Network Working Group
INTERNET-DRAFT

S.E. Hardcastle-Kille
ISODE Consortium

November 1992
Expires: January 1993

MHS use of Directory to support MHS Content Conversion

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

User Agents have various capabilities for support of multimedia messages. To facilitate interworking between UAs of differing capabilities, it is useful for the MTS to be able to perform content conversion. This document specifies an approach for X.400 Message Handling Systems to perform application level routing using the OSI Directory in order to support content conversion. This document assumes MHS use of directory to perfom routing accoreding to "MHS use of Directory to support MHS Routing" [HK92].

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>.

1 Protocol Extensions

A number of protocol extensions are needed, primarily to support content conversion. These extensions may be used with either MTA Abstract Service (P1) or MTS Abstract Service (P3). The per-recipient extensions are given in Figure 1.

The following per-recipient extensions are defined.

general-explicit-conversion The explicit conversion of X.400 only allows for a listed set of conversions. This attribute allows for any conversion of body parts to be requested. Built in EITs must be represented by their object identifiers.

content-conversion This requests for different content types to be converted (e.g., content type 22 to 2 downgrade).

source-route This gives a sequence of points (either MTAs or MDs) which must be traversed in the order given. The points may be accessed directly or indirectly for each step. This should be stripped as each point is reached. If this element is marked as critical for transfer, the route is mandatory, otherwise it is advisory. There is some risk of spurious loops being detected. This can be specified by a UA or an MTA. An MTA may introduce further source routing if a need arises.

route-barring This gives a list of points which must *not* be traversed. If this element is marked as critical for transfer, the barring is mandatory, otherwise it is advisory.

route-restriction This give a list of points which the route must be restricted to (e.g., to constrain costs).

The following per-message extensions are defined in Figure 2. These are:

warning-interval Some MTAs offer a facility to send (IPMS) warnings about a messages delayed transit. This parameter allows the warning interval to be specified.

last-warning This parameter specifies the time at which the last warning was sent. It will ensure that warnings are sent to the originator at even intervals.

2 Format Conversion

An MTA will determine reformatting requirements for a message in two ways:

1. First, any explicitly requested conversions will be dealt with

```
general-explicit-conversion EXTENSION
    GeneralExplicitConversion
    ::= id-general-explicit-conversion

GeneralExplicitConversion ::= SEQUENCE {
    from          ExternalEncodedInformationType,
    to      ExternalEncodedInformationType }

content-conversion EXTENSION
    ContentConverson
    ::= id-content-conversion

ContentConversion ::= SEQUENCE {
    from          ContentType,
    to      ContentType }

source-route EXTENSION
    SourceRoute
    ::= id-source-route

SourceRoute ::= SEQUENCE OF RouteElement

RouteElement ::= CHOICE {
    mta [0]       DistinguishedName,
    md [1]        DistinguishedName }

route-barring EXTENSION
    RouteBarring
    ::= id-route-barring

RouteBarring ::= SET OF RouteElement

route-restriction EXTENSION
    RouteRestriction
    ::= id-route-restriction

RouteRestriction ::= SET OF RouteElement
```

Figure 1: Per-recipient Protocol Extensions

```
warning−interval EXTENSION       −− interval between user warnings
   INTEGER                       −− in minutes
   ::= id−warning−interval

last−warning EXTENSION           −− time last warning was send
   UTCTime
   ::= id−last−warning
```

Figure 2: Per-message Protocol Extensions

2. Second, the recipients capabilities are determined (from the directory), and any necessary conversions made. UA capability may be determined from any of:

   - The User's entry
   - The O/R Address entry
   - A subtree capability restriction as defined in [HK92].

   A choice to do reformatting can only be made by an MTA with access to this information. Once an MTA has determined a requirement to do reformatting, it should attempt to do this. If this cannot be done, or if the requirements are unknown, the message should be routed without conversion.

In many cases, the MTA will be able to perform conversion locally. In some cases, particularly simple MTAs, it may be necessary to perform a conversion at a remote MTA. First the remote MTA must be identified. Each MTA will maintain a list of MTAs (or MDs??) it uses for performing remote conversions. Any MTA offering conversions will register them in the directory. Thus it is a matter of simple matching to determine if one of the MTAs used for conversion services offers the conversion needed (perhaps in more than one stage). The attributes needed to do this are defined in Figure 3.

The attributes defined are:

mTAUsedForConversion This defines the set of MTAs which are used by an MTA to provide conversion.

bodyPartConversionService This defines the services offered by an MTA for body part conversion.

contentConversionService This defines the services offered by an MTA for content type conversion (e.g., 22 to 2 downgrade).

This facility can be used for either body part conversion or content type conversion.

```
mTAUsedForConversion ATTRIBUTE
   WITH ATTRIBUTE-SYNTAX distinguishedNameSyntax
   ::= at-mta-used-for-conversion

bodyPartConversionService ATTRIBUTE
   WITH ATTRIBUTE-SYNTAX GeneralExplicitConversion
   ::= at-body-part-conversion-service

contentConversionService ATTRIBUTE
   WITH ATTRIBUTE-SYNTAX ContentConversion
   ::= at-content-conversion-service
```

Figure 3: Format Conversion Registration

In some cases, the MTA performing the conversion will be accessed indirectly. In this case, a mandatory source route should be specified, in order to ensure that the message is routed to the correct MTA.

References

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

3 Security Considerations

Security considerations are not discussed in this INTERNET–DRAFT .

4 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 (7)}

conversion OBJECT IDENTIFIER ::= {mhs-ds 7}

oc OBJECT IDENTIFIER ::= {routing 1}
at OBJECT IDENTIFIER ::= {routing 2}
id OBJECT IDENTIFIER ::= {routing 3}


oc-conversion-mta OBJECT IDENTIFIER ::= {oc 1}

at-body-part-conversion-service OBJECT IDENTIFIER ::= {at 1}
at-content-conversion-service OBJECT IDENTIFIER ::= {at 2}
at-mta-used-for-conversion OBJECT IDENTIFIER ::= {at 3}

id-alternative-address-information OBJECT IDENTIFIER ::= {id 1}
id-content-conversion OBJECT IDENTIFIER ::= {id 2}
id-general-explicit-conversion OBJECT IDENTIFIER ::= {id 3}
id-logging-level OBJECT IDENTIFIER ::= {id 4}
id-route-barring OBJECT IDENTIFIER ::= {id 5}
id-route-restriction OBJECT IDENTIFIER ::= {id 6}
id-source-route OBJECT IDENTIFIER ::= {id 7}
id-last-warning OBJECT IDENTIFIER ::= {id 8}
id-warning-interval OBJECT IDENTIFIER ::= {id 9}
```

Figure 4: Object Identifier Assignment