Network Working Group
INTERNET-DRAFT

S.E. Hardcastle-Kille
ISODE Consortium

November 1992
Expires: June 1993

MHS use of the Directory to support distribution lists

Status of this Memo

This document is an Internet Draft. Internet Drafts are working documents of the Internet Engineering Task Force (IETF), its Areas, and its Working Groups. Note that other groups may also distribute working documents as Internet Drafts.

Internet Drafts are draft documents valid for a maximum of six months. Internet Drafts may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet Drafts as reference material or to cite them other than as a "working draft" or "work in progress."

Please check the I-D abstract listing contained in each Internet Draft directory to learn the current status of this or any other Internet Draft.

Abstract

This document specifies a procedure for managing distribution lists using the directory. This is an extension of the mechanism defined in X.400 [MHS88].

This draft document will be submitted to the RFC editor as a protocol standard. Distribution of this memo is unlimited. Please send comments to the author or to the discussion group `<mhs-ds@mercury.udev.cdc.com>`.

*** NOTE **** Information from EXPLODE project has not been factored into this note

1 Model

This section describes the operation of Distribution Lists, including use of the OSI Directory. There are two models of distribution list expansion:

1. Following the standard
2. UA level expansion

The standard mechanism needs no explanation here. The UA leve model is almost the same. The differences are

- The MTS originator is changed to a list manager.
- The MTS identifier is changed (doing the first without the second is not really sensible).
- In terms of the model, the message is delivered, then the list is expanded outside the MTS, and then the message is resubmitted. Essentially, this makes list expansion a "firewall".
- DL Expansion history is submitted with the new message.

In practice, the UA level expansion is very close to the standard. Disadvantages of the UA level approach are:

- It is non-standard
- It provides less information for elimination of duplicates

Features which can be argued as an advantage of either approach are:

- The only service which is lost, is the ability to return to the originator errors which occur after expansion.

  The argument in favour of this is that this is sometimes a useful service.

  The argument against this is that where such a service is required, it is better to achieve this by a mechanism where the orginatinge UA expands the distribution list prior to message submission.

The advantages of the UA level approach are:

- Where a list has a member on RFC 822 or X.400(84), errors will always be sent to the list maintainer (they will go to the originator if the standard is followed).

```
listUA OBJECT−CLASS
   SUBCLASS OF routedUA
   MUST CONTAIN {distributionListName}
   ::= oc−list−ua

distributionListName ATTRIBUTE
   SUBTYPE OF userName
   SINGLE VALUE
   ::= at−distribution−list−name
```

Figure 1: O/R Address of Distribution List

- Support for lists in the context of X.400(1984) and RFC 822.
- After the expansion, parameters (e.g. message priority) come under the control of the list manager (whose policy can take the initial parameters into consideration).
- It gives a basis for extension to more powerful variations of distributed lists, where some additional processing is done at the expansion point (e.g. moderated lists).

In cases where lists follow the UA level model, they must correctly deal with other MTAs which follow the standard model.

2 Identifying Lists

It is important to be able to identify that an object is a list either from its directory name, or from its O/R Address, and to be able to determine the other form.

- When the distinguished name is used, it is identified as a list from the object class of the entry (`mhs-distribution-list`) The `mhs-or-addresses` attribute in the list entry should point at the O/R Address form.
- When the O/R address form is used to identifies a list, it will also be of object class `listUA`, as defined in Figure 1, and the attribute `distributionListName` will point at the distinguished name.

Lists may have aliases pointing to them. It seems very desirable to establish a part of the DIT to hold well known lists, to facilitate the location of such lists. For example, the list of the WG which is discussing this might have an alias:

```
CN=MHS-DS, OU=Lists, O=Internet
```

3 Schema Definitions

The use of the OSI Directory to provide distribution lists defined in X.400 is a suitable basis for Distribution lists. The extensions defined in this specification are given in Figure 2.

```
IMPORTS
        mhs-distribution-list, mhs-dl-members, mhs-or-name-syntax;
                FROM {X.402}


distributionList OBJECT-CLASS
        SUBCLASS OF mhs-distribution-list
        MAY CONTAIN {dlPolicy,
                dlAccessControl,
                dlErrorsToName
                dlDynamicMembers}
        ::= oc-distribution-list

dlErrorsTo OBJECT-CLASS
        SUBCLASS OF ua

dlErrorsToName  ATTRIBUTE
        WITH ATTRIBUTE-SYNTAX mhs-or-name-syntax
        MULTI VALUE
        ::= at-dl-errors-to-name

dlPolicy ATTRIBUTE
        WITH ATTRIBUTE-SYNTAX DlPolicy
        SINGLE VALUE
        MATCHES FOR EQUALITY
        ::= at-dl-policy

DlPolicy ::= SEQUENCE {
        operation-mode [0] ENUMERATED {
                standard(0),
                p2-level(1)},
        strip-trace [1] ENUMERATED {
                strip-all(0),
                leave-all(1),
                leave-first-element(2)} DEFAULT leave-first-element,
        futher-expansion-permitted [2] BOOLEAN DEFAULT TRUE,
        conversion-prohibited [3] MappedBoolean DEFAULT original,
        priority [4] ENUMERATED {
                original (1),
                low (2),
                normal (3),
                high (4)} DEFAULT low
        suppress-warnings [5] BOOLEAN DEFAULT TRUE,
        submit-dn-only [6] BOOLEAN DEFAULT FALSE,
        public-expansion-ok [7] BOOLEAN DEFAULT TRUE,
        disclosure-of-recipients [8] MappedBoolean DEFAULT false
        }

MappedBoolean ::=  ENUMERATED {
```

```
                    original (1),
                    false (2),
                    true (3) }

dlDynamicMembers ATTRIBUTE
          WITH ATTRIBUTE-SYNTAX DLDynamicMembers
          MULTI VALUE
          ::= at-dl-dynamic-members

DLDynamicMembers ::= SEQUENCE {
                    base DistinguishedName,
                    filter Filter
                    search ENUMERATED {single-level(0), subtree(1)}
                              DEFAULT subtree}

dlAccesControl ::= ATTRIBUTE
          WITH ATTRIBUTE-SYNTAX ENUMERATED {
                    no-specific-controls (0),
                    add-and-delete-self (1) }
          ::= at-dl-access-control
```

Figure 2: List Definitions

The additions to the standard are:

- Definition of an address to which errors are returned.
- Definition of an attribute to control the policy of list expansion.
- Definition of a means for dynamic list expansion.

A list need not have any members. List management interfaces should ensure that this is not done accidentally.

The `dlErrorsToName` attribute defines the O/R Name of where list errors should be directed. Typically, this might follow the "list-request" convention. The O/R Address may be omitted from the O/R Name. In this case a directory entry of this name must exist to identify the O/R address. The object class `dlErrorsTo` is created solely for this purpose.

Note that the access control for managing the membership of the list is dealt with by the normal Directory access control mechanisms. The owner of the list may be identified by the "owner" attribute. This identifies the user to whom queries about the list (e.g., requests to be added) should be addressed.

The policy of list expansion is represented by the `dlPolicy` attribute. This provides the following options:

operation-mode This defines whether expansion of the list should follow X.400, or be UA level as defined here.

strip-trace This only affects UA level expansion. If trace is stripped, as full UA level expansion would require, the original submission date is lost. Therefore, some or all trace may be resubmitted.

futher-expansion-permitted This controls whether further DL expansion of this list is allowed.

conversion-prohibited This allows the list to override the conversion prohibition of the original message.

priority This allows the priority of the original message to be overridden. Typically, the priority of all messages will be set to low.

suppress-warnings Don't send delay warnings to the error return address.

submit-dn-only If the DN (Distinguished Name) is present, do not submit the O/R address. This forces submit time directory name lookup.

public-expansion-ok This allows any directory-capable MTA to expand the list (not just the MTA responsible).

disclose-recipients Allows disclosure of recipients to be controlled by the list manager. This will generally be set to false.

The `dlDynamicMembers` attribute defines a means for expressing members of a list based on the directory search operation. This might, for example, be used to define a list based on userclass of student. It may be more useful to define a list in this way, than to explicitly list members. This is expanded by a subtree or single level search using the filter. If there is a sizelimit error, the message should be rejected, with an appropriate error message. If there are no matches, an error report should be sent to the orginatore and a local administrator should be warned.

The `dlAccessControl` attribute allows distribution list specific access control to be enabled. In particular, it allows a distribution list to be operated with add/delete self capability. To achieve this, the supporting DSA will need to recognise the syntax of the `mhs-dl-members` attribute and be able to recognise that the distinguished name in this matches that of the bound UA.

There is a user requirement to annotate the membership of a list. This (standard) approach does not provide a sensible means to achieve this, as there is no means to add per-user information.

**** Make members of DL optional (need to repeat X.402 stuff), as this is overridden by dynamic stuff

4 Operation of Distribution Lists

**** Reference X.402 as basic list model

Each distribution lists is represented as an Entry in the Directory. They will be given names relative to an Organisation or Organisational Unit. A set of sibling lists will usually be supported by a set of MTAs.

The members of a list (`mhs-dl-members`) will be of three types:

- Directory Name only. This will force submission-time expansion.
- OR Address only. This will be for users not in the directory.
- Both, which should become increasingly common. The Directory Name is the "managed" portion, with the O/R Address derived as an optimisation. The list management tool should be able to update O/R Addresses. User interfaces should facilitate input of RFC 822 addresses.

The permission to submit (attribute `mhs-dl-submit-premissions`) is handled in the following manner:

individual In the obvious way.

member-of-dl The members of the identified distribution list are allowed to submit messages to the list. Where the identified list contains lists as members, this right is not conferred to members of those sublists.

pattern-match As described in X.400. This might be extended slightly after initial experience.

member-of-group Here, the property "members" of the name given identifies users allowed to submit messages. This might be an expensive calculation, as in many cases, the Directory Name of the submitter will not be present. It might be sensible to avoid this one for now.

If the attribute is omitted, this is interpreted as public rights: that is any user may submit messages to the list

Note: It may be useful to extend submit permissions in a manner analogous to DynamicMembers, in order to have dynamic control of submission rights.

******* Note P3 expansion problem

References

[HK92] S.E. Hardcastle-Kille. MHS use of the directory to support MHS routing, April 1992. Internet Draft.

[MHS88] CCITT recommendations X.400 / ISO 10021, April 1988. CCITT SG 5/VII / ISO/IEC JTC1, Message Handling: System and Service Overview.

5 Security Considerations

Security considerations are not discussed in this INTERNET–DRAFT .

6 Author's Address

Steve Hardcastle-Kille
ISODE Consortium
PO Box 505
London
SW11 1DX
England

Phone: +44-71-223-4062

EMail: S.Kille@ISODE.COM

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

UFN: S. Hardcastle-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 (3)}

list OBJECT IDENTIFIER ::= {mhs-ds 5}

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



oc-list-ua OBJECT IDENTIFIER ::= {oc 1}
oc-distribution-list OBJECT IDENTIFIER ::= {oc 2}


at-distribution-list-name OBJECT IDENTIFIER ::= {at 1}
at-dl-errors-to-name OBJECT IDENTIFIER ::= {at 2}
at-dl-policy OBJECT IDENTIFIER ::= {at 3}
at-dl-dynamic-members OBJECT IDENTIFIER ::= {at 4}
at-dl-access-control OBJECT IDENTIFIER ::= {at 5}
```

Figure 3: Object Identifier Assignment