



class BSON::Binary

Container for binary data

Table of Contents

1 [Methods](#)

1.1 [new](#)

1.2 [perl](#)

1.3 [encode](#)

1.4 [decode](#)

```
package BSON { class Binary { ... } }
```

Use as

```
use UUID;
use BSON::Binary;

my UUID $uuid .= new(:version(4));
my BSON::Binary $bin .= new(
    :data($uuid.Blob),
    :type(BSON::C-UUID)
);
```

Methods

new

```
submethod BUILD ( Buf :$data, Int :$type = BSON::C-GENERIC )
```

Store binary data in the object. The following codes are available:

- **C-GENERIC**. Generic binary data. Used for any non structured data.
- **C-FUNCTION**. Encoding of function code.
- **C-UUID**. Encoding of UUID.
- **C-MD5**. Encoding of MD5.

perl

```
method perl ( Int $indent = 0 --> Str )
```

Should be used like any other perl without the \$indent set. Otherwise it starts of indented.

encode

```
method encode ( --> Buf )
```

Encode the binary data. Mostly called from BSON::Document.

decode

```
method decode (  
    Buf:D $b,  
    Int:D $index is copy,  
    Int:D $nbr-bytes  
    --> BSON::Binary  
)
```

Decode a Buf to BSON::Binary. Mostly called from BSON::Document.

Generated using Pod::Render, Pod::To::HTML, wkhtmltopdf