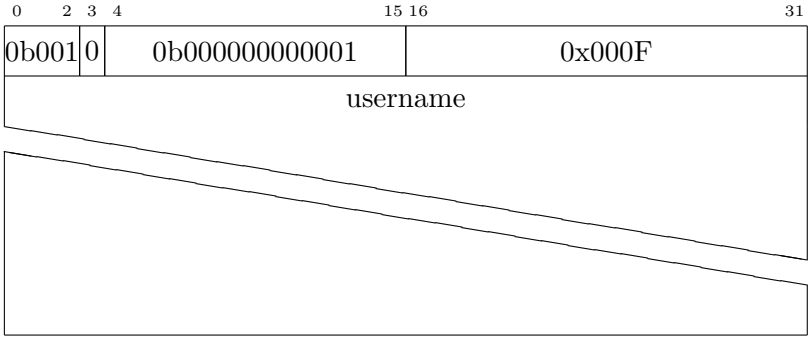

Figure 5.24: Get player inventory operation structure

5.3.23 Get player inventory response

...

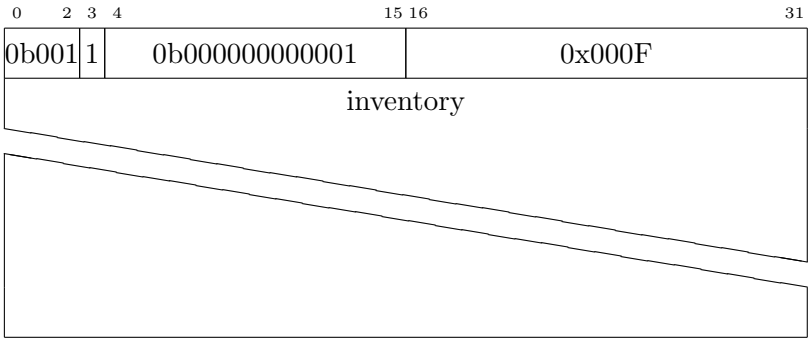

Figure 5.25: Get player inventory response structure

5.3.24 Get entities operation

...

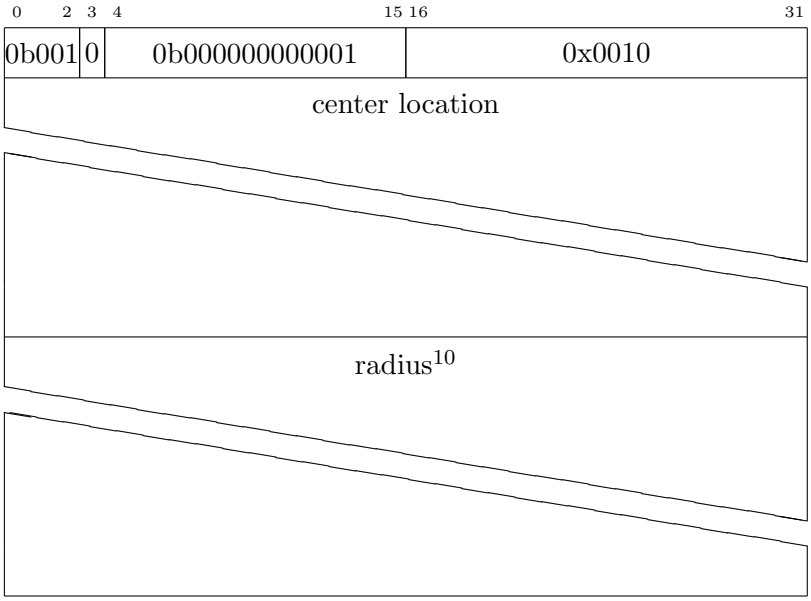

Figure 5.26: Get entities operation structure

5.3.25 Get entities response

...

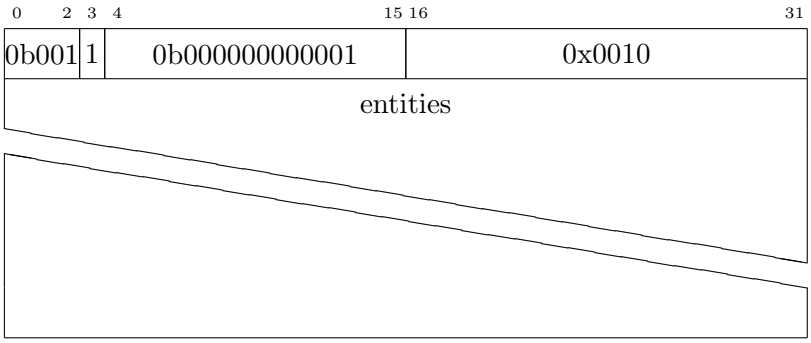

Figure 5.27: Get entities response structure

5.3.26 Spawn entity operation

...

[10]Large radius or locations in unloaded chunks may return unexpected data.

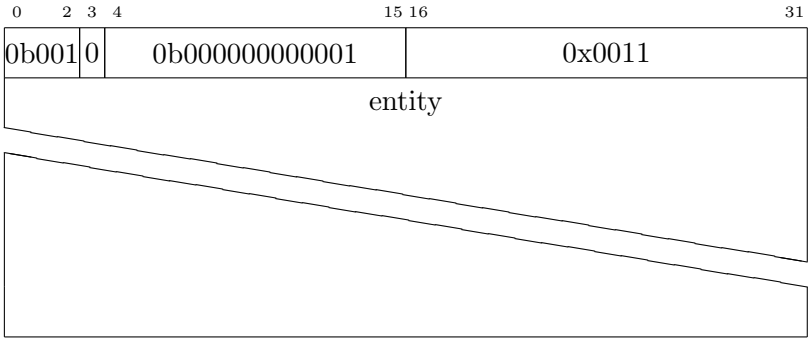

Figure 5.28: Spawn entity operation structure

5.3.27 Spawn entity response

...

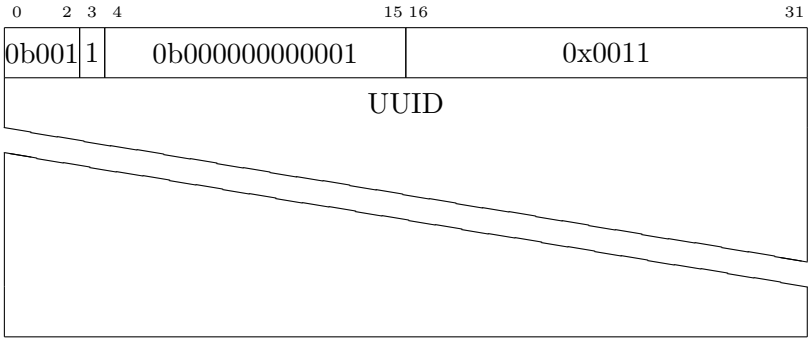

Figure 5.29: Spawn entity response structure

5.4 Performance operations

...

5.5 WorldGuard operations

...

5.6 Residence operations

...

6 Clients manager petition

...

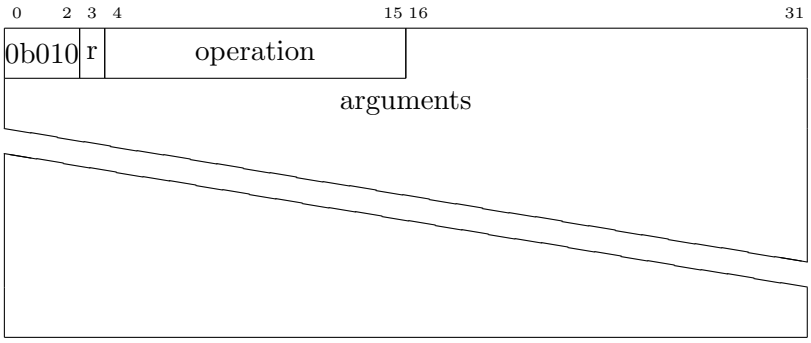

Figure 6.1: Clients manager petition structure

6.1 Start client operation

...

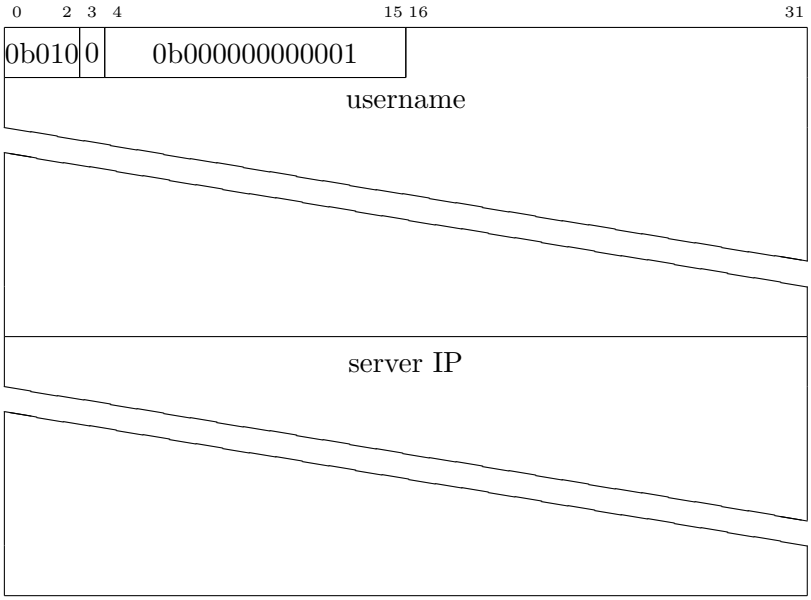

Figure 6.2: Start client petition structure

6.2 Client started response operation

...

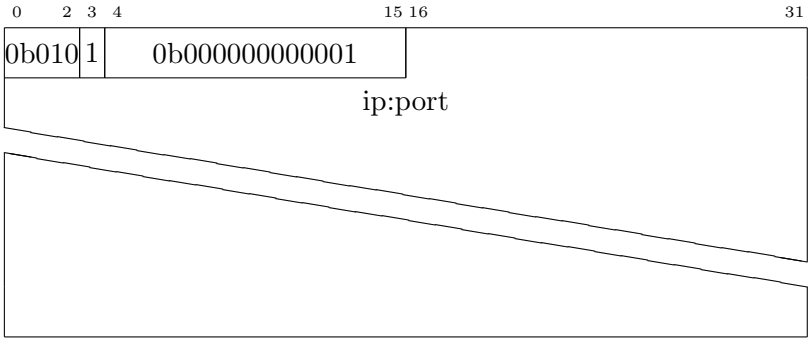

Figure 6.3: Client started petition response structure

7 Client petition

...

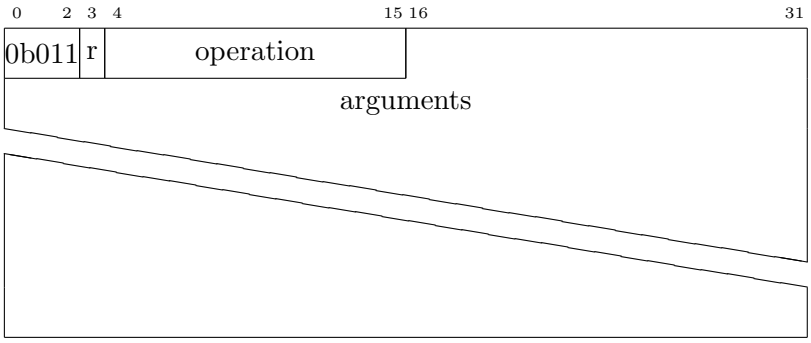

Figure 7.1: Client petition structure

7.1 Stop client operation

...

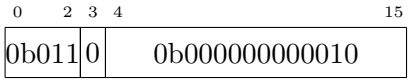

Figure 7.2: Stop client petition structure

7.2 Send message operation

... don't use this to run commands

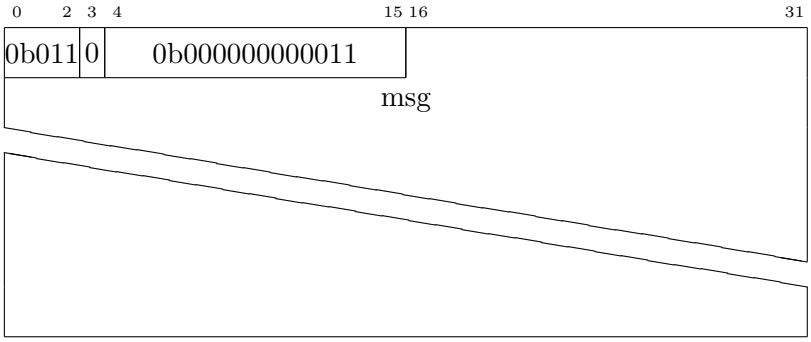

Figure 7.3: Send message petition structure

7.3 Got message notification

... async

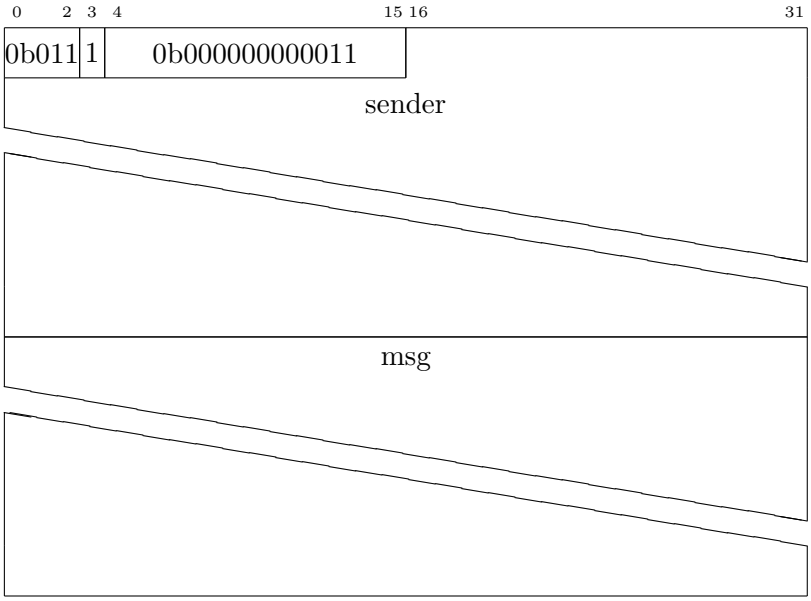

Figure 7.4: Got message notification structure

7.4 Run command operation

...

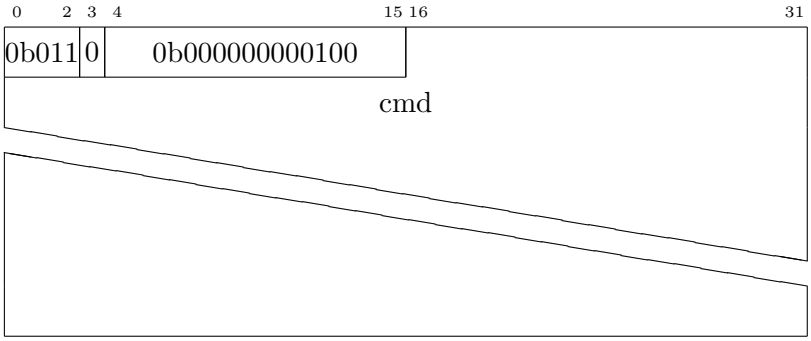

Figure 7.5: Run command petition structure

7.5 Break block operation

...

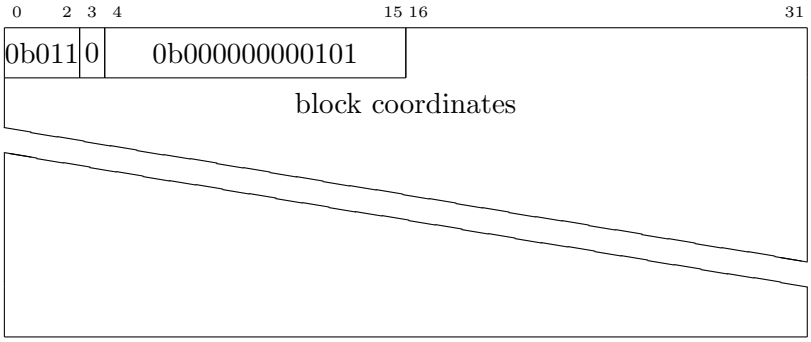

Figure 7.6: Run command petition structure

7.6 Equip item in hand operation

...

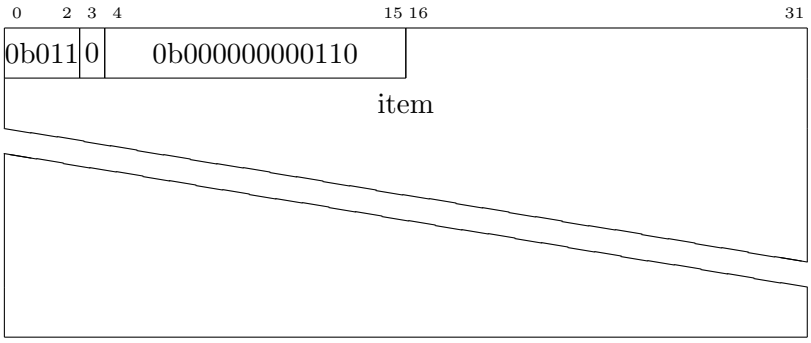

Figure 7.7: Run command petition structure

7.7 Move at operation

...

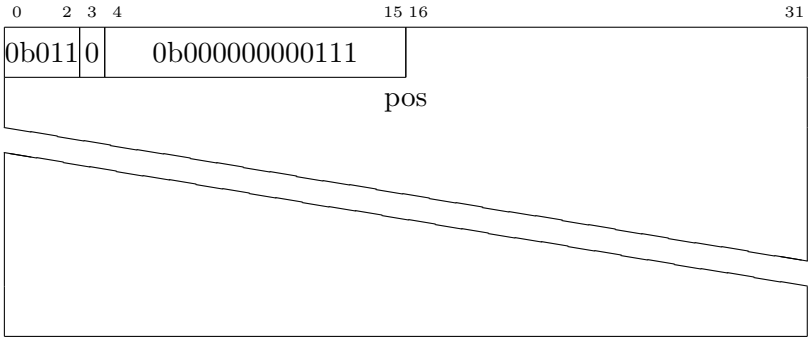

Figure 7.8: Move at petition structure

7.8 Look at operation

...

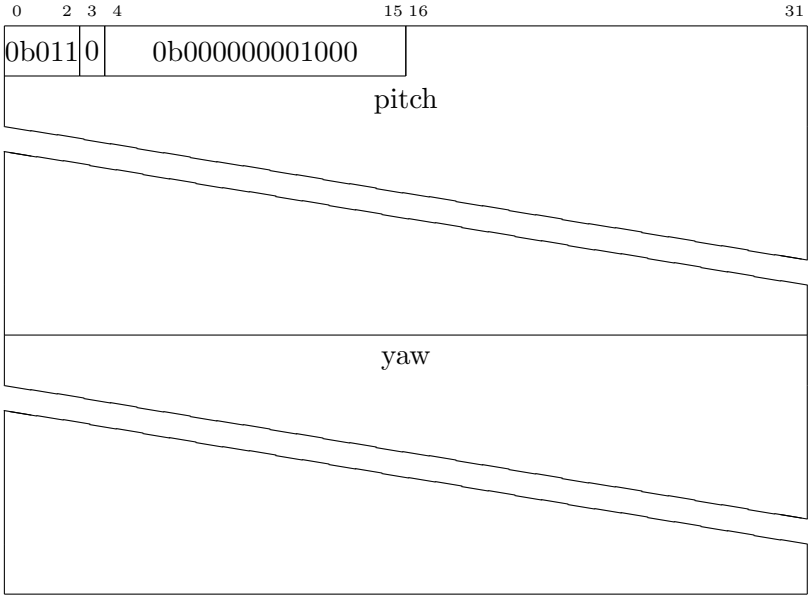

Figure 7.9: Move at petition structure

7.9 Synchronize operation

...

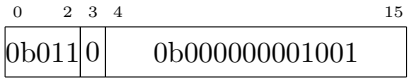

Figure 7.10: Synchronize petition structure

7.10 Synchronize response

...

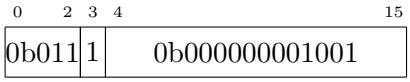

Figure 7.11: Synchronize petition response structure

7.11 Hit operation

... left click with item in hand

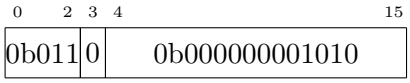

Figure 7.12: Hit petition structure

7.12 Use operation

... right click with item in hand

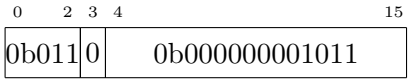

Figure 7.13: Use petition structure

7.13 Place block operation

... place current item in hand

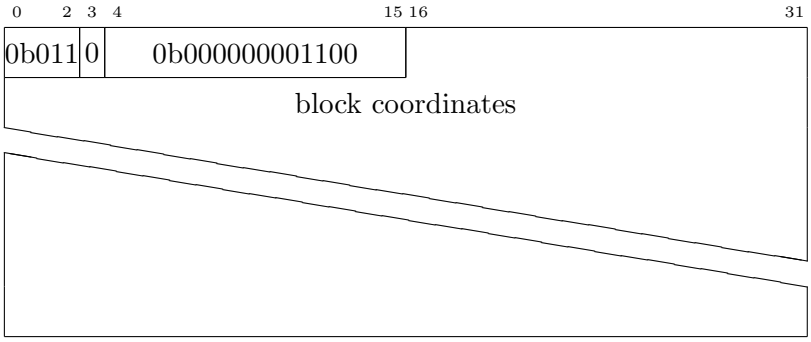

Figure 7.14: Place block petition structure

7.14 Hit entity operation

... attack

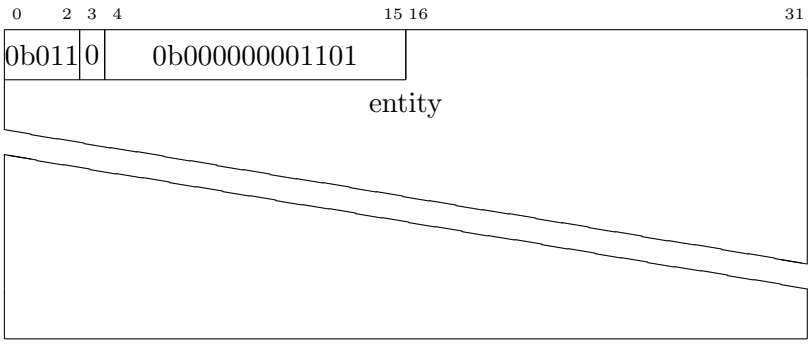

Figure 7.15: Place block petition structure

8 Revision history

Date	Revision	Changes
date	1	Initial public release, *v0.1.14*.

Table 8.1: 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 *Spigot - Enum Material - 1.19* (n.d.).

A.1 Unused material 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
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
has_book	Inventory dependent
sculk_sensor_phase	Admin block

Modifier name	Reason for discarding
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
hatch	Unable to concatenate
thickness	Block dependent
snowy	Block dependent
in_wall	Same block
moisture	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
WEEPING_VINES	age
ANDESITE_WALL	up
BLACKSTONE_WALL	up
BRICK_WALL	up
COBBLED_DEEPSLATE_WALL	up
COBBLESTONE_WALL	up

Block name	Modifier name
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
ACACIA_DOOR	powered
ACACIA_FENCE_GATE	powered
ACACIA_TRAPDOOR	powered
ACTIVATOR_RAIL	powered
BELL	powered
BIRCH_DOOR	powered
BIRCH_FENCE_GATE	powered
BIRCH_TRAPDOOR	powered
CRIMSON_DOOR	powered
CRIMSON_FENCE_GATE	powered
CRIMSON_TRAPDOOR	powered
DARK_OAK_DOOR	powered
DARK_OAK_FENCE_GATE	powered

Block name	Modifier name
DARK_OAK_TRAPDOOR	powered
IRON_DOOR	powered
IRON_TRAPDOOR	powered
JUNGLE_DOOR	powered
JUNGLE_FENCE_GATE	powered
JUNGLE_TRAPDOOR	powered
LECTERN	powered
MANGROVE_DOOR	powered
MANGROVE_FENCE_GATE	powered
MANGROVE_TRAPDOOR	powered
NOTE_BLOCK	powered
OAK_DOOR	powered
OAK_FENCE_GATE	powered
OAK_TRAPDOOR	powered
POWERED_RAIL	powered
SPRUCE_DOOR	powered
SPRUCE_FENCE_GATE	powered
SPRUCE_TRAPDOOR	powered
TRIPWIRE	powered
WARPED_DOOR	powered
WARPED_FENCE_GATE	powered
WARPED_TRAPDOOR	powered
SMOKER	lit
FURNACE	lit
CHEST	type
TRAPPED_CHEST	type

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

A.2 Material modifiers

A.2.1 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
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.2.2 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.2.3 Axis

Represents the axis along whilst this block is oriented.

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

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

Material	Age range
STRIPPED_ACACIA_LOG	x/y/z
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.2.4 Berries

Indicates whether the block has berries.

Defaults to false.

Material	Values
CAVE_VINES	true/false
CAVE_VINES_PLANT	true/false

Table A.6: Materials with berries

A.2.5 Bites

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

Defaults to 0.

Material	Values
CAKE	0-6

Table A.7: Cake

A.2.6 Candles

Represents the number of candles which are present.

Defaults to 1.

combine with cake if ¿0

Material	Values
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

Material	Values
PINK_CANDLE	1-4
PURPLE_CANDLE	1-4
RED_CANDLE	1-4
WHITE_CANDLE	1-4
YELLOW_CANDLE	1-4

Table A.8: Materials with candles

A.2.7 Charges

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

Defaults to 0.

Material	Values
RESPAWN_ANCHOR	0-4

Table A.9: Charged materials

A.2.8 Conditional

Denotes whether this command block is conditional or not.

Defaults to false.

Material	Values
CHAIN_COMMAND_BLOCK	true/false
COMMAND_BLOCK	true/false
REPEATING_COMMAND_BLOCK	true/false

Table A.10: Conditionable materials

A.2.9 Delay

Propagation delay of a repeater.

Defaults to 1.

Material	Values
REPEATER	1-4

Table A.11: Delayable materials

A.2.10 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.

Material	Values
CHORUS_PLANT	true/false
GLOW_LICHEN	true/false
SCULK_VEIN	true/false
BROWN_MUSHROOM_BLOCK	true/false
MUSHROOM_STEM	true/false
RED_MUSHROOM_BLOCK	true/false

Table A.12: Materials with down option

A.2.11 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**

Material	Options (default on bold)
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**
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

Material	Options (default on bold)
REDSTONE_WIRE	**none**/up/slide
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
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
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.13: Orientable materials

A.2.12 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

Table A.14: Materials with up option

A.2.13 Eggs

Number of eggs which appear in the block.

Defaults to 1.

Material	Values
TURTLE_EGG	1-4

Table A.15: Materials with eggs

A.2.14 Extended

Denotes whether the piston head is currently extended or not.

Defaults to false.

Material	Values
PISTON	true/false
STICKY_PISTON	true/false

Table A.16: Extendable materials

A.2.15 Eye

Defaults to false.

Material	Values
END_PORTAL_FRAME	true/false

Table A.17: Materials with eye

A.2.16 Face

Represents the face to which a lever or button is stuck.

WALL_SIGN

Defaults to wall.

Material	Directions
ACACIA_BUTTON	wall/floor/ceiling
BIRCH_BUTTON	wall/floor/ceiling
CRIMSON_BUTTON	wall/floor/ceiling
DARK_OAK_BUTTON	wall/floor/ceiling
GRINDSTONE	wall/floor/ceiling
JUNGLE_BUTTON	wall/floor/ceiling
LEVER	wall/floor/ceiling
MANGROVE_BUTTON	wall/floor/ceiling
OAK_BUTTON	wall/floor/ceiling
POLISHED_BLACKSTONE_BUTTON	wall/floor/ceiling
SPRUCE_BUTTON	wall/floor/ceiling
STONE_BUTTON	wall/floor/ceiling
WARPED_BUTTON	wall/floor/ceiling

Table A.18: Directional materials

A.2.17 Facing

Represents the face towards which the block is pointing.

Material	Options (default on bold)
HOPPER	**down**/north/south/east/west

Material	Options (default on bold)
OBSERVER	up/down/north/**south**/east/west
BARREL	up/down/**north**/south/east/west
CHAIN_COMMAND_BLOCK	up/down/**north**/south/east/west
COMMAND_BLOCK	up/down/**north**/south/east/west
DISPENSER	up/down/**north**/south/east/west
DROPPER	up/down/**north**/south/east/west
PISTON	up/down/**north**/south/east/west
PISTON_HEAD	up/down/**north**/south/east/west
REPEATING_COMMAND_BLOCK	up/down/**north**/south/east/west
STICKY_PISTON	up/down/**north**/south/east/west
ACACIA_BUTTON	**north**/south/east/west
ACACIA_DOOR	**north**/south/east/west
ACACIA_FENCE_GATE	**north**/south/east/west
ACACIA_STAIRS	**north**/south/east/west
ACACIA_TRAPDOOR	**north**/south/east/west
ACACIA_WALL_SIGN	**north**/south/east/west
ANDESITE_STAIRS	**north**/south/east/west
ANVIL	**north**/south/east/west
ATTACHED_MELON_STEM	**north**/south/east/west
ATTACHED_PUMPKIN_STEM	**north**/south/east/west
BEEHIVE	**north**/south/east/west
BEE_NEST	**north**/south/east/west
BELL	**north**/south/east/west
BIG_DRIPLEAF	**north**/south/east/west
BIG_DRIPLEAF_STEM	**north**/south/east/west
BIRCH_BUTTON	**north**/south/east/west
BIRCH_DOOR	**north**/south/east/west
BIRCH_FENCE_GATE	**north**/south/east/west
BIRCH_STAIRS	**north**/south/east/west
BIRCH_TRAPDOOR	**north**/south/east/west

Material	Options (default on bold)
BIRCH_WALL_SIGN	**north**/south/east/west
BLACKSTONE_STAIRS	**north**/south/east/west
BLACK_BED	**north**/south/east/west
BLACK_GLAZED_TERRACOTTA	**north**/south/east/west
BLACK_WALL_BANNER	**north**/south/east/west
BLAST_FURNACE	**north**/south/east/west
BLUE_BED	**north**/south/east/west
BLUE_GLAZED_TERRACOTTA	**north**/south/east/west
BLUE_WALL_BANNER	**north**/south/east/west
BRAIN_CORAL_WALL_FAN	**north**/south/east/west
BRICK_STAIRS	**north**/south/east/west
BROWN_BED	**north**/south/east/west
BROWN_GLAZED_TERRACOTTA	**north**/south/east/west
BROWN_WALL_BANNER	**north**/south/east/west
BUBBLE_CORAL_WALL_FAN	**north**/south/east/west
CAMPFIRE	**north**/south/east/west
CARVED_PUMPKIN	**north**/south/east/west
CHEST	**north**/south/east/west
CHIPPED_ANVIL	**north**/south/east/west
COBBLED_DEEPSLATE_STAIRS	**north**/south/east/west
COBBLESTONE_STAIRS	**north**/south/east/west
COCOA	**north**/south/east/west
COMPARATOR	**north**/south/east/west
CREEPER_WALL_HEAD	**north**/south/east/west
CRIMSON_BUTTON	**north**/south/east/west
CRIMSON_DOOR	**north**/south/east/west
CRIMSON_FENCE_GATE	**north**/south/east/west
CRIMSON_STAIRS	**north**/south/east/west
CRIMSON_TRAPDOOR	**north**/south/east/west
CRIMSON_WALL_SIGN	**north**/south/east/west

Material	Options (default on bold)
CUT_COPPER_STAIRS	**north**/south/east/west
CYAN_BED	**north**/south/east/west
CYAN_GLAZED_TERRACOTTA	**north**/south/east/west
CYAN_WALL_BANNER	**north**/south/east/west
DAMAGED_ANVIL	**north**/south/east/west
DARK_OAK_BUTTON	**north**/south/east/west
DARK_OAK_DOOR	**north**/south/east/west
DARK_OAK_FENCE_GATE	**north**/south/east/west
DARK_OAK_STAIRS	**north**/south/east/west
DARK_OAK_TRAPDOOR	**north**/south/east/west
DARK_OAK_WALL_SIGN	**north**/south/east/west
DARK_PRISMARINE_STAIRS	**north**/south/east/west
DEAD_BRAIN_CORAL_WALL_FAN	**north**/south/east/west
DEAD_BUBBLE_CORAL_WALL_FAN	**north**/south/east/west
DEAD_FIRE_CORAL_WALL_FAN	**north**/south/east/west
DEAD_HORN_CORAL_WALL_FAN	**north**/south/east/west
DEAD_TUBE_CORAL_WALL_FAN	**north**/south/east/west
DEEPSLATE_BRICK_STAIRS	**north**/south/east/west
DEEPSLATE_TILE_STAIRS	**north**/south/east/west
DIORITE_STAIRS	**north**/south/east/west
DRAGON_WALL_HEAD	**north**/south/east/west
ENDER_CHEST	**north**/south/east/west
END_PORTAL_FRAME	**north**/south/east/west
END_STONE_BRICK_STAIRS	**north**/south/east/west
EXPOSED_CUT_COPPER_STAIRS	**north**/south/east/west
FIRE_CORAL_WALL_FAN	**north**/south/east/west
FURNACE	**north**/south/east/west
GRANITE_STAIRS	**north**/south/east/west
GRAY_BED	**north**/south/east/west
GRAY_GLAZED_TERRACOTTA	**north**/south/east/west

Material	Options (default on bold)
GRAY_WALL_BANNER	**north**/south/east/west
GREEN_BED	**north**/south/east/west
GREEN_GLAZED_TERRACOTTA	**north**/south/east/west
GREEN_WALL_BANNER	**north**/south/east/west
GRINDSTONE	**north**/south/east/west
HORN_CORAL_WALL_FAN	**north**/south/east/west
IRON_DOOR	**north**/south/east/west
IRON_TRAPDOOR	**north**/south/east/west
JACK_O_LANTERN	**north**/south/east/west
JUNGLE_BUTTON	**north**/south/east/west
JUNGLE_DOOR	**north**/south/east/west
JUNGLE_FENCE_GATE	**north**/south/east/west
JUNGLE_STAIRS	**north**/south/east/west
JUNGLE_TRAPDOOR	**north**/south/east/west
JUNGLE_WALL_SIGN	**north**/south/east/west
LADDER	**north**/south/east/west
LECTERN	**north**/south/east/west
LEVER	**north**/south/east/west
LIGHT_BLUE_BED	**north**/south/east/west
LIGHT_BLUE_GLAZED_TERRACOTTA	**north**/south/east/west
LIGHT_BLUE_WALL_BANNER	**north**/south/east/west
LIGHT_GRAY_BED	**north**/south/east/west
LIGHT_GRAY_GLAZED_TERRACOTTA	**north**/south/east/west
LIGHT_GRAY_WALL_BANNER	**north**/south/east/west
LIME_BED	**north**/south/east/west
LIME_GLAZED_TERRACOTTA	**north**/south/east/west
LIME_WALL_BANNER	**north**/south/east/west
LOOM	**north**/south/east/west
MAGENTA_BED	**north**/south/east/west
MAGENTA_GLAZED_TERRACOTTA	**north**/south/east/west

Material	Options (default on bold)
MAGENTA_WALL_BANNER	**north**/south/east/west
MANGROVE_BUTTON	**north**/south/east/west
MANGROVE_DOOR	**north**/south/east/west
MANGROVE_FENCE_GATE	**north**/south/east/west
MANGROVE_STAIRS	**north**/south/east/west
MANGROVE_TRAPDOOR	**north**/south/east/west
MANGROVE_WALL_SIGN	**north**/south/east/west
MOSSY_COBBLESTONE_STAIRS	**north**/south/east/west
MOSSY_STONE_BRICK_STAIRS	**north**/south/east/west
MUD_BRICK_STAIRS	**north**/south/east/west
NETHER_BRICK_STAIRS	**north**/south/east/west
OAK_BUTTON	**north**/south/east/west
OAK_DOOR	**north**/south/east/west
OAK_FENCE_GATE	**north**/south/east/west
OAK_STAIRS	**north**/south/east/west
OAK_TRAPDOOR	**north**/south/east/west
OAK_WALL_SIGN	**north**/south/east/west
ORANGE_BED	**north**/south/east/west
ORANGE_GLAZED_TERRACOTTA	**north**/south/east/west
ORANGE_WALL_BANNER	**north**/south/east/west
OXIDIZED_CUT_COPPER_STAIRS	**north**/south/east/west
PINK_BED	**north**/south/east/west
PINK_GLAZED_TERRACOTTA	**north**/south/east/west
PINK_WALL_BANNER	**north**/south/east/west
PLAYER_WALL_HEAD	**north**/south/east/west
POLISHED_ANDESITE_STAIRS	**north**/south/east/west
POLISHED_BLACKSTONE_BRICK_STAIRS	**north**/south/east/west
POLISHED_BLACKSTONE_BUTTON	**north**/south/east/west
POLISHED_BLACKSTONE_STAIRS	**north**/south/east/west
POLISHED_DEEPSLATE_STAIRS	**north**/south/east/west

Material	Options (default on bold)
POLISHED_DIORITE_STAIRS	**north**/south/east/west
POLISHED_GRANITE_STAIRS	**north**/south/east/west
PRISMARINE_BRICK_STAIRS	**north**/south/east/west
PRISMARINE_STAIRS	**north**/south/east/west
PURPLE_BED	**north**/south/east/west
PURPLE_GLAZED_TERRACOTTA	**north**/south/east/west
PURPLE_WALL_BANNER	**north**/south/east/west
PURPUR_STAIRS	**north**/south/east/west
QUARTZ_STAIRS	**north**/south/east/west
REDSTONE_WALL_TORCH	**north**/south/east/west
RED_BED	**north**/south/east/west
RED_GLAZED_TERRACOTTA	**north**/south/east/west
RED_NETHER_BRICK_STAIRS	**north**/south/east/west
RED_SANDSTONE_STAIRS	**north**/south/east/west
RED_WALL_BANNER	**north**/south/east/west
REPEATER	**north**/south/east/west
SANDSTONE_STAIRS	**north**/south/east/west
SKELETON_WALL_SKULL	**north**/south/east/west
SMALL_DRIPLEAF	**north**/south/east/west
SMOKER	**north**/south/east/west
SMOOTH_QUARTZ_STAIRS	**north**/south/east/west
SMOOTH_RED_SANDSTONE_STAIRS	**north**/south/east/west
SMOOTH_SANDSTONE_STAIRS	**north**/south/east/west
SOUL_CAMPFIRE	**north**/south/east/west
SOUL_WALL_TORCH	**north**/south/east/west
SPRUCE_BUTTON	**north**/south/east/west
SPRUCE_DOOR	**north**/south/east/west
SPRUCE_FENCE_GATE	**north**/south/east/west
SPRUCE_STAIRS	**north**/south/east/west
SPRUCE_TRAPDOOR	**north**/south/east/west

Material	Options (default on bold)
SPRUCE_WALL_SIGN	**north**/south/east/west
STONECUTTER	**north**/south/east/west
STONE_BRICK_STAIRS	**north**/south/east/west
STONE_BUTTON	**north**/south/east/west
STONE_STAIRS	**north**/south/east/west
TRAPPED_CHEST	**north**/south/east/west
TRIPWIRE_HOOK	**north**/south/east/west
TUBE_CORAL_WALL_FAN	**north**/south/east/west
WALL_TORCH	**north**/south/east/west
WARPED_BUTTON	**north**/south/east/west
WARPED_DOOR	**north**/south/east/west
WARPED_FENCE_GATE	**north**/south/east/west
WARPED_STAIRS	**north**/south/east/west
WARPED_TRAPDOOR	**north**/south/east/west
WARPED_WALL_SIGN	**north**/south/east/west
WAXED_CUT_COPPER_STAIRS	**north**/south/east/west
WAXED_EXPOSED_CUT_COPPER_STAIRS	**north**/south/east/west
WAXED_OXIDIZED_CUT_COPPER_STAIRS	**north**/south/east/west
WAXED_WEATHERED_CUT_COPPER_STAIRS	**north**/south/east/west
WEATHERED_CUT_COPPER_STAIRS	**north**/south/east/west
WHITE_BED	**north**/south/east/west
WHITE_GLAZED_TERRACOTTA	**north**/south/east/west
WHITE_WALL_BANNER	**north**/south/east/west
WITHER_SKELETON_WALL_SKULL	**north**/south/east/west
YELLOW_BED	**north**/south/east/west
YELLOW_GLAZED_TERRACOTTA	**north**/south/east/west
YELLOW_WALL_BANNER	**north**/south/east/west
ZOMBIE_WALL_HEAD	**north**/south/east/west
AMETHYST_CLUSTER	**up**/down/north/south/east/west
BLACK_SHULKER_BOX	**up**/down/north/south/east/west

Material	Options (default on bold)
BLUE_SHULKER_BOX	**up**/down/north/south/east/west
BROWN_SHULKER_BOX	**up**/down/north/south/east/west
CYAN_SHULKER_BOX	**up**/down/north/south/east/west
END_ROD	**up**/down/north/south/east/west
GRAY_SHULKER_BOX	**up**/down/north/south/east/west
GREEN_SHULKER_BOX	**up**/down/north/south/east/west
LARGE_AMETHYST_BUD	**up**/down/north/south/east/west
LIGHTNING_ROD	**up**/down/north/south/east/west
LIGHT_BLUE_SHULKER_BOX	**up**/down/north/south/east/west
LIGHT_GRAY_SHULKER_BOX	**up**/down/north/south/east/west
LIME_SHULKER_BOX	**up**/down/north/south/east/west
MAGENTA_SHULKER_BOX	**up**/down/north/south/east/west
MEDIUM_AMETHYST_BUD	**up**/down/north/south/east/west
ORANGE_SHULKER_BOX	**up**/down/north/south/east/west
PINK_SHULKER_BOX	**up**/down/north/south/east/west
PURPLE_SHULKER_BOX	**up**/down/north/south/east/west
RED_SHULKER_BOX	**up**/down/north/south/east/west
SHULKER_BOX	**up**/down/north/south/east/west
SMALL_AMETHYST_BUD	**up**/down/north/south/east/west
WHITE_SHULKER_BOX	**up**/down/north/south/east/west
YELLOW_SHULKER_BOX	**up**/down/north/south/east/west

Table A.19: Directional materials

A.2.18 Half

Denotes which half of a two block tall material this block is.

Material	Options (default on bold)
ACACIA_STAIRS	**bottom**/top
ACACIA_TRAPDOOR	**bottom**/top

Material	Options (default on bold)
ANDESITE_STAIRS	**bottom**/top
BIRCH_STAIRS	**bottom**/top
BIRCH_TRAPDOOR	**bottom**/top
BLACKSTONE_STAIRS	**bottom**/top
BRICK_STAIRS	**bottom**/top
COBBLED_DEEPSLATE_STAIRS	**bottom**/top
COBBLESTONE_STAIRS	**bottom**/top
CRIMSON_STAIRS	**bottom**/top
CRIMSON_TRAPDOOR	**bottom**/top
CUT_COPPER_STAIRS	**bottom**/top
DARK_OAK_STAIRS	**bottom**/top
DARK_OAK_TRAPDOOR	**bottom**/top
DARK_PRISMARINE_STAIRS	**bottom**/top
DEEPSLATE_BRICK_STAIRS	**bottom**/top
DEEPSLATE_TILE_STAIRS	**bottom**/top
DIORITE_STAIRS	**bottom**/top
END_STONE_BRICK_STAIRS	**bottom**/top
EXPOSED_CUT_COPPER_STAIRS	**bottom**/top
GRANITE_STAIRS	**bottom**/top
IRON_TRAPDOOR	**bottom**/top
JUNGLE_STAIRS	**bottom**/top
JUNGLE_TRAPDOOR	**bottom**/top
MANGROVE_STAIRS	**bottom**/top
MANGROVE_TRAPDOOR	**bottom**/top
MOSSY_COBBLESTONE_STAIRS	**bottom**/top
MOSSY_STONE_BRICK_STAIRS	**bottom**/top
MUD_BRICK_STAIRS	**bottom**/top
NETHER_BRICK_STAIRS	**bottom**/top
OAK_STAIRS	**bottom**/top
OAK_TRAPDOOR	**bottom**/top

Material	Options (default on bold)
OXIDIZED_CUT_COPPER_STAIRS	**bottom**/top
POLISHED_ANDESITE_STAIRS	**bottom**/top
POLISHED_BLACKSTONE_BRICK_STAIRS	**bottom**/top
POLISHED_BLACKSTONE_STAIRS	**bottom**/top
POLISHED_DEEPSLATE_STAIRS	**bottom**/top
POLISHED_DIORITE_STAIRS	**bottom**/top
POLISHED_GRANITE_STAIRS	**bottom**/top
PRISMARINE_BRICK_STAIRS	**bottom**/top
PRISMARINE_STAIRS	**bottom**/top
PURPUR_STAIRS	**bottom**/top
QUARTZ_STAIRS	**bottom**/top
RED_NETHER_BRICK_STAIRS	**bottom**/top
RED_SANDSTONE_STAIRS	**bottom**/top
SANDSTONE_STAIRS	**bottom**/top
SMOOTH_QUARTZ_STAIRS	**bottom**/top
SMOOTH_RED_SANDSTONE_STAIRS	**bottom**/top
SMOOTH_SANDSTONE_STAIRS	**bottom**/top
SPRUCE_STAIRS	**bottom**/top
SPRUCE_TRAPDOOR	**bottom**/top
STONE_BRICK_STAIRS	**bottom**/top
STONE_STAIRS	**bottom**/top
WARPED_STAIRS	**bottom**/top
WARPED_TRAPDOOR	**bottom**/top
WAXED_CUT_COPPER_STAIRS	**bottom**/top
WAXED_EXPOSED_CUT_COPPER_STAIRS	**bottom**/top
WAXED_OXIDIZED_CUT_COPPER_STAIRS	**bottom**/top
WAXED_WEATHERED_CUT_COPPER_STAIRS	**bottom**/top
WEATHERED_CUT_COPPER_STAIRS	**bottom**/top
ACACIA_DOOR	**lower**/upper
BIRCH_DOOR	**lower**/upper

Material	Options (default on bold)
CRIMSON_DOOR	**lower**/upper
DARK_OAK_DOOR	**lower**/upper
IRON_DOOR	**lower**/upper
JUNGLE_DOOR	**lower**/upper
LARGE_FERN	**lower**/upper
LILAC	**lower**/upper
MANGROVE_DOOR	**lower**/upper
OAK_DOOR	**lower**/upper
PEONY	**lower**/upper
ROSE_BUSH	**lower**/upper
SMALL_DRIPLEAF	**lower**/upper
SPRUCE_DOOR	**lower**/upper
SUNFLOWER	**lower**/upper
TALL_GRASS	**lower**/upper
TALL_SEAGRASS	**lower**/upper
WARPED_DOOR	**lower**/upper

Table A.20: Two-blocks materials

A.2.19 Hanging

Denotes whether the block is hanging.

Defaults to false.

Material	Values
LANTERN	true/false
MANGROVE_PROPAGULE	true/false
SOUL_LANTERN	true/false

Table A.21: Hangable materials

A.2.20 Hinge

Indicates which hinge this door is attached to and will rotate around when opened.

Defaults to left.

Material	Options
ACACIA_DOOR	left/right
BIRCH_DOOR	left/right
CRIMSON_DOOR	left/right
DARK_OAK_DOOR	left/right
IRON_DOOR	left/right
JUNGLE_DOOR	left/right
MANGROVE_DOOR	left/right
OAK_DOOR	left/right
SPRUCE_DOOR	left/right
WARPED_DOOR	left/right

Table A.22: Doors

A.2.21 Honey level

Represents the amount of honey stored in the hive.

Defaults to 0.

Material	Values
BEEHIVE	0/5[11]
BEE_NEST	0/5[11]

Table A.23: Hives

A.2.22 Inverted

Denotes whether this daylight detector is in the inverted mode.

[11] The block is the same from honey-level 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 **??**, Modifier concatenation.

Defaults to false.

Material	Values
DAYLIGHT_DETECTOR	true/false

Table A.24: Invertible blocks

A.2.23 Layers

Represents the amount of layers of snow which are present in this block.

Defaults to 1.

Material	Values
SNOW	1-8

Table A.25: Snow

A.2.24 Leaves

Represents the size of the leaves of the bamboo block.

Defaults to none.

Material	Values
BAMBOO	none/small/large

Table A.26: Bamboo

A.2.25 Level

Represents the amount of fluid contained within this block, either by itself or inside a cauldron.

Defaults on 0 except for POWDER_SNOW_CAULDRON and WATER_CAULDRON (which defaults on 1).

Material	Values
COMPOSTER	0-8
LAVA	0-7
WATER	0-7
POWDER_SNOW_CAULDRON	1-3
WATER_CAULDRON	1-3

Table A.27: Levelled materials

A.2.26 Lit

Denotes whether this block is currently lit.

Except for CAMPFIRE, REDSTONE_TORCH, REDSTONE_WALL_TORCH and SOUL_CAMPFIRE (which defaults to true), it defaults to false.

Material	Options
BLACK_CANDLE	true/false
BLACK_CANDLE_CAKE	true/false
BLAST_FURNACE	true/false
BLUE_CANDLE	true/false
BLUE_CANDLE_CAKE	true/false
BROWN_CANDLE	true/false
BROWN_CANDLE_CAKE	true/false
CANDLE	true/false
CANDLE_CAKE	true/false
CYAN_CANDLE	true/false
CYAN_CANDLE_CAKE	true/false
DEEPSLATE_REDSTONE_ORE	true/false
GRAY_CANDLE	true/false
GRAY_CANDLE_CAKE	true/false
GREEN_CANDLE	true/false
GREEN_CANDLE_CAKE	true/false
LIGHT_BLUE_CANDLE	true/false

Material	Options
LIGHT_BLUE_CANDLE_CAKE	true/false
LIGHT_GRAY_CANDLE	true/false
LIGHT_GRAY_CANDLE_CAKE	true/false
LIME_CANDLE	true/false
LIME_CANDLE_CAKE	true/false
MAGENTA_CANDLE	true/false
MAGENTA_CANDLE_CAKE	true/false
ORANGE_CANDLE	true/false
ORANGE_CANDLE_CAKE	true/false
PINK_CANDLE	true/false
PINK_CANDLE_CAKE	true/false
PURPLE_CANDLE	true/false
PURPLE_CANDLE_CAKE	true/false
REDSTONE_LAMP	true/false
REDSTONE_ORE	true/false
RED_CANDLE	true/false
RED_CANDLE_CAKE	true/false
WHITE_CANDLE	true/false
WHITE_CANDLE_CAKE	true/false
YELLOW_CANDLE	true/false
YELLOW_CANDLE_CAKE	true/false
CAMPFIRE	true/false
REDSTONE_TORCH	true/false
REDSTONE_WALL_TORCH	true/false
SOUL_CAMPFIRE	true/false

Table A.28: Lightable materials

A.2.27 Locked

Denotes whether the repeater is in the locked state or not.

Defaults to false.

Material	Values
REPEATER	true/false

Table A.29: Repeater

A.2.28 Mode

Indicates what mode the block will operate in.

Material	Options (default on bold)
COMPARATOR	**compare**/subtract
STRUCTURE_BLOCK	**load**/corner/save

Table A.30: Materials with mode

A.2.29 Note

Specified tuned pitch that the instrument will be played in.

Defaults to 0.

Material	Values
NOTE_BLOCK	0-24

Table A.31: Note block

A.2.30 Open

Denotes whether this block is currently opened.

Defaults to false.

Material	Values
ACACIA_DOOR	true/false
ACACIA_FENCE_GATE	true/false

Material	Values
ACACIA_TRAPDOOR	true/false
BARREL	true/false
BIRCH_DOOR	true/false
BIRCH_FENCE_GATE	true/false
BIRCH_TRAPDOOR	true/false
CRIMSON_DOOR	true/false
CRIMSON_FENCE_GATE	true/false
CRIMSON_TRAPDOOR	true/false
DARK_OAK_DOOR	true/false
DARK_OAK_FENCE_GATE	true/false
DARK_OAK_TRAPDOOR	true/false
IRON_DOOR	true/false
IRON_TRAPDOOR	true/false
JUNGLE_DOOR	true/false
JUNGLE_FENCE_GATE	true/false
JUNGLE_TRAPDOOR	true/false
MANGROVE_DOOR	true/false
MANGROVE_FENCE_GATE	true/false
MANGROVE_TRAPDOOR	true/false
OAK_DOOR	true/false
OAK_FENCE_GATE	true/false
OAK_TRAPDOOR	true/false
SPRUCE_DOOR	true/false
SPRUCE_FENCE_GATE	true/false
SPRUCE_TRAPDOOR	true/false
WARPED_DOOR	true/false
WARPED_FENCE_GATE	true/false
WARPED_TRAPDOOR	true/false

Table A.32: Openable materials

A.2.31 Orientation

Direction the block is facing.

Defaults to north_up, and can take any of the values shown in Table A.33, Jigsaw orientations.

Orientation
north_up
south_up
east_up
west_up
up_north
up_south
up_east
up_west
down_north
down_south
down_east
down_west

Table A.33: Jigsaw orientations

Material
JIGSAW

Table A.34: Jigsaw

A.2.32 Part

Denotes which half of the bed this block corresponds to.

Defaults to foot.

Material	Values
BLACK_BED	foot/head

Material	Values
BLUE_BED	foot/head
BROWN_BED	foot/head
CYAN_BED	foot/head
GRAY_BED	foot/head
GREEN_BED	foot/head
LIGHT_BLUE_BED	foot/head
LIGHT_GRAY_BED	foot/head
LIME_BED	foot/head
MAGENTA_BED	foot/head
ORANGE_BED	foot/head
PINK_BED	foot/head
PURPLE_BED	foot/head
RED_BED	foot/head
WHITE_BED	foot/head
YELLOW_BED	foot/head

Table A.35: Openable materials

A.2.33 Pickles

Indicates the number of pickles in this block.

Defaults to 1.

Material	Values
SEA_PICKLE	1-4

Table A.36: Materials with pickles

A.2.34 Powered

Indicates whether this block is in the powered state or not (emitting current).

Defaults to false.

Material	Powered value
ACACIA_BUTTON	true/false
ACACIA_PRESSURE_PLATE	true/false
BIRCH_BUTTON	true/false
BIRCH_PRESSURE_PLATE	true/false
COMPARATOR	true/false
CRIMSON_BUTTON	true/false
CRIMSON_PRESSURE_PLATE	true/false
DARK_OAK_BUTTON	true/false
DARK_OAK_PRESSURE_PLATE	true/false
DETECTOR_RAIL	true/false
JUNGLE_BUTTON	true/false
JUNGLE_PRESSURE_PLATE	true/false
LEVER	true/false
LIGHTNING_ROD	true/false
MANGROVE_BUTTON	true/false
MANGROVE_PRESSURE_PLATE	true/false
OAK_BUTTON	true/false
OAK_PRESSURE_PLATE	true/false
OBSERVER	true/false
POLISHED_BLACKSTONE_BUTTON	true/false
POLISHED_BLACKSTONE_PRESSURE_PLATE	true/false
REPEATER	true/false
SPRUCE_BUTTON	true/false
SPRUCE_PRESSURE_PLATE	true/false
STONE_BUTTON	true/false
STONE_PRESSURE_PLATE	true/false
TRIPWIRE_HOOK	true/false
WARPED_BUTTON	true/false
WARPED_PRESSURE_PLATE	true/false

Material	Powered value

Table A.37: Powerabled materials

A.2.35 Rotation

Denotes where the block is looking.

Defaults to 0 and goes up to 15.

Rotation value	Direction
0	South
4	West
8	North
12	East

Table A.38: Relation between rotation and where is looking

Material
ACACIA_SIGN
BIRCH_SIGN
BLACK_BANNER
BLUE_BANNER
BROWN_BANNER
CREEPER_HEAD
CRIMSON_SIGN
CYAN_BANNER
DARK_OAK_SIGN
DRAGON_HEAD
GRAY_BANNER
GREEN_BANNER
JUNGLE_SIGN
LIGHT_BLUE_BANNER
LIGHT_GRAY_BANNER

Material
LIME_BANNER
MAGENTA_BANNER
MANGROVE_SIGN
OAK_SIGN
ORANGE_BANNER
PINK_BANNER
PLAYER_HEAD
PURPLE_BANNER
RED_BANNER
SKELETON_SKULL
SPRUCE_SIGN
WARPED_SIGN
WHITE_BANNER
WITHER_SKELETON_SKULL
YELLOW_BANNER
ZOMBIE_HEAD

Table A.39: Directional materials

A.2.36 Shape - Rails

Represents the current layout of a minecart rail.

Defaults to north_south, and it can take any value of those mentioned in Table A.40, Rail directions.

Shape
north_south
east_west
north_east
north_west
south_east

Shape
south_west
ascending_north
ascending_south
ascending_east
ascending_west

Table A.40: Rail directions

Material
ACTIVATOR_RAIL
DETECTOR_RAIL
POWERED_RAIL
RAIL

Table A.41: Rails

A.2.37 Shape - Stairs

Represents the texture and bounding box shape of these stairs.

Defaults to straight.

Material	Shape
ACACIA_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
ANDESITE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
BIRCH_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
BLACKSTONE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
BRICK_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right

Material	Shape
COBBLED_DEEPSLATE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
COBBLESTONE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
CRIMSON_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
CUT_COPPER_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
DARK_OAK_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
DARK_PRISMARINE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
DEEPSLATE_BRICK_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
DEEPSLATE_TILE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
DIORITE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
END_STONE_BRICK_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
EXPOSED_CUT_COPPER_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
GRANITE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
JUNGLE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
MANGROVE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
MOSSY_COBBLESTONE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right

Material	Shape
MOSSY_STONE_BRICK_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
MUD_BRICK_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
NETHER_BRICK_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
OAK_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
OXIDIZED_CUT_COPPER_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
POLISHED_ANDESITE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
POLISHED_BLACKSTONE_BRICK_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
POLISHED_BLACKSTONE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
POLISHED_DEEPSLATE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
POLISHED_DIORITE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
POLISHED_GRANITE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
PRISMARINE_BRICK_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
PRISMARINE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
PURPUR_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
QUARTZ_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right

Material	Shape
RED_NETHER_BRICK_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
RED_SANDSTONE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
SANDSTONE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
SMOOTH_QUARTZ_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
SMOOTH_RED_SANDSTONE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
SMOOTH_SANDSTONE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
SPRUCE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
STONE_BRICK_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
STONE_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
WARPED_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
WAXED_CUT_COPPER_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
WAXED_EXPOSED_CUT_COPPER_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
WAXED_OXIDIZED_CUT_COPPER_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
WAXED_WEATHERED_CUT_COPPER_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right
WEATHERED_CUT_COPPER_STAIRS	straight/inner_right/inner_left/ outer_right/inner_right

Material	Shape

Table A.42: Stairs

A.2.38 Type

Represents the type of piston which that block corresponds to, or represents what state the slab is in.

Material	Options (default on bold)
ACACIA_SLAB	**bottom**/top/double
ANDESITE_SLAB	**bottom**/top/double
BIRCH_SLAB	**bottom**/top/double
BLACKSTONE_SLAB	**bottom**/top/double
BRICK_SLAB	**bottom**/top/double
COBBLED_DEEPSLATE_SLAB	**bottom**/top/double
COBBLESTONE_SLAB	**bottom**/top/double
CRIMSON_SLAB	**bottom**/top/double
CUT_COPPER_SLAB	**bottom**/top/double
CUT_RED_SANDSTONE_SLAB	**bottom**/top/double
CUT_SANDSTONE_SLAB	**bottom**/top/double
DARK_OAK_SLAB	**bottom**/top/double
DARK_PRISMARINE_SLAB	**bottom**/top/double
DEEPSLATE_BRICK_SLAB	**bottom**/top/double
DEEPSLATE_TILE_SLAB	**bottom**/top/double
DIORITE_SLAB	**bottom**/top/double
END_STONE_BRICK_SLAB	**bottom**/top/double
EXPOSED_CUT_COPPER_SLAB	**bottom**/top/double
GRANITE_SLAB	**bottom**/top/double
JUNGLE_SLAB	**bottom**/top/double
MANGROVE_SLAB	**bottom**/top/double
MOSSY_COBBLESTONE_SLAB	**bottom**/top/double

Material	Options (default on bold)
MOSSY_STONE_BRICK_SLAB	**bottom**/top/double
MUD_BRICK_SLAB	**bottom**/top/double
NETHER_BRICK_SLAB	**bottom**/top/double
OAK_SLAB	**bottom**/top/double
OXIDIZED_CUT_COPPER_SLAB	**bottom**/top/double
PETRIFIED_OAK_SLAB	**bottom**/top/double
POLISHED_ANDESITE_SLAB	**bottom**/top/double
POLISHED_BLACKSTONE_BRICK_SLAB	**bottom**/top/double
POLISHED_BLACKSTONE_SLAB	**bottom**/top/double
POLISHED_DEEPSLATE_SLAB	**bottom**/top/double
POLISHED_DIORITE_SLAB	**bottom**/top/double
POLISHED_GRANITE_SLAB	**bottom**/top/double
PRISMARINE_BRICK_SLAB	**bottom**/top/double
PRISMARINE_SLAB	**bottom**/top/double
PURPUR_SLAB	**bottom**/top/double
QUARTZ_SLAB	**bottom**/top/double
RED_NETHER_BRICK_SLAB	**bottom**/top/double
RED_SANDSTONE_SLAB	**bottom**/top/double
SANDSTONE_SLAB	**bottom**/top/double
SMOOTH_QUARTZ_SLAB	**bottom**/top/double
SMOOTH_RED_SANDSTONE_SLAB	**bottom**/top/double
SMOOTH_SANDSTONE_SLAB	**bottom**/top/double
SMOOTH_STONE_SLAB	**bottom**/top/double
SPRUCE_SLAB	**bottom**/top/double
STONE_BRICK_SLAB	**bottom**/top/double
STONE_SLAB	**bottom**/top/double
WARPED_SLAB	**bottom**/top/double
WAXED_CUT_COPPER_SLAB	**bottom**/top/double
WAXED_EXPOSED_CUT_COPPER_SLAB	**bottom**/top/double
WAXED_OXIDIZED_CUT_COPPER_SLAB	**bottom**/top/double

Material	Options (default on bold)
WAXED_WEATHERED_CUT_COPPER_SLAB	**bottom**/top/double
WEATHERED_CUT_COPPER_SLAB	**bottom**/top/double
PISTON_HEAD	**normal**/sticky

Table A.43: Type materials

A.2.39 Vertical direction

Represents the dripstone orientation.

Defaults to up.

Material	Values
POINTED_DRIPSTONE	up/down

Table A.44: Dripstone

A.2.40 Waterlogged

Denotes whether this block has fluid in it.

Besides underwater blocks[12] (which defaults to true), it defaults to false. All the possible options are true or false.

Material	Aquatic block[13]
ACACIA_FENCE	✗
ACACIA_LEAVES	✗

[12]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, 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[13]
ACACIA_SIGN	✗
ACACIA_SLAB	✗
ACACIA_STAIRS	✗
ACACIA_TRAPDOOR	✗
ACACIA_WALL_SIGN	✗
ACTIVATOR_RAIL	✗
AMETHYST_CLUSTER	✗
ANDESITE_SLAB	✗
ANDESITE_STAIRS	✗
ANDESITE_WALL	✗
AZALEA_LEAVES	✗
BIG_DRIPLEAF	✗
BIG_DRIPLEAF_STEM	✗
BIRCH_FENCE	✗
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	✗

Material	Aquatic block[13]
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	✗
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	✗

Material	Aquatic block[13]
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	✗
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	✗

Material	Aquatic block[13]
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	✗
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	✗

Material	Aquatic block[13]
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	✗
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	✗

Material	Aquatic block[13]
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	✗
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	✗

Material	Aquatic block[13]
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	✗
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	✗

Material	Aquatic block[13]
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	✗
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	✓

Material	Aquatic block[13]
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	✓
TUBE_CORAL_WALL_FAN	✓

Table A.45: Waterlogged materials

A.3 Material modifiers aggregation

some modifiers are similar, so let's add them together

A.3.1 Direction aggregation

modifier, followed by the condition to set the bit to 1

Property	U	D	N	S	E	W
up	true					
down		true				
north			true			
south				true		
east					true	
west						true
face	ceiling	floor				
attachment	ceiling	floor				
half	top or upper	bottom or lower				
facing	up	down	north	south	east	west
vertical-direction	up	down				
type	top or double	bottom or double				

[13] If it's an underwater block (defaults to true).

Property	U	D	N	S	E	W
orientation	up_X	down_X	X_north or north_up	X_south or south_up	X_east or east_up	X_west or west_up
shape	ascending_X		ascending_north or north_X	ascending_south or south_X or north_south	ascending_east or X_east or east_west	ascending_west or X_west
hanging	true					

Table A.46: Direction aggregation

A.3.2 Axis aggregation

modifier up to 2b

Property	MSB	LSB
axis	Y or Z	X or Z
attachment	double_wall	single_wall

Table A.47: Axis aggregation

A.3.3 Age aggregation

modifier up to 8b

Property
age
berries
honey_level

Table A.48: Age aggregation

A.3.4 Grouped materials aggregation

modifier up to 3b

Property
candles
eggs
pickles

Table A.49: Grouped materials aggregation

A.3.5 Stages aggregation

modifier up to 8b

Property
charges
level
bites
layers

Table A.50: Stages aggregation

A.3.6 Parts aggregation

modifier; up to 4b

Property
part
piston_head
stair_shape

Table A.51: Parts aggregation

References

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

Rajaraman, V. (2016, January). Ieee standard for floating point numbers. *Resonance*. Retrieved from https://www.ias.ac.in/article/fulltext/reso/021/01/0011-0030

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

Spigot - enum material - 1.19. (n.d.). Retrieved 2022-08-05, from https://hub.spigotmc.org/javadocs/spigot/org/bukkit/material/package-summary.html

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

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