Network Working Group   S. Kille
Request for Comments: 1779   ISODE Consortium
Obsoletes: 1485   March 1995
Category: Standards Track

               A String Representation of Distinguished Names

Status of this Memo

Abstract

   The OSI Directory uses distinguished names as the primary keys to
   entries in the directory. Distinguished Names are encoded in ASN.1.
   When a distinguished name is communicated between to users not using
   a directory protocol (e.g., in a mail message), there is a need to
   have a user-oriented string representation of distinguished name.
   This specification defines a string format for representing names,
   which is designed to give a clean representation of commonly used
   names, whilst being able to represent any distinguished name.

Table of Contents

---

[1](1). Why a notation is needed

Many OSI Applications make use of Distinguished Names (DN) as defined
in the OSI Directory, commonly known as X.500 [[1](1)]. This
specification assumes familiarity with X.500, and the concept of
Distinguished Name. It is important to have a common format to be
able to unambiguously represent a distinguished name. This might be
done to represent a directory name on a business card or in an email
message. There is a need for a format to support human to human
communication, which must be string based (not ASN.1) and user
oriented. This notation is targeted towards a general user oriented
system, and in particular to represent the names of humans. Other
syntaxes may be more appropriate for other uses of the directory.
For example, the OSF Syntax may be more appropriate for some system
oriented uses. (The OSF Syntax uses "/" as a separator, and forms
names in a manner intended to resemble UNIX filenames).

[2](2). A notation for Distinguished Name

[2.1](2.1) Goals

The following goals are laid out:

   o To provide an unambiguous representation of a distinguished name

   o To be an intuitive format for the majority of names

   o To be fully general, and able to represent any distinguished name

   o To be amenable to a number of different layouts to achieve an
      attractive representation.

   o To give a clear representation of the contents of the
      distinguished name

[2.2](2.2) Informal definition

This notation is designed to be convenient for common forms of name.
Some examples are given. The author's directory distinguished name
would be written:

CN=Steve Kille,
O=ISODE Consortium, C=GB

This may be folded, perhaps to display in multi-column format. For
example:

CN=Steve Kille,
O=ISODE Consortium,
C=GB

Another name might be:

CN=Christian Huitema, O=INRIA, C=FR

Semicolon (";") may be used as an alternate separator. The
separators may be mixed, but this usage is discouraged.

CN=Christian Huitema; O=INRIA; C=FR

In running text, this would be written as <CN=Christian Huitema;
O=INRIA; C=FR>. Another example, shows how different attribute types
are handled:

CN=James Hacker,
L=Basingstoke,
O=Widget Inc,
C=GB

Here is an example of a multi-valued Relative Distinguished Name,
where the namespace is flat within an organisation, and department is
used to disambiguate certain names:

OU=Sales + CN=J. Smith, O=Widget Inc., C=US

The final examples show both methods quoting of a comma in an
Organisation name:

CN=L. Eagle, O="Sue, Grabbit and Runn", C=GB

CN=L. Eagle, O=Sue\, Grabbit and Runn, C=GB

2.3 Formal definition

   A formal definition can now be given. The structure is specified in
   a BNF grammar in Figure 1. This BNF uses the grammar defined in RFC
   822, with the terminals enclosed in <> [2]. This definition is in an
   abstract character set, and so may be written in any character set
   supporting the explicitly defined special characters. The quoting
   mechanism is used for the following cases:

    o Strings containing ",", "+", "=" or """ , <CR>, "<",
       ">", "#", or ";".

    o Strings with leading or trailing spaces

    o Strings containing consecutive spaces

   There is an escape mechanism from the normal user oriented form, so
   that this syntax may be used to print any valid distinguished name.
   This is ugly. It is expected to be used only in pathological cases.
   There are two parts to this mechanism:

   1. Attributes types are represented in a (big-endian) dotted
       notation. (e.g., OID.2.6.53).

   2. Attribute values are represented in hexadecimal (e.g. #0A56CF).
       Each pair of hex digits defines an octet, which is the ASN.1 Basic
       Encoding Rules value of the Attribute Value.

   The keyword specification is optional in the BNF, but mandatory for

this specification. This is so that the same BNF may be used for the
related specification on User Friendly Naming [5]. When this
specification is followed, the attribute type keywords must always be
present.

A list of valid keywords for well known attribute types used in
naming is given in Table 1. Keywords may contain spaces, but shall
not have leading or trailing spaces. This is a list of keywords
which must be supported. These are chosen because they appear in
common forms of name, and can do so in a place which does not
correspond to the default schema used. A register of valid keywords
is maintained by the IANA.

```
    <name> ::= <name-component> ( <spaced-separator> )
            | <name-component> <spaced-separator> <name>

    <spaced-separator> ::= <optional-space>
                    <separator>
                    <optional-space>

    <separator> ::=  "," | ";"

    <optional-space> ::= ( <CR> ) *( " " )

    <name-component> ::= <attribute>
            | <attribute> <optional-space> "+"
              <optional-space> <name-component>

    <attribute> ::= <string>
            | <key> <optional-space> "=" <optional-space> <string>

    <key> ::= 1*( <keychar> ) | "OID." <oid> | "oid." <oid>
    <keychar> ::= letters, numbers, and space

    <oid> ::= <digitstring> | <digitstring> "." <oid>
```

```
<digitstring> ::= 1*<digit>
<digit> ::= digits 0-9

<string> ::= *( <stringchar> | <pair> )
        | '"' *( <stringchar> | <special> | <pair> ) '"'
        | "#" <hex>


<special> ::= "," | "=" | <CR> | "+" | "<" |  ">"
        | "#" | ";"

<pair> ::= "\" ( <special> | "\" | '"')
<stringchar> ::= any character except <special> or "\" or '"'


<hex> ::= 2*<hexchar>
<hexchar> ::= 0-9, a-f, A-F


                Figure 1:  BNF Grammar for Distinguished Name
```

Kille   [Page 5]

```
                    Key      Attribute (X.520 keys)
                    -----------------------------
                    CN       CommonName
                    L        LocalityName
                    ST       StateOrProvinceName
                    O        OrganizationName
                    OU       OrganizationalUnitName
                    C        CountryName
                    STREET   StreetAddress


                Table 1:  Standardised Keywords
```

   Only string type attributes are considered, but other attribute

syntaxes could be supported locally (e.g., by use of the syntexes defined in [3].) It is assumed that the interface will translate from the supplied string into an appropriate Directory String encoding. The "+" notation is used to specify multi-component RDNs. In this case, the types for attributes in the RDN must be explicit.

The name is presented/input in a little-endian order (most significant component last). When an address is written in a context where there is a need to delimit the entire address (e.g., in free text), it is recommended that the delimiters <> are used. The terminator > is a special in the notation to facilitate this delimitation.

3. Examples

This section gives a few examples of distinguished names written using this notation:

CN=Marshall T. Rose, O=Dover Beach Consulting, L=Santa Clara, ST=California, C=US

CN=FTAM Service, CN=Bells, OU=Computer Science, O=University College London, C=GB

CN=Markus Kuhn, O=University of Erlangen, C=DE

CN=Steve Kille,
O=ISODE Consortium,
C=GB

CN=Steve Kille ,

O =   ISODE Consortium,
C=GB

CN=Steve Kille, O=ISODE Consortium, C=GB

4. Acknowledgements

This work was based on research work done at University College London [4], and evolved by the IETF OSI-DS WG.

Input for this version of the document was received from: Allan Cargille (University of Wisconsin); John Dale (COS); Philip Gladstone (Onsett); John Hawthorne (US Air Force); Roland Hedberg (University of Umea); Kipp Hickman (Mosaic Communications Corp.) Markus Kuhn (University of Erlangen); Elisabeth Roudier (E3X); Mark Wahl (ISODE Consortium).

5. References

[1] The Directory --- overview of concepts, models and services, 1993. CCITT X.500 Series Recommendations.

[2] Crocker, D., "Standard of the Format of ARPA-Internet Text Messages", STD 11, RFC 822, University of Delaware, August 1982.

[3] Yeong, W., Howes, T., and S. Kille, "Lightweight Directory Access Protocol", RFC 1777, Performance Systems International, University of Michigan, ISODE Consortium, March 1995.

[4] S.E. Kille. Using the OSI directory to achieve user friendly naming. Research Note RN/20/29, Department of Computer Science, University College London, February 1990.

[5] Kille, S., "Using the OSI Directory to Achieve User Friendly Naming", RFC 1781, ISODE Consortium, March 1995.

6. Security Considerations

Security issues are not discussed in this memo.

7. Author's Address

    Steve Kille
    ISODE Consortium
    The Dome
    The Square
    Richmond, Surrey
    TW9 1DT
    England

    Phone: +44-181-332-9091
    EMail: S.Kille@ISODE.COM

    DN: CN=Steve Kille,
    O=ISODE Consortium, C=GB

    UFN: S. Kille,
    ISODE Consortium, GB