

**NAME**

Net::DNS::Question – DNS question record

**SYNOPSIS**

```
use Net::DNS::Question;

$question = new Net::DNS::Question('example.com', 'AAAA', 'IN');
```

**DESCRIPTION**

A Net::DNS::Question object represents a record in the question section of a DNS packet.

**METHODS****new**

```
$question = new Net::DNS::Question('example.com', 'AAAA', 'IN');
$question = new Net::DNS::Question('example.com', 'A', 'IN');
$question = new Net::DNS::Question('example.com');

$question = new Net::DNS::Question('2001::DB8::dead:beef', 'PTR', 'IN');
$question = new Net::DNS::Question('2001::DB8::dead:beef');
```

Creates a question object from the domain, type, and class passed as arguments. One or both type and class arguments may be omitted and will assume the default values shown above.

RFC4291 and RFC4632 IP address/prefix notation is supported for queries in both in-addr.arpa and ip6.arpa namespaces.

**decode**

```
$question = decode Net::DNS::Question(\$data, $offset);

($question, $offset) = decode Net::DNS::Question(\$data, $offset);
```

Decodes the question record at the specified location within a DNS wire-format packet. The first argument is a reference to the buffer containing the packet data. The second argument is the offset of the start of the question record.

Returns a Net::DNS::Question object and the offset of the next location in the packet.

An exception is raised if the object cannot be created (e.g., corrupt or insufficient data).

**encode**

```
$data = $question->encode( $offset, $hash );
```

Returns the Net::DNS::Question in binary format suitable for inclusion in a DNS packet buffer.

The optional arguments are the offset within the packet data where the Net::DNS::Question is to be stored and a reference to a hash table used to index compressed names within the packet.

**print**

```
$object->print;
```

Prints the record to the standard output. Calls the **string()** method to get the string representation.

**string**

```
print "string = ", $question->string, "\n";
```

Returns a string representation of the question record.

**name**

```
$name = $question->name;
```

Internationalised domain name corresponding to the qname attribute.

Decoding non-ASCII domain names is computationally expensive and undesirable for names which are likely to be used to construct further queries.

When required to communicate with humans, the 'proper' domain name should be extracted from a query

or reply packet.

```
$query = new Net::DNS::Packet( $example, 'ANY' );
$reply = $resolver->send($query) or die;
($question) = $reply->question;
$name = $question->name;
```

#### **qname, zname**

```
$qname = $question->qname;
$zname = $question->zname;
```

Fully qualified domain name in the form required for a query transmitted to a nameserver. In dynamic update packets, this attribute is known as **zname()** and refers to the zone name.

#### **qtype, ztype, type**

```
$qtype = $question->type;
$qtype = $question->qtype;
$ztype = $question->ztype;
```

Returns the question type attribute. In dynamic update packets, this attribute is known as **ztype()** and refers to the zone type.

#### **qclass, zclass, class**

```
$qclass = $question->class;
$qclass = $question->qclass;
$zclass = $question->zclass;
```

Returns the question class attribute. In dynamic update packets, this attribute is known as **zclass()** and refers to the zone class.

## **COPYRIGHT**

Copyright (c)1997–2000 Michael Fuhr.

Portions Copyright (c)2002,2003 Chris Reinhardt.

Portions Copyright (c)2003,2006–2011 Dick Franks.

All rights reserved.

## **LICENSE**

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of the author not be used in advertising or publicity pertaining to distribution of the software without specific prior written permission.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## **SEE ALSO**

perl, Net::DNS, Net::DNS::DomainName, Net::DNS::Packet, RFC 1035 Section 4.1.2