NAME

Net::DNS::Resolver::Recurse − DNS recursive resolver

SYNOPSIS

```
use Net::DNS::Resolver::Recurse;

my $resolver = new Net::DNS::Resolver::Recurse();
$resolver->debug(1);

$resolver->hints('198.41.0.4');      # A.ROOT-SERVER.NET.

my $packet = $resolver->send( 'www.rob.com.au.', 'A' );
```

DESCRIPTION

This module is a subclass of Net::DNS::Resolver.

METHODS

This module inherits almost all the methods from Net::DNS::Resolver. Additional module-specific methods are described below.

hints

This method specifies a list of the IP addresses of nameservers to be used to discover the addresses of the root nameservers.

```
$resolver->hints(@ip);
```

If no hints are passed, the priming query is directed to nameservers drawn from a built-in list of IP addresses.

query, search, send

The **query**(), **search**() and **send**() methods produce the same result as their counterparts in Net::DNS::Resolver.

```
$packet = $resolver->send( 'www.example.com.', 'A' );
```

Server-side recursion is suppressed by clearing the recurse flag in query packets and recursive name resolution is performed explicitly.

The **query()** and **search()** methods are inherited from Net::DNS::Resolver and invoke **send()** indirectly.

callback

This method specifies a code reference to a subroutine, which is then invoked at each stage of the recursive lookup.

For example to emulate dig's +trace function:

```
my $coderef = sub {
    my $packet = shift;

    printf ";; Received %d bytes from %s\n\n",
            $packet->answersize, $packet->answerfrom;
};

$resolver->callback($coderef);
```

The callback subroutine is not called for queries for missing glue records.

ACKNOWLEDGEMENT

This package is an improved and compatible reimplementation of the Net::DNS::Resolver::Recurse.pm created by Rob Brown in 2002, whose contribution is gratefully acknowledged.

COPYRIGHT

Copyright (c)2014,2019 Dick Franks.

Portions Copyright (c)2002 Rob Brown.

LICENSE

SEE ALSO

Net::DNS::Resolver