

X/Open Preliminary Specification

EDI Messaging Package

X/Open Company Ltd.



© November 1991, X/Open Company Limited and X.400 API Association

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the copyright owners.

X/Open Preliminary Specification

EDI Messaging Package

ISBN: 1-872630-25-1

X/Open Document Number: P150

Published by X/Open Company Ltd., U.K.

Any comments relating to the material contained in this document may be submitted to X/Open at:

X/Open Company Limited
Apex Plaza
Forbury Road
Reading
Berkshire, RG1 1AX
United Kingdom

or by Electronic Mail to:

XoSpecs@xopen.co.uk

Contents

Chapter 1	Introduction.....	1
1.1	Purpose	1
1.2	Overview	2
1.3	Conformance	3
1.3.1	Vendor Extensions	3
1.4	Terminology	3
1.5	Organisation	4
Chapter 2	Class Hierarchy	5
Chapter 3	Class Definitions	7
3.1	Authorization Information	8
3.2	Content.....	9
3.3	Cross referencing Information	10
3.4	Delivery Envelope	11
3.5	EDI Application Security Elements	12
3.6	EDI Body Part	13
3.7	EDI Extension	14
3.8	EDI Heading	15
3.9	EDI Notification	19
3.10	EDI Notification Requests.....	21
3.11	EDIM.....	23
3.12	EDIM Body Part	24
3.13	EDIM Externally Defined Body Part	25
3.14	EDIM Identifier.....	26
3.15	EDIN Receiver	27
3.16	Externally Defined Body Part.....	28
3.17	FN PDAU Reason Code	29
3.18	FN Reason Code.....	30
3.19	FN UAMS Reason Code	31
3.20	FN User Reason Code	32
3.21	Forwarded Notification	33
3.22	Integrity Check Basis.....	34
3.23	Interchange Sender-Recipient	35
3.24	Negative Notification	36
3.25	NN PDAU Reason Code	37
3.26	NN Reason Code	38
3.27	NN UAMS Reason Code.....	39
3.28	NN User Reason Code.....	41
3.29	Object.....	42
3.30	OR Name	43
3.31	Positive Notification.....	44

3.32	Primary Body Part.....	45
3.33	Reason Code	46
3.34	Recipient Reference	47
3.35	Recipients	48
3.36	Security Elements	50
3.37	Service String Advice	51
3.38	Syntax Identifier	52
Chapter 4	Declaration Summary.....	53
	Glossary	61
	Index.....	63

Preface

This Document

This document is a Preliminary Specification. It defines the object classes that constitute the *EDI Package*. It is intended for use in conjunction with the **X.400** and **XOM** API Specifications (see **Referenced Documents**). Together, these three specifications provide a program interface to the functionality of an EDI messaging system, based on the international EDI messaging standards CCITT Recommendations X.435, F.435 and their ISO counterparts ISO/IEC 10021-8 and 10021-9 (see **Referenced Documents**), often referred to as *Pedi*.

The content of this specification has been developed in collaboration with the X.400 API Association, which is publishing its own version of the specification.

This is one of several specifications that X/Open has developed in collaboration with the X.400 API Association. Other documents include:

- OSI-Abstract-Data Manipulation (XOM) API
- API to Directory Services (XDS)
- API to Electronic Mail (X.400)
- Message Store API
- Guide to Selected X.400 and Directory Services APIs.

Trade Marks

X/Open and the 'X' device are trademarks of X/Open Company Limited in the U.K. and other countries.

Referenced Documents

The following documents are referenced in this document:

MHS

Data Communication Networks: Message Handling Systems, CCITT Recommendations X.400 to X.420

Information Processing Systems - Text Communication - Message Oriented Text Interchange System, ISO/IEC 10021

DS

Data Communication Networks: Directory, CCITT Recommendations X.500 to X.521

Information Processing Systems - Open Systems Interconnection - The Directory, ISO/IEC 9594

X.400

API to Electronic Mail (X.400), CAE Specification, X/Open Company Limited and X.400 API Association, C191, November 1991

XDS

API to Directory Services (XDS), CAE Specification, X/Open Company Limited and X.400 API Association, C190, November 1991

XOM

OSI-Abstract-Data Manipulation (XOM) API, CAE Specification, X/Open Company Limited and X.400 API Association, C180, November 1991

Pedi

Message Handling: EDI Messaging System, CCITT Recommendation X.435 | ISO/IEC 10021-9

Message Handling: EDI Messaging Service, CCITT Recommendation F.435 | ISO/IEC 10021-8

Note: This document has been aligned with Draft Recommendation X.435, Geneva, 1990. The ballot on this text has been approved but the final Recommendation has not yet been published. Any changes to the final Recommendation text will have to be reviewed for potential impact on this specification.

ISO/IEC SC18 has formally agreed to fast track the final Recommendation text as soon as it becomes available. Their stated intention is to approve the Recommendation text unchanged as an International Standard. According to the SC18 Secretariat, using the fast track procedures, ISO/IEC 10021-8 and 10021-9 are scheduled to become International Standards within ten months. References to these ISO documents have been included in this specification and are consistent with the references contained in the CCITT recommendation text.

EDIFACT

Electronic Data Interchange for Administration, Commerce and Transport (EDIFACT), ISO 9735, 1987

UN/TDI

United Nations Trade Data Interchange (UN/TDI), United Nations Economic Commission for Europe, 1985

ANSIX12
American National Standards Institute X12

1.1 Purpose

This specification defines the object classes that constitute the *EDI Package*. It is intended for, but is not limited to, use in conjunction with the **X.400 Specification** and the **XOM Specification** (see **Referenced Documents**). Together, these three specifications provide a program interface to the functionality of an EDI messaging system, based on the international EDI messaging standards CCITT Recommendations X.435, F.435 and their ISO counterparts **ISO/IEC 10021-8** and **10021-9**, often referred to as *Pedi* (see **Referenced Documents**).

Note: This document has been aligned with Draft Recommendation X.435, Geneva, 1990. The ballot on this text has been approved but the final Recommendation has not yet been published. Any changes to the final Recommendation text will have to be reviewed for potential impact on this specification.

ISO/IEC SC18 has formally agreed to fast track the final Recommendation text as soon as it becomes available. Their stated intention is to approve the Recommendation text unchanged as an International Standard. According to the SC18 Secretariat, using the fast track procedures, ISO/IEC 10021-8 and 10021-9 are scheduled to become International Standards within ten months. References to these ISO documents have been included in this specification and are consistent with the references contained in the CCITT recommendation text.

EDI application developers can use the interface to access the services of *Pedi*, while remaining isolated from the underlying complexity involved in providing these services and the communication systems that support them. A typical application that might use this interface would be a computer process responsible for passing EDI information between an EDI information processing system and a *Message Handling System* (MHS) used for transporting the EDI information over a network. Such an application would not require knowledge of how EDI information is managed, encoded, or transported by the MHS. It would only be responsible for submitting and receiving the EDI information at the interface in the form of the objects defined in this specification.

1.2 Overview

The **X.400 Specification** (see **Referenced Documents**) defines two interfaces to the functionality of an MHS based on international messaging standards: the *Message Access Interface* (or *Application API*) and the *Message Transfer Interface* (or *Gateway API*). While they differ from each other in the type of messaging functionality they provide, both interfaces present a model whereby messages, reports, and probes are passed across the interface between the user of the interface, or *client*, and the provider of the interface functionality, or *service*.

Messages, probes, and reports are represented at the interface by data structures called *objects*. The precise definition of objects and the functions available for creating, examining, modifying and destroying them is fully defined in the **XOM Specification** (see **Referenced Documents**).

Briefly, objects are categorised on the basis of their purpose and structure into categories called *classes*. Related classes can be grouped into collections called *packages*. The set of EDI object classes defined in this specification is such a collection of related classes which constitute the *EDI Package*. The service provides and the client uses interface functions and object packages. Using the EDI Package, objects representing EDI information can be passed across the interface between client and service.

Support for the EDI Package by the service is optional, and is negotiated by the client and service when the interface is initiated. A service that supports the EDI Package allows the client to include and extract EDI information in the content of a message in the same way that the *IM Package* defined in the **X.400 Specification** (see **Referenced Documents**) allows the client to include and extract Interpersonal Messaging information in the content of a message.

While the content of a message might contain EDI information or Interpersonal Messaging information, the envelope of the message, containing control information used by the MHS to deliver the message, is common to both content types, and is represented by objects belonging to the Message Handling or *MH Package*, also defined in the **X.400 Specification** (see **Referenced Documents**). Since the control information represented by objects belonging to the MH Package is essential for message transfer through the MHS, support for the MH package by the service is mandatory.

The decision as to whether the EDI Package should be used in conjunction with the Message Access (MA) Interface or the Message Transfer (MT) Interface is dependent on the EDI application in question. Guidance on the appropriateness of these interfaces is given in the **X.400 Specification** (see **Referenced Documents**).

Other API specifications may define other interfaces to the functionality of an MHS. A Message Store API is an example of such an API. The decision as to whether the EDI Package should be used in conjunction with one of these interfaces is dependent on the EDI application in question. For guidance on the appropriateness of an interface for a given application, consult the specification which defines it.

1.3 Conformance

A manufacturer shall claim conformance to this edition of this document only if it and its product collectively satisfy the following requirements:

1. The product shall implement all of the classes in the EDI Package and its closure except the forwarding, security, and physical delivery classes. The product may optionally implement the forwarding, security, and physical delivery classes, in which case the product shall state which of these classes are implemented. The package closure is the set of classes that need to be supported in order to be able to create all possible instances of all classes defined in the package.
2. The product shall state for which classes it provides the *OM-Encode()* function.
3. The product shall support the *OM-Create()* function for all concrete classes.

1.3.1 Vendor Extensions

Vendors may provide additional OM attributes in their implementation of particular OM classes, and their individual documentation will give details of the specification and usage of these. The presence of extensions can be negotiated by use of different packages identified by different object identifiers. All such vendor extension OM attributes will have default values that lead to the behaviour described in this specification.

1.4 Terminology

The objects defined in this document are to be understood in the context provided by the EDI messaging standards (see **Message Handling** references in **Referenced Documents**). The EDI messaging user (EDIMG user) is normally an EDI application or computer process, not a person. For brevity, the term *user* is used throughout the rest of this document with the meaning of EDIMG user. Similarly, the terms *notification* and *message*, when used without qualifiers, refer to EDI notifications (EDIN) and EDI messages (EDIM), respectively. The terms *originator* and *recipient* refer to the roles that various users play in the conveyance of EDI messages and EDI notifications via the message transfer system (MTS).

Throughout this document, care is taken to distinguish between Object Management (OM) classes and attributes, and the attributes of the *Message Store* (MS) or the classes and attributes of the *Directory*. The former is a construct of the Object Management interface, which is used with application-specific APIs to provide access to information objects, including EDI objects (**XOM Specification**, see **Referenced Documents**). The terms “object class” and “attribute” denote the OM constructs, while the phrases “directory class”, “directory attribute” and “MS attribute” denote the constructs defined in the Directory and Messaging standards (see **Data Communication Networks** references in **Referenced Documents**).

Abbreviations

The abbreviations used throughout this Specification are included in the Glossary at the end of this document.

1.5 Organisation

The remainder of this document defines the EDI classes that constitute the EDI Package. **Chapter 2, Class Hierarchy** depicts the hierarchical organisation of the EDI classes. **Chapter 3, Class Definitions** defines each class. The constants that represent the OM classes and OM attributes in the C binding are defined in **Chapter 4, Declaration Summary**.

Class Hierarchy

This section depicts the hierarchical organisation of the EDI classes. Subclassification is indicated by indentation. The names of abstract classes are in *italics*. As an example, **Negative Notification** is an immediate subclass of *EDI Notification*, an abstract class. The service is not required to support the *OM-Encode()* function for any classes in the EDI Package. The *OM-Create()* function must be supported for all concrete classes.

In addition to all classes defined in this specification, the EDI Package closure contains the following classes which are referenced directly in this specification: Content, Delivery Envelope, Externally Defined Body Part, Integrity Check Basis, OR Name (see the **API to Electronic Mail (X.400) CAE Specification**), and Object (see the **API to OSI Object Management (XOM) CAE Specification**). The EDI Package closure also contains the closure of each of these classes (that is, the set of classes needed to be able to create all possible instances of each class). The class hierarchy indicates, as a guide to the reader, those classes in the package closure which are directly referenced in this specification. The definitions of these classes and the classes in their closures can be found in the referenced specifications.

Note: The terms package closure, abstract and concrete class are defined in the **API to OSI Object Management (XOM) CAE Specification**. Readers are referred to the XOM Specification for a complete discussion of the information structuring notions (including class inheritance) used in the following sections.

Object (defined in the Object Management Package)

- Authorization Information
- *Content* (defined in the Message Handling Package)
 - *EDI Notification*
 - Forwarded Notification
 - Negative Notification
 - Positive Notification
 - EDIM
- Cross Referencing Information
- Delivery Envelope (defined in the Message Handling Package)
- EDI Application Security Elements
- EDI Extension
- EDI Heading
- EDI Notification Requests
- EDIM Externally Defined Body Part
- EDIM Identifier
- EDIN Receiver
- Externally Defined Body Part (defined in the Interpersonal Messaging Package)
- Integrity Check Basis (defined in the Secure Messaging Package)
- Interchange Sender-Recipient
- OR Name (defined in the Message Handling Package)
- *Primary Body Part*
 - EDI Body Part
 - EDIM Body Part
- *Reason Code*
 - *FN Reason Code*
 - FN PDAU Reason Code
 - FN UAMS Reason Code

- FN User Reason Code
- *NN Reason Code*
 - NN PDAU Reason Code
 - NN UAMS Reason Code
 - NN User Reason Code
- Recipient Reference
- Recipients
- Security Elements
- Service String Advice
- Syntax Identifier

Class Definitions

This chapter defines the classes of the EDI Package. The object identifier associated with this package is

{joint-iso-ccitt mhs-motis(6) group(6) white(1) api(2) edi(7) edi-pkg(1)}

(with the encoding `\x56\x06\x01\x02\x07\x01`).

This object identifier is represented by the constant **EDI_PACKAGE** *{EDI_PACKAGE}*. Each OM class is described in a separate section, which identifies the OM attributes specific to that OM class. The OM attributes that may be found in an instance of an OM class are those OM attributes specific to that OM class and those inherited from each of its superclasses. The OM attributes specific to an OM class are defined in a table. The table gives the name of each OM attribute, the syntax of each of its values, any restrictions upon the length of each value, any restrictions upon the number of values, and the value, if any, that the *OM-Create()* function supplies.

The **API to OSI Object Management (XOM) CAE Specification** defines the value syntaxes used in the tables. The attribute descriptions are derived directly from the EDI Messaging System standards (see **Referenced Documents**). It is not the intent of this specification to add any additional constraining semantics to the information elements described. In cases of ambiguity or conflict in the semantics of attributes, the EDI messaging standards are to be taken as definitive. This specification provides a mapping of the structure of information objects of the standards into a form suitable for use with APIs and in particular with the Message Access, Message Transfer and Object Management Interfaces (see the **X.400** and **XOM CAE Specifications**).

3.1 Authorization Information

An instance of class **Authorization Information** indicates who authorised the interchange. This class is semantically identical to the "Authorization information" in the ANSIX12 Interchange.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Authorization Information	String(Teletex)	1-10	1	-
Authorization Information Qualifier	String(Teletex)	1-2	0-1	-

Table 3-1. OM Attributes of an Authorization Information

Further details on the meaning of this class and its attributes can be found in the ANSIX12 standard (see **Referenced Documents**).

3.2 Content

The **Content** class is defined in the MH (Message Handling) Package (see **API to Electronic Mail (X.400) CAE Specification**). The EDI Package closure contains this class and its closure.

An instance of class **Content** is the information that a message is intended to convey to its recipients. The purpose of this abstract class is to provide a common superclass for content type-dependent classes defined by different messaging applications. The value of the OM attribute, Content, of the Message class (in the MH package) may be an instance of any concrete subclass of the Content class. Examples of such subclasses include *General Content* (defined in the MH Package), *Interpersonal Message* and *Interpersonal Notification* (defined in the IM Package), *EDIM* and *EDI Notification* (defined in this chapter), and message content type-dependent classes that may be defined in other specifications. The value of the Content Type attribute of the Message class indicates the syntax and semantics of the Content attribute. Similarly, the Content Type attribute of the Probe class (also part of the MH Package) indicates the syntax and semantics of the message content to which the probe refers. The **API to Electronic Mail (X.400) CAE Specification** defines values of the Content Type attribute for use with Interpersonal Messaging. The content type associated with EDI is represented by the constant **EDI_CONTENT_TYPE** {*EDI_CONTENT_TYPE*}. This content type is used with both EDI Notifications and EDIMs.

3.3 Cross referencing Information

An instance of class **Cross Referencing Information** contains a reference to a body part. The referenced body part may be in the same EDIM containing the Cross Referencing Information instance, or within other EDIMs.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Application Cross Reference	String(Octet)	-	1	-
Message Reference	Object(EDIM Identifier)	-	0-1	-
Body Part Reference	Integer	-	1	-

Table 3-2. OM Attributes of a Cross Referencing Information

Application Cross Reference

Application-specific information about the referenced body part.

Message Reference

Indicates the EDIM containing the referenced body part. If this attribute is absent, the message referred to is the EDIM containing the Cross Referencing Information instance.

Body Part Reference

Indicates the referenced body part. See the EDIM Externally Defined Body Part class.

3.4 Delivery Envelope

The **Delivery Envelope** class is defined in the MH (Message Handling) Package (see **API to Electronic Mail (X.400) CAE Specification**). The EDI Package closure contains this class and its closure.

3.5 EDI Application Security Elements

An instance of class **EDI Application Security Elements** identifies a set of security elements. This class is used to allow an EDI application to exchange security elements having an end-to-end significance.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
EDI Application Security Element	String(Bit)	0-8191	0-1	-
EDI Encrypted Primary Bodypart	Boolean	-	0-1	-
EDI Application Security Extensions	Object(EDI Extension)	-	0 or more	-

Table 3-3. OM Attributes of EDI Application Security Elements

EDI Application Security Element

Indicates the Application Security Elements referred to by this instance.

EDI Encrypted Primary Bodypart

Indicates whether the associated primary bodypart is encrypted.

EDI Application Security Extensions

Contains extensions to the EDI Application Security Elements class.

3.6 EDI Body Part

An instance of class **EDI Body Part** carries a single EDI Interchange. The reference definition of an EDI Interchange used here is that of EDIFACT (ISO 9735) (see **Referenced Documents**).

An instance of this class has the attributes of its superclasses - *Primary Body Part* and *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Body	String(Octet)	-	1	-

Table 3-4. OM Attributes of an EDI Body Part

Body Contains an octet string encoding the EDI Interchange.

3.7 EDI Extension

An instance of class **EDI Extension** provides a mechanism for inclusion of future extensions to the EDI messaging standards (see **EDI Messaging Systems**, in **Referenced Documents**) in information classes defined in this specification.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Criticality	Boolean	-	1	False
Extension Type	String(Object Identifier)	-	1	-
Extension Value	<i>any</i>	-	0-1	-

Table 3-5. OM Attributes of an EDI Extension

Criticality

Indicates whether the extension is critical for acceptance of Responsibility. An extension marked as non-critical (criticality set to FALSE) may be ignored or discarded, while an extension marked as critical must be known and performed for acceptance of Responsibility of an EDIM.

Extension Type

Indicates the type of the extension.

Extension Value

Indicates the value of the extension. The value must be of the type indicated by the value of the Extension Type attribute.

3.8 EDI Heading

An instance of class **EDI Heading** is a set of characteristics describing an EDIM.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
This EDIM	Object(EDIM Identifier)	-	1	-
Originator	Object(OR Name)	-	0-1	-
Recipients	Object(Recipients)	-	0 or more	-
EDIN Receiver	Object(EDIN Receiver)	-	0-1	-
Responsibility Forwarded	Boolean	-	1	False
EDI Body Part Type	String(Object Identifier)	-	1	edifact-ISO646
Incomplete Copy	Boolean	-	1	False
Expiry Time	String(UTC Time)	-	0-1	-
Related EDI Messages	Object(EDIM Identifier)	-	0 or more	-
Related non-EDI Messages	Object(External)	-	0 or more	-
Obsolete EDIMs	Object(EDIM Identifier)	-	0 or more	-
EDI Application Security Elements	Object(EDI Application Security Elements)	-	0-1	-
Cross Referencing Information	Object(Cross Referencing Information)	-	0 or more	-
EDI Message Type	String(Teletex)	1-6	0 or more	-
Service String Advice	Object(Service String Advice)	-	0-1	-
Syntax Identifier	Object(Syntax Identifier)	-	0-1	-
Interchange Sender	Object(Interchange Sender-Recipient)	-	0-1	-
Date And Time of Preparation	String(UTC Time)	-	0-1	-
Application Reference	String(Teletex)	1-14	0-1	-
Heading Extensions	Object(EDI Extension)	-	0 or more	-

Table 3-6. OM Attributes of an EDI Heading

This EDIM

Identifies the EDIM.

Originator

The OR Name that identifies the EDIM's originator. If the Originator attribute is not present in the EDIM Heading on reception, then the Originating-name of the delivery envelope shall be used to determine the originator of the EDIM (see CCITT X.411 | ISO/IEC 10021-4, listed in **Referenced Documents**). The Originating-name is contained in the Originator Name attribute of an instance of the class Delivery Envelope defined in the Message Handling Package of the **API to Electronic Mail (X.400) CAE Specification**.

Recipients

Identifies the user(s) and distribution lists (*DL*) that are the (preferred) recipient(s) of the EDIM. The values of this attribute are instances of the Recipient class, one instance for each recipient. If the Recipients attribute is not present in the EDIM Heading on reception, then the This-recipient-name of the delivery envelope shall be used to determine the recipient of the EDIM (see CCITT X.411 | ISO/IEC 10021-4). The This-recipient-name is contained in the Intended Recipient Name attribute of an instance of the class Delivery Envelope defined in the Message Handling Package of the **API to**

Electronic Mail (X.400) CAE Specification). A message can be redirected or forwarded so the preferred recipients indicated in this attribute are not necessarily the final recipients of the EDIM.

EDIN Receiver

Identifies the recipient to whom EDINs are to be sent. The value of this attribute is supplied by the originator of the EDIM when the Recipient of a requested notification is different from the Originator of the message. It consists of a sequence of Recipient Name, EDIM Identifier and First Recipient (see the definition of the EDIN Receiver class). This attribute shall not be present if EDI Notification Requests are not made. (EDI Notification Requests are made within the value of the Recipients attribute.)

This attribute shall be present in a forwarded message when the forwarding EDI user agent (EDI-UA) or EDI message store (EDI-MS) forwards Responsibility. This attribute may be present when the forwarding EDI-UA accepts Responsibility. Rules related to the construction of this attribute are given in the EDI messaging standards (see **Referenced Documents**).

Responsibility Forwarded

Indicates whether Responsibility was forwarded. If this attribute has the value TRUE it indicates to a receiving UA that Responsibility was forwarded. If this attribute has the value FALSE (or is absent) it indicates to a receiving UA, that the security elements of the inner envelope have been checked. Subject to the security policy in force, the security elements may have been checked when the message was forwarded. However, when Responsibility is accepted, the security elements shall be checked.

EDI Body Part Type

Indicates the EDI standard and EDI character sets used in the Primary Body Part. It is represented by a single object identifier. The following standard values have object identifiers defined in the EDI messaging standards (see **Referenced Documents**). The first column of the table lists the symbolic values. The second column specifies, in ASN.1, the object identifiers denoted by the values.

Value	Object Identifier (ASN.1)
edifact-ISO646	{joint-iso-ccitt mhs-motis(6) edims(7) 11 0}
edifact-T61	{joint-iso-ccitt mhs-motis(6) edims(7) 11 1}
edifact-octet	{joint-iso-ccitt mhs-motis(6) edims(7) 11 2}
ansiX12-ISO646	{joint-iso-ccitt mhs-motis(6) edims(7) 11 3}
ansiX12-T61	{joint-iso-ccitt mhs-motis(6) edims(7) 11 4}
ansiX12-octet	{joint-iso-ccitt mhs-motis(6) edims(7) 11 5}
ansiX12-ebcdic	{joint-iso-ccitt mhs-motis(6) edims(7) 11 6}
unt-di-ISO646	{joint-iso-ccitt mhs-motis(6) edims(7) 11 7}
unt-di-T61	{joint-iso-ccitt mhs-motis(6) edims(7) 11 8}
unt-di-octet	{joint-iso-ccitt mhs-motis(6) edims(7) 11 9}
private-octet	{joint-iso-ccitt mhs-motis(6) edims(7) 11 10}
undefined-octet	{joint-iso-ccitt mhs-motis(6) edims(7) 11 11}

Table 3-7. Selected Values of the EDI Body Part Type Attribute

The default value of this attribute is edifact-ISO646.

Both the EDI Body Part, and those values within an instance of an EDI Heading that are Octet Strings and are derived from the EDI Interchange, are encoded in the character set referred to by this attribute.

The value of the EDI Body Part Type attribute shall be used with the *EITs* (Encoded Information Types) class defined in the MH Package. This enables a UA to signal to the

MTS the EDI standard to which the EDIM's Primary Body Part complies. 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 EDIM.

Note: The term *Encoded Information Type* is defined in CCITT X.402 | ISO/IEC 10021-2. See also CCITT X.411 | ISO/IEC 10021-4 (see **Referenced Documents**).

Incomplete Copy

The Incomplete Copy attribute indicates that the forwarded EDIM is an incomplete copy of an EDIM. This attribute shall have the value TRUE if body parts are removed when an EDIM is forwarded. The default value is FALSE.

Note: The term *forwarded EDIM* is defined in the EDI messaging standards (see **Referenced Documents**).

Expiry Time

The date and time when the originator considers this EDIM to be no longer valid.

Related EDI Messages

Identifies EDIMs that the originator of this EDIM considers related to it. If the related message identifies messages from other services, the user component of the message identifier (EDIM Identifier) must be present. Message identifier values of the referenced message of other service types than EDIMG are carried in this attribute.

Related non-EDI Messages

Identifies messages (other than EDIMs) that the originator of this EDIM considers related to it. A related message may be an EDIM or any other type of message (for example, an IPM).

Obsoleted EDIMs

Identifies one or more EDIMs that the present EDIM obsoletes. Each obsoleted EDIM is identified by an EDIM Identifier.

EDI Application Security Elements

Allows an EDI application to exchange security elements having an end-to-end significance.

Cross Referencing Information

Allows an EDI application to reference individual body parts within the same EDIM and within other EDIMs. It contains a set of cross reference data.

EDI Message Type

Indicates the EDI Message type(s) present in the EDI Interchange. The values of this attribute are distinct strings identifying the message types defined in the EDI standards. The values for this attribute shall be:

- EDIFACT, Message Type from the UNH segment
- ANSIX12, Transaction Set ID from the ST segment
- UNTDI, Message Type from the MHD segment

Service String Advice

Indicates the Service String Advice of the EDI Interchange. This is semantically identical to the "UNA, Service string advice" of the EDIFACT Interchange. The meaning of this attribute is described fully in the EDIFACT standard (see **Referenced Documents**).

Syntax Identifier

Indicates the syntax used in this EDI Interchange. This is semantically identical to the

“Syntax identifier” of the EDIFACT UNB segment (see **Referenced Documents**).

Interchange Sender

Indicates the sender of the EDI Interchange. This is semantically identical to the “Interchange sender” of the EDIFACT UNB segment (see **Referenced Documents**).

Date And Time of Preparation

Indicates the date and time of preparation of the EDIM. This is in UTC Time and is derived from the “Date and time of preparation” of the EDIFACT UNB segment (see **Referenced Documents**).

Application Reference

Provides a general reference to an application or function. This is semantically identical to the “Application reference” segment of the EDIFACT UNB segment (see **Referenced Documents**).

Heading Extensions

Contains extensions to the EDI Heading class.

3.9 EDI Notification

An instance of class **EDI Notification** is an information object conveyed between users of EDI Messaging. The information objects that users exchange in EDI Messaging are of two kinds: EDI Messages (EDIM) and EDI Notifications (EDIN). The EDI Notification class is an abstract class for which three concrete subclasses are defined in this specification: Forwarded Notification (FN), Negative Notification (NN) and Positive Notification (PN).

The recipient of the EDIN is the Originator of the subject EDIM, or, if present, the OR Name indicated in the EDIN Receiver attribute of the EDIM. There shall be at most one recipient specified for an EDIN. There shall be at most one PN, NN or FN originated for each subject EDIM by each recipient of whom notifications are requested (except that an NN may be originated by the same UA subsequent to an FN, in accordance with the EDI messaging standards). One FN is originated, if, and only if, it is requested by each recipient that forwards an EDIM. In accordance with the provisions of the EDI Messaging standards (see **Referenced Documents**), the originator of the subject EDIM shall receive at most one PN or NN for each recipient of whom notifications were requested, regardless of how many times the EDIM is forwarded. The originator may, however, receive multiple FNs.

This class is an abstract class which has the attributes of its superclasses - *Content* and *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Subject EDIM	Object(EDIM Identifier)	-	1	-
EDIN Originator	Object(OR Name)	-	1	-
First Recipient	Object(OR Name)	-	0-1	-
Notification Time	String(UTC Time)	-	1	-
Security Elements	Object(Security Elements)	-	0-1	-
EDIN Initiator	Enum(EDIN Initiator)	-	1	-
Notification Extensions	Object(EDI Extension)	-	0 or more	-
Supplementary Information	String(Teletex)	1-256	0-1	-

Table 3-8. OM Attributes of an EDI Notification

Subject EDIM

Identifies the EDIM to which this EDIN refers. If Responsibility has been forwarded, this identifier is passed in the EDIN Receiver attribute of the Heading of the EDIM, otherwise it is passed in the This EDIM attribute.

EDIN Originator

Contains the OR Name of the UA constructing the notification. The class OR Name is defined in the MH Package.

First Recipient

Contains the OR Name of the first recipient in a forwarding chain. The class OR Name is defined in the MH Package. This attribute, together with other attributes, is used by the recipient of the notification to correlate the notification and the original message. If the originator of the EDIN is not the recipient specified by the originator of the subject EDIM, then the First Recipient attribute shall be present in the EDIN.

Notification Time

Contains the date and time at which this notification was generated.

Security Elements

Indicates the security services requested on this notification.

EDIN Initiator

Indicates who initiated the notification. Each value is one of:

- **internal-ua**, the UA generated the EDIN either for local reasons or because the generation had been delegated to it by the user.
- **internal-ms**, the MS generated the EDIN either for local reasons or because the generation had been delegated to it by the user
- **external-ua**, the generation of the EDIN was requested by the user.

Origination of a Positive Notification implies that Responsibility has been accepted, regardless of the value of this attribute. The value of this attribute shall be consistent with the choice (UA, user, PDAU) of the Reason Code attribute for Negative Notifications and Forwarded Notifications.

Notification Extensions

Contains extensions to the EDI Notification class.

Supplementary Information

The Supplementary Information attribute may be used to return further information to the EDIN recipient to clarify the notification.

3.10 EDI Notification Requests

An instance of class **EDI Notification Requests** describes the type of notifications and the security required for the notifications which are requested of a preferred recipient.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax ¹	Value Length	Value Number	Value Initially
Positive Notification	Boolean	-	1	False
Negative Notification	Boolean	-	1	False
Forwarded Notification	Boolean	-	1	False
Proof of Notification	Boolean	-	1	False
Non-repudiation of Notification	Boolean	-	1	False
Proof of Reception	Boolean	-	1	False
Non-repudiation of Reception	Boolean	-	1	False

¹ The Bit String syntax of the EDI Messaging standards (see **Referenced Documents**) has been mapped into Booleans in the interface class definition.

Table 3-9. OM Attributes of EDI Notification Requests

Each of the attributes in this class corresponds to a request that may be made of the preferred recipient. All attributes default to False, indicating that the request is not made.

Positive Notification

Requests a positive notification (PN), that reports its originator's acceptance of Responsibility of an EDIM.

Negative Notification

Requests a negative notification (NN), that reports its originator's refusal to accept Responsibility of an EDIM.

Forwarded Notification

Requests a forwarded notification (FN), that reports that Responsibility of an EDIM has been forwarded together with the EDIM.

Proof of Notification

Indicates whether content-integrity-check shall be requested when submitting the EDIN to the MTS. (See CCITT Recommendation X.411 | ISO/IEC 10021-4, listed in **Referenced Documents**.)

Non-repudiation of Notification

Indicates whether content-integrity-check shall be requested with a non-repudiable certificate when submitting the EDIN to the MTS. (See CCITT Recommendation X.411 | ISO/IEC 10021-4, listed in **Referenced Documents**.)

Proof of Reception

Indicates whether, when submitting the EDIN 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) as defined in CCITT X.411 | ISO/IEC 10021-4 (see **Referenced Documents**).

Non-repudiation of Reception

Indicates whether when submitting the EDIN 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) as defined in CCITT X.411 | ISO/IEC 10021-4 (see **Referenced Documents**).

Note: Security services are available only if the MTS supports secure messaging.

3.11 EDIM

An instance of class **EDIM** is an information object conveyed between users in EDI Messaging. The information objects that users exchange in EDI messaging are of two kinds: EDI Messages (EDIM) and EDI Notifications (EDIN).

An instance of this class has the attributes of its superclasses - *Content* and *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Heading	Object(EDI Heading)	-	1	-
Primary Body Part	Object(Primary Body Part)	-	1	-
Additional Body Parts	Object(EDIM Externally Defined Body Part)	-	0 or more	-

Table 3-10. OM Attributes of an EDIM

Heading

A set of heading attributes, each an information item that gives a characteristic of the EDI Message.

Primary Body Part

Contains an EDI information object. Examples of types of EDI information objects are EDI Interchanges defined by **EDIFACT**, **UNTDI** and **ANSIX12** (see **Referenced Documents**).

Additional Body Parts

Additional body parts related to the Primary Body Part but of different types. Examples of related body parts include textual information, voice annotation or graphics to be used in conjunction with the interchange.

3.12 EDIM Body Part

An instance of class **EDIM Body Part** contains an EDIM, and optionally, its delivery envelope. This body part is used for forwarding EDIMs. When an EDIM is forwarded, its structure shall comply with the rules given in the EDI messaging standards (see **Referenced Documents**).

An instance of this class has the attributes of its superclasses - *Primary Body Part* and *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Delivery Envelope	Object(Delivery Envelope)	-	0-1	-
EDI Supplementary Info.	String(Teletex)	1-256	0-1	-
Heading	Object(EDI Heading)	-	1	-
Primary Body Part	Object(Primary Body Part)	-	0-1	-
Additional Body Parts	Object(EDIM Ext. Defined Body Part)	-	0 or more	-

Table 3-11. OM Attributes of an EDIM Body Part

Delivery Envelope

The delivery envelope of the EDIM. The Delivery Envelope class is part of the Message Handling Package. A value of the Delivery Envelope attribute must be an instance of the Delivery Envelope class as specified in the MH Package with the exception that the MTS Identifier attribute shall not be present and the Delivery Time attribute is optional (that is, it may not be present).

The delivery envelope shall be present if security services are invoked.

EDI Supplementary Information

Additional parameters specific to EDI.

Heading

Heading of the EDIM.

Primary Body Part

The primary body part of the EDIM. This attribute may be absent if the primary body part has been removed from the EDIM.

Additional Body Parts

Additional body parts of the EDIM. This attribute may contain complete body parts or body parts from which the data portion has been removed. The latter, incomplete values, are used as place holders for removed body parts. They may consist of only the Body Part Reference, or a modified Externally Defined Body Part. In the latter case the object identifier and Body Part Reference of the removed body part are preserved; from the External Parameters attribute (if present) and External Data attribute of the removed body part, only the object identifier and the identifier octets of the appropriate Encoding attribute of the External object instance are preserved. That is, in the instance of the External object, the Encoding attribute which is present shall be of zero length and hence, have no content. (Refer to the definition of the External class in the Object Management Package.)

3.13 EDIM Externally Defined Body Part

An instance of class **EDIM Externally Defined Body Part** contains body parts that relate to the Primary Body Part and may be carried together with an EDI Interchange. These body parts are not EDI Interchanges and do not include EDI Interchanges. 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 and optionally a Body Part Reference that may be used for cross referencing to a body part.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Body Part Reference	Integer	-	0-1	-
External Body Part	Object(Externally Defined Body Part)	-	0-1	-

Table 3-12. OM Attributes of an EDIM Externally Defined Body Part

Body Part Reference

A reference to this body part. The reference may be used for cross referencing body parts. The references assigned to body parts shall be unique within an EDIM. They are assigned when a body part is created, and are not modified subsequently. A reference shall be present if the originator wishes to cross reference the body part at creation or in the future.

External Body Part

The content of the body part. The value of this attribute is an instance of the class Externally Defined Body Part which is part of the Interpersonal Messaging Package defined in the **API to Electronic Mail (X.400) CAE Specification**. Some Externally Defined body part types are defined in CCITT X.420 | ISO/IEC 10021-7 (see **Referenced Documents**).

3.14 EDIM Identifier

An instance of class **EDIM Identifier** identifies an EDIM. It contains an OR Name and a string which may contain a time, sequence number, or other information sufficient to uniquely identify an EDIM.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
User	Object(OR Name)	-	1	-
User-relative-identifier	String(Printable)	0-64 ¹	1	-

¹A length of zero is discouraged.

Table 3-13. OM Attributes of an EDIM Identifier

User Identifies the user who originates the EDIM. One of the user's OR Names. The class OR Name is defined in the MH Package.

User-relative-identifier

Unambiguously identifies the EDIM, distinguishing it from all other EDIMs that the user originates.

3.15 EDIN Receiver

An instance of class **EDIN Receiver** identifies the recipient to whom EDINs are to be sent. An instance is created by the originator of the EDIM when the Recipient of a requested notification is different from the Originator of the message.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
EDIN Receiver Name	Object(OR Name)	-	1	-
Original EDIM Identifier	Object(EDIM Identifier)	-	0-1	-
First Recipient	Object(OR Name)	-	0-1	-

Table 3-14. OM Attributes of an EDIN Receiver

EDIN Receiver Name

Identifies the recipient of a requested notification. The class OR Name is defined in the MH Package.

Original EDIM Identifier

Identifies the original EDIM for which the notification is being sent.

First Recipient

Identifies the first recipient in a forwarding chain. This attribute (together with information from instances of other classes), is used by the recipient of the notification to correlate the notification and the original message. The First Recipient attribute shall not be present if the corresponding EDIM contains more than one recipient with associated EDI Notification Requests. The class OR Name is defined in the MH Package.

Note: The Original EDIM Identifier and the First Recipient attributes shall not be present when the Primary Body Part is an EDI Body Part (that is, when the original originator first creates the EDIM). The Original EDIM Identifier and First Recipient attributes are included in order to allow the recipient to construct the EDIN for a forwarded EDIM.

3.16 Externally Defined Body Part

The **Externally Defined Body Part** class is defined in the IM (Interpersonal Messaging) Package (see **API to Electronic Mail (X.400) CAE Specification**). The EDI Package closure contains this class and its closure.

3.17 FN PDAU Reason Code

An instance of class **FN PDAU Reason Code** indicates why the Responsibility of the subject EDIM was forwarded. An instance of this class contains reasons indicated by a PDAU for the issuing of an FN.

An instance of this class has the attributes of its superclasses - *FN Reason Code*, *Reason Code* and *Object* - and no additional attributes. A set of standard values for the Reason Code attributes are defined below. These standard values are specific to this subclass of Reason Code.

Basic Code

The following standard values are defined:

- **unspecified**, the reason has not been specified by the PDAU.
- **forwarded-for-physical-rendition-and-delivery**, the PDAU has determined that it can render the EDIM for physical delivery.

3.18 FN Reason Code

An instance of class **FN Reason Code** indicates why the Responsibility of the subject EDIM was forwarded. This class is an abstract class which has the attributes of its superclasses - *Reason Code* and *Object* - and no other attributes. This specification defines three subclasses of FN Reason Code, for reason codes issued from an EDI-UA or EDI-MS (FN UAMS Reason Code), from a user (FN User Reason Code), or from a physical delivery access unit, PDAU, (FN PDAU Reason Code).

3.19 FN UAMS Reason Code

An instance of class **FN UAMS Reason Code** indicates why the Responsibility of the subject EDIM was forwarded. An instance of this class contains reasons indicated by a UA or MS for the issuing of a forwarded notification.

An instance of this class has the attributes of its superclasses - *FN Reason Code*, *Reason Code* and *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
FN Security Check	Boolean	-	1	False

Table 3-15. OM Attributes of an FN UAMS Reason Code

Basic Code

This attribute is inherited from the superclass Reason Code. It has the following standard values:

- **unspecified**, the reason has not been specified by the UA or MS.
- **onward-routing**, used whenever the UA decides to re-route the subject EDIM for local reasons.
- **recipient-unknown**
- **originator-unknown**
- **forwarded-by-edi-ms**

Diagnostic

This attribute is inherited from the superclass Reason Code. It has the following standard values:

- **recipient-name-changed**
- **recipient-name-deleted**

FN Security Check

Indicates whether all security features present have been validated.

3.20 FN User Reason Code

An instance of class **FN User Reason Code** indicates why the Responsibility of the subject EDIM was forwarded. An instance of this class contains reasons indicated by a User for the issuing of an FN.

An instance of this class has the attributes of its superclasses - *FN Reason Code*, *Reason Code* and *Object* - and no additional attributes. A set of standard values for the Reason Code attributes are defined below. These standard values are specific to this subclass of Reason Code.

Basic Code

The following standard values are defined:

- **unspecified**, the reason has not been specified by the user.
- **forwarded-for-archiving**
- **forwarded-for-information**
- **forwarded-for-additional-action**
- **subscription-changed**
- **heading-field-not-supported**
- **bodypart-type-not-supported**
- **message-type-not-supported**
- **syntax-identifier-not-supported**
- **interchange-sender-unknown**
- **user-defined-reason**

3.21 Forwarded Notification

An instance of class **Forwarded Notification** reports that Responsibility of an EDIM has been forwarded together with the subject EDIM. A Forwarded Notification (FN) is sent by a UA, if, and only if, the originator has requested forwarding notification. A Forwarded Notification is sent when the UA determines that it cannot accept Responsibility, and decides to forward the EDIM, and the EDI Notification Requests contained in the EDIM, to another UA.

An instance of this class has the attributes of its superclasses - *EDI Notification*, *Content* and *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Forwarded To	Object(OR Name)	-	1	-
FN Reason Code	Object(FN Reason Code)	-	1	-

Table 3-16. OM Attributes of a Forwarded Notification

Forwarded To

Indicates the new recipient of the (forwarded) subject EDIM.

FN Reason Code

The Forwarded Notification Reason indicates why the Responsibility of the subject EDIM was forwarded. Additional information may be carried in any combination of the Reason Code attribute or the Supplementary Info attribute.

3.22 Integrity Check Basis

The **Integrity Check Basis** class is defined in the SM (Secure Messaging) Package. The EDI Package closure contains this class and its closure.

3.23 Interchange Sender-Recipient

An instance of class **Interchange Sender-Recipient** indicates the sender or recipient of an EDI Interchange and is semantically identical to the “Interchange sender/Interchange recipient” of the EDIFACT UNB segment.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Identification Code	String(Teletex)	1-35	1	-
Identification Code Qualifier	String(Teletex)	1-4	0-1	-
Routing Address	String(Teletex)	1-14	0-1	-

Table 3-17. OM Attributes of an Interchange Sender-Recipient

Further details on the meaning of this class and its attributes can be found in the EDIFACT standard (see **Referenced Documents**).

Identification Code

Identifies the sender or recipient of an interchange and is semantically identical to the “Sender identification/Recipient identification” component of the Interchange sender/recipient of the EDIFACT UNB segment.

Identification Code Qualifier

The Identification Code Qualifier, if present, is a qualifier of the Identification Code of a sender or recipient. The qualifier is semantically identical to the “Identification Code Qualifier” component of the Interchange sender/recipient of the EDIFACT UNB segment.

Routing Address

The Routing Address, if present, is an address for routing to the sender or recipient specified in the Identification Code. This is semantically identical to the “Address for reverse routing/Routing address” component of the Interchange sender/recipient of the EDIFACT UNB segment.

3.24 Negative Notification

An instance of class **Negative Notification** reports its originator's refusal to accept Responsibility of an EDIM. A Negative Notification (NN) is sent by a UA, if, and only if, the originator has requested negative notification. An NN is sent when the UA determines that it can neither accept Responsibility, nor forward the EDIM and the EDI Notification Requests contained in the EDIM to another UA.

An instance of this class has the attributes of its superclasses - *EDI Notification*, *Content* and *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
NN Reason Code	Object(NN Reason Code)	-	1	-

Table 3-18. OM Attributes of a Negative Notification

NN Reason Code

The Negative Notification Reason indicates why the subject EDIM could not be passed to the user by the UA originating the EDIN. Additional information may be carried in any combination of the Reason Code attribute or the Supplementary Info attribute.

3.25 NN PDAU Reason Code

An instance of class **NN PDAU Reason Code** indicates why the subject EDIM could not be passed to the user by the UA originating the EDIN. An instance of this class contains reasons indicated by a PDAU for the issuing of an NN.

An instance of this class has the attributes of its superclasses - *NN Reason Code*, *Reason Code* and *Object* - and no additional attributes. A set of standard values for the Reason Code attributes are defined below. These standard values are specific to this subclass of Reason Code.

Basic Code

The following standard values are defined:

- **unspecified**, the reason has not been specified by the PDAU.
- **undeliverable-mail**, the PDAU has determined that it cannot perform physical delivery of the EDIM.
- **physical-rendition-not-performed**, the PDAU cannot perform the physical rendition of the EDIM.

Diagnostic

The following standard values are defined:

- **physical-rendition-attributes-not-supported**
- Reasons why the mail was undeliverable by the PDAU:
- **physical-delivery-address-incorrect**
 - **physical-delivery-office-incorrect-or-invalid**
 - **physical-delivery-address-incomplete**
 - **recipient-unknown**
 - **recipient-deceased**
 - **organization-expired**
 - **recipient-refused-to-accept**
 - **recipient-did-not-claim**
 - **recipient-changed-address-permanently**
 - **recipient-changed-address-temporarily**
 - **recipient-changed-temporary-address**
 - **new-address-unknown**
 - **recipient-did-not-want-forwarding**
 - **originator-prohibited-forwarding**

3.26 NN Reason Code

An instance of class **NN Reason Code** indicates why the subject EDIM could not be passed to the user by the UA originating the EDIN. This class is an abstract class which has the attributes of its superclasses - *Reason Code* and *Object* - and no other attributes. This specification defines three subclasses of NN Reason Code, for reason codes issued from an EDI-UA or EDI-MS (NN UAMS Reason Code), from a user (NN User Reason Code), or from a physical delivery access unit, PDAU, (NN PDAU Reason Code).

3.27 NN UAMS Reason Code

An instance of class **NN UAMS Reason Code** indicates why the subject EDIM could not be passed to the user by the UA originating the EDIN. An instance of this class contains reasons indicated by a UA or MS for the issuing of a negative notification.

An instance of this class has the attributes of its superclasses - *NN Reason Code*, *Reason Code* and *Object* - and no additional attributes. A set of standard values for the Reason Code attributes are defined below. These standard values are specific to this subclass of Reason Code.

Basic Code

The following standard values are defined:

- **unspecified**, the reason has not been specified by the UA or MS.
- **cannot-deliver-to-user**, the EDI Interchange cannot be passed on to the user (physical delivery errors are indicated by this code).
- **delivery-timeout**, the EDI Interchange could not be passed on to the user within a specified time limit.
- **message-discarded**, the UA/MS discarded the message before handoff to user.
- **subscription-terminated**, the recipient's subscription terminated after delivery but before handoff to user.
- **forwarding-error**, EDI Forwarding was attempted, but failed.
- **security-error**, there was a security error.

Diagnostic

The following standard values are defined:

- **protocol-violation**, used if the UA detects a protocol error.
- **edim-originator-unknown**, the originator of the EDIM is unknown to the UA or MS.
- **edim-recipient-unknown**, the recipient of the EDIM is unknown to the UA or MS.
- **edim-recipient-ambiguous**, the EDIM recipients or originator are not valid.
- **action-request-not-supported**, the action requested by the recipient is not performed.
- **edim-expired**, the expiry date of the received EDIM occurred before the subject EDIM was successfully passed to the user or forwarded by the EDI-UA.
- **edim-obsolete**, the EDIM Identifier of the received EDIM was contained in the Obsolete EDIM attribute of a previously received EDIM.
- **duplicate-edim**, the same EDIM was received more than once from the same originator.
- **unsupported-extension**, the EDIM contains an extension which is not supported by the UA.
- **incomplete-copy-rejected**, the EDI-UA does not accept EDIMs with the Incomplete Copy Indication true.
- **edim-too-large-for-application**, the EDIM cannot be delivered to the user due to length constraints forwarding error diagnostic codes.

- **forwarded-edim-not-delivered**, an Non-Delivery Report was received for the forwarded EDIM.
- **forwarded-edim-delivery-time-out**, no Delivery Report is received within a given period.
- **forwarding-loop-detected**, the UA received an EDIM which contained a previously forwarded EDIM.
- **unable-to-accept-responsibility**, the EDI-UA cannot accept or forward responsibility.

Interchange header diagnostic codes:

- **interchange-sender-unknown**, the UA does not recognise the interchange-sender of the EDI interchange.
- **interchange-recipient-unknown**, the UA cannot find a valid interchange recipient in the Recipient Specifier.
- **invalid-heading-field**, the interchange contains an invalid heading attribute.
- **invalid-bodypart-type**, the interchange contains an invalid body part type.
- **invalid-message-type**, the interchange contains an invalid message type.
- **invalid-syntax-id**, the interchange contains an invalid syntax identifier.

Security error diagnostic codes:

Depending on the security policy in force, the security error diagnostic codes may or may not be present.

- **message-integrity-failure**
- **forwarded-message-integrity-failure**
- **unsupported-algorithm**
- **decryption-failed**
- **token-error**
- **unable-to-sign-notification**
- **unable-to-sign-message-receipt**
- **authentication-failure**
- **security-context-failure**
- **message-sequence-failure**
- **message-security-labelling-failure**
- **repudiation-failure**
- **proof-of-failure**

3.28 NN User Reason Code

An instance of class **NN User Reason Code** indicates why the subject EDIM could not be passed to the user by the UA originating the EDIN. An instance of this class contains reasons indicated by the user for the issuing of an NN.

An instance of this class has the attributes of its superclasses - *NN Reason Code*, *Reason Code* and *Object* - and no additional attributes. A set of standard values for the Reason Code attributes are defined below. These standard values are specific to this subclass of Reason Code.

Basic Code

The following standard values are defined:

- **unspecified**, the reason has not been specified by the user.
- **syntax-error**, the user has discovered a syntax error within the EDI Interchange.
- **interchange-sender-unknown**
- **interchange-recipient-unknown**, the UA cannot find a valid interchange recipient in the Recipient Specifier attribute of the interchange.
- **invalid-heading-field**
- **invalid-bodypart-type**
- **invalid-message-type**
- **functional-group-not-supported**
- **subscription-terminated**, the recipient's subscription has been terminated, unknown to the EDIMS-User service.
- **no-bilateral-agreement**
- **user-defined-reason**

3.29 Object

The **Object** class is defined in the OM (Object Management) Package. The EDI Package closure contains this class. All classes in this specification are subclasses of Object.

3.30 OR Name

The **OR Name** class is defined in the MH (Message Handling) Package. The EDI Package closure contains this class and its closure.

3.31 Positive Notification

An instance of class **Positive Notification** is an EDIN that reports its originator's acceptance of Responsibility of an EDIM. A Positive Notification (PN) is sent by the recipient UA if, and only if, the originator of the EDIM has requested positive notification. The exact procedures which constitute acceptance of Responsibility are a local matter. For example, the UA may construct the PN as soon as it passes the message to the user. Alternatively, it may wait for an external stimulus from the user indicating that the message has been accepted and a PN can therefore be sent by the UA.

An instance of this class has the attributes of its superclasses - *EDI Notification*, *Content* and *Object* - and no additional attributes.

3.32 Primary Body Part

An instance of class **Primary Body Part** contains an EDI information object. An EDIM contains exactly one instance of a concrete subclass of the Primary Body Part class.

This class is an abstract class which has the attributes of its superclass - *Object* - and no additional attributes.

3.33 Reason Code

An instance of class **Reason Code** indicates why a notification was issued. This specification defines subclasses of Reason Code, each of which defines a set of standard values for the reason code attributes which are specific to different situations.

This class is an abstract class which has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Basic Code	Integer	-	1	-
Diagnostic	Integer	-	0-1	-

Table 3-19. OM Attributes of a Reason Code

Basic Code

Indicates the basic reason a notification is sent. The value must be between 0 and 32767.

Diagnostic

This attribute is used to further specify the error signalled in the basic code. The value must be between 1 and 32767, if this attribute is present. Additional information may be indicated in the supplementary information attribute of the notification.

3.34 Recipient Reference

An instance of class **Recipient Reference** identifies a reference meaningful to the recipient's EDI application. This class is semantically identical to the "Recipient's Reference, Password" of the EDIFACT UNB segment.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Recipient Reference	String(Teletex)	1-14	1	-
Recipient Reference Qualifier	String(Teletex)	1-2	0-1	-

Table 3-20. OM Attributes of a Recipient Reference

Further details on the meaning of this class and its attributes can be found in the EDIFACT standard (see **Referenced Documents**).

3.35 Recipients

An instance of class **Recipients** identifies the user(s) and distribution lists (*DL*) that are the (preferred) recipient(s) of the EDIM. This class also contains attributes used to make certain requests of the preferred recipients including requests for notifications and acknowledgements.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Recipient	Object(OR Name)	-	1	-
Action Request	String(Object Identifier)	-	1	for-action
EDI Notification Requests	Object(EDI Notification Requests)	-	0-1	-
Responsibility Passing Allowed	Boolean	-	1	False
Interchange Recipient	Object(Interchange Sender-Recipient)	-	0-1	-
Recipient Reference	Object(Recipient Reference)	-	0-1	-
Interchange Control Reference	String(Teletex)	1-14	0-1	-
Processing Priority Code	String(Teletex)	1	0-1	-
Acknowledgement Request	Boolean	-	1	False
Comms Agreement Id	String(Teletex)	1-35	0-1	-
Test Indicator	Boolean	-	1	False
Authorization Information	Object(Authorization Information)	-	0-1	-
Recipient Extensions	Object(EDI Extension)	-	0 or more	-

Table 3-21. OM Attributes of Recipients

Recipient

Identifies a preferred recipient. The class OR Name is defined in the MH Package.

Action Request

Indicates what action the originator requests from the recipient. The following standard values have object identifiers defined in the EDI messaging standards (see **Referenced Documents**). The first column of the table lists the symbolic values. The second column specifies in ASN.1 the object identifiers denoted by the values.

Value	Object Identifier (ASN.1)
for-action	{joint-iso-ccitt mhs-motis(6) edims(7) 13 0}
copy	{joint-iso-ccitt mhs-motis(6) edims(7) 13 1}

Table 3-22. Selected Values of the Action Request Attribute

The default value of the attribute is for-action. Additional values for this attribute can be defined by any interested parties.

EDI Notification Requests

Contains requests made of the preferred recipient denoted by the Recipient attribute. Requests may include the type of notification and security constraints on the notification.

Responsibility Passing Allowed

Indicates whether forwarding of Responsibility is allowed. A recipient of a message with the Responsibility Passing Allowed attribute set to FALSE shall originate EDIN's as requested, and shall not forward Responsibility. If this attribute is set to TRUE,

Responsibility may be forwarded to at most one recipient.

Interchange Recipient

The Interchange Recipient identifies the EDI Interchange recipient and is semantically identical to the “Interchange recipient” of the EDIFACT UNB segment.

Recipient Reference

Identifies a reference meaningful to the recipient’s EDI application. This attribute is semantically identical to the “Recipient’s Reference, Password” of the EDIFACT UNB segment.

Interchange Control Reference

Indicates the Interchange Control Reference as assigned by the Interchange sender. This is semantically identical to the “Interchange control reference” of the EDIFACT UNB segment.

Processing Priority Code

Indicates the EDI application Processing Priority Code. This attribute is semantically identical to the “Processing priority code” in the EDIFACT UNB segment. It consists of a string.

Acknowledgement Request

Indicates the request for EDI acknowledgement as indicated by the interchange sender. This attribute is semantically identical to the “Acknowledgement request” in the EDIFACT UNB segment. Its value is a Boolean, where the value TRUE indicates a request for acknowledgement. Absence of this attribute shall be interpreted as the value FALSE.

Communications Agreement Id

Indicates the type of Communications Agreement controlling the interchange, for example a Customs agreement. This attribute is semantically identical to the “Communications agreement id” in the EDIFACT UNB segment.

Test Indicator

Indicates that the EDI Interchange is a test. This attribute is semantically identical to the “Test indicator” in the EDIFACT UNB segment. The value TRUE indicates that the EDI Interchange is a test. Absence of this attribute shall be interpreted as the value FALSE.

Authorization Information

Indicates who authorised the interchange. This attribute is semantically identical to the “Authorization information” in the ANSIX12 Interchange.

Recipient Extensions

Contains extensions to the Recipient class.

3.36 Security Elements

An instance of class **Security Elements** is used to indicate whether “proof or non-repudiation of content received”, or “EDI application security” services are required for the notification.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Original Content	Object(Content)	-	0-1	-
Original Content Integrity Check	Object(Integrity Check Basis)	-	0-1	-
EDI Appl Security Elements	Object(EDI Appl Sec Elements)	-	0-1	-

Table 3-23. OM Attributes of Security Elements

Original Content

Content of the subject EDIM. The Content class is part of the Message Handling Package defined in the **API to Electronic Mail (X.400) CAE Specification**).

Original Content Integrity Check

Provides the notification recipient with a means of validating that the message content of the EDIM has not been modified. The use of this attribute is described in the messaging standards (see **Referenced Documents**). The Integrity Check Basis class is part of the Secure Messaging Package defined in the **API to Electronic Mail (X.400) CAE Specification**).

EDI Application Security Elements

Identifies a set of security elements. This attribute is used to allow an EDI application to exchange security elements having an end-to-end significance.

3.37 Service String Advice

An instance of class **Service String Advice** indicates the service string advice of an EDI Interchange and is semantically identical to the “UNA, Service string advice” of the EDIFACT Interchange.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Component Data Element Separator	String(Octet)	1	1	-
Data Element Separator	String(Octet)	1	1	-
Decimal Notation	String(Octet)	1	1	-
Release Indicator	String(Octet)	1	0-1	-
Reserved	String(Octet)	1	0-1	-
Segment Terminator	String(Octet)	1	1	-

Table 3-24. OM Attributes of a Service String Advice

Further details on the meaning of this class and its attributes can be found in the EDIFACT standard (see **Referenced Documents**).

3.38 Syntax Identifier

An instance of class **Syntax Identifier** indicates the syntax used in an EDI Interchange and is semantically identical to the "Syntax identifier" of the EDIFACT UNB segment.

An instance of this class has the attributes of its superclass - *Object* - and additionally the attributes listed below.

OM Attribute	Value Syntax	Value Length	Value Number	Value Initially
Syntax Identifier String	String(Teletex)	1-4	1	-
Syntax Version	String(Printable)	1-5	1	-

Table 3-25. OM Attributes of a Syntax Identifier

Further details on the meaning of this class and its attributes can be found in the EDIFACT standard (see **Referenced Documents**).

Declaration Summary

This section lists the declarations that define the C representation of the EDI Package. The declarations for classes that form part of the EDI Package closure but are defined in other specifications are not repeated here in order to avoid inconsistencies that may arise with repeated definitions. Readers are referred to the indicated specifications for the definitions associated with the following classes (and their closures): Content, Delivery Envelope, Externally Defined Body Part, Integrity Check Basis, OR Name (see the **API to Electronic Mail (X.400) CAE Specification**) and Object (see the **API to OSI Object Management (XOM) CAE Specification**).

The declarations, as assembled here, constitute the contents of a header file `<xedip.h>` to be made accessible to client programmers. The symbols that the declarations define are the only EDI symbols that the service makes visible to the client.

```

/* EDI SYMBOLIC CONSTANTS */

/* Class */

#define OMP_O_EDIC_AUTHORIZATION_INFO      "\x56\x06\x01\x02\x07\x01\x00"
#define OMP_O_EDIC_CROSS_REF_INFO         "\x56\x06\x01\x02\x07\x01\x01"
#define OMP_O_EDIC_EDIPAPP_SECUR_ELTS     "\x56\x06\x01\x02\x07\x01\x02"
#define OMP_O_EDIC_EDIBODYPART            "\x56\x06\x01\x02\x07\x01\x03"
#define OMP_O_EDIC_EDIEXTENSION           "\x56\x06\x01\x02\x07\x01\x04"
#define OMP_O_EDIC_EDIHEADING              "\x56\x06\x01\x02\x07\x01\x05"
#define OMP_O_EDIC_EDINOTIF                "\x56\x06\x01\x02\x07\x01\x06"
#define OMP_O_EDIC_EDINOTIFREQUESTS       "\x56\x06\x01\x02\x07\x01\x07"
#define OMP_O_EDIC_EDIM                    "\x56\x06\x01\x02\x07\x01\x08"
#define OMP_O_EDIC_EDIMBODYPART            "\x56\x06\x01\x02\x07\x01\x09"
#define OMP_O_EDIC_EDIMEXDFBODYPART       "\x56\x06\x01\x02\x07\x01\x0A"
#define OMP_O_EDIC_EDIMID                   "\x56\x06\x01\x02\x07\x01\x0B"
#define OMP_O_EDIC_EDINRECEIVER            "\x56\x06\x01\x02\x07\x01\x0C"
#define OMP_O_EDIC_FNPDAU_REASON_CODE      "\x56\x06\x01\x02\x07\x01\x0D"
#define OMP_O_EDIC_FN_REASON_CODE          "\x56\x06\x01\x02\x07\x01\x0E"
#define OMP_O_EDIC_FNUAMS_REASON_CODE      "\x56\x06\x01\x02\x07\x01\x0F"
#define OMP_O_EDIC_FNUSER_REASON_CODE      "\x56\x06\x01\x02\x07\x01\x10"
#define OMP_O_EDIC_FORWARD_NOTIF           "\x56\x06\x01\x02\x07\x01\x11"
#define OMP_O_EDIC_INTERSENDER_RECIP       "\x56\x06\x01\x02\x07\x01\x12"
#define OMP_O_EDIC_NEGATIVE_NOTIF          "\x56\x06\x01\x02\x07\x01\x13"
#define OMP_O_EDIC_NNPDAU_REASON_CODE      "\x56\x06\x01\x02\x07\x01\x14"
#define OMP_O_EDIC_NN_REASON_CODE          "\x56\x06\x01\x02\x07\x01\x15"
#define OMP_O_EDIC_NNUAMS_REASON_CODE      "\x56\x06\x01\x02\x07\x01\x16"
#define OMP_O_EDIC_NNUSER_REASON_CODE      "\x56\x06\x01\x02\x07\x01\x17"
#define OMP_O_EDIC_POSITIVE_NOTIF          "\x56\x06\x01\x02\x07\x01\x18"
#define OMP_O_EDIC_PRIMARYBODYPART         "\x56\x06\x01\x02\x07\x01\x19"
#define OMP_O_EDIC_REASON_CODE              "\x56\x06\x01\x02\x07\x01\x1A"
#define OMP_O_EDIC_RECIPREFERENCE          "\x56\x06\x01\x02\x07\x01\x1B"
#define OMP_O_EDIC_RECIPS                    "\x56\x06\x01\x02\x07\x01\x1C"
#define OMP_O_EDIC_SECUR_ELEMENTS           "\x56\x06\x01\x02\x07\x01\x1D"
#define OMP_O_EDIC_SERVSTRING_ADVICE        "\x56\x06\x01\x02\x07\x01\x1E"
#define OMP_O_EDIC_SYNTAX_ID                "\x56\x06\x01\x02\x07\x01\x1F"

/* Enumeration */

/* EDIN Initiator */
#define EDI_EI_INTERNAL_UA                  ( (OM_enumeration) 0 )
#define EDI_EI_INTERNAL_MS                  ( (OM_enumeration) 1 )
#define EDI_EI_EX_UA                        ( (OM_enumeration) 2 )

/* Integer */

/* Content Type */
/* Note: this constant may be used as a value of the Content Type */
/* attribute of the Communicate class defined in the X.400 API */

#define EDI_CONTENT_TYPE                    ( (OM_integer) 35 )

```

Declaration Summary

```
/* FN PDAU Basic Code */
#define EDI_FPB_UNSPECIFIED ( (OM_integer) 0 )
#define EDI_FPB_FORWARD_FOR_PD_AND_REND ( (OM_integer) 1 )

/* FN UAMS Basic Code */
#define EDI_FUMB_UNSPECIFIED ( (OM_integer) 0 )
#define EDI_FUMB_ONWARD_ROUTING ( (OM_integer) 1 )
#define EDI_FUMB_RECIP_UNKNOWN ( (OM_integer) 2 )
#define EDI_FUMB_ORIG_UNKNOWN ( (OM_integer) 3 )
#define EDI_FUMB_FORWARD_BY_ED MS ( (OM_integer) 4 )

/* FN UAMS Diagnostic Code */
#define EDI_FUMD_RECIP_NAME_CHANGED ( (OM_integer) 1 )
#define EDI_FUMD_RECIP_NAME_DELETED ( (OM_integer) 2 )

/* FN User Basic Code */
#define EDI_FUB_UNSPECIFIED ( (OM_integer) 0 )
#define EDI_FUB_FORWARD_FOR_ARCHIVING ( (OM_integer) 1 )
#define EDI_FUB_FORWARD_FOR_INFO ( (OM_integer) 2 )
#define EDI_FUB_FORWARD_FOR_ADD_INFO ( (OM_integer) 3 )
#define EDI_FUB_SUB_UNCHANGED ( (OM_integer) 4 )
#define EDI_FUB_HEADING_FIELD_NOT_SUPP ( (OM_integer) 5 )
#define EDI_FUB_BODYPART_TYPE_NOT_SUPP ( (OM_integer) 6 )
#define EDI_FUB_MSG_TYPE_NOT_SUPP ( (OM_integer) 7 )
#define EDI_FUB_SYNTAX_ID_NOT_SUPP ( (OM_integer) 8 )
#define EDI_FUB_INTER_SENDER_UNKNOWN ( (OM_integer) 9 )
#define EDI_FUB_USER_DEFINED_REASON ( (OM_integer) 10 )

/* NN PDAU Basic Code */
#define EDI_NPB_UNSPECIFIED ( (OM_integer) 0 )
#define EDI_NPB_UNDELIV_MAIL ( (OM_integer) 1 )
#define EDI_NPB_P_REND_NOT_PERFORMED ( (OM_integer) 2 )

/* NN NP Diagnostic Code */
#define EDI_NPD_P_REND_ATTS_NOT_SUPP ( (OM_integer) 31 )
#define EDI_NPD_PD_ADDRESS_INCORRECT ( (OM_integer) 32 )
#define EDI_NPD_PD_OFFICE_I_OR_INVALID ( (OM_integer) 33 )
#define EDI_NPD_PD_ADDRESS_INC ( (OM_integer) 34 )
#define EDI_NPD_RECIP_UNKNOWN ( (OM_integer) 35 )
#define EDI_NPD_RECIP_DECEASED ( (OM_integer) 36 )
#define EDI_NPD_ORGANIZATION_EXPIRED ( (OM_integer) 37 )
#define EDI_NPD_RECIP_REFUSED_TO_ACCEPT ( (OM_integer) 38 )
#define EDI_NPD_RECIP_DID_NOT_CLAIM ( (OM_integer) 39 )
#define EDI_NPD_RECIP_CHANGED_ADDRESS_P ( (OM_integer) 40 )
#define EDI_NPD_RECIP_CHANGED_ADDRESS_T ( (OM_integer) 41 )
#define EDI_NPD_RECIP_CHANGED_T_ADDRESS ( (OM_integer) 42 )
#define EDI_NPD_NEW_ADDRESS_UNKNOWN ( (OM_integer) 43 )
#define EDI_NPD_RECIP_NOT_WANT_FORWARD ( (OM_integer) 44 )
#define EDI_NPD_ORIG_PROHIBITED_FORWARD ( (OM_integer) 45 )
```

```

/* NN UAMS Basic Code */
#define EDI_NUMB_UNSPECIFIED ( (OM_integer) 0 )
#define EDI_NUMB_CANNOT_DELIV_TO_USER ( (OM_integer) 1 )
#define EDI_NUMB_DELIV_TIMEOUT ( (OM_integer) 2 )
#define EDI_NUMB_MSG_DISCARDED ( (OM_integer) 3 )
#define EDI_NUMB_SUB_TERMINATED ( (OM_integer) 4 )
#define EDI_NUMB_FORWARDING_ERROR ( (OM_integer) 5 )
#define EDI_NUMB_SECUR_ERROR ( (OM_integer) 6 )

/* NN NUM Diagnostic Code */
#define EDI_NUMD_PROTOCOL_VIOLATION ( (OM_integer) 1 )
#define EDI_NUMD_EDIM_ORIG_UNKNOWN ( (OM_integer) 2 )
#define EDI_NUMD_EDIM_RECIP_UNKNOWN ( (OM_integer) 3 )
#define EDI_NUMD_EDIM_RECIP_AMBIGUOUS ( (OM_integer) 4 )
#define EDI_NUMD_ACT_REQUEST_NOT_SUPP ( (OM_integer) 5 )
#define EDI_NUMD_EDIM_EXPIRED ( (OM_integer) 6 )
#define EDI_NUMD_EDIM_OBSOLETED ( (OM_integer) 7 )
#define EDI_NUMD_DUPLICATE_EDIM ( (OM_integer) 8 )
#define EDI_NUMD_UNSUPP_EXTENSION ( (OM_integer) 9 )
#define EDI_NUMD_INC_COPY_REJECTED ( (OM_integer) 10 )
#define EDI_NUMD_EDIM_TOO_LARGE_FOR_APP ( (OM_integer) 11 )
#define EDI_NUMD_FORWARD_EDIM_NOT_DELIV ( (OM_integer) 12 )
#define EDI_NUMD_FORWARD_EDIM_D_TIME ( (OM_integer) 13 )
#define EDI_NUMD_FORWARD_LOOP_DETECTED ( (OM_integer) 14 )
#define EDI_NUMD_UNABLE_TO_ACCEPT_RESP ( (OM_integer) 15 )
#define EDI_NUMD_INTER_SENDER_UNKNOWN ( (OM_integer) 16 )
#define EDI_NUMD_INTER_RECIP_UNKNOWN ( (OM_integer) 17 )
#define EDI_NUMD_INVALID_HEADING_FIELD ( (OM_integer) 18 )
#define EDI_NUMD_INVALID_BODYPART_TYPE ( (OM_integer) 19 )
#define EDI_NUMD_INVALID_MSG_TYPE ( (OM_integer) 20 )
#define EDI_NUMD_INVALID_SYNTAX_ID ( (OM_integer) 21 )
#define EDI_NUMD_MSG_INTEG_FAIL ( (OM_integer) 22 )
#define EDI_NUMD_FORWARD_MSG_INTEG_FAIL ( (OM_integer) 23 )
#define EDI_NUMD_UNSUPP_ALGORITHM ( (OM_integer) 24 )
#define EDI_NUMD_DECRYPTION_FAILED ( (OM_integer) 25 )
#define EDI_NUMD_TOKEN_ERROR ( (OM_integer) 26 )
#define EDI_NUMD_UNABLE_TO_SIGN_NOTIF ( (OM_integer) 27 )
#define EDI_NUMD_UNABLE_TO_SIGN_MSG_R ( (OM_integer) 28 )
#define EDI_NUMD_AUTHENTICATION_FAIL ( (OM_integer) 29 )
#define EDI_NUMD_SECUR_CONTEXT_FAIL ( (OM_integer) 30 )
#define EDI_NUMD_MSG_SEQUENCE_FAIL ( (OM_integer) 31 )
#define EDI_NUMD_MSG_SECUR_LABEL_FAIL ( (OM_integer) 32 )
#define EDI_NUMD_REPUDIATION_FAIL ( (OM_integer) 33 )
#define EDI_NUMD_PROOF_OF_FAIL ( (OM_integer) 34 )

```


Declaration Summary

```
/* NN User Basic Code */
#define EDI_NUB_UNSPECIFIED ( (OM_integer) 0 )
#define EDI_NUB_SYNTAX_ERROR ( (OM_integer) 1 )
#define EDI_NUB_INTER_SENDER_UNKNOWN ( (OM_integer) 2 )
#define EDI_NUB_INTER_RECIP_UNKNOWN ( (OM_integer) 3 )
#define EDI_NUB_INVALID_HEADING_FIELD ( (OM_integer) 4 )
#define EDI_NUB_INVALID_BODYPART_TYPE ( (OM_integer) 5 )
#define EDI_NUB_INVALID_MSG_TYPE ( (OM_integer) 6 )
#define EDI_NUB_FUNCT_GROUP_NOT_SUPP ( (OM_integer) 7 )
#define EDI_NUB_SUB_TERMINATED ( (OM_integer) 8 )
#define EDI_NUB_NO_BILATERAL_AGREEMENT ( (OM_integer) 9 )
#define EDI_NUB_USER_DEFINED_REASON ( (OM_integer) 10 )

/* Object Identifier */

/* EDI Package */
#define OMP_O_EDI_PACKAGE "\x56\x06\x01\x02\x07\x01"

/* EDI Body Part Type Attribute */
#define OMP_O_EDI_BPO_EDIFACT_ISO646 "\x56\x06\x07\x0B\x00"
#define OMP_O_EDI_BPO_EDIFACT_T61 "\x56\x06\x07\x0B\x01"
#define OMP_O_EDI_BPO_EDIFACT_OCTET "\x56\x06\x07\x0B\x02"
#define OMP_O_EDI_BPO_ANSIX12_ISO646 "\x56\x06\x07\x0B\x03"
#define OMP_O_EDI_BPO_ANSIX12_T61 "\x56\x06\x07\x0B\x04"
#define OMP_O_EDI_BPO_ANSIX12_OCTET "\x56\x06\x07\x0B\x05"
#define OMP_O_EDI_BPO_ANSIX12_EBCDIC "\x56\x06\x07\x0B\x06"
#define OMP_O_EDI_BPO_UNTDI_ISO646 "\x56\x06\x07\x0B\x07"
#define OMP_O_EDI_BPO_UNTDI_T61 "\x56\x06\x07\x0B\x08"
#define OMP_O_EDI_BPO_UNTDI_OCTET "\x56\x06\x07\x0B\x09"
#define OMP_O_EDI_BPO_PRIVATE_OCTET "\x56\x06\x07\x0B\x0A"
#define OMP_O_EDI_BPO_UNDEFINED_OCTET "\x56\x06\x07\x0B\x0B"

/* Action Request Attribute */
#define OMP_O_EDI_ARO_FOR_ACTION "\x56\x06\x07\x0D\x00"
#define OMP_O_EDI_ARO_COPY "\x56\x06\x07\x0D\x01"

/* Type */

#define EDI_ACKNOWLEDGEMENT_REQUEST ( (OM_type) 1000 )
#define EDI_ACT_REQUEST ( (OM_type) 1001 )
#define EDI_ADD_BODYPARTS ( (OM_type) 1002 )
#define EDI_APP_CROSS_REFERENCE ( (OM_type) 1003 )
#define EDI_APP_REFERENCE ( (OM_type) 1004 )
#define EDI_APP_SECUR_ELEMENT ( (OM_type) 1005 )
#define EDI_APP_SECUR_EXTENSIONS ( (OM_type) 1006 )
#define EDI_AUTHORIZATION_INFO ( (OM_type) 1007 )
#define EDI_AUTHORIZATION_INFO_Q ( (OM_type) 1008 )
#define EDI_BASIC_CODE ( (OM_type) 1009 )
#define EDI_BODY ( (OM_type) 1010 )
#define EDI_BODYPART_REFERENCE ( (OM_type) 1011 )
#define EDI_COMMS_AGREEMENT_ID ( (OM_type) 1012 )
#define EDI_COMPONENT_DATA_ELEMENT_SEP ( (OM_type) 1013 )
#define EDI_CRITICALITY ( (OM_type) 1014 )
#define EDI_CROSS_REFERENCING_INFO ( (OM_type) 1015 )
#define EDI_DATA_ELEMENT_SEPARATOR ( (OM_type) 1016 )
#define EDI_DATE_AND_TIME_OF_PREP ( (OM_type) 1017 )
#define EDI_DECIMAL_NOTATION ( (OM_type) 1018 )
#define EDI_DELIV_ENVELOPE ( (OM_type) 1019 )
```

```

#define EDI_DIAGNOSTIC ( (OM_type) 1020 )
#define EDI_EDI_APP_SECUR_ELEMENTS ( (OM_type) 1021 )
#define EDI_EDI_BODYPART_TYPE ( (OM_type) 1022 )
#define EDI_EDI_MSG_TYPE ( (OM_type) 1023 )
#define EDI_EDI_NOTIF_REQUESTS ( (OM_type) 1024 )
#define EDI_EDI_SUPPLEMENTARY_INFO ( (OM_type) 1025 )
#define EDI_EDIN_INITIATOR ( (OM_type) 1026 )
#define EDI_EDIN_ORIG ( (OM_type) 1027 )
#define EDI_EDIN_RECEIVER ( (OM_type) 1028 )
#define EDI_EDIN_RECEIVER_NAME ( (OM_type) 1029 )
#define EDI_ENCRYPTED_PRIMARY_BODYPART ( (OM_type) 1030 )
#define EDI_EXPIRY_TIME ( (OM_type) 1031 )
#define EDI_EXTENSION_TYPE ( (OM_type) 1032 )
#define EDI_EXTENSION_VALUE ( (OM_type) 1033 )
#define EDI_EX_BODYPART ( (OM_type) 1034 )
#define EDI_FIRST_RECIP ( (OM_type) 1035 )
#define EDI_FN_REASON_CODE ( (OM_type) 1036 )
#define EDI_FN_SECUR_CHECK ( (OM_type) 1037 )
#define EDI_FORWARD_NOTIF ( (OM_type) 1038 )
#define EDI_FORWARD_TO ( (OM_type) 1039 )
#define EDI_HEADING ( (OM_type) 1040 )
#define EDI_HEADING_EXTENSIONS ( (OM_type) 1041 )
#define EDI_IDENTIFICATION_CODE ( (OM_type) 1042 )
#define EDI_IDENTIFICATION_CODE_Q ( (OM_type) 1043 )
#define EDI_INC_COPY ( (OM_type) 1044 )
#define EDI_INTER_CONTROL_REFERENCE ( (OM_type) 1045 )
#define EDI_INTER_RECIP ( (OM_type) 1046 )
#define EDI_INTER_SENDER ( (OM_type) 1047 )
#define EDI_MSG_REFERENCE ( (OM_type) 1048 )
#define EDI_NEGATIVE_NOTIF ( (OM_type) 1049 )
#define EDI_NN_REASON_CODE ( (OM_type) 1050 )
#define EDI_NON_REPUD_OF_NOTIF ( (OM_type) 1051 )
#define EDI_NON_REPUD_OF_RECEPTION ( (OM_type) 1052 )
#define EDI_NOTIF_EXTENSIONS ( (OM_type) 1053 )
#define EDI_NOTIF_TIME ( (OM_type) 1054 )
#define EDI_OBSOLETEED_EDIMS ( (OM_type) 1055 )
#define EDI_ORIG ( (OM_type) 1056 )
#define EDI_ORIGINAL_CONTENT ( (OM_type) 1057 )
#define EDI_ORIGINAL_CONTENT_INTEG_CHK ( (OM_type) 1058 )
#define EDI_ORIGINAL_EDIM_ID ( (OM_type) 1059 )
#define EDI_POSITIVE_NOTIF ( (OM_type) 1060 )
#define EDI_PRIMARY_BODYPART ( (OM_type) 1061 )
#define EDI_PROCESSING_PRIORITY_CODE ( (OM_type) 1062 )
#define EDI_PROOF_OF_NOTIFICATION ( (OM_type) 1063 )
#define EDI_PROOF_OF_RECEPTION ( (OM_type) 1064 )
#define EDI_RECIP ( (OM_type) 1065 )
#define EDI_RECIP_EXTENSIONS ( (OM_type) 1066 )
#define EDI_RECIP_REFERENCE ( (OM_type) 1067 )
#define EDI_RECIP_REFERENCE_Q ( (OM_type) 1068 )
#define EDI_RECIPS ( (OM_type) 1069 )
#define EDI_RELATED_EDI_MSGS ( (OM_type) 1070 )
#define EDI_RELATED_NON_EDI_MSGS ( (OM_type) 1071 )
#define EDI_RELEASE_INDICATOR ( (OM_type) 1072 )
#define EDI_RESERVED ( (OM_type) 1073 )
#define EDI_RESPONS_FORWARD ( (OM_type) 1074 )
#define EDI_RESPONS_PASSING_ALLOWED ( (OM_type) 1075 )
#define EDI_ROUTING_ADDRESS ( (OM_type) 1076 )
#define EDI_SECUR_ELEMENTS ( (OM_type) 1077 )

```

Declaration Summary

```
#define EDI_SEGMENT_TERMINATOR          ( (OM_type) 1078 )
#define EDI_SERVICE_STRING_ADVICE      ( (OM_type) 1079 )
#define EDI_SUBJECT_EDIM                ( (OM_type) 1080 )
#define EDI_SUPPLEMENTARY_INFO         ( (OM_type) 1081 )
#define EDI_SYNTAX_ID                  ( (OM_type) 1082 )
#define EDI_SYNTAX_ID_STRING           ( (OM_type) 1083 )
#define EDI_SYNTAX_VERSION              ( (OM_type) 1084 )
#define EDI_TEST_INDICATOR             ( (OM_type) 1085 )
#define EDI_THIS_EDIM                  ( (OM_type) 1086 )
#define EDI_USER                        ( (OM_type) 1087 )
#define EDI_USER_RELATIVE_ID           ( (OM_type) 1088 )

/* Value Length */

#define EDI_VL_APP_REFERENCE            ( (OM_value_length) 14 )
#define EDI_VL_APP_SECUR_ELEMENT        ( (OM_value_length) 8191 )
#define EDI_VL_AUTHORIZATION_INFO       ( (OM_value_length) 10 )
#define EDI_VL_AUTHORIZATION_INFO_Q    ( (OM_value_length) 2 )
#define EDI_VL_COMMS_AGREEMENT_ID      ( (OM_value_length) 35 )
#define EDI_VL_EDI_MSG_TYPE             ( (OM_value_length) 6 )
#define EDI_VL_IDENTIFICATION_CODE     ( (OM_value_length) 35 )
#define EDI_VL_IDENTIFICATION_CODE_Q   ( (OM_value_length) 4 )
#define EDI_VL_INTER_CONTROL_REFERENCE  ( (OM_value_length) 14 )
#define EDI_VL_RECIP_REFERENCE          ( (OM_value_length) 14 )
#define EDI_VL_RECIP_REFERENCE_Q        ( (OM_value_length) 2 )
#define EDI_VL_ROUTING_ADDRESS          ( (OM_value_length) 14 )
#define EDI_VL_SYNTAX_ID_STRING         ( (OM_value_length) 4 )
#define EDI_VL_SYNTAX_VERSION           ( (OM_value_length) 5 )
#define EDI_VL_USER_RELATIVE_ID         ( (OM_value_length) 64 )
```


Glossary

ANSI

American National Standards Institute

ANSIX12

ANSI committee X12

API

application program interface

ASN.1

Abstract Syntax Notation One

AU

access unit

BER

Basic Encoding Rules

CCITT

International Telegraph and Telephone Consultative Committee

DL

distribution list

EDI

electronic data interchange

EDIM

EDI message

EDIMG

EDI messaging

EDIN

EDI notification

EDIFACT

Electronic Data Interchange for Administration, Commerce and Transport

EIT

encoded information type

FN

forwarded notification

IM

Interpersonal Messaging

IPM

interpersonal message

IPN

interpersonal notification

ISO

International Organisation for Standardisation

MA

Message Access

MH

Message Handling

MHS

message handling system

MOTIS

Message-oriented Text Interchange System

MS

message store

MT

Message Transfer

MTA

message transfer agent

MTS

message transfer system

NN

negative notification

OR

originator/recipient

OM

Object Management

OSI

Open Systems Interconnection

PDAU

physical delivery access unit

Pedi

Program interface to the functionality of EDI

PN

positive notification

UA

EDI user agent

UAMS

user agent or message store

UNTDI

United Nations / Trade Data Interchange

Index

<i>Pedi</i>	62	Integrity Check Basis	33
abstract class	5	Interchange Sender-Recipient.....	34
ANSI.....	61	IPM.....	17, 61
ANSIX12.....	61	IPN.....	61
API.....	61	ISO	61
ASN.1.....	61	MA	62
AU.....	61	MA Interface.....	2
Authorisation Information	7	MH.....	62
Authorization Information.....	8	MHS.....	1, 62
Authorization Information Qualifier.....	8	MOTIS.....	62
BER	61	MS.....	3, 62
CCITT.....	61	MT.....	62
class hierarchy.....	5	MT Interface.....	2
Component Data Element Separator	51	MTA.....	62
concrete class	5	MTS	3, 62
Content	9	Negative Notification	35
Cross referencing Information.....	10	NN	62
Data Element Separator	51	NN PDAU Reason Code.....	36
Decimal Notation	51	NN Reason Code	37
Declaration Summary.....	53	NN UAMS Reason Code.....	38
Delivery Envelope	11	NN User Reason Code	40
DL.....	15, 48, 61	Object	41
EDI.....	61	OM.....	62
EDI Application Security Elements	12	OR.....	62
EDI Body Part.....	13	OR Name.....	42
EDI Extension.....	14	OSI	62
EDI Heading	15	package closure.....	5
EDI Notification.....	19	PDAU	62
EDI Notification Requests	21	PN	62
EDIFACT	61	Positive Notification	43
EDIM.....	3, 23, 61	Primary Body Part	44
EDIM Body Part	24	Reason Code.....	45
EDIM Externally Defined Body Part.....	25	Recipient Reference	46-47
EDIM Identifier	26	Recipient Reference Qualifier	47
EDIMG.....	3, 61	Recipients	47
EDIN.....	3, 61	Release Indicator.....	51
EDIN Receiver.....	27	Reserved	51
EIT.....	61	Security Elements	49
Externally Defined Body Part.....	27	Segment Terminator.....	51
FN	61	Service String Advice.....	50
FN PDAU Reason Code	28	Syntax Identifier.....	51
FN Reason Code	29	Syntax Identifier String.....	52
FN UAMS Reason Code.....	30	Syntax Version.....	52
FN User Reason Code.....	31	UA.....	62
Forwarded Notification.....	32	UAMS.....	62
IM.....	61	UNTDI.....	62

xedip.h53