# NSQUERY User's Manual

**January, 1989**

This manual contains information about the use of NSQUERY, a program for testing Internet domain service (RFC 1035) name servers.

**Revision/Update Information:** This is a new manual.

**Engineering Computing Services**
**Rensselaer Polytechnic Institute**
**Troy, New York**

**09 January 1989**

Copyright ©1989 Rensselaer Polytechnic Institute

All Rights Reserved.
Printed in U.S.A.

This document was prepared using VAX DOCUMENT, Version 1.1

# Preface

This document is intended for users of the NSQUERY program developed at Rensselaer Polytechnic Institute for VAX/VMS systems running CMU/Tektronix TCP/IP. The program currently runs under VAX/VMS V4.7 and V5.0-2 on CMU TCP/IP V6.3.

The reader is expected to be familiar with, or at least have access to, the information in Internet RFC's 1034 and 1035.

# 1 Using NSQUERY

NSQUERY is designed to be invoked as a DCL ‘‘foreign command’’:

```
$ NSQ*UERY :== $disk:[dir]NSQUERY
```

The command description on the following pages describes the qualifiers and parameters available with NSQUERY.

# NSQUERY—Issue a Name Service Query

This command issues a query to an Internet domain server and displays the resultant reply.

## FORMAT

**NSQUERY** *query-name [server]*

| Command Qualifiers | Defaults |
|---|---|
| */PORT=number* | */PORT=53* |
| */PROTOCOL=proto* | */PROTOCOL=UDP* |
| */TYPE=qtype* | */TYPE=ALL* |
| */CLASS=qclass* | */CLASS=INTERNET* |

## PARAMETERS

### *query-name*

Name about which information should be obtained. This should be an internet dotted domain-style name.

### *server*

Server to be queried. If omitted, defaults to ''C.NYSER.NET''. If the specified host name has multiple addresses, all addresses are obtained and each is tried until a query is successfully sent and reply received.

## DESCRIPTION

All resource records defined in RFC 1035 are interpreted by NSQUERY. Refer to that RFC for information about the abbreviations used in the resource record display. All ''time-to-live'' values are displayed as VMS delta-time strings. NSQUERY never requests recursion nor does it note the availability of recursion from a server. Authoritative responses and truncated messages are noted before the resource records are displayed.

## QUALIFIERS

### */PORT=port*

Directs the query to the specified TCP or UDP port. If omitted, port 53 (the port used by most, if not all, domain servers) is used.

### */PROTOCOL=proto*

Selects either **TCP** or **UDP** protocol. The default is UDP. UDP is generally used for simple domain queries; TCP is generally only used for major domain system events, such as zone transfers.

### */TYPE=qtype*

Selects the type of resource record to be returned about the name. The default is ALL, which selects all available resource record types. Available types are as follows:

**Table 1–1 Resource record types**

| Type/QType | Description |
|---|---|
| A | host address |
| NS | authoritative name server |
| MD | mail destination (obsolete) |
| MF | mail forwarder (obsolete) |
| CNAME | canonical name for an alias |
| SOA | start of zone authority |
| MB | mailbox domain name (experimental) |
| MG | mail group member (experimental) |
| MR | mail rename domain name (experimental) |
| NULL | null resource record |
| WKS | well-known service description |
| PTR | domain name pointer |
| HINFO | host information |
| MINFO | mailbox or mail list information |
| MX | mail exchange |
| TXT | text strings |
| AXFR | complete zone transfer |
| MAILB | all mailbox-related records (MB, MG, MR) |
| MAILA | all mail agent records (obsolete) |

Refer to RFC 1035 for a complete description of resource record types and their uses. Note that not all servers support all record types. You should probably never attempt to use the AXFR query type.

## /CLASS=qclass

Selects the class of resource records to be returned. The default is INTERNET. Available **qclass** values are:

**Table 1–2 Resource record classes**

| Class/QClass | Description |
|---|---|
| INTERNET | the Internet |
| CSNET | the CSNET class (obsolete) |
| CHAOSNET | the CHAOS class |
| HESIOD | the Hesiod class |
| ALL | all classes |

For more information about resource record class values, see RFC 1035. Note that not all servers support all classes; most servers' authority extends only over Internet-class information.

### Example 1–0

```
$ NSQUERY RPI.EDU FS3.CS.RPI.EDU

%NSQUERY-I-TRYNS, trying address 128.213.1.14...
%NSQUERY-I-AUTHRESP, authoritative response
%NSQUERY-I-QUERYOK, query operation completed successfully

QUERY  #1: QName=RPI.EDU., QType=255, QClass=1

ANSWER #1: RPI.EDU., SOA, INTERNET,    0 01:00:00
    Primary NS: fs3.cs.rpi.edu.rpi.edu.
    Responsible: weltyc.fs3.cs.rpi.edu.
    Serial=600010, Refresh=   1 06:00:00, Retry=   0 00:05:00
    Expire=  41 16:00:00, Minimum=   0 01:00:00
ANSWER #2: RPI.EDU., NS, INTERNET,    0 01:00:00, NS=fs3.cs.rpi.edu.
ANSWER #3: RPI.EDU., NS, INTERNET,    0 01:00:00, NS=columbia.edu.
ANSWER #4: RPI.EDU., NS, INTERNET,    0 01:00:00, NS=nisc.nyser.net.

ADDTNL #1: fs3.cs.rpi.edu., A, INTERNET,    0 01:00:00, Addr=128.213.1.14
ADDTNL #2: columbia.edu., A, INTERNET,    1 21:28:12, Addr=128.59.16.1
ADDTNL #3: columbia.edu., A, INTERNET,    1 21:28:12, Addr=128.59.32.1
ADDTNL #4: columbia.edu., A, INTERNET,    1 21:28:12, Addr=128.121.13.2
ADDTNL #5: nisc.nyser.net., A, INTERNET,    0 00:02:00, Addr=192.33.4.10
```