Network Working Group   S. Kille
Request for Comments: 1836   ISODE Consortium
Category: Experimental   August 1995

Representing the O/R Address hierarchy in the X.500 Directory Information Tree

Status of this Memo

Abstract

This document defines a representation of the O/R Address hierarchy in the Directory Information Tree [6, 1]. This is useful for a range of purposes, including:

o Support for MHS Routing [4].

o Support for X.400/RFC 822 address mappings [2, 5].

Object Class	Mandatory
mHSCountry	M
aDMD	M
pRMD	O
mHSX121	O
mHSNumericUserIdentifier	O
mHSOrganization	O
mHSOrganizationalUnit	O
mHSPerson	O
mHSNamedObject	O
mHSTerminalID	O
mHSDomainDefinedAttribute	O

Table 1: Order of O/R Address Directory Components

1. The O/R Address Hierarchy

An O/R Address hierarchy is represented in the X.500 directory by associating directory name components with O/R Address components. An example of this is given in Figure 1. The object classes and attributes required to support this representation are defined in Figure 2. The schema, which defines the hierarchy in which these objects are represented in the directory information tree is specified in Table 1. A given object class defined in the table will always be higher in the DIT than an object class defined lower down the table. Valid combinations of O/R Address components are defined in X.400.

```
                                  /\
                                 /   \
                    C=GB        /      \   Numeric-C=234
                               /         \
                              /            \
                             /               \
                +------------+<----------------+----+
                | Country    |                 |    |
                +------------+                 +----+
                     /\
                    /   \
                   /      \
                  /         \
     ADMD=" "    /            \  ADMD=Gold 400
     +-------------+           +------------+
     |   ADMD      |           |   ADMD     |
     +-------------+           +------------+
           \                       \
             \                       \
               \ PRMD=UK.AC            \ PRMD=UK.AC
                 \                       \
                +----------+             +----+
                |  PRMD    |< -----------|    |
                +----------+             +----+
                     /
                    /
                 O=UCL
                  /
                 /
     +------------+
     | MHS-Org    |
     +------------+
          \
            \  OU=CS
              \
                \
              +-----------+
              | MHS-OU    |
              +-----------+
```

Figure 1: Example O/R Address Tree

```
IMPORTS
  ub-domain-name-length, ub-organization-name-length,
  ub-organizational-unit-name-length, ub-common-name-length,
  ub-x121-address-length, ub-domain-defined-attribute-type-length,
  ub-domain-defined-attribute-value-length, ub-terminal-id-length,
  ub-numeric-user-id-length, ub-country-name-numeric-length,
  ub-surname-length, ub-given-name-length,  ub-initials-length,
  ub-generation-qualifier-length

    FROM MTSUpperBounds {joint-iso-ccitt mhs-motis(6) mts(3)
        modules(0) upper-bounds(3) };

mHSCountry OBJECT-CLASS ::= {
    SUBCLASS OF {country}
    MAY CONTAIN {mHSNumericCountryName}
    ID oc-mhs-country}

mHSNumericCountryName ATTRIBUTE ::= {
    WITH SYNTAX NumericString (SIZE (1..ub-country-name-numeric-length))
    SINGLE VALUE
    ID at-mhs-numeric-country-name}

aDMD OBJECT-CLASS ::= {
    SUBCLASS OF {top}
    MUST CONTAIN {aDMDName}
    ID oc-admd}

aDMDName ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString {ub-domain-name-length}
    ID at-admd-name}

pRMD OBJECT-CLASS ::= {
    SUBCLASS OF {top}
    MUST CONTAIN {pRMDName}
    ID oc-prmd}

pRMDName ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString {ub-domain-name-length}
    ID at-prmd-name}

mHSOrganization OBJECT-CLASS ::= {
    SUBCLASS OF {top}
    MUST CONTAIN {mHSOrganizationName }
    ID oc-mhs-organization}
```

```
mHSOrganizationName ATTRIBUTE ::= {
    SUBTYPE OF organizationName
    WITH SYNTAX DirectoryString {ub-organization-name-length}
    ID at-mhs-organization-name}

mHSOrganizationalUnit OBJECT-CLASS ::= {
    SUBCLASS OF {top}
    MUST CONTAIN {mHSOrganizationalUnitName}
    ID oc-mhs-organizational-unit}

mHSOrganizationalUnitName ATTRIBUTE ::= {
    SUBTYPE OF organizationalUnitName
    WITH SYNTAX DirectoryString {ub-organizational-unit-name-length}
    ID at-mhs-organizational-unit-name}

mHSPerson OBJECT-CLASS ::= {
    SUBCLASS OF {top}
    MUST CONTAIN {mHSSurname}
    MAY CONTAIN {mHSGivenName|
                mHSInitials|
                mHSGenerationalQualifier}
    ID oc-mhs-person}

mHSSurname ATTRIBUTE ::= {
    SUBTYPE OF surname
    WITH SYNTAX DirectoryString {ub-surname-lenght}
    ID at-mhs-surname}

mHSGivenName ATTRIBUTE ::= {
    SUBTYPE OF givenName
    WITH SYNTAX DirectoryString {ub-given-name-length}
    ID at-mhs-given-name}

mHSInitials ATTRIBUTE ::= {
    SUBTYPE OF initials
    WITH SYNTAX DirectoryString {ub-initials-length}
    ID at-mhs-initials}

mHSGenerationQualifier ATTRIBUTE ::= {
    SUBTYPE OF generationQualifier
    WITH SYNTAX DirectoryString {ub-generation-qualifier-length}
    ID at-mhs-generation-qualifier}

mHSNamedObject OBJECT-CLASS ::= {
    SUBCLASS OF {top}
    MUST CONTAIN {mHSCommonName}
    ID oc-mhs-named-object}
```

```
mHSCommonName ATTRIBUTE ::= {
    SUBTYPE OF commonName
    WITH SYNTAX DirectoryString {ub-common-name-length}
    ID at-mhs-common-name}

mHSX121 OBJECT-CLASS ::= {
    SUBCLASS OF {top}
    MUST CONTAIN {mHSX121Address}
    ID oc-mhs-x121}

mHSX121Address ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString {ub-x121-address-length}
    ID at-x121-address}

mHSDomainDefinedAttribute OBJECT-CLASS ::= {
    SUBCLASS OF {top}
    MUST CONTAIN {
        mHSDomainDefinedAttributeType|
        mHSDomainDefinedAttributeValue}
    ID oc-mhs-domain-defined-attribute}

mHSDomainDefinedAttributeType ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString {ub-domain-defined-attribute-type-length}
    SINGLE VALUE
    ID at-mhs-domain-defined-attribute-type}

mHSDomainDefinedAttributeValue ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString {ub-domain-defined-attribute-value-length}
    SINGLE VALUE
    ID at-mhs-domain-defined-attribute-value}

mHSTerminalID OBJECT-CLASS ::= {
    SUBCLASS OF {top}
    MUST CONTAIN {mHSTerminalIDName}
    ID oc-mhs-terminal-id}

mHSTerminalIDName ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString {ub-terminal-id-length}
    ID at-mhs-terminal-id-name}

mHSNumericUserIdentifier OBJECT-CLASS ::= {
    SUBCLASS OF {top}
    MUST CONTAIN {mHSNumericUserIdentifierName}
    ID oc-mhs-numeric-user-id}
```

```
mHSNumericeUserIdentifierName ATTRIBUTE ::= {
    SUBTYPE OF name
    WITH SYNTAX DirectoryString {ub-numeric-user-id-length}       150
    ID at-mhs-numeric-user-id-name}
```

Figure 2: O/R_Address Hierarchy

The hierarchy is defined so that:

1. The representation is defined so that it is straightforward to make a mechanical transformation in either direction. This requires that each node is named by an attribute whose type can determine the mapping.

2. Where there are multiple domain defined attributes, the first in the sequence is the most significant.

3. Physical Delivery (postal) addresses are not represented in this hierarchy. This is primarily because physical delivery can be handled by the Access Unit routing mechanisms defined in [4], and there is no need for this representation.

4. Terminal and network forms of address are not handled, except for X.121 form, which is useful for addressing faxes.

5. MHSCountry is defined as a subclass of Country, and so the same entry will be used for MHS Routing as for the rest of the DIT.

6. The numeric country code will be an alias.

7. ADMD will always be present in the hierarchy. This is true in the case of " " and of "0". This facilitates an easy mechanical transformation between the two forms of address.

8. Each node is named by the relevant part of the O/R Address.

9. Aliases may be used in other parts of the tree, in order to normalise alternate values. Where an alias is used, the value of the alias should be present as an alternate value in the node aliased to. Aliases may not be used for domain defined attributes.

10. Domain Defined Attributes are named by a multi-valued RDN (Relative Distinguished Name), consisting of the type and value. This is done so that standard attribute syntaxes can be used.

11. Where an O/R Address has a valid Printable String and T.61 form, both must be present, with one as an alias for the other. This is so that direct lookup of the name will work, independent of the variant used. When both are present in an O/R Address being looked up, either may be used to construct the distinguished name.

12. Personal name is handled by use of the mHSPerson object class. Each of the components of the personal name will be present in the relative distinguished name, which will usually be multi-valued.

The relationship between X.400 O/R Addresses and the X.400 Entries (Attribute Type and Object Class) are given in Table 2. Where there are multiple Organizational Units or Domain Defined Attributes, each component is mapped onto a single X.500 entry.

Note: When an X.121 address is used for addressing fax transmission, this may only be done relative to the PRMD or ADMD. This is in line with the current X.400 standards position. This means that it is not possible to use this form of addressing for an organisational or departmental fax gateway service.

O/R Address	Object Class	Naming Attribute
C	mHSCountry	countryName or mHSNumericCountryName
A	aDMD	aDMDName
P	pRMD	pRMDName
O	mHSOrganization	mHSOrganizationName
OU/OU1/OU2 OU3/OU4	mHSOrganizationalUnit	mHSOrganizationalUnitName
PN	mHSPerson	personName
CN	mHSNamedObject	mHSCommonName
X121	mHSX121	mHSX121Address
T-ID	mHSTerminalID	mHSTerminalIDName
UA-ID	mHSNumericUserIdentifier	mHSNumericUserIdentifierName
DDA	mHSDomainDefinedAttribute	mHSDomainDefinedAttributeType and mHSDomainDefinedAttributeValue

Table 2: O/R Address relationship to Directory Name

2. Notation

O/R Addresses are written in the standard X.400 Notation. Distinguished Names use the string representation of distinguished names defined in [3]. The keywords used for the attributes defined in this specification are given in Table 3.

3. Example Representation

The O/R Address:

```
I=S; S=Kille; OU1=CS; O=UCL,
P=UK.AC; A=Gold 400; C=GB;
```

would be represented in the directory as:

```
MHS-I=S + MHS-S=Kille, MHS-OU=CS, MHS-O=UCL,
```

Attribute	Keyword
mHSNumericCountryName	MHS-Numeric-Country
aDMDName	ADMD
pRMDName	PRMD
mHSOrganizationName	MHS-O
mHSOrganizationalUnitName	MHS-OU
mHSSurname	MHS-S
mHSGivenName	MHS-G
mHSInitials	MHS-I
mHSGenerationalQualifier	MHS-GQ
mHSCommonName	MHS-CN
mHSX121Address	MHS-X121
mHSDomainDefinedAttributeType	MHS-DDA-Type
mHSDomainDefinedAttributeValue	MHS-DDA-Value
mHSTerminalIDName	MHS-T-ID
mHSNumericeUserIdentifierName	MHS-UA-ID

Table 3: Keywords for String DN Representation

```
PRMD=UK.AC, ADMD=Gold 400, C=GB
```

4. Mapping from O/R Address to Directory Name

The primary application of this mapping is to take an X.400 encoded O/R Address and to generate an equivalent directory name. This mapping is only used for selected types of O/R Address:

- o Mnemonic form
- o Numeric form
- o Terminal form, where country is present and X121 addressing is used

Other forms of O/R address are handled by Access Unit mechanisms. The O/R Address is treated as an ordered list, with the order as defined in Table 1. For each O/R Address attribute, generate the equivalent directory naming attribute. In most cases, the mapping is mechanical. Printable String or Teletex encodings are chosen as appropriate. Where both forms are present in the O/R Address, either form may be used to generate the distinguished name. Both will be represented in the DIT. There are two special cases:

1. A DDA generates a multi-valued RDN
2. The Personal Name is mapped to a multi-valued RDN

In many cases, an O/R Address will be provided, and only the higher components of the address will be represented in the DIT. In this case, the "longest possible match" should be returned.

5. Mapping from Directory Name to O/R Address

The reverse mapping is also needed in some cases. All of the naming attributes are unique, so the mapping is mechanically reversible.

6. Acknowledgements

Acknowledgements for work on this document are given in [4].

References

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

[2] Kille, S., "Mapping between X.400(1988)/ISO 10021 and RFC 822", RFC 1327, Department of Computer Science, University College London, May 1992.

[3] Kille, S., "A String Representation of Distinguished Names", RFC 1779, Department of Computer Science, University College London, March 1995.

[4] Kille, S., "MHS Use of the X.500 Directory to Support MHS Routing, RFC 1801, ISODE Consortium, June 1995.

[5] Kille, S., "Use of the X.500 Directory to Support Mapping between X.400 and RFC 822 Addresses, RFC 1838, ISODE Consortium, August 1995.

[6] CCITT recommendations X.400 / ISO 10021, April 1988. CCITT SG 5/VII / ISO/IEC JTC1, Message Handling: System and Service Overview.

7. Security Considerations

Security issues are not discussed in this memo.

8. Author's Address

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

Phone: +44-81-332-9091
Internet EMail: S.Kille@ISODE.COM

X.400: I=S; S=Kille; O=ISODE Consortium; P=ISODE;
A=Mailnet; C=FI;

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

UFN: S. Kille, ISODE Consortium, GB

A. Object Identifier Assignment

```
-----------------------------------------------------------------------
mhs-ds OBJECT IDENTIFIER ::= {iso(1) org(3) dod(6) internet(1)
          private(4) enterprises(1) isode-consortium (453) mhs-ds (7)}


tree OBJECT IDENTIFIER ::= {mhs-ds 2}

oc OBJECT IDENTIFIER ::= {tree 1}
at OBJECT IDENTIFIER ::= {tree 2}

oc-admd OBJECT IDENTIFIER ::= {oc 1}
oc-mhs-country OBJECT IDENTIFIER ::= {oc 2}
oc-mhs-domain-defined-attribute OBJECT IDENTIFIER ::= {oc 3}
oc-mhs-named-object OBJECT IDENTIFIER ::= {oc 4}
oc-mhs-organization OBJECT IDENTIFIER ::= {oc 5}
oc-mhs-organizational-unit OBJECT IDENTIFIER ::= {oc 6}
oc-mhs-person OBJECT IDENTIFIER ::= {oc 7}
oc-mhs-x121 OBJECT IDENTIFIER ::= {oc 8}
oc-prmd OBJECT IDENTIFIER ::= {oc 9}
oc-mhs-terminal-id OBJECT IDENTIFIER ::= {oc 10}
oc-mhs-numeric-user-id OBJECT IDENTIFIER ::= {oc 11}

at-admd-name OBJECT IDENTIFIER ::= {at 1}
at-mhs-common-name OBJECT IDENTIFIER ::= {at 2}
at-mhs-domain-defined-attribute-type OBJECT IDENTIFIER ::= {at 3}
at-mhs-domain-defined-attribute-value OBJECT IDENTIFIER ::= {at 4}
at-mhs-numeric-country-name OBJECT IDENTIFIER ::= {at 5}
at-mhs-organization-name OBJECT IDENTIFIER ::= {at 6}
at-mhs-organizational-unit-name OBJECT IDENTIFIER ::= {at 7}
at-prmd-name OBJECT IDENTIFIER ::= {at 10}
at-x121-address OBJECT IDENTIFIER ::= {at 12}
at-mhs-terminal-id-name OBJECT IDENTIFIER ::= {at 13}
at-mhs-numeric-user-id-name  OBJECT IDENTIFIER ::= {at 14}
at-mhs-surname OBJECT IDENTIFIER ::= {at 15}
at-mhs-given-name OBJECT IDENTIFIER ::= {at 16}
at-mhs-initials OBJECT IDENTIFIER ::= {at 17}
at-mhs-generation-qualifier OBJECT IDENTIFIER ::= {at 18}
```

Figure 3: Object Identifier Assignment