Network Working Group
INTERNET-DRAFT

S.E. Hardcastle-Kille
University College London
September 1991

DSA Naming

Status of this Memo

This INTERNET–DRAFT describes a few problems with DSA Naming as currently deployed in pilot exercises, and suggests a new approach. This approach is suggested for use in the Internet Directory Pilot.

I believe this note to be an important step forward, as it begins to evolve a clear model of a Directory Management Domain.

This draft document will be submitted to the RFC editor as a protocol specification. Distribution of this memo is unlimited. Please send comments to the author or to the discussion group <osi-ds@CS.UCL.AC.UK>.

1 Problem Statement

There are a number of difficulties with the approach being used in QUIPU based pilots, which is described in the original QUIPU Design and in the QUIPU documentation. The problems are:

1. It is hard to achieve very high replication of knowledge information, as this is very widely spread. This is particularly true of secondary DSAs, which are named inside an organisation. Due to the sensitivity of sibling information, replication of this data outside the organisation is less likely.

2. There is a need to give DSAs more reasonable names. The current names are too high up the tree to indicate the role of the DSA. This can lead to management confusion.

3. There is too much DIT clutter — more endangered South American animals than organisations!

4. There is no real concept of DMD (Directory Management Domain) authority. This needs to be introduced, in order to allow DSAs to be named by the authority that runs them.

The following are good properties of the current approach, which must not be lost in any changes.

1. The directory should be used to look up the presentation addresses of DSAs as described in [Kil91] and not require them to be manually associated with all knowledge references as implied in X.500.

2. The approach must be deadlock free. This will not be the case if DSAs are named in a careless manner, and the directory is used to look up the presentation addresses of DSAs.

3. There should be no manual replication of information, which is implied by the presentation address configuration approach of the standard.

2 What is a DMD

A DMD is a special type of Organisation, which has two roles.

1. Managing directory configuration information.

2. Operating one or more DSAs (A DSA belongs to the DMD that operates it)

Because a DMD is an Organisation, it should be represented in the DIT as if it was any other Organisation. A common case is for an Organisation (or Organisational Unit) to operate its own DSAs. In this case the Organisation has a dual role as both a DMD and an Organisation which is not a DMD. Such a DMD is primarily responsible for operating a number of DSAs, and some minimal configuration information associated with those DSAs. For most organisations, the number of DSAs will be small, although a large organisation may run quite a large number.

Another common case is for a DMD to be primarily concerned with managing configuration information between a large number of DSAs, and perhaps operating a small number of DSAs in order to achieve this. A research network pilot is an example of such a DMD. Such a DMD will:

- Collect and distribute configuration information
- Collect and distribute user data (replication)

An Administrative DMD (as distinct from a Private DMD) is one which systematically maintains knowledge of the root configuration within the DMD. This will only be a useful distinction if this knowledge is restricted (it is not in the current pilot environment). For now it should be assumed that this information is widely replicated, as adequate performance of the global directory depends on it.

A DMD may have a peer relationship with other DMDs. There may also be a hierarchical relationship. This is appropriate for a DMD which wishes to obtain all of its configuration information and replicated from a parent DMD. A hierarchical relationship does not preclude other relationships between DMDs (e.g., to mutually replicate data).

A DMD has the following responsibilities:

- It must operate at least one DSA, to manage the information associated with the DMD.
- It may operate DSAs to store information for Organisations supported by the DMD. These DSAs are named within the DMD
- Allocate sub-DMDs. The EDB for such a DMD should be mastered in the DMD DSA
- Manage the replication of knowledge information within the DMD, and make it available for external replication.

3 Naming DMDs and DSAs

The problem of giving DSAs reasonable names, and remaining deadlock-free is awkward. To resolve this, the concept of a DSA Naming Tree is

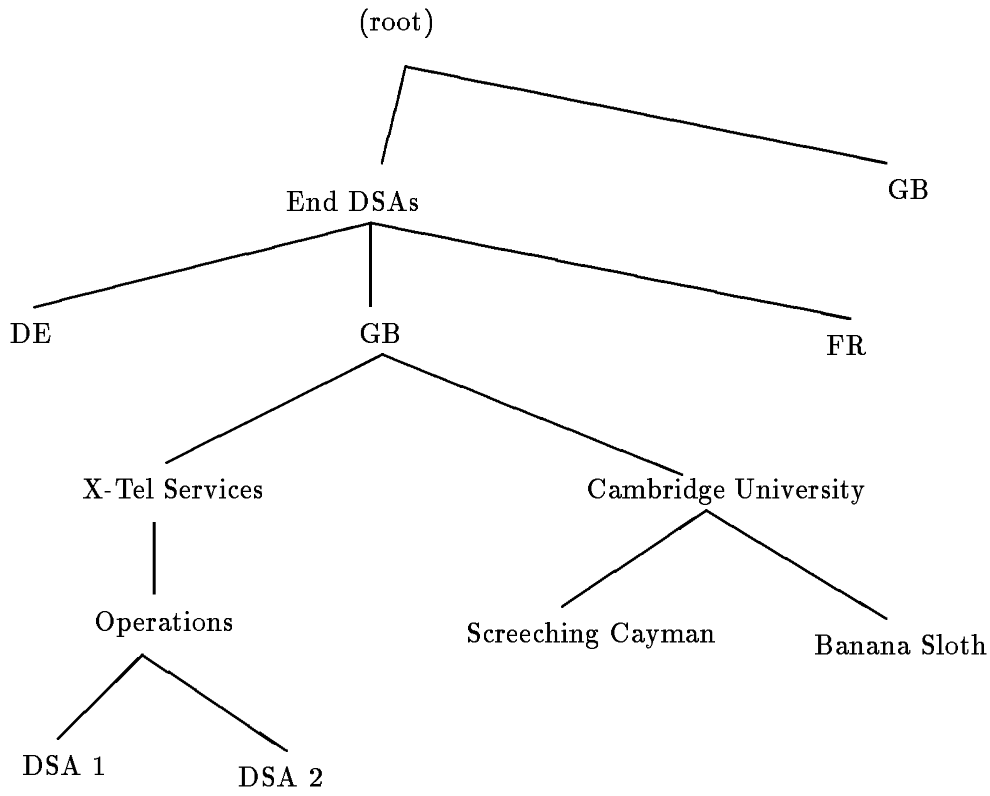

Figure 1: Example DSA Naming Tree

introduced. An example of this is given in Figure 1. DSA Naming trees have the following rules:

1. There may be multiple DSA Naming Trees
2. All DSAs are named within a DSA Naming Tree. This prevents deadlock for all of the DIT not in a DSA Naming Tree.
3. DSA Naming Trees are oderered by Rank, and each DSA named in a given DSA Naming Tree is at the same Rank as the DSA Naming Tree.
4. Each DSA Naming Tree has a string RDN. The two DSA Naming Trees defined initially have names "End DSAs" and "DMD DSAs", the latter being of higher rank.
5. All data in a DSA Naming Tree of a given rank must be mastered in a DSA of higher Rank. This prevents deadlock occurring for any DSA Naming Tree. Slave copies of data in a DSA Naming Tree of a given rank which are referenced within that DSA Naming Tree should be stored in DSA of a higher rank, in order to improve performance and availability.
6. All entries in a DSA Naming Tree have analogous entries in the DIT. This means that most naming authority functions are handled in the normal manner. For example, if there is an entry in the DMD DSAs Naming Tree:

   Performance Systems International,
   US,
   Common Name=DMD DSAs

   There will be an entry in the DIT of:

   Performance Systems International,
   US

7. Non-DSA (usually non-leaf) entries in a DSA Naming Tree will contain a subset of the information in the analogous DIT entry. Typically, this will either be the minimum required to support naming or a full copy of the analogous DIT entry. This data must be systematically maintained.
8. The DIT entry analogous to a DSA entry in the DSA Naming Tree will be a special object used for name assignment. The DSA Entry will be indicated by a SeeAlso attribute in the DIT Entry. This will allow users to look at DSA entries, without having to see the DSA Naming Tree. This special object class is used, as opposed to an alias, in order to prevent operational problems if the DIT entry was referred to by a DSA. This object class is defined as:

```
DSAAssignedObject OBJECT-CLASS
    SUBCLASS OF pilotObject
    MUST CONTAIN   {
       commonName,
       seeAlso }
    MAY CONTAIN {
       description,
       localityName,
       organistationName,
       organisationalUnitName }
```

DSA Naming Trees are rooted by an entry of a special object class

```
DSANamingTreeRoot OBJECT-CLASS
    MUST CONTAIN {commonName}
```

However many Ranks of DSA Naming Tree there are, there is a bootstrap problem for the root level data and the highest rank DSA Naming Tree. This is solved by naming a small number of DSAs at the root level. One of these, "Giant Tortoise", is the master for all root information and for information in the highest Rank DSA Naming Tree. The other DSAs hold copies of this information, and are intended to give higher availability of this data. It is expected that there will be a very small number of these DSAs (perhaps two or three). This information will also be replicated in most of the DSAs in the highest Rank DSA naming tree, and so the DSAs named at the root level should be accessed infrequently. In general, information in the DSA Naming Trees will be highly replicated.

Two Ranks of DSA Naming Tree will be used initially. It is not expected to increase the number of ranks to more than two in the short or medium term. The "End DSAs" DSA Naming Tree is fundamental, and so there will never be DSAs below this Rank. The "DMD DSAs" DSA Naming Tree may have DSA Naming Trees inserted above or below.

End DSAs This is used to name DSAs, which hold end information (i.e., data other than DSA Names). This will be most DSAs.

DMD DSAs This is used to name DSAs which name other DSAs. These will typically be DMD operated DSAs, used to connect pilots. In the current pilot there are typically one or two such DSAs per country.

4 Information on DSA Managers

A related problem is to obtain information on a DSA Manager. This is referenced by a the DSAManager attribute in the DSA Entry. It is quite common for this information to be particularly useful when the DSA is unavailable, and this information is often stored in the DSA in question. To solve this, a new attribute is proposed for each DSA, which holds a cache of all of the publically readable information in the DSA Manager's entry. This cache should be maintained automatically

```
attributeSyntax ATTRIBUTE-SYNTAX
    Attribute
    MATCHES FOR EQUALITY

DSAManagerInformation ATTRIBUTE
    WITH ATTRIBUTE-SYNTAX attributeSyntax
    SINGLE VALUE
```

5 Acknowledgements

Particular credit is due to Wengyik Yeong of PSI, to Marshall T. Rose of DBC and to Colin Robbins of X-Tel for useful input to this document.

References

[BK90] P. Barker and S.E. Kille. The COSINE and Internet X.500 naming architecture, December 1990. Internet Draft: draft-ietf-osids-cosinex500-02.txt.

[Kil91] S.E. Kille. Replication requirement to provide an internet directory using X.500, January 1991. Internet Draft: draft-ietf-osids-replsoln-01.txt, ps.

6 Security Considerations

Security issues are not considered in this INTERNET–DRAFT .

7 Author's Address

Steve Kille
Department of Computer Science
University College London
Gower Street
WC1E 6BT
England

Phone: +44-71-380-7294

EMail: S.Kille@CS.UCL.AC.UK