



class MongoDB::Server

Mongod server object

Table of Contents

- 1 [Synopsis](#)
- 2 [Readonly attributes](#)
 - 2.1 [server-name](#)
 - 2.2 [server-port](#)
 - 2.3 [max-sockets](#)
- 3 [Methods](#)
 - 3.1 [new](#)
 - 3.2 [server-init](#)
 - 3.3 [get-status](#)
 - 3.4 [tap-monitor](#)
 - 3.5 [get-socket](#)
 - 3.6 [raw-query](#)
 - 3.7 [name](#)

```
unit package MongoDB;  
class Server { ... }
```

Synopsis

```
my MongoDB::Client $client .= new(:uri<mongodb://>);  
my MongoDB::Server $server = $client.select-server;
```

This class is used to provide sockets for communication with a mongo database server. The class is loaded and controlled by the [MongoDB::Client](#) module. *Users should not use this class directly but ask for a server using methods from a Client object. And even that users should do it carefully. All server communication is controlled internally.*

Readonly attributes

server-name

```
has Str $.server-name;
```

Server name or ip of this object.

server-port

```
has Int $.server-port;
```

Server port of this object.

max-sockets

```
has Int $.max-sockets;
```

Maximum number of open sockets per server object Can be changed by user of Server object.
New values must ≥ 3 .

Methods

new

Defined as:

```
submethod BUILD (  
  Str:D :$host!,  
  Int:D :$port! where (0 <= $_ <= 65535),  
  Int :$max-sockets where $_ >= 3 = 3,  
  Hash :$uri-data,  
  MongoDB::Database:D :$db-admin,  
  MongoDB::Client:D :$client  
)
```

Creates a new [MongoDB::Server](#) object. The object tries to establish a connection to the server. The `IO::Socket::INET` class is used for that and will throw an exception when it fails.

[MongoDB::Client](#) creates the Server object normally and catches the exception on failure. *Do not use this method directly!*

server-init

```
method server-init ( )
```

Do not use it. Get a Server object from the Client object.

get-status

Get current status about this server. [server-status\(\)](#) in the Client object will return the same

information.

tap-monitor

```
method tap-monitor ( |c --> Tap )
```

Do not use it! This is for internal use only.

get-socket

```
method get-socket ( --> MongoDB::Socket )
```

Get a MongoDB::Socket object. By default there can be only 3 open sockets per server object. See also set-max-sockets() below. When operations are done the socket must be closed to prevent problems.

raw-query

```
method raw-query (
  Str:D $full-collection-name, BSON::Document:D $query,
  Int :$number-to-skip = 0, Int :$number-to-return = 1,
  Bool :$authenticate = True
--> BSON::Document
)
```

Direct a query to this server. Not often used but needed when to get specific information or to modify a server e.g. to transform a standalone server into a replica server. The normal queries using run-command().will sit and wait for a master or primary server becomes available.

name

```
method name ( --> Str )
```

Returns a string in the format <servername> ':' <port number>.