

Network Working Group  
Request for Comments: 1712  
Category: Experimental

C. Farrell  
M. Schulze  
S. Pleitner  
D. Baldoni  
Curtin University of Technology  
November 1994

## DNS Encoding of Geographical Location

### Status of this Memo

This memo defines an Experimental Protocol for the Internet community. This memo does not specify an Internet standard of any kind. Discussion and suggestions for improvement are requested. Distribution of this memo is unlimited.

### Abstract

This document defines the format of a new Resource Record (RR) for the Domain Naming System (DNS), and reserves a corresponding DNS type mnemonic and numerical code. This definition deals with associating geographical host location mappings to host names within a domain. The data shown in this document is fictitious and does not necessarily reflect the real Internet.

### 1. Introduction

It has been a long standing problem to relate IP numbers to geographical locations. The availability of Geographical location information has immediate applications in network management. Such information can be used to supplement the data already provided by utilities such as whois [Har85], traceroute [VJ89], and nslookup [UCB89]. The usefulness and functionality of these already widely used tools would be greatly enhanced by the provision of reliable geographical location information.

The ideal way to manage and maintain a database of information, such as geographical location of internet hosts, is to delegate responsibility to local domain administrators. A large distributed database could be implemented with a simple mechanism for updating the local information. A query mechanism also has to be available for checking local entries, as well as inquiring about data from non-local domains.

## 2. Background

The Internet continues to grow at an ever increasing rate with IP numbers allocated on a first-come-first-serve basis. Deciding when and how to setup a database of geographical information about internet hosts presented a number of options. The uumap project [UU85] was the first serious attempt to collect geographical location data from sites and store it centrally. This project met with limited success because of the difficulty in maintaining and updating a large central database. Another problem was the lack of tools for the checking the data supplied, this problem resulted in some erroneous data entering the database.

### 2.1 SNMP:

Using an SNMP get request on the sysLocation MIB (Management Information Base) variable was also an option, however this would require the host to be running an appropriate agent with public read access. It was also felt that MIB data should reflect local management data (e.g., "this" host is on level 5 room 74) rather than a hosts geographical position. This view is supported in the examples given in literature in this area [ROSE91].

### 2.2 X500:

The X.500 Directory service [X.500.88] defined as part of the ISO standards also appears as a potential provider of geographical location data. However due to the limited implementations of this service it was decided to defer this until this service gains wider use and acceptance within the Internet community.

### 2.3 BIND:

The DNS [Mock87a][Mock87b] represents an existing system ideally suited to the provision of host specific information. The DNS is a widely used and well-understood mechanism for providing a distributed database of such information and its extensible nature allows it to be used to disseminate virtually any information. The most commonly used DNS implementation is the Berkeley Internet Name Domain server BIND [UCB89]. The information we wished to make available needed to be updated locally but available globally; a perfect match with the services provided by the DNS. Current DNS servers provide a variety of useful information about hosts in their domain but lack the ability to report a host's geographical location.

### 3. RDATA Format

```

      MSB                                                    LSB
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
/                                LONGITUDE                                /
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
/                                LATITUDE                                /
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
/                                ALTITUDE                                /
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

where:

**LONGITUDE** The real number describing the longitude encoded as a printable string. The precision is limited by 256 characters within the range -90..90 degrees. Positive numbers indicate locations north of the equator.

**LATITUDE** The real number describing the latitude encoded as a printable string. The precision is limited by 256 characters within the range -180..180 degrees. Positive numbers indicate locations east of the prime meridian.

**ALTITUDE** The real number describing the altitude (in meters) from mean sea-level encoded as a printable string. The precision is limited by 256 characters. Positive numbers indicate locations above mean sea-level.

Latitude/Longitude/Altitude values are encoded as strings as to avoid the precision limitations imposed by encoding as unsigned integers. Although this might not be considered optimal, it allows for a very high degree of precision with an acceptable average encoded record length.

### 4. The GPOS RR

The geographical location is defined with the mnemonic GPOS and type code 27.

GPOS has the following format:

```
<owner> <ttl> <class> GPOS <longitude> <latitude> <altitude>
```

A floating point format was chosen to specify geographical locations for reasons of simplicity. This also guarantees a concise unambiguous description of a location by enforcing three compulsory numerical values to be specified.

The owner, ttl, and class fields are optional and default to the last defined value if omitted. The longitude is a floating point number ranging from -90 to 90 with positive values indicating locations north of the equator. For example Perth, Western Australia is located at 32° 7' 19" south of the equator which would be specified as -32.68820. The latitude is a number ranging from -180.0 to 180.0. For example Perth, Western Australia is located at 116° 2' 25" east of the prime meridian which would be specified as 116.86520. Curtin University, Perth is also 10 meters above sea-level.

The valid GPOS record for a host at Curtin University in Perth Western Australia would therefore be:

```
GPOS -32.6882 116.8652 10.0
```

There is no limit imposed on the number of decimal places, although the length of the encoded string is limited to 256 characters for each field. It is also suggested that administrators limit their entries to the minimum number of necessary characters in each field.

## 5. Master File Format

Each host requires its own GPOS field in the corresponding DNS RR to explicitly specify its geographical location and altitude. If the GPOS field is omitted, a DNS enquiry will return no position information for that host.

Consider the following example:

```
; Authoritative data for cs.curtin.edu.au.
;
@      IN      SOA      marsh.cs.curtin.edu.au. postmaster.cs.curtin.edu.au.
      (
          94070503      ; Serial (yymddnn)
          10800          ; Refresh (3 hours)
          3600           ; Retry (1 hour)
          3600000        ; Expire (1000 hours)
          86400          ; Minimum (24 hours)
      )

      IN      NS      marsh.cs.curtin.edu.au.

marsh      IN      A      134.7.1.1
           IN      MX      0      marsh
           IN      HINFO   SGI-Indigo IRIX-4.0.5F
           IN      GPOS    -32.6882 116.8652 10.0
ftp        IN      CNAME   marsh
```

lillee	IN	A	134.7.1.2
	IN	MX	0 marsh
	IN	HINFO	SGI-Indigo IRIX-4.0.5F
	IN	GPOS	-32.6882 116.8652 10.0
hinault	IN	A	134.7.1.23
	IN	MX	0 marsh
	IN	HINFO	SUN-IPC SunOS-4.1.3
	IN	GPOS	-22.6882 116.8652 250.0
merckx	IN	A	134.7.1.24
	IN	MX	0 marsh
	IN	HINFO	SUN-IPC SunOS-4.1.1
ambrose	IN	A	134.7.1.99
	IN	MX	0 marsh
	IN	HINFO	SGI-CHALLENGE_L IRIX-5.2
	IN	GPOS	-32.6882 116.8652 10.0

The hosts marsh, lillee, and ambrose are all at the same geographical location, Perth Western Australia (-32.68820 116.86520). The host hinault is at a different geographical location, 10 degrees north of Perth in the mountains (-22.6882 116.8652 250.0). For security reasons we do not wish to give the location of the host merckx.

Although the GPOS clause is not a standard entry within BIND configuration files, most vendor implementations seem to ignore whatever is not understood upon startup of the DNS. Usually this will result in a number of warnings appearing in system log files, but in no way alters naming information or impedes the DNS from performing its normal duties.

## 7. References

- [ROSE91] Rose M., "The Simple Book: An Introduction to Management of TCP/IP-based Internets", Prentice-Hall, Englewood Cliffs, New Jersey, 1991.
- [X.500.88] CCITT: The Directory - Overview of Concepts, Models and Services", Recommendations X.500 - X.521.
- [Har82] Harrenstein K, Stahl M., and E. Feinler, "NICNAME/WHOIS" RFC 812, SRI NIC, March 1982.
- [Mock87a] Mockapetris P., "Domain Names - Concepts and Facilities", STD 13, RFC 1034, USC/Information Sciences Institute, November 1987.
- [Mock87b] Mockapetris P., "Domain Names - Implementation and Specification", STD 13, RFC 1035, USC/Information Sciences Institute, November 1987.
- [FRB93] Ford P., Rekhter Y., and H-W. Braun, "Improving the Routing and Addressing of IP", IEEE Network Vol.7, No. 3, pp. 11-15, May 1993.
- [VJ89] Jacobsen V., "The Traceroute(8) Manual Page", Lawrence Berkeley Laboratory, Berkeley, CA, February 1989.
- [UCB89] University of California, "BIND: Berkeley Internet Name Domain Server", 1989.
- [UU85] UUCP Mapping Project, Software available via anonymous FTP from ftp.uu.net., 1985.

## 8. Security Considerations

Once information has been entered into the DNS, it is considered public.

## 9. Authors' Addresses

Craig Farrell  
Department of Computer Science  
Curtin University of technology  
GPO Box U1987 Perth,  
Western Australia

EMail: craig@cs.curtin.edu.au

Mike Schulze  
Department of Computer Science  
Curtin University of technology  
GPO Box U1987 Perth,  
Western Australia

EMail: mike@cs.curtin.edu.au

Scott Pleitner  
Department of Computer Science  
Curtin University of technology  
GPO Box U1987 Perth,  
Western Australia

EMail: pleitner@cs.curtin.edu.au

Daniel Baldoni  
Department of Computer Science  
Curtin University of technology  
GPO Box U1987 Perth,  
Western Australia

EMail: flint@cs.curtin.edu.au

