



INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.440

(06/1999)

SERIES X: DATA NETWORKS AND OPEN SYSTEM
COMMUNICATIONS

Message Handling Systems

Message handling systems: Voice messaging system

ITU-T Recommendation X.440

ITU-T X-SERIES RECOMMENDATIONS
DATA NETWORKS AND OPEN SYSTEM COMMUNICATIONS

PUBLIC DATA NETWORKS	
Services and facilities	X.1–X.19
Interfaces	X.20–X.49
Transmission, signalling and switching	X.50–X.89
Network aspects	X.90–X.149
Maintenance	X.150–X.179
Administrative arrangements	X.180–X.199
OPEN SYSTEMS INTERCONNECTION	
Model and notation	X.200–X.209
Service definitions	X.210–X.219
Connection-mode protocol specifications	X.220–X.229
Connectionless-mode protocol specifications	X.230–X.239
PICS proformas	X.240–X.259
Protocol Identification	X.260–X.269
Security Protocols	X.270–X.279
Layer Managed Objects	X.280–X.289
Conformance testing	X.290–X.299
INTERWORKING BETWEEN NETWORKS	
General	X.300–X.349
Satellite data transmission systems	X.350–X.369
IP-based networks	X.370–X.399
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	
Networking	X.600–X.629
Efficiency	X.630–X.639
Quality of service	X.640–X.649
Naming, Addressing and Registration	X.650–X.679
Abstract Syntax Notation One (ASN.1)	X.680–X.699
OSI MANAGEMENT	
Systems Management framework and architecture	X.700–X.709
Management Communication Service and Protocol	X.710–X.719
Structure of Management Information	X.720–X.729
Management functions and ODMA functions	X.730–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	
Commitment, Concurrency and Recovery	X.850–X.859
Transaction processing	X.860–X.879
Remote operations	X.880–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999
TELECOMMUNICATION SECURITY	X.1000–

For further details, please refer to the list of ITU-T Recommendations.

ITU-T Recommendation X.440

Message handling systems: Voice messaging system

Summary

This revision of Recommendation X.440 is a consolidation of Recommendation X.440 (09/92) and Amendment 1 (11/95).

This Recommendation defines an OSI Message Handling System protocol that is intended to be used in the exchange of voice encoded messages in a store and forward manner as specified in Recommendation F.440.

The protocol specified in this Recommendation is based on the ITU-T X.400-series Recommendations and ISO/IEC 10021 Parts 1 to 7.

Source

Following the ITU-T decision to publish new editions of the set of Message Handling Recommendations, this edition of ITU-T Recommendation X.440, dated 18 June 1999, consolidates X.440 (09/92) and X.440 Amendment 1 (11/95).

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSC Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g. interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

© ITU 2004

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

CONTENTS

	Page
1 Scope	1
2 Normative references	1
2.1 Identical Recommendations International Standards	1
2.2 Paired Recommendations International Standards equivalent in technical content	2
2.3 Additional references	2
3 Definitions	2
3.1 Common definitions for MHS	2
3.2 Common definitions for Abstract Syntax Notation One	3
3.3 Voice service definitions	3
3.4 Voice Messaging System Definitions	3
4 Abbreviations	3
5 Conventions	4
5.1 Terms	4
5.2 ASN.1	4
6 Information objects	5
7 Common data types	5
7.1 VM Identifier	5
7.2 OR Descriptor	6
7.3 Spoken Name	6
7.4 Extensions	6
8 Voice message	6
8.1 Heading fields	8
8.2 Body part types	13
9 Voice notifications	15
9.1 Common fields	16
9.2 Receipt notifications	19
9.3 Service notifications	19
9.4 Non-receipt notifications	20
10 Primary object types	22
10.1 Voice messaging system user	23
10.2 Voice messaging system	23
11 Primary port types	23
11.1 Origination port	24
11.2 Reception port	24
11.3 Management port	24
12 Abstract operations	24
12.1 Origination abstract operations	25
12.2 Reception abstract operations	26
12.3 Management abstract operations	27
12.4 Performance of management operations	29
13 Abstract errors	30
13.1 Subscription error	30
13.2 Recipient improperly specified	30
14 Other capabilities	30
15 Secondary object types	31
15.1 VMGS user agent	32
15.2 VMGS message store	32
15.3 Telephone service access unit	33
15.4 Message transfer system	33

	Page
16 Secondary port types	33
16.1 Submission port	33
16.2 Delivery port	33
16.3 Retrieval port	34
16.4 Administration port	34
16.5 Import port	34
16.6 Export port	34
16.7 Management port	34
17 User agent operation	34
17.1 Performance of origination operations	34
17.2 Invocation of reception operations	36
17.3 Internal procedures	37
18 Message Store operation	42
18.1 Binding to the MS	42
18.2 Creation of Information Objects	43
18.3 Maintenance of attributes	45
18.4 Voice Messaging MS auto forward action types	46
18.5 Voice messaging MS auto service notification action type	50
18.6 Message store attributes	51
18.7 Procedures for Voice Messaging MS	64
19 Message contents	65
19.1 Content	65
19.2 Content type	65
19.3 Content length	65
19.4 Encoded information types	65
20 Port realization	66
21 Conformance	66
21.1 Origination versus reception	66
21.2 Requirements of a VMG conformance statement	66
Annex A – VM Object Identifiers – Reference definition	67
Annex B – VM Abstract Information Objects – Reference definition	70
Annex C – VM Message Store attributes – Reference definition	78
Annex D – Message Store VMGS Auto-Action types – Reference definition	86
Annex E – VMGS functional objects – Reference definition	88
Annex F – VMGS abstract service – Reference definition	91
Annex G – VMGS upper bounds parameters – Reference definition	94
Annex H – Mapping of G.726 (1990) and G.728 (1992) into the data component of a voice body part	95
H.1 Illustrative mapping of the analogue/digital converted data	95
H.2 Support for CCITT Rec. G.726 – 32 kbit/s ADPCM	95
H.3 Support for CCITT Rec. G.728 – 16 kbit/s LD-CELP	95
INDEX	97

ITU-T Recommendation X.440

Message handling systems: Voice messaging system

1 Scope

This Recommendation is one of a set of ITU-T Recommendations for Message Handling. The entire set provides a comprehensive blueprint for a message handling system (MHS) realized by any number of cooperating open systems.

The purpose of an MHS is to enable users to exchange messages on a store-and-forward basis. A message submitted on behalf of one user, the originator, is conveyed by the message transfer system (MTS) and subsequently delivered to the agents of one or more additional users, the recipients. Access units (AU) link the MTS to other non-MHS services such as Teletex or the Telephone Service. Effectively, an AU enables a subscriber of a non-MHS service to become an indirect user of MHS. An MHS direct user is assisted in the preparation, storage, and rendering of messages by a user agent (UA). Optionally, it is assisted in the storage of messages by a message store (MS). The MTS comprises a number of message transfer agents (MTA) which collectively perform the store-and-forward message transfer function.

This Recommendation defines the message handling application called a *voice messaging system* which specifies *voice messaging* (VMG), a form of message handling tailored for exchange of (voice) encoded information between users, a new message content type and associated procedures known as P_{vm} . It is designed to meet the requirements of users of CCITT Recommendation G.726 (1990) 32 kbit/s ADPCM and other commonly used voice syntaxes.

This Recommendation is one of a series on message handling. ITU-T Rec. X.402 | ISO/IEC 10021-2 constitutes the introduction to the series and identifies the other documents in it.

The architectural basis and foundation for message handling are defined in still other ITU-T Recommendations | International Standards. ITU-T Rec. X.402 | ISO/IEC 10021-2 identifies those documents as well.

2 Normative references

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision: all users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published.

2.1 Identical Recommendations | International Standards

- ITU-T Recommendation X.402 (1999) | ISO/IEC 10021-2:2003, Information technology – *Message Handling Systems (MHS): Overall architecture*.
- ITU-T Recommendation X.411 (1999) | ISO/IEC 10021-4:2003, Information technology – *Message Handling Systems (MHS): Message transfer system: Abstract service definition and procedures*.
- ITU-T Recommendation X.413 (1999) | ISO/IEC 10021-5:1999, Information technology – *Message Handling Systems (MHS): Message store – Abstract service definition*.
- ITU-T Recommendation X.419 (1999) | ISO/IEC 10021-6:2003, Information technology – *Message Handling Systems (MHS): Protocol specifications*.
- ITU-T Recommendation X.420 (1999) | ISO/IEC 10021-7:2003, Information technology – *Message Handling Systems (MHS): Interpersonal messaging system*.
- ITU-T Recommendation X.500 (1997) | ISO/IEC 9594-1:1998, Information technology – *Open Systems Interconnection – The Directory: Overview of concepts, models and services*.
- ITU-T Recommendation X.501 (1997) | ISO/IEC 9594-2:1998, Information technology – *Open Systems Interconnection – The Directory: Models*.
- ITU-T Recommendation X.509 (1997) | ISO/IEC 9594-8:1998, Information technology – *Open Systems Interconnection – The Directory: Authentication Framework*.
- ITU-T Recommendation X.511 (1997) | ISO/IEC 9594-3:1998, Information technology – *Open systems Interconnection – The Directory: Abstract service definition*.
- ITU-T Recommendation X.518 (1997) | ISO/IEC 9594-4:1998, Information technology – *Open Systems Interconnection – The Directory: Procedures for distributed operation*.

- ITU-T Recommendation X.519 (1997) | ISO/IEC 9594-5:1998, *Information technology – Open systems Interconnection – The Directory: Protocol specifications.*
- ITU-T Recommendation X.520 (1997) | ISO/IEC 9594-6:1998, *Information technology – Open Systems Interconnection – The Directory: Selected attribute types.*
- ITU-T Recommendation X.521 (1997) | ISO/IEC 9594-7:1998, *Information technology – Open systems Interconnection – The Directory: Selected object classes.*

2.2 Paired Recommendations | International Standards equivalent in technical content

- CCITT Recommendation X.208 (1988), *Specification of Abstract Syntax Notation One (ASN.1).*
ISO/IEC 8824:1990, *Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1).*
- CCITT Recommendation X.209 (1988), *Specification of basic encoding rules for Abstract Syntax Notation One (ASN.1)*
ISO/IEC 8825:1990, *Information technology – Open Systems Interconnection – Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1).*
- ITU-T Recommendation F.400/X.400 (1999), *Message handling system and service overview.*
ISO/IEC 10021-1:1999, *Information technology – Text communication – Message Handling Systems (MHS) – Part 1: System and Service Overview.*
- CCITT Recommendation X.407 (1988), *Message handling systems: Abstract service definition Conventions.*
ISO/IEC 10021-3:1990, *Information technology – Text Communication – Message-oriented Text Interchange Systems (MOTIS) – Part 3: Abstract service definition Conventions.*

2.3 Additional references

- CCITT Recommendation F.440 (1992), *The voice messaging service.*
- CCITT Recommendation G.721 (1988), *32 kbit/s Adaptive Differential Pulse Coding Modulation (ADPCM).*
- CCITT Recommendation G.726 (1990), *40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM).*
- CCITT Recommendation G.728 (1992), *Coding of speech at 16 kbit/s using low-delay code excited linear prediction.*
- CCITT Recommendation X.403 (1988), *Message Handling Systems: Conformance Testing.*
- CCITT Recommendation X.408 (1988), *Message handling systems: Encoded information type conversion rules.*

3 Definitions

For the purposes of this Recommendation, the following definitions apply.

3.1 Common definitions for MHS

This Recommendation uses terms defined in ITU-T Rec. F.400/X.400 and ISO/IEC 10021-1, ITU T Rec. X.402 | ISO/IEC 10021-2 and ITU-T Rec. X.413 | ISO/IEC 10021-5.

- access unit;
- body;
- content;
- distribution list;
- encoded information types;
- envelope;
- message handling system;
- message-oriented text interchange system;
- message store;
- message transfer agent;
- message transfer system;

- m) recipient;
- n) submission identifier;
- o) submission time;
- p) synopsis;
- q) telematic agent;
- r) user;
- s) user agent.

3.2 Common definitions for Abstract Syntax Notation One

This Recommendation uses the full extent of the Abstract Syntax Notation One (ASN.1) as defined in CCITT Rec. X.208 and ISO/IEC 8824.

3.3 Voice service definitions

This Recommendation uses terms defined in CCITT Rec. F.440.

- a) VM forwarding;
- b) Voice message;
- c) Voice messaging;
- d) Voice messaging notification;
- e) Voice messaging user;
- f) Telephone service access unit.

3.4 Voice Messaging System Definitions

For the purposes of this Recommendation, the following definitions apply:

3.4.1 voice messaging system message store: A Voice Messaging System Message Store is a specialized message store for the purposes of voice messaging

3.4.2 voice messaging system: The Voice Messaging System is the functional object by means of which all users communicate with one another in voice messaging.

3.4.3 voice messaging system user agent: A Voice Messaging System User Agent is a specialized user agent for the purposes of voice messaging.

4 Abbreviations

This Recommendation uses the following abbreviations:

ADPCM	Adaptive differential pulse code modulation
AU	Access Unit
ASN.1	Abstract Syntax Notation One
DL	Distribution List
EIT	Encoded Information Types
IA5	International Alphabet No. 5
IPM	Interpersonal messaging
MD	Management Domain
MHS	Message Handling System
MOTIS	Message-oriented Text Interchange System
MS	Message Store
MTA	Message Transfer Agent
MTS	Message Transfer System
NRN	Non-receipt (Forwarded) Notification

PICS	Protocol Implementation Conformance Statement
RN	Receipt Notification
SN	Service Notification
TSAU	Telephone Service Access Unit
UA	User Agent
UTC	Coordinated Universal Time
VM	Voice Message
VMG	Voice Messaging
VMGE	Voice Messaging Environment
VMGS	Voice Messaging System
VMGS-MS	Voice Messaging System Message Store
VMGS-UA	Voice Messaging System User Agent
VMGS-user	Voice Messaging System user
VN	Voice Notification

5 Conventions

5.1 Terms

Throughout the rest of this Recommendation, terms that refer to ASN.1 types are written with upper-case letters for all words that actually appear in the ASN.1 type (for example, OR Name).

- Voice Messaging System User:** The voice messaging system user (VMGS-user) is normally a person, not an application or computer process. A VMGS-user may be a direct user or indirect user. In MHS, direct users interact with the MHS through a User Agent or Message Store while indirect users interact with the MHS an Access Unit. For brevity, the term user is used throughout the rest of this Recommendation with the meaning of VMGS-user.
- Subject Voice Message:** The term *subject Voice Message*, abbreviated as **subject VM** refers to a specific VMG information object, a VM as defined in clause 8, that is either being held, processed or forwarded by a Voice Messaging application entity. This is not to be confused with the Subject VM field found in a VM notification, see 9.1.1.
- VM Forwarding:** The term identifies the VMG message forwarding behaviour of VMGS-UAs and VMGS-MSs. It covers both manual and automatic forwarding aspects. This term is further refined in 17.3.3.

5.2 ASN.1

ASN.1 definitions appear both in the main text and in the annexes. In case of inconsistency between a definition presented in the text, and a definition presented in an annex forming an integral part of this Recommendation, the definition in the annex shall be used. ASN.1 notation is defined in CCITT Rec. X.208 and ISO/IEC 8824.

This Recommendation uses ASN.1-based descriptive conventions:

- to define the information objects of Voice Messaging, and other data types and values of all kinds, and the ASN.1 itself;
- to define the functional objects of Voice Messaging, the OBJECT and REFINE macros of CCITT Rec. X.407 and ISO/IEC 10021-3;
- to define the abstract service of Voice Messaging, the PORT and ABSTRACT-operation and ERROR macros of CCITT Rec. X.407 and ISO/IEC 10021-3;
- to define the protocol extensions, the VM-EXTENSION macro of this Recommendation;
- to define extended body part types, the EXTENDED-BODY-PART-TYPE macro of ITU-T Rec. X.420 | ISO/IEC 10021-7;
- to define MS Auto-actions, the AUTO-ACTION macro of ITU-T Rec. X.413 | ISO/IEC 10021-5;
- to define MS attributes, the ATTRIBUTE macro of ITU-T Rec. X.413 | ISO/IEC 10021-5.

ASN.1 tags are implicit throughout the ASN.1 modules defined in any annex; the module is definitive in that respect.

NOTE – The use of ASN.1 to describe a class or piece of information does not in itself imply that information is transported between open systems. The fact that the information, by virtue of its description in ASN.1 and of ASN.1's basic encoding rules, has a concrete

transfer syntax may be immaterial. Information actually conveyed between systems is designated as such by its inclusion in an application protocol.

5.2.1 Conventions for attribute types in Table 1

This Recommendation uses the conventions listed below in its definition of attribute types for the MS abstract services.

For the column headed *Single-/Multi-valued*, the following values can occur:

- S: Single-valued;
- M: Multi-valued.

For the column headed *Support level by MS and UA* (where UA refers only to a UA that accesses an MS), the following values can occur:

- M: Mandatory;
- O: Optional.

For the columns headed *Presence in delivered VM*, *Presence in delivered RN*, *Presence in delivered NRN* and *Presence in delivered SN*, the presence of each attribute type is described by one of the following values:

- P: *Always present* in the entry because it is mandatory for generation by the MS or it is a mandatory or defaulted parameter in the relevant abstract operation.
- C: *Conditionally present* in the entry. It will be present because it is supported by the MS and subscribed to by the user and it was present in an optional parameter in the relevant abstract operation.
- *Always absent*, otherwise.

For the columns headed *Available for list, alert* and *Available for summarize*, the following values can occur:

- N: no
- Y: yes

NOTE – All attributes are available to the fetch abstract operation, subject to support by the implementation and subscription.

5.2.2 Conventions for attribute types in Table 2

This Recommendation uses the conventions listed below in its definition of attribute types for the MS abstract services.

For the column headed *Generated by* the following values can occur:

- MD: MessageDelivery abstract-operation;
- MS: MessageStore;
- RD: ReportDelivery abstract-operation.

6 Information objects

The information objects that users exchange in voice messaging are of two kinds: voice messages (VM), and voice notifications (VN). Specific details of the information objects used in the Voice Messaging System are defined in this clause and in clauses 7, 8, and 9.

```
InformationObject ::= CHOICE {  
    vm  
    vn  
    [0] VM,  
    [1] VN }
```

7 Common data types

Information items of several kinds appears both in voice messages and voice notifications. These common items are defined below.

7.1 VM Identifier

A VM Identifier is an information item that unambiguously, globally and forever uniquely identifies a VM.

It comprises an OR Name and a string which may, for example, contain a time or sequence number or other sufficient information to make this VM unique.

```
VMIdentifier ::= SET {  
    user  
    [0] ORName,
```

user-relative-reference [1] LocalReference }

NOTE – OR Name is defined in 8.5.5/X.411 | ISO/IEC 10021-4.

The VM Identifier shares the same value set with the IPM Identifier defined in ITU-T Rec. X.420 | ISO/IEC 10021-7. Therefore a VMGS-UA or VMGS-MS that is capable of handling both IPMs and VMs shall make sure that the Local Reference is unique both for IPMs and VMs.

A VM Identifier has the following components:

- user:** Identifies the user who originates the VM. One of the user's OR Names.
- user-relative-reference:** Unambiguously identifies the VM, distinguishing it from all other VMs that the user who is identified by the User component originates. A Printable String value shall be of a size from zero to a prescribed number of characters (see Annex G). A length of zero is discouraged.

LocalReference ::= PrintableString (SIZE (0..ub-vmg-local-reference))

7.2 OR Descriptor

An OR Descriptor is an information item that is defined in the IPM protocol specifications (7.1.3 of ITU-T Rec. X.420 | ISO/IEC 10021-7). It identifies the user or DL by either its OR Name, free-form-name or telephone number, all of which are text encoded. In VM, it is most likely that the ORDescriptor will identify a user by a telephone number.

7.3 Spoken Name

The Spoken Name parameter carries an audio encoding of an associated name. It may be associated with the VMOriginatorField, RecipientField, or VNOriginatorField. The encoding of the value is per the VoiceEncodingType parameter in a VM or the VNVoiceEncodingParameter in a VN. Its size is constrained in seconds of duration.

SpokenName ::= OCTET STRING

- Encoding is defined by voice-encoding-type
- or vn-voice-encoding-type.
- Maximum 10 seconds.

NOTE – The value is padded to end on an octet boundary.

7.4 Extensions

A mechanism is provided which allows for future extensions to this Recommendation.

ExtensionField ::= SEQUENCE {

type	[0] VM-EXTENSION,
criticality	[1] Criticality DEFAULT FALSE,
value	[2] ANY DEFINED BY type DEFAULT NULL:NULL }

An Extension field can be marked critical (Criticality set to TRUE) or non-critical (Criticality set to FALSE) for acceptance of the message. An extension marked as non-critical may be ignored or discarded, while an extension marked as critical must be known and performed for acceptance of a VM.

Criticality ::= BOOLEAN

As a notation support for future definitions of extensions, a MACRO is defined.

VM-EXTENSION MACRO ::=

BEGIN

TYPE NOTATION ::= **DataType Critical | empty**
VALUE NOTATION ::= **value (VALUE OBJECT IDENTIFIER)**

DataType ::= **type (X) Default**
Default ::= **"DEFAULT" value (X) | empty**
Critical ::= **"CRITICAL" | empty**

END -- of extension

8 Voice message

A Voice Message (VM) is a member of the primary class of information objects conveyed between users in Voice Messaging.

NOTE 1 – The term *message* when used throughout the rest of this Recommendation is a synonym for Voice Message whose spoken components are encoded per CCITT Rec. G.726 (1990) 32 kbit/s ADPCM.

VM ::= SEQUENCE {
 heading **Heading,**
 body **Body }**

A Voice Message consists of the following components:

- a) **Heading:** A set of Heading Fields (or Fields), each an information item that gives a characteristic of the Voice Message.
- b) **Body:** A sequence of one or more body parts.

Body ::= SEQUENCE {
 primary-body-part **PrimaryBodyPart,**
 additional-body-parts **AdditionalBodyParts OPTIONAL }**

PrimaryBodyPart ::= CHOICE {
 vm-body-part **[0] VBodyPart,**
 forwarded-VM **[1] VMBodyPart }**

AdditionalBodyParts ::= SEQUENCE OF VM-ExternallyDefinedBodyPart

NOTE 2 – VM-Externally Defined Body Part is defined in 8.2.3.

The Body has one Primary Body Part that contains a voice information object. This body part is either the VMGS-user's recorded message itself or a forwarded VM. An example of the encoding of a Voice information object is a 32 kbit/s ADPCM Encoded Message defined by CCITT Rec. G.726 (1990).

NOTE 3 – The scope of a voice information object type is rather large and includes, for example, Privately Defined types.

The following rules comply with the requirements stated in clause 3/F.440:

- c) When a VM is first created, the Primary Body Part shall contain one Voice Body Part.
- d) When a VM is forwarded, its structure shall comply with the rules given in 17.3.3.

The Primary Body Part has one of two basic forms:

- the Primary Body Part contains a voice information object;
- the Primary Body Part contains a forwarded Voice Message.

Additionally, other body parts may be present in a message related to the Primary Body Part but of a different type. Examples of related body parts might be non-voice encoded information, such as drawings/graphics to be used in conjunction with the VM. A VM, generated as a result of VM Forwarding, may contain an additional voice encoded body part for each instance of forwarding.

The structure of a Voice Message is depicted in Figure 1.

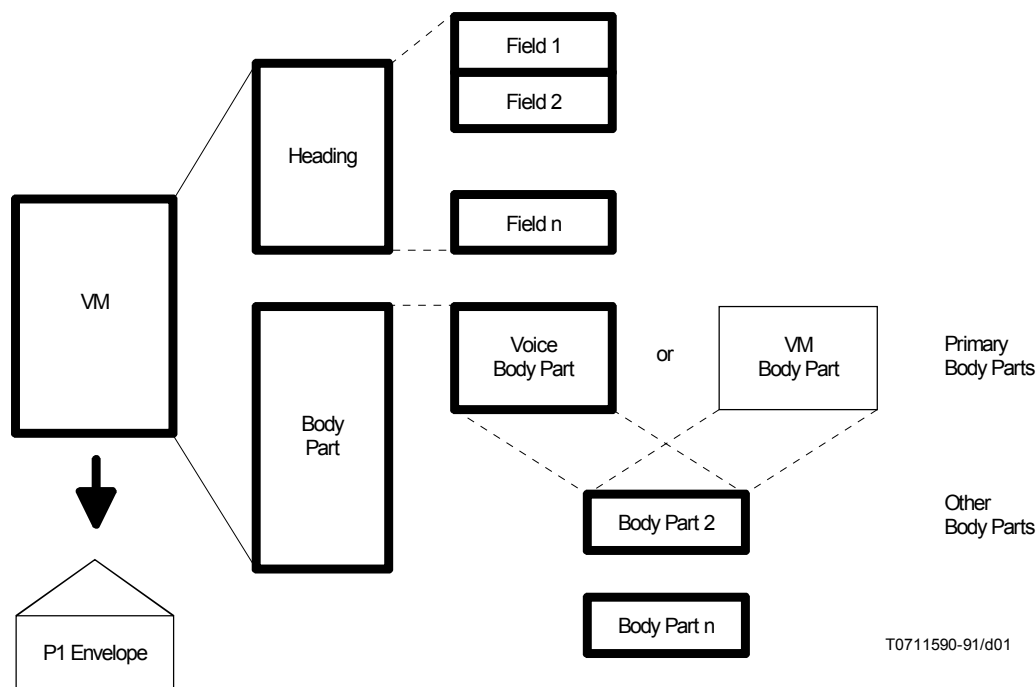


Figure 1/X.440 – Voice message structure

8.1 Heading fields

The fields that may appear in the Heading of a VM are defined and described below.

Heading ::= SEQUENCE {	
this-VM	ThisVMField,
originator	[0] VMOriginatorField OPTIONAL,
recipients	[1] RecipientsField OPTIONAL,
obsoleted-vm	[2] ObsoletedVMField OPTIONAL,
vm-subject	[3] VMSpokenSubjectField OPTIONAL,
expiry-time	[4] ExpiryTimeField OPTIONAL,
importance	[5] ImportanceField DEFAULT {normal}, -- from IPM
sensitivity	[6] SensitivityField OPTIONAL, -- from IPM
message-forwarded	[7] MessageForwarded DEFAULT FALSE,
vm-forwarding-permitted	[8] MessageForwardingPermitted DEFAULT TRUE,
language	[9] LanguageField OPTIONAL,
voice-encoding-type	[10] VoiceEncodingType DEFAULT {id-vmg-g721-32k-adpcm},
vm-creation-time	[11] VMCreationTime OPTIONAL,
vn-receiver-field	[12] VNReceiverField OPTIONAL,
vmgs-user-security-elements	[13] VMGSUserSecurityElementsField OPTIONAL,
heading-extensions	[14] HeadingExtensionsField OPTIONAL }

8.1.1 This VM

The field, This VM, identifies the VM. It comprises a VM Identifier which provides a globally and forever unique identification for the VM.

ThisVMField ::= VMIdentifier

NOTE – VM Identifier is defined in 7.1.

8.1.2 Originator

Identifies the VM's originator. It may comprise an OR Descriptor and optionally a voice encoded version of that name. If the Originator field is not present in the VM Heading, then the P1 envelope shall be used to determine the originator of the VM (see 8.2.1.1.1 of ITU-T Rec. X.411 | ISO/IEC 10021-4).

VMOriginatorField ::= SET {
COMPONENTS OF ORDescriptor, -- from IPM
originator-spoken-name [3] SpokenName OPTIONAL }

NOTE – OR Descriptor is defined in 7.2 and Spoken Name is defined in 7.3.

8.1.3 Recipients

The Recipients field identifies the user(s) and (DL)(s) who are the preferred recipient(s) of the VM. It comprises a set of Recipients subfields, one for each recipient. If the Recipients field is not present in the VM Heading, then the P1 envelope shall be used to determine the recipient of the VM (see 8.2.1.1.2 of ITU-T Rec. X.411 | ISO/IEC 10021-4).

NOTE – The fact that a message can be redirected or forwarded is reflected in the word "preferred" above.

RecipientsField ::= SET OF RecipientsSubField (SIZE (1..ub-recipients))

The Recipients Sub Field is an information item that identifies a recipient of a VM and any special requests made of that specific recipient.

The Recipients Sub Field has the following components:

RecipientsSubField ::= SEQUENCE {
recipient [0] RecipientField,
vn-requests-field [1] VNotificationRequestsField,
attendant-assisted-delivery-request [2] AttendantAssistedDeliveryRequest
DEFAULT { }, -- default NONE
recipient-extensions [3] RecipientExtensionsField OPTIONAL }

8.1.3.1 Recipient

The heading field Recipient identifies one or more preferred recipients for this VM. For each recipient, it comprises an OR Descriptor, and optionally, a voice encoded version of this recipient's name.

RecipientField ::= SEQUENCE {
recipient-name COMPONENTS OF ORDescriptor, -- from IPM
spoken-name [2] SpokenName OPTIONAL }

NOTE – OR Descriptor is defined in 7.2 and Spoken Name is defined in 7.3.

8.1.3.2 VNotification requests field

The field, VNotification Requests Field, is a compound field. It conveys certain requests of the preferred recipient or recipients denoted by the Recipients field.

NOTE 1 – The fact that a message can be redirected or forwarded is reflected in the word "preferred" above.

This field, represented by the data type *VNotificationRequestsField*, consists of a sequence of three bit strings and one extension mechanism. Of the bit strings, the first selects the type of notification, the second selects which security function should be applied to that notification, and the third may make certain security requests for proof or non-repudiation of reception of this VM by the recipient. Voice Notification Security and Voice Reception Security shall not be requested if Voice Notifications are not requested.

The default semantic of this field is that there are no VM extensions present, and neither VM notification, nor notification security nor reception security services are requested of the recipient.

VNotificationRequestsField ::= SEQUENCE {
vm-notification-requests [0] VNotificationRequests DEFAULT { }, -- default NONE
vm-notification-security-requests [1] VNotificationSecurity DEFAULT { }, -- default NONE
vm-reception-security-requests [2] VMReceptionSecurity DEFAULT { }, -- default NONE
vm-notification-extension-requests [3] VMNotificationExtensions DEFAULT { } -- default NONE

1) The *VM Notification Requests* bit string may assume any of the following values simultaneously. The absence of the Voice Notification Requests bit string implies that no Voice Notification Requests are made.

- a) **receipt-notice:** A notification of acceptance of the message is requested in the circumstances prescribed in clause 9.
- b) **service-notice:** A service notification which includes the subject VM is requested in the circumstances prescribed in clause 9.
- c) **non-receipt-notice:** A non-receipt notification is requested in the circumstances prescribed in clause 9.

VNotificationRequests ::= BIT STRING {
receipt-notice (0),
service-notice (1),
non-receipt-notice (2) } (SIZE (0..ub-bit-options))

- 2) The *Voice Notification Security* bit string may assume any of the following values simultaneously. The absence of the Voice Notification Security bit string implies that no Voice Notification Security requests are made.
- d) **proof:** When submitting the VN to the MTS, content-integrity-check shall be requested in the Message-submission-argument as defined in 8.2.1.1.1.28 of ITU-T Rec. X.411 | ISO/IEC 10021-4.
 - e) **non-repudiation:** When submitting the VN to the MTS, content-integrity-check shall be requested in the Message-submission-argument as defined of 8.2.1.1.1.28 of ITU-T Rec. X.411 | ISO/IEC 10021-4 with a non-repudiable certificate.

**VNotificationSecurity ::= BIT STRING {
proof (0),
non-repudiation (1) } (SIZE (0..ub-bit-options))**

- 3) The *Voice Reception Security* bit string may assume any of the following values simultaneously. The absence of the Voice Reception Security field implies that no Voice Reception Security requests are made.
- f) **proof:** When submitting the VN to the MTS, content-integrity-check (possibly in the message token), or the message-origin-authentication-check (depending on the security policy in force) shall be requested. A notification shall contain the security elements and shall be signed on submission to the MTS, using content-integrity-check (possibly in the message token) or message-origin authentication-check (depending on the security policy in force) in the Message-submission-argument as defined in 8.2.1.1.1.26, 8.2.1.1.1.28 and 8.2.1.1.1.29 of ITU-T Rec. X.411 | ISO/IEC 10021-4.
 - g) **non-repudiation:** When submitting the VN to the MTS, a non-repudiable content-integrity-check (possibly in the message token) or a message-origin-authentication-check (depending on the security policy in force) shall be requested. A notification shall contain the security elements and shall be signed on submission to the MTS, using non-repudiable content-integrity-check (possibly in the message token) or message-origin-authentication-check (depending on the security policy in force) in the Message-submission-argument as defined in 8.2.1.1.1.26, 8.2.1.1.1.28 and 8.2.1.1.1.29 of ITU-T Rec. X.411 | ISO/IEC 10021-4.

**VMReceptionSecurity ::= BIT STRING {
proof (0),
non-repudiation (1) } (SIZE (0..ub-bit-options))**

NOTE 2 – Some security services are available only if the MTA supports secure messaging.

- 4) The *VM Notification Extensions* field contains notification extensions to the VNotificationRequestsField subfield. There are no extensions defined in this Recommendation.

VMNotificationExtensions ::= SET OF VMNotificationExtensionsSubField

VMNotificationExtensionsSubField ::= ExtensionField

8.1.3.3 Attendant-assisted delivery request

The Attendant-Assisted Delivery Request field allows for the originator of the subject VM to request call assistance services of a Telephone Service attendant. The absence of this field means that Attendant-Assisted Delivery is not requested by the originator.

**AttendantAssistedDeliveryRequest ::= INTEGER {
person-to-person (0),
anyone (1) }**

The Attendant-Assisted Delivery Request field may assume the values:

- a) **person-to-person:** Allows the originator to request person-to-person delivery to the recipient. The Telephone Service attendant will use the RecipientSpokenName component of the RecipientField to ask for the intended person.
- b) **anyone:** Allows the originator to request an attendant to assist in the delivery of the message to anyone at the named recipient's OR-address.

8.1.3.4 Recipient extensions

The Recipient Extensions contains extensions to the Recipients subfield.

RecipientExtensionsField ::= SET OF RecipientExtensionsSubField

RecipientExtensionsSubField ::= ExtensionField

There are no extensions defined in this Recommendation. Support requirements for any future extensions shall be in accordance with the corresponding element of service to which the heading extension applies.

8.1.4 Obsoleted VM

The Obsoleted VM field identifies one or more VMs that the present VM obsoletes. It is a sequence of subfields, each a VM Identifier.

ObsoletedVMField ::= SEQUENCE OF ObsoletedVMSubfield

ObsoletedVMSubfield ::= VMIdentifier

8.1.5 VM (Spoken) subject

The VM Subject field indicates the message subject of the VM Primary Body Part. It is encoded per the algorithm identified in the Voice-Encoding-Type field. The field contains a brief audio description of the subject VM. Its size is expressed in seconds.

NOTE 1 – A duration of twenty seconds is the upper limit set in Annex G.

SpokenSubject ::= OCTET STRING

-- *Encoding is defined by voice-encoding-type;*

-- *Maximum 20 seconds.*

NOTE 2 – The value is padded to end on an octet boundary.

NOTE 3 – The algorithm used to encode this value is indicated by the *Voice Encoding Type* field, see 8.1.12.

8.1.6 Expiry time

This field indicates when the originator considers this VM to lose its validity. It comprises a date and time (UTC).

NOTE – Expiry Time field is defined in 7.2.11 of ITU-T Rec. X.420 | ISO/IEC 10021-7.

8.1.7 Importance

This field conveys the originator's perceived importance for this message. It may convey one of the levels of importance: low, normal or high. The absence of this element of protocol implies that the originator considers this message to be of **normal** importance.

NOTE – *Importance* field is defined in 7.2.8 of ITU-T Rec. X.420 | ISO/IEC 10021-7.

8.1.8 Sensitivity

The Sensitivity field conveys the originator's perceived sensitivity that recipients are to consider in effect for this message. It conveys one of 3 levels of sensitivity: personal, private, company-confidential. The absence of this element of protocol implies that this message is not considered to be sensitive.

NOTE – *Sensitivity* field is defined in 7.2.15 of ITU-T Rec. X.420 | ISO/IEC 10021-7.

8.1.9 Message forwarded

The Message Forwarded field is used to indicate whether the message was forwarded. Absence of this field shall be interpreted as the value FALSE, i.e. not forwarded.

MessageForwarded ::= BOOLEAN -- Default False

If this field has the value TRUE, it indicates to a receiving UA that the security elements in the inner envelope of the Primary Body Part need to be examined.

NOTE – Rules regarding the use of this field are contained in 17.3.3.1, 17.3.3.2 and 17.3.3.3.

8.1.10 VMessage forwarding permitted

The VMessage Forwarding Permitted field, a Boolean, indicates that the forwarding of this message is permitted by the originator if this field is set to TRUE. Absence of the field shall be interpreted as the value TRUE.

A recipient of a message with the VM Forwarding Permitted field set to FALSE shall originate VNs as requested, and shall not forward this message.

MessageForwardingPermitted ::= BOOLEAN -- Default TRUE, forwarding is permitted

8.1.11 Language

The Language field indicates the internationally recognized language and optionally the local usage that was used by the originator when creating the message. The absence of this field shall be interpreted as unspecified by the originator.

LanguageField ::= SEQUENCE OF Language -- from IPM

The codes used in the language-code and national-usage fields are defined in ISO 639-2 and ISO 3166, respectively.

NOTE 1 – The syntax of the Language field is imported from Annex H of ITU-T Rec. X.420 | ISO/IEC 10021-7. It is defined to be a 5-character field where the first 2 characters indicates the language-code and the last 2 characters indicates the national-usage. The middle character position is a space character as a separator. Usage of the national-usage section of the field is optional.

NOTE 2 – If the language of the spoken message is English as spoken by a Canadian, then the language-code value would be "en", and the national-usage value would be "CA".

NOTE 3 – The type LanguageField is encoded as a Sequence to allow for future growth of the protocol to handle VMs of multiple voice encoded body parts.

NOTE 4 – For this information to be useful for human usage, the receiving system may need to present a spoken version of this information to the user. For example, the value "CA" would be heard in the local language as "canadian" or "Canada".

8.1.12 Voice encoding type

Indicates the voice standard used to encode the contents of the Primary Body Part and any other voice encoded data type contained in the subject VM. It is represented by a single object identifier.

NOTE 1 – This object identifier implements the element of service *VM Encoding Algorithm Identifier (E.10)* defined in Recommendation F.440 (1992).

VoiceEncodingType ::= OBJECT IDENTIFIER -- default CCITT's G.726 32 kbit/s ADPCM

The following standard values have object identifiers defined in this Recommendation:

- 32 kbit/s-ADPCM: ITU-T Rec. G.726 (1990);
- PRIVATE: UNDEFINED OCTETS;
- UNDEFINED: UNDEFINED OCTETS.

The absence of this field shall be interpreted as having the default value set to 32 kbit/s-ADPCM.

The actual mapping of 32 kbit/s-ADPCM transmission encoding onto ASN.1 octets is for further study.

NOTE 2 – The voice encoding referred to by the object identifier is that in which both the Voice Body Part, and those Heading fields whose values are spoken are encoded, notwithstanding the fact that these types are defined as an ASN.1 ANY.

The value of the Voice Body Part Type field shall be used in the Encoded Information Types in the MTS abstract operations (see also 17.1.1). This enables a UA to signal to the MTS which Voice standard the VM's Primary Body Part complies with. The MTS shall make use of this information, if the recipient UA has registered delivery restrictions on Encoded Information Types, to decide if it can deliver the VM or to provide conversion services.

8.1.13 VM creation time

Indicates the date and time of creation of the spoken message. It comprises a UTC time.

VMCreationTime ::= UTCTime

8.1.14 VN receiver

Identifies the recipient to whom VNs are to be sent. This is created by the originator of the VM when the Recipient of a requested notification is different from the Originator of the message. It consists of a sequence of OR Name, VM Identifier and First Recipient.

This field shall not be present if Voice Notification Requests are not made.

This field shall be present in a forwarded message when the forwarding Voice messaging system user agent (VMGS-UA) or Voice messaging system message store (VMGS-MS) forwards the message. This field may be present when the forwarding VMGS-UA accepts the message. Rules related to the construction of this field are given in 17.3.3.

NOTE 1 – For brevity, the term user agent (UA) is used throughout the rest of this Recommendation with the meaning of VMGS-UA, and the term message store (MS) is used throughout the rest of this Recommendation with the meaning of VMGS-MS.

VNReceiverField ::= SEQUENCE {
vn-receiver-name [0] ORName,
original-vm-identifier [1] VMIdentifier OPTIONAL,
first-recipient [2] FirstRecipientField OPTIONAL }

The *first-recipient* field shall not be present if more than one recipient contains Voice Notification Requests.

The *original-vm-identifier* and the *first-recipient* fields shall not be present when the Primary Body Part is a Voice Body Part (that is, when the original originator first creates the VM).

NOTE 2 – The Original VM Identifier and First Recipient fields are included in order to allow the recipient to construct the VN for a forwarded VM. See 9.1 (more specifically 9.1.3) and 17.3.1.1 for rules related to the construction of a VN; see 17.3.3.4 for rules related to the First Recipient field when constructing a forwarded VM. OR Name is defined in 8.5.5 of ITU-T Rec. X.411 | ISO/IEC 10021-4. First Recipient Field is defined in 9.1.3.

8.1.15 VMGS-user security elements

The *VMGS-user security elements* field allows a user to exchange security elements having an end-to-end significance.

```
VMGSUserSecurityElementsField ::= SEQUENCE {
    vmgs-user-security-element      [0] VMGSUserSecurityElement OPTIONAL,
    vm-encrypted-primary-bodypart  [1] BOOLEAN OPTIONAL,
    vmgs-user-security-extensions  [2] VMGSUserSecurityExtensions OPTIONAL }

VMGSUserSecurityElement          ::= BIT STRING (SIZE (0..ub-vmgs-user-security-elements))

VMGSUserSecurityExtensions       ::= SEQUENCE OF VMGSUserSecurityExtension

VMGSUserSecurityExtension        ::= ExtensionField
```

8.1.16 Heading extensions

The Heading extensions allows for future extensions to the Heading.

```
HeadingExtensionsField ::= SET OF HeadingExtensionsSubField

HeadingExtensionsSubField ::= ExtensionField
```

There is no Extensions to the Heading defined in this Recommendation. Support requirements for any future extensions shall be in accordance with the corresponding element of service that the heading extension applies.

8.2 Body part types

The types of body parts that may appear in the Body of a VM are defined and described below.

8.2.1 Voice body part

A Voice Body Part carries a single voice encoded message (object).

```
VBodyPart ::= SEQUENCE {
    voice-parameters      [0] VoiceParameters OPTIONAL,
    voice-data             [1] VoiceData }

VoiceParameters ::= SEQUENCE {
    voice-message-duration [0] VMDuration OPTIONAL,
    voice-encoding-type    [1] VoiceEncodingType OPTIONAL, -- for use in IPM
    other-parameters       [2] VMSupplementaryInformation OPTIONAL,
    extension-parameters   [3] VBPPParameterExtensions OPTIONAL }

VMDuration ::= INTEGER, -- size indicated in seconds

VBPPParameterExtensionsField ::= SET OF VBPEExtensionsSubField

VBPEExtensionsSubField ::= ExtensionField

VoiceData ::= OCTET STRING -- defined by VoiceEncodingType
```

NOTE 1 – The value is padded to end on an octet boundary.

The *VoiceParameters* component of the **VBodyPart** shall be used as follows:

- voice-message-duration:** The duration of the voice encoded data component in seconds.
- voice-encoding-type:** An object identifier which identifies the voice encoding applied to the data portion of this body part. It is intended for use within IPM where voice encoded body parts are not restricted to be of the same encoding within the body of the message. If used in VM, it shall have the same value as the voice-encoding-type field carried in the Heading.
- other-parameters:** Encoded as VMSupplementaryInformation. It conveys additional information necessary for decoding and processing the voice encoded data component of this body part. Its usage is subject to bilateral or local agreements.
- extension-parameters:** Conveys parameter extensions needed for processing the voice encoded data component of this body part.

NOTE 2 – There are no *extension-parameters* defined in this Recommendation.

The reference definition of the encoded voice object used is that used by G.726 32 kbit/s ADPCM. The object identifier used in the default is defined in Annex A. By employing the use of object identifiers, this Recommendation does not draw a distinction between voice, or audio or music quality encoded objects.

The *VoiceData* component of the **VBodyPart** carries the digital encoding of the voice message. The encoding of the contents of this component is identified in the Voice Encoding Type field, see 9.1.5.

This Recommendation identifies two encoding for voice object. Annex H provides guidelines for mapping these encodings, 32 kbit/s ADPCM and 16 kbit/s LDCLP, onto MHS protocol. It is provided to assist developers in creating a consistent way of mapping between the two environments.

8.2.2 VM body part

A VM Body Part contains a forwarded VM, and optionally, its delivery envelope. The delivery envelope shall be present when MHS security services are employed. When a VM is forwarded, its structure shall comply with the rules given in 17.3.3.2.

```

VMBodyPart ::= SEQUENCE {
    parameters                                [0] MessageParameters OPTIONAL,
    data                                       [1] MessageData }

MessageParameters ::= SET {
    delivery-time                            [0] MessageDeliveryTime OPTIONAL,
    delivery-envelope                        [1] OtherMessageDeliveryFields OPTIONAL,
    -- delivery-time and delivery-envelope shall both be present or both shall be absent.
    other-parameters                        [2] VMSupplementaryInformation OPTIONAL }

MessageData ::= VM

VMSupplementaryInformation ::= IA5String (SIZE (1..ub-supplementary-info-length))

```

NOTE – Primary Body Part is defined in clause 8. Message Delivery Time and Other Message Delivery Fields are defined in 8.3.1.1 of ITU-T Rec. X.411 | ISO/IEC 10021-4.

The VM Supplementary Information field is intended to convey additional pre-registered information when performing VM Forwarding.

The structure of a VM Body Part which results from VM Forwarding is depicted in Figure 2.

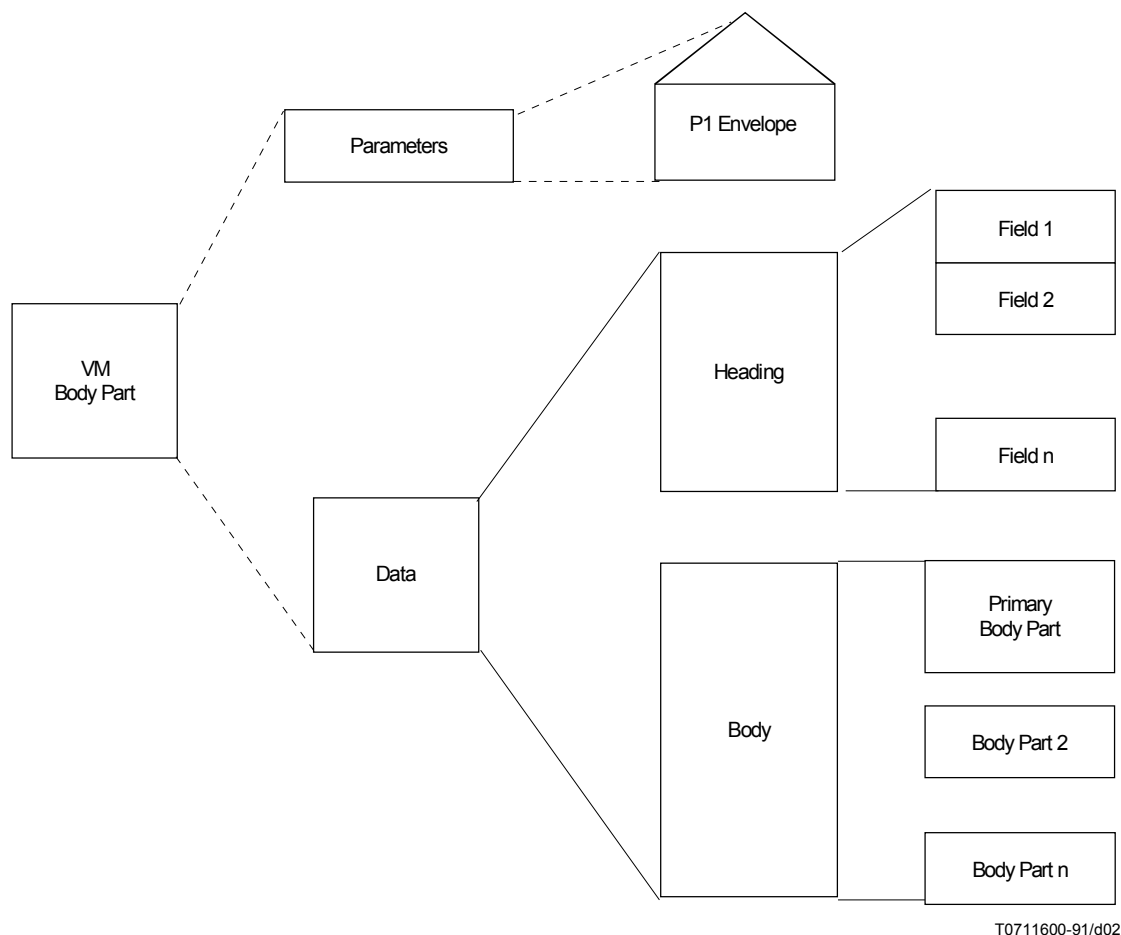


Figure 2/X.440 – VM body (Forwarded) part structure

8.2.3 VM externally defined body parts

Additional body parts, that relate to the Primary Body Part, may be carried together with a Voice Body Part. For submission of the original VM, the body parts shall not be or include voice encoded information objects.

Additional body parts are externally defined and represent information objects whose semantics and abstract syntax are denoted by an object identifier which the body part carries. They have Parameters and Data components.

VM-ExternallyDefinedBodyPart ::= ExternallyDefinedBodyPart -- *from IPMS*

NOTE – The IPMS Externally Defined Body Part type is defined in 7.3.12 of ITU-T Rec. X.420 | ISO/IEC 10021-7.

9 Voice notifications

A Voice Notification (VN) is a member of a secondary class of information object conveyed between users in voice messaging.

NOTE 1 – The term notification is used throughout the rest of this Recommendation as a synonym for voice notification.

NOTE 2 – Voice messaging information objects are defined in clause 8.

VN ::= CHOICE {
 receipt-notification **[0] ReceiptNotificationFields**, -- *referred to as RN*
 service-notification **[1] ServiceNotificationFields**, -- *referred to as SN*
 non-receipt-notification **[2] NonReceiptNotificationFields** -- *referred to as NRN --*
}

- receipt notification (RN): A VN that reports its originator's acceptance of a VM.
- service notification (SN): A VN that reports its originator's software did not support all of the requested EOSs although the message was received by the recipient.
- non-receipt notification (NRN): A VN that reports its originator's refusal to accept a VM and that the subject VM has been forwarded without accepting the message or discarded.

NOTE 3 – RN, SN, and NRN are used in the User Agent section of this Recommendation. However, according to a user's local policy, a UA may delegate to its MS the responsibility for responding to requested VNs.

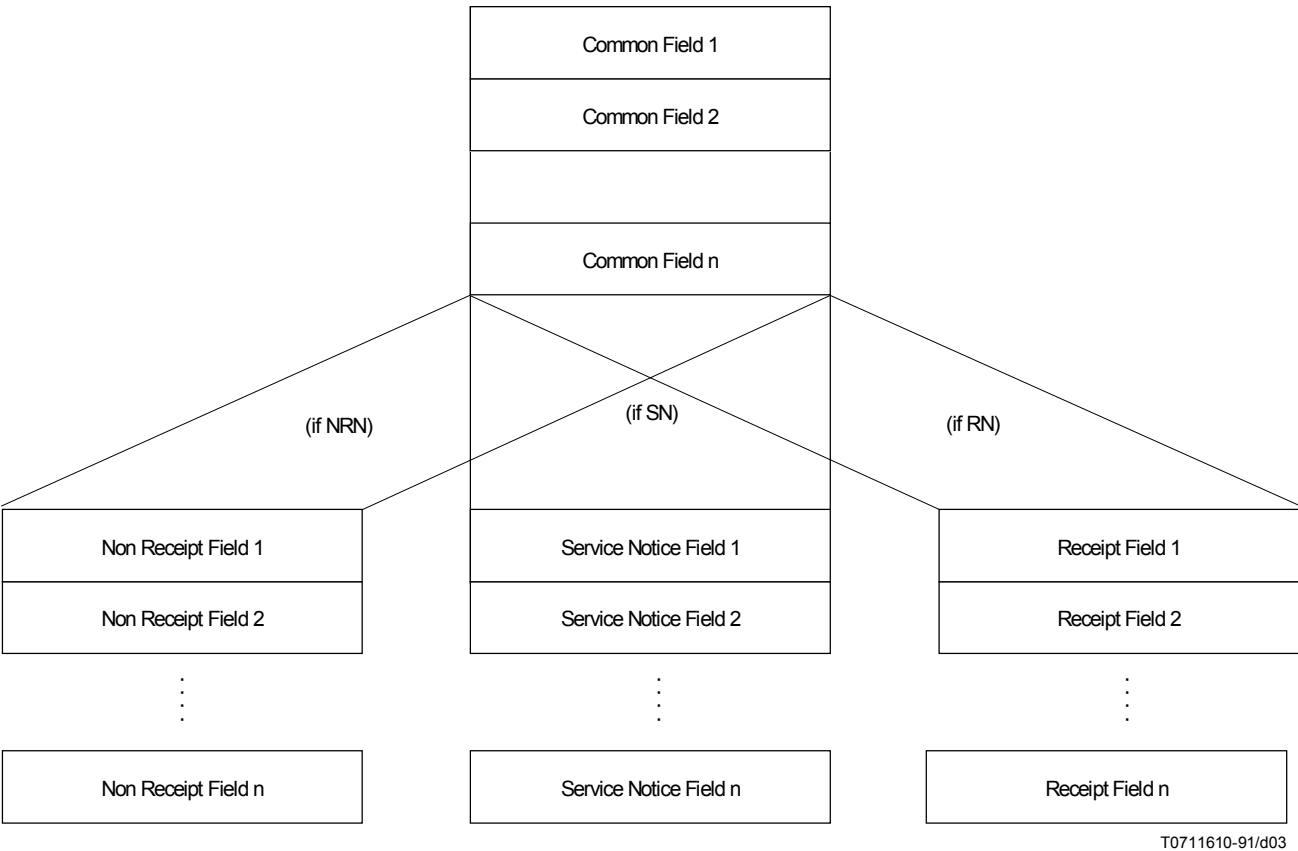
The VM to which a VN refers is called the subject VM (see 17.3.3).

If VM Forwarding has not occurred, then the recipient of the VN is the Originator of the subject VM, or, if present, the OR Name indicated in the VN Receiver field. However, if VM Forwarding has occurred, then the recipient of the VN may be either the originator of the VM or the originator of the forwarded VM.

There shall be at most one recipient specified for a VN. There shall be at most one RN and SN originated for each subject VM by each recipient from whom these notifications are requested. A NRN may be originated subsequent to a SN, in accordance with c of 17.3.3.1. One NRN is originated, if and only if requested, by each recipient that forwards a VM. In accordance with the provision of 17.3.3, the subject VM originator shall receive at most one RN or SN for each recipient from whom these notifications were requested. This is true regardless of how many times the VM is forwarded. However, the originator may receive multiple NRNs as a consequence of VM Forwarding.

A VN consists of fields of information specific to Receipt, Service or Non-Receipt Notification. Additionally, each VN type contains the Common fields which are described below. Figure 3 illustrates the structure of a voice notification.

The specific notification type conveyed by the VNP is also identified in the P1 PerMessageIndicators field as specified in Figure 2 of ITU-T Rec. X.411 | ISO/IEC 10021-4.



T0711610-91/d03

Figure 3/X.440 – Voice notification structure

9.1 Common fields

The Common fields used in the Voice Notifications are defined and described below. The data type which follows is carried in a RN as *rn-common-fields*, in a SN as *sn-common-fields* and in a NRN as *nrn-common-fields*.

CommonFields ::= SET {
 subject-vm
 vn-originator
 first-recipient
 notice-creation-time
 vn-voice-encoding-type
 conversion-indication
 notification-security-elements
 orig-vm-spoken-subject
 subject-vm-other-recipients

SubjectVMField,
 [1] VNOriginatorField,
 [2] FirstRecipientField OPTIONAL,
 [3] NoticeCreationTimeField,
 [4] VNVoiceEncodingType,
 [5] ConversionEITsField OPTIONAL,
 [6] SecurityElementsField OPTIONAL,
 [7] VMSpokenSubjectField OPTIONAL,
 [8] RecipientField OPTIONAL,

vn-supplementary-info
notifications-extensions

[9] VNSupplementaryInfo OPTIONAL,
[10] NotificationExtensionsField OPTIONAL }

NOTE – The Common fields appear in the RN, SN and NRN fields defined below.

9.1.1 Subject VM

The Subject VM field contains the VM Identifier either passed in the VN Receiver field, if the subject VM has been forwarded, or the This VM field, if not.

SubjectVMField ::= VMIdentifier

NOTE – VM Identifier is defined in 7.1. Subject VM is defined in 5.1 b.

9.1.2 VN originator

The VN Originator contains the OR Name component of the Originator field for the UA constructing the notification. Additionally, it may contain a voice encoded representation of this originator and the identity of the VM application entity that initiated the notice.

VNOriginatorField ::= SEQUENCE {
originator-name **[0] VMOriginatorField,**
vn-initiator **[1] VNInitiatorField OPTIONAL }**

NOTE – OR Name is defined in 8.5.5 of ITU-T Rec. X.411 | ISO/IEC 10021-4. The VMOriginatorField parameter is defined in 8.1.2. The audio encoded component of the VMOriginator field, the SpokenName, is defined in 7.3.

The VN Initiator field can take one of the following values:

- a) **internal-UA** means that the UA generated the VN either for local reasons or because the generation had been delegated to it by the user.
- b) **internal-MS** means that the MS generated the VN either for local reasons or because the generation had been delegated to it by the user.
- c) **external-UA** means that the generation of the VN was requested by the user via the abstract operation Originate VN (see 12.1.2).
- d) **internal-tsau** means that the TSAU generated the VN either for local reasons or because the generation had been delegated to it by the user.

VNInitiatorField ::= ENUMERATED {
internal-ua (0),
external-ua (1),
internal-ms (2),
internal-tsau (3) }

Origination of a Receipt Notification implies that the message has been accepted, regardless of the value of this field.

The value of this field shall be consistent with the choice (UA/MS, user) of the Reason Code field for SNs and NRNs.

9.1.3 First recipient

The First Recipient field contains the OR Name of the first recipient in a forwarding chain. This field, together with other fields, is used by the recipient of the notification to correlate the notification and the original message.

FirstRecipientField ::= ORName

NOTE – OR Name is defined in 8.5.5 of ITU-T Rec. X.411 | ISO/IEC 10021-4. If the receiver of this VN desires the audio version of this recipient it is extracted from a locally held copy of the original subject VM.

If the originator of the VN is not the preferred recipient specified by the VBodyPart originator, then the First Recipient field shall be present in the VN (see 17.3 and more specifically 17.3.1.1).

9.1.4 Notice creation time

Notice Creation Time contains the date and time, in UTC format, at which the notification for the subject VM was generated.

NoticeCreationTimeField ::= UTCTime

9.1.5 VN voice encoding type

The VN Voice Encoding Type field conveys the identity of the voice encoding algorithm used to encode spoken data element fields contained in this notification as well as any additional information needed to decode any spoken data elements by the recipient of this VN. If conversion was performed on the subject VM that this VN references, then its value can be different from the *Voice Encoding Type* field found in the subject VM.

VNVoiceEncodingType ::= VoiceEncodingType

NOTE – Its syntax is specified in 8.1.12.

9.1.6 Conversion indication

The Conversion Indication common field identifies the EITs contained in the subject VM upon delivery to the VN's originator. Its value is an EIT descriptor. This field is present only if the subject VM was subjected to MTA conversion services for delivery to the VN originator.

The encoding of this field is found in 8.1.4 of ITU-T Rec. X.420 | ISO/IEC 10021-7, and more specifically in 19.4 below.

9.1.7 Notification security elements

The Notification Security Elements field is used to provide "proof/non repudiation of content received", and "Voice security" services.

```
SecurityElementsField ::= SEQUENCE {  
  original-content [0] Content OPTIONAL, -- from PI  
  original-content-integrity-check [1] ContentIntegrityCheck OPTIONAL, -- from PI  
  vmgs-user-security-elements [2] VMGSUserSecurityElementsField OPTIONAL,  
  security-extensions [3] SecurityExtensionsField OPTIONAL }
```

SecurityExtensionsField ::= SET OF SecurityExtensionsSubField

SecurityExtensionsSubField ::= ExtensionField

NOTE – The *VMGS-user Security Elements* field is defined in 8.1.15. Content and Content Integrity Check are defined in, respectively, 8.2.1.1.1.37 and 8.2.1.1.1.28 of ITU-T Rec. X.411 | ISO/IEC 10021-4. Security services are available only if the MTA supports secure messaging.

Subclause 17.1.2 below specifies how these fields are filled in.

9.1.8 Original VM spoken subject

The Original VM Spoken Subject field conveys the value of the *VM Subject* field received in the subject VM. The subject VM is the cause of this VN being generated. If the *VM Subject* field is not provided in the subject VM, then this notification field shall not have a value.

NOTE – *VM Subject* field which contains voice encoded information is defined in 8.1.5.

9.1.9 Subject VM other recipients

The *Subject VM Other Recipients* conveys the identities of the other intended recipients of the subject VM that prompted this notification. If the subject VM was originally addressed to only one recipient and VM Forwarding was not performed, then this field shall be omitted. If the original subject VM had undergone VM Forwarding, then the union of all recipients shall be present in the value of this field.

SubjectVMOtherRecipients ::= SEQUENCE OF RecipientField

NOTE – Recipient Field is defined in 8.1.3.1.

9.1.10 VN supplementary information

The VN Supplementary Information field carried in a notification, if present, may be encoded in either IA5 or voice. When present, the duration of (audio encoded) *Spoken Supplementary Information* is not to exceed the upper bounds set as follows.

When VN Supplementary Information contains voice encoded information voice, the encoding is that of the original subject VM. If subject VM had been subject to conversion prior to delivery then the *SpokenSupplementaryInfo* field shall be absent.

```
VNSupplementaryInfo ::= SEQUENCE {  
  supplementary-info [0] VMSupplementaryInformation OPTIONAL,  
  v-supplementary-info [1] SpokenSupplementaryInfo OPTIONAL }
```

SpokenSupplementaryInfo ::= OCTET STRING

-- Encoding is defined by vn-voice-encoding-type;
-- Maximum 20 seconds.

NOTE 1 – The value is padded to end on an octet boundary.

NOTE 2 – *VM Supplementary Information* field is defined in 8.2.2.

9.1.11 Notifications extensions

The Notification Extensions allows for future extensions to the VN.

NotificationExtensionsField ::= SET OF NotificationExtensionsSubField

NotificationExtensionsSubField ::= ExtensionField

There are no extensions to the VN defined in this Recommendation. Support requirements for any future extensions shall be in accordance with the corresponding element of service to which the heading extension applies.

Extensions shall not be critical in VNs.

9.2 Receipt notifications

A Receipt Notification (RN) indicates that the subject VM has been accepted by a recipient. It is generated, if, and only if, the originator has requested one and the subject VM has been accepted by the user.

The exact procedures which constitute acceptance of message are a local matter; for example, the UA may construct the RN as soon as it passes the message to the user or it may wait for an external stimulus from the user that the message has been accepted and therefore can generate the RN.

An RN may be generated as a result of VM Forwarding. RNs generated as a result of VM Forwarding is described in 17.3.3 and in 18.4.2 when a VMGS-MS is involved.

An RN is composed of Receipt Notification Fields which consist of VN common fields, and may consist of supplementary information, receipt extension fields and the subject VM.

The structure of an RN is defined as follows:

ReceiptNotificationFields ::= SEQUENCE {
 rn-common-fields **[0] CommonFields,**
 rn-extensions **[1] NotificationExtensionsField OPTIONAL }**

9.2.1 RN common fields

The RN Common Fields carried in this notification are those fields relevant to Receipt notices as specified in 17.3.1.1. The structure of this data type is found in 9.1.

9.2.2 Receipt notification extensions

The Receipt Notification Extensions allows for future extensions to the RN.

There are no extensions to the RN defined in this Recommendation. Support requirements for any future extensions shall be in accordance with the corresponding element of service to which the heading extension applies.

Extensions shall not be critical in RNs.

NOTE – The structure of NotificationExtensionsField is defined in 9.1.11.

9.3 Service notifications

A Service Notification (SN) is sent by a UA, if and only if the originator has requested one. It conveys the semantic that the VMGS-UA or VMGS-MS has been presented the subject VM although the subject VM contained *Optional Additional* elements of service that were not available to the recipient.

NOTE – This notice is not supported in IPM.

Service Notification fields are defined and described below.

ServiceNotificationFields ::= SEQUENCE {
 sn-common-fields **[0] CommonFields,**
 sn-reason-code-field **[1] SNReasonCode,**
 sn-extensions **[2] NotificationExtensionsField OPTIONAL }**

9.3.1 SN common fields

The SN Common Fields carried in this notification are those fields relevant to a Service Notification as specified in 17.3.2.1. The structure of this data type is found in 9.1.

9.3.2 Service notification reason

Generally, the Service Notification Reason indicates that the subject VM was passed to the VMGS-UA or VMGS-MS generating this notice and that one or more requested elements of service are not available to that user. If the SN Reason

Field has the value *"unspecified"*, additional information may be carried in any combination of a Diagnostic field or the VN Supplementary Info field. Depending on the security policy in force, the security error diagnostic code may or may not be present. Any SN Reason Code value set to **TRUE** indicates that this requested optional element of service is not available to the VMGS-user. The receiving VMGS-UA or VMGS-MS may be authorized to initiate a SN on behalf of the receiving user.

SNReasonCode ::= SEQUENCE {
sn-reason [0] SNReasonField,
sn-diagnostic [1] SNDiagnosticField OPTIONAL }

-- Service Notification Basic Reason Codes from a VMGS-UA or VMGS-MS or VMGS-TSAU. These codes are those specified in Annex B/F.440 for the Element of Service "VM Notification Request".

SNReasonField ::= BIT STRING {
unspecified (0),
auto-forwarding-ind (1),
language-ind (2),
obsoleting-ind (3),
attendant-assisted-delivery-request (4),
expiry-date-ind (5),
body-part-encryption-ind (6) } (SIZE (1..ub-sn-reasons))

-- Service Notification Diagnostic Codes from a VMGS-UA or VMGS-MS

SNDiagnosticField ::= INTEGER {
-- This field may be used to further specify the error signalled in sn-ua-ms-basic-code.
-- Additional information may be indicated in sn-supplementary-information.

-- general diagnostic codes

language-national-usage-problem (1),
-- used if the national usage of the language included is incompatible

vm-language-not-understood (2),
-- language not understood by this recipient

vm-unsupported-voice-encoding (3),
-- recipient does not support the VM's encoding

-- security error diagnostic codes
local-security-not-supported (4) } (1..ub-vn-reason-code)

9.3.3 Service notification extensions

The Service Notification Extensions allows for future extensions to the SN.

There are no extensions to the SN defined in this Recommendation. Support requirements for any future extensions shall be in accordance with the corresponding element of service to which the heading extension applies.

Extensions shall not be critical in SNs.

NOTE – SN Extensions, which is defined to be the field NotificationExtensionsField, is defined in 9.1.11.

9.4 Non-receipt notifications

A Non-Receipt Notification (NRN) is sent by a UA, if and only if the originator has requested one, when the UA determines that it cannot accept the message, or decides to forward the VM and the Voice Notification Requests contained in the VM, to another UA.

Non-Receipt Notification fields are defined and described below.

NonReceiptNotificationFields ::= SEQUENCE {
nrn-common-fields [0] CommonFields,
nrn-reason-codes [1] VNRNReasonCodeField,
nrn-extensions [2] NotificationExtensionsField OPTIONAL }

9.4.1 NRN common fields

The NRN Common Fields carried in this notification are those fields relevant to a Non-Receipt Notification as specified in 17.3.3.6. The structure of NRNCommonFields is found in 9.1.

9.4.2 Non-receipt reason codes

A VN Non-Receipt Reason Codes field indicates why the subject VM could not be accepted by the user of the UA originating the VN. If any Basic Code field has the value "unspecified", additional information may be carried in any combination of a Diagnostic field or the VN Supplementary Information field. Depending on the security policy in force, the security error diagnostic may or may not be present.

```
VNRNReasonCodeField ::= CHOICE {  
    nrn-ua-ms-reason-code      [0] VNRNUAMSReasonCodeField,  
    nrn-user-reason-code      [1] VNRNUserReasonCodeField,  
    nrn-tsau-reason-code      [2] VNRNTSAUReasonCodeField }
```

-- NRN Reason Codes from a VMGS-UA or VMGS-MS.

```
VNRNUAMSReasonCodeField ::= SEQUENCE {  
    nrn-ua-ms-basic-codes      [0] VNRNUAMSBasicCodeField,  
    nrn-ua-ms-diagnostics      [1] VNRNUAMSDiagnosticField OPTIONAL }
```

-- VN NRN Basic Reason Codes from a VMGS-UA or VMGS-MS.

```
VNRNUAMSBasicCodeField ::= INTEGER {  
    unspecified (0),  
    auto-forwarded (1),  
    can-not-pass-to-mhs-user (2),  
    delivery-timeout (3),  
    message-discarded (4),  
    subscription-terminated (5),  
    forwarding-error (6),  
    security-error (7),  
    message-forwarded (8),  
    voice-encoding-not-supported (9) } (0..ub-vn-reason-code)
```

-- NRN Diagnostic Codes from a VMGS-UA or VMGS-MS or TSAU.

```
VNRNUAMSDiagnosticField ::= INTEGER {  
    -- This field may be used to further specify the error  
    -- signalled in the field "nrn-ua-ms-basic-codes".  
    -- Additional information may be  
    -- indicated in the "vn-supplementary-info" field.
```

```
    protocol-violation (0) -- this is a place holder  
} (0..ub-vn-reason-code)
```

-- NRN Reason Codes from a VMGS-User.

```
VNRNUserReasonCodeField ::= SEQUENCE {  
    vn-user-basic-codes      [0] VNRNUserBasicCodeField,  
    vn-user-diagnostics      [1] VNRNUserDiagnosticField OPTIONAL }
```

-- VN NRN Basic Reason Codes from a VMGS-User.

```
VNRNUserBasicCodeField ::= INTEGER {  
    unspecified (0),  
    user-defined-reason (1) -- this is a place holder  
} (0..ub-vn-reason-code)
```

-- VN NRN Basic Reason Codes from a VMGS-User.

```
VNRNUserDiagnosticField ::= INTEGER  
-- Contains the reason passed by the user when the value  
-- of "nrn-user-basic-code" is "user-defined-reason".  
-- Additional information may be indicated in the  
-- "vn-supplementary-info" field. The values used in this field  
-- are outside the scope of this Recommendation.
```

-- NRN Reason Codes from a TSAU-User.

```
VNRNTSAUReasonCodeField ::= SEQUENCE {  
    nrn-user-basic-codes      [0] VNRNTSAUBasicCodeField,
```

nrn-user-diagnostics [1] VNRNTSAUDiagnosticField OPTIONAL }

VNRNTSAUReasonBasicCodeField ::= INTEGER {
attendant-assisted-delivery-failure (0),
unknown-telephone-number (1),
attendant-assisted-delivery-not-provided (2),
delivery-timeout (3),
security-error (4),
message-forwarded (5),
unspecified (6),
inappropriate-voice-encoding (7),
telephone-number-unreachable (8),
recipient-refused-message (9),
no-answer-on-every-attempt (10),
busy-on-every-attempt (11),
no-answer-or-busy-on-every-attempt (12),
sensitivity-not-supported (13),
importance-not-supported (14),
busy-on-every-attempt (15)
-- TSAU providers may define additional values above (1000). }

VNRNTSAUDiagnosticField ::= INTEGER {
person-to-person-spoken-name-not-provided (0),
person-unavailable (1),
number-not-in-service (2),
message-expired (3),
importance-not-conveyed (4)
} (0..ub-vn-reason-code)

9.4.3 NRN extensions

The NRN Extensions field allows for future extensions to the structure of an NRN.

There are no extensions to the NRN defined in this Recommendation. Support requirements for any future extensions shall be in accordance with the corresponding element of service to which the heading extension applies.

Extensions shall not be critical in NRNs.

NOTE – NRN Extensions, which is defined to be the field NotificationExtensionsField, is defined in 9.1.11.

10 Primary object types

The environment in which Voice Messaging takes place can be modelled as an abstract object which is hereafter referred to as the Voice Messaging Environment (VMGE).

vmge OBJECT ::= id-ot-vmge

When refined (i.e. functionally decomposed), the VMGE can be seen to comprise lesser objects which interact by means of ports.

vmge-refinement REFINE vmge AS
vmgs
 origination [S] PAIRED WITH vmgs-user
 reception [S] PAIRED WITH vmgs-user
 management [S] PAIRED WITH vmgs-user
 vmgs-user RECURRING
::= id-ref-primary

The lesser objects are referred to as the primary objects of Voice Messaging. They include a single, central object, the Voice Messaging System (VMGS), and numerous peripheral objects called Voice Messaging System users (users). The structure of the VMGE is depicted in Figure 4.

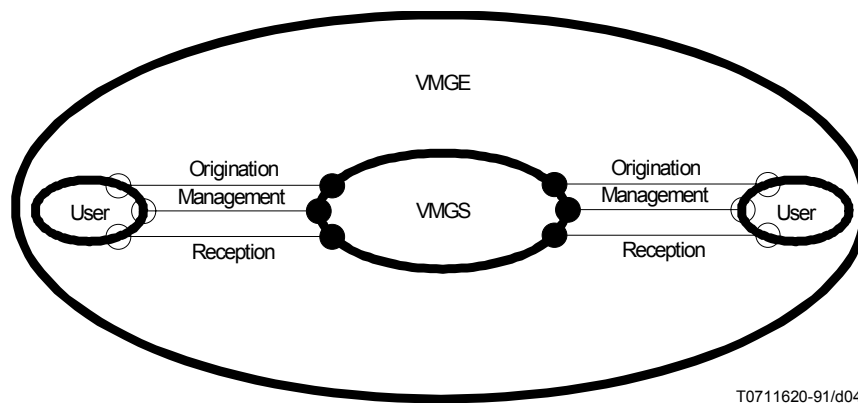


Figure 4/X.440 – The voice messaging environment

The primary object types are defined and described below. The types of ports by means of which they interact are discussed in clause 11.

10.1 Voice messaging system user

A Voice Messaging System user (VMGS-user) is typically a human that engages in Voice Messaging. Such a person is referred to by the term "user" in this Recommendation as identified earlier in 7.1. A user originates, receives, or both originates and receives Information Objects of the types defined in clause 6. Hence, a VMGS-user object is defined as follows:

```
vmgs-user OBJECT
  PORTS {
    origination      [C],
    reception        [C],
    management       [C]
  }
  ::= id-ot-vmgs-user
```

The VMGE comprises any number of users.

NOTE – Voice messaging is typically an activity between two or more humans. The primary access device is anticipated to be a telephone hand set. This does not preclude the possibility of human interaction with the information processing systems which are performing voice messaging adjunct services, or more direct interaction of a human user with the VMGS. These information processes are referred to as Voice applications.

10.2 Voice messaging system

The Voice Messaging System is the object by means of which all users communicate a voice message (VM) with one another in Voice Messaging. Users may choose to use interpersonal messaging systems to exchange voice encoded messages which is described in ITU-T Rec. X.420. As the primary communication object for Voice Messaging, VMGS supplies the services of message origination, reception, and management through *ports*. The VMGS object is defined as follows:

```
vmgs OBJECT
  PORTS {
    origination      [S],
    reception        [S],
    management       [S]
  }
  ::= id-ot-vmgs
```

The VMGE comprises exactly one VMGS.

11 Primary port types

The primary objects of Voice Messaging are joined to and interact with one another by means of ports. These ports, which the VMGS supplies, are referred to as the primary ports of Voice Messaging. They are of the types defined below.

NOTE – In clause 15, the VMGS is decomposed into still lesser objects, among which is the MTS. This fact is anticipated here by the inclusion of certain MTS capabilities in the VMGS Abstract Service.

11.1 Origination port

An Origination Port is the means by which a single user conveys to the VMGS messages containing Information Objects of the types defined in clause 6. Through such a port the user originates a Voice Message and a Voice Notification. A Voice Notification is a Receipt Notification (RN), a Non-Receipt Notification (NRN), and a Service Notification (SN). The abstract operations available at the origination port are defined as:

```
origination PORT
  CONSUMER INVOKES {
    OriginateVM,
    OriginateVN }
  ::= id-pt-origination
```

The VMGS supplies one Origination Port to each user (with the exception of indirect users served by TSAUs (see 15.3).

11.2 Reception port

A Reception Port is the means by which the VMGS conveys to a single user messages containing Information Objects of the types defined in clause 6. Through such a port the user receives Voice Messages and Voice Notifications. In addition, the user may receive Delivery and Non-Delivery reports through such a port. The abstract operations available at the reception port are defined to be:

```
reception PORT
  SUPPLIER INVOKES {
    ReceiveReport,
    ReceiveVM,
    ReceiveVN }
  ::= id-pt-reception
```

The VMGS supplies one Reception Port to each user.

11.3 Management port

A Management Port is the means by which a single user changes personally managed information that is recorded within the VMGS. The specific types of information is defined in clause 6. By means of such a port the user enables and disables auto-discard, -acknowledgement, and -SNs, and -forwarding. The abstract operations available at the management port are defined as:

```
management PORT
  CONSUMER INVOKES {
    ChangeAutoDiscard,
    ChangeAutoAcknowledgements,
    ChangeAutoSNotice,
    ChangeAutoForwarding }
  ::= id-pt-management
```

12 Abstract operations

What follows defines the abstract service that characterizes Voice Messaging, and describes the environment in which that service is supplied and consumed. It does both using the abstract service definition conventions of CCITT Rec. X.407 and ISO/IEC 10021-3.

The VMGS Abstract Service is the set of capabilities that the VMGS provides to each user by means of one Origination Port, one Reception Port and one Management Port. These capabilities are modelled as abstract operations, which may encounter abstract errors when invoked.

The purpose of the VMGS Abstract Service definition is not to prescribe the interface between the VMG-user and the VMGS-UA, but rather to clarify the meaning and intended use of the Information Objects. A user interface need not provide commands in one-to-one correspondence to the service's abstract operations, nor indeed even divide the labour between the user and the VMGS as the service does.

The abstract operations available at the Origination Port and Reception Port are defined and described below. The abstract errors they may provoke are the subject of clause 13.

The VMGS Abstract Service involves neither abstract bind nor abstract unbind operations.

The VMGS authenticates (i.e. establishes the identity of) the typical user before offering the VMGS Abstract Service to him. By this means it can verify, e.g., that the user is a VMG subscriber. Authentication, where required, is implicit (rather than explicit) in the definition of the VMGS Abstract Service.

NOTE – In clause 15, the VMGS is decomposed into objects among which is the MTS. The text here reflects this fact by its inclusion of various MTS-defined information items in the VMGS Abstract Service.

12.1 Origination abstract operations

The abstract operations available at an origination port are invoked by the user and performed by the VMGS.

12.1.1 Originate VM

The Originate VM abstract operation originates a message whose content is a VM.

OriginateVM ::= ABSTRACT-OPERATION

ARGUMENT SET {

envelope

[0] MessageSubmissionEnvelope,

content

[1] VM }

RESULT SET {

submission-identifier

[0] MessageSubmissionIdentifier,

submission-time

[1] MessageSubmissionTime }

ERRORS {

SubscriptionError,

RecipientImproperlySpecified }

This abstract operation has the following arguments:

- a) **Envelope:** A message submission envelope as defined by the MTS Abstract Service. The UA supplies all but the following envelope components, which the user provides:
 - 1) The desired per-message options (i.e. priority, per-message indicators, deferred delivery time, and extensions).
 - 2) The OR Names of the preferred recipients and the per-recipient options (i.e. originator report request, explicit conversion, and extensions) desired for each.
- b) **Content:** The VM being originated. If application to application security services are required, the user shall supply the value for the Voice Application Security Elements field.

This abstract operation has the following results:

- c) **Submission-identifier:** The message submission identifier the MTS assigns to the submission.
- d) **Submission-time:** The date and time the message was directly submitted.

12.1.2 Originate voice notifications

The Originate VN abstract operation originates a message whose content is a VN. A VN may be either a RN, NRN or SN.

OriginateVN ::= ABSTRACT-OPERATION

ARGUMENT SET {

envelope

[0] MessageSubmissionEnvelope,

content

[1] VNType }

RESULT SET {

submission-identifier

[0] MessageSubmissionIdentifier,

submission-time

[1] MessageSubmissionTime }

ERRORS {

SubscriptionError,

RecipientImproperlySpecified }

VNType ::= CHOICE {

[0] RN,

[1] NRN,

[2] SN }

A user may, if notifications are requested, invoke an Originate VN abstract operation to indicate to the UA that it should accept, refuse or forward the subject VM, or generate a SN. The exact type of VN to be generated (RN or NRN or SN) is determined from the VN structure conveyed in the Content argument.

A VN shall be originated only by an actual recipient of the subject VM from whom a VN is requested by means of the Voice Notification Request field of the subject VM's Recipient field.

The user shall not have previously originated a VN in response to the subject VM requested voice notice type, by means of either the present abstract operation or auto-acknowledgement.

A user may delegate the task of generating VNs to the UA. In this case, this abstract operation is not present at the abstract interface between the UA and the user, that is, the operation is not available at the Origination Port. In this case the UA behaves as described in 17.3.

This abstract operation has the following arguments:

- a) **Envelope:** A message submission envelope as defined by the MTS Abstract Service. The UA supplies all but the following envelope components, which the user provides:
 - 1) The desired per-message options (i.e. priority, per-message indicators, and extensions). Implicit conversion shall be prohibited; priority is that of the subject VM.
 - 2) The OR Names of the preferred recipients and the per-recipient options (i.e. explicit conversion and extensions) desired for each. The preferred recipient of the VN is the originator of the subject VM or, if present, the OR Name indicated in the VN Receiver field. Reports shall not be requested.
- b) **Content:** The VN type (RN/NRN/SN) being originated.
 - 1) The user shall supply the value for the VN type being originated as part of the notification structure.
 - 2) If UA-to-UA security services are required, the user shall supply the value for the VM Application Security Elements field and in accordance with the user's security policy.

The VN shall be constructed as described in 17.3.

This abstract operation has the following results:

- c) **Submission-identifier:** The message submission identifier the MTS assigns to the submission.
- d) **Submission-time:** The date and time the message was directly submitted.

12.2 Reception abstract operations

The abstract operations available at a Reception Port are invoked by the VMGS and performed by the user.

NOTE – As abstractly defined, the VMGS provides no storage for received messages because the provisioning of said storage does not impact that user's ability to communicate with other VMGS-users. Thus the provision of storage is a local matter.

12.2.1 Receive Report

The Receive Report abstract operation receives a report.

```

ReceiveReport ::= ABSTRACT-OPERATION
  ARGUMENT SET {
    envelope
    undelivered-object
  }
  RESULT
  ERRORS { }
  [0] ReportDeliveryEnvelope,
  [1] InformationObject OPTIONAL }

```

A report may be received as a result of either:

- a) The invocation of an Originate-VM abstract operation of a VM or by forwarding.
- b) A message whose content was a VN that was originated as a result of a previously received message. The VN could be any of RN, SN or NRN.

This abstract operation has the following arguments:

- c) **Envelope:** A report delivery envelope, whose make-up the MTS Abstract Service defines.
- d) **Undelivered-object:** The content of the message whose status is being reported. A VM or VN.
If the report was provoked by a previous Originate VM abstract operation invocation, the argument shall be present if, and only if, P1 content return was requested.

This abstract operation has no results.

12.2.2 Receive VM

The Receive VM abstract operation receives a message whose content is a VM.

```

ReceiveVM ::= ABSTRACT-OPERATION
  ARGUMENT SET {
    envelope
    content
  }
  RESULT
  ERRORS { }
  [0] MessageDeliveryEnvelope,
  [1] VM }

```

This abstract operation has the following arguments:

- a) **Envelope:** The message's delivery envelope.
- b) **Content:** The VM that is the message's content.

This abstract operation has no results.

When the received VM contains a VM Body Part (that is, when the original VM has been forwarded), it may be necessary to scan several levels of nested Heading fields in order to determine the correct original value for optional Heading fields (see 8.2.2 for the nested structure of a forwarded VM and 17.3.3 for rules related to Heading fields).

12.2.3 Receive VN

The Receive VN abstract operation receives a message whose content is a VN. The VN is provoked by a VM originated with the Originate VM abstract operation. A voice notification may be a RN, NRN or SN.

```
ReceiveVN ::= ABSTRACT-OPERATION
  ARGUMENT SET {
    envelope           [0] MessageDeliveryEnvelope,
    content             [1] VN }
  RESULT
  ERRORS { }
```

This abstract operation has the following arguments:

- a) **Envelope:** The message's delivery envelope.
- b) **Content:** The VN that is the message's content. A voice notification may be a RN, NRN or SN.

This abstract operation has no results.

12.3 Management abstract operations

12.3.1 Change auto-discard

The Change auto-discard abstract operation enables or disables auto-discard, the automatic discard by the Voice Messaging System of expired or obsolete messages delivered to, but not yet received by the user.

```
ChangeAutoDiscard ::= ABSTRACT-OPERATION
  ARGUMENT SET {
    auto-discard-expired-VM [0] BOOLEAN,
    auto-discard-obsolete-VM [1] BOOLEAN }
  RESULT
  ERRORS { }
```

When it auto-discards a message, the Voice Messaging System originates a NRN on the user's behalf if, and only if, one was requested by the originator. This request is indicated by means of the V-notification-requests component of the subject VM's VNotificationRequestsField.

This abstract operation has the following arguments:

- a) **auto-discard-expired-VM:** Whether or not *expired* VMs are to be automatically discarded by the Voice Messaging System. A Boolean.
- b) **auto-discard-obsolete-VM:** Whether or not *obsolete* VMs are to be automatically discarded by the Voice Messaging System. A Boolean.

This abstract operation does not have any results.

12.3.2 Change auto-acknowledgement

The Change auto-acknowledgement abstract operation enables or disables auto-acknowledgement, the automatic origination of RNs by the VMGS-MS on the user's behalf. The automatic origination of RNs occurs on delivery of VMs that request RNs of the user. This request is conveyed by means of the V-notification-requests component of the subject VM's VNotificationRequestsField.

```
ChangeAutoAcknowledgements ::= ABSTRACT-OPERATION
  ARGUMENT SET {
    auto-acknowledge-VM           [0] BOOLEAN,
    auto-acknowledge-suppl-recipient-info [1] VNSupplementaryInfo OPTIONAL }
  RESULT
  ERRORS { SubscriptionError }
```

This abstract operation has the following arguments: