The ATN SARPs

Sub-Volume III

Ground-Ground Applications

Editors' Draft

Please note that this is the final editor's draft circulated within the ATNP. This text will be passed to ICAO for publication. However, it should be noted that this text in no way replaces the ICAO version nor can it be considered of equal status. The official definitive version is that published in hardcopy by ICAO and all claims of compliance must be made against that version.

This PDF version has been prepared for the ATNP Working Groups by FANS Information Services Ltd - http://www.fans-is.com

Please check our Web Site regularly for information on updates to the draft SARPs

Errata and Disclaimer

Please note that this document was prepared from a number of separate files prepared by different editors. The initial page numbers in these files are not completely synchronised, and we have made no attempt to change this, in order to avoid problems with references from Working Papers. You may therefore find some overlap between pages numbers, when those pages came from different files.

The preparation of this document has been on a "best efforts" basis and no warrantee is offered as to its correctness.

FOREWORD

The material contained in this document was originally developed as the detailed part of the first set of Standards and Recommended Practices (SARPs) for the aeronautical telecommunication network (ATN) which has commonly been referred to as the CNS/ATM-1 Package. It was intended to make the material an appendix to the new Chapter 3 of Annex 10, Volume III, Part I, containing broad, general, stable and mostly regulatory-type provisions (the core part of new ATN SARPs).

In December 1997, the Air Navigation Commission (ANC), while conducting the final review of draft ATN SARPs, noted that actual implementation and operational experience was yet to be gained by the international civil aviation community. In this regard, the ANC agreed that the detailed part of ATN SARPs should be published as an ICAO manual, while retaining its SARPs-style language. The ANC will review the status of the document, in its entirety or in parts, after sufficient implementation and operational experience has been gained and the requirements for further standardization, in the interests of safety, regularity and efficiency of international civil aviation have been better ascertained.

This document consists of five Sub-Volumes:

Sub-Volume I — Introduction and System Level Requirements
Sub-Volume II — Air-Ground Applications
Sub-Volume III — Ground-Ground Applications
Sub-Volume IV — Upper Layer Communications Service (ULCS)
Sub-Volume V — Internet Communications Service (ICS)

Provisions contained in Sub-Volumes II, III, IV and V have been developed in accordance with system requirements specified in Sub-Volume I.

TABLE OF CONTENTS

SUB-VOLUME III. GROUND-GROUND APPLICATIONS

3.1 ATS MESSAGE HANDLING SERVICES (ATSMHS)

3.1.1 INTRODUCTION

The ATS (Air Traffic Services) Message Handling Services (ATSMHS) applications allow ATS Messages to be exchanged between service users. The ATS Message Handling Services are the ATS Message Service and the ATN Pass-Through Service.

Note 1.— These ATS Message Handling Services aim at providing generic message services over the Aeronautical Telecommunication Network (ATN) Internet. They may in turn be used as a communication system by user-applications communicating over the ATN. This may be achieved e.g. by means of application program interfaces to either the ATS Message Service or to the ATN Pass-Through Service.

Note 2.— ATS Message Service

a) The ATS Message Service is provided by the implementation over the ATN Internet Communication Services of the Message Handling Systems specified in ISO/IEC (International Organization for Standardization/ International Electrotechnical Commission) 10021 and CCITT (Consultative Committee of International Telegraph and Telephone) or ITU-T (International Telecommunication Union - Telecommunications Standards) X.400, and complemented with the additional requirements specified in 3.1. The two sets of documents, the ISO/IEC MOTIS (Message-Oriented Text Interchange System) International Standards and the CCITT X.400 Series of Recommendations (1988 or later) are in principle aligned to each other. However there are a small number of differences. In 3.1 reference is made to the relevant ISO International Standards, and International Standardized Profiles (ISP) where applicable. Where necessary, e.g. for reasons of interworking or to point out differences, reference is also made to the relevant X.400 Recommendations.

b) Two levels of service are intended to be defined within the ATS Message Service:

i) the Basic ATS Message Service.

ii) the Extended ATS Message Service.

c) This specification of the ATS Message Service supports only the Basic ATS Message Service. The Extended ATS Message Service could be incorporated in future packages.

Note 3.— The ATN Pass-Through Service is the ATS Message Handling Service offered over the ATN Internet Communication Services by the use of the Dialogue Service and of the associated ATN upper layer architecture to exchange AFTN (Aeronautical Fixed Telecommunication Network) Messages formatted in IA-5 (International Alphabet No 5) in compliance with the provisions of Annex 10, Volume II.

Note 4.— End systems performing ATS Message Handling Services

a) Four types of ATN End Systems are defined in 3.1:

1) an ATS Message Server,

2) an ATS Message User Agent,

3) an AFTN/AMHS Gateway (Aeronautical Fixed Telecommunication Network / ATS Message Handling System), and

4) an AFTN/ATN Type A Gateway.

b) Connections may be established over the Internet Communications Service between any pair constituted of these ATN End Systems and listed in Table 3.1.1-1. Although included in Table 3.1.1-1, the communication between an ATS Message Server and an ATS Message User Agent is not specified in 3.1.

Table 3.1.1-1. Communications between ATN End Systems implementing ATS Message Handling Services

ATN End System 1	*ATN End System 2*
ATS Message Server	*ATS Message Server*
ATS Message Server	*AFTN/AMHS Gateway*
ATS Message Server	*ATS Message User Agent*
AFTN/AMHS Gateway	*AFTN/AMHS Gateway*
AFTN/ATN Type A Gateway	*AFTN/ATN Type A Gateway*

Note 5.— Structure of 3.1

a) 3.1.1: INTRODUCTION contains the purpose and structure, and a summary of the functionalities offered by the ATS Message Handling Services.

b) 3.1.2: ATS MESSAGE SERVICE contains three sections as follows:

1) 3.1.2.1: System Level Provisions, provides a high level specification of the application and of the environment in which it operates;

2) *3.1.2.2: ATS Message Service Specification, provides the detailed specification of the service and protocol requirements for each type of ATN End System (ATS Message User Agent and ATS Message Server) implementing the ATS Message Service;*

3) *3.1.2.3: AFTN/AMHS Gateway Specification, provides the detailed specification of an AFTN/AMHS Gateway and of the related functional requirements such as conversion.*

c) *3.1.3: ATN PASS-THROUGH SERVICE contains three sections as follows:*

1) *3.1.3.1: System Level Provisions, provides a high level specification of the application and of the environment in which it operates;*

2) *3.1.3.2: ATN Pass-Through Service Specification, provides the detailed specification of the protocol requirements between two AFTN/ATN Type A Gateways implementing the ATN Pass-Through Service;*

3) *3.1.3.3: AFTN/ATN Type A Gateway Specification, provides the detailed specification of an AFTN/ATN Type A Gateway and of the related functional requirements.*

Note 6.— The following terminology applies in 3.1:

a) ***AFTN acknowledgement message:*** *an AFTN service message acknowledging the receipt of an AFTN message which priority indicator has the value "SS".*

b) ***direct AMHS user****: an ATS Message Service user who engages in the ATS Message Service at an ATS Message User Agent. A direct AMHS user may belong to two subgroups as follows:*

1) *human users who interact with the ATS Message Service by means of an ATS Message User Agent connected to an ATS Message Server; and*

2) *host users which are computer applications running on ATN end systems and interacting with the ATS Message Service by means of application programme interfaces.*

c) ***indirect AMHS user:*** *an ATS Message Service user at an AFTN station using an AFTN/AMHS Gateway to communicate with other ATS Message Service users.*

d) ***subject AFTN message:*** *an AFTN message which causes an AFTN service message or an AMHS report to be generated.*

e) ***subject AMHS message:*** *an AMHS message which causes an AFTN service message or an AMHS report to be generated.*

f) ***subject IPM:*** *the IPM which is the content of an AMHS message and which causes an AMHS Receipt Notification to be generated.*

g) ***unknown address AFTN service message:*** *an AFTN service message requesting correction by the originator of a message received with an unknown addressee indicator.*

Note 7.— The classifications defined in the ISPs apply for expressing conformance requirements - i.e. static capability - in 3.1. The ISP classifications refine the ISO/IEC 9646-7 classification to include different levels of mandatory support, depending on the level of functionality to be supported by the considered Message Handling System. These classifications include the following elements, of which the complete definition may be found in each referenced ISP:

a) ***mandatory full support*** *(M).*

b) ***mandatory minimal support*** *(M-).*

c) ***mandatory O/R name minimal support*** *(M1) (see ISO/IEC ISP 12062-2).*

d) ***optional support*** *(O).*

e) ***conditional support*** *(C).*

f) ***out of scope*** *(I).*

g) ***not applicable*** *(-).*

Note 8.— The following classification applies for expressing dynamic behaviour requirements - i.e. the action performed by the ATN end system - related to parameters or elements in the Profile Requirement Lists (PRLs) included in 3.1.2.3, for the specification of the AFTN/AMHS Gateway:

a) ***generated*** *(G): used to describe the generation of an AMHS or AFTN information object. It means that the element is generated by the AFTN/AMHS Gateway, and that its value does not depend on the value of an element of the information object received by the AFTN/AMHS Gateway which caused the current generation of an information object, but that the value of the element is based on parameters related to the AFTN/AMHS Gateway itself or takes a pre-determined value. If an element comprises several components, then the element is classified as generated if at least one of its components is generated, and the others are either generated or excluded;*

b) ***optionally generated*** *(G1): used with the same meaning as "generated", with the exception that the generation of the element is optional, the decision being a matter of policy local to the Management Domain operating the AFTN/AMHS Gateway;*

c) ***conditionally generated*** *(G2): used only to describe the generation of an AMHS report or RN (Receipt Notification) element. It means, for a report generation, that the element is generated in the report or RN based on some condition related to the subject AMHS message being true. If the element is generated, it takes a value*

derived from elements present in the received AMHS information object which caused the generation of the report or RN;

d) ***translated*** *(T): used to describe either the generation of an AMHS or AFTN information object or the use of a received information object. It means that the element is translated by the AFTN/AMHS Gateway, using a dependence relationship between the value of an element of the received information object and the value of the translated element in the generated information object. If an element comprises several components, then the element is classified as translated if at least one of its components is translated, and the others are either generated or excluded in generation, discarded or out of scope in reception;*

e) ***conditionally translated*** *(T1): used with the same meaning as "translated", with the exception that the translation of the element is subject to some condition being true, e.g. the presence of an optional element in the received information object;*

f) ***discarded*** *(D): used to describe the use of a received AMHS or AFTN information object. It means that the value of the element is not used by the Message Transfer and Control Unit when generating the elements of the information object converted from the received information object, and that the semantic information conveyed in the element is discarded during the process of conversion in the AFTN/AMHS Gateway. However the presence or value of the element may be used by the Message Transfer and Control Unit for purposes other than conversion, such as report generation and logging;*

g) ***excluded*** *(X): used to describe either the generation of an AMHS or AFTN information object or the use of a received information object. Upon generation of an information object, it means that the element is not used nor present in the generated information object. Upon reception of an AMHS information object, it means that the presence of the element causes rejection of the information object, and generation of an AMHS non-delivery report as appropriate;*

h) ***out of scope*** *or* ***not-applicable*** *(-): used to describe the use of a received information object, when the element is either a format element which cannot be processed in any way or an element which is not in the scope of the section, but which presence is included in the ISPICS (ISP Implementation Conformance Statement) serving as a basis for the mapping specification.*

Note 9.— Application Functionalities

a) *The Basic ATS Message Service meets the basic requirements of the Message Handling Systems Profiles published by ISO/IEC as ISPs (International Standardized Profiles), and it incorporates additional features to support the service offered by the AFTN. The Basic ATS Message Service is further specified in 3.1.2.2. This includes the specification of which ISPs apply in this context.*

b) *The ATN Pass-Through Service encapsulates and decapsulates AFTN messages at an AFTN/ATN type A Gateway, using the Dialogue Service and the associated*

upper layer protocol architecture. The ATN Pass-Through Service is further specified in 3.1.3.2.

Note 10.— Applicability

a) *The implementation of the ATS Message Service is mandatory for conformance with 3.1. However, as a matter of organisations' policy, interim conformance may be achieved with the implementation of the ATN Pass-Through Service. The choice to implement the ATN Pass-Through Service as an interim solution does not replace the requirement to implement the ATS Message Service at the earliest possible date.*

b) *The interoperability between the ATS Message service and the ATN Pass-Through Service is a local implementation matter, provided that such an implementation has an external behaviour identical to that of an AFTN/AMHS Gateway and of an AFTN/ATN Type A Gateway, as appropriate. The choice to implement the ATN Pass-Through Service implies the requirement to provide the interoperability facilities to the ATS Message Service implementations.*

3.1.2 ATS MESSAGE SERVICE

3.1.2.1 System level provisions

The ATS Message Service shall be implemented for conformance with 3.1.

3.1.2.1.1 ATS Message Service Users

3.1.2.1.1.1 Direct AMHS users shall use the Basic ATS Message Service at an ATS Message User Agent.

3.1.2.1.1.2 Indirect AMHS users shall use only that part of the Basic ATS Message Service which corresponds to AFTN functionalities, by using the interworking capability provided by an AFTN/AMHS Gateway as specified in 3.1.2.3.

3.1.2.1.2 AMHS Model

3.1.2.1.2.1 AMHS functional model

3.1.2.1.2.1.1 Model components

The systems comprising the AMHS shall themselves be comprised of the following functional objects, the general role of which is described in ISO/IEC 10021-2:

a) message transfer agent(s) (MTA),

b) user agent(s) (UA),

c) message store(s) (MS), and

d) access unit(s) (AU).

Note.— The ISO/IEC 10021 Elements of Service and Protocols used by these functional objects are specified in 3.1.2.2 and 3.1.2.3.

3.1.2.1.2.1.2 **ATS Message Server**

An ATS Message Server shall include a MTA and optionally one or several MSs, as specified in 3.1.2.2.2.

3.1.2.1.2.1.3 ATS Message User Agent

An ATS Message User Agent shall include a UA as specified in 3.1.2.2.1.

3.1.2.1.2.1.4 AFTN/AMHS Gateway

An AFTN/AMHS Gateway shall include a MTA, which is part of the ATN Component of the AFTN/AMHS Gateway, and an AU, as specified in 3.1.2.3.

Note.— The AU is the Message Transfer and Control Unit of the AFTN/AMHS Gateway.

3.1.2.1.2.2 AMHS information model

The following three categories of AMHS information objects shall be used:

a) messages;

b) probes; and

c) reports.

3.1.2.1.2.2.1 Messages

Note.— The provisions in 3.1.2 concerning ISO/IEC 10021 envelopes apply to Transfer Envelopes only.

In the Basic ATS Message Service, each AMHS message shall correspond unequivocally to an ATS Message.

3.1.2.1.2.2.2 Probes

Only direct AMHS users shall be able to submit AMHS probes.

3.1.2.1.2.2.3 Reports

AMHS reports shall be delivered only to direct AMHS users.

3.1.2.1.2.3 Security and management models

Recommendation.— *In the Basic ATS Message Service, security should be obtained by procedural means rather than by technical features inherent to the AMHS.*

Note 1.— In the Basic ATS Message Service, the security at each ATS Message Server or AFTN/AMHS Gateway is deemed a local issue to be addressed by the authority in charge of the system.

Note 2.— In the Basic ATS Message Service, management is limited to the logging provisions which are defined for the ATS Message User Agent, for the ATS Message Server and for the AFTN/AMHS Gateway. No provision is made for retrieval or exchange of this information, which is deemed a local issue to be addressed by the authority in charge of the system.

3.1.2.1.3 Organization of the AMHS

The AMHS shall be organisationally composed of AMHS Management Domains.

Note 1.— An AMHS Management Domain may elect to operate as either an ADMD (Administration Management Domain) or a PRMD (Private Management Domain), depending on the national telecommunications regulation in force in the country(ies) where it operates and on its relationships with other Management Domains.

Note 2.— A PRMD which is subordinate to one or several AMHS ADMDs may qualify as AMHS Management Domain if it satisfies the provisions of 3.1.2.

3.1.2.1.4 AMHS Management Domain configurations

3.1.2.1.4.1 Minimal set of systems

The minimal set of systems implemented and operated by an AMHS Management Domain shall be one of the following:

a) an ATS Message Server and one or several ATS Message User Agents;

b) an AFTN/AMHS Gateway; or

c) any combination of a) and b).

3.1.2.1.4.2 Interconnection between two AMHS Management Domains

An interconnection between two AMHS Management Domains shall be implemented as one of the following:

a) a connection between two ATS Message Servers;

b) a connection between an ATS Message Server and an AFTN/AMHS Gateway; or

c) a connection between two AFTN/AMHS Gateways.

3.1.2.1.5 Naming and addressing principles

3.1.2.1.5.1 AMHS Naming and Addressing

3.1.2.1.5.1.1 AMHS O/R Names

For the support of the Basic ATS Message Service, the O/R (originator/recipient) name of an AMHS user shall comprise:

a) the O/R address of the AMHS user, called an MF-address; and

b) optionally the directory name of the AMHS user, if the policy of the AMHS Management Domain, to which the AMHS user belongs, includes the local support of directory-names.

Note.— As a matter of policy local to an AMHS Management Domain, the directory name component of an O/R name may be used by the implementation of the Optional DIR (Use of Directory) FG (Functional Group).

3.1.2.1.5.1.2 Structure of a MF-Address

The MF-Address (MHS-form address) of an AMHS user shall comprise:

a) a set of attributes as specified in 3.1.2.1.5.1.3, identifying the AMHS Management Domain of which the AMHS user, either direct or indirect, is a service-user; and

b) a set of attributes as specified in 3.1.2.1.5.1.4, identifying uniquely the AMHS user within the AMHS Management Domain, in compliance with the AMHS addressing scheme implemented by the AMHS Management Domain.

Note.— The attributes present in the identifier defined in item b) may include any standard or domain-defined attribute as specified in section 18 of ISO/IEC 10021-2, other than country-name, administration-domain-name and private-domain-name.

3.1.2.1.5.1.3 AMHS Management Domain identifier

The attributes identifying an AMHS Management Domain shall include the following standard attributes as specified in ISO/IEC 10021-2, section 18.3, depending on the status under which the AMHS Management Domain has elected to operate:

a) *country-name*,

b) *administration-domain-name*,

c) *private-domain-name*, if the AMHS Management Domain has elected to operate as a PRMD.

3.1.2.1.5.1.4 AMHS Addressing Schemes

3.1.2.1.5.1.4.1 General provisions

Note 1.— It is a matter of policy local to each AMHS Management Domain to implement either a locally defined AMHS Addressing Scheme, or a Common AMHS Addressing Scheme, or a combination of these. The single Common ICAO AMHS Addressing Scheme defined in the present version of this document is the XF-Addressing Scheme. Aeronautical Industry X.400 Addressing Schemes are defined in appropriate Aeronautical Industry Standards.

Note 2.— Each AMHS Addressing Scheme includes the set of attributes identifying the AMHS Management Domain as specified in 3.1.2.1.5.1.3.

3.1.2.1.5.1.4.2 XF-Addressing Scheme

The XF-Address (translated address) of a direct or indirect AMHS user shall be composed exclusively of the following:

a) an AMHS Management Domain identifier as specified in 3.1.2.1.5.1.3;

b) an *organization-name* attribute:

1) as specified in ISO/IEC 10021-2, Section 18.5,

2) taking the 4-character value “AFTN”, and

3) encoded as a Printable String; and

c) an *organizational-unit-names* attribute:

1) as specified in ISO/IEC 10021-2, Section 18.5,

2) comprising a sequence of one single element, which takes the 8-character alphabetical value of the AF-Address (AFTN-form address) of the user, and

3) encoded as a Printable String.

Note 1.— An XF-Address is a particular MF-Address of which the attributes identifying the user within an AMHS Management Domain (i.e. those attributes other than country-name, administration-domain-name and private-domain-name) may be converted by an algorithmic method to and from an AF-Address. The algorithmic method requires the additional use of look-up tables which are limited, i.e. which include only a list of AMHS Management Domains rather than a list of individual users, to determine the full MF-address of the user.

Note 2.— No distinction is made between upper case and lower case.

3.1.2.1.5.2 Upper Layer Naming and Addressing

3.1.2.1.5.2.1 Application Process Titles

3.1.2.1.5.2.1.1 **Recommendation.**— *The Application Process Title of an ATS Message Server should be as specified in 4.3.3.2.*

3.1.2.1.5.2.1.2 **Recommendation.**— *The Application Process Title of an AFTN/AMHS Gateway should be as specified in 4.3.3.2.*

3.1.2.1.5.2.1.3 **Recommendation.—** *The Application Process Title of an ATS Message User Agent should be as specified in 4.3.3.2.*

3.1.2.1.5.2.2 Application Entity Qualifiers

3.1.2.1.5.2.2.1 **Recommendation.—** *The Application Entity Qualifier of an ATS Message Server should be AMS (7).*

3.1.2.1.5.2.2.2 **Recommendation.**— *The Application Entity Qualifier of an AFTN/AMHS Gateway should be GWB (8).*

3.1.2.1.5.2.2.3 **Recommendation.—** *The Application Entity Qualifier of an ATS Message User Agent should be AUA (9).*

3.1.2.1.5.2.3 Transport, Session and Presentation Addresses

The TSAP (Transport Service Access Point) of an ATS Message Server or of an ATS Message User Agent shall comply with the provisions of 5.4.

Note 1.— The assignment of a transport selector value is a matter local to an AMHS Management Domain.

Note 2.— The format and encoding of a session selector in the AMHS is specified in ISO/IEC ISP 11188-1, section 9.3.

Note 3.— The assignment and administration of session selectors is a matter local to an AMHS Management Domain.

Note 4.— The format and encoding of a presentation selector in the AMHS is specified in ISO/IEC ISP 11188-1, section 7.2.

Note 5.— The assignment and administration of presentation selectors is a matter local to an AMHS Management Domain.

3.1.2.1.6 AMHS Routing and rerouting

3.1.2.1.6.1 The definition of AMHS routing shall be subject to multilateral agreements.

3.1.2.1.6.2 The MTAs implemented by an AMHS Management Domain shall be collectively able to route on *country-name, ADMD-name, PRMD-name, organization-name* and *organizational-units-name* attributes.

3.1.2.1.7 AMHS Traffic logging upon origination

An AMHS Management Domain shall be responsible for long-term logging of all messages in their entirety which are originated by its direct AMHS users, for a period of at least thirty days.

Note.— This requirement implies the logging of the entire BER-encoded ASN.1 messages.

3.1.2.2 ATS Message Service Specification

3.1.2.2.1 ATS Message User Agent Specification

Note.— For the support of the Basic ATS Message Service, an ATS Message User Agent complies with:

a) profile AMH21 as specified in ISO/IEC ISP 12062-2;

b) the requirements of Repertoire Group A, for messages including a body part whose type is an Extended Body Part Type of general-text-body-part type;

c) the additional provisions relating to parameters generated at an ATS Message User Agent, as specified in 3.1.2.2.1.1; and

d) the provisions related to traffic logging as specified in 3.1.2.2.1.2.

3.1.2.2.1.1 Additional provisions on parameters

3.1.2.2.1.1.1 Message Content Profile Specification

In an ATS Message User Agent, the content of the Inter-Personal Messages conveyed in support of the Basic ATS Message Service shall conform to the basic requirements of AMH21 as specified in Clause A.1 of ISO/IEC ISP 12062-2, Annex A and to the additional requirements described in Table 3.1.2-1 which are specific to the Basic ATS Message Service.

Note 1.— Table 3.1.2-1 specifies the additional requirements in the form of a PRL (Profile Requirement List) expressing restrictions to a set of rows of the AMH21 profile, which are referred to using their reference in ISO/IEC ISP 12062-2.

Note 2.— There is no profile specification for the ATS Message User Agent at the level of the access protocol, i.e. at the level of the communication with the associated ATS Message Server, as this is considered to be a matter local to each AMHS Management Domain. If it is desired to use standard ISO/IEC 10021 protocols for this communication, then profile AMH23 (for P3) or profile AMH24 (for P7) as specified in ISO/IEC ISP 12062-4 or ISO/IEC ISP 12062-5, respectively, may be implemented.

Note 3.— The use of the ia5-text body part as specified in Table 3.1.2-1/AMH21/A1.3/1 ensures operability with both 1984 and 1988 IPM (Inter-Personal Message) UAs for the exchange of unstructured character data.

Table 3.1.2-1. Requirements specific to the Basic ATS Message Service in addition to profile AMH21

Ref	Element	Origination		Reception		Basic ATS Message Service Support	ATN reference	ISP 12062-2 Notes/ References
		Base	ISP	Base	ISP			
Part 1 : AMH21/A.1.3		**IPM body**						
1	ia5-text	O	O	O	M	O/M		
1.2	data	M	M	M	M	M/M	3.1.2.2.3.2	

Ref	Element	Origination		Reception		Basic ATS Message Service Support	ATN reference	ISP 12062-2 Notes/ References
		Base	ISP	Base	ISP			
Part 2 : AMH21/A.1.3.1 Extended body part support								
1	ia5-text-body-part	O	O	O	M	O/M		see AMH21/ A.1.3/1
11	general-text-body-part	O	M	O	M	M/M	3.1.2.2.3.2 and Table 3.1.2-1 Part 4	
Part 3 : AMH21/A.1.5 Common data types								
1	RecipientSpecifier							
1.2	notification-requests	O	O	M	M	M/M	3.1.2.2.3.3	
1.2.1	rn	O	O	O	O	M/M	3.1.2.2.3.3	
1.2.2	nrn	O	O	M	M	M/M		
2	ORDescriptor							
2.1	formal-name	M	M1	M	M1	M1/M1	3.1.2.2.3.1	
Part 4 : AMH21/A.1.3.2 General text repertoire support								
1	Basic (ISO 646) (repertoire identifiers {1, 6})	M	M	M	M	M/M		Repertoire Group A
2	Basic-1 (ISO 8859-1) (repertoire identifiers {1, 6, 100}	O	M	O	M	O/O		Repertoire Group B

Legend : see 3.1.1
M = mandatory support
M1 = mandatory O/R name minimal support
O = optional support

3.1.2.2.1.1.2 Additional requirements upon MT-Elements of Service at an ATS Message User Agent

For the support of the Basic ATS Message Service, the *priority* element of an AMHS Message generated at an ATS Message User Agent shall take the value "urgent" if, and only if, the value of the priority-indicator in the ATS-Message-Priority as specified in 3.1.2.2.3.2.1 is "SS".

3.1.2.2.1.2 Traffic logging requirements at an ATS Message User Agent

Note.— The requirement expressed in 3.1.2.1.7 may be implemented in the ATS Message User Agent.

3.1.2.2.2 ATS Message Server Specification

Note.— For the support of the Basic ATS Message Service, an ATS Message Server complies with:

a) the profile specification expressed in 3.1.2.2.2.1; and

b) the provisions related to traffic logging as specified in 3.1.2.2.2.2.

3.1.2.2.2.1 Profile Specification

3.1.2.2.2.1.1 Upper Layer Requirements

In an ATS Message Server, the Message Transfer (P1) implementation of the IPM Service in support of the Basic ATS Message Service shall conform to:

a) the basic requirements of AMH22 as specified in Clause B.1 of ISO/IEC ISP 12062-3, Annex B; and

b) the additional requirements described in Clause B.2.2. for the support of the IPM Distribution List Functional Group.

Note 1.— This in turn places no requirements concerning the P1 implementation other than:

a) the basic requirements of AMH11 specified for Common Messaging in annex A.1 of ISO/IEC ISP 10611-3, implying the mandatory support of the AMH111 Profile implementing the mts-transfer application context; and

b) the additional requirements specified for the Common Messaging DL (Distribution List) Functional Group in annex A.2.2 of ISO/IEC ISP 10611-3.

Note 2.— As a consequence of Note 2 in 3.1.2.2.1.1.1, the optional implementation of Message Stores (MS) in an ATS Message Server, being related to the access protocol from an ATS Message User Agent to an ATS Message Server, is a matter local to each AMHS Management Domain.

Note 3.— The additional support by an ATS Message Server of the AMH112 Profile as specified in ISO/IEC ISP 10611-3, for conformance to CCITT X.400 in order to interconnect with public ADMDs is a matter of policy local to each AMHS Management Domain.

Note 4.— For the use of the Association Control Service Element (ACSE) by an AMHS application, the application-context name which is used as a parameter in an A-ASSOCIATE is defined in the base standards (see ISO/IEC 10021-6).

Note 5.— The specification in 3.1.2.2.2.1.1 places no requirements for the Reliable Transfer Service Element (RTSE) and for ACSE other than conformance with ISO/IEC ISP 10611-2 in accordance with the P1 application-context(s) for which conformance is claimed.

Note 6.— The specification in 3.1.2.2.2.1.1 places no requirements for the Presentation and Session Layers other than conformance with ISO/IEC ISP 10611-2 in accordance with the P1 application-context(s) for which conformance is claimed.

3.1.2.2.2.1.2 Use of the Transport Service

3.1.2.2.2.1.2.1 The Basic ATS Message Service shall make use of the Connection Mode Transport Service as specified in 5.5.

Note.— For the support of the Basic ATS Message Service, the use of the expedited data option at the establishment of the transport connection is a local matter which may depend on the implemented application-context.

3.1.2.2.2.1.2.2 For the support of the Basic ATS Message Service, transport connections shall be established over the ATN Transport Service between systems belonging to the AMHS using the Residual Error Rate (RER) abstract-value “high”.

3.1.2.2.2.1.2.3 For the support of the Basic ATS Message Service, transport connections shall be established over the ATN Transport Service between systems belonging to the AMHS using the Transport Connection Priority abstract-value “6”, which corresponds to the message category “flight regularity communications”.

3.1.2.2.2.1.2.4 For the support of the Basic ATS Message Service, transport connections shall be established over the ATN Transport Service between systems belonging to the AMHS using the value of the ATN Security Label as specified in 5.6, which corresponds to:

a) the ATN Traffic Type “ATN Operational Communications”;

b) the Sub-Type “Air Traffic Services Communications” (ATSC); and

c) “No Traffic Type Policy Preference”.

3.1.2.2.2.2 Traffic logging requirements at an ATS Message Server

3.1.2.2.2.2.1 The ATS Message Server shall perform a long-term logging, for a period of at least thirty days, of the actions taken with respect to every message received at the ATS Message Server, whether from an ATS Message User Agent or from another ATS Message Server, and to every report received or generated at the ATS Message Server.

3.1.2.2.2.2.2 For the long-term logging of information related to a message submitted to or received by an ATS Message Server, the following parameters related to the message shall be logged:

a) *message-identifier*;

b) *priority*;

c) *content-type*;

d) *originator-name*;

e) *recipient-name* elements on responsibility list;

f) message-content-size;

g) last element of the *trace-information* (if any);

h) arrival-time or submission-time;

i) transfer destination (if any);

j) transfer time (if any);

k) *this-recipient-name* (if message delivery is performed by the ATS Message Server);

l) *delivery-time* (if any);

m) delivery and/or non-delivery reports generated (if any); and

n) event date/time.

Note.— The responsibility list identifies recipients whose perRecipientIndicator responsibility bit has the abstract-value "responsible".

3.1.2.2.2.2.3 For the long-term logging of information related to a report generated or received by an ATS Message Server, the following parameters related to the report shall be logged:

a) *report-identifier*;

b) *subject-identifier*;

c) *actual-recipient-name* elements;

d) *report-type* elements;

e) *report-destination-name*;

f) last element of the *trace-information* (if any);

g) *arrival-time* in the ATS Message Server or generation time;

h) transfer destination (if any);

i) transfer time (if any);

j) *OR-name* of the report recipient (if report delivery is performed by the ATS Message Server);

k) *delivery-time* (if any); and

l) event date/time.

3.1.2.2.3 Parameters

3.1.2.2.3.1 AMHS Addresses

In the AMHS, the O/R address of a direct AMHS user belonging to an AMHS Management Domain shall be a MF-Address.

3.1.2.2.3.2 Text

The body of an Inter-Personal Message (IPM) shall comprise a single body part carrying IA-5 characters and structured as depicted in Table 3.1.2-2.

Note 1.— This body part structure and its components which are described in the subsequent clauses are specific to the Basic ATS Message Service.

Note 2.— This clause places no constraint on its implementation, which may take place at the level of the user-interface.

Table 3.1.2-2. Structure of an IPM in the Basic ATS Message Service

Ref	Element	Basic ATS Message Service Support		Value	IA-5 Encoding
		Orig	Rec		
1	ATS-Message-Header	M	M		
1.1	start-of-heading	M	M	(SOH)	(0/1)
1.2	ATS-Message-Priority	M	M		
1.2.1	priority-prompt	M	M	PRI:(single space)	(5/0)(5/2)(4/9)(3/10)(2/0)
1.2.2	priority-indicator	M	M	see 3.1.2.2.3.2.1	see 3.1.2.2.3.2.1
1.2.3	priority-separator	M	M	(CR)(LF)	(0/13)(0/10)
1.3	ATS-Message-Filing-Time	M	M		
1.3.1	filing-time-prompt	M	M	FT:(single space)	(4/6)(5/4)(3/10)(2/0)
1.3.2	filing-time	M	M	see 3.1.2.2.3.2.2	see 3.1.2.2.3.2.2
1.3.3	filing-time-separator	M	M	(CR)(LF)	(0/13)(0/10)
1.4	ATS-Message-Optional-Heading-Info	O	M		
1.4.1	OHI-prompt	M	M	OHI:(single space)	(4/15)(4/8)(4/9)(3/10)(2/0)
1.4.2	optional-heading-information	M	M	see 3.1.2.2.3.2.3	see 3.1.2.2.3.2.3
1.4.3	OHI-separator	M	M	(CR)(LF)	(0/13)(0/10)
1.5	end-of-heading-blank-line	M	M	(LF)	(0/10)
1.6	start-of-text	M	M	(STX)	(0/2)
2	ATS-Message-Text	M	M	see 3.1.2.2.3.2.4	see 3.1.2.2.3.2.4

Legend (see 3.1.1):
M = mandatory support
O = optional support
3.1.2.2.3.2.1 ATS Message Priority

Each message shall be assigned to one of five priority groups which are designated, and have the value of, the priority indicators SS, DD, FF, GG and KK.

3.1.2.2.3.2.2 ATS Message Filing Time

Each message shall include a filing-time element, designated as a date-time group consisting of six numerical characters, the first two digits representing the date of the month and the last four digits the hours and minutes in UTC.

3.1.2.2.3.2.3 ATS Message Optional Heading Info

3.1.2.2.3.2.3.1 It shall be possible to associate an optional heading information with each message.

3.1.2.2.3.2.3.2 The value of the optional-heading-information element shall comprise a character string with a maximum length of 54 characters.

3.1.2.2.3.2.4 ATS Message Text

The ATS-Message-Text element shall be composed of IA-5 characters with no further restriction.

3.1.2.2.3.3 Notification requests

The *notification-requests* element in a RecipientSpecifier in an IPM Heading shall take the abstract-value "rn" if, and only if, the value of the priority-indicator is "SS".

Note.— This clause places no constraint on its implementation, which takes place at the level of the user-interface.

3.1.2.3 AFTN/AMHS Gateway Specification

3.1.2.3.1 General

3.1.2.3.1.1 An AFTN/AMHS Gateway shall provide for an interworking between the AFTN and the ATN such that communication with other AFTN/AMHS Gateways and with ATS Message Servers is possible.

3.1.2.3.1.2 An AFTN/AMHS Gateway shall consist of the four following logical components:

a) AFTN Component;

b) ATN Component;

c) Message Transfer and Control Unit; and

d) Control Position.

Note.— This division into logical components is a convenient way of specifying functions of a gateway. There is no requirement for an AFTN/AMHS Gateway to be implemented according to this structure.

3.1.2.3.1.3 An AFTN/AMHS Gateway shall be able to perform actions upon receipt of any category of AMHS information object by its ATN Component.

3.1.2.3.1.4 An AFTN/AMHS Gateway shall be able to perform actions upon receipt of any type of AFTN message by its AFTN Component.

3.1.2.3.2 AFTN/AMHS Gateway components

3.1.2.3.2.1 AFTN component

3.1.2.3.2.1.1 The AFTN component shall handle the interface to the AFTN and provide an interface to the Message Transfer and Control Unit, implementing:

a) all the applicable requirements of Annex 10, Volume II in a manner so as to be indistinguishable from an operational AFTN station by the AFTN centre to which the gateway is connected; and

b) additional requirements which are necessary due to the AFTN Component pertaining to an AFTN/AMHS Gateway.

3.1.2.3.2.1.2 If an AFTN/AMHS Gateway is connected to an AFTN centre which is capable of using only ITA-2 (International Telegraph Alphabet No 2) format, the AFTN component shall convert messages to/from the IA-5 format.

Note.— This allows the Message Transfer and Control Unit to use IA-5 characters internally, as specified in 3.1.2.3.2.3.2.

3.1.2.3.2.1.3 The AFTN Component shall incorporate an AFTN procedure handler providing for all AFTN functions prescribed for the interface to the AFTN.

3.1.2.3.2.1.4 When received by the AFTN Component, AFTN service messages as generally specified in Annex 10, Volume II, 4.4.1.1.9 and subclauses, shall be handled by the AFTN Component of the Gateway in one of four mutually exclusive manners, depending on the category of the service message:

a) transfer to the Message Transfer and Control Unit to be processed as specified in 3.1.2.3.4 if the service message is an AFTN acknowledgement message, as specified in Annex 10, Volume II, 4.4.10.1.6.1 and 4.4.15.6;

b) transfer to the Message Transfer and Control Unit to be processed as specified in 3.1.2.3.4 if the service message is an AFTN service message requesting correction of a message received with an unknown addressee indicator as specified in Annex 10, Volume II, 4.4.11.13.3;

c) processing as specified in 3.1.2.3.2.1.12 if the service message is an AFTN service message requesting from the originator repetition of an incorrectly received message when it is detected that a message has been mutilated, as specified in Annex 10, Volume II, 4.4.11.1 and 4.4.16.2.2; or

d) processing in compliance with the provisions of Annex 10, Volume II, without being passed to the Message Transfer and Control Unit, if the service message belongs to any other category of AFTN service message.

3.1.2.3.2.1.5 When received by an AFTN/AMHS Gateway, AFTN channel-check transmissions as specified in Annex 10, Volume II, 4.4.9.3 and 4.4.15.5 shall:

a) be handled by the AFTN Component in compliance with the provisions of Annex 10, Volume II; and

b) be prevented from being passed to the Message Transfer and Control Unit.

3.1.2.3.2.1.6 The AFTN Component shall pass all messages, other than those referred to in 3.1.2.3.2.1.4 c) and d), and in 3.1.2.3.2.1.5, received from the AFTN to the Message Transfer and Control Unit for processing as specified in 3.1.2.3.4, and provided that the conditions of 3.1.2.3.2.1.7 are met.

3.1.2.3.2.1.7 The processing by the AFTN Component shall ensure that all messages and service messages received from the AFTN and passed to the Message Transfer and Control Unit for further processing by the AFTN/AMHS Gateway are constructed in strict accordance with the provisions of Annex 10, Volume II, paragraphs 4.4.15.1 through 4.4.15.3.12 and 4.4.15.6.

3.1.2.3.2.1.8 The AFTN Component shall perform short-term retention of all messages transmitted towards the AFTN in a manner equivalent to that specified for an AFTN communication centre in Annex 10, Volume II, 4.4.1.7.

3.1.2.3.2.1.9 The AFTN Component shall perform long-term retention of the heading, address and origin parts of all messages received from the AFTN, with the message receipt-time and the action taken thereon, for a period of at least thirty days.

3.1.2.3.2.1.10 The AFTN Component shall perform long-term retention of all AFTN messages, in their entirety, that it generates, for a period of at least thirty days.

3.1.2.3.2.1.11 The AFTN Component shall perform long-term retention of the heading, address and origin parts of all messages received from the Message Transfer and Control Unit and the action taken thereon, for a period of at least thirty days.

3.1.2.3.2.1.12 Upon reception by an AFTN/AMHS Gateway of an AFTN service message requesting repetition by the originator of an incorrectly received message as specified in Annex 10, Volume II, 4.4.11.1 or 4.4.16.2.2, the AFTN Component shall perform one of the following actions:

a) terminate the procedure and report an error situation to a control position if the referenced subject AFTN message did not pass through the gateway or if the AFTN Component is not in possession of an unmutilated copy of the subject AFTN message; or

b) reassume responsibility for the mutilated message and repeat the message in compliance with the provisions of Annex 10, Volume II, 4.4.11.3, if the mutilated message is detected as having passed through the gateway and if the AFTN Component is in possession of an unmutilated copy of the message.

Note.— The determination whether the AFTN Component is in possession of an unmutilated copy of the message, as mentioned in items a) and b) above, may require the assistance of a control position.

3.1.2.3.2.1.13 If, for any reason, the Message Transfer and Control Unit is unable to accept AFTN messages passed by the AFTN Component, then the AFTN Component shall handle this situation in compliance with the provisions of Annex 10, Volume II, 4.4.1.5.2.3.

Note.— Such a condition may be caused by the inability of the Message Transfer and Control Unit to pass AMHS messages to the ATN Component.

3.1.2.3.2.1.14 The AFTN Component shall ensure that all information objects constructed by the Message Transfer and Control Unit for transmission over the AFTN are handled in accordance with the AFTN procedure, in application of 3.1.2.3.2.1.3 above.

3.1.2.3.2.1.15 If the AFTN Component is unable to handle an AFTN service message or an AFTN channel-check transmission in compliance with the provisions of Annex 10, Volume II, as specified in 3.1.2.3.2.1.4 d) or 3.1.2.3.2.1.5, then the error condition shall be logged and reported to a control position.

3.1.2.3.2.1.16 An AFTN address shall be allocated to the AFTN Component.

3.1.2.3.2.2 ATN Component

3.1.2.3.2.2.1 The ATN Component shall allow the AFTN/AMHS Gateway to function as an end system on the ATN.

3.1.2.3.2.2.2 The ATN Component shall handle the interface to the AMHS, and provide an interface to the Message Transfer and Control Unit as specified in 3.1.2.3.2.4, implementing a MTA complying with the profile specification included in 3.1.2.2.2.1 so as to be externally indistinguishable from an ATS Message Server by the ATS Message Server(s) or other AFTN/AMHS Gateway(s) to which it is connected.

3.1.2.3.2.2.3 If, for any reason, the Message Transfer and Control Unit is unable to accept messages or probes passed by the ATN Component, then the ATN Component shall behave as follows:

a) attempt to reroute the message or probe as specified in ISO/IEC 10021-4, 14.3.4.4;

b) if no alternate route is available in the MTA-routing tables or all such routes cannot be succesfully used, reject the message for all the message recipients, whose *responsibility* element in the *per-recipient-indicators* has the abstract-value “responsible” in the received message, with the *non-delivery-reason-code* and *non-delivery-diagnostic-code* elements of the non-delivery report taking the abstract-values specified in the base standards (ISO/IEC 10021-4, 14.3.4.4., item 1).

Note.— Such a condition may be caused by the inability of the Message Transfer and Control Unit to pass AFTN messages to the AFTN Component.

3.1.2.3.2.2.4 If the AMHS Management Domain operating an AFTN/AMHS Gateway desires to implement Message Handling System optional functional groups in addition to the specification of 3.1.2.3.2.2.2 above, this shall be performed in the ATN Component.

Note.— This applies in particular to the Redirection Functional Group. If implemented, redirection may be performed by the ATN Component, caused by a failure situation as envisaged in 3.1.2.3.2.2.3 above for example.

3.1.2.3.2.2.5 The ATN Component shall ensure that all information objects constructed by the Message Transfer and Control Unit for transfer in the AMHS are handled in accordance with the procedures specified in the base standards for a relaying MTA implementing the profile specified in 3.1.2.2.2.1, in application of 3.1.2.3.2.2.2 above.

3.1.2.3.2.2.6 The ATN Component shall implement a traffic logging function identical to that of the MTA included in an ATS Message Server as specified in 3.1.2.2.2.2.

3.1.2.3.2.2.7 The ATN Component shall ensure that all AMHS information objects passed to the Message Transfer and Control Unit comply with the base standards.

3.1.2.3.2.3 Message Transfer and Control Unit

3.1.2.3.2.3.1 The Message Transfer and Control Unit in an AFTN/AMHS Gateway shall provide a bi-directional conversion facility between the AFTN component and the ATN component, consisting of:

a) a set of general functions as specified in 3.1.2.3.3; and

b) AFTN/AMHS conversion functions as respectively specified in 3.1.2.3.4 for the AFTN to AMHS conversion and in 3.1.2.3.5 for the AMHS to AFTN conversion.

3.1.2.3.2.3.2 The Message Transfer and Control Unit shall use IA-5 characters internally.

3.1.2.3.2.3.3 The Message Transfer and Control Unit in an AFTN/AMHS Gateway shall pass all the AMHS information objects which it constructs in application of 3.1.2.3.4 and 3.1.2.3.5.6 to the ATN Component of the gateway, for further conveyance in the AMHS.

3.1.2.3.2.3.4 For the generation of AMHS messages and reports, and for the processing of received AMHS messages, probes and reports, the Message Transfer and Control Unit shall have the capability to interpret the semantics and to perform actions related to the ISO/IEC 10021 Elements of Service which are part of the basic requirements of the MT service as specified in ISO/IEC ISP 12062-3.

3.1.2.3.2.3.5 The Message Transfer and Control Unit in an AFTN/AMHS Gateway shall pass all the AFTN messages which it constructs in application of 3.1.2.3.5 and 3.1.2.3.4.2.1.4.2 to the AFTN Component of the AFTN/AMHS Gateway, for further conveyance in the AFTN.

3.1.2.3.2.3.6 The Message Transfer and Control Unit shall ensure that all the AMHS information objects which it constructs comply with section 7 (for IPMs) and section 8 (for RNs) of ISO/IEC 10021-7, complemented with the additional requirements included in Table 3.1.2-1, and with the section 12.2.1.1 of ISO/IEC 10021-4 (for messages) and section 12.2.1.3 of ISO/IEC 10021-4 (for reports).

3.1.2.3.2.3.7 The Message Transfer and Control Unit shall ensure that all the AFTN information objects which it constructs comply with Annex 10, Volume II, 4.4.15.

3.1.2.3.2.4 Interface between the ATN Component and the Message Transfer and Control Unit

3.1.2.3.2.4.1 The ATN Component shall exchange information objects with the Message Transfer and Control Unit via its MTA transfer-port as specified in ISO/IEC 10021-4, section 12.2.

3.1.2.3.2.4.2 The ATN Component shall invoke the Message-transfer, Report-transfer and Probe-transfer abstract operations, respectively, to pass AMHS messages, reports and probes to the Message Transfer and Control Unit.

3.1.2.3.2.4.3 The Message Transfer and Control Unit shall invoke the Message-transfer and Report-transfer abstract operations, respectively, to pass AMHS messages and reports to the ATN Component.

3.1.2.3.2.5 Interface between the AFTN Component and the Message Transfer and Control Unit

3.1.2.3.2.5.1 An AFTN message or service message passed by the AFTN Component to the Message Transfer and Control Unit in application of 3.1.2.3.2.1.4 items a) and b), 3.1.2.3.2.1.6 and 3.1.2.3.2.1.7 shall be:

a) transferred according to the table of priorities as specified in Annex 10, Volume II, 4.4.1.2.1; and

b) passed as received by the AFTN Component from the adjacent AFTN centre, with the possible exception of an ITA-2 to IA-5 conversion performed in application of 3.1.2.3.2.1.2, and including the unaltered AFTN heading if present in the received message.

3.1.2.3.2.5.2 An AFTN message or service message passed by the Message Transfer and Control Unit to the AFTN Component in application of 3.1.2.3.2.3.5 shall be:

a) transferred according to the table of priorities as specified in Annex 10, Volume II, 4.4.1.2.1; and

b) passed as constructed by the Message Transfer and Control Unit, and thus without message heading as specified in Annex 10, Volume II, 4.4.15.1.1.

3.1.2.3.2.5.3 The AFTN Component shall return to the Message Transfer and Control Unit, as the result of the transfer operation described in 3.1.2.3.2.5.2, the Transmission Identification, if any, constructed by the AFTN Component for the transmission of the message or service message over the AFTN.

3.1.2.3.2.6 AFTN/AMHS Gateway Control Position

3.1.2.3.2.6.1 The AFTN/AMHS Gateway Control Position shall be used as the place where errors which occurred in the AFTN/AMHS Gateway and certain non-deliveries which occurred in the AMHS are reported for appropriate action.

3.1.2.3.2.6.2 The appropriate action to be undertaken on reporting of an error or of a non-delivery to an AFTN/AMHS Gateway control position shall be either:

a) a matter of policy which is local to the AMHS Management Domain operating the AFTN/AMHS Gateway; or

b) subject to multilateral agreements.

Note.— For some categories of error situations, 3.1.2 specify the actions to be taken, e.g. message rejection and generation of an appropriate service message (to the AFTN) or non-delivery report (to the AMHS). The specified actions aim at minimizing the assistance of the control position. However it may be a matter of policy local to the AMHS Management Domain operating an AFTN/AMHS Gateway to try to reduce the occurrence of message rejection with the assistance of the control position.

3.1.2.3.2.6.3 When the action chosen to handle an error situation includes the generation of an AMHS information object, the category of information object used for this purpose shall be an IPM conveying appropriate service information.

Note 1.— The service information to be conveyed may be derived, for example, from an AFTN service message.

Note 2.— The presentation of the service information is a matter of local policy.

3.1.2.3.3 General functions

3.1.2.3.3.1 Traffic logging

3.1.2.3.3.1.1 The Message Transfer and Control Unit shall perform long-term logging, as specified in 3.1.2.3.3.1.2 to 3.1.2.3.3.1.6, for a period of at least thirty days, of information related to the following exchanges of information objects with the ATN Component and with the AFTN Component:

a) AMHS message transfer out (to the ATN Component);

b) AMHS report transfer out (to the ATN Component);

c) AMHS message transfer in (from the ATN Component);

d) AMHS report transfer in (from the ATN Component);

e) AFTN message conveyance out (to the AFTN Component);

f) AFTN message conveyance in (from the AFTN Component);

g) AFTN service message indicating an unknown addressee indicator conveyance in (from the AFTN Component); and

h) AFTN service message indicating an unknown addressee indicator conveyance out (to the AFTN Component).

3.1.2.3.3.1.2 For the long-term logging of information related to an AMHS Message Transfer In and AFTN message conveyance out, the following parameters, relating to the messages, shall be logged by the Message Transfer and Control Unit:

a) input *message-identifier*;

b) *IPM-identifier*, if any;

c) *common-fields* and either *receipt-fields* or *non-receipt-fields* of IPN (Inter-Personal Notification), if any;

d) action taken thereon (reject with *non-delivery-reason-code* and *non-delivery-diagnostic-code*, convert as AFTN message, convert as AFTN acknowledgement message, splitting due to number of recipients or message length, delivery report generation);

e) event date/time;

f) Origin line of converted AFTN message or service message, if any; and

g) transmission identification of AFTN message(s) or service message(s), if returned by the AFTN Component.

3.1.2.3.3.1.3 For the long-term logging of information related to AFTN message conveyance in and AMHS Message Transfer Out, the following parameters, relating to the messages, shall be logged by the Message Transfer and Control Unit:

a) Origin line of AFTN message (or AFTN acknowledgement message);

b) transmission identification of AFTN message or service message, if any;

c) action taken thereon (reject with rejection cause, convert as IPM, convert as RN, AFTN service message indicating an unknown addressee indicator generation);

d) event date/time;

e) *MTS-identifier*, if any; and

f) *IPM-identifier*, if any.

3.1.2.3.3.1.4 For the long-term logging of information related to an AMHS Message Report In and/or AFTN Service Message indicating an unknown addressee indicator conveyance out, the following parameters, relating to the report and/or service message, shall be logged by the Message Transfer and Control Unit:

a) *report-identifier* (if report in);

b) *subject-identifier* (if report in);

c) action taken thereon if report in (discard, convert into AFTN service message);

d) event date/time;

e) Origin line of converted AFTN service message (if service message out);

f) Origin line of subject AFTN message (if service message out and no report in); and

g) transmission identification of AFTN message or service message, if any.

3.1.2.3.3.1.5 For the long-term logging of information related to an AFTN Service Message indicating an unknown addressee indicator conveyance in and/or to an AMHS Message Report Out, the following parameters, relating to the service message and/or report, shall be logged by the Message Transfer and Control Unit:

a) Origin line of converted AFTN service message (if service message in);

b) Origin line of subject AFTN message (if service message in);

c) transmission identification of AFTN message or service message, if any;

d) action taken thereon if AFTN service message in (discard, convert into AMHS report);

e) *report-identifier* (if report out);

f) *subject-identifier* (if report out); and

g) event date/time

3.1.2.3.3.2 Address look-up tables

The Message Transfer and Control Unit shall include look-up tables used for address conversion, covering two aspects:

a) a MD look-up table as specified in 3.1.2.3.3.2.1, for the algorithmic conversion of an AF-Address to an XF-Address; and

b) a user address look-up table of individual users as specified in 3.1.2.3.3.2.2, for the conversion of an AF-Address to and from an MF-Address of any AMHS Addressing Scheme.

Note.— The way in which these tables are populated and maintained up-to-date is an organisational matter.

3.1.2.3.3.2.1 MD look-up Tables

3.1.2.3.3.2.1.1 The MD (Management Domain) look-up table maintained by in the Message Transfer and Control Unit shall include a list of entries identifying an organizational entity, which either is an AMHS Management Domain, or collectively uses the services of a given AMHS Management Domain, each entry comprising:

a) a string of characters identifying one of the following:

1) a country (two-letter designator as specified in ICAO Document 7910);

2) a location (four-letter designator as specified in ICAO Document 7910);

3) an organization within a country (combination of a two-letter designator as specified in ICAO Document 7910 with a three-letter designator as specified in ICAO Document 8585); or

4) an organization at a location (combination of a four-letter designator as specified in ICAO Document 7910 with a three-letter designator as specified in ICAO Document 8585); and

b) the set of attributes identifying either the AMHS Management Domain implemented by the organizational entity defined in a), if existing, or the AMHS Management Domain whose AFTN/AMHS Gateway may be used to communicate with indirect AMHS users within the aforementioned organisational entity, this set of attributes being composed of:

1) country-name;

2) ADMD-name; and

3) PRMD-name (if any).

3.1.2.3.3.2.1.2 It shall be possible to derive unambiguously a single item b) from item a) by a search operation in the MD look-up table.

3.1.2.3.3.2.2 User address look-up Tables

3.1.2.3.3.2.2.1 The user address look-up table maintained by the Message Transfer and Control Unit shall include a list of entries, each of them comprising:

a) the AF-Address of either an indirect AMHS user who also has a MF-Address, or of a direct AMHS user who has an AF-Address for communication with indirect AMHS users; and

b) the MF-Address of that AMHS user, either direct or indirect, including all its address attributes.

3.1.2.3.3.2.2.2 It shall be possible to derive unambiguously item b) from item a), and vice-versa, by a searching operation in the user address look-up table.

3.1.2.3.3.2.2.3 In order not to restrict the potential form of an MF-Address, a user address look-up table shall support in the attributes included under item b) all the general attribute types authorized in ISO/IEC 10021-2, section 18.5, Table 10.

3.1.2.3.4 AFTN to AMHS Conversion

Note.— This clause specifies the actions to be performed by an AFTN/AMHS Gateway upon reception of messages from the AFTN for conveyance in the AMHS, after the accomplishment of the AFTN-related procedures by the AFTN Component as specified in 3.1.2.3.2.1.

3.1.2.3.4.1 Control function

3.1.2.3.4.1.1 Upon reception by the Message Transfer and Control Unit of a message passed from the AFTN Component, as the result of the provisions of 3.1.2.3.2.1.4 items a) and b), and of 3.1.2.3.2.1.6, the received message shall be processed in one of three mutually exclusive manners depending on the message category:

a) processing as specified in 3.1.2.3.4.3, if the received message is an AFTN acknowledgement message as specified in Annex 10, Volume II, 4.4.15.6;

b) processing as specified in 3.1.2.3.4.4, if the received message is an AFTN service message requesting correction by the originator of a message received with an unknown addressee indicator as specified in Annex 10, Volume II, 4.4.11.13.3; or

c) processing as specified in 3.1.2.3.4.2, if the received message is other than those referred to in a) and b) above.

3.1.2.3.4.1.2 Upon completion of the processing specified in 3.1.2.3.4.1.1, the following transfers shall take place:

a) transfer of the resulting AMHS information objects, if any, to the ATN Component for conveyance in the AMHS; and

b) transfer of the resulting AFTN service messages, if any, to the AFTN Component for conveyance over the AFTN.

3.1.2.3.4.1.3 If, for any reason, the processing specified in clauses 3.1.2.3.4.1.1 and 3.1.2.3.4.1.2 cannot be properly achieved, the procedure shall unsuccessfully terminate, resulting in:

a) logging of the error situation and reporting to a control position; and

b) storage of the AFTN message for appropriate action at the control position.

3.1.2.3.4.2 Conversion of AFTN Messages

Upon reception by the Message Transfer and Control Unit of an AFTN message passed from the AFTN Component to be conveyed over the AMHS, this AFTN message shall be converted into an IPM conveyed with a Message Transfer Envelope to be transferred and delivered in the AMHS in compliance with the following:

a) the specification of how the components of the AFTN Message are used for mapping onto the AMHS message parameters, as included in 3.1.2.3.4.2.1;

b) the specification of how the IPM is generated, as included in 3.1.2.3.4.2.2; and

c) the specification of how the Message Transfer Envelope elements are generated, as included in 3.1.2.3.4.2.3.

3.1.2.3.4.2.1 Use of AFTN Message components

3.1.2.3.4.2.1.1 Each component of an AFTN Message shall be processed as specified in the column "action" of Table 3.1.2-3.

3.1.2.3.4.2.1.2 These components which are classified as "T" or "T1" in the column "action" of Table 3.1.2-3 shall be translated into the AMHS parameter specified in the column "AMHS parameter" of Table 3.1.2-3 and according to the specification in the clause referred to in the column "mapping".

Table 3.1.2-3. Use of AFTN Message Components

AFTN Message Part	Component	Action	AMHS parameter	Mapping
Heading	Start-of-Heading Character	-	-	–
	Transmission Identification	D	-	-
Address	Alignment Function	-	-	-
	Priority Indicator	T	ATS-Message-Priority (see Table 3.1.2-5/Part 5/1.2) priority (see Table 3.1.2-6/Part 1/1.1.6)	see 3.1.2.3.4.2.1.3
	Addressee Indicator(s)	T	primary-recipients (see Table 3.1.2-5/Part 2/4) recipient-name (see Table 3.1.2-6/Part 1/1.2.1)	see 3.1.2.3.4.2.1.4.2
	Alignment Function	-	-	–
Origin	Filing Time	T	ATS-Message-Filing-Time (see Table 3.1.2-5/ Part 5/1.3)	see 3.1.2.3.4.2.1.5
	Originator Indicator	T	originator (see Table 3.1.2-5/Part 2/2) this-IPM (see Table 3.1.2-5/Part 2/1) originator-name (see Table 3.1.2-6/Part 1/1.1.2)	see 3.1.2.3.4.2.1.4.1
	Priority Alarm	D	-	-
	Optional Heading Information	T1	ATS-Message-Optional-Heading-Info (see Table 3.1.2-5/Part 5/1.4)	see 3.1.2.3.4.2.1.6
	Alignment Function	-	-	-
	Start-of-Text Character	-		-

AFTN Message Part	Component	Action	AMHS parameter	Mapping
Text		T	ATS-Message-Text (see Table 3.1.2-5/Part 5/2)	see 3.1.2.3.4.2.1.7
Ending	Alignment Function	-	-	-
	Page-feed sequence	-	-	-
	End-of-Text Character	-	-	-

Legend (see 3.1.1):
T1 = conditionally translated
D = discarded
T = translated
- = not applicable

3.1.2.3.4.2.1.3 The value of the priority indicator of an AFTN message shall be:

a) mapped into the abstract-value of the *priority* element of the message transfer envelope of the converted AMHS message as specified in the second column of Table 3.1.2-4; and

b) conveyed as the value of the priority-indicator in the ATS-Message-Priority element of the IPM text of the converted AMHS message as specified in the third column of Table 3.1.2-4.

Note.—The transport priority used for the conveyance of AMHS messages is specified in 3.1.2.2.2.1.2.3.

Table 3.1.2-4. Mapping of AFTN Priority Indicator

AFTN Priority Indicator	AMHS Message Transfer Envelope priority	AMHS ATS-Message-Priority priority-indicator
SS	urgent	SS
DD	normal	DD
FF	normal	FF
GG	non-urgent	GG
KK	non-urgent	KK

3.1.2.3.4.2.1.4 The value of an AFTN address included in an AFTN message shall be converted into an MF-Address as respectively specified in 3.1.2.3.4.2.1.4.1 and 3.1.2.3.4.2.1.4.2 depending whether it is an originator indicator or an addressee indicator.

3.1.2.3.4.2.1.4.1 The following actions shall be performed in order to translate the originator indicator of an AFTN Message into the MF-Address included in the *originator-name* of the converted AMHS message:

a) translation into the single MF-Address matching exactly the AF-Address of the originator, if such an MF-Address can be determined from the User address look-up table maintained in the Message Transfer and Control Unit; or

b) if a) cannot be achieved, translation into the XF-address constructed using the single Management Domain identified by the set of *country-name*, *administration-domain-name* and (if any) *private-domain-name* attributes, determined among the entries in the MD look-up table, if any, matching exactly the following character substrings of the AFTN address and selected among these entries, if several are found, on the basis of a decreasing order of precedence from 1) to 4):

1) characters 1 to 7,

2) characters 1, 2, 5, 6 and 7,

3) characters 1, 2, 3 and 4,

4) characters 1 and 2; or

c) if no adequate entry can be found in the MD look-up table, or if the procedure defined in b) does not result in a single resulting MD, unsuccessful termination of the procedure resulting in:

1) logging of the error situation and reporting to a control position, and

2) storage of the AFTN message for appropriate action at the control position.

Note.— The specification above does not constrain the search algorithm provided that the expected result is achieved.

3.1.2.3.4.2.1.4.2 Each addressee indicator of an AFTN Message shall be translated into the MF-Address included in a *recipient-name* of the converted AMHS message in the same way as an originator indicator, with the exception that the unsuccessful termination for one or several addressee indicators additionally results in the generation, in compliance with the provisions of Annex 10, Volume II, 4.4.11.13.3, of an AFTN service message requesting correction by the originator of a message received with an unknown addressee indicator, the unknown addressee indicator(s) included in item 8) of the text message taking the value of these addressee indicators for which the translation process failed.

Note.— A PDAI included in the addressee indicator(s) of an AFTN Message is translated into an MF-Address in the same way as any addressee indicator.

3.1.2.3.4.2.1.5 The value of the Filing Time of an AFTN message shall be conveyed as the value of the filing-time element in the ATS-Message-Filing-Time element of the IPM text of the converted AMHS message.

3.1.2.3.4.2.1.6 The ATS-Message-Optional-Heading-Info element of the IPM text in the converted AMHS message shall either:

a) convey the value of the Optional Heading Information of the AFTN message as the value of its optional-heading-information element, if the Optional Heading Information element is present in the AFTN message; or

b) be omitted in the converted AMHS message, if the Optional Heading Information element is not present in the AFTN message.

3.1.2.3.4.2.1.7 The content of the Text of an AFTN message, shall be conveyed in its entirety as the value of the ATS-Message-Text element in the IPM text of the converted AMHS message.

3.1.2.3.4.2.2 Generation of IPM

3.1.2.3.4.2.2.1 Each of the elements composing the IPM resulting from the conversion of an AFTN message in the Message Transfer and Control Unit shall be processed as specified in the column "action" of Table 3.1.2-5.

3.1.2.3.4.2.2.2 These elements which are classified as "G" or "T" in the column "action" of Table 3.1.2-5 shall be either generated or translated according to the specification in the clause referred to in the column "mapping" of Table 3.1.2-5.

Note.— Table 3.1.2-5 is structured as a PRL derived from the profile specification included in 2.2 and consequently from the ISPICS Proforma included in ISO/IEC ISP 12062-2 (AMH21) as well as from Table 3.1.2-2 in 3.1.2.2.3.2. The columns "Base" and "ISP" under "Origination" are extracted from ISO/IEC ISP 12062-2 and the column "Basic ATS Message Service" specifies the static capability of an IPM AU supporting the Basic ATS Message Service, i.e. the ability to generate the element as part of an IPM carrying an ATS Message. The references to the ISP Profile are indicated in the part titles as AMH21/ref where appropriate. The references in column Ref are those of the ISP.

Table 3.1.2-5. IPM Generation

Ref	Element	Origination			Action	Mapping / Notes
		Base	ISP	Basic ATS Mess. Service		
PART 1 : AMH21/A.1.1 SUPPORTED INFORMATION OBJECTS						
1	Interpersonal message (IPM)	M	M	M	T	see Part 1/1.1 and 1.2
1.1	heading	M	M	M	T	see Part 2
1.2	body	M	M	M	T	see Part 3

Ref	Element	Origination			Action	Mapping / Notes
		Base	**ISP**	**Basic ATS Mess. Service**		
2	Interpersonal Notification (IPN)	M	M	M	-	out of the scope of this clause
PART 2 : AMH21/A.1.2 IPM HEADING FIELDS						
1	this-IPM	M	M	M	T	see Part 4/3
2	originator	M	M	M	T	see 3.1.2.3.4.2.2.3 and Part 4/2
3	authorizing-users	O	O	O	X	-
4	primary-recipients	M	M	M	T	see 3.1.2.3.4.2.2.4 and Part 4/1
5	copy-recipients	M	M	M	X	-
6	blind-copy-recipients	O	O	O	X	-
7	replied-to-IPM	M	M	M	X	-
8	obsoleted-IPMs	O	O	O	X	-
9	related-IPMs	O	O	O	X	-
10	subject	M	M	M	X	-
11	expiry-time	O	O	O	X	-
12	reply-time	O	O	O	X	-
13	reply-recipients	O	O	O	X	-
14	importance	O	O	O	X	-
15	sensitivity	O	O	O	X	-
16	auto-forwarded	O	O	O	X	-
17	extensions	O	O	O	X	-
17.1	incomplete-copy	O	O	O	X	-
17.2	languages	O	O	O	X	-
17.3	auto-submitted	O	I	I	X	-

Ref	Element	Origination			Action	Mapping / Notes
		Base	**ISP**	**Basic ATS Mess. Service**		
PART 3 : AMH21/A.1.3 IPM BODY						
1	ia5-text	O	O	M	T	see Part 3/1.1 and 1.2
1.1	parameters	M	M	M	G	see Part 3/1.1.1
1.1.1	repertoire	O	O	O	G	see 3.1.2.3.4.2.2.5
1.2	data	M	M	M	T	see Part 5
2	voice	I	I	I	X	-
3	g3-facsimile	O	O	O	X	-
4	g4-class-1	O	O	O	X	-
5	teletex	O	O	O	X	-
6	videotex	O	O	O	X	-
7	encrypted	I	I	I	X	-
8	message	O	O	O	X	-
9	mixed-mode	O	O	O	X	-
10	bilaterally-defined	O	O	O	X	-
11	nationally-defined	O	O	O	X	-
12	externally-defined	O	M	M	X	-
PART 4 : AMH21/A.1.5 COMMON DATA TYPES						
1	RecipientSpecifier					
1.1	recipient	M	M	M	T	see 3.1.2.3.4.2.2.6 and Part 4/2
1.2	notification-requests	O	O	M	T	see Part 4/1.2.1-1.2.3
1.2.1	rn	O	O	M	T	see 3.1.2.3.4.2.2.7
1.2.2	nrn	O	O	M	T	see 3.1.2.3.4.2.2.7
1.2.3	ipm-return	O	O	O	X	-
1.3	reply-requested	O	O	O	X	-
1.4	recipient-extensions	O	I	I	X	-
2	ORDescriptor					
2.1	formal-name	M	M1	M	T	see 3.1.2.3.4.2.2.8

Ref	Element	Origination			Action	Mapping / Notes
		Base	**ISP**	**Basic ATS Mess. Service**		
2.2	free-form-name	O	O	O	X	-
2.3	telephone-number	O	O	O	X	-
3	IPMIdentifier					
3.1	user	M	M	M	T	see 3.1.2.3.4.2.2.9
3.2	user-relative-identifier	M	M	M	G	-
PART 5 : IPM SUPPORT OF THE BASIC ATS MESSAGE SERVICE						
1	ATS-Message-Header	-	-	M	T	see Part 5/1.1-1.6
1.1	start-of-heading	-	-	M	G	see 3.1.2.2.3.2
1.2	ATS-Message-Priority	-	-	M	T	see Part 5/1.2.1-1.2.3
1.2.1	priority-prompt	-	-	M	G	see 3.1.2.2.3.2
1.2.2	priority-indicator	-	-	M	T	see 3.1.2.3.4.2.1.3
1.2.3	priority-separator	-	-	M	G	see 3.1.2.2.3.2
1.3	ATS-Message-Filing-Time	-	-	M	T	see Part 5/1.3.1-1.3.3
1.3.1	filing-time-prompt	-	-	M	G	see 3.1.2.2.3.2
1.3.2	filing-time	-	-	M	T	see 3.1.2.3.4.2.1.5
1.3.3	filing-time-separator	-	-	M	G	see 3.1.2.2.3.2
1.4	ATS-Message-Optional-Heading-Info	-	-	O	T1	see Part 5/1.4.1-1.4.3
1.4.1	OHI-prompt	-	-	M	G	see 3.1.2.2.3.2
1.4.2	optional-heading-information	-	-	M	T	see 3.1.2.3.4.2.1.6
1.4.3	OHI-separator	-	-	M	G	see 3.1.2.2.3.2

Ref	Element	Origination			Action	Mapping / Notes
		Base	ISP	Basic ATS Mess. Service		
1.5	end-of-heading-blank-line	-	-	M	G	see 3.1.2.2.3.2
1.6	start-of-text	-	-	M	G	see 3.1.2.2.3.2
2	ATS-Message-Text	-	-	M	T	see 3.1.2.3.4.2.1.7

Legend (see 3.1.1) :
M = mandatory support
M1 = minimal O/R name mandatory support
O = optional support
I = out of scope
- = not applicable
G = generated
T = translated
T1 = conditionally translated
X = excluded (not used)

3.1.2.3.4.2.2.3 The *originator* heading field shall:

a) identify the indirect AMHS user who originated the AFTN message; and

b) be structured as specified in Table 3.1.2-5/ Part 4/2.

3.1.2.3.4.2.2.4 The *primary-recipients* heading field shall:

a) include the identification of the recipient(s) of the AFTN message; and

b) be structured as specified in Table 3.1.2-5/ Part 4/1.

3.1.2.3.4.2.2.5 The element *repertoire* shall take the abstract value “ia5”.

3.1.2.3.4.2.2.6 The element(s) *recipient* in the *primary-recipients* heading field shall:

a) identify the recipient(s) of the AFTN message; and

b) be structured as specified in Table 3.1.2-5/ Part 4/2.

3.1.2.3.4.2.2.7 The values “rn” and “nrn” shall be taken simultaneously by the element *notification-requests* if, and only if the element *priority-indicator* included in the message, as specified Table 3.1.2-5 / Part 5/1.2.2, has the value “SS”.

3.1.2.3.4.2.2.8 The element *formal-name* shall:

a) take the form of an MF-Address; and

b) be converted as specified in 3.1.2.3.4.2.1.4.

3.1.2.3.4.2.2.9 The element *user* in the *this-IPM* heading field shall:

a) be the MF-Address of the indirect AMHS user who originated the AFTN message; and

b) be converted as specified in 3.1.2.3.4.2.1.4.1.

3.1.2.3.4.2.3 Generation of Message Transfer Envelope

3.1.2.3.4.2.3.1 Each of the elements composing the Message Transfer Envelope conveyed with an IPM resulting from the conversion of an AFTN message shall be processed as specified in the column "action" of Table 3.1.2-6.

3.1.2.3.4.2.3.2 These elements which are classified as "G", "G1" and "T" in the column "action" of Table 3.1.2-6 shall be handled according to the specification in the clause referred to in the column "mapping" of Table 3.1.2-6.

Note 1.— Table 3.1.2-6 is structured as a PRL derived from the ISPICS Proforma included in ISO/IEC ISP 10611-3. The columns "Base" and "ISP" are extracted from ISO/IEC ISP 10611-3, and the column "Basic ATS Message Service" specifies the static capability of an AU, for the MT-Elements of Service, i.e. the ability to convey, handle and act in relation with the element. The references to the ISP Profile are indicated in the part titles as AMH11/ref where appropriate.

Table 3.1.2-6. MessageTransfer for conveyance of an IPM

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
PART 1 : AMH11/A.1.4.2 MESSAGETRANSFER						
1	MessageTransferEnvelope	M	M	M	T	see Part 1/1.1 and 1.2
1.1	(per message fields)					
1.1.1	message-identifier	M	M	M	G	see Part 2/1
1.1.2	originator-name	M	M	M	T	see 3.1.2.3.4.2.3.3
1.1.3	original-encoded-information-types	M	M-	M-	G	see 3.1.2.3.4.2.3.4 and Part 2/3
1.1.4	content-type	M	M-	M-	G	see 3.1.2.3.4.2.3.5 and Part 2/8

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
1.1.5	content-identifier	M	M	M	G1	see 3.1.2.3.4.2.3.6
1.1.6	priority	M	M	M	T	see 3.1.2.3.4.2.1.3
1.1.7	per-message-indicators	M	M	M	G	see Part 2/4
1.1.8	deferred-delivery-time	O	M-	M-	X	-
1.1.9	per-domain-bilateral-information	O	M-	M-	G1	see 3.1.2.3.4.2.3.7 and Part 2/5
1.1.10	trace-information	M	M	M	G	see Part 2/6
1.1.11	extensions	M	M	M	G/X	see 3.1.2.3.4.2.3.8 and Part 3/1
1.1.11.1	recipient-reassignment-prohibited	O	M	M	X	-
1.1.11.2	dl-expansion-prohibited	O	M	M	X	-
1.1.11.3	conversion-with-loss-prohibited	O	M	M	X	-
1.1.11.4	latest-delivery-time	O	M-	M-	X	-
1.1.11.5	originator-return-address	O	M-	M-	X	-
1.1.11.6	originator-certificate	O	M-	M-	X	-
1.1.11.7	content-confidentiality-algorithm-identifier	O	M-	M-	X	-
1.1.11.8	message-origin-authentication-check	O	M-	M-	X	-
1.1.11.9	message-security-label	O	M-	M-	X	-
1.1.11.10	content-correlator	M	M	M	G1	see 3.1.2.3.4.2.3.6

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
1.1.11.11	dl-expansion-history	M	M-	M-	X	see Note 2
1.1.11.12	internal-trace-information	M	M	M	G	see Part 3/5
1.2	per-recipient-fields	M	M	M	T	see Part 1/1.2.1-1.2.5
1.2.1	recipient-name	M	M	M	T	see 3.1.2.3.4.2.3.9
1.2.2	originally-specified-recipient-number	M	M	M	G	see 3.1.2.3.4.2.3.10
1.2.3	per-recipient-indicators	M	M	M	G	see 3.1.2.3.4.2.3.11
1.2.4	explicit-conversion	O	M-	M-	X	-
1.2.5	extensions	M	M	M	X	-
2	content	M	M	M	T	see 3.1.2.3.4.2.2
PART 2 : AMH11/A.1.5 COMMON DATA TYPES						
1	MTSIdentifier					
1.1	global-domain-identifier	M	M	M	G	see 3.1.2.3.4.2.3.12 and Part 2/2
1.2	local-identifier	M	M	M	G	see 3.1.2.3.4.2.3.13
2	GlobalDomainIdentifier					
2.1	country-name	M	M	M	G	see 3.1.2.3.4.2.3.14
2.2	administration-domain-name	M	M	M	G	see 3.1.2.3.4.2.3.15

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
2.3	private-domain-identifier	M	M	M	G	see 3.1.2.3.4.2.3.16
3	EncodedInformationTypes					
3.1	built-in-encoded-information-types	M	M	M	G	see 3.1.2.3.4.2.3.4
3.2	(non-basic parameters)	O	M-	M-	X	-
3.3	extended-encoded-information-types	M	M	M	X	-
4	PerMessageIndicators					
4.1	disclosure-of-other-recipients	M	M	M	G	see 3.1.2.3.4.2.3.17
4.2	implicit-conversion-prohibited	M	M	M	G	see 3.1.2.3.4.2.3.18
4.3	alternate-recipient-allowed	M	M	M	G	see 3.1.2.3.4.2.3.19
4.4	content-return-request	O	M-	M-	X	see 3.1.2.3.4.2.3.20
4.5	reserved	O	M-	M-	X	-
4.6	bit-5	O	M-	M-	X	-
4.7	bit-6	O	M-	M-	X	-
4.8	service-message	O	M-	M-	X	-
5	PerDomainBilateralInformation					
5.1	country-name	M	M-	M-	G1	see 3.1.2.3.4.2.3.21
5.2	administration-domain-name	M	M-	M-	G1	see 3.1.2.3.4.2.3.21
5.3	private-domain-identifier	O	M-	M-	G1	see 3.1.2.3.4.2.3.21

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
5.4	bilateral-information	M	M-	M-	G1	see 3.1.2.3.4.2.3.22
6	TraceInformation					
6.1	TraceInformationElement	M	M	M	G	see Part 2/6.1.1 and 6.1.2
6.1.1	global-domain-identifier	M	M	M	G	see 3.1.2.3.4.2.3.23 and Part 2/2
6.1.2	domain-supplied-information	M	M	M	G	see Part 2/6.1.2.1-6.1.2.4
6.1.2.1	arrival-time	M	M	M	G	see 3.1.2.3.4.2.3.24
6.1.2.2	routing-action	M	M	M	G	see Part 2/6.1.2.2.1 and 6.1.2.2.2
6.1.2.2.1	relayed	M	M	M	G	see 3.1.2.3.4.2.3.25
6.1.2.2.2	rerouted	O	C1	C1	X	see Note 3
6.1.2.3	attempted-domain	O	C1	C1	X	see Note 3
6.1.2.4	(additional actions)					
6.1.2.4.1	deferred-time	M	C2	C2	X	-
6.1.2.4.2	converted-encoded-information-types	O	M-	M-	X	-
6.1.2.4.3	other-actions	O	M-	M-	X	-
6.1.2.4.3.1	redirected	O	M-	M-	X	see Note 4
6.1.2.4.3.2	dl-operation	O	M-	M-	X	see Note 2

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
8	ContentType					
8.1	built-in	M	M-	M-	G	see 3.1.2.3.4.2.3.5
8.2	extended	O	M-	M-	X	-
PART 3 : AMH11/A.1.6 EXTENSION DATA TYPES						
1	ExtensionField					
1.1	type	M	M	M	G	see Part 3/1.1.1 and 1.1.2
1.1.1	standard-extension	M	M	M	G	see 3.1.2.3.4.2.3.8
1.1.2	private-extension	O	M-	M-	X	-
1.2	criticality	M	M	M	G	see 3.1.2.3.4.2.3.8
1.3	value	M	M	M	G	see 3.1.2.3.4.2.3.8
5	InternalTraceInformation					
5.1	global-domain-identifier	M	M	M	G	see 3.1.2.3.4.2.3.23
5.2	mta-name	M	M	M	G	see 3.1.2.3.4.2.3.26
5.3	mta-supplied-information	M	M	M	G	see Part 3/5.3.1-5.3.4
5.3.1	arrival-time	M	M	M	G	see 3.1.2.3.4.2.3.24
5.3.2	routing-action	M	M	M	G	see Part 3/5.3.2.1-5.3.2.2
5.3.2.1	relayed	M	M	M	G	see 3.1.2.3.4.2.3.25
5.3.2.2	rerouted	O	C1	C1	X	see Note 3

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
5.3.3	attempted	O	C1	C1	X	see Note 3
5.3.4	(additional actions)					
5.3.4.1	deferred-time	M	C2	C2	X	-
5.3.4.2	converted-encoded-information-types	O	M-	M-	X	-
5.3.4.3	other-actions	O	M-	M-	X	-
5.3.4.3.1	redirected	O	M-	M-	X	see Note 4
5.3.4.3.2	dl-operation	O	M-	M-	X	see Note 2

Legend (see 3.1.1):
M = mandatory support
M- = minimal mandatory support
O = optional support
I = out of scope
- = not applicable
C1 = if rerouting is supported then M else M-
C2 = if deferred delivery is supported then M else M-
G = generated
G1 = optionally generated
T = translated
X = excluded

Note 2.— The DL-expansion capability of an AFTN/AMHS Gateway is implemented in the ATN Component rather than in the Message Transfer and Control Unit.

Note 3.— The rerouting capability of an AFTN/AMHS Gateway, if any, is implemented in the ATN Component rather than in the Message Transfer and Control Unit.

Note 4.— The redirection capability of an AFTN/AMHS Gateway, if any, is implemented in the ATN Component rather than in the Message Transfer and Control Unit.

3.1.2.3.4.2.3.3 The value of the element *originator-name* shall:

a) be the address of the indirect AMHS user who originated the AFTN message;

b) take the form of an MF-Address; and

c) be converted as specified in 3.1.2.3.4.2.1.4.1.

3.1.2.3.4.2.3.4 The element *original-encoded-information-types* shall:

a) take the abstract-value “ia5-text”, which is a value of type BuiltInEncodedInformationTypes; and

b) be formed as specified in Table 3.1.2-6/ Part 2/ 3.

3.1.2.3.4.2.3.5 The element *content-type* shall:

a) take the abstract-value “interpersonal-messaging-1984”, which is a value of type BuiltInContentType; and

b) be formed as specified in Table 3.1.2-6/ Part 2/ 8.

3.1.2.3.4.2.3.6 The generation of this element shall be optional, as a matter of policy local to the AMHS Management Domain operating the AFTN/AMHS Gateway.

3.1.2.3.4.2.3.7 The element *per-domain-bilateral-information* shall be:

a) optionally generated, as a matter of policy local to the AMHS Management Domain operating the AFTN/AMHS Gateway; and

b) if present, structured as specified in Table 3.1.2-6/ Part 2/ 5.

3.1.2.3.4.2.3.8 The only extensions used shall:

a) belong to the type “standard-extension”;

b) contain the following elements:

1) *content-correlator*, if used; and

2) *internal-trace-information*;

3) *conversion-with-loss-prohibited* elements;

e) take a criticality value as specified in ISO/IEC 10021-4, Figure 2; and

f) take values as specified in 3.1.2.3.4.2.3.6 and Table 3.1.2-6/Part 3/5, respectively.

Note.— The non-use of the elements recipient-reassignment-prohibited, dl-expansion-prohibited and conversion-with-loss-prohibited implies, in compliance with ISO/IEC 10021-4, that they are assumed to take their default abstract-values, which are “recipient-reassignment allowed”, “DL-expansion-allowed” and “conversion-with-loss-allowed”, respectively.

3.1.2.3.4.2.3.9 The value of the element *recipient-name* in each of the *per-recipient-fields* elements shall:

a) be the address of each addressee indicated in the AFTN message, respectively;

b) take the form of a MF-Address; and

c) be converted as specified in 3.1.2.3.4.2.1.4.2.

3.1.2.3.4.2.3.10 The value of the element *originally-specified-recipient-number* in each of the *per-recipient-fields* elements shall be generated by the Message Transfer and Control Unit as specified in ISO/IEC 10021-4, 12.2.1.1.1.5.

3.1.2.3.4.2.3.11 The components of the element *per-recipient-indicators* in each of the *per-recipient-fields* elements shall be generated taking the following abstract-values:

a) "responsible" for the *responsibility* element;

b) "non-delivery-report" for the *originating-MTA-report-request* element; and

c) "non-delivery-report" for the *originator-report-request* element.

3.1.2.3.4.2.3.12 The element *global-domain-identifier* in the *MTS-identifier* shall:

a) identify the AMHS Management Domain operating the AFTN/AMHS Gateway; and

b) be composed as specified in Table 3.1.2-6 / Part 2/2.

3.1.2.3.4.2.3.13 The element *local-identifier* in the *MTS-identifier* shall be generated locally so as to ensure that it distinguishes the message from all other messages, probes or reports generated in the AMHS Management Domain operating the AFTN/AMHS Gateway.

3.1.2.3.4.2.3.14 The element *country-name* in the *global-domain-identifier* element of the *MTS-identifier* and of the first *trace-information-element* shall:

a) be part of the identification of the AMHS Management Domain operating the AFTN/AMHS Gateway by taking one of the following values:

1) the two-character alphabetical country-indicator as specified in ISO 3166 for the country, or for one of the countries, where the AMHS Management Domain has been registered, if the AMHS Management Domain has been subject to national or multi-national registration; or

2) a two-character alphabetical indicator dedicated to an international organization, if the AMHS Management Domain has been subject to international registration as specified in ITU-T Recommendation X.666; and

b) be encoded as a Printable String.

3.1.2.3.4.2.3.15 The element *administration-domain-name* in the *global-domain-identifier* element of the *MTS-identifier* and of the first *trace-information-element* shall:

a) be part of the identification of the AMHS Management Domain operating the AFTN/AMHS Gateway by taking one of the following values, depending on its status:

1) the name of the ADMD under which the AMHS Management Domain has been registered, either nationally or internationally, if the AMHS Management Domain operates as an ADMD;

2) the name of the ADMD to which the AMHS Management Domain is connected, if the AMHS Management Domain operates as a PRMD; or

3) the value single-space if the AMHS Management Domain operates as a PRMD and is unique with regard to the country-name identifying the area where it is registered, either nationally or internationally; and

b) be encoded as a Printable String.

3.1.2.3.4.2.3.16 The element *private-domain-identifier* in the *global-domain-identifier* element of the *MTS-identifier* and of the first *trace-information-element* shall be handled in one of the following manners, depending on the status under which the AMHS Management Domain operates:

a) generation of the element, with the value of the name of the PRMD, encoded as a Printable String, if the AMHS Management Domain operates as an PRMD; or

b) omission in the *global-domain-identifier* if the AMHS Management Domain operates as an ADMD.

3.1.2.3.4.2.3.17 The element *disclosure-of-other-recipients* shall take its default abstract-value, which is “disclosure-of-other-recipients-prohibited”.

3.1.2.3.4.2.3.18 The element *implicit-conversion-prohibited* shall take its default abstract-value, which is “implicit-conversion-allowed”.

3.1.2.3.4.2.3.19 The element *alternate-recipient-allowed* shall take the abstract-value “alternate-recipient-allowed”.

3.1.2.3.4.2.3.20 The element *content-return-request* shall take its default abstract-value, which is “content-return-not-requested”.

3.1.2.3.4.2.3.21 The elements *country-name*, *administration-domain-name* and *private-domain-identifier* shall together identify the AMHS Management Domain for which the bilateral-information is intended if, and only if, the element *bilateral-information* as specified in 3.1.2.3.4.2.3.22 is present.

3.1.2.3.4.2.3.22 The generation of this element shall be optional, as a matter of bilateral agreement between the AMHS Management Domain operating the AFTN/AMHS Gateway and an other AMHS Management Domain.

3.1.2.3.4.2.3.23 The element *global-domain-identifier* in the *trace-information* or in the *internal-trace-information* shall:

a) identify the AMHS Management Domain operating the AFTN/AMHS Gateway; and

b) be composed as specified in Table 3.1.2-6 / Part 2/2.

3.1.2.3.4.2.3.24 The element *arrival-time* in the first element of *trace-information* or of *internal-trace-information* shall take the semantic value of the time when the message was received by the Message Transfer and Control Unit for conveyance in the AMHS.

3.1.2.3.4.2.3.25 The element *routing-action* in the first element of *trace-information* or of *internal-trace-information* shall take the abstract-value “relayed”.

3.1.2.3.4.2.3.26 The element *mta-name* in the first element of *internal-trace-information* shall be the mta-name assigned to the Message Transfer and Control Unit included in the AFTN/AMHS Gateway.

Note.— The structure of the mta-name of the Message Transfer and Control Unit included in an AFTN/AMHS Gateway within an AMHS Management Domain is a matter of policy internal to the AMHS Management Domain.

3.1.2.3.4.3 Conversion of AFTN Acknowledgement Messages

3.1.2.3.4.3.1 Initial processing of AFTN Acknowledgement Message

3.1.2.3.4.3.1.1 Upon reception by the Message Transfer and Control Unit of an AFTN acknowledgement message, passed from the AFTN Component to be conveyed in the AMHS, the received message shall be processed in one of the following manners depending on whether or not the subject AFTN message previously passed through the Message Transfer and Control Unit:

a) processing as specified in 3.1.2.3.4.3.1.2, if the subject AFTN message, as identified in the text of AFTN acknowledgement message, previously passed through the Message Transfer and Control Unit; or

b) processing as follows, if the subject AFTN message did not previously pass through the Message Transfer and Control Unit:

1) logging of the error situation and reporting to a control position; and

2) conversion of the AFTN acknowledgement message into an IPM conveyed with a Message Transfer Envelope as specified in 3.1.2.3.4.3.1.5.

3.1.2.3.4.3.1.2 If the subject AFTN message previously passed through the Message Transfer and Control Unit, the AFTN acknowledgement message shall then be processed in one of the following manners depending on whether the subject IPM was received from the AMHS without or with *receipt-notification-request*:

a) processing as follows, if the subject IPM was received from the AMHS without *receipt-notification-request*:

1) conversion into an IPM conveyed with a Message Transfer Envelope as specified in 3.1.2.3.4.3.1.5; and

2) logging of the error situation and reporting to a control position; or

b) processing as specified in 3.1.2.3.4.3.1.3, if the subject IPM was received from the AMHS with *receipt-notification-request*.

3.1.2.3.4.3.1.3 If the subject IPM had been received from the AMHS with *receipt-notification-request*, the AFTN acknowledgement message shall be converted by the AFTN/AMHS Gateway into an Interpersonal Notification (IPN) taking the form of a Receipt Notification (RN), conveyed with a Message Transfer Envelope generated in compliance with the provisions of 3.1.2.3.4.3.1.4.

3.1.2.3.4.3.1.4 When the provisions of 3.1.2.3.4.3.1.3 apply, the generation of the RN and of the Message Transfer Envelope shall be performed in compliance with the following:

a) the specification of how the components of the AFTN Service Message are used, as included in 3.1.2.3.4.3.2;

b) mthe specification of how the RN is generated, as included in 3.1.2.3.4.3.3; and

c) the provisions of 3.1.2.3.4.2.3 concerning the generation of the Message Transfer Envelope, with the exception of the differences specified in 3.1.2.3.4.3.4.

3.1.2.3.4.3.1.5 When an acknowledgement message is converted into an IPM as the result of 3.1.2.3.4.3.1.1 or 3.1.2.3.4.3.1.2, the specification of 3.1.2.3.4.2 shall apply with the exception of the *subject* element in the IPM heading fields, initially specified in Table 3.1.2-5/Part 2/10, which is then generated and takes the value "AFTN service information".

3.1.2.3.4.3.2 Use of AFTN Service Message components

3.1.2.3.4.3.2.1 Each component of an AFTN acknowledgement message shall be processed for the generation of a RN as specified in the column "action" of Table 3.1.2-7.

3.1.2.3.4.3.2.2 These components which are classified as "T" or "T1" in the column "action" of Table 3.1.2-7 shall be translated into the AMHS parameter specified in the column "AMHS parameter" of Table 3.1.2-7 and according to the specification in the clause referred to in the column "mapping".

Table 3.1.2-7. Use of AFTN Service Message Components

AFTN Message Part	Component	Action	AMHS parameter	Mapping
Heading	Start-of-Heading Character	-	-	-
	Transmission Identification	D	-	-
Address	Alignment Function	-	-	-
	Priority Indicator	T	priority (see Table 3.1.2-9/Part 1/1.1.6)	see 3.1.2.3.4.3.4.3
	Addressee Indicator	T	recipient-name (see Table 3.1.2-9/Part 1/1.2.1)	see 3.1.2.3.4.3.4.4
	Alignment Function	-	-	-
Origin	Filing Time	T	receipt-time (see Table 3.1.2-8/Part 2/7.1)	see 3.1.2.3.4.3.2.4

AFTN Message Part	**Component**	**Action**	**AMHS parameter**	**Mapping**
	Originator Indicator	T	ipn-originator (see Table 3.1.2-8/Part 2/2) originator-name (see Table 3.1.2-6/Part 1/1.1.2)	see 3.1.2.3.4.3.2.3 see 3.1.2.3.4.2.1.4.1
	Priority Alarm	D	-	-
	Optional Heading Information	D	-	-
	Alignment Function	-	-	-
	Start-of-Text Character	-	-	-
Text		D	-	-
Ending	Alignment Function	-	-	-
	Page-feed sequence	-	-	-
	End-of-Text Character	-	-	-

Legend: (see 3.1.1.)
D = discarded
T = translated
- = not applicable

3.1.2.3.4.3.2.3 Upon generation of a RN as the result of the receipt of an AFTN acknowledgement message by the Message Transfer and Control Unit, the originator indicator element of the AFTN acknowledgement message shall be translated into the *ipn-originator* element of the RN.

3.1.2.3.4.3.2.4 Upon generation of a RN as the result of the receipt of an AFTN acknowledgement message by the Message Transfer and Control Unit, the filing time of the AFTN acknowledgement message shall be converted into the *receipt-time* element, which is of ASN.1 (Abstract syntax notation one) type UTCTime, as the result of the following:

a) generation by the Message Transfer and Control Unit of the YY figures identifying the year (characters 1 and 2 of the string) in the *receipt-time* element;

b) generation by the Message Transfer and Control Unit of the MM figures identifying the month (characters 3 and 4 of the string) in the *receipt-time* element;

c) mapping of the value of the first two figures of the date-time group into the value of the DD figures identifying the day (characters 5 and 6 of the string) in the *receipt-time* element;

d) mapping of the value of the four last figures of the date-time group, which together represent the hours and minutes, into the value of the hhmm figures (characters 7 to 10 of the string) in the *receipt-time* element; and

e) addition by the Message Transfer and Control Unit of an eleventh and last character in the string composing the *receipt-time* element taking the value "Z".

3.1.2.3.4.3.3 Generation of RN

3.1.2.3.4.3.3.1 Each of the elements composing the RN resulting from the receipt of an AFTN acknowledgement message in the Message Transfer and Control Unit shall be processed as specified in the column "action" of Table 3.1.2-8.

3.1.2.3.4.3.3.2 These elements are classified as "G" or "T" in the column "action" of Table 3.1.2-8 shall be either generated or translated according to the specification in the clause referred to in the column "mapping" of Table 3.1.2-8.

Note.— Table 3.1.2-8 is structured as a PRL derived from the profile specification included in 2.2 and consequently from the ISPICS Proforma included in ISO/IEC ISP 12062-2 (AMH21). The columns "Base" and "ISP" under "Origination" are extracted from ISO/IEC ISP 12062-2, and the column "Basic ATS Message Service" specifies the static capability of an IPM AU supporting the Basic ATS Message Service, i.e. the ability to generate the element as part of an IPN in the AMHS. The references to the ISP Profile are indicated in the part titles as AMH21/ref where appropriate. The references in column Ref are those of the ISP.

Table 3.1.2-8. RN Generation

Ref	Element	Origination			Action	Mapping / Notes
		Base	ISP	Basic ATS Mess. Service		
PART 1 : AMH21/A.1.1 SUPPORTED INFORMATION OBJECTS						
1	Interpersonal Message (IPM)	M	M	M	-	out of the scope of this clause
2	Interpersonal Notification (IPN)	M	M	M		see Part 2
PART 2 : AMH21/A.1.4 IPN FIELDS						
1	subject-ipm	M	M	M	G	see 3.1.2.3.4.3.3.3
2	ipn-originator	O	M	M	T	see 3.1.2.3.4.3.2.3 and Part 3/2
3	ipm-preferred-recipient	M	M	M	G2	see 3.1.2.3.4.3.3.4
4	conversion-eits	O	O	O	G2	see 3.1.2.3.4.3.3.5
5	notification-extensions	O	I	I	X	-
6	non-receipt-fields	M	M	M	X	-
7	receipt-fields	O	O	O	T	see Part 2/7.1-7.4
7.1	receipt-time	M	M	M	T	see 3.1.2.3.4.3.2.4
7.2	acknowledgment-mode	O	O	O	G	see 3.1.2.3.4.3.3.6
7.3	suppl-receipt-info	O	O	O	X	-
7.4	rn-extensions	O	I	I	X	-
8	other-notification-type-fields	O	I	I	X	-

Ref	Element	Origination			Action	Mapping / Notes
		Base	**ISP**	**Basic ATS Mess. Service**		
PART 3 : AMH21/A.1.5 COMMON DATA TYPES						
2	ORDescriptor					
2.1	formal-name	M	M1	M	T	see 3.1.2.3.4.3.3.7
2.2	free-form-name	O	O	O	X	
2.3	telephone-number	O	O	O	X	

Legend (see 3.1.1) :
M = mandatory support
M1 = minimal O/R name mandatory support
O = optional support
I = out of scope
G = generated
G2 = conditionally generated
T = translated
X = excluded (not used)

3.1.2.3.4.3.3.3 The element *subject-ipm* shall take the value of the *this-IPM* heading field of the subject IPM.

3.1.2.3.4.3.3.4 The element *ipm-preferred-recipient* shall:

a) be present if, and only if:

1) it would be different from the *ipn-originator* specified in 3.1.2.3.4.3.2.3; and

2) it would not be the result of a DL-expansion;

b) if present, identify the recipient of the subject IPM which caused the receipt of the AFTN acknowledgement message by the Message Transfer and Control Unit (as a result of the receipt by its addressee of the subject AFTN message); and

c) if present, be the *O/R descriptor* of the recipient of the subject IPM.

3.1.2.3.4.3.3.5 The element *conversion-eits* shall:

a) be present if, and only if, this encoded-information-types is different of the *originally-encoded-information-types* included in the subject IPM; and

b) if present, take the value of the encoded-information-types of the subject IPM received by the Message Transfer and Control Unit.

3.1.2.3.4.3.3.6 The element *acknowledgement-mode* shall take the abstract-value "manual", which is its default value.

3.1.2.3.4.3.3.7 The element *formal-name* in an *ORDescriptor* shall take the form of an O/R address and be converted from the originator indicator of the AFTN acknowledgement message as specified in 3.1.2.3.4.2.1.4.1.

3.1.2.3.4.3.4 Differences in the generation of Message Transfer Envelope

3.1.2.3.4.3.4.1 The elements composing the Message Transfer Envelope which is conveyed with a RN resulting from the receipt of an AFTN acknowledgement message by the Message Transfer and Control Unit, which are different from the specification of 3.1.2.3.4.2.3 shall be processed according to the specification in the clause referred to in the column "mapping" of Table 3.1.2-9.

3.1.2.3.4.3.4.2 An element subject to the provisions of 3.1.2.3.4.3.4.1 shall be processed as specified in the column "action" of Table 3.1.2-9, and in accordance with the specification referred to in the column "mapping" of Table 3.1.2-9.

Note.— Table 3.1.2-9 is structured as an extract of Table 3.1.2-6. The references used in the part titles and in the column "Ref" are those of Table 3.1.2-6.

Table 3.1.2-9. MessageTransfer Envelope generation for conveyance with a RN (Differences with Table 3.1.2-6)

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
PART 1 : AMH11/A.1.4.2 MESSAGETRANSFER						
1	MessageTransferEnvelope	M	M	M	T	see Part 1/1.1 and 1.2
1.1	(per message fields)					
1.1.3	original-encoded-information-types	M	M-	M-	X	-
1.1.6	priority	M	M	M	G	see 3.1.2.3.4.3.4.3

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
1.1.7	per-message-indicators	M	M	M	G	see Part 2/4
1.2	per-recipient-fields	M	M	M	T	see Part 1/1.2.1 and 1.2.3
1.2.1	recipient-name	M	M	M	T	see 3.1.2.3.4.3.4.4
1.2.3	per-recipient-indicators	M	M	M	G	see 3.1.2.3.4.3.4.5
2	content	M	M	M	T	see 3.1.2.3.4.3.3
PART 2 : AMH11/A.1.5 COMMON DATA TYPES						
4	PerMessageIndicators					
4.2	implicit-conversion-prohibited	M	M	M	G	see 3.1.2.3.4.3.4.6

Legend (see 3.1.1) :
M = mandatory support
M- = minimal mandatory support
G = generated
T = translated
X = excluded (not used)

3.1.2.3.4.3.4.3 The element *priority* shall take the abstract-value “urgent”.

3.1.2.3.4.3.4.4 The element *recipient-name* shall:

a) identify the originator of the subject IPM; and

b) take the form of an MF-Address.

3.1.2.3.4.3.4.5 The components of the element *per-recipient-indicators* shall be generated taking the following abstract-values:

a) “responsible” for the *responsibility* element;

b) “non-delivery-report” for the *originating-MTA-report-request* element; and

c) “no-report” for the *originator-report-request* element.

3.1.2.3.4.3.4.6 The element *implicit-conversion-prohibited* shall take the abstract-value “implicit-conversion-prohibited”.

3.1.2.3.4.4 Conversion of AFTN Service Messages related to unknown addressee indicators

3.1.2.3.4.4.1 Initial Processing of the AFTN Service Message

3.1.2.3.4.4.1.1 Upon reception by the Message Transfer and Control Unit of an unknown address AFTN service message, passed from the AFTN Component to be conveyed in the AMHS, the received message shall be processed in one of the following manners:

a) processing as specified in 3.1.2.3.4.4.1.2, if the subject AFTN message, as identified in the unknown address AFTN service message text, previously passed through the Message Transfer and Control Unit; or

b) if the subject AFTN message did not previously pass through the Message Transfer and Control Unit, conversion of the unknown address AFTN service message into an IPM conveyed with a Message Transfer Envelope as specified in 3.1.2.3.4.4.1.7.

3.1.2.3.4.4.1.2 If the subject AMHS message previously passed through the Message Transfer and Control Unit, the received message shall be processed in either of the following manners depending on whether or not the unknown addressee indicator(s) which caused the generation of the unknown address AFTN service message can be determined:

a) processing as specified in 3.1.2.3.4.4.1.3, if at least one valid addressee indicator which caused the generation of the unknown address AFTN service message can be found; or

b) if no such valid addressee indicator can be found, conversion of the unknown address AFTN service message into an IPM conveyed with a Message Transfer Envelope as spcified in 3.1.2.3.4.4.1.7.

3.1.2.3.4.4.1.3 For the addressee indicators determined as causing the generation of the unknown address AFTN service message, as the result of 3.1.2.3.4.4.1.2, the received message shall be processed as follows, depending on whether or not the conversion of each unknown addressee indicator into a recipient MF-Address in the same way as specified for an originator indicator in 3.1.2.3.4.2.1.4.1 can be succesfully performed by the Message Transfer and Control Unit:

a) processing as specified in 3.1.2.3.4.4.1.4, for the set of unknown addressee indicators which can be succesfully translated into an MF-Address, if any; and

b) for the set of unknown addressee indicators which cannot be succesfully translated, if any, processing as follows:

1) deletion in the text of the unknown address AFTN service message of all unknown addressee indicators processed as specified in a) above; and

2) conversion of the resulting unknown address AFTN service message into an IPM conveyed with a Message Transfer Envelope as specified in 3.1.2.3.4.4.1.7.

3.1.2.3.4.4.1.4 For the unknown recipient MF-Addresses determined as the result of 3.1.2.3.4.4.1.3 a), the received message shall be processed as follows, depending on the abstract-values of the *originator-report-request* and of the *originating-MTA-report-request* elements in the *per-recipient-indicators* in the corresponding *per-recipient-fields* of the subject AMHS message:

a) processing as specified in 3.1.2.3.4.4.1.5, for the set of recipients which meet the following condition, if any:

1) the abstract-value of the *originator-report-request* differs from "report"; and

2) the abstract-value of the *originating-MTA-report-request* differs from "report" and from "audited-report"; or

b) processing as follows, for all other recipients, if any:

1) replacement, in the text of the unknown address AFTN service message, of the entire list of unknown addressee indicators with a list restricted to the addressee indicators of these recipients; and

2) conversion of the resulting unknown address AFTN service message into an IPM conveyed with a Message Transfer Envelope as specified in 3.1.2.3.4.4.1.7.

Note.— This clause aims at avoiding the generation of a non-delivery-report after the generation of a delivery-report by the MTCU for the same subject AMHS message.

3.1.2.3.4.4.1.5 For each unknown recipient MF-Address which has not been subject to the generation of a delivery-report, the received message shall be processed in one of the following manners:

a) processing as specified in 3.1.2.3.4.4.1.6, if, for a given recipient, no non-delivery report has been generated yet in relation with the same subject AMHS message and with the same message recipient; or

b) discarding of the unknown address AFTN service message for the considered unknown recipient MF-Address and termination of the procedure for the given recipient if a non-delivery report has already been generated in relation with the same subject AMHS message and with the same message recipient.

Note.— This clause aims at avoiding the generation of a multiple non-delivery-reports in relation with a single subject AMHS message which would have been split in several AFTN messages when converted from the AMHS to the AFTN, as the result of 3.1.2.3.5.2.1.7.

3.1.2.3.4.4.1.6 A non-delivery report related to the unknown recipient MF-Addresses which have not caused the conversion of the unknown address AFTN service message into an IPM as the result of 3.1.2.3.4.4.1.4 and 3.1.2.3.4.4.1.5, shall be generated in compliance with:

a) the specification of 3.1.2.3.5.6 using the elements of the subject AMHS message; and

b) the following specification of abstract-values:

1) "unable-to-transfer" for the *non-delivery-reason-code*; and

2) "unrecognised-OR-name" for the *non-delivery-diagnostic-code*; and

c) the exception with respect to 3.1.2.3.5.6, that the *actual-recipient-name* element(s) in each *per-recipient-fields* element of the report take the value of the unknown recipient MF-Address(es) as determined in 3.1.2.3.4.4.1.5.

Note.— The potential future reception of an unknown address AFTN service message to be converted into a non-delivery-report requires the retention by the AFTN/AMHS Gateway of certain elements of the subject AMHS message for later report generation, if required.

3.1.2.3.4.4.1.7 When an unknown address AFTN service message is converted into an IPM as the result of 3.1.2.3.4.4.1.1 to 3.1.2.3.4.4.1.4, the specification of 3.1.2.3.4.2 shall apply, with the exception of the *subject* element in the IPM heading fields, initially specified in Table 3.1.2-5/Part2/10, which is then generated and takes the value "AFTN service information".

3.1.2.3.5 AMHS to AFTN Conversion

Note.— This clause specifies the actions to be performed by an AFTN/AMHS Gateway upon reception of information objects from the AMHS for conveyance over the AFTN, after the accomplishment of the AMHS-related procedures by the ATN Component as specified in 3.1.2.3.2.2.

3.1.2.3.5.1 Control Function

3.1.2.3.5.1.1 Upon reception by the Message Transfer and Control Unit of an AMHS message passed by the ATN Component, the received message shall be processed in one of the following manners, depending on the abstract-value of the *content-type* element in the Message Transfer Envelope:

a) processing as specified in 3.1.2.3.5.1.2 if the abstract-value of the element is either "interpersonal-messaging-1984", or "interpersonal-messaging-1988"; or

b) if the abstract-value of the element is neither "interpersonal-messaging-1984", nor "interpersonal-messaging-1988":

1) rejection of the message for all the message recipients for which the *responsibility* element of the *per-recipient-indicators* had the abstract-value "responsible"; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values:

i) "unable-to-transfer" for the non-delivery-reason-code; and

ii) "content-type-not-supported" for the non-delivery-diagnostic-code.

Note 1.— The message recipients towards which the Message Transfer and Control Unit conveys the message are those identified by a recipient-name element in the per-recipient-fields element of the Message Transfer Envelope, and for which the responsibility element in the per-recipient-indicators element has the abstract-value "responsible". In 3.1.2.3.5 the term "message recipient" refers to such a recipient.

Note 2.— Support of other content-types, e.g. edi-messaging, may be added in future packages.

3.1.2.3.5.1.2 Upon reception by the Message Transfer and Control Unit of an AMHS message whose *content-type* is either "interpersonal-messaging-1984" or "interpersonal-messaging-1988" passed from the ATN Component, the message shall be processed for conversion into an AFTN message in one of three mutually exclusive manners, depending on the nature of the content:

a) processing for conversion into an AFTN message as specified in 3.1.2.3.5.2, if the content is an IPM;

b) processing for conversion into an AFTN service message as specified in 3.1.2.3.5.3, if the content is an IPN which is a Receipt Notification (RN); or

c) unsuccessful termination of the procedure, if the content is an IPN but not a RN, resulting in:

1) logging of the error situation and reporting to a control position; and

2) storage of the message for appropriate processing at the control position.

3.1.2.3.5.1.3 Upon reception by the Message Transfer and Control Unit of an AMHS non-delivery report passed from the ATN Component, the report shall be processed as specified in 3.1.2.3.5.4.

3.1.2.3.5.1.4 Upon reception by the Message Transfer and Control Unit of an AMHS probe passed by the ATN Component, the received probe shall be processed in one of the following manners, depending on the abstract-value of the *content-type* element in the Probe Transfer Envelope:

a) processing for conveyance test as specified in 3.1.2.3.5.5 if the abstract-value of the element is either "interpersonal-messaging-1984", or "interpersonal-messaging-1988"; or

b) if the abstract-value of the element is neither "interpersonal-messaging-1984", nor "interpersonal-messaging-1988":

1) rejection of the probe for all the probe recipients for which the *responsibility* element of the *per-recipient-indicators* had the abstract-value "responsible"; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values:

i) "unable-to-transfer" for the *non-delivery-reason-code*; and

ii) "content-type-not-supported" for the *non-delivery-diagnostic-code*.

3.1.2.3.5.1.5 Upon reception by the Message Transfer and Control Unit of an ISO/IEC 10021 information object other than those referred to in clauses 3.1.2.3.5.1.1 to 3.1.2.3.5.1.4 above, the processing by the Message Transfer and Control Unit shall unsuccesfully terminate, resulting in:

a) logging of the error situation and reporting to a control position; and

b) storage of the information object for appropriate processing at the control position.

Note.— The Message Transfer and Control Unit requests non-delivery-reports, but never delivery-reports when generating AMHS messages.

3.1.2.3.5.1.6 Upon completion by the Message Transfer and Control Unit of the processing specified in clauses 3.1.2.3.5.1.1 to 3.1.2.3.5.1.4 above, the resulting AFTN message(s) or AFTN service message(s), if any, shall be passed to the AFTN component, for conveyance over the AFTN.

3.1.2.3.5.1.7 If the generation of a report is required in relation with the result of the processing specified in clauses 3.1.2.3.5.1.1 or 3.1.2.3.5.1.4 above, either due to message rejection or probe test failure by the Message Transfer and Control Unit, or due to a delivery-report request in the subject AMHS message or probe, an appropriate AMHS report shall be generated as specified in 3.1.2.3.5.6.

3.1.2.3.5.2 AMHS IPM Conversion

Upon reception by the Message Transfer and Control Unit of an IPM conveyed with a Message Transfer Envelope passed from the ATN Component to be conveyed over the AFTN, this message shall be converted into an AFTN message in compliance with the following:

a) the specification of the initial processing to be performed by the Message Transfer and Control Unit to determine the ability to convert the message and to split it into individually convertible messages, as included in 3.1.2.3.5.2.1;

b) the specification of how the AFTN message is generated and how the AFTN message components are mapped from AMHS parameters, as included in 3.1.2.3.5.2.2;

c) the specification of how the elements of the received IPM are handled, as included in 3.1.2.3.5.2.3; and

d) the specification of how the Message Transfer Envelope elements are handled, as included in 3.1.2.3.5.2.4.

3.1.2.3.5.2.1 Initial processing of AMHS Messages

3.1.2.3.5.2.1.1 Upon reception by the Message Transfer and Control Unit of an IPM, the received message shall be processed in one of the following manners, depending on the abstract-value of the current encoded-information-types, determined as either the abstract-value of the latest *converted-encoded-information-types*, if existing, in the *trace-information* element, or as the abstract-value of the *original-encoded-information-types* element if the previous does not exist:

a) processing as specified in 3.1.2.3.5.2.1.2 if the abstract-value of the current encoded-information-types is any of the following:

1) basic “ia5-text”;

2) externally-defined “ia5-text”;

3) OID {id-cs-eit-authority 1};

4) OID {id-cs-eit-authority 2};

5) OID {id-cs-eit-authority 6}; or

6) OID {id-cs-eit-authority 100}; or

b) if the abstract-value differs from all values indicated in item a) above:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) “unable-to-transfer” for the *non-delivery-reason-code*; and

ii) “encoded-information-types-unsupported” for the *non-delivery-diagnostic-code*.

3.1.2.3.5.2.1.2 A message which was not rejected as the result of 3.1.2.3.5.2.1.1 shall be processed in one of the following manners:

a) processing as specified in 3.1.2.3.5.2.1.3 if the abstract-value of the *implicit-conversion-prohibited* in the *per-message-indicators* element in the Message Transfer Envelope differs from “prohibited”; or

b) if the abstract-value of the element is “prohibited” and if the abstract-value of the encoded-information-types includes OID {id-cs-eit-authority 100}:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) "conversion-not-performed" for the *non-delivery-reason-code*;

ii) "implicit-conversion-prohibited" for the *non-delivery-diagnostic-code*; and

iii) "unable to convert to AFTN" for the *supplementary-information*.

3.1.2.3.5.2.1.3 A message which was not rejected as the result of 3.1.2.3.5.2.1.2 shall be processed in one of the following manners:

a) processing as specified in 3.1.2.3.5.2.1.4 if there is one single body part in the IPM body; or

b) if there are multiple body parts in the IPM body:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) "unable-to-transfer" for the *non-delivery-reason-code*;

ii) "content-syntax-error" for the *non-delivery-diagnostic-code*; and

iii) "unable to convert to AFTN due to multiple body parts" for the *supplementary-information*.

3.1.2.3.5.2.1.4 A message which was not rejected as the result of 3.1.2.3.5.2.1.3 shall be processed in one of the following manners:

a) processing as specified in 3.1.2.3.5.2.1.5 if the body part type is one of the following:

1) a basic body part type "ia5-text";

2) a standard extended body part type "ia5-text-body-part";

3) a standard extended body part type "general-text-body-part" of which the repertoire set description is Basic (ISO 646); or

4) a standard extended body part type "general-text-body-part" of which the repertoire set description is Basic-1 (ISO 8859-1), if and only if the local policy of the AMHS Management Domain is to support the conversion of

this repertoire set into IA5IRV characters according to locally defined conversion rules; or

b) if the body part type is different from the body part types 1) to 4) under a) above, or if the local policy of the AMHS Management Domain is not to support the conversion of the ISO 8859-1 repertoire set:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) "unable-to-transfer" for the *non-delivery-reason-code*;

ii) "content-syntax-error" for the *non-delivery-diagnostic-code*; and

iii) "unable to convert to AFTN due to unsupported body part type" for the *supplementary-information*.

Note.— The locally defined conversion rules mentioned in bullet 4), item a) may be for example CCITT Recommendation X.408.

3.1.2.3.5.2.1.5 A message not rejected as the result of 3.1.2.3.5.2.1.4 shall then be processed in one of the following manners:

a) processing as specified in 3.1.2.3.5.2.1.6 if the text structure in the body part in the body part complies with the requirements of 3.1.2.2.3.2; or

b) if the text structure does not comply with the requirements of 3.1.2.2.3.2:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the per-recipient-fields of the report:

i) "unable-to-transfer" for the *non-delivery-reason-code*;

ii) "content-syntax-error" for the *non-delivery-diagnostic-code*; and

iii) "unable to convert to AFTN due to ATS-Message-Header syntax error" for the *supplementary-information*.

Note.— The compliance requested to meet the condition of item b) includes the requirement that the element is present and has a value which is syntactically valid for the priority indicator, i.e. a value among SS, DD, FF, GG and KK, and for the filing time, i.e. a value in which the first six figures in the sequence build a valid date-time group.

3.1.2.3.5.2.1.6 A message which was not rejected as the result of 3.1.2.3.5.2.1.5 shall be processed in one of five mutually exclusive manners:

a) processing as specified in 3.1.2.3.5.2.1.7 if the abstract-value of the *conversion-with-loss-prohibited* element in the *extensions* of the per message fields is"allowed";

b) if the abstract-value of the element *conversion-with-loss-prohibited* is "prohibited" and at least one line in the message exceeds 69 characters:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) "conversion-not-performed" for the *non-delivery-reason-code*; and

ii) "line-too-long" for the *non-delivery-diagnostic-code*;

c) if the abstract-value of the element *conversion-with-loss-prohibited* is "prohibited" and at least one punctuation symbol in the text is not authorized in Annex 10, Volume II, 4.1.2:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the per-recipient-fields of the report:

i) "conversion-not-performed" for the *non-delivery-reason-code*; and

ii) "punctuation-symbol-loss" for the *non-delivery-diagnostic-code*;

d) if the abstract-value of the element *conversion-with-loss-prohibited* is "prohibited" and at least one alphabetical character in the text is not authorized in Annex 10, Volume II, 4.1.2:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) "conversion-not-performed" for the *non-delivery-reason-code*; and

ii) "alphabetical-character-loss" for the *non-delivery-diagnostic-code*; or

e) if several of the conditions under b) to d) above are simultaneously met:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) "conversion-not-performed" for the *non-delivery-reason-code*; and

ii) "multiple-information-loss" for the *non-delivery-diagnostic-code*.

3.1.2.3.5.2.1.7 A message which was not rejected as the result of 3.1.2.3.5.2.1.6 shall be processed in one of three mutually exclusive manners:

a) if the length of the ATS-Message-Text element exceeds 1800 characters, and if, due to system resource limitation, the procedure proposed in Annex 10, Volume II, Attachment D cannot be properly achieved by the AFTN/AMHS Gateway:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) "unable-to-transfer" for the *non-delivery-reason-code*;

ii) "content-too-long" for the *non-delivery-diagnostic-code*; and

iii) "unable to convert to AFTN due to message text length" for the *supplementary-information*.

b) if the length of the ATS-Message-Text element exceeds 1800 characters, and if the procedure proposed in Annex 10, Volume II, Attachment D is applied in the AFTN/AMHS Gateway:

1) splitting of the message, internally to the Message Transfer and Control Unit, into several messages in accordance with the aforementioned Annex 10 procedure:

i) each of the resulting messages having for conversion purposes the same Message Transfer Envelope, the same IPM Heading and the ATS-Message-Header as the message subject to the splitting; and

ii) only the ATS-Message-Text element varying between the different resulting messages; and

2) processing of each of these messages as specified in 3.1.2.3.5.2.1.8; or

c) processing as specified in 3.1.2.3.5.2.1.8 if the length of the ATS-Message-Text element does not exceed 1800 characters.

3.1.2.3.5.2.1.8 A message resulting from the situations in items b) and c) of 3.1.2.3.5.2.1.7 above shall be processed in one of three manners, depending on the number of message recipients towards which the Message Transfer and Control Unit is responsible for conveyance of the message, and on the AFTN/AMHS Gateway resources:

a) if this number exceeds 21 message recipients:

1) attempt to split the message, internally to the Message Transfer and Control Unit, into several messages, each of them with no more than 21 message recipients:

i) each of the resulting messages having for conversion purposes the same *per-message-fields* in the Message Transfer Envelope, and the same content as the message subject to the splitting; and

ii) only the *per-recipient-fields* elements in the Message Transfer Envelope varying between the different resulting messages; and

2) processing of each of these messages as specified in 3.1.2.3.5.2.2 to 3.1.2.3.5.2.4;

b) if this number exceeds 21 message recipients, and if, due to system resource limitation, the splitting attempt made by the gateway as specified in item a) above cannot be properly achieved:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) “unable-to-transfer” for the *non-delivery-reason-code*;

ii) “too-many-recipients” for the *non-delivery-diagnostic-code*; and

iii) “unable to convert to AFTN due to number of recipients” for the *supplementary-information*; or

c) processing as specified in 3.1.2.3.5.2.2 to 3.1.2.3.5.2.4, if this number does not exceed 21 message recipients.

Note 1.— In the processing defined in item a), the per-recipient-fields related to a particular recipient remain unchanged by the splitting. This applies in particular to the originally-specified-recipient-number, which is not altered by the processing specified in this clause.

Note 2.— The combination of 3.1.2.3.5.2.1.7 and 3.1.2.3.5.2.1.8 above may result in a very high number of AFTN messages being generated from one single AMHS message. Items 3.1.2.3.5.2.1.7 a) and 3.1.2.3.5.2.1.8 b) may, as a local matter, be used under such circumstances.

3.1.2.3.5.2.2 Generation of AFTN Message

3.1.2.3.5.2.2.1 Each message resulting from the processing specified in 3.1.2.3.5.2.1 above shall be converted by the Message Transfer and Control Unit into an AFTN Message composed of elements as specified in Table 3.1.2-10.

3.1.2.3.5.2.2.2 Those components which are classified as "G" in the column "action" of Table 3.1.2-10 shall be generated in compliance with the provisions of Annex 10, Volume II referred to in the column "mapping".

3.1.2.3.5.2.2.3 Those components which are classified as "T" or "T1" in the column "action" of Table 3.1.2-10 shall be converted from the AMHS parameter specified in the column "converted from AMHS parameter" of Table 3.1.2-10 and according to the specification in the clause referred to in the column "mapping".

Table 3.1.2-10. AFTN Message Generation

AFTN Message Part	Component	Action	Converted from AMHS parameter	Mapping
Heading	Start-of-Heading Character	X	-	-
	Transmission Identification	X	-	see 3.1.2.3.5.2.2.4
Address	Alignment Function	G	-	see Annex 10, Vol. II, 4.4.15.2.1
	Priority Indicator	T	ATS-Message-Priority (see Table 3.1.2-11/ Part 6/1.2)	see 3.1.2.3.5.2.2.5
	Addressee Indicator(s)	T	recipient-name (see Table 3.1.2-12/Part 1/1.2.1)	see 3.1.2.3.5.2.2.6.2
	Alignment Function	G	-	see Annex 10, Vol. II, 4.4.15.2.1
Origin	Filing Time	T	ATS-Message-Filing-Time (see Table 3.1.2-11/Part 6/1.3)	see 3.1.2.3.5.2.2.7
	Originator Indicator	T	originator-name (see Table 3.1.2-12/ Part 1/1.1.2)	see 3.1.2.3.5.2.2.6.1
	Priority Alarm	G	-	see Annex 10, Vol. II, 4.4.15.2.2
	Optional Heading Information	T1	ATS-Message-Optional-Heading-Info (see Table 3.1.2-11/Part 6/1.4)	see 3.1.2.3.5.2.2.8
	Alignment Function	G	-	see Annex 10, Vol. II, 4.4.15.2.2
	Start-of-Text Character	G	-	see Annex 10, Vol. II, 4.4.15.2.2
Text		T	ATS-Message-Text (see Table 3.1.2-11/Part 6/2)	see 3.1.2.3.5.2.2.9
Ending	Alignment Function	G	-	see Annex 10, Vol. II, 4.4.15.3.12

AFTN Message Part	Component	Action	Converted from AMHS parameter	Mapping
	Page-feed sequence	G	-	see Annex 10, Vol. II, 4.4.15.3.12
	End-of-Text Character	G	-	see Annex 10, Vol. II, 4.4.15.3.12

Legend: (see 3.1.1)
X = excluded (not used)
T1 = conditionally translated
G = generated
T = translated

3.1.2.3.5.2.2.4 As specified in 3.1.2.3.2.5.3, the element transmission identification shall be:

a) generated by the AFTN Component rather than by the Message Transfer and Control Unit; and

b) returned to the Message Transfer and Control Unit as the result of the operation transferring the generated AFTN Message from the Message Transfer and Control Unit to the AFTN Component.

3.1.2.3.5.2.2.5 The value of the priority indicator of the converted AFTN message shall be the value of the priority-indicator in the ATS-message-priority element of the AMHS message.

3.1.2.3.5.2.2.6 The value of an AF-Address included in the converted AFTN message shall be converted from an MF-Address as respectively specified in 3.1.2.3.5.2.2.6.1 and 3.1.2.3.5.2.2.6.2 depending whether it is an originator MF-Address or a recipient MF-Address.

3.1.2.3.5.2.2.6.1 The originator MF-Address included in an AMHS message shall be processed for translation into the originator indicator of the converted AFTN Message in one of three mutually exclusive manners, depending on the value of the *organization-name* attribute and on the contents of the User address look-up table, after preliminary conversion of the value of all AMHS address attributes from lower case IA5IRV characters, if any, to upper case IA5IRV characters:

a) allocation of the value of the first element of the *organizational-unit-names* attribute to the originator indicator of the converted AFTN Message, if this value is a syntactically valid AF-Address and if the *organization-name* attribute has the value "AFTN";

b) determination of an AF-Address matching exactly the MF-Address of the originator in the User address look-up table maintained in the Message Transfer and Control Unit, if the value of the *organization-name* attribute differs from "AFTN" and if such an exact match can be found; or

c) if none of the conditions in a) and b) can be met, then:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) "unable-to-transfer" for the *non-delivery-reason-code*;

ii) "invalid-arguments" for the *non-delivery-diagnostic-code*; and

iii) "unable to convert to AFTN due to unrecognized originator O/R address" for the *supplementary-information*.

3.1.2.3.5.2.2.6.2 To build the address part of the converted AFTN Message as specified in Annex 10, Volume II, 4.4.15.2.1, each of the recipient MF-Addresses included in an AMHS message, whose *responsibility* element in the *per-recipient-indicators* has the abstract-value "responsible", shall be processed for translation into an addressee indicator in one of three mutually exclusive manners:

a) allocation of the value of the first element of the *organizational-unit-names* attribute, converted from lower case IA5IRV characters, if any, to upper case IA5IRV characters, to an addressee indicator in the converted AFTN Message, if this value is a syntactically valid AF-Address and if the *organization-name* attribute has the value "AFTN";

b) determination of an AF-Address matching exactly the MF-Address of the recipient in the User address look-up table maintained in the Message Transfer and Control Unit, if the value of the *organization-name* attribute differs from "AFTN" and if such an exact match can be found; or

c) if none of the conditions in a) and b) can be met, then:

1) rejection of the message for the considered message recipient; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) "unable-to-transfer" for the *non-delivery-reason-code*; and

ii) "unrecognised-OR-name" for the *non-delivery-diagnostic-code*.

Note.— Although the potential generation of a non-delivery report is mentioned for each recipient-name which cannot be properly translated into an AF-Address, a single report with different per-recipient-fields may be generated for all recipient-names which cannot be translated.

3.1.2.3.5.2.2.7 The value of the filing time of a converted AFTN message shall be the value of the filing-time component in the ATS-Message-Filing-Time element of the AMHS message.

3.1.2.3.5.2.2.8 The Optional Heading Information of a converted AFTN message shall either:

a) take the value of the optional-heading-information in the ATS-Message-Optional-Heading-Info element, if this element is present; or

b) be omitted in the converted AFTN message, if the ATS-Message-Optional-Heading-Info element is absent from the AMHS message.

3.1.2.3.5.2.2.9 The content of the Text part of a converted AFTN message shall be derived from the value of the ATS-Message-Text element of the IPM text of the AMHS message, in compliance with the following procedure:

a) conversion of each character which is not in the IA5IRV character repertoire, into an IA5IRV character according to the locally defined conversion rules;

b) conversion of each IA5IRV character, if it is in lower case, into the equivalent upper case character;

c) replacement by question-marks ("?") of all characters or character sequences in the text, if any, of which the use is not authorized in Annex 10, Volume II, 4.1.2;

d) folding of any line longer than 69 characters; and

e) allocation of the result of items a) to d) above to the Text part of the converted AFTN message.

Note 1.— The locally defined conversion rules mentioned in item a) may be for example CCITT Recommendation X.408, if support of the ISO 8859-1 character set is a local policy of the AMHS Management Domain.

Note 2.— A lower case IA5IRV character is one whose position is between 6/1 and 6/15 or 7/0 and 7/10. The corresponding upper case IA5IRV characters have positions extending from 4/1 to 4/15 and 5/0 to 5/10.

3.1.2.3.5.2.3 Use of IPM elements

3.1.2.3.5.2.3.1 Each of the elements composing the IPM in an AMHS message to be converted into an AFTN message in the Message Transfer and Control Unit shall be processed as specified in the column "action" of Table 3.1.2-11.

3.1.2.3.5.2.3.2 The elements composing the IPM shall be used according to the specification in the clause referred to in the column "mapping" of Table 3.1.2-11.

Note 1.— Table 3.1.2-11 is structured as a PRL derived from the profile specification included in 2.2 and consequently from the ISPICS Proforma included in ISO/IEC ISP 12062-2 as well as from Table 3.1.2-2 in 3.1.2.2.3.2. The columns "Base" and "ISP" under "Reception" are extracted from ISO/IEC ISP 12062-2 and the column "Basic ATS Message Service" specifies the static capability of an IPM AU supporting the Basic ATS Message Service, i.e. the ability to handle in reception the element as part of an IPM carrying an ATS Message. The references to the ISP Profile are indicated in the part titles as AMH21/ref where appropriate. The references in column Ref are those of the ISP.

Table 3.1.2-11. Use of IPM Elements

Ref	Element	Reception			Action	Mapping / Notes
		Base	**ISP**	**Basic ATS Mess. Service**		
PART 1 : AMH21/A.1.1 SUPPORTED INFORMATION OBJECTS						
1	Interpersonal Message (IPM)	M	M	M	T	see Part 1/1.1 and 1.2
1.1	heading	M	M	M	T	see Part 2
1.2	body	M	M	M	T	see Part 3
2	Interpersonal Notification (IPN)	O	M	M	-	out of the scope of this clause
PART 2 : AMH21/A.1.2 IPM HEADING FIELDS						
1	this-IPM	M	M	M	D	-
2	originator	M	M	M	D	-
3	authorizing-users	M	M	M	D	-
4	primary-recipients	M	M	M	D	see 3.1.2.3.5.2.3.3 and Part 5/1
5	copy-recipients	M	M	M	D	see 3.1.2.3.5.2.3.3 and Part 5/1
6	blind-copy-recipients	M	M	M	D	see 3.1.2.3.5.2.3.3 and Part 5/1
7	replied-to-IPM	M	M	M	D	-
8	obsoleted-IPMs	M	M	M	D	-
9	related-IPMs	M	M	M	D	-

Ref	Element	Reception			Action	Mapping / Notes
		Base	**ISP**	**Basic ATS Mess. Service**		
10	subject	M	M	M	D	-
11	expiry-time	M	M	M	D	-
12	reply-time	M	M	M	D	-
13	reply-recipients	M	M	M	D	-
14	importance	M	M	M	D	-
15	sensitivity	M	M	M	D	-
16	auto-forwarded	M	M	M	D	-
17	extensions	M	M	M	D	-
17.1	incomplete-copy	O	M	M	D	-
17.2	languages	M	M	M	D	-
17.3	auto-submitted	O	I	I	D	-
PART 3 : AMH21/A.1.3 IPM BODY						
1	ia5-text	O	M	M	T	see Part 3/1.1 and 1.2
1.1	parameters	M	M	M	D	-
1.1.1	repertoire	M	M	M	D	-
1.2	data	M	M	M	T	see Part 6
2	voice	I	I	I	X	see Note 2
3	g3-facsimile	O	O	O	X	see Note 2

Ref	Element	Reception			Action	Mapping / Notes
		Base	**ISP**	**Basic ATS Mess. Service**		
4	g4-class-1	O	O	O	X	see Note 2
5	teletex	O	O	O	X	see Note 2
6	videotex	O	O	O	X	see Note 2
7	encrypted	I	I	I	X	see Note 2
8	message	O	M	M	X	see Note 2
9	mixed-mode	O	O	O	X	see Note 2
10	bilaterally-defined	O	O	O	X	see Note 2
11	nationally-defined	O	O	O	X	see Note 2
12	externally-defined	O	M	M	X/T	see Note 3 and Part 4
PART 4 : AMH21/A.1.3.1 EXTENDED BODY PART SUPPORT						
1	ia5-text-body-part	O	M	M	T	see Part 3/1
2	g3-facsimile-body-part	O	O	O	X	see Note 2
3	g4-class1-body-part	O	O	O	X	see Note 2
4	teletex-body-part	O	O	O	X	see Note 2
5	videotex-body-part	O	O	O	X	see Note 2
6	encrypted-body-part	I	I	I	X	see Note 2
7	message-body-part	O	M	M	X	see Note 2
8	mixed-mode-body-part	O	O	O	X	see Note 2

Ref	Element	Reception			Action	Mapping / Notes
		Base	**ISP**	**Basic ATS Mess. Service**		
9	bilaterally-defined-body-part	O	O	O	X	see Note 2
10	nationally-defined-body-part	O	O	O	X	see Note 2
11	general-text-body-part	O	M	M	T/X	see 3.1.2.3.5.2.1.4, 3.1.2.3.5.2.3.4 and Part 6
12	file-transfer-body-part	O	I	I	X	see Note 2
13	voice-body-part	O	I	I	X	see Note 2
14	oda-body-part	O	O	O	X	see Note 2
PART 5 : AMH21/A.1.5 COMMON DATA TYPES						
1	RecipientSpecifier					
1.1	recipient	M	M	M	D	-
1.2	notification-requests	M	M	M	D	see Part 5/1.2.1-1.2.3
1.2.1	rn	O	O	O	D	see 3.1.2.3.5.2.3.3
1.2.2	nrn	M	M	M	D	-
1.2.3	ipm-return	O	O	O	D	-
1.3	reply-requested	M	M	M	D	-
1.4	recipient-extensions	O	I	I	D	-

Ref	Element	Reception			Action	Mapping / Notes
		Base	**ISP**	**Basic ATS Mess. Service**		
PART 6 : IPM SUPPORT OF THE BASIC ATS MESSAGE SERVICE						
1	ATS-Message-Header	-	-	M	T	see Part 6/1.1-1.6
1.1	start-of-heading	-	-	M	-	-
1.2	ATS-Message-Priority	-	-	M	T	see Part 6/1.2.1-1.2.3
1.2.1	priority-prompt	-	-	M	-	-
1.2.2	priority-indicator	-	-	M	T	see 3.1.2.3.5.2.2.5 and 3.1.2.3.5.2.3.3
1.2.3	priority-separator	-	-	M	-	-
1.3	ATS-Message-Filing-Time	-	-	M	T	see Part 6/1.3.1-1.3.3
1.3.1	filing-time-prompt	-	-	M	-	-
1.3.2	filing-time	-	-	M	T	see 3.1.2.3.5.2.2.7
1.3.3	filing-time-separator	-	-	M	-	-
1.4	ATS-Message-Optional-Heading-Info	-	-	M	T1	see Part 6/1.4.1-1.4.3
1.4.1	OHI-prompt	-	-	M	-	-
1.4.2	optional-heading-information	-	-	M	T	see 3.1.2.3.5.2.2.8
1.4.3	OHI-separator	-	-	M	-	-
1.5	end-of-heading-blank-line	-	-	M	-	-

Ref	Element	Reception			Action	Mapping / Notes
		Base	**ISP**	**Basic ATS Mess. Service**		
1.6	start-of-text	-	-	M	-	-
2	ATS-Message-Text	-	-	M	T	see 3.1.2.3.5.2.2.9

Legend (see 3.1.1) :
M = mandatory support
O = optional support
I = out of scope
- = not applicable
T1 = conditionally translated
D = discarded
T = translated
X = excluded

Note 2.— This body part type is excluded as the result of 3.1.2.3.5.2.1.4.

Note 3.— This body part type may be either excluded or translated, depending on whether or not it is a standard extended body part type, and if yes,depending on the type of extended body part type, as specified in Part 4 and as the result of 3.1.2.3.5.2.1.4.

3.1.2.3.5.2.3.3 If the priority-indicator of a received AMHS message has the value "SS" and if the *responsibility* element of the corresponding *per-recipient-fields* of the Message Transfer Envelope has the value "responsible", then an error situation shall be logged and reported to a control position for appropriate action if any of the following situations, or both, occurs:

a) if the *notification-requests* element of either a *primary-recipient*, or a *copy-recipient*, or a *blind-copy-recipient* element has an abstract-value different from "rn"; or

b) if the *priority* element of the Message Transfer Envelope has an abstract-value different from "urgent".

Note 1.— The Message Transfer and Control Unit generates RNs only for SS priority messages, since they are the only messages for which an end-to-end acknowledgement is possible in the AFTN. A receipt-notification-request included in a message with another priority is ignored, considering that the Message Transfer and Control Unit cannot ensure the actual reception of the message by the end-user.

Note 2.— The above specified error situation, if any, does not cause message rejection.

3.1.2.3.5.2.3.4 The components of a general-text body part shall be used as follows for the conversion of the IPM body into the text of the AFTN Message:

a) the parameters component identify the character set used for the message, as specified in ISO/IEC 10021-7, B.2; and

b) the data component of a general-text body part are used for the generation of the converted AFTN message as specified in Part 6 of Table 3.1.2-11.

3.1.2.3.5.2.4 Use of Message Transfer Envelope parameters

3.1.2.3.5.2.4.1 Each of the elements composing the Message Transfer Envelope of an AMHS message to be converted into an AFTN message in a Message Transfer and Control Unit shall be processed as specified in the column "action" of Table 3.1.2-12.

3.1.2.3.5.2.4.2 The elements composing the Message Transfer Envelope shall be handled according to the specification in the clause referred to in the column "mapping" of Table 3.1.2-12.

Note 1.— Table 3.1.2-12 is structured as a PRL derived from the ISPICS Proforma included in ISO/IEC ISP 10611-3. The columns "Base" and "ISP" are extracted from ISO/IEC ISP 10611-3 and the column "Basic ATS Message Service" specifies the static capability of an AU in relation with the MT-EoS (Message Transfer Elements of Service), i.e. the ability to convey, handle and act in relation with the element. The references to the ISP Profile are indicated in the part titles as AMH11/ref where appropriate.

Note 2.— Although not used for mapping, some elements may generate specific actions for the gateway in the handling of the considered message.

Note 3.— Some elements may have two classifications, e.g. D/X where certain values of the element may cause message rejection, while other values are simply discarded when the AMHS message is converted into an AFTN message.

Table 3.1.2-12. Use of the MessageTransfer Envelope

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
PART 1 : AMH11/A.1.4.2 MESSAGETRANSFER						
1	MessageTransferEnvelope	M	M	M	T	see Part 1/1.1 and 1.2
1.1	(per message fields)					
1.1.1	message-identifier	M	M	M	D	-
1.1.2	originator-name	M	M	M	T	see 3.1.2.3.5.2.2.6.1
1.1.3	original-encoded-information-types	M	M-	M-	D/X	see 3.1.2.3.5.2.1.2
1.1.4	content-type	M	M-	M-	D/X	see 3.1.2.3.5.1.1

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
1.1.5	content-identifier	M	M	M	D	-
1.1.6	priority	M	M	M	D	-
1.1.7	per-message-indicators	M	M	M	D	see Part 2/4
1.1.8	deferred-delivery-time	O	M-	M-	D	see 3.1.2.3.5.2.4.4
1.1.9	per-domain-bilateral-information	O	M-	M-	D	see 3.1.2.3.5.2.4.5 and Part 2/5
1.1.10	trace-information	M	M	M	D	see Part 2/6
1.1.11	extensions	M	M	M	D/X	see 3.1.2.3.5.2.4.6 and Part 3/1
1.1.11.1	recipient-reassignment-prohibited	O	M	M	D	see 3.1.2.3.5.2.4.3
1.1.11.2	dl-expansion-prohibited	O	M	M	D	see 3.1.2.3.5.2.4.7
1.1.11.3	conversion-with-loss-prohibited	O	M	M	D/X	see 3.1.2.3.5.2.1.6
1.1.11.4	latest-delivery-time	O	M-	M-	D/X	see 3.1.2.3.5.2.4.8
1.1.11.5	originator-return-address	O	M-	M-	D	-
1.1.11.6	originator-certificate	O	M-	M-	X	see 3.1.2.3.5.2.4.9
1.1.11.7	content-confidentiality-algorithm-identifier	O	M-	M-	X	see 3.1.2.3.5.2.4.9
1.1.11.8	message-origin-authentication-check	O	M-	M-	X	see 3.1.2.3.5.2.4.9
1.1.11.9	message-security-label	O	M-	M-	X	see 3.1.2.3.5.2.4.9
1.1.11.10	content-correlator	M	M	M	D	-
1.1.11.11	dl-expansion-history	M	M-	M-	D	-
1.1.11.12	internal-trace-information	M	M	M	D	-

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
1.2	per-recipient-fields	M	M	M	T	see Part 1/1.2.1-1.2.5
1.2.1	recipient-name	M	M	M	T	see 3.1.2.3.5.2.2.6.2
1.2.2	originally-specified-recipient-number	M	M	M	D	-
1.2.3	per-recipient-indicators	M	M	M	D	-
1.2.4	explicit-conversion	O	M-	M-	D	-
1.2.5	extensions	M	M	M	D/X	see 3.1.2.3.5.2.4.6 and Part 3/1
1.2.5.1	originator-requested-alternate-recipient	O	M-	M-	D	see 3.1.2.3.5.2.4.3
1.2.5.2	requested-delivery-method	O	M-	M-	D	see 3.1.2.3.5.2.4.10
1.2.5.3	physical-forwarding-prohibited	O	M-	M-	X	see 3.1.2.3.5.2.4.11
1.2.5.4	physical-forwarding-address-request	O	M-	M-	X	see 3.1.2.3.5.2.4.11
1.2.5.5	physical-delivery-modes	O	M-	M-	X	see 3.1.2.3.5.2.4.11
1.2.5.6	registered-mail-type	O	M-	M-	X	see 3.1.2.3.5.2.4.11
1.2.5.7	recipient-number-for-advice	O	M-	M-	X	see 3.1.2.3.5.2.4.11
1.2.5.8	physical-rendition-attributes	O	M-	M-	X	see 3.1.2.3.5.2.4.11
1.2.5.9	physical-delivery-report-request	O	M-	M-	X	see 3.1.2.3.5.2.4.11
1.2.5.10	message-token	O	M-	M-	X	see 3.1.2.3.5.2.4.9
1.2.5.11	content-integrity-check	O	M-	M-	X	see 3.1.2.3.5.2.4.9
1.2.5.12	proof-of-delivery-request	O	M-	M-	X	see 3.1.2.3.5.2.4.9
1.2.5.13	redirection-history	M	M-	M-	D	-

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
2	content	M	M	M	T	see 3.1.2.3.5.2.3
PART 2 : AMH11/A.1.5 COMMON DATA TYPES						
4	PerMessageIndicators					
4.1	disclosure-of-other-recipients	M	M	M	D	-
4.2	implicit-conversion-prohibited	M	M	M	D/X	see 3.1.2.3.5.2.1.1
4.3	alternate-recipient-allowed	M	M	M	D	see 3.1.2.3.5.2.4.3
4.4	content-return-request	O	M-	M-	D	-
4.5	reserved	O	M-	M-	D	-
4.6	bit-5	O	M-	M-	D	-
4.7	bit-6	O	M-	M-	D	-
4.8	service-message	O	M-	M-	D	-
5	PerDomainBilateralInformation					
5.1	country-name	M	M-	M-	D	see 3.1.2.3.4.2.4.5
5.2	administration-domain-name	M	M-	M-	D	see 3.1.2.3.4.2.4.5
5.3	private-domain-identifier	O	M-	M-	D	see 3.1.2.3.4.2.4.5
5.4	bilateral-information	M	M-	M-	D	see 3.1.2.3.4.2.4.5
6	TraceInformation					

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
6.1	TraceInformationElement	M	M	M	D	-

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
6.1.1	global-domain-identifier	M	M	M	D	-
6.1.2	domain-supplied-information	M	M	M	D	-
6.1.2.1	arrival-time	M	M	M	D	-
6.1.2.2	routing-action	M	M	M	D	-
6.1.2.2.1	relayed	M	M	M	D	-
6.1.2.2.2	rerouted	O	C1	C1	D	-
6.1.2.3	attempted-domain	O	C1	C1	D	-
6.1.2.4	(additional actions)				D	-
6.1.2.4.1	deferred-time	M	C2	C2	D	-
6.1.2.4.2	converted-encoded-information-types	O	M-	M-	D	see 3.1.2.3.5.2.1.2
6.1.2.4.3	other-actions	O	M-	M-	D	-
6.1.2.4.3.1	redirected	O	M-	M-	D	-
6.1.2.4.3.2	dl-operation	O	M-	M-	D	-
PART 3 : AMH11/A.1.6 EXTENSION DATA TYPES						
1	ExtensionField					
1.1	type	M	M	M	D/X	see Part 3/1.1.1 and 1.1.2
1.1.1	standard-extension	M	M	M	D/X	see 3.1.2.3.5.2.4.6
1.1.2	private-extension	O	M-	M-	D/X	see 3.1.2.3.5.2.4.6

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
1.2	criticality	M	M	M	D/X	see 3.1.2.3.5.2.4.6
1.3	value	M	M	M	D	-

Legend (see 3.1.1) :
M = mandatory support
M- = minimal mandatory support
O = optional support
C1 = if rerouting is supported then M else M-
C2 = if deferred delivery is supported then M else M-
D = discarded
T = translated
X = excluded

3.1.2.3.5.2.4.3 The elements *alternate-recipient-allowed* and *originator-requested-alternate-recipient* shall be discarded by the Message Transfer and Control Unit, since the optional Redirection Functional Group, if implemented in an AFTN/AMHS Gateway, is supported by the ATN Component and not by the Message Transfer and Control Unit.

3.1.2.3.5.2.4.4 The element *deferred-delivery-time* shall be discarded by the Message Transfer and Control Unit, since this functionality, if implemented in an AFTN/AMHS Gateway, is supported by the ATN Component and not by the Message Transfer and Control Unit.

3.1.2.3.5.2.4.5 For mapping purposes the whole *per-domain-bilateral-information* element shall be discarded.

Note.— If the elements country-name, administration-domain-name and private-domain-identifier in an element of the per-domain-bilateral-information together identify the AMHS Management Domain operating the AFTN/AMHS Gateway, the use made of the bilateral-information element is a local matter.

3.1.2.3.5.2.4.6 If any extension-field is present in the *extensions* of the Message Transfer Envelope and not semantically understood by the Message Transfer and Control Unit, then the element shall either:

a) cause the following actions to be performed if its criticality is set to “CRITICAL FOR TRANSFER” or to “CRITICAL FOR DELIVERY”:

1) message rejection of the message for either:

i) all the message recipients if the extension is part of the *per-message-fields*; or

ii) the considered message recipient if the extension is part of the *per-recipient-fields*; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in the appropriate *per-recipient-fields* of the report:

i) "unable-to-transfer" for the *non-delivery-reason-code*; and

ii) "unsupported-critical-function" for the *non-delivery-diagnostic-code*; or

b) be simply discarded if there is no criticality given.

3.1.2.3.5.2.4.7 The element *dl-expansion-prohibited* shall be discarded by the Message Transfer and Control Unit, since the DL-expansion capability of an AFTN/AMHS Gateway is supported by the ATN Component and not by the Message Transfer and Control Unit.

3.1.2.3.5.2.4.8 If the *latest-delivery-time* element is present, and if, when the AMHS message is handled by the Message Transfer and Control Unit, the current time exceeds the value of the *latest-delivery-time*, then the following actions shall be performed:

a) message rejection for all the message recipients; and

b) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in the appropriate *per-recipient-fields* of the report:

1) "transfer-failure" for the *non-delivery-reason-code*; and

2) "maximum-time-expired" for the *non-delivery-diagnostic-code*.

3.1.2.3.5.2.4.9 The Message Transfer and Control Unit does not implement Security Elements of Service. Thus, if any security-related extension-field set to "CRITICAL FOR DELIVERY" is present in the *extensions* of the Message Transfer Envelope, the following actions shall be performed:

a) message rejection of the message for either:

1) all the message recipients if the extension is part of the *per-message-fields*; or

2) the considered message recipient if the extension is part of the *per-recipient-fields*; and

b) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in the appropriate *per-recipient-fields* of the report:

1) "unable-to-transfer" for the *non-delivery-reason-code*; and

2) "unsupported-critical-function" for the *non-delivery-diagnostic-code*.

3.1.2.3.5.2.4.10 The element *requested-delivery-method* shall be discarded by the Message Transfer and Control Unit.

Note.— The Message Transfer and Control Unit handles the message irrespective of the value of this attribute, since it indicates only a preferred delivery method (see Technical Corrigendum 5 to ISO/IEC 10021-4).

3.1.2.3.5.2.4.11 The Message Transfer and Control Unit does not implement Physical Delivery Elements of Service. Thus, if any physical delivery-related extension-field set to "CRITICAL FOR DELIVERY" is present in the *extensions* of the Message Transfer Envelope, the following actions shall be performed:

a) message rejection of the message for either:

1) all the message recipients if the extension is part of the *per-message-fields*; or

2) the considered message recipient if the extension is part of the *per-recipient-fields*; and

b) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in the appropriate *per-recipient-fields* of the report:

1) "physical-rendition-not-performed" for the *non-delivery-reason-code*; and

2) "unsupported-critical-function" for the *non-delivery-diagnostic-code*.

3.1.2.3.5.3 AMHS RN Conversion

Upon reception by the Message Transfer and Control Unit of a RN conveyed with a Message Transfer Envelope passed from the ATN Component, for the acknowledgement of a SS message, this message shall be converted into an AFTN acknowledgement message in compliance with the following:

a) the specification of the initial processing performed to determine the Message Transfer and Control Unit ability to convert the RN, as included in 3.1.2.3.5.3.1;

b) the specification of how the AFTN service message is generated and how the AFTN service message components are mapped from AMHS parameters, as included in 3.1.2.3.5.3.2;

c) the specification of how the elements of the received RN are handled, as included in 3.1.2.3.5.3.3; and

d) the specification of how the Message Transfer Envelope elements are handled, as included in 3.1.2.3.5.3.4.

3.1.2.3.5.3.1 Initial processing of AMHS Receipt Notifications

3.1.2.3.5.3.1.1 Upon reception by the Message Transfer and Control Unit of a RN, passed from the ATN Component to be potentially converted into an AFTN acknowledgement message, the received RN shall be processed in one of the following manners:

a) processing as specified in 3.1.2.3.5.3.1.2, if the subject IPM has been previously generated by the Message Transfer and Control Unit; or

b) unsuccessful termination of the procedure, if the subject IPM has not been previously generated by the Message Transfer and Control Unit, resulting in:

1) logging of the error situation and reporting to a control position;

2) storage of the RN for appropriate action at the control position; and

3) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values:

i) "unable-to-transfer" for the *non-delivery-reason-code*;

ii) "invalid-arguments" for the *non-delivery-diagnostic-code*; and

iii) "unable to convert RN to AFTN Ack service message due to misrouted RN" for the *supplementary-information*.

3.1.2.3.5.3.1.2 For an AMHS RN passed from the ATN Component to the Message Transfer and Control Unit and not rejected as the result of 3.1.2.3.5.3.1.1, the received RN shall be processed in one of the following manners:

a) processing as specified in 3.1.2.3.5.3.1.3, if the value of the priority indicator of the subject AFTN message was "SS"; or

b) unsuccessful termination of the procedure, if the value of the priority indicator was different from "SS", resulting in:

1) logging of the error situation and reporting to a control position; and

2) storage of the RN for appropriate action at the control position.

3.1.2.3.5.3.1.3 An AMHS RN passed from the ATN Component to the Message Transfer and Control Unit and not rejected as the result of 3.1.2.3.5.3.1.2 shall be processed as specified in 3.1.2.3.5.3.2.

3.1.2.3.5.3.2 Generation of the AFTN acknowledgement message

3.1.2.3.5.3.2.1 An AMHS RN received by the Message Transfer and Control Unit and not rejected as the result of 3.1.2.3.5.3.1 shall be converted into an AFTN acknowledgement message in compliance with:

a) the specification of 3.1.2.3.5.2.2 with the exception of the components listed in Table 3.1.2-13; and

b) the classification of the components included in Table 3.1.2-13, as specified in the column "action" of Table 3.1.2-13.

3.1.2.3.5.3.2.2 These components which are classified as "G" shall be generated in compliance with the clause referred to in the column "mapping" of Table 3.1.2-13.

3.1.2.3.5.3.2.3 These components which are classified as "T" shall be converted from the AMHS parameter specified in the column "converted from AMHS parameter" of Table 3.1.2-13 and according to the specification in the clause referred to in the column "mapping".

Table 3.1.2-13. Generation of AFTN acknowledgement message

AFTN Message Part	Component	Action	converted from AMHS parameter	Mapping
Address	Priority Indicator	G	-	see 3.1.2.3.5.3.2.4
Origin	Filing Time	T	receipt-time (see Table 3.1.2-14/Part 1/7.1)	see 3.1.2.3.5.3.2.5
	Optional Heading Information	X	-	-
Text		G	-	see 3.1.2.3.5.3.2.6

Legend: (see 3.1.1)
G = generated
T = translated
X = excluded (not used)

3.1.2.3.5.3.2.4 In an AFTN acknowledgement message, generated as the result of the conversion of an AMHS RN message, the priority indicator component shall take the value SS.

3.1.2.3.5.3.2.5 In an AFTN acknowledgement message, generated as the result of the conversion of an AMHS RN message, the filing time component shall:

a) be a date-time group as specified in Annex 10, Volume II, 4.4.15.2.2.1; and

b) take the value of the six characters between the fifth and tenth position from the *receipt-time* element of the RN.

3.1.2.3.5.3.2.6 In an AFTN acknowledgement message, generated as the result of the conversion of an AMHS RN message, the value of the Text component shall be generated as specified in Annex 10, Volume II, 4.4.15.6 using the origin of the subject AFTN message.

3.1.2.3.5.3.3 Use of RN fields

3.1.2.3.5.3.3.1 Each of the elements composing the RN to be converted into an AFTN acknowledgement message in an AFTN/AMHS Gateway shall be processed as specified in the column "action" of Table 3.1.2-14.

3.1.2.3.5.3.3.2 The elements composing the RN shall be handled according to the specification in the clause referred to in the column “mapping” of Table 3.1.2-14.

Note.— Table 3.1.2-14 is structured as a PRL derived from the profile specification included in 2.2 and consequently from the ISPICS Proforma included in ISO/IEC ISP 12062-2 (AMH21). The columns “Base” and “ISP” under “Reception” are extracted from ISO/IEC ISP 12062-2, and the column “Basic ATS Message Service” specifies the static capability of an IPM AU supporting the Basic ATS Message Service, i.e. the ability to handle in reception the element as part of a RN. The references to the ISP Profile are indicated in the part titles as AMH21/ref where appropriate. The references in column Ref are those of the ISP.

Table 3.1.2-14. Use of RN fields

Ref	**Element**	**Reception**			**Action**	**Mapping / Notes**
		Base	**ISP**	**Basic ATS Mess. Service**		
PART 1: AMH21/A.1.4 IPN FIELDS						
1	subject-ipm	M	M	M	D	-
2	ipn-originator	M	M	M	D	-
3	ipm-preferred-recipient	M	M	M	D	-
4	conversion-eits	M	M	M	D	-
5	notification-extensions	O	I	I	-	-
6	non-receipt-fields	O	M	M	-	out of the scope of this clause
7	receipt-fields	O	M	M	T	see Part 1/7.1-7.4
7.1	receipt-time	M	M	M	T	see 3.1.2.3.5.3.2.5
7.2	acknowledgment-mode	M	M	M	D	-
7.3	suppl-receipt-info	O	O	O	D	-
7.4	rn-extensions	O	I	I	-	-

Ref	Element	Reception			Action	Mapping / Notes
		Base	**ISP**	**Basic ATS Mess. Service**		
8	other-notification-type-fields	O	I	I	-	-

Legend (see 3.1.1) :
M = mandatory support
O = optional support
I = out of scope
- = not applicable
D = discarded
T = translated
- = out of scope

3.1.2.3.5.3.4 Use of Message Transfer Envelope parameters conveyed with a RN

3.1.2.3.5.3.4.1 The elements composing the Message Transfer Envelope conveyed with a RN to be converted into an AFTN acknowledgement message shall be used in compliance with:

a) the specification of 3.1.2.3.5.2.4 with the exception of those elements included in Table 3.1.2-15; and

b) the specification included in the clause referred to in the column “Mapping” of Table 3.1.2-15.

Note.— Table 3.1.2-15 is structured as an extraction of Table 3.1.2-12.

Table 3.1.2-15. Use of the MessageTransfer Envelope conveyed with a RN (differences from Table 3.1.2-12)

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
PART 1 : AMH11/A.1.4.2 MESSAGE TRANSFER						
1	MessageTransferEnvelope	M	M	M	T	see Part 1/1.1 and 1.2
1.1	(per message fields)					
1.1.3	original-encoded-information-types	M	M-	M-	D	see 3.1.2.3.5.3.4.2

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
1.1.7	per-message-indicators	M	M	M	D	see Part 2/4
1.1.10	trace-information	M	M	M	D	see Part 2/6
1.2	per-recipient-fields	M	M	M	D	see Part 1/1.2.1
1.2.1	recipient-name	M	M	M	D	see 3.1.2.3.5.3.4.3
2	content	M	M	M	T	see 3.1.2.3.5.3.3
PART 2 : AMH11/A.1.5 COMMON DATA TYPES						
4	PerMessageIndicators					
4.2	implicit-conversion-prohibited	M	M	M	D	see 3.1.2.3.5.3.4.2
6	TraceInformation					
6.1	TraceInformationElement	M	M	M	D	-
6.1.2	domain-supplied-information	M	M	M	D	-
6.1.2.4	(additional actions)				D	-
6.1.2.4.2	converted-encoded-information-types	O	M-	M-	D	see 3.1.2.3.5.3.4.2

Legend (see 3.1.1) :
M = mandatory support
M- = minimal mandatory support
O = optional support
D = discarded
T = translated

3.1.2.3.5.3.4.2 The elements related to the encoded-information-types in the Message Transfer Envelope conveyed with a RN shall be discarded when converting the RN into an AFTN acknowledgement message.

3.1.2.3.5.3.4.3 The *recipient-name* element in the Message Transfer Envelope conveyed with a RN shall be discarded when converting the RN into an AFTN acknowledgement message.

Note.— The Message Transfer and Control Unit uses the information contained in the subject AFTN message to construct an AFTN acknowledgement message.

3.1.2.3.5.4 AMHS Non-delivery Report Conversion

Upon reception by the Message Transfer and Control Unit of an AMHS Non-Delivery Report passed from the ATN Component, this report shall be processed in compliance with the following:

a) the specification of the initial processing performed to determine the Message Transfer and Control Unit ability to convert the report, as included in 3.1.2.3.5.4.1;

b) the specification of how the AFTN service message is generated, if any, and how the AFTN service message components are mapped from AMHS parameters, as included in 3.1.2.3.5.4.2; and

c) the specification of how the Report Transfer Envelope elements are handled, as included in 3.1.2.3.5.4.3.

3.1.2.3.5.4.1 Initial processing of AMHS Non-Delivery Reports

3.1.2.3.5.4.1.1 Upon reception by the Message Transfer and Control Unit of a non-delivery report, passed from the ATN Component to be potentially converted into an AFTN service message, the received non-delivery report shall be processed in one of the following manners:

a) processing as specified in 3.1.2.3.5.4.1.2, if the subject AMHS message has been previously generated by the Message Transfer and Control Unit; or

b) unsuccessful termination of the procedure, if the subject AMHS message has not been previously generated by the Message Transfer and Control Unit, resulting in:

1) logging of the error situation and reporting to a control position; and

2) storage of the non-delivery report for appropriate action at the control position.

3.1.2.3.5.4.1.2 A non-delivery report received by the Message Transfer and Control Unit, and regarding a subject message which had been generated by the Message Transfer and Control Unit, shall be processed by the Message Transfer and Control Unit in one of three mutually exclusive manners:

a) processing as specified in 3.1.2.3.5.4.1.3 if there is no *originally-intended-recipient-name* element with a value different of the *actual-recipient-name* in any of the *per-recipient-fields* elements of the report;

b) processing as follows, if at least one *originally-intended-recipient-name* element in one of the *per-recipient-fields* elements has a value different from the value of the *actual-recipient-name*, and if at least one *per-recipient-fields* element in the report does not meet the same condition:

1) logging of the error situation and reporting to a control position;

2) storage of the non-delivery report and of the corresponding *per-recipient-fields* elements for appropriate action at the control position;

3) processing of the report as specified in 2.3.5.4.1.3 for the *per-recipient-fields* where there is no *originally-intended-recipient-name* element with a value different of the *actual-recipient-name*; or

c) processing as follows, if all *per-recipient-fields* elements of the report include an *originally-intended-recipient-name* element which has a value different from the value of the *actual-recipient-name*:

1) logging of the error situation and reporting to a control position;

2) storage of the non-delivery report and of the corresponding *per-recipient-fields* elements for appropriate action at the control position.

3.1.2.3.5.4.1.3 If the non-delivery report did not cause any error situation to be reported, or for the *per-recipient-fields* of the report which did not cause any error to be reported, the report shall be processed by the Message Transfer and Control Unit in one of the following manners:

a) conversion of the report into an unknown address AFTN service message as specified in 3.1.2.3.5.4.2, if the *non-delivery-diagnostic-code* has the abstract-value "unrecognised-OR-name"; or

b) processing as follows, if the *non-delivery-diagnostic-code* has any abstract-value other than "unrecognised-OR-name"

1) logging of the non-delivery situation and reporting to a control position;

2) storage of the non-delivery report for appropriate action at the control position.

3.1.2.3.5.4.2 Generation of unknown address AFTN service message

3.1.2.3.5.4.2.1 An AMHS Non-Delivery Report received by the Message Transfer and Control Unit, which *non-delivery-diagnostic-code* has the abstract-value "unrecognised-OR-name", and not stored for action at the control position as the result of 3.1.2.3.5.4.1, shall be converted into an AFTN service message to the originator of the subject AFTN message, indicating that an unknown addressee indicator was specified in the subject AFTN message (unknown address AFTN service message) in compliance with:

a) the specification of Annex 10, Volume II, 4.4.11.13.3; and

b) the classification of the components included in Table 3.1.2-16, as specified in the column "action" of Table 3.1.2-16 in accordance with the definition in 3.1.1.

3.1.2.3.5.4.2.2 These components which are classified as "G" shall be generated in compliance with the provisions of Annex 10, Volume II or with the clause referred to in the column "mapping" of Table 3.1.2-16.

3.1.2.3.5.4.2.3 These components which are classified as “T” shall be converted from the AMHS parameter specified in the column “converted from AMHS parameter” of Table 3.1.2-16 and according to the specification in the clause referred to in the column “mapping”.

Table 3.1.2-16. Generation of unknown address AFTN service message

AFTN Message Part	Component	Action	converted from AMHS parameter	Mapping
Heading	Start-of-Heading Character	G	-	see Annex 10, Vol. II, 4.4.15.1.1
	Transmission Identification	G	-	see Annex 10, Vol. II, 4.4.15.1.1
Address	Alignment Function	G	-	see Annex 10, Vol. II, 4.4.15.2.1
	Priority Indicator	G	-	see 3.1.2.3.5.4.2.4
	Addressee Indicator(s)	G	-	see 3.1.2.3.5.4.2.5
	Alignment Function	G	-	see Annex 10, Vol. II, 4.4.15.2.1

AFTN Message Part	Component	Action	converted from AMHS parameter	Mapping
Origin	Filing Time	G	-	see 3.1.2.3.5.4.2.6
	Originator Indicator	G	-	see 3.1.2.3.5.4.2.7
	Priority Alarm	G	-	see Annex 10, Vol. II, 4.4.15.2.2
	Optional Heading Information	X	-	-
	Alignment Function	G	-	see Annex 10, Vol. II, 4.4.15.2.2
	Start-of-Text Character	G	-	see Annex 10, Vol. II, 4.4.15.2.2
Text		T	actual-recipient-name (see Table 3.1.2-17/ Part 1/2.2.1)	see 3.1.2.3.5.4.2.8
Ending	Alignment Function	G	-	see Annex 10, Vol. II, 4.4.15.3.12
	Page-feed sequence	G	-	see Annex 10, Vol. II, 4.4.15.3.12
	End-of-Text Character	G	-	see Annex 10, Vol. II, 4.4.15.3.12

Legend: (see 3.1.1)
G = generated
T = translated
X = excluded (not used)

3.1.2.3.5.4.2.4 The priority indicator component shall take the value of the priority indicator of the subject AFTN message.

3.1.2.3.5.4.2.5 The addressee indicator(s) component shall contain a single AF-Address which is the originator indicator of the subject AFTN message.

3.1.2.3.5.4.2.6 The filing time component, expressed as a date-time group in compliance with Annex 10, Volume II, 4.4.15.2.2.1, shall take the value of the time at which the AFTN service message is generated by the Message Transfer and Control Unit.

3.1.2.3.5.4.2.7 The originator indicator shall be the AFTN Address of the AFTN Component of the AFTN/AMHS Gateway, as specified in 3.1.2.3.2.1.16.

3.1.2.3.5.4.2.8 The value of the message text component shall be structured as follows:

a) a first line composed as specified in Annex 10, Volume II, 4.4.11.13.3, items 1) to 4), using the origin of the subject AFTN message;

b) a second line composed as specified in Annex 10, Volume II, 4.4.11.13.3, items 5) and 6), using the first address line of the subject AFTN message; and

c) the third and following lines as appropriate composed as specified in Annex 10, Volume II, 4.4.11.13.3, items 7) to 9), using the AF-Address(es) translated as specified in 3.1.2.3.5.4.2.9 from the *actual-recipient-name* elements of the *per-recipient-fields* of the Non-Delivery Report which were not stored for action at the control position as the result of 3.1.2.3.5.4.1.2.

3.1.2.3.5.4.2.9 Each *actual-recipient-name* element used to generate an unknown address AFTN service message as specified in item c) of 3.1.2.3.5.4.2.8 above shall be processed for translation into an AF-Address in one of three mutually exclusive manners, after preliminary conversion of the value of all AMHS address attributes from lower case IA5IRV characters, if any, to upper case IA5IRV characters:

a) allocation of the value of the first element of the *organizational-unit-names* attribute to the AF-Address, if this value is a syntactically valid AF-Address and if the *organization-name* attribute has the value "AFTN";

b) determination of an AF-Address matching exactly the MF-Address of the recipient in the User address look-up table maintained in the Message Transfer and Control Unit, if the value of the *organization-name* attribute differs from "AFTN" and if such an exact match can be found; or

c) if none of the conditions in a) and b) can be met, then:

1) logging of the error situation and reporting to a control position; and

2) storage of the MF-Address and of the non-delivery report for appropriate action at the control position.

3.1.2.3.5.4.3 Use of Report Transfer Envelope and Content parameters

3.1.2.3.5.4.3.1 Each of the elements composing the Report Transfer Envelope and Report Transfer Content of an AMHS report to be converted into an AFTN service message in the Message Transfer and Control Unit shall be processed as specified in the column "action" of Table 3.1.2-17.

3.1.2.3.5.4.3.2 These elements shall be handled according to the specification in the clause referred to in the column "mapping" of Table 3.1.2-17.

Note.— Table 3.1.2-17 is structured as a PRL derived from the ISPICS Proforma included in ISO/IEC ISP 10611-3. The columns "Base" and "ISP" are extracted from ISO/IEC ISP 10611-3, and the column "Basic ATS Message Service" specifies the static capability of an AU for the MT-EoS, i.e. the ability to convey, handle and act in relation with the element. The references to the ISP Profile are indicated in the part titles as AMH11/ref where appropriate.

Table 3.1.2-17. Use of Report Transfer Envelope and Content parameters

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
PART 1 : AMH11/A.1.4.3 REPORTTRANSFER						
1	ReportTransferEnvelope	M	M	M	D	-

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Mapping / Notes
2	ReportTransferContent	M	M	M	T	see Part 1/2.1 and 2.2
2.1	(per report fields)					
2.1.1	subject-identifier	M	M	M	D	-
2.1.2	subject-intermediate-trace-information	O	M	M	D	-
2.1.3	original-encoded-information-types	M	M	M	D	-
2.1.4	content-type	M	M	M	D	-
2.1.5	content-identifier	M	M	M	D	-
2.1.6	returned-content	O	M-	M-	D	-
2.1.7	additional-information	O	M-	M-	D	-
2.1.8	extensions	M	M	M	D	-
2.2	per-recipient-fields	M	M	M		
2.2.1	actual-recipient-name	M	M	M	T	see 3.1.2.3.5.4.2.8
2.2.2	originally-specified-recipient-number	M	M	M	D	-
2.2.3	per-recipient-indicators	M	M	M	D	-
2.2.4	last-trace-information	M	M	M	D	-
2.2.5	originally-intended-recipient-name	M	M	M	X	see 3.1.2.3.5.4.1.3
2.2.6	supplementary-information	O	M-	M-	D	-
2.2.7	extensions	M	M	M	D	-

Legend (see 3.1.1) :
M = mandatory support
M- = minimal mandatory support
O = optional support
D = discarded

T = translated
X = excluded

3.1.2.3.5.5 **Action upon reception of AMHS Probe**

3.1.2.3.5.5.1 Upon reception by the Message Transfer and Control Unit of an AMHS probe which content type is either “interpersonal-messaging-1984" or “interpersonal-messaging-1988", the received probe shall be processed in one of the following manners, depending on the abstract-value of the current-encoded-information-types, determined as either the abstract-value of the latest *converted-encoded-information-types*, if existing, in the *trace-information* element, or as the abstract-value of the *original-encoded-information-types* element in the Probe Transfer Envelope if the previous does not exist:

a) processing as specified in 3.1.2.3.5.5.2 if the abstract-value of the current encoded-information-types is “ia5-text” or extended “ia5-text”; or

b) if the abstract-value differs from built-in “ia5-text” and from extended “ia5-text”:

1) rejection of the probe for all the probe recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) “unable-to-transfer” for the *non-delivery-reason-code*; and

ii) “encoded-information-types-unsupported” for the *non-delivery-diagnostic-code*.

3.1.2.3.5.5.2 A probe which was not rejected as the result of 3.1.2.3.5.5.1 shall be processed in one of the following manners:

a) processing as specified in 3.1.2.3.5.5.3 if the abstract-value of the *implicit-conversion-prohibited* in the *per-message-indicators* element in the Probe Transfer Envelope differs from “prohibited”; or

b) if the abstract-value of the element is “prohibited”:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) “conversion-not-performed” for the *non-delivery-reason-code*;

ii) “implicit-conversion-prohibited” for the *non-delivery-diagnostic-code*; and

iii) “unable to convert to AFTN” for the *supplementary-information*.

3.1.2.3.5.5.3 A probe which was not rejected as the result of 3.1.2.3.5.5.2 shall be processed in one of three mutually exclusive manners:

a) if, due to system resource limitation, the value of the element *content-length* in the Probe Transfer Envelope exceeds the conversion capability of the Message Transfer and Control Unit, then:

1) rejection of the message for all the message recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) "unable-to-transfer" for the *non-delivery-reason-code*; and

ii) "content-too-long" for the *non-delivery-diagnostic-code*; or

b) processing as specified in 3.1.2.3.5.5.4 for further conveyance test if the *content-length* does not exceed the conversion capability of the Message Transfer and Control Unit.

Note.— The way to determine the conversion capability of the Message Transfer and Control Unit in terms of message length is a matter local to the AMHS Management Domain operating the AFTN/AMHS Gateway.

3.1.2.3.5.5.4 A probe which was not rejected as the result of 3.1.2.3.5.5.3 shall be processed in one of three mutually exclusive manners, depending on the number of probe recipients towards which the Message Transfer and Control Unit is responsible for conveyance test, and on the AFTN/AMHS Gateway resources:

a) if this number exceeds 21 probe recipients:

1) attempt to split the probe, internally to the Message Transfer and Control Unit, into several probes, each of them with no more than 21 probe recipients:

i) each of the resulting probes having for conveyance test purposes the same *per-probe-fields* in the Probe Transfer Envelope; and

ii) only the *per-recipient-fields* elements in the Probe Transfer Envelope varying between the different resulting probes; and

2) processing of each of these probes as specified in 3.1.2.3.5.5.5;

b) if this number exceeds 21 probe recipients, and if, due to system resource limitation, the splitting attempt made by the gateway as specified in item a) above cannot be properly achieved:

1) rejection of the probe for all the probe recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) "unable-to-transfer" for the *non-delivery-reason-code*;

ii) "too-many-recipients" for the *non-delivery-diagnostic-code*; and

iii) "unable to convert to AFTN due to number of recipients" for the *supplementary-information*; or

d) processing as specified in 3.1.2.3.5.5.5, if this number does not exceed 21 probe recipients.

3.1.2.3.5.5.5 A probe which was not rejected as the result of 3.1.2.3.5.5.4 shall be processed in one of the following manners, depending on the ability of the Message Transfer and Control Unit to translate the originator-name element of the Probe Transfer Envelope into an AF-Address:

a) processing as specified in 3.1.2.3.5.5.6 if either of the following conditions is met:

1) if, after conversion from lower case IA5IRV characters, if any, to upper case IA5IRV characters, the *organization-name* attribute has the value "AFTN" and if the value of the first element of the *organizational-unit-names* is a syntactically valid AF-Address; or

2) if, after conversion from lower case IA5IRV characters, if any, to upper case IA5IRV characters, the value of the *organization-name* attribute differs from "AFTN" and if an AF-Address matching exactly the MF-Address of the originator can be found in the User address look-up table maintained in the Message Transfer and Control Unit; or

b) if none of the conditions 1) or 2) in a) above can be met, then:

1) rejection of the probe for all the probe recipients; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in all the *per-recipient-fields* of the report:

i) "unable-to-transfer" for the *non-delivery-reason-code*;

ii) "invalid-arguments" for the *non-delivery-diagnostic-code*; and

iii) "unable to convert to AFTN due to unrecognized originator O/R address" for the *supplementary-information*.

3.1.2.3.5.5.6 For each probe recipient, a probe which was not rejected as the result of 3.1.2.3.5.5.5 shall be processed in one of the following manners, depending on the ability of the Message Transfer and Control Unit to translate the considered *recipient-name* element of the Probe Transfer Envelope into an AF-Address:

a) processing as specified in 3.1.2.3.5.5.7 if either of the following conditions is met:

1) if, after conversion from lower case IA5IRV characters, if any, to upper case IA5IRV characters, the *organization-name* attribute has the value "AFTN" and if the value of the first element of the *organizational-unit-names* is a syntactically valid AF-Address; or

2) if, after conversion from lower case IA5IRV characters, if any, to upper case IA5IRV characters, the value of the *organization-name* attribute differs from "AFTN" and if an AF-Address matching exactly the MF-Address of the recipient can be found in the User address look-up table maintained in the Message Transfer and Control Unit; or

b) if none of the conditions 1) or 2) in a) above can be met, then:

1) rejection of the probe for the considered recipient; and

2) generation of a non-delivery report as specified in 3.1.2.3.5.6 with the following elements taking the following abstract-values in the corresponding *per-recipient-fields* of the report:

i) "unable-to-transfer" for the *non-delivery-reason-code*; and

ii) "unrecognised-OR-name" for the *non-delivery-diagnostic-code*.

3.1.2.3.5.5.7 For the probe recipients which were not rejected as the result of 3.1.2.3.5.5.6, a delivery-report shall be generated as specified in 3.1.2.3.5.6, if requested, to indicate the successful result of the probe conveyance test.

3.1.2.3.5.6 Generation of AMHS Reports

3.1.2.3.5.6.1 General

3.1.2.3.5.6.1.1 A non-delivery report shall be generated by the Message Transfer and Control Unit:

a) for each message or probe which was rejected at the AFTN/AMHS Gateway, as the result of the procedures described in 3.1.2.3.5.1.1, 3.1.2.3.5.1.4, 3.1.2.3.5.2 and 3.1.2.3.5.5, either for all the recipients or for certain recipients; and

b) as the result of the conversion of an unknown address AFTN service message, as specified in 3.1.2.3.4.4.1.6.

3.1.2.3.5.6.1.2 **Recommendation.—** *When the generation of a non-delivery report is required in relation with the rejection at the AFTN/AMHS Gateway of the subject AMHS message for more than one recipient of the subject AMHS message, a single non-delivery report should be generated to report on the rejection for multiple recipients, using several per-recipient-fields elements in the Report Transfer Content.*

3.1.2.3.5.6.1.3 For each AMHS message which was converted by the Message Transfer and Control Unit as the result of the procedures specified in 3.1.2.3.5.2.2 to 3.1.2.3.5.2.4 and then succesfully passed to the AFTN Component as specified in 3.1.2.3.5.1.6, a delivery report shall be generated by the Message Transfer and Control Unit for each message recipient of which:

a) the *originating-MTA-report-request* element has the abstract-value "report" or "audited-report"; or

b) the *originator-report-request* element has the abstract-value "report"; or

c) both conditions a) and b) above are met.

3.1.2.3.5.6.1.4 **Recommendation.—** *When the generation of a delivery report is required as specified in 3.1.2.3.5.6.1.3 for more than one recipient of the subject AMHS message, a single delivery report should be generated to report on the conveyance towards multiple recipients, using several per-recipient-fields elements in the Report Transfer Content.*

3.1.2.3.5.6.1.5 When the generation of a delivery report is required in relation with the result of a probe conveyance test as specified in 3.1.2.3.5.5, the clauses 3.1.2.3.5.6.1.3 to 3.1.2.3.5.6.1.4 above shall apply with the difference that the event which triggers the generation of the delivery report is the success of the probe conveyance test.

3.1.2.3.5.6.1.6 A report resulting from the clauses above shall be generated as specified in 3.1.2.3.5.6.2.

3.1.2.3.5.6.2 Generation of Report Transfer Envelope and Content

3.1.2.3.5.6.2.1 Each report resulting from the specification of 3.1.2.3.5.6.1 shall be generated by the Message Transfer and Control Unit, in the form of an AMHS Report Transfer Envelope and Report Transfer Content, composed of elements as specified in the column "action" of Table 3.1.2-18.

3.1.2.3.5.6.2.2 These elements which are classified as "G" or "G2" shall be either generated or conditionally generated according to the specification in the clause referred to in the column "generation action" of Table 3.1.2-18.

Note.— Table 3.1.2-18 is structured as a PRL derived from the ISPICS Proforma included in ISO/IEC ISP 10611-3. The columns "Base" and "ISP" are extracted from ISO/IEC ISP 10611-3, and the column "Basic ATS Message Service" specifies the static capability of an AU in relation with the MT-EoS, i.e. the ability to convey, handle and act in relation with the element. The references to the ISP Profile are indicated in the part titles as AMH11/ref where appropriate.

Table 3.1.2-18. Generation of AMHS Report

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Generation action
PART 1 : AMH11/A.1.4.3 REPORTTRANSFER						
1	ReportTransferEnvelope	M	M	M	G	see Part 1/1.1-1.4
1.1	report-identifier	M	M	M	G	see 3.1.2.3.5.6.2.3 and Part 2/1
1.2	report-destination-name	M	M	M	G	see 3.1.2.3.5.6.2.6
1.3	trace-information	M	M	M	G	see 3.1.2.3.5.6.2.7
1.4	extensions	M	M	M		see 3.1.2.3.5.6.2.8
1.4.1	message-security-label	O	M-	M-	X	-
1.4.2	originator-and-DL-expansion-history	M	M	M	G2	see 3.1.2.3.5.6.2.9
1.4.3	reporting-DL-name	O	M-	M-	X	-
1.4.4	reporting-MTA-certificate	O	M-	M-	X	-
1.4.5	report-origin-authentication-check	O	M-	M-	X	-
1.4.6	internal-trace-information	M	M	M	G	see 3.1.2.3.5.6.2.10
2	ReportTransferContent	M	M	M	G	see Part1/2.1 and 2.2
2.1	(per report fields)					
2.1.1	subject-identifier	M	M	M	G	see 3.1.2.3.5.6.2.11
2.1.2	subject-intermediate-trace-information	O	M	M	G2	see 3.1.2.3.5.6.2.12
2.1.3	original-encoded-information-types	M	M	M	G	see 3.1.2.3.5.6.2.13
2.1.4	content-type	M	M	M	G	see 3.1.2.3.5.6.2.14
2.1.5	content-identifier	M	M	M	G2	see 3.1.2.3.5.6.2.15

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Generation action
2.1.6	returned-content	O	M-	M-	G2	see 3.1.2.3.5.6.2.16
2.1.7	additional-information	O	M-	M-	X	-
2.1.8	extensions	M	M	M		see 3.1.2.3.5.6.2.8
2.1.8.1	content-correlator	M	M	M	G2	see 3.1.2.3.5.6.2.17
2.2	per-recipient-fields	M	M	M		see Part1/2.2.1-2.2.7
2.2.1	actual-recipient-name	M	M	M	G	see 3.1.2.3.5.6.2.18
2.2.2	originally-specified-recipient-number	M	M	M	G	see 3.1.2.3.5.6.2.19
2.2.3	per-recipient-indicators	M	M	M	G	see 3.1.2.3.5.6.2.20
2.2.4	last-trace-information	M	M	M	G	see Part 2/7
2.2.5	originally-intended-recipient-name	M	M	M	G2	see 3.1.2.3.5.6.2.26
2.2.6	supplementary-information	O	M-	M-	G2	see 3.1.2.3.5.6.2.27
2.2.7	extensions	M	M	M		see 3.1.2.3.5.6.2.8
2.2.7.1	redirection-history	M	M	M	G2	see 3.1.2.3.5.6.2.28
2.2.7.2	physical-forwarding-address	O	M-	M-	X	-
2.2.7.3	recipient-certificate	O	M-	M-	X	-
2.2.7.4	proof-of-delivery	O	M-	M-	X	-
PART 2 : AMH11/A.1.5 COMMON DATA TYPES						
1	MTSIdentifier					
1.1	global-domain-identifier	M	M	M	G	see 3.1.2.3.5.6.2.4 and Part 2/2
1.2	local-identifier	M	M	M	G	see 3.1.2.3.5.6.2.5

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Generation action
2	GlobalDomainIdentifier					
2.1	country-name	M	M	M		see 3.1.2.3.5.6.2.29
2.2	administration-domain-name	M	M	M		see 3.1.2.3.5.6.2.30
2.3	private-domain-identifier	M	M	M		see 3.1.2.3.5.6.2.31
6	TraceInformation					
6.1	TraceInformationElement	M	M	M	G	see Part 2/6.1.1 and 6.1.2
6.1.1	global-domain-identifier	M	M	M	G	see 3.1.2.3.5.6.2.32 and Part 2/2
6.1.2	domain-supplied-information	M	M	M	G	see Part 2/6.1.2.1-6.1.2.4
6.1.2.1	arrival-time	M	M	M	G	see 3.1.2.3.5.6.2.33
6.1.2.2	routing-action	M	M	M	G	see Part 2/6.1.2.2.1 and 6.1.2.2.2
6.1.2.2.1	relayed	M	M	M	G	see 3.1.2.3.5.6.2.34
6.1.2.2.2	rerouted	O	C1	C1	X	-
6.1.2.3	attempted-domain	O	C1	C1	X	-
6.1.2.4	(additional actions)					
6.1.2.4.1	deferred-time	M	C2	C2	X	-
6.1.2.4.2	converted-encoded-information-types	O	M-	M-	X	-
6.1.2.4.3	other-actions	O	M-	M-	X	-
6.1.2.4.3.1	redirected	O	M-	M-	X	-
6.1.2.4.3.2	dl-operation	O	M-	M-	X	-

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Generation action
7	LastTraceInformation					
7.1	arrival-time	M	M	M	G	see 3.1.2.3.5.6.2.21
7.2	converted-encoded-information-types	M	M	M	G2	see 3.1.2.3.5.6.2.22
7.3	report-type	M	M	M	G	see Part 2/7.3.1and 7.3.2
7.3.1	delivery	M	M	M	G2	see Part 2/7.3.1.1 and 7.3.1.2
7.3.1.1	message-delivery-time	M	M	M	G	see 3.1.2.3.5.6.2.23
7.3.1.2	type-of-MTS-user	M	M	M	G	see 3.1.2.3.5.6.2.24
7.3.2	non-delivery	M	M	M	G2	see Part 2/7.3.2.1 and 7.3.2.2
7.3.2.1	non-delivery-reason-code	M	M	M	G	see 3.1.2.3.5.6.2.25
7.3.2.2	non-delivery-diagnostic-code	M	M	M	G	see 3.1.2.3.5.6.2.25
PART 3 : AMH11/A.1.6 EXTENSION DATA TYPES						
1	ExtensionField					
1.1	type	M	M	M	G	see Part 3/1.1.1 and 1.1.2
1.1.1	standard-extension	M	M	M	G	see 3.1.2.3.5.6.2.8
1.1.2	private-extension	O	M-	M-	X	-
1.2	criticality	M	M	M	G	see 3.1.2.3.5.6.2.8
1.3	value	M	M	M	G	see 3.1.2.3.5.6.2.8
5	InternalTraceInformation					
5.1	global-domain-identifier	M	M	M	G	see 3.1.2.3.5.6.2.32

Ref	Element	Base	ISP	Basic ATS Mess. Service	Action	Generation action
5.2	mta-name	M	M	M	G	see 3.1.2.3.5.6.2.35
5.3	mta-supplied-information	M	M	M	G	see Part 3/5.3.1-5.3.4
5.3.1	arrival-time	M	M	M	G	see 3.1.2.3.5.6.2.33
5.3.2	routing-action	M	M	M	G	see Part 3/5.3.2.1-5.3.2.2
5.3.2.1	relayed	M	M	M	G	see 3.1.2.3.5.6.2.34
5.3.2.2	rerouted	O	C1	C1	X	-
5.3.3	attempted	O	C1	C1	X	-
5.3.4	(additional actions)					
5.3.4.1	deferred-time	M	C2	C2	X	-
5.3.4.2	converted-encoded-information-types	O	M-	M-	X	-
5.3.4.3	other-actions	O	M-	M-	X	-
5.3.4.3.1	redirected	O	M-	M-	X	-
5.3.4.3.2	dl-operation	O	M-	M-	X	-

Legend (see 3.1.1) :
M = mandatory support
M- = minimal mandatory support
O = optional support
I = out of scope
- = not applicable
C1 = if rerouting is supported then M else M-
C2 = if deferred delivery is supported then M else M-
G = generated
G2 = conditionally generated
X = excluded (not used)

3.1.2.3.5.6.2.3 The element *report-identifier* in the Report Transfer Envelope shall:

a) be generated locally so as to ensure that it distinguishes the report from all other messages, probes or reports generated in the AMHS, as specified in ISO/IEC 10021-4, 12.2.1.3.1.1; and

b) be composed as specified in Table 3.1.2-18/Part 2/1.

3.1.2.3.5.6.2.4 The element *global-domain-identifier* in the *report-identifier*, or in the *trace-information*, or in the *internal-trace-information* shall:

a) identify the AMHS Management Domain operating the AFTN/AMHS Gateway; and

b) be composed as specified in Table 3.1.2-18/Part 2/2.

3.1.2.3.5.6.2.5 The element *local-identifier* in the *report-identifier* shall be generated locally so as to ensure that it distinguishes the report from all other messages, probes or reports generated in the AMHS Management Domain operating the AFTN/AMHS Gateway.

3.1.2.3.5.6.2.6 The *report-destination-name* element in the Report Transfer Envelope shall be one of the following:

a) the last OR-name in the *DL-expansion-history* element, if present, of the subject AMHS message as specified in Table 3.1.2-12/Part 1/1.1.11.11; or

b) the *originator-name* of the subject AMHS message, as specified in Table 3.1.2-12/Part 1/1.1.2, if there is no *DL-expansion-history* element in the subject AMHS message.

3.1.2.3.5.6.2.7 The first *trace-information-element* in the *trace-information* of the Report Transfer Envelope shall be generated as specified in Table 3.1.2-18/Part 2/6.

3.1.2.3.5.6.2.8 Only extensions of type “standard-extension” as defined in the base standards shall be used, as further specified in the classification of Table 3.1.2-18.

3.1.2.3.5.6.2.9 If a *DL-expansion-history* element as specified in Table 3.1.2-12/Part 1/1.1.11.11 was present in the subject AMHS message, the *originator-and-DL-expansion-history* element shall be generated as the sequence of the *originator-name* of the subject AMHS message, as specified in Table 3.1.2-12/Part 1/1.1.2, and of the aforementioned *DL-expansion-history* element of the subject AMHS message.

3.1.2.3.5.6.2.10 The first *internal-trace-information-element* in the *internal-trace-information* of the Report Transfer Envelope shall be generated as specified in Table 3.1.2-18/Part 3/5.

3.1.2.3.5.6.2.11 The *subject-identifier* element in the Report Transfer Content shall take the value of the *message-identifier* element of the subject AMHS message as specified in Table 3.1.2-12/Part 1/1.1.1.

3.1.2.3.5.6.2.12 The *subject-intermediate-trace-information* element in the Report Transfer Content shall take the value which the *trace-information* element of the subject AMHS message as specified in Table 3.1.2-12/Part 1/1.1.10 had when the subject AMHS message entered the AMHS Management Domain operating the Message Transfer and Control Unit, if and only if the *originating-MTA-report-request* element in the *per-recipient-indicators* of all the subject AMHS message recipients in the subject Message Transfer Envelope has the abstract-value "audited-report".

3.1.2.3.5.6.2.13 The *original-encoded-information-types* element in the Report Transfer Content shall take the value of the *original-encoded-information-types* element of the subject AMHS message as specified in Table 3.1.2-12/Part 1/1.1.3.

3.1.2.3.5.6.2.14 The *content-type* element in the Report Transfer Content shall take the value of the *content-type* element of the subject AMHS message as specified in Table 3.1.2-12/Part 1/1.1.4.

3.1.2.3.5.6.2.15 The *content-identifier* element in the Report Transfer Content shall either:

a) take the value of the *content-identifier* element of the subject AMHS message as specified in Table 3.1.2-12/Part 1/1.1.5, if present; or

b) be omitted in the report if there is no such element in the subject AMHS message.

3.1.2.3.5.6.2.16 The *returned-content* element in the Report Transfer Content shall optionally take the value of the *content* of the subject AMHS message, if and only if the *content-return-request* element in the *per-message-indicators* of the subject AMHS message in the subject Message Transfer Envelope has the abstract-value "content-return-requested".

Note.— The Message Transfer and Control Unit is not mandated to implement the Return Of Content (RoC) Optional Functional Group as defined in ISO/IEC ISP 10611-1.

3.1.2.3.5.6.2.17 The *content-correlator* element in the Report Transfer Content shall either:

a) take the value of the *content-correlator* element of the subject AMHS message as specified in Table 3.1.2-12/Part 1/1.1.11.10, if present; or

b) be omitted in the report if there is no such element in the subject AMHS message.

3.1.2.3.5.6.2.18 The *actual-recipient-name* element in a *per-recipient-fields* element of the Report Transfer Content shall take the value of the corresponding *recipient-name* element in the *per-recipient-fields* of the subject AMHS message as specified in Table 3.1.2-12/Part 1/1.2.1.

3.1.2.3.5.6.2.19 The *originally-specified-recipient-number* element in a *per-recipient-fields* element of the Report Transfer Content shall take the value of the corresponding *originally-specified-recipient-number* element in the *per-recipient-fields* of the subject AMHS message as specified in Table 3.1.2-12/Part 1/1.2.2.

3.1.2.3.5.6.2.20 The *per-recipient-indicators* element in a *per-recipient-fields* element of the Report Transfer Content shall take the value of the corresponding *per-recipient-indicators* element in the *per-recipient-fields* of the subject AMHS message as specified in Table 3.1.2-12/Part 1/1.2.3.

3.1.2.3.5.6.2.21 The *arrival-time* element in the *last-trace-information* of a *per-recipient-fields* element shall take the value of the time at which the subject AMHS message entered the AMHS Management Domain operating the AFTN/AMHS Gateway, as found in the last *trace-information-element* of the subject AMHS message, as specified in Table 3.1.2-12/Part 2/6.1.2.1.

3.1.2.3.5.6.2.22 The *converted-encoded-information-types* element in the *last-trace-information* of a *per-recipient-fields* element shall either:

a) take the last value of the *converted-encoded-information-types* element in the *trace-information* of the subject AMHS message, as specified in Table 3.1.2-12/Part 2/6.1.2.4.2, if this element exists; or

b) be omitted in the report, if no such element is present in the *trace-information* of the subject AMHS message.

3.1.2.3.5.6.2.23 If the report is a delivery-report, the *message-delivery-time* element in the *last-trace-information* of a *per-recipient-fields* element shall be the time at which the subject AMHS message has been successfully passed to the AFTN Component by the Message Transfer and Control Unit.

3.1.2.3.5.6.2.24 If the report is a delivery-report, the *type-of-MTS-user* element in the *last-trace-information* of a *per-recipient-fields* element shall take the abstract-value “other”.

3.1.2.3.5.6.2.25 If the report is a non-delivery-report, the *non-delivery-reason-code* and *non-delivery-diagnostic-code* elements in the *last-trace-information* of a *per-recipient-fields* element shall take the abstract-values specified in the clause which caused the generation of the report.

3.1.2.3.5.6.2.26 The *originally-intended-recipient-name* element in a *per-recipient-fields* element shall either:

a) take the value of the first O/R name found in the *redirection-history* element of the subject AMHS message, if present, as specified in Table 3.1.2-12/Part 1/1.2.5.13; or

b) be omitted in the report if there is no *redirection-history* element in the subject AMHS message.

3.1.2.3.5.6.2.27 The *supplementary-information* element in a *per-recipient-fields* element shall take one of the following values:

a) the value “This report only indicates successful (potential) conversion to AFTN, not delivery to a recipient” if the report is a delivery-report; or

b) the value, if any, specified in the clause which caused the generation of the report if it is a non-delivery-report.

3.1.2.3.5.6.2.28 The *redirection-history* element in a *per-recipient-fields* element shall either:

a) take the value of the *redirection-history* element of the subject AMHS message, if present, as specified in Table 3.1.2-12/Part 1/1.2.5.13; or

b) be omitted in the report if there is no *redirection-history* element in the subject AMHS message.

3.1.2.3.5.6.2.29 The element *country-name* in the *global-domain-identifier* element of the *MTS-identifier* and of the first *trace-information-element* shall:

a) be part of the identification of the AMHS Management Domain operating the AFTN/AMHS Gateway by taking one of the following values:

1) the two-character alphabetical country-indicator as specified in ISO 3166 for the country, or for one of the countries, where the AMHS Management Domain has been registered, if the AMHS Management Domain has been subject to national or multi-national registration; or

2) a two-character alphabetical indicator dedicated to an international organization, if the AMHS Management Domain has been subject to international registration as defined in ITU-T Recommendation X.666; and

b) be encoded as a Printable String.

3.1.2.3.5.6.2.30 The element *administration-domain-name* in the *global-domain-identifier* element of the *MTS-identifier* and of the first *trace-information-element* shall:

a) be part of the identification of the AMHS Management Domain operating the AFTN/AMHS Gateway by taking one of the following values, depending on its status:

1) the name of the ADMD under which the AMHS Management Domain has been registered, either nationally or internationally, if the AMHS Management Domain operates as an ADMD;

2) the name of the ADMD to which the AMHS Management Domain is connected, if the AMHS Management Domain operates as a PRMD; or

3) the value single-space if the AMHS Management Domain operates as a PRMD and is unique with regard to the *country-name* identifying the area where it is registered, either nationally or internationally; and

b) be encoded as a Printable String.

3.1.2.3.5.6.2.31 The element *private-domain-identifier* in the *global-domain-identifier* element of the *MTS-identifier* and of the first *trace-information-element* shall be handled in one of the following manners, depending on the status under which the AMHS Management Domain operates:

a) generation of the element, with the value of the name of the PRMD, encoded as a Printable String, if the AMHS Management Domain operates as an PRMD; or

b) omission in the *global-domain-identifier* if the AMHS Management Domain operates as an ADMD.

3.1.2.3.5.6.2.32 The element *global-domain-identifier* in the *trace-information* or in the *internal-trace-information* shall:

a) identify the AMHS Management Domain operating the AFTN/AMHS Gateway; and

b) be composed as specified in Table 3.1.2-18 / Part 2/2.

3.1.2.3.5.6.2.33 The element *arrival-time* in the first element of *trace-information* or of *internal-trace-information* shall take the semantic value of the time when the report was generated by the Message Transfer and Control Unit for conveyance in the AMHS.

3.1.2.3.5.6.2.34 The element *routing-action* in the first element of *trace-information* or of *internal-trace-information* shall take the abstract-value “relayed”.

3.1.2.3.5.6.2.35 The element *mta-name* in the first element of *internal-trace-information* shall be the mta-name assigned to the Message Transfer and Control Unit included in the AFTN/AMHS Gateway.

Note.— The structure of the mta-name of the Message Transfer and Control Unit included in an AFTN/AMHS Gateway within an AMHS Management Domain is a matter of policy internal to the AMHS Management Domain.

3.1.3 ATN PASS-THROUGH SERVICE

3.1.3.1 System level provisions

3.1.3.1.1 The ATN Pass-Through Service shall provide a message environment for the exchange of IA-5 encoded AFTN messages over the ATN Internet Communications Service and with the AFTN via the AFTN/ATN Type A gateway.

Note 1.— This service does not provide classical store and forward messages services such as found in the AFTN and the ATS Message Service, nor is it visible to users at AFTN stations.

Note 2.— As a matter of organisations' policy, the implementation of the ATS Message Service may be deferred. In order to take early advantage of the enhanced connectivity provided by the ATN, ATS Organisations with such a policy may implement and operate in the interim the ATN Pass-Through Service. This service provides connectivity for the AFTN traffic as presently defined in Annex 10, Volume II, through the ATN. The interoperability between the ATS Message Service and the ATN Pass-Through Service is a local implementation matter.

3.1.3.1.2 **Recommendation.—** *ATS Organisations which choose to implement the ATN Pass-Through Service should plan to implement the ATS Message Service at the earliest possible time.*

3.1.3.1.3 **Recommendation**.— *ATS Organisations which choose to implement the ATN Pass-Through Service should provide the interoperability facilities to the ATS Message Service implementations.*

3.1.3.1.4 AFTN/ATN Type A Gateway users

The AFTN/ATN Type A Gateway users shall consist of AFTN stations (as defined in Annex 10, Volume II) exchanging AFTN messages.

3.1.3.1.5 AFTN/ATN Type A Gateway model

If an AFTN/ATN Type A Gateway is connected to an AFTN Centre which is capable of using only ITA-2 format, the AFTN Component of the gateway shall convert messages to/from the IA-5 format.

Note.— An ATS organisation may choose to connect an AFTN/ATN Type A Gateway to the AFTN only via its AFTN Centre. In this case, some requirements placed on the AFTN Component may not have to be fulfilled, provided that the AFTN Centre and AFTN/ATN Type A Gateway together fulfill all requirements.

3.1.3.1.5.1 AFTN/ATN Type A Gateway information model

The AFTN/ATN Type A Gateway information elements shall consist only of AFTN messages.

3.1.3.1.5.2 Security and management models

Recommendation.— *Security should be obtained by procedural means rather than by technical features inherent in the ATN Pass-Through Service.*

Note 1.— The security at each AFTN/ATN Type A Gateway is deemed a local issue to be addressed by the ATS organisation in charge of the system.

Note 2.— Management is limited to the logging provisions which are defined for the AFTN/ATN Type A Gateway. No provision is made for retrieval or exchange of this information, which is deemed a local issue to be addressed by the authority in charge of the system.

3.1.3.1.6 AFTN/ATN Type A Gateway System configurations

3.1.3.1.6.1 The minimal set of systems implemented and operated by an ATS organisation for the ATN Pass-Through Service shall be one AFTN/ATN Type A Gateway system.

3.1.3.1.6.2 The minimal set of communications circuits implemented by an ATS organisation operating an AFTN/ATN Type A Gateway shall be:

a) when integrated with an AFTN centre, access to one ATN subnetwork;

b) when not integrated with an AFTN centre, one AFTN circuit utilizing a code and byte independent procedure and access to one ATN subnetwork; or

c) when not integrated with an AFTN centre, one AFTN circuit utilizing any Annex 10, Volume I controlled or Volume II uncontrolled circuit procedure and access to one ATN subnetwork.

Note.— The effect of selecting either 3.1.3.1.6.2 a) or b) is the elimination of the requirement for the AFTN/ATN Type A Gateway to implement the manual teletypewriter procedures, such as service message procedures, channel-check and transmission identification procedures, and code conversion procedures contained in Annex 10, Volume II.

3.1.3.1.7 AFTN/ATN Type A Gateway System naming principles

Naming for each AFTN/ATN Type A Gateway system shall consist of an AP-title set and an AE-qualifier set, as specified in 4.3.2.

3.1.3.1.8 AFTN/ATN Type A Gateway System addressing principles

3.1.3.1.8.1 There shall be two address forms used in the AFTN/ATN Type A Gateway System:

a) an AFTN address comprising an AFTN addressee indicator as specified in Annex 10, Volume II, 4.4.3.1.2 and 4.4.14.2; and

b) an ATN end-system address comprising a facility designation, as specified in 4.

3.1.3.1.8.2 A facility designation shall be assigned to each AFTN/ATN Type A Gateway.

3.1.3.1.9 Routing principles

Routing of messages shall be provided by the AFTN Centres to which the AFTN/ATN Type A Gateway is connected.

3.1.3.1.10 Processing of communication failure

If, for any reason, the Message Transfer and Control Unit is unable to accept AFTN messages passed by the AFTN Component, then the AFTN Component shall handle this situation in compliance with the provisions of Annex 10, Volume II, 4.4.1.5.2.3.

Note.— Such a condition may be caused by the inability of the Message Transfer and Control Unit to pass messages to the ATN Component.

3.1.3.2 ATN Pass-Through Service Specification

3.1.3.2.1 An AFTN/ATN Type A Gateway shall consist of the following three logical components:

a) AFTN component;

b) ATN component; and

c) Message transfer and control unit.

3.1.3.2.2 The three logical components shall interact according to the architecture specified in 4.

3.1.3.2.3 For both the configurations specified in 3.1.3.1.6.2 a) and b), the ATN Pass-Through Service shall be totally transparent for AFTN messages and AFTN service messages to the users of the service, except when applying the procedures for address stripping.

3.1.3.2.4 For the configuration specified in 3.1.3.1.6.2 c), the AFTN/ATN Type A Gateways shall handle the AFTN procedures as specified in Annex 10, Volume II.

3.1.3.3 AFTN/ATN Type A Gateway Specification

3.1.3.3.1 AFTN component

3.1.3.3.1.1 The AFTN component shall handle the interface to the AFTN and provide an interface to the Message Transfer and Control Unit.

3.1.3.3.1.2 The AFTN component shall implement:

a) all the applicable requirements of Annex 10, Volume II, in a manner so as to be indistinguishable from an operational AFTN station by the AFTN Centre to which the gateway is connected; and

b) additional requirements which are not placed on AFTN stations by Annex 10, Volume II but which are necessary due to the AFTN Component requirements pertaining to an AFTN/ATN Type A Gateway.

3.1.3.3.1.3 The AFTN component shall incorporate an AFTN procedure handler that provides all of the AFTN functions prescribed for the interface to the AFTN.

3.1.3.3.1.4 The AFTN Component shall isolate all AFTN procedures from the Message Transfer and Control Unit Component.

Note.— The AFTN procedure handler includes the procedures for managing the order of AFTN messages based on the transmission priority specified. Using the AFTN procedure handler for managing priority eliminates the need for the Message Transfer and Control Unit to manage message priorities.

3.1.3.3.1.5 The AFTN Component of an AFTN/ATN Type A Gateway shall perform short term retention of all messages transmitted towards the AFTN in a manner equivalent to that specified for an AFTN communication centre in Annex 10, Volume II, 4.4.1.7 to provide recovery from communication protocol errors.

3.1.3.3.1.6 The AFTN Component shall perform long-term retention of all AFTN messages, in their entirety, that it generates, for a period of at least thirty days.

3.1.3.3.1.7 The AFTN Component shall perform long-term retention of the heading, address and origin parts of all messages received from the Message Transfer and Control Unit and the action taken thereon, for a period of at least thirty days.

3.1.3.3.2 ATN component

3.1.3.3.2.1 The ATN Component shall implement the procedures required of an ATN End System as specified by the ATS Message protocol stack Type A.

3.1.3.3.2.2 ATN Component service

3.1.3.3.2.2.1 The ATN Component service shall consist of a single service primitive between it and the Message Transfer and Control Unit, the GA-Data request and indication.

Table 3.1.3-1. ATN Component Service

GA-Data Service Primitive	Req	Ind
User Data	M	M(=)
Called Address	M	
Calling Address	M	M(=)
Priority (transmission)	U	U(=)

3.1.3.3.2.2.2 The User Data parameter shall contain the IA5 form of a complete AFTN message, as defined in Annex 10, Volume II.

3.1.3.3.2.2.3 The Called Address parameter shall contain the ATN-end system id of the destination AFTN/ATN Type A Gateway consisting of the 8-character facility designation as defined in 4.

3.1.3.3.2.2.4 The Calling Address parameter shall contain the ATN-end system id of the AFTN/ATN Type A Gateway consisting of the 8-character facility designation as defined in 4.

3.1.3.3.2.2.5 The AFTN Priority parameter, if present, shall contain the AFTN priority indicator of the AFTN message, as defined in Annex 10, Volume II.

3.1.3.3.2.3 The ATS Message protocol stack Type A shall consist of protocols and procedures specified in 4; and consisting of:

a) the ATN Component Control Function, which incorporates the Control Function of the Upper Layer Communication Service as specified in 4.3.3 and the additional provisions specified in 3.1.3.3.2.4;

b) the Dialogue Service as specified in 4.2, consisting of:

1) the Association Control Service Element,

2) the Presentation Efficiency enhancements, and

3) the Session Efficiency enhancements

c) the Application Layer Naming and Context Definition as specified in 4.3.2; and

d) the ATN Communication Services requirements as specified in 5.

3.1.3.3.2.4 ATN Component Control Function

Note.— The specification of the ATN Component Control Function (CF) does not constrain the implementation of the ATN Component as long as the latter exhibits the external behaviour of the CF as specified.

3.1.3.3.2.4.1 The ATN Component control function (CF) shall map the GA-Data requests and indications to and from the Dialogue Service as specified in 4.

3.1.3.3.2.4.2 Upon receipt of a GA-Data request, the CF shall determine if a dialogue exists with the destination ATN End-System by examining the Called Address parameter.

3.1.3.3.2.4.3 If a dialogue does not exist, the CF shall formulate a D-START-request primitive.

3.1.3.3.2.4.4 The parameters of the D-START-request shall be set according to Table 3.1.3-2.

Table 3.1.3-2. D-START-request/indication Parameters

D-START-request Parameter	GA-Data Request Parameter	Value
Called Peer ID	Called Address	atn-facility designation
Calling Peer ID	Calling Address	atn-facility designation
DS-User Version Number	-------	1
Security Requirements	-------	<not used>
QOS Parameters Routing class Priority Residual Error Rate	 ------ AFTN Priority -----	 “ATSC: No Traffic Type Policy Preference” see Table 3.1.3-7 abstract-value “high”
User Data	-------	<not used>

3.1.3.3.2.4.5 Upon receipt of an D-START-indication, the CF shall determine if the parameters values are as indicated in Table 3.1.3-2.

3.1.3.3.2.4.6 If the parameters received in a D-START indication are acceptable and sufficient resources available to support the association, the CF shall accept the association by sending a D-START-response, in which the parameters are set according to Table 3.1.3-3 with the Result parameter set to the abstract-value "accepted".

Note.— The actual definition of "acceptable values" for the parameters of the D-START-indication is a local matter.

3.1.3.3.2.4.7 If the parameters received in a D-START indication are unacceptable or there are insufficient resources available to support the association, the CF shall reject the association by sending an D-START-response, in which the parameters are set according to Table 3.1.3-3, with the Result parameter set to the abstract-value "rejected (permanent)" in the case of invalid parameters and set to the abstract-value "rejected (transient)"if there are insufficient resources.

Table 3.1.3-3. D-START-response/confirmation Parameters

D-START-response Parameter	GA - Data Indication Parameter	Value
DS-User Version	-----	1
Security Requirements	------	<not used>
QOS (routing class, priority, and residual error rate)	-----	<not used>
User Data		<not used>
Result	------	"accepted" "rejected (permanent)" "rejected (transient)"

Note 1.— The actual definition of "unacceptable values" for the parameters of the D-START-indication is a local matter.

Note 2.— The use of a security policy (such as only accepting associations from particular remote ATN end systems) to limit acceptance of associations is a local matter.

3.1.3.3.2.4.8 If the result parameter in the D-START-confirmation is set to the abstract-value "rejected (transient)", the ATN Component CF shall:

a) re-attempt (a locally defined number of times) the establishment of a dialogue with the same gateway; and

b) if subsequent to the procedure defined in a) a dialogue still cannot be established, attempt the establishment of a dialogue with a prioritised list of gateways which are defined as being alternates to the one which has been determined as unreachable.

3.1.3.3.2.4.9 If the result parameter in the D-START-confirmation is set to the abstract-value "rejected (permanent)", the ATN Component CF shall attempt (a locally defined number of times) the establishment of a dialogue with a prioritised list of gateways which are defined as being alternates to the one which has been determined as unreachable.

3.1.3.3.2.4.10 If any of the QOS parameters in the D-START-confirmation is unacceptable, the ATN Component CF shall perform the following:

a) abort the dialogue as specified in 3.1.3.3.2.4.19; and

b) attempt (a locally number of times) the establishment of a dialogue with a prioritised list of gateways which are defined as being alternates to the one which has been determined as unreachable with an acceptable QOS.

Note.— The actual definition of "unacceptable values" for the parameters of the D-START-confirmation is a local matter.

3.1.3.3.2.4.11 If subsequent to the procedures defined in 3.1.3.3.2.4.8 to 3.1.3.3.2.4.10 a dialogue still cannot be established, the ATN Component CF shall:

a) log the error situation;

b) store the message for further processing and

c) ensure that the message is not discarded.

Note.— The processing to be performed in application of this clause is a local matter.

3.1.3.3.2.4.12 Upon the completion of the dialogue set-up, or in the case of using an existing dialogue, the CF shall formulate a D-DATA request by taking the data in the User Data parameter in the GA-Data-request and encoding it as the user data field in the D-DATA-request.

3.1.3.3.2.4.13 The data received in the User Data parameter of the GA-Data-request is the complete AFTN message, which shall be passed transparently to the destination system.

3.1.3.3.2.4.14 Upon the receipt of a D-DATA-indication, the CF shall extract the user data and place it in the User Data parameter of the GA-Data-indication.

3.1.3.3.2.4.15 If the CF does not have any data to send over a dialogue for a time period t_1, it shall release the dialogue by formulating an D-END-request.

Note.— The time period to wait before releasing a dialogue is a local matter to be determined by cost and expected data traffic.

3.1.3.3.2.4.16 The parameters of the D-END-request shall be set according to Table 3.1.3-4.

Table 3.1.3-4. D-END-request Parameters

D-END-request Parameter	Type	Value
User Data		<not used>

3.1.3.3.2.4.17 Upon receiving an D-END-indication, the CF shall release the dialogue as soon as it no longer has any data to send (over that dialogue) by formulating a D-END-response.

3.1.3.3.2.4.18 The parameters of the D-END-response shall be set according to Table 3.1.3-5.

Table 3.1.3-5. D-END-response Parameters

D-END-response Parameter	Type	Value
Result	INT	abstract-value “accepted”
User Data	----	<not used>

3.1.3.3.2.4.19 For immediate termination of the dialogue, the D-ABORT parameters shall be set according to Table 3.1.3-6.

Note.— The conditions under which this primitive is used is a local matter.

Table 3.1.3-6. D-ABORT-request Parameters

D-ABORT-request Parameter	Type	Value
Originator	------	<not used>
User Data	------	<not used>

3.1.3.3.2.4.20 Upon receipt of a D-ABORT-indication the dialogue shall be considered to have failed.

Note.— The recovery mechanisms, if any, to be applied are a matter of local implementation choices.

3.1.3.3.2.4.21 Upon receipt of a D-P-ABORT-indication the dialogue shall be terminated.

Note.— The recovery mechanisms, if any, to be applied are a matter of local implementation choices.

3.1.3.3.2.5 Dialogue Service QOS (Priority)

3.1.3.3.2.5.1 For transmission of messages across the ATN, the AFTN priority indicators, as found in Annex 10, Volume II, 4.4.1.2, shall map to Dialogue Service QOS (Priorities) in accordance with Table 3.1.3-7.

Table 3.1.3-7. AFTN/ATN Priority Mapping

AFTN Priority Indicator	Dialogue Service Quality of Service (Priority) Parameter
SS	distress communications
DD	urgent communications
FF	high priority flight safety messages
GG	flight regularity communications
KK	aeronautical administrative messages

3.1.3.3.2.5.2 The ATN component shall process incoming and outgoing messages according to the priority of the message.

3.1.3.3.3 Message Transfer and Control Unit Component

Note.— The Message Transfer and Control Unit Component provides a bi-directional conversion facility between the AFTN component and the ATN component and consists of:

a) a set of general functions as specified in 3.1.3.3.3.1;

b) a set of AFTN to ATN mapping functions as specified in 3.1.3.3.3.3;

c) a set of ATN to AFTN mapping functions as specified in 3.1.3.3.3.4;

d) a set of interface requirements between the Message Transfer and Control Unit Component and the ATN Component as specified in 3.1.3.3.3.5; and

e) a set of interface requirements between the Message Transfer and Control Unit Component and the AFTN Component as specified in 3.1.3.3.3.6.

3.1.3.3.3.1 General functions

The Message Transfer and Control Unit of an AFTN/ATN Type A Gateway shall log all messages and information related to the following events that have occurred at its interfaces with the ATN Component and with the AFTN Component, and in its internal procedures:

a) the messages transferred out (to the ATN Component);

b) the messages transferred in (from the ATN Component);

c) the AFTN messages conveyed out (to the AFTN Component);

d) the AFTN messages conveyed in (from the AFTN Component);

e) the AFTN service messages indicating unknown addressee indicator conveyed out (to the AFTN Component).

Note.— This requirement is not intended to fulfill the 30 day message requirements for an AFTN station.

3.1.3.3.3.2 Determination of gateway address

3.1.3.3.3.2.1 The Message Transfer and Control Unit Component shall maintain an address mapping function which maps between an AFTN addressee indicator and the ATN address of the AFTN/ATN Type A Gateway via which the addressee may be reached.

3.1.3.3.3.2.2 The address mapping function shall, at a minimum, provide the following mappings:

a) a map from an entire AFTN address to an ATN address,

b) a map from sets of AFTN addresses based on a portion of the AFTN address to a single ATN address.

Note.— All AFTN address Indicators are treated as explicit addresses, including predetermined address indicators (PDAIS), thus a single AFTN address can only map to a single ATN address.

3.1.3.3.3.2.3 **Recommendation**.— *The address mapping function should provide a default mapping of an AFTN Addressee Indicator to an alternate gateway ATN address when the primary gateway ATN address is not in service.*

3.1.3.3.3.3 AFTN to ATN mapping

3.1.3.3.3.3.1 Upon the reception by the Message Transfer and Control Unit of a message passed from the AFTN Component, it shall examine the AFTN Address Indicators to determine the onward routing requirements of the message over the ATN Internet.

3.1.3.3.3.3.2 Prior to delivery of the message to the ATN Component, the Message Transfer and Control Unit Component shall apply the address stripping procedures defined in Annex 10, Volume II, 4.4.8 to omit from the address any AFTN Address Indicators not related to the selected ATN address and provide for message replication if more than one ATN address is required.

Note.— In applying the procedures of 3.1.3.3.3.3.2 the Message Transfer and Control Unit Component provides sufficient copies of the message to reach each ATN address obtained by applying the procedures of 3.1.3.3.3.2.1. In most cases, a set of AFTN addresses will map to a single ATN address (the address of the corresponding ATN Gateway).

3.1.3.3.3.3.3 The Message Transfer and Control Unit shall send an appropriate service message to the AFTN originator indicator advising of an unknown address indicator according to the following:

a) the abbreviation SVC,

b) the procedure signal ADS,

c) the alignment function,

d) the indication UNKNOWN,

e) the unknown address indicator(s),

f) the end-of-text signal.

3.1.3.3.3.4 ATN to AFTN mapping

3.1.3.3.3.4.1 Upon the reception by the Message Transfer and Control Unit of a GA-Data-Indication passed from the ATN Component, the message shall be extracted from the User Data parameter.

3.1.3.3.3.4.2 The extracted message shall be passed unmodified to the AFTN Component.

3.1.3.3.3.5 Interface between the ATN Component and the Message Control Unit Component

3.1.3.3.3.5.1 The interface between the ATN Component and the Message Control Unit Component shall be according to the ATN Component service as specified in 3.1.3.3.2.2.

3.1.3.3.3.5.2 To send an AFTN message across the ATN, the Message Control Unit Component shall invoke a GA-Data-request primitive to the ATN Component.

Note.— The requirement to invoke the GA-Data-request to the ATN component is not intended to constrain an implementation. The requirement is to pass the required information to the ATN component in a manner consistent with the logical service.

3.1.3.3.3.5.3 The AFTN message, forwarded by the Message Transfer and Control Unit, shall comprise the User Data parameter.

3.1.3.3.3.5.4 The called address parameter in the GA-Data-request shall be the facility designation of the destination ATN-end system.

3.1.3.3.3.5.5 The calling address parameter in the GA-Data-request shall be the local facility designation of the AFTN/ATN Type A Gateway.

3.1.3.3.3.5.6 The AFTN priority parameter in the GA-Data-request shall be set according to the value of the AFTN priority indicator of the message.

3.1.3.3.3.5.7 Upon receipt of a D-DATA-indication primitive, the ATN Component shall invoke a GA-Data-indication to the Message Transfer and Control Unit Component.

Note.— The requirement to invoke an GA-Data-indication primitive to the Message Transfer and Control Unit Component is not intended to constrain an implementation. The requirement is to pass the required information with the Message Transfer and Control Unit Component in a manner consistent with the logical service.

3.1.3.3.3.5.8 The AFTN message, as found in the User Data parameter of the D-DATA-indication, shall comprise the User Data parameter of the GA-Data-indication.

3.1.3.3.3.5.9 The calling address parameter in the GA-Data-indication shall be the facility designation of the AFTN/ATN Type A Gateway which initiated the GA-Data-request, as found in the D-START-indication Calling Address parameter.

3.1.3.3.3.5.10 The AFTN priority parameter, if present in the GA-Data-indication, shall be derived, using Table 3.1.3-7, from the value of the QOS (priority) parameter of the corresponding D-START-indication.

3.1.3.3.3.6 Interface between the AFTN Component and the Message Control Unit Component

3.1.3.3.3.6.1 All AFTN messages or service messages passed by the AFTN Component to the Message Transfer and Control Unit shall be transferred in the order received.

3.1.3.3.3.6.2 An AFTN message or service message passed by the Message Transfer and Control Unit to the AFTN Component shall be transferred in the order received.

3.2 ATS INTERFACILITY DATA COMMUNICATIONS

3.2.1 INTRODUCTION

The AIDC application exchanges information between ATS Units (ATSUs) for support of critical Air Traffic Control (ATC) functions, such as notification of flights approaching a Flight Information Region (FIR) boundary, coordination of boundary conditions and transfer of control and communications authority.

AIDC is strictly an ATC application for exchanging tactical control information between ATS units, not with other offices or facilities.

Structure of this document

a) 3.2.1: INTRODUCTION identifies the document's structure, the functions of the AIDC application and a description of the AIDC functional model.

b) 3.2.2: GENERAL REQUIREMENTS identifies the version of the AIDC application and the Upper Layer requirements for the AIDC application.

c) 3.2.3: THE AIDC-AE ABSTRACT SERVICE describes the AIDC-AE service and the associated primitives provided to the user of the AIDC service.

d) 3.2.4: THE AIDC-ASE SERVICE describes the AIDC-ASE service and the associated primitives.

e) 3.2.5: THE AIDC CONTROL FUNCTION describes the Control Function (CF) mapping of the AIDC-user invoked primitives, the AIDC-ASE service, the ACSE service and the service provided by the communications service provider.

f) 3.2.6: THE AIDC-ASE PROTOCOL DEFINITION describes the exchanges of messages allowed by the AIDC protocol, as well as time constraints and AIDC-ASE protocol descriptions.

g) 3.2.7: FORMAL DEFINITIONS contains the ASN.1 abstract syntax for the AIDC-AE.

h) 3.2.8: COMMUNICATION REQUIREMENTS contains the requirements that the AIDC application imposes on the underlying communication system.

i) 3.2.9: AIDC USER REQUIREMENTS defines the requirements that the user of the AIDC service must meet.

j) 3.2.10: SEQUENCE DIAGRAMS AND PRIMITIVE SEQUENCING

3.2.1.1 General

3.2.1.1.1 **Recommendation.**— *AIDC is an ATN application which should be employed by two Air Traffic Service (ATS) units when exchanging Air Traffic Control (ATC) information for an active flight.*

3.2.1.2 AIDC Functional Descriptions

Flight Notification: — This function allows the Controlling ATS Unit (C-ATSU) to notify the Downstream ATS Unit (D-ATSU) of a flight's cleared profile some time before the flight enters the D-ATSU's area of interest. This function may be initiated a multiple number of times for the same flight, depending on the number and type of changes made to the flight's cleared profile.

Flight Coordination: — This function allows the C-ATSU to coordinate the conditions of transfer for a flight with a D-ATSU.

Transfer of Control: — This function allows the C-ATSU to transfer control authority for a flight to the Receiving ATS Unit (R-ATSU) and allows the R-ATSU to accept the control authority for the flight.

Transfer of Communications: — This function allows one of the following to take place:

a) the C-ATSU to transfer the control authority and communications authority for a flight to the R-ATSU and the R-ATSU to accept the control authority and communications authority for the flight; or

b) the R-ATSU to take the control authority and communications authority for a flight.

Transfer of Surveillance Data: — This function allows an ATSU1 to transfer surveillance data to an ATSU2.

General Information Interchange: — This function allows an ATSU1 to exchange general flight related data including free text messages (i.e. unstructured) with an ATSU2.

3.2.1.3 The AIDC Functional Model

Figure 3.2.1-1 shows the functional model of the AIDC Application.

The functional elements identified in this model are the following :

a) the AIDC-User,

b) the AIDC Application Entity (AIDC-AE) Service Interface,

c) the AIDC-AE,

d) the AIDC Control Function (CF),

e) the AIDC Application Service Element (AIDC-ASE) Service Interface,

f) the AIDC-ASE,

g) the Association Control Service Element, and

h) the ATN Service Provider Interface.

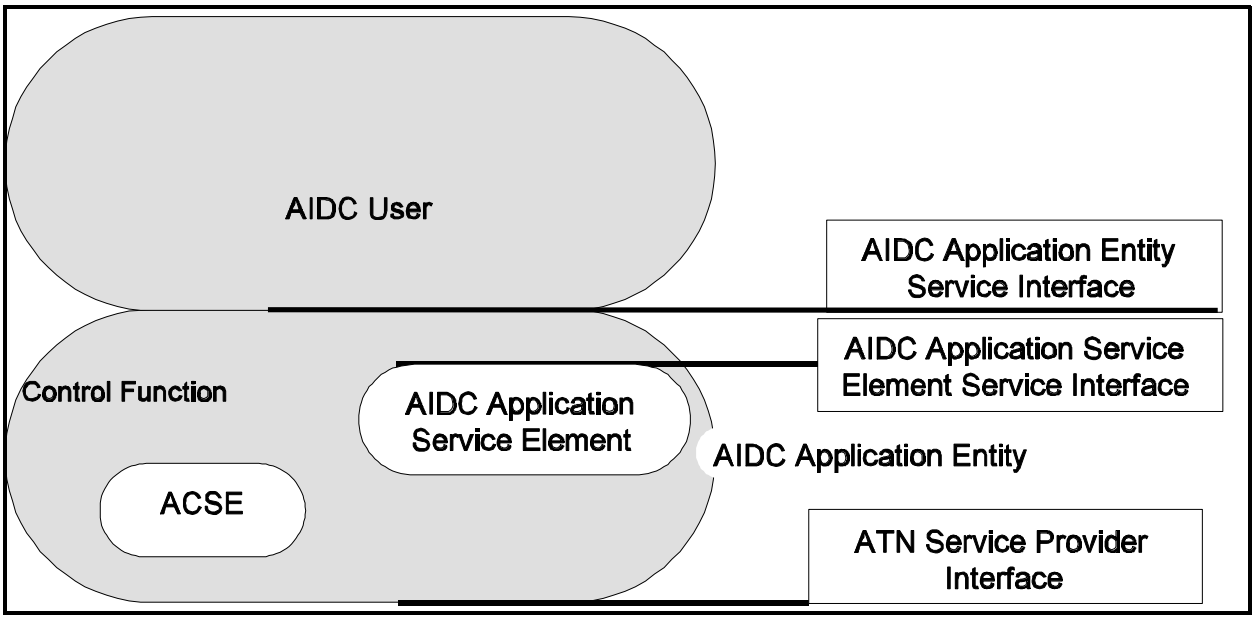

Figure 3.2.1-1: Functional Model of the AIDC Application

The AIDC-User represents the operational part of the AIDC system. This user does not perform the communication functions but relies on a communication service provided by the AIDC-AE through the AIDC-AE service interface.

The AIDC-ASE is the element in the communication system which executes the AIDC specific protocol. In other words, it enforces the AIDC specific primitive sequencing actions, timer management, and error handling.

The AIDC-AE CF is responsible for mapping primitives received from one element to another within the AIDC functional model.

3.2.1.4 Modelling Conventions

In modelling the AIDC application, each service description includes a table listing both the service primitives and the parameters of the service.

For a given primitive, the presence of each parameter is described by one of the following values in the parameter tables:

blank not applicable;

C conditional upon some predicate explained in the text;
C(=) conditional upon the value of the parameter to the left being present. In addition, the value of the parameter is equal to the value of the parameter to the left;
M mandatory;
M(=) mandatory. In addition, the value of the parameter is equal to the value of the parameter to the left;
U user option.

The following abbreviations are used in this part:

Req — request; data is input by the user initiating the service.
Ind — indication; data is indicated by the service to its respective user.
Rsp — response; data is input by the service user.
Cnf — confirmation; data is confirmed by the service to its respective user.

3.2.2 GENERAL REQUIREMENTS

3.2.2.1 AIDC-AE Version Number

3.2.2.1.1 For the first version of the AIDC application, the AIDC-AE version number shall be set to 1 (one).

3.2.2.2 Upper Layer Requirements

3.2.2.2.1 The AIDC application shall utilise the Upper Layer Architecture as defined in Sub-Volume IV.

Note.— The basis of the Upper Layer Architecture is the Application Layer, consisting of an Application Entity (AE) formed by an Application Service Element (ASE), an Association Control Service Element (ACSE) and a Control Function (CF), using the efficiency enhancements of the Presentation and Session Layers

3.2.2.2.2 The AIDC application shall use the following aspects contained in Sub-Volume IV:

a) the Naming, Addressing, Presentation Context Identification and Registration from 4.3.2;

b) the APRL for Session from 4.4;

c) the APRL for Presentation from 4.5;

d) the APRL for ACSE as specified in 4.6.

3.2.3 THE AIDC-AE ABSTRACT SERVICE

Note.— The following defines the abstract service interface used by an AIDC-User to access the AIDC-AE services.

3.2.3.1 Standard Parameters

Note.— The following service primitive parameters are defined here rather than repeated for each service definition.

3.2.3.1.1 Called ICAO Facility Designation

3.2.3.1.1.1 The *Called ICAO Facility Designation* parameter shall be provided by the AIDC-User.

3.2.3.1.1.2 The *Called ICAO Facility Designation* parameter shall be the called ATS system's ICAO four-letter location indicator or the ICAO eight-letter combined location indicator, three letters designator and an additional letter.

3.2.3.1.2 Calling ICAO Facility Designation

3.2.3.1.2.1 The *Calling ICAO Facility Designation* parameter shall be provided by the AIDC-AE.

3.2.3.1.2.2 The *Calling ICAO Facility Designation* parameter shall be the calling ATS system's ICAO four-letter location indicator or the ICAO eight-letter combined location indicator, three letters designator and an additional letter.

3.2.3.1.3 Message Number

3.2.3.1.3.1 The *Message Number* parameter shall be provided by the AIDC-User except when invoking the User-abort or User-confirmation request primitives.

3.2.3.1.3.2 The *Message Number* parameter value shall consist of a unique identifier for reference in the User-confirmation.

3.2.3.1.3.3 The *Message Number* parameter shall conform to the ASN.1 abstract syntax of *MessageNumber*.

3.2.3.2 AIDC-User Service

3.2.3.2.1 The AIDC service shall exhibit external behaviour consistent with having implemented an AIDC-AE with the following abstract services, making them available to the AIDC-User:

a) User-confirmation service as defined in 3.2.3.4.

b) Notify service as defined in 3.2.3.5.1

c) Coordinate-start service as defined in 3.2.3.6.4

d) Coordinate-end service as defined in 3.2.3.6.5

e) Coordinate-negotiate service as defined in 3.2.3.6.6

f) Coordinate-standby service as defined in 3.2.3.6.7

g) Transfer-initiate service as defined in 3.2.3.7.3

h) Transfer-request service as defined in 3.2.3.7.4

i) Transfer-conditions-proposal service as defined in 3.2.3.7.5

j) Transfer-conditions-accept service as defined in 3.2.3.7.6

k) Transfer-control service as defined in 3.2.3.7.7

l) Transfer-communication service as defined in 3.2.3.7.8

m) Transfer-communication-assume service as defined in 3.2.3.7.9

n) Info-transfer service as defined in 3.2.3.8.1

o) End service as defined in 3.2.3.9.1

p) User-abort service as defined in 3.2.3.9.2

q) Provider-abort service as defined in 3.2.3.9.3

3.2.3.3 Service Primitive Sequencing

3.2.3.3.1 The AIDC-User Service shall consist of three regimes, one asynchronous service, and a set of termination services.

3.2.3.3.2 The three regimes shall occur in a sequence:

a) Notifying regime,

b) Coordinating regime, and

c) Transferring regime.

3.2.3.3.3 The Coordinating regime shall be allowed to be re-entered at the end of the Transferring regime.

3.2.3.3.4 **Recommendation.—** *The Notifying regime should consist of the Notify service.*

3.2.3.3.5 **Recommendation.**— *The Coordinating regime should consist of the:*

a) *Coordinate-start service,*

b) *Coordinate-negotiate service,*

c) *Coordinate-standby service, and*

d) *Coordinate-end service.*

3.2.3.3.6 **Recommendation.**— *The Transferring regime should consist of the:*

a) *Transfer-initiate service,*

b) *Transfer-conditions-proposal service,*

c) *Transfer-conditions-accept service,*

d) *Transfer-request service,*

e) *Transfer-control service,*

f) *Transfer-communication service, and*

g) *Transfer-communication-assume service.*

3.2.3.4 User Confirmation Service

3.2.3.4.1 Upon the receipt of any primitive indication containing an Information parameter (e.g. Notification Information, Coordinate Start Information), the AIDC-User shall validate the value of this parameter and respond with a User-confirmation request primitive.

3.2.3.4.2 The User-confirmation service shall be an unconfirmed service.

3.2.3.4.3 Table 3.2.3-1 specifies the parameters that shall be passed when the primitives of the User-confirmation service are invoked.

Table 3.2.3-1: User-confirmation Service Primitive Parameters

Parameter Name	Req	Ind
Result	M	M(=)
Reason	U	C(=)
Referenced Number	M	M(=)

3.2.3.4.4 Result

3.2.3.4.4.1 The *Result* parameter shall be provided by the AIDC-User.

3.2.3.4.4.2 The *Result* parameter shall conform to the ASN.1 abstract syntax of *Result*.

3.2.3.4.5 Reason

3.2.3.4.5.1 **Recommendation.—** *The Reason parameter should be provided by the AIDC-User when the Response parameter has the abstract value of "rejected".*

3.2.3.4.5.2 The *Reason* parameter shall conform to the ASN.1 abstract syntax *ApplicationErrorData*.

3.2.3.4.6 Referenced Number

3.2.3.4.6.1 The *Referenced Number* parameter shall be provided by the AIDC-User.

3.2.3.4.6.2 The *Referenced Number* parameter shall contain the Message Number of the message being confirmed.

3.2.3.4.6.3 The *Referenced Number* parameter shall conform to the ASN.1 abstract syntax of *MessageNumber*.

3.2.3.5 Notifying Regime

3.2.3.5.1 Notify Service

Note.— The purpose of the Notify service is to allow a C-ATSU to update the information a D-ATSU maintains on a flight that is expected to enter its area of interest at some future time.

3.2.3.5.1.1 Service Primitives

3.2.3.5.1.1.1 The Notify service shall be an unconfirmed service.

3.2.3.5.1.1.2 Table 3.2.3-2 specifies the parameters that shall be passed when the primitives of the Notify service are invoked.

Table 3.2.3-2: Notify Service Primitives Parameters

Parameter Name	Req	Ind
Called ICAO Facility Designation	M	
Calling ICAO Facility Designation		M
Notification Information	M	M(=)
Message Number	M	M(=)

3.2.3.5.1.1.3 Notification Information

3.2.3.5.1.1.3.1 The Notification Information parameter shall be provided by the AIDC-User when invoking the Notify request primitive.

3.2.3.5.1.1.3.2 The Notification Information parameter shall conform to the ASN.1 abstract syntax *Notify*.

3.2.3.6 Coordinating Regime

3.2.3.6.1 The Coordinating regime shall begin with the invocation of a Coordinate-start primitive.

3.2.3.6.2 The Coordinating regime shall end with the invocation of a Coordinate-end primitive.

3.2.3.6.3 Upon entering the Coordinating regime, further use of the Notify service shall be prohibited.

3.2.3.6.4 Coordinate-start Service

Note 1.— The purpose of the Coordinate-start service is to allow an ATSU1 to begin the coordination of, or update, the conditions of transfer of a flight with an ATSU2.

Note 2.— If rejected, any currently agreed coordination conditions remain in affect.

3.2.3.6.4.1 Service Primitives

3.2.3.6.4.1.1 The Coordinate-start service shall be an unconfirmed service.

3.2.3.6.4.1.2 Table 3.2.3-3 specifies the parameters that shall be passed when the primitives of the Coordinate-start service are invoked.

Table 3.2.3-3: Coordinate-start Service primitive Parameters

Parameter Name	Req	Ind
Called ICAO Facility Designation	M	

Parameter Name	Req	Ind
Calling ICAO Facility Designation		M
Coordinate Start Information	M	M(=)
Message Number	M	M(=)

3.2.3.6.4.1.3 Coordinate Start Information

3.2.3.6.4.1.3.1 The *Coordinate Start Information* parameter shall be provided by the AIDC-User.

3.2.3.6.4.1.3.2 The *Coordinate Start Information* parameter shall conform to the ASN.1 abstract syntax of *CoordinateInitial* or *CoordinateUpdate*.

3.2.3.6.5 Coordinate-end Service

Note 1.— The purpose of the Coordinate-end service is to allow an ATSU1 to notify an ATSU2 that the ATSU1 either accepts or rejects the coordination conditions proposed.

Note 2.— The successful completion of this service ends the coordinating regime for a flight.

3.2.3.6.5.1 Service Primitives

3.2.3.6.5.1.1 The Coordinate-end service shall be an unconfirmed service.

3.2.3.6.5.1.2 The Coordinate-end service primitives shall only be invoked while in the Coordinating regime.

3.2.3.6.5.1.3 Table 3.2.3-4 specifies the parameters that shall be passed when the primitives of the Coordinate-end service are invoked.

Table 3.2.3-4: Coordinate-end Service Primitive Parameters

Parameter Name	Req	Ind
Coordinate End Information	M	M(=)
Result	M	
Message Number	M	M(=)

3.2.3.6.5.1.4 Coordinate End Information

3.2.3.6.5.1.4.1 The *Coordinate End Information* parameter shall be provided by the AIDC-User.

3.2.3.6.5.1.4.2 The *Coordinate End Information* parameter shall conform to the ASN.1 abstract syntax of *CoordinateAccept* or *CoordinateReject*.

3.2.3.6.5.1.5 Result

3.2.3.6.5.1.5.1 The *Result* parameter shall be provided by the AIDC-User.

3.2.3.6.5.1.5.2 The *Result* parameter shall conform to the ASN.1 abstract syntax of *Result*.

3.2.3.6.6 Coordinate-negotiate Service

Note 1.— The purpose of the Coordinate-negotiate service is to allow an ATSU1 to negotiate modifications to a flight's existing coordination conditions with an ATSU2.

Note 2.— If rejected, any currently agreed coordination conditions remain in affect.

3.2.3.6.6.1 Service Primitives

3.2.3.6.6.1.1 The Coordinate-negotiate service shall be an unconfirmed service.

3.2.3.6.6.1.2 The Coordinate-negotiate service primitives shall only be invoked while in the Coordinating regime.

3.2.3.6.6.1.3 Table 3.2.3-5 specifies the parameters that shall be passed when the primitives of the Coordinate-negotiate service are invoked.

Table 3.2.3-5: Coordinate-negotiate Service Primitive Parameters

Parameter Name	Req	Ind
Coordinate Negotiate Information	M	M(=)
Message Number	M	M(=)

3.2.3.6.6.1.4 Coordinate Negotiate Information

3.2.3.6.6.1.4.1 The *Coordinate Negotiate Information* parameter shall be provided by the AIDC-User.

3.2.3.6.6.1.4.2 The *Coordinate Negotiate Information* parameter shall conform to the ASN.1 abstract syntax *CoordinateNegotiate*.

3.2.3.6.7 Coordinate-standby Service

Note 1.— The purpose of the Coordinate-standby service is to allow an ATSU1 to notify an ATSU2 that the coordination dialogue between the ATSUs is being temporarily suspended.

Note 2.— Each invocation of this service effectively extends, by a defined amount, the time before a user response to a coordinate service indication is given.

3.2.3.6.7.1 Service Primitives

3.2.3.6.7.1.1 The Coordinate-standby service shall be an unconfirmed service.

3.2.3.6.7.1.2 The Coordinate-standby service primitives shall only be invoked while in the Coordinating regime.

3.2.3.6.7.1.3 Table 3.2.3-6 specifies the parameters that shall be passed when the primitives of the Coordinate-standby service are invoked.

Table 3.2.3-6: Coordinate-standby Service Primitive Parameters

Parameter Name	Req	Ind
Coordinate Standby Information	M	M(=)
Message Number	M	M(=)

3.2.3.6.7.1.4 Coordinate Standby Information

3.2.3.6.7.1.4.1 The *Coordinate Standby Information* parameter shall be provided by the AIDC-User.

3.2.3.6.7.1.4.2 The *Coordinate Standby Information* parameter shall conform to the ASN.1 abstract syntax *CoordinateStandby*.

3.2.3.7 Transferring regime

3.2.3.7.1 The Transferring regime shall be entered after the completion of the Coordinating regime.

3.2.3.7.2 The Transferring regime shall begin with the invocation of any of the following:

a) the Transfer-initiate service;

b) the Transfer-request service; or

c) the Transfer-control service.

3.2.3.7.3 Transfer-initiate Service

Note. — The purpose of the Transfer-initiate service is to allow a C-ATSU to initiate the transfer phase for a flight by passing executive control information to an R-ATSU.

3.2.3.7.3.1 Service Primitives

3.2.3.7.3.1.1 The Transfer-initiate service shall be an unconfirmed service.

3.2.3.7.3.1.2 Table 3.2.3-7 specifies the parameters that shall be passed when the primitives of the Transfer-initiate service are invoked.

Table 3.2.3-7: Transfer-initiate Service Primitive Parameters

Parameter Name	Req	Ind
Transfer Initiate Information	M	M(=)
Message Number	M	M(=)

3.2.3.7.3.1.3 Transfer Initiate Information

3.2.3.7.3.1.3.1 The *Transfer Initiate Information* parameter shall be provided by the AIDC-User.

3.2.3.7.3.1.3.2 The *Transfer Initiate Information* parameter shall conform to the ASN.1 abstract syntax *TransferInitiate*.

3.2.3.7.4 Transfer-request Service

Note.) The purpose of the Transfer-request service is to allow an R-ATSU to request control and communications authority for a flight from a C-ATSU.

3.2.3.7.4.1 Service Primitives

3.2.3.7.4.1.1 The Transfer-request service shall be an unconfirmed service.

3.2.3.7.4.1.2 Table 3.2.3-8 specifies the parameters that shall be passed when the primitives of the Transfer-request service are invoked.

Table 3.2.3-8: Transfer-request Service Primitive Parameters

Parameter Name	Req	Ind
Transfer Request Information	M	M(=)
Message Number	M	M(=)

3.2.3.7.4.1.3 Transfer Request Information

3.2.3.7.4.1.3.1 The *Transfer Request Information* parameter shall be provided by the AIDC-User.

3.2.3.7.4.1.3.2 The *Transfer Request Information* parameter shall conform to the ASN.1 abstract syntax *TransferRequest*.

3.2.3.7.5 Transfer-conditions-proposal Service

Note.— The purpose of the Transfer-conditions-proposal service is to allow a C-ATSU to propose the conditions under which control authority for a flight can be given to an R-ATSU.

3.2.3.7.5.1 Service Primitives

3.2.3.7.5.1.1 The Transfer-conditions-proposal service shall be an unconfirmed service.

3.2.3.7.5.1.2 The Transfer-conditions-proposal service primitives shall only be invoked after the successful completion of the Transfer-initiate service.

3.2.3.7.5.1.3 Table 3.2.3-9 specifies the parameters that shall be passed when the primitives of the Transfer-conditions-proposal service are invoked.

Table 3.2.3-9: Transfer-conditions-proposal Service Primitive Parameters

Parameter Name	Req	Ind
Transfer Conditions Proposal Information	M	M(=)
Message Number	M	M(=)

3.2.3.7.5.1.4 Transfer Conditions Proposal Information

3.2.3.7.5.1.4.1 The *Transfer Conditions Proposal Information* parameter is provided by the AIDC-User.

3.2.3.7.5.1.4.2 The *Transfer Conditions Proposal Information* parameter shall conform to the ASN.1 abstract syntax *TransferConditionsProposal*.

3.2.3.7.6 Transfer-conditions-accept Service

Note 1.— The purpose of the Transfer-conditions-accept service is to allow an R-ATSU to indicate that it is willing to accept control conditions proposed for a flight by the C-ATSU.

Note 2.— This service, when used, is only used in response to the Transfer-conditions-proposal service.

3.2.3.7.6.1 Service Primitives

3.2.3.7.6.1.1 The Transfer-conditions-accept service shall be an unconfirmed service.

3.2.3.7.6.1.2 The Transfer-conditions-accept service primitives shall only be invoked after the successful completion of the Transfer-conditions-proposal service.

3.2.3.7.6.1.3 Table 3.2.3-10 specifies the parameters that shall be passed when the primitives of the Transfer-conditions-accept service are invoked.

Table 3.2.3-10: Transfer-conditions-accept Service Primitive Parameters

Parameter Name	Req	Ind
Transfer Conditions Accept Information	M	M(=)
Message Number	M	M(=)

3.2.3.7.6.1.4 Transfer Conditions Accept Information

3.2.3.7.6.1.4.1 The *Transfer Conditions Accept Information* parameter shall be provided by the AIDC-User.

3.2.3.7.6.1.4.2 The *Transfer Conditions Accept Information* parameter shall conform to the ASN.1 abstract syntax *TransferConditionsAccept*.

3.2.3.7.7 Transfer-control Service

Note.— The purpose of the Transfer-control service is to allow a C-ATSU to indicate that it wants to relinquish control authority for a flight to an R-ATSU.

3.2.3.7.7.1 Service Primitives

3.2.3.7.7.1.1 The Transfer-control service shall be a confirmed service.

3.2.3.7.7.1.2 Table 3.2.3-11 specifies the parameters that shall be passed when the primitives of the Transfer-control service are invoked.

Table 3.2.3-11: Transfer-control Service Primitive Parameters

Parameter Name	Req	Ind	Rsp	Cnf
Transfer Control Information	M	M(=)	M	M(=)
Result			M	M(=)
Message Number	M	M(=)	M	M(=)

3.2.3.7.7.1.3 Transfer Control Information

3.2.3.7.7.1.3.1 The *Transfer Control Information* parameter shall be provided by the C-ATSU AIDC-User when invoking the Transfer-control request primitive.

3.2.3.7.7.1.3.2 The *Transfer Control Information* parameter shall conform to the ASN.1 abstract syntax *TransferControl*, when the C-ATSU AIDC-User invokes the Transfer-control request primitive.

3.2.3.7.7.1.3.3 The *Transfer Control Information* parameter shall be provided by the R-ATSU AIDC-User when invoking the Transfer-control response service primitive.

3.2.3.7.7.1.3.4 The *Transfer Control Information* parameter shall conform to the ASN.1 abstract syntax *TransferControlData*, when the R-ATSU AIDC-User invokes the Transfer-control response primitive.

3.2.3.7.7.1.4 Result

3.2.3.7.7.1.4.1 The *Result* parameter shall be provided by the C-ATSU AIDC-User.

3.2.3.7.7.1.4.2 The *Result* parameter shall conform to the ASN.1 abstract syntax *Result*.

3.2.3.7.7.1.4.3 When the *Result* parameter value is set to "accept", the Transfer Control Information parameter shall conform to the ASN.1 abstract syntax *TransferControlAssume*.

3.2.3.7.7.1.4.4 When the *Result* parameter value is set to "reject", the Transfer Control Information parameter shall conform to the ASN.1 abstract syntax *TransferControlReject*.

3.2.3.7.8 Transfer-communication Service

Note.— The purpose of the Transfer-communication service is to allow a C-ATSU to indicate that it, is relinquishing communication authority for a flight to an R-ATSU.

3.2.3.7.8.1 Service Primitives

3.2.3.7.8.1.1 The Transfer-communication service shall be an unconfirmed service.

3.2.3.7.8.1.2 The Transfer-communication service primitives shall only be invoked after the successful completion of the Transfer-initiate service.

3.2.3.7.8.1.3 Table 3.2.3-12 specifies the parameters that shall be passed when the primitives of the Transfer-communication service are invoked.

Table 3.2.3-12: Transfer-communication Service Primitive Parameters

Parameter Name	Req	Ind
Transfer Communication Information	M	M(=)
Message Number	M	M(=)

3.2.3.7.8.1.4 Transfer Communication Assume Information

3.2.3.7.8.1.4.1 The *Transfer Communication Information* parameter shall be provided by the AIDC-User.

3.2.3.7.8.1.4.2 The *Transfer Communication Information* parameter shall conform to the ASN.1 abstract syntax *TransferCommunication*.

3.2.3.7.9 Transfer-communication-assume Service

Note.— The purpose of the Transfer-communication-assume service is to allow an R-ATSU to indicate to a C-ATSU that communication with a flight has been established.

3.2.3.7.9.1 Service Primitives

3.2.3.7.9.1.1 The Transfer-communication-assume service shall be an unconfirmed service.

3.2.3.7.9.1.2 The Transfer-communication-assume service primitives shall only be invoked after the successful completion of the Transfer-initiate service.

3.2.3.7.9.1.3 Table 3.2.3-13 specifies the parameters that shall be passed when the primitives of the Transfer-communication-assume service are invoked.

Table 3.2.3-13: Transfer-communication-assume Service Primitive Parameters

Parameter Name	Req	Ind
Transfer Communication Assume Information	M	M(=)
Message Number	M	M(=)

3.2.3.7.9.1.4 Transfer Communication Assume Information

3.2.3.7.9.1.4.1 The *Transfer Communication Assume Information* parameter shall be provided by the AIDC-User.

3.2.3.7.9.1.4.2 The *Transfer Communication Assume Information* parameter shall conform to the ASN.1 abstract syntax *TransferCommunicationAssume*.

3.2.3.8 Asynchronous Services

3.2.3.8.1 Info-transfer Service

Note 1.— The Info-transfer service permits an ATSU1 to transmit general executive data, surveillance data, general free text data, or emergency free text data, or to point-out a flight to an ATSU2.

Note 2.— The Info-transfer service may be invoked by the C-ATSU even when no regime has been established.

3.2.3.8.1.1 Service Primitives

3.2.3.8.1.1.1 It shall be possible, for any ATSU to invoke the primitives of the Info-transfer service at any time after the initial invocation of primitives of the Notify service or after the initial invocation of the Coordinate-start service primitives when the Notify service is not used.

3.2.3.8.1.1.2 It shall be possible for the C-ATSU only, to invoke the primitives of the Info-transfer service before invoking any other service primitives.

3.2.3.8.1.2 The Info-transfer service shall be an unconfirmed service.

3.2.3.8.1.3 Table 3.2.3-14 specifies the parameters that shall be passed when the primitives of the Info-transfer service are invoked.

Table 3.2.3-14: Info-transfer Service Primitive Parameters

Parameter Name	Req	Ind
Called ICAO Facility Designation	C	
Calling ICAO Facility Designation		C
Information	M	M(=)
Message Number	M	M(=)

3.2.3.8.1.4 Called ICAO Facility Designation

3.2.3.8.1.4.1 The *Called ICAO Facility Designation* parameter shall be supplied, as specified in 3.2.3.1.1, only when the Info-transfer request primitive is invoked and no regime has yet been established by the AIDC-user.

3.2.3.8.1.5 Calling ICAO Facility Designation

3.2.3.8.1.5.1 The *Calling ICAO Facility Designation* parameter shall be supplied, as specified in 3.2.3.1.2, only when the Info-transfer indication primitive corresponds to a request primitive in which the Called ICAO Facility Designation was supplied.

3.2.3.8.1.6 Information

3.2.3.8.1.6.1 The *Information* parameter shall be provided by the AIDC-User.

3.2.3.8.1.6.2 The *Information* parameter shall conform to the ASN.1 abstract syntax of any of the following:

a) *GeneralExecutiveData,*

b) *GeneralPoint,*

c) *SurveillanceData,*

d) *GeneralFreeText,* or

e) *EmergencyFreeText.*

3.2.3.9 Termination Services

3.2.3.9.1 End Service

Note.) The purpose of the End service is to allow an ATSU1 to indicate to an ATSU2 that it is in the process of:

a) cancelling the notification for a flight and terminating the AIDC service; or

b) cancelling the coordination for a flight and terminating the AIDC service; or

c) *terminating the AIDC service.*

3.2.3.9.1.1 Service Primitives

3.2.3.9.1.1.1 The End service shall be an unconfirmed service.

3.2.3.9.1.1.2 Table 3.2.3-15 specifies the parameters that shall be passed when the primitives of the End service are invoked.

Table 3.2.3-15: End Service primitive Parameters

Parameter Name	Req	Ind
Cancel Information	C	C(=)
Message Number	M	M(=)

3.2.3.9.1.1.3 Cancel Information

3.2.3.9.1.1.3.1 The *Cancel Information* parameter shall be provided by the AIDC-User when cancelling the notification or coordination for a flight.

3.2.3.9.1.1.3.2 The *Cancel Information* parameter shall conform to the ASN.1 abstract syntax of *Cancel*.

3.2.3.9.2 User-abort Service

Note 1.— The purpose of the User-abort service is to allow an AIDC-User to immediately terminate the AIDC service.

Note 2.— The User-abort service may be used for both operational and technical reasons.

3.2.3.9.2.1 Service Primitives

3.2.3.9.2.1.1 The User-abort service shall be an unconfirmed service.

3.2.3.9.2.1.2 It shall be possible to invoke the primitives of the User-abort service at any time.

3.2.3.9.2.1.3 The User-abort service primitives shall have no parameters.

3.2.3.9.3 Provider-abort Service

Note.— The purpose of the Provider-abort service is to provide the capability for the AIDC-service provider to inform the AIDC-User that it can no longer provide the AIDC service.

3.2.3.9.3.1 Service Primitives

3.2.3.9.3.1.1 The Provider-abort service shall be an unconfirmed service.

3.2.3.9.3.1.2 The primitives of the Provider-abort service shall be invoked by the AIDC-AE service-provider.

3.2.3.9.3.1.3 Table 3.2.3-16 specifies the parameter that shall be passed when the primitive of the Provider -abort service is invoked.

Table 3.2.3-16: Provider-abort Service primitive Parameters

Parameter Name	Ind
Provider Abort Reason	M
Result Source	C

3.2.3.9.3.1.4 Provider Abort Reason

3.2.3.9.3.1.4.1 The *Provider Abort Reason* parameter shall be provided by the AIDC-AE service provider.

3.2.3.9.3.1.4.2 The *Provider Abort Reason* parameter shall conform to the ASN.1 abstract syntax *ProviderAbortReason.*

3.2.3.9.3.1.5 Result Source

3.2.3.9.3.1.5.1 The *Result Source* parameter shall optionally be provided by the AIDC-AE service provider, only when the *Provider Abort Reason* parameter has one of the abstract values “rejectedpermanent” or “rejectedtransient”.

3.2.3.9.3.1.5.2 The *Result Source* parameter shall conform to the abstract syntax of the A-ASSOCIATE *Result Source* parameter, as defined in ISO/IEC 8649.

3.2.4 THE AIDC-ASE ABSTRACT SERVICE

Note.— The following defines the abstract service interface used by an AIDC-ASE user to access the AIDC-ASE services and the services assumed to be supporting the AIDC-ASE.

3.2.4.1 Standard Parameters

Note.— The following service primitive parameters are defined here rather than repeated.

3.2.4.1.1 User Data

3.2.4.1.1.1 The *User Data* parameter, if provided, shall contain data provided by the user of the AIDC-ASE service.

Note.— The User Data parameter conforms to one of the following ASN.1 abstract syntaxes:

a) *Notify,*

b) CoordinateInitial,

c) CoordinateUpdate,

d) CoordinateAccept,

e) CoordinateReject,

f) CoordinateNegotiate,

g) CoordinateStandby,

h) TransferInitiate,

i) TransferRequest,

j) *TransferConditionsProposal,*

k) TransferConditionsAccept,

l) TransferControl,

m) TransferControlAssume,

n) *TransferControlReject,*

o) TransferCommunication,

p) *TransferCommunicationAssume,*

q) *GeneralExecutiveData,*

r) *GeneralPoint,*

s) *SurveillanceData,*

t) *GeneralFreeText,*

u) *EmergencyFreeText,*

v) *Cancel.*

3.2.4.1.2 Msg Number

3.2.4.1.2.1 The *Msg Number* parameter shall be provided by the AIDC-ASE user except when invoking the AIDC-usr-abort or AIDC-ucf request primitives.

3.2.4.1.2.2 The *Msg Number* parameter shall consist of a unique identifier for reference in the AIDC-ucf primitives.

Note.— The Msg Number parameter conforms to the ASN.1 abstract syntax MessageNumber.

3.2.4.2 AIDC-ASE Services

3.2.4.2.1 List of AIDC-ASE Services

3.2.4.2.1.1 An implementation of the AIDC-AE shall exhibit behaviour consistent with having implemented an AIDC-ASE with the following abstract services:

a) AIDC-ucf service as defined in 3.2.4.2.2

b) AIDC-nfy service as defined in 3.2.4.2.3

c) AIDC-crd-start service as defined in 3.2.4.2.4

d) AIDC-crd-end service as defined in 3.2.4.2.5

e) AIDC-crd-ngtt service as defined in 3.2.4.2.6

f) AIDC-crd-stndby service as defined in 3.2.4.2.7

g) AIDC-tfr-init service as defined in 3.2.4.2.8

h) AIDC-tfr-rqst service as defined in 3.2.4.2.9

i) AIDC-tfr-prpsl service as defined in 3.2.4.2.10

j) AIDC-tfr-accept service as defined in 3.2.4.2.11

k) AIDC-tfr-cntrl service as defined in 3.2.4.2.12

l) AIDC-tfr-comm service as defined in 3.2.4.2.13

m) AIDC-tfr-comm-assm service as defined in 3.2.4.2.14

n) AIDC-inf-tfr service as defined in 3.2.4.2.15

o) AIDC-end service as defined in 3.2.4.2.16

p) AIDC-usr-abrt service as defined in 3.2.4.2.17

q) AIDC-pvd-abrt service as defined in 3.2.4.2.18.

3.2.4.2.2 AIDC-ucf Service

3.2.4.2.2.1 Service Primitives

3.2.4.2.2.1.1 The AIDC-ucf service shall be an unconfirmed service.

3.2.4.2.2.1.2 Table 3.2.4-1 specifies the parameters that shall be passed when the primitives of the AIDC-ucf service are invoked.

Table 3.2.4-1: AIDC-ucf Service Primitive Parameters

Parameter Name	Req	Ind
Result	M	M(=)
Reason	U	C(=)
Reference ID	M	M(=)

3.2.4.2.2.1.3 Result

3.2.4.2.2.1.3.1 The *Result* parameter shall be provided by the AIDC-ASE user.

3.2.4.2.2.1.3.2 The *Result* parameter shall indicate the acceptance or rejection of the service primitive.

Note.— The Result parameter conforms to the ASN.1 abstract syntax Result.

3.2.4.2.2.1.4 Reason

3.2.4.2.2.1.4.1 The *Reason* parameter shall be provided by the AIDC-ASE user.

Note.— The Reason parameter conforms to the ASN.1 abstract syntax ApplicationErrorData.

3.2.4.2.2.1.5 Reference ID

3.2.4.2.2.1.5.1 The *Reference ID* parameter shall be provided by the AIDC-ASE user.

3.2.4.2.2.1.5.2 The *Reference ID* parameter shall contain the Msg Number of the service that is being accepted or rejected.

Note.— The Reference ID parameter conforms to the ASN.1 abstract syntax MessageNumber.

3.2.4.2.3 AIDC-nfy Service

3.2.4.2.3.1 Service Primitives

3.2.4.2.3.1.1 The AIDC-nfy service shall be an unconfirmed service.

3.2.4.2.3.1.2 Table 3.2.4-2 specifies the parameters that shall be passed when the primitives of the AIDC-nfy service are invoked.

Table 3.2.4-2: AIDC-nfy Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Msg Number	M	M(=)

3.2.4.2.4 AIDC-crd-start Service.

3.2.4.2.4.1 Service Primitives

3.2.4.2.4.1.1 The AIDC-crd-start service shall be an unconfirmed service.

3.2.4.2.4.1.2 Table 3.2.4-3 specifies the parameters that shall be passed when the primitives of the AIDC-crd-start service are invoked.

Table 3.2.4-3: AIDC-crd-start Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Msg Number	M	M(=)

3.2.4.2.5 AIDC-crd-end Service

3.2.4.2.5.1 Service Primitives

3.2.4.2.5.1.1 The AIDC-crd-end service shall be an unconfirmed service.

3.2.4.2.5.1.2 Table 3.2.4-4 specifies the parameters that shall be passed when the primitives of the AIDC-crd-end service are invoked.

Table 3.2.4-4: AIDC-crd-end Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Result	M	M(=)
Msg Number	M	M(=)

3.2.4.2.5.1.3 Result

3.2.4.2.5.1.3.1 The *Result* parameter shall be provided by the AIDC-ASE user.

3.2.4.2.5.1.3.2 The *Result* parameter shall be used to indicate acceptance or rejection of the ending of the Coordinating Regime.

Note.— The Result parameter conforms to the ASN.1 abstract syntax Result.

3.2.4.2.6 AIDC-crd-ngtt Service

3.2.4.2.6.1 Service Primitives

3.2.4.2.6.1.1 The AIDC-crd-ngtt service shall be an unconfirmed service.

3.2.4.2.6.1.2 Table 3.2.4-5 specifies the parameters that shall be passed when the primitives of the AIDC-crd-ngtt service are invoked.

Table 3.2.4-5: AIDC-crd-ngtt Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Msg Number		

3.2.4.2.7 AIDC-crd-stndby Service

3.2.4.2.7.1 Service Primitives

3.2.4.2.7.1.1 The AIDC-crd-stndby service shall be an unconfirmed service.

3.2.4.2.7.1.2 Table 3.2.4-6 specifies the parameters that shall be passed when the primitives of the AIDC-crd-stndby service are invoked.

Table 3.2.4-6: AIDC-crd-stndby Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Msg Number	M	M(=)

3.2.4.2.8 AIDC-tfr-init Service

3.2.4.2.8.1 Service Primitives

3.2.4.2.8.1.1 The AIDC-tfr-init service shall be an unconfirmed service.

3.2.4.2.8.1.2 Table 3.2.4-7 specifies the parameters that shall be passed when the primitives of the AIDC-tfr-init service are invoked.

Table 3.2.4-7: AIDC-tfr-init Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Msg Number	M	M(=)

3.2.4.2.9 AIDC-tfr-rqst Service

3.2.4.2.9.1 Service Primitives

3.2.4.2.9.1.1 The AIDC-tfr-rqst service shall be an unconfirmed service.

3.2.4.2.9.1.2 Table 3.2.4-8 specifies the parameters that shall be passed when the primitives of the AIDC-tfr-rqst service are invoked.

Table 3.2.4-8: AIDC-tfr-rqst Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Msg Number	M	M(=)

3.2.4.2.10 AIDC-tfr-prpsl Service

3.2.4.2.10.1 Service Primitives

3.2.4.2.10.1.1 The AIDC-tfr-prpsl service shall be an unconfirmed service.

3.2.4.2.10.1.2 Table 3.2.4-9 specifies the parameters that shall be passed when the primitives of the AIDC-tfr-prpsl service are invoked.

Table 3.2.4-9: AIDC-tfr-prpsl Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Msg Number	M	M(=)

3.2.4.2.11 AIDC-tfr-accept Service

3.2.4.2.11.1 Service Primitives

3.2.4.2.11.1.1 The AIDC-tfr-accept service shall be an unconfirmed service.

3.2.4.2.11.1.2 Table 3.2.4-10 specifies the parameters that shall be passed when the primitives of the AIDC-tfr-accept service are invoked.

Table 3.2.4-10: AIDC-tfr-accept Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Msg Number	M	M(=)

3.2.4.2.12 AIDC-tfr-cntrl Service

3.2.4.2.12.1 Service Primitives

3.2.4.2.12.1.1 The AIDC-tfr-cntrl service shall be a confirmed service.

3.2.4.2.12.1.2 Table 3.2.4-11 below specifies the parameters that shall be passed when the primitives of the AIDC-tfr-cntrl service are invoked.

Table 3.2.4-11: AIDC-tfr-cntrl Service Primitive Parameters

Parameter Name	Req	Ind	Rsp	Cnf
User Data	U	C(=)	U	C(=)
Result			M	M(=)
Msg Number	M	M(=)	M	M(=)

3.2.4.2.12.1.3 Result

3.2.4.2.12.1.3.1 The *Result* parameter, shall be provided by the AIDC-ASE user.

Note.— The Result parameter conforms to the ASN.1 abstract syntax Result.

3.2.4.2.13 AIDC-tfr-comm Service

3.2.4.2.13.1 Service Primitives

3.2.4.2.13.1.1 The AIDC-tfr-comm service shall be an unconfirmed service.

3.2.4.2.13.1.2 Table 3.2.4-12 specifies the parameters that shall be passed when the primitives of the AIDC-tfr-comm service are invoked.

Table 3.2.4-12: AIDC-tfr-comm Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Msg Number	M	M(=)

3.2.4.2.14 AIDC-tfr-comm-assm Service

3.2.4.2.14.1 Service Primitives

3.2.4.2.14.1.1 The AIDC-tfr-comm-assm service shall be an unconfirmed service.

3.2.4.2.14.1.2 Table 3.2.4-13 specifies the parameters that shall be passed when the primitives of the AIDC-tfr-comm-assm service are invoked.

Table 3.2.4-13: AIDC-tfr-comm-assm Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Msg Number	M	M(=)

3.2.4.2.15 AIDC-inf-tfr Service

3.2.4.2.15.1 Service Primitives

3.2.4.2.15.1.1 The AIDC-inf-tfr service shall be an unconfirmed service.

3.2.4.2.15.1.2 Table 3.2.4-14 specifies the parameters that shall be passed when the primitives of the AIDC-inf-tfr service are invoked.

Table 3.2.4-14: AIDC-inf-tfr Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Msg Number	M	M(=)

3.2.4.2.16 AIDC-end Service

3.2.4.2.16.1 Service Primitives

3.2.4.2.16.1.1 The AIDC-end service shall be an unconfirmed service.

3.2.4.2.16.1.2 Table 3.2.4-15 specifies the parameters that shall be passed when the primitives of the AIDC-end service are invoked.

Table 3.2.4-15: AIDC-end Service Primitive Parameters

Parameter Name	Req	Ind
User Data	U	C(=)
Msg Number	M	M(=)

3.2.4.2.17 AIDC-usr-abrt Service

3.2.4.2.17.1 Service Primitives

3.2.4.2.17.1.1 The AIDC-usr-abrt service shall be an unconfirmed service.

3.2.4.2.17.1.2 The AIDC-usr-abrt service primitives shall have no parameters.

3.2.4.2.18 AIDC-pvd-abrt Service

3.2.4.2.18.1 The AIDC-pvd-abrt service shall be an AIDC-ASE service-provider initiated service.

3.2.4.2.18.2 Service Primitive

3.2.4.2.18.2.1 The AIDC-pvd-abrt service shall be an unconfirmed service.

3.2.4.2.18.2.2 Table 3.2.4-16 specifies the parameter that shall be passed when the primitive of the AIDC-pvd-abrt service is invoked.

Table 3.2.4-16: AIDC-pvd-abrt Service Primitive Parameter

Parameter Name	Ind
Abort Reason	M

3.2.4.2.18.2.3 AbortReason

3.2.4.2.18.2.3.1 The *Abort Reason* parameter shall be provided by the AIDC-ASE.

3.2.4.2.18.2.3.2 The *Abort Reason* parameter shall be used to identify the reason for the abort.

Note.— The Abort Reason parameter conforms to the ASN.1 abstract syntax ProviderAbortReason.

3.2.4.3 Services Supporting the AIDC-ASE

Note.— The AIDC-ASE may be incorporated in any application entity that provides the following services.

3.2.4.3.1 List of Supporting Services

3.2.4.3.1.1 An implementation of the AIDC-AE shall exhibit the behaviour consistent with having implemented an AIDC-ASE supported by the following abstract services:

a) AIDC-DATA service as defined in 3.2.4.3.2

b) AIDC-ABORT service as defined in 3.2.4.3.3

c) AIDC-P-ABORT service as defined in 3.2.4.3.4

3.2.4.3.2 AIDC-DATA Service

3.2.4.3.2.1 Service Primitives

3.2.4.3.2.1.1 The AIDC-DATA service shall be an unconfirmed service.

3.2.4.3.2.1.2 Table 3.2.4-17 specifies the parameter that shall be passed when the primitives of the AIDC-DATA service are invoked.

Table 3.2.4-17: AIDC-DATA Service Primitive Parameters

Parameter Name	Req	Ind
AIDC Data	U	C(=)

3.2.4.3.2.1.3 AIDC Data

3.2.4.3.2.1.3.1 The *AIDC Data* parameter, if any, shall be provided by the AIDC-ASE.
3.2.4.3.2.1.3.2 The *AIDC Data* parameter shall conform to the ASN.1 syntax *AIDC-APDU*.

3.2.4.3.3 AIDC-ABORT Service

3.2.4.3.3.1 Service Primitives

3.2.4.3.3.1.1 The AIDC-ABORT service shall be an unconfirmed service.

3.2.4.3.3.1.2 Table 3.2.4-18 specifies the parameter that shall be passed when the primitives of the AIDC-ABORT service primitives are invoked.

Table 3.1.4-18: AIDC-ABORT Service Primitive Parameters

Parameter Name	Req	Ind
Abort Reason	U	C(=)

3.2.4.3.3.2 Abort Reason

3.2.4.3.3.2.1 The *Abort Reason* parameter, if any, shall be provided by the AIDC-ASE.

3.2.4.3.3.2.2 The *Abort Reason* parameter shall conform to the ASN.1 syntax of *ProviderAbortReason*.

3.2.4.3.4 AIDC-P-ABORT Service

3.2.4.3.4.1 Service Primitives

3.2.4.3.4.1.1 The AIDC-P-ABORT service shall be an unconfirmed service.

3.2.4.3.4.1.2 The AIDC-P-ABORT service primitives shall have no parameters.

3.2.5 THE AIDC CONTROL FUNCTION

3.2.5.1 **Recommendation. —** *An implementation of the AIDC-AE should behave as if there existed a Control Function as defined below.*

Note.— The following specifies the AIDC Control Function (CF) in terms of state definitions, and service mappings. The sequence diagrams for the various services are shown in 3.2.10.

3.2.5.1.1 With the permissible exception of abort service primitives, the AIDC-AE shall process service primitives in the order in which they are received: this ensures that the AE will, with the exception of aborts, guarantee message sequencing.

3.2.5.2 AIDC-AE CF State Definitions

3.2.5.2.1 The AIDC-AE CF shall be in one of the following states at a given time:

a) *Null* (STA0) – This is the state of the CF when there is no association in existence.

b) *Association Pending* (STA1) – The CF enters this state when the AIDC-User has invoked a Notify or Coordinate-start request primitive or an indication has been received that the peer has made a request to establish an association.

c) *Data Transfer* (STA2) – The CF enters this state once the establishment phase is complete. An association has successfully been established and the communicating partners are free to send and receive data.

d) *Release Pending* (STA3) – The CF enters this state when either the AIDC-User has requested the termination of the AIDC service or an indication has been received that the peer has made a request to terminate the association.

3.2.5.3 AIDC-AE CF Service Mappings

Note.— Figure 3.2.5-1 indicates which parts of this document specifies the behaviour of the CF in response to events at the various service interfaces.

3.2.5.3.1 AIDC-User Services Primitives Submitted to the CF

Note.— The following specifies the actions of the CF in response to events which occur at the upper service boundary of the AIDC-AE: specifically, request and response primitives generated by the AIDC-User.

3.2.5.3.1.1 Implicit Association

3.2.5.3.1.1.1 The invocation of an Info-transfer request primitive, or a Notify request primitive or a Coordinate-start request primitive shall implicitly cause the establishment of an association, if one does not exist, between two peer AIDC-AEs.

Note.— For a given flight, the handling of double associations between two peer AIDC-AEs is not managed by the AIDC-CF.

3.2.5.3.1.1.2 The association establishment and release between peer AIDC-AEs shall be performed by invoking the primitives of ACSE.

3.2.5.3.1.1.3 Upon the receipt of an Info-transfer request primitive, a Notify request primitive or Coordinate-start request primitive, the CF shall:

a) construct the Calling AP Title from the local ICAO Facility Designation according to the specification in 4.3.2;

b) invoke an A-ASSOCIATE Request primitive with the parameters specified in Table 3.2.5-1; and

c) enter the ASSOCIATION PENDING state.

Note.—To construct the Calling AP Title, the CF is assumed to have local knowledge of the ICAO Facility Designation of the AIDC-AE which it defines.

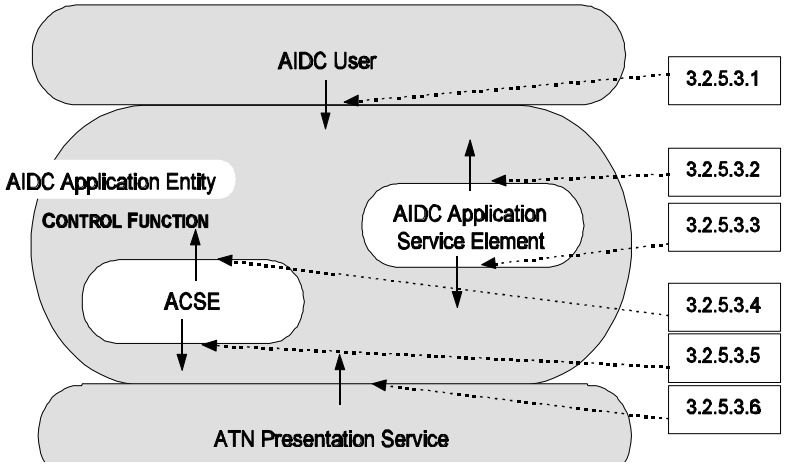

Figure 3.2.5-1: Elements of the AIDC-AE

Table 3.2.5-1: A-ASSOCIATE Request Primitive Parameters

A-ASSOCIATE Request Parameter	Value
Mode	Not used (default value)
Application Context Name	{iso (1) identified-organization (3) icao (27) atn-ac (3) 1 }
Application Context Name List	Not used
Calling AP Title	{iso (1) identified-organization (3) icao (27) atn-end-system-ground (2) <end-system-id> (n) operational (0) }
Calling AE Qualifier	idc (6)
Calling AP Invocation-identifier	Not used
Calling AE Invocation-identifier	Not used
Called AP Title	Not used
Called AE Qualifier	Not used
Called AP Invocation-identifier	Not used
Called AE Invocation-identifier	Not used
ACSE Requirements	Not used
Authentication-mechanism Name	Not used
Authentication-value	Not used
User Information	Not used
Calling Presentation Address	Local Implementation
Called Presentation Address	Local Implementation
Presentation Context Definition List	Not used
Default Presentation Context Name	Not used
Quality of Service	See 3.2.8.2
Presentation Requirements	Not used (default value)
Session Requirements	No Orderly Release (NOR), Duplex
Initial Synchronization Point Serial No	Not used
Initial Assignment of Tokens	Not used
Session-connection Identifier	Not used

3.2.5.3.1.2 User-confirmation Request Primitive

3.2.5.3.1.2.1 When Invoked

3.2.5.3.1.2.1.1 It shall be valid to invoke the User-confirmation request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.2.1.2 **Recommendation.**— *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, and the User-confirmation request primitive is invoked, then appropriate error recovery action should be taken.*

3.2.5.3.1.2.2 Action Upon Invocation

3.2.5.3.1.2.2.1 The CF shall invoke an AIDC-ucf request primitive with:

a) the User-confirmation *Result* parameter as the AIDC-ucf *Result* parameter;

b) the User-confirmation *Referenced Number* parameter as the AIDC-ucf *Reference ID* parameter; and

c) if present, the User-confirmation *Reason* parameter as the AIDC-ucf *Reason* parameter.

3.2.5.3.1.3 Notify Request Primitive

3.2.5.3.1.3.1 When Invoked

3.2.5.3.1.3.1.1 It shall be valid to invoke the Notify request primitive when the CF is in the NULL state, or the DATA TRANSFER state.

3.2.5.3.1.3.1.2 **Recommendation.**— *If the CF is in the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.3.2 Action Upon Invocation

3.2.5.3.1.3.2.1 If in the NULL state the CF shall:

a) store the value of the Notify request primitive parameters;

b) set the Boolean variable **nfy-assc;** and

c) perform the implicit association function as specified in 3.2.5.3.1.1.

3.2.5.3.1.3.2.2 If in the DATA TRANSFER state, the CF shall invoke an AIDC-nfy request primitive with:

a) the Notify request primitive *Notification Information* parameter as the AIDC-nfy request primitive *User Data* parameter; and

b) the Notify request primitive *Message Number* parameter as the AIDC-nfy request primitive *Msg Number* parameter.

3.2.5.3.1.4 Coordinate-start Request Primitive

3.2.5.3.1.4.1 When Invoked

3.2.5.3.1.4.1.1 It shall be valid to invoke the Coordinate-start request primitive when the CF is in the NULL state, or the DATA TRANSFER state.

3.2.5.3.1.4.1.2 **Recommendation.** — *If the CF is in the ASSOCIATION PENDING state, or the RELEASE PENDING state then appropriate error recovery action should be taken.*

3.2.5.3.1.4.2 Action Upon Invocation

3.2.5.3.1.4.2.1 If in the NULL state the CF shall:

a) store the value of the Coordinate-start request primitive parameters;

b) set the Boolean variable **crd-assc;** and

c) perform the implicit association function as specified in 3.2.5.3.1.1.

3.2.5.3.1.4.2.2 If in the DATA TRANSFER state, the CF shall invoke an AIDC-crd-start request primitive with:

a) the Coordinate-start request primitive *Coordination Start Information* parameter as the AIDC-crd-start request primitive *User Data* parameter; and

b) the Coordinate-start request primitive *Message Number* parameter as the AIDC-crd-start request primitive *Msg Number* parameter.

3.2.5.3.1.5 Coordinate-end Request Primitive

3.2.5.3.1.5.1 When Invoked

3.2.5.3.1.5.1.1 It shall be valid to invoke the Coordinate-end request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.5.1.2 **Recommendation.** — *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.5.2 Action Upon Invocation

3.2.5.3.1.5.2.1 Upon the receipt of a Coordinate-end request primitive, the CF shall invoke an AIDC-crd-end request primitive with:

a) the Coordinate-end request primitive *Coordinate End Information* parameter as the AIDC-crd-end request primitive *User Data* parameter;

b) the Coordinate-end request primitive *Result* parameter as the AIDC-crd-end request primitive *Result* parameter; and

c) the Coordinate-end request primitive *Message Number* parameter as the AIDC-crd-end request primitive *Msg Number* parameter.

3.2.5.3.1.6 Coordinate-negotiate Request Primitive

3.2.5.3.1.6.1 When Invoked

3.2.5.3.1.6.1.1 It shall be valid to invoke the Coordinate-negotiate request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.6.1.2 **Recommendation.** — *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.6.2 Action Upon Invocation

3.2.5.3.1.6.2.1 Upon the receipt of a Coordinate-negotiate request primitive, the CF shall invoke an AIDC-crd-ngtt request primitive with:

a) the Coordinate-negotiate request primitive *Coordinate Negotiate Information* parameter as the AIDC-crd-ngtt request primitive *User Data* parameter; and

b) the Coordinate-negotiate request primitive Message Number parameter as the AIDC-crd-ngtt request primitive Msg Number parameter.

3.2.5.3.1.7 Coordinate-standby Request Primitive

3.2.5.3.1.7.1 When Invoked

3.2.5.3.1.7.1.1 It shall be valid to invoke the Coordinate-standby request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.7.1.2 **Recommendation.** — *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.7.2 Action Upon Invocation

3.2.5.3.1.7.2.1 Upon the receipt of a Coordinate-standby request primitive, the CF shall invoke an AIDC-crd-stndby request primitive with:

a) the Coordinate-standby request primitive *Coordinate Standby Information* parameter as the AIDC-crd-stndby request primitive *User Data* parameter; and

b) the Coordinate-standby request primitive *Message Number* parameter as the AIDC-crd-stndby request primitive *Msg Number* parameter.

3.2.5.3.1.8 Transfer-initiate Request Primitive

3.2.5.3.1.8.1 When Invoked

3.2.5.3.1.8.1.1 It shall be valid to invoke the Transfer-initiate request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.8.1.2 **Recommendation.** — *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.8.2 Action Upon Invocation

3.2.5.3.1.8.2.1 Upon the receipt of a Transfer-initiate request primitive, the CF shall invoke an AIDC-tfr-init request primitive with:

a) the Transfer-initiate request primitive *Transfer Initiate Information* parameter as the AIDC-tfr-init request primitive *User Data* parameter; and

b) the Transfer-initiate request primitive *Message Number* parameter as the AIDC-tfr-init request primitive *Msg Number* parameter.

3.2.5.3.1.9 Transfer-request Request Primitive

3.2.5.3.1.9.1 When Invoked

3.2.5.3.1.9.1.1 It shall be valid to invoke the Transfer-request request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.9.1.2 **Recommendation.** — *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.9.2 Action Upon Invocation

3.2.5.3.1.9.2.1 Upon the receipt of a Transfer-request request primitive, the CF shall invoke an AIDC-tfr-rqst request primitive with:

a) the Transfer-request request primitive *Transfer Request Information* parameter as the AIDC-tfr-rqst request primitive *User Data* parameter; and

b) the Transfer-request request primitive *Message Number* parameter as the AIDC-tfr-rqst request primitive *Msg Number* parameter.

3.2.5.3.1.10 Transfer-conditions-proposal Request Primitive

3.2.5.3.1.10.1 When Invoked

3.2.5.3.1.10.1.1 It shall be valid to invoke the Transfer-conditions-proposal request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.10.1.2 **Recommendation.** — *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.10.2 Action Upon Invocation

3.2.5.3.1.10.2.1 Upon the receipt of a Transfer-conditions-proposal request primitive, the CF shall invoke an AIDC-tfr-prpsl request primitive with:

a) the Transfer-conditions-proposal request primitive *Transfer Conditions Proposal Information* parameter as the AIDC-tfr-prpsl request primitive *User Data* parameter; and

b) the Transfer-conditions-proposal request primitive *Message Number* parameter as the AIDC-tfr-prpsl request primitive *Msg Number* parameter.

3.2.5.3.1.11 Transfer-conditions-accept Request Primitive

3.2.5.3.1.11.1 When Invoked

3.2.5.3.1.11.1.1 It shall be valid to invoke the Transfer-conditions-accept request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.11.1.2 **Recommendation.**— *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.11.2 Action Upon Invocation

3.2.5.3.1.11.2.1 Upon the receipt of a Transfer-conditions-accept request primitive, the CF shall invoke an AIDC-tfr-accept request primitive with:

a) the Transfer-conditions-accept request primitive *Transfer Conditions Accept Information* parameter as the AIDC-tfr-accept request primitive *User Data* parameter; and

b) the Transfer-conditions-accept request primitive *Message Number* parameter as the AIDC-tfr-accept request primitive *Msg Number* parameter.

3.2.5.3.1.12 Transfer-control Request Primitive

3.2.5.3.1.12.1 When Invoked

3.2.5.3.1.12.1.1 It shall be valid to invoke the Transfer-control request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.12.1.2 **Recommendation.** — *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.12.2 Action Upon Invocation

3.2.5.3.1.12.2.1 Upon the receipt of a Transfer-control request primitive, the CF shall invoke an AIDC-tfr-cntrl request primitive with:

a) the Transfer-control request primitive *Transfer Control Information* parameter as the AIDC-tfr-cntrl request primitive *User Data* parameter; and

b) the Transfer-control request primitive *Message Number* parameter as the AIDC-tfr-cntrl request primitive *Msg Number* parameter.

3.2.5.3.1.13 Transfer-control Response Primitive

3.2.5.3.1.13.1 When Invoked

3.2.5.3.1.13.1.1 It shall be valid to invoke the Transfer-control response primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.13.1.2 **Recommendation.** — *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.13.2 Action Upon Invocation

3.2.5.3.1.13.2.1 Upon the receipt of a Transfer-control response primitive, the CF shall invoke an AIDC-tfr-cntrl response primitive with:

a) the Transfer-control response primitive *Transfer Control Information* parameter as the AIDC-tfr-cntrl response primitive *User Data* parameter;

b) the Transfer-control response primitive *Result* parameter as the AIDC-tfr-cntrl response primitive *Result* parameter; and

c) the Transfer-control response primitive *Message Number* parameter as the AIDC-tfr-cntrl response primitive *Msg Number* parameter.

3.2.5.3.1.14 Transfer-communication Request Primitive

3.2.5.3.1.14.1 When Invoked

3.2.5.3.1.14.1.1 It shall be valid to invoke the Transfer-communication request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.14.1.2 **Recommendation.** — *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.14.2 Action Upon Invocation

3.2.5.3.1.14.2.1 Upon the receipt of a Transfer-communication request primitive, the CF shall invoke an AIDC-tfr-comm request primitive with:

a) the Transfer-communication request primitive *Transfer Communication Information* parameter as the AIDC-tfr-comm request primitive *User Data* parameter; and

b) the Transfer-communication request primitive *Message Number* parameter as the AIDC-tfr-comm request primitive *Msg Number* parameter.

3.2.5.3.1.15 Transfer-communication-assume Request Primitive

3.2.5.3.1.15.1 When Invoked

3.2.5.3.1.15.1.1 It shall be valid to invoke the Transfer-communication-assume request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.15.1.2 **Recommendation.**— *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.15.2 Action Upon Invocation

3.2.5.3.1.15.2.1 Upon the receipt of a Transfer-communication-assume request primitive, the CF shall invoke an AIDC-tfr-comm-assm request primitive with:

a) the Transfer-communication request primitive *Transfer Communication Assume Information* parameter as the AIDC-tfr-comm-assm request primitive *User Data* parameter; and

b) the Transfer-communication request primitive *Message Number* parameter as the AIDC-tfr-comm-assm request primitive *Msg Number* parameter.

3.2.5.3.1.16 Info-transfer Request Primitive

3.2.5.3.1.16.1 When Invoked

3.2.5.3.1.16.1.1 It shall be valid to invoke the Info-transfer request primitive when the CF is in the NULL state, or the DATA TRANSFER state.

3.2.5.3.1.16.1.2 **Recommendation.**— *If the CF is in the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.16.2 Action Upon Invocation

3.2.5.3.1.16.2.1 If in the NULL state the CF shall:

a) store the value of the Info-transfer request primitive parameter;

b) set the Boolean variable **inf-assc**; and

c) perform the implicit association function as specified in 3.2.5.3.1.1.

3.2.5.3.1.16.2.2 If in the DATA TRANSFER state, the CF shall invoke an AIDC-inf-tfr request primitive with:

a) the Info-transfer request primitive *Information* parameter as the AIDC-inf-tfr request primitive *User Data* parameter; and

b) the Info-transfer request primitive *Message Number* parameter as the AIDC-inf-tfr request primitive *Msg Number* parameter.

3.2.5.3.1.17 End Request Primitive

3.2.5.3.1.17.1 When Invoked

3.2.5.3.1.17.1.1 It shall be valid to invoke the End request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.1.17.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.1.17.2 Action Upon Invocation

3.2.5.3.1.17.2.1 Upon the receipt of a End request primitive, the CF shall invoke an AIDC-end request primitive with:

a) the End request primitive *Cancel Information* parameter, if present, as the AIDC-end request primitive *User Data* parameter; and

b) the End request primitive *Message Number* parameter as the AIDC-end request primitive *Msg Number* parameter.

3.2.5.3.1.18 User-abort Request Primitive

3.2.5.3.1.18.1 When Invoked

3.2.5.3.1.18.1.1 It shall be valid to invoke the User-abort request primitive when the CF is in the ASSOCIATION PENDING state, or the DATA TRANSFER state, or the RELEASE PENDING state.

3.2.5.3.1.18.1.2 **Recommendation. —** *If the CF is in the NULL state, then appropriate error recovery action should be taken.*

3.2.5.3.1.18.2 Action Upon Invocation

3.2.5.3.1.18.2.1 Upon the receipt of a User-abort request primitive in either the DATA TRANSFER or the RELEASE PENDING state, the CF shall:

a) invoke an AIDC-usr-abrt request primitive; and

b) enter or remain in, the RELEASE PENDING state.

3.2.5.3.1.18.2.2 Upon the receipt of a User-abort request primitive in the ASSOCIATION PENDING state, the CF shall:

a) invoke an A-ABORT request primitive; and

b) enter the RELEASE PENDING state.

3.2.5.3.2 AIDC-ASE Service Primitives Delivered to the CF

Note.— The following specifies the actions of the CF in response to events which occur at the upper service boundary of the AIDC-ASE: specifically, indication and confirmation primitives which are generated by the AIDC-ASE protocol machine.

3.2.5.3.2.1 AIDC-ucf Indication Primitive

3.2.5.3.2.1.1 When Invoked

3.2.5.3.2.1.1.1 It shall be valid to invoke the AIDC-ucf indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.1.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.1.2 Action Upon Invocation

3.2.5.3.2.1.2.1 Upon the receipt of a AIDC-ucf indication primitive, the CF shall invoke a User-confirmation indication primitive with:

a) the AIDC-ucf indication primitive *Result* parameter as the User-confirmation indication primitive *Result* parameter;

b) if present, the AIDC-ucf indication primitive *Reason* parameter as the User-confirmation primitive *Reason* parameter; and

c) the AIDC-ucf indication *Reference ID* parameter as the User-confirmation indication *Referenced Number* parameter.

3.2.5.3.2.2 AIDC-nfy Indication Primitive

3.2.5.3.2.2.1 When Invoked

3.2.5.3.2.2.1.1 It shall be valid to invoke the AIDC-nfy indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.2.1.2 **Recommendation.—** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.2.2 Action Upon Invocation

3.2.5.3.2.2.2.1 Upon the receipt of a AIDC-nfy indication primitive, the CF shall invoke the Notify indication primitive with:

a) the stored Calling ICAO Facility Designation as the Notify indication primitive *Calling ICAO Facility Designation* parameter;

b) the AIDC-nfy indication primitive *User Data* parameter as the Notify indication primitive *Notification Information* parameter; and

c) the AIDC-nfy indication primitive *Msg Number* parameter as the Notify indication primitive *Message Number* parameter.

3.2.5.3.2.3 AIDC-crd-start Indication Primitive

3.2.5.3.2.3.1 When Invoked

3.2.5.3.2.3.1.1 It shall be valid to invoke the AIDC-crd-start indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.3.1.2 **Recommendation.—** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.3.2 Action Upon Invocation

3.2.5.3.2.3.2.1 Upon the receipt of a AIDC-crd-start indication primitive, the CF shall invoke the Coordinate-start indication primitive with:

a) the stored Calling ICAO Facility Designation as the Coordinate-start indication primitive *Calling ICAO Facility Designation* parameter;

b) the AIDC-crd-start indication primitive *User Data* parameter as the Coordinate-start indication primitive *Coordinate Start Information* parameter; and

c) the AIDC-crd-start indication primitive *Msg Number* parameter as the Coordinate-start indication primitive *Message Number* parameter.

3.2.5.3.2.4 AIDC-crd-end Indication Primitive

3.2.5.3.2.4.1 When Invoked

3.2.5.3.2.4.1.1 It shall be valid to invoke the AIDC-crd-end indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.4.1.2 **Recommendation.—** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.4.2 Action Upon Invocation

3.2.5.3.2.4.2.1 Upon the receipt of a AIDC-crd-end indication primitive the CF shall invoke a Coordinate-end indication primitive with:

a) the AIDC-crd-end indication primitive *User Data* parameter as the Coordinate-end indication primitive *Coordinate End Information* parameter; and

b) the AIDC-crd-end indication primitive *Msg Number* parameter as the Coordinate-end indication primitive *Message Number* parameter.

3.2.5.3.2.5 AIDC-crd-ngtt Indication Primitive

3.2.5.3.2.5.1 When Invoked

3.2.5.3.2.5.1.1 It shall be valid to invoke the AIDC-crd-ngtt indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.5.1.2 **Recommendation.—** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.5.2 Action Upon Invocation

3.2.5.3.2.5.2.1 Upon the receipt of a AIDC-crd-ngtt indication primitive the CF shall invoke a Coordinate-negotiate indication primitive with:

a) the AIDC-crd-ngtt indication primitive *User Data* parameter as the Coordinate-negotiate indication primitive *Coordinate Negotiate Information* parameter; and

b) the AIDC-crd-ngtt indication primitive *Msg Number* parameter as the Coordinate-negotiate indication primitive *Message Number* parameter.

3.2.5.3.2.6 AIDC-crd-stndby Indication Primitive

3.2.5.3.2.6.1 When Invoked

3.2.5.3.2.6.1.1 It shall be valid to invoke the AIDC-crd-stndby indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.6.1.2 **Recommendation.—** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.6.2 Action Upon Invocation

3.2.5.3.2.6.2.1 Upon the receipt of a AIDC-crd-stndby indication primitive the CF shall invoke a Coordinate-standby indication primitive with:

a) the AIDC-crd-stndby indication primitive *User Data* parameter as the Coordinate-standby indication primitive *Coordinate Standby Information* parameter; and

b) the AIDC-crd-stndby indication primitive *Msg Number* parameter as the Coordinate-standby indication primitive *Message Number* parameter.

3.2.5.3.2.7 AIDC-tfr-init Indication Primitive

3.2.5.3.2.7.1 When Invoked

3.2.5.3.2.7.1.1 It shall be valid to invoke the AIDC-tfr-init indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.7.1.2 **Recommendation.—** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.7.2 Action Upon Invocation

3.2.5.3.2.7.2.1 Upon the receipt of a AIDC-tfr-init indication primitive the CF shall invoke a Transfer-initiate indication primitive with:

a) the AIDC-tfr-init indication primitive *User Data* parameter as the Transfer-initiate indication primitive *Transfer Initiate Information* parameter; and

b) the AIDC-tfr-init indication primitive *Msg Number* parameter as the Transfer-initiate indication primitive *Message Number* parameter.

3.2.5.3.2.8 AIDC-tfr-rqst Indication Primitive

3.2.5.3.2.8.1 When Invoked

3.2.5.3.2.8.1.1 It shall be valid to invoke the AIDC-tfr-rqst indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.8.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.8.2 Action Upon Invocation

3.2.5.3.2.8.2.1 Upon the receipt of a AIDC-tfr-rqst indication primitive the CF shall invoke a Transfer-request indication primitive with:

a) the AIDC-tfr-rqst indication primitive *User Data* parameter as the Transfer-request indication primitive *Transfer Request Information* parameter; and

b) the AIDC-tfr-rqst indication primitive *Msg Number* parameter as the Transfer-request indication primitive *Message Number* parameter.

3.2.5.3.2.9 AIDC-tfr-prpsl Indication Primitive

3.2.5.3.2.9.1 When Invoked

3.2.5.3.2.9.1.1 It shall be valid to invoke the AIDC-tfr-prpsl indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.9.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.9.2 Action Upon Invocation

3.2.5.3.2.9.2.1 Upon the receipt of a AIDC-tfr-prpsl indication primitive the CF shall invoke a Transfer-conditions-proposal indication primitive with:

a) the AIDC-tfr-prpsl indication primitive *User Data* parameter as the Transfer-conditions-proposal indication primitive *Transfer Conditions Proposal Information* parameter; and

b) the AIDC-tfr-prpsl indication primitive *Msg Number* parameter as the Transfer-conditions-proposal indication primitive *Message Number* parameter.

3.2.5.3.2.10 AIDC-tfr-accept Indication Primitive

3.2.5.3.2.10.1 When Invoked

3.2.5.3.2.10.1.1 It shall be valid to invoke the AIDC-tfr-accept indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.10.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.10.2 Action Upon Invocation

3.2.5.3.2.10.2.1 Upon the receipt of a AIDC-tfr-accept indication primitive the CF shall invoke a Transfer-conditions-accept indication primitive with:

a) the AIDC-tfr-accept indication primitive *User Data* parameter as the Transfer-conditions-accept indication primitive *Transfer Conditions Accept Information* parameter; and

b) the AIDC-tfr-accept indication primitive *Msg Number* parameter as the Transfer-conditions-accept indication primitive *Message Number* parameter.

3.2.5.3.2.11 AIDC-tfr-cntrl Indication Primitive

3.2.5.3.2.11.1 When Invoked

3.2.5.3.2.11.1.1 It shall be valid to invoke the AIDC-tfr-cntrl indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.11.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.11.2 Action Upon Invocation

3.2.5.3.2.11.2.1 Upon the receipt of a AIDC-tfr-cntrl indication primitive the CF shall invoke a Transfer-control indication primitive with:

a) the AIDC-tfr-cntrl indication primitive *User Data* parameter as the Transfer-control indication primitive *Transfer Control Information* parameter; and

b) the AIDC-tfr-cntrl indication primitive *Msg Number* parameter as the Transfer-control indication primitive *Message Number* parameter.

3.2.5.3.2.12 AIDC-tfr-cntrl Confirmation Primitive

3.2.5.3.2.12.1 When Invoked

3.2.5.3.2.12.1.1 It shall be valid to invoke the AIDC-tfr-cntrl confirmation primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.12.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.12.2 Action Upon Invocation

3.2.5.3.2.12.2.1 Upon the receipt of a AIDC-tfr-cntrl confirmation primitive the CF shall invoke a Transfer-control confirmation primitive with:

a) the AIDC-tfr-cntrl confirmation primitive *User Data* parameter as the Transfer-control confirmation primitive *Transfer Control Information* parameter;

b) the AIDC-tfr-cntrl confirmation primitive *Result* parameter as the Transfer-control confirmation primitive *Result* parameter; and

c) the AIDC-tfr-cntrl confirmation primitive *Msg Number* parameter as the Transfer-control confirmation primitive *Message Number* parameter.

3.2.5.3.2.13 AIDC-tfr-comm Indication Primitive

3.2.5.3.2.13.1 When Invoked

3.2.5.3.2.13.1.1 It shall be valid to invoke the AIDC-tfr-comm indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.13.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.13.2 Action Upon Invocation

3.2.5.3.2.13.2.1 Upon the receipt of a AIDC-tfr-comm indication primitive the CF shall invoke a Transfer-communication indication primitive with:

a) the AIDC-tfr-comm indication primitive *User Data* parameter as the Transfer-communication indication primitive *Transfer Communication Information* parameter; and

b) the AIDC-tfr-comm indication primitive *Msg Number* parameter as the Transfer-communication indication primitive *Message Number* parameter.

3.2.5.3.2.14 AIDC-tfr-comm-assm Indication Primitive

3.2.5.3.2.14.1 When Invoked

3.2.5.3.2.14.1.1 It shall be valid to invoke the AIDC-tfr-comm-assm indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.14.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.14.2 Action Upon Invocation

3.2.5.3.2.14.2.1 Upon the receipt of a AIDC-tfr-comm-assm indication primitive the CF shall invoke a Transfer-communication-assume indication primitive with:

a) the AIDC-tfr-comm-assm indication primitive *User Data* parameter as the Transfer-communication-assume indication primitive *Transfer Communication Assume Information* parameter; and

b) the AIDC-tfr-comm-assm indication primitive *Msg Number* parameter as the Transfer-communication-assume indication primitive *Message Number* parameter.

3.2.5.3.2.15 AIDC-inf-tfr Indication Primitive

3.2.5.3.2.15.1 When Invoked

3.2.5.3.2.15.1.1 It shall be valid to invoke the AIDC-inf-tfr indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.15.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.15.2 Action Upon Invocation

3.2.5.3.2.15.2.1 Upon the receipt of a AIDC-inf-tfr indication primitive the CF shall invoke a Info-transfer indication primitive with:

a) the AIDC-inf-tfr indication primitive *User Data* parameter as the Info-transfer indication primitive *Information* parameter; and

b) the AIDC-inf-tfr indication primitive *Msg Number* parameter as the Info-transfer indication primitive *Message Number* parameter.

3.2.5.3.2.16 AIDC-end Indication Primitive

3.2.5.3.2.16.1 When Invoked

3.2.5.3.2.16.1.1 It shall be valid to invoke the AIDC-end indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.2.16.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.16.2 Action Upon Invocation

3.2.5.3.2.16.2.1 Upon the receipt of a AIDC-end indication primitive the CF shall invoke a End indication primitive with:

a) the AIDC-end indication primitive *User Data* parameter, if present, as the End indication primitive *Cancel Information* parameter; and

b) the AIDC-end indication primitive *Msg Number* parameter as the End indication primitive *Message Number* parameter.

3.2.5.3.2.17 AIDC-usr-abrt Indication Primitive

3.2.5.3.2.17.1 When Invoked

3.2.5.3.2.17.1.1 It shall be valid to invoke the AIDC-usr-abrt indication primitive when the CF is in the ASSOCIATION PENDING state, or the DATA TRANSFER state.

3.2.5.3.2.17.1.2 **Recommendation. —** *If the CF is in the NULL state, then appropriate error recovery action should be taken.*

3.2.5.3.2.17.2 Action Upon Invocation

3.2.5.3.2.17.2.1 Upon the receipt of a AIDC-usr-abrt indication primitive the CF shall invoke a User-abort indication primitive.

3.2.5.3.2.18 AIDC-pvd-abrt Indication Primitive

3.2.5.3.2.18.1 When Invoked

3.2.5.3.2.18.1.1 It shall be valid to invoke the AIDC-pvd-abrt indication primitive when the CF is in the DATA TRANSFER state, or the RELEASE PENDING state.

3.2.5.3.2.18.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.2.18.2 Action Upon Invocation

3.2.5.3.2.18.2.1 Upon the receipt of a AIDC-pvd-abrt indication primitive the CF shall:

a) if the CF is in the DATA TRANSFER state:

1) invoke a Provider-abort indication primitive with the Provider-abort *Provider Abort Reason* parameter having the value delivered in the AIDC-pvd-abrt *Abort Reason* parameter; and

2) enter the RELEASE PENDING state.

b) if the CF is in the RELEASE PENDING state invoke a Provider-abort indication primitive with the Provider-abort *Provider Abort Reason* parameter having the abstract value delivered in the AIDC-pvd-abrt *Abort Reason* parameter.

3.2.5.3.3 AIDC-ASE Service Primitives Submitted to the CF

Note.— The following specifies the actions of the CF in response to events which occur at the lower service boundary of the AIDC-ASE: specifically, request primitives which are generated by the AIDC-ASE protocol machine.

3.2.5.3.3.1 AIDC-DATA Request primitive

3.2.5.3.3.1.1 When Invoked

3.2.5.3.3.1.1.1 It shall be valid to invoke the AIDC-DATA request primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.3.1.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.3.1.2 Action Upon Invocation

3.2.5.3.3.1.2.1 Upon the receipt of a AIDC-DATA request primitive the CF shall:

a) encode the AIDC-DATA Request *User Data* parameter using the definition of presentation-user-data in 4.3.2.6; and

b) invoke a P-DATA request primitive with the resulting encoding from a) above, as the *User Data* parameter.

3.2.5.3.3.2 AIDC-ABORT Request Primitive

3.2.5.3.3.2.1 When Invoked

3.2.5.3.3.2.1.1 It shall be valid to invoke the AIDC-ABORT request primitive when the CF is in the DATA TRANSFER state, or the RELEASE PENDING state.

3.2.5.3.3.2.1.2 **Recommendation.—** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.3.2.2 Action Upon Invocation

3.2.5.3.3.2.2.1 Upon the receipt of a AIDC-ABORT request primitive the CF shall:

a) if the CF is in the DATA TRANSFER state:

1) invoke an A-ABORT request primitive with parameters as follows:

i) if no AIDC-ABORT *Abort Reason* parameter is present, the A-ABORT *AbortSource* parameter set to abstract value “acse-service-user” and the A-ABORT *Diagnostic* value set to the abstract value “no-reason-given”;

ii) if the AIDC-ABORT *Abort Reason* parameter has either one of the abstract values “protocolerror” or “timerexpired”, the A-ABORT *AbortSource* prameter set to abstract value “acse-service-user” and the A-ABORT *Diagnostic* value set to the abstract value “protocol-error”;

iii) otherwise, the A-ABORT AbortSource parameter set to the abstract value “acse-service-user” and no A-ABORT Diagnostic value; and

2) enter the RELEASE PENDING state;

b) if the CF is in the RELEASE PENDING state, invoke an A-ABORT request primitive with no parameters.

3.2.5.3.4 ACSE Service Primitives Delivered to the CF

Note.— The following specifies the action of the CF in response to events which occur at the upper service boundary of ACSE: specifically, indication and confirmation primitives which are generated by the ACSE Protocol Machine (ACPM).

3.2.5.3.4.1 A-ASSOCIATE Indication Primitive

3.2.5.3.4.1.1 When Invoked

3.2.5.3.4.1.1.1 It shall be valid to invoke the A-ASSOCIATE indication primitive when the CF is in the ASSOCIATION PENDING state.

3.2.5.3.4.1.1.2 **Recommendation. —** *If the CF is in the NULL state or the DATA TRANSFER state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.4.1.2 Action Upon Invocation

3.2.5.3.4.1.2.1 Upon the receipt of an A-ASSOCIATE indication primitive, the CF shall:

a) extract and store the encoded Calling ICAO Facility Designation from *Calling AP Title* parameter; and

b) invoke an A-ASSOCIATE response primitive with the parameters as shown in Table 3.2.5-2.

Table 3.2.5-2: A-ASSOCIATE Response Primitive Parameters

Parameter	Value
Mode	Not used (default value)
Application Context Name	{iso (1) identified-organization (3) icao (27) atn-ac (3) 1 }
Application Context Name List	Not used
Responding AP Title	Not used
Responding AE Qualifier	Not used
Responding AP Invocation-identifier	Not used
Responding AE Invocation-identifier	Not used
ACSE Requirements	Not used
Authentication-mechanism Name	Not used

Parameter	Value
Authentication-value	Not used
User Information	Not used
Result	Not used
Diagnostic	Not used
Responding Presentation Address	Local Implementation
Presentation Context Definition Result List	Not used
Default Presentation Context Result	Not used
Quality of Service	Not used
Presentation Requirements	Not used (default value)
Session Requirements	No Orderly Release (NOR), Duplex
Initial Synchronization Point Serial No	Not used
Initial Assignment of Tokens	Not used
Session-connection Identifier	Not used

3.2.5.3.4.2 A-ASSOCIATE Confirmation Primitive

3.2.5.3.4.2.1 When Invoked

3.2.5.3.4.2.1.1 It shall be valid to invoke the A-ASSOCIATE confirmation primitive when the CF is in the ASSOCIATION PENDING state.

3.2.5.3.4.2.1.2 **Recommendation. —** *If the CF is in the NULL state, or the DATA TRANSFER state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.4.2.2 Action Upon Invocation

3.2.5.3.4.2.2.1 Upon the receipt of an A-ASSOCIATE confirmation primitive, the CF shall:

a) if the *Result* parameter has the abstract value “accepted”, then:

1) if the **nfy-assc** variable is set:- invoke an AIDC-nfy request primitive with:

i) the stored *Notification Information* as the *User Data* parameter;

ii) the stored *Message Number* parameter as the *Msg Number* parameter; and

iii) enter the DATA TRANSFER state; or

2) if the **crd-assc** variable is set:- invoke an AIDC-crd-start request primitive with:

i) the stored *Coordinate Start Information* as the *User Data* parameter;

ii) the stored *Message Number* parameter as the *Msg Number* parameter; and

iii) enter the DATA TRANSFER state.

3) if the **inf-assc** variable is set:- invoke an AIDC-inf-tfr request primitive with:

i) the stored Info-transfer *Information* as the *User Data* parameter;

ii) the stored *Message Number* parameter as the *Msg Number* parameter; and

iii) enter the DATA TRANSFER state.

b) if the *Result* parameter has the abstract value "rejected (permanent)" or "rejected (transient)", then:

1) invoke a Provider-abort indication primitive with the A-ASSOCIATE confirmation primitive *Result Source* parameter as the Provider-abort indication primitive *Provider Abort Reason* parameter set to the abstract value "rejectedpermanent" or "rejectedtransient" corresponding to the value of the A-ASSOCIATE indication Result parameter; and

2) enter the NULL state.

3.2.5.3.4.3 A-ABORT Indication primitive

3.2.5.3.4.3.1 When Invoked

3.2.5.3.4.3.1.1 It shall be valid to invoke the A-ABORT indication primitive when the CF is in the ASSOCIATION PENDING state, or the DATA TRANSFER state, or the RELEASE PENDING state.

3.2.5.3.4.3.1.2 **Recommendation. —** *If the CF is in the NULL state, then appropriate error recovery action should be taken.*

3.2.5.3.4.3.2 Action Upon Invocation

3.2.5.3.4.3.2.1 Upon the receipt of an A-ABORT indication primitive, the CF shall:

a) if the CF is in the ASSOCIATION PENDING state, or the DATA TRANSFER state:

1) invoke an AIDC-ABORT indication primitive as follows:

i) in each case, ignore any delivered A-ABORT *UserInformation* parameter value;

ii) if the A-ABORT *AbortSource* parameter has the abstract value "acse-service-user" and the A-ABORT *Diagnostic* parameter has the abstract value "no-reason-given", omit the AIDC-ABORT *AbortReason* parameter;

iii) if the A-ABORT *Diagnostic* parameter has the abstract value "protocol-error", set the AIDC-ABORT *AbortReason* parameter to the abstract value "protocolerror";

iv) otherwise, set the AIDC-ABORT *Abort Reason* parameter to the abstract value "undefinederror"; and

2) enter the RELEASE PENDING state.

b) if the CF is in the RELEASE PENDING state, enter the NULL state.

3.2.5.3.4.4 A-P-ABORT Indication Primitive

3.2.5.3.4.4.1 When Invoked

3.2.5.3.4.4.1.1 It shall be valid to invoke the A-P-ABORT indication primitive when the CF is in the ASSOCIATION PENDING state, or the DATA TRANSFER state, or the RELEASE PENDING state.

3.2.5.3.4.4.1.2 **Recommendation. —** *If the CF is in the NULL state, then appropriate error recovery action should be taken.*

3.2.5.3.4.4.2 Action Upon Invocation

3.2.5.3.4.4.2.1 Upon the receipt of an A-P-ABORT indication primitive, the CF shall:

a) if in the ASSOCIATION PENDING state:

1) invoke a Provider-abort indication with abstract value "providererror" as the value of the Provider-abort indication primitive *Provider Abort Reason* parameter, and discard any ProviderReason parameter in the A-P-ABORT indication; and

2) enter the NULL state.

b) if in the DATA TRANSFER state, or the RELEASE PENDING state:

1) invoke an AIDC-P-ABORT indication primitive, and discard any *Provider Reason* parameter in the A-P-ABORT indication; and

2) enter the RELEASE PENDING state.

3.2.5.3.5 ACSE Service Primitives Submitted to the CF

Note 1.— The following specifies the actions of the CF in response to events at the lower service boundary of ACSE: specifically, request and response primitives generated by the ACPM.

Note 2.— ACSE (Edition 2) mandates the mapping between ACSE and the underlying Presentation service provider. Invocations of Presentation service primitives by ACSE are "intercepted" by the CF and re-mapped to the "actual" Presentation service as appropriate.

3.2.5.3.5.1 P-CONNECT Request Primitive

3.2.5.3.5.1.1 When Invoked

3.2.5.3.5.1.1.1 It shall be valid to invoke the P-CONNECT request primitive when the CF is in the ASSOCIATION PENDING state.

3.2.5.3.5.1.1.2 **Recommendation. —** *If the CF is in the NULL state, or the DATA TRANSFER state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.5.1.2 Action Upon Invocation

3.2.5.3.5.1.2.1 Upon the receipt of a P-CONNECT request primitive, the CF shall invoke the equivalent Presentation service primitive of the ATN service provider.

3.2.5.3.5.2 P-CONNECT Response Primitive

3.2.5.3.5.2.1 When Invoked

3.2.5.3.5.2.1.1 It shall be valid to invoke the P-CONNECT response primitive when the CF is in the ASSOCIATION PENDING state.

3.2.5.3.5.2.1.2 **Recommendation. —** *If the CF is in the DATA TRANSFER state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.5.2.2 Action Upon Invocation

3.2.5.3.5.2.2.1 Upon the receipt of a P-CONNECT response primitive, the CF shall:

a) transparently invoke the equivalent presentation service primitive; and

b) enter the DATA TRANSFER state.

3.2.5.3.5.3 P-U-ABORT Request Primitive

3.2.5.3.5.3.1 When Invoked

3.2.5.3.5.3.1.1 It shall be valid to invoke the P-U-ABORT request primitive when the CF is in the ASSOCIATION PENDING state, or the DATA TRANSFER state, or the RELEASE PENDING state.

3.2.5.3.5.3.1.2 **Recommendation. —** *If the CF is in the NULL state, then appropriate error recovery action should be taken.*

3.2.5.3.5.3.2 Action Upon Invocation

3.2.5.3.5.3.2.1 Upon the receipt of a P-U-ABORT request primitive, the CF shall:

a) if the P-U-Abort request user data parameter is present, and the CF is in the DATA TRANSFER state:

1) encode the presentation user data as indicated in 4.3.2.6 with the P-U-ABORT user data parameter (an ACSE ABRT APDU) as the presentation data value and presentation context identifier value corresponding to “acse-apdu”; and

2) invoke a P-DATA Request primitive with the resulting encoding as the value of the UserData parameter;

b) otherwise, invoke a P-U-ABORT request primitive without any parameters;

c) in either case, enter the NULL state.

Note.— The invocation of the P-U-ABORT request primitive will abort the connection to the underlying ATN Service Provider.

3.2.5.3.6 Presentation Service Primitives Delivered to the CF

Note 1.— The following specifies the actions of the CF in response to events which occur at the lower service boundary of the AIDC-AE: specifically, indication and confirmation primitives which are delivered by the Presentation service.

Note 2.— ACSE (Edition 2) mandates the mapping between ACSE and the underlying Presentation service provider. Presentation service primitives are "intercepted" by the CF.

3.2.5.3.6.1 P-CONNECT Indication Primitive

3.2.5.3.6.1.1 When Invoked

3.2.5.3.6.1.1.1 It shall be valid to invoke the P-CONNECT indication primitive when the CF is in the NULL state.

3.2.5.3.6.1.1.2 **Recommendation. —** *If the CF is in the ASSOCIATION PENDING state, or the DATA TRANSFER state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.6.1.2 Action Upon Invocation

3.2.5.3.6.1.2.1 Upon the receipt of a P-CONNECT indication primitive, the CF shall:

a) enter the ASSOCIATION PENDING state; and

b) invoke the equivalent Presentation service primitive at the lower ACSE service boundary.

3.2.5.3.6.2 P-CONNECT Confirmation Primitive

3.2.5.3.6.2.1 When Invoked

3.2.5.3.6.2.1.1 It shall be valid to invoke the P-CONNECT confirmation primitive when the CF is in the ASSOCIATION PENDING state.

3.2.5.3.6.2.1.2 **Recommendation. —** *If the CF is in the NULL state, or the DATA TRANSFER state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.6.2.2 Action Upon Invocation

3.2.5.3.6.2.2.1 Upon the receipt of a P-CONNECT confirmation primitive, the CF shall invoke the equivalent Presentation service primitive at the lower ACSE service boundary.

3.2.5.3.6.3 P-U-ABORT Indication Primitive

3.2.5.3.6.3.1 When Invoked

3.2.5.3.6.3.1.1 It shall be valid to invoke the P-U-ABORT indication primitive when the CF is in the ASSOCIATION PENDING state, or the DATA TRANSFER state, or the RELEASE PENDING state.

3.2.5.3.6.3.1.2 **Recommendation. —** *If the CF is in the NULL state, then appropriate error recovery action should be taken.*

3.2.5.3.6.3.2 Action Upon Invocation

3.2.5.3.6.3.2.1 Upon the receipt of a P-U-ABORT indication primitive, the CF shall invoke the equivalent Presentation service primitive at the lower ACSE service boundary.

3.2.5.3.6.4 P-P-ABORT Indication Primitive

3.2.5.3.6.4.1 When Invoked

3.2.5.3.6.4.1.1 It shall be valid to invoke the P-P-ABORT indication primitive when the CF is in the ASSOCIATION PENDING state, or the DATA TRANSFER state, or the RELEASE PENDING state.

3.2.5.3.6.4.1.2 **Recommendation. —** *If the CF is in the NULL state, then appropriate error recovery action should be taken.*

3.2.5.3.6.4.2 Action Upon Invocation

3.2.5.3.6.4.2.1 Upon the receipt of a P-P-ABORT indication primitive, the CF shall invoke the corresponding Presentation service primitive at the lower ACSE service boundary.

3.2.5.3.6.5 P-DATA Indication Primitive

3.2.5.3.6.5.1 When Invoked

3.2.5.3.6.5.1.1 It shall be valid to invoke the P-DATA indication primitive when the CF is in the DATA TRANSFER state.

3.2.5.3.6.5.1.2 **Recommendation. —** *If the CF is in the NULL state, or the ASSOCIATION PENDING state, or the RELEASE PENDING state, then appropriate error recovery action should be taken.*

3.2.5.3.6.5.2 Action Upon Invocation

3.2.5.3.6.5.2.1 Upon the receipt of a P-DATA indication primitive, the CF shall:

a) decode the presentation-user-data as indicated in 4.3.2.6 to determine the destination ASE of the APDU, and extract the Presentation data value.

b) if the presentation context identifier has the abstract value “user-ase”, invoke an AIDC-DATA indication primitive with the extracted AIDC-APDU as the AIDC-DATA *AIDC Data* parameter;

c) if the presentation context identifier has the abstract value “acse-ase” and the ACSE-APDU has the syntax *ABRT-apdu*, invoke a P-U-Abort indication primitive with the *UserData* parameter containing the received APDU.

Note.— ABRT is the only ACSE APDU that is transmitted using P-DAT by the AIDC CF.

3.2.5.4 AIDC-CF State Table

3.2.5.4.1 The AIDC-AE shall behave as if it has a CF which functions in accordance with the following state table.

Note.— Table 3.2.5-3 shows the state transitions and actions performed by the AIDC-CF in response to service primitives submitted to the AIDC-CF. The source of the service primitive invocation is shown in column one of the Table 3.2.5-3 and the service primitive invocations are shown in the second column.

3.2.5.4.2 In the event of discrepancies between the state Table 3.2.5-3 and the text above, the text shall take precedence.

3.2.5.4.3 Each cell in the state Table 3.2.5-3 shows:

a) the action, if any, which the CF shall perform; and

b) the new state that the CF shall enter.

3.2.5.4.4 Blank cells shall indicate error conditions.

3.2.5.4.4.1 The error handling shall result in the association being aborted, if one exists.

3.2.5.4.4.2 **Recommendation.**— *The AIDC-User should be notified when an association is aborted.*

Table 3.2.5-3: AIDC CF State Table

Source ⇓	**State ⇒** **Event ⇓**	**STA0** **NULL**	**STA1** **ASSOCIATION PENDING**	**STA2** **DATA TRANSFER**	**STA3** **RELEASE PENDING**
From AIDC-User	Notify Req	• Store Notification Information • set nfy-assc • A-ASSOCIATE Req ⇒STA1		• AIDC-nfy Req ⇒STA2	
	Coordinate-start Req	• Store Coordination Start Information • set crd-assc • A-ASSOCIATE Req ⇒STA1		• AIDC-crd-start Req ⇒STA2	

Source ⇓	**State ⇒** **Event ⇓**	**STA0** **NULL**	**STA1** **ASSOCIATION PENDING**	**STA2** **DATA TRANSFER**	**STA3** **RELEASE PENDING**
	Info-transfer Req	• Store Info-transfer information • set inf-assc • A-ASSOCIATE Req ⇒STA1		• AIDC-inf-tfr Req ⇒STA2	
	End Req			• set end = Message Number • AIDC-end Req ⇒STA2	
	User-abort Req		•A-ABORT Req ⇒STA3	• AIDC-usr-abrt Req ⇒STA3	• AIDC-usr-abrt Req ⇒STA3
	All other Req			• equivalent AIDC-ASE Req ⇒STA2	
From AIDC-ASE (upper)	AIDC-ucf Ind			• User-confirmation Ind ⇒STA2	
	AIDC-usr-abrt Ind		• User-abort Ind ⇒STA0	• User-abort Ind ⇒STA0	• User-abort Ind ⇒STA0
	AIDC-pvd-abrt Ind			• Provider-abort Ind ⇒STA3	• Provider-abort Ind ⇒STA3
	All other Ind			• equivalent AIDC-User service invocation ⇒STA2	
From AIDC-ASE (lower)	AIDC-DATA Req			• P-DATA Req (APDU) ⇒STA2	
	AIDC-ABORT Req			• A-ABORT Req ⇒STA3	• A-ABORT Req ⇒STA3
From ACSE (upper)	A-ASSOCIATE Ind		• A-ASSOCIATE Rsp+ ⇒STA1		

Source ⇓	**State ⇒** **Event ⇓**	**STA0** **NULL**	**STA1** **ASSOCIATION PENDING**	**STA2** **DATA TRANSFER**	**STA3** **RELEASE PENDING**
	A-ASSOCIATE Cnf+		if nfy-assc • AIDC-nfy Req or if crd-assc • AIDC-crd-start Req or if inf-assc • AIDC-inf-tfr Req ⇒STA2		
	A-ASSOCIATE Cnf-		• Provider-abort Ind ⇒STA0		
	A-ABORT Ind		• Provider-abort Ind ⇒STA0	• AIDC-ABORT Ind ⇒STA3	⇒STA0
	A-P-ABORT Ind		• Provider-abort Ind ⇒STA0	• AIDC-P-ABORT Ind ⇒STA3	• AIDC-P-ABORT Ind ⇒STA3
From ACSE (lower)	P-CONNECT Req		• P-CONNECT Req ⇒STA1		
	P-CONNECT Rsp+		• P-CONNECT Rsp+ ⇒STA2		
	P-CONNECT Rsp-		• P-CONNECT Rsp- ⇒STA0		
	P-U-ABORT Req		• P-U-ABORT Req ⇒STA0	• P-DATA Req with ACSE ABRT APDU as UserData and “acse-ase” as presentation context identifier value ⇒STA0	• P-U-ABORT Req ⇒STA0
From ATN Service Provider	P-CONNECT Ind	• P-CONNECT Ind ⇒STA1			
	P-CONNECT Cnf+/-		• P-CONNECT Cnf ⇒STA1		

Source ⇓	**State ⇒** **Event ⇓**	**STA0** **NULL**	**STA1** **ASSOCIATION PENDING**	**STA2** **DATA TRANSFER**	**STA3** **RELEASE PENDING**
	P-DATA Ind			• if presentation context identifier = “user-ase” { • P-DATA Ind ⇒ STA2} • if presentation context identifier = “acse-ase” { • P-U-Abort Ind with ABRT APDU as UserData ⇒ STA2}	
	P-U-ABORT Ind		• P-U-ABORT Ind ⇒STA 1	• P-U-ABORT Ind ⇒STA2	• P-U-ABORT Ind ⇒STA3
	P-P-ABORT Ind		• P-P-ABORT Ind ⇒STA1	• P-P-ABORT Ind ⇒STA2	• P-P-ABORT Ind ⇒STA3

3.2.6 THE AIDC-ASE PROTOCOL DEFINITION

Note.— The following specifies the AIDC-ASE protocol.

3.2.6.1 AIDC-ASE Protocol Description

3.2.6.1.1 Only if requirements are described for an AIDC-ASE primitive when an AIDC-ASE is in a particular state, shall the invocation of that primitive be permitted while the AIDC-ASE is in that state.

3.2.6.1.2 If no requirements are described for the arrival of an APDU when the AIDC-ASE is in a particular state, then exception handling procedures as described in 3.2.6.2 shall apply.

3.2.6.1.3 **Recommendation.—** *Appropriate error recovery action should be taken, upon the arrival of an unexpected APDU.*

3.2.6.1.4 Predicates

Note.— The AIDC-ASE protocol has one predicate, P1, defined. This predicate conditions the use of the AIDC-tfr-accept service.

3.2.6.1.4.1 When the predicate P1 is set true, then the AIDC-tfr-accept service shall be used in response to the AIDC-tfr-prpsl service.

3.2.6.1.4.2 When the predicate P1 is set false, the invocation of the AIDC-tfr-accept service shall be prohibited.

3.2.6.1.5 AIDC-nfy Request Primitive

3.2.6.1.5.1 When Invoked

3.2.6.1.5.1.1 It shall be valid to invoke the AIDC-nfy request primitive when the AIDC-ASE protocol machine is in the IDLE state, or the NOTIFY state.

3.2.6.1.5.2 Action Upon Invocation

3.2.6.1.5.2.1 Upon the receipt of an AIDC-nfy request primitive the AIDC-ASE shall:

a) If in the IDLE state or the NOTIFY state:

1) create an AIDC-nfy-apdu based on the *User Data* parameter, and the *Msg Number* parameter;

2) invoke an AIDC-DATA request primitive with the AIDC-nfy-apdu as the *AIDC Data* parameter;

3) if running, stop timer t_{2IN};

4) if in the NOTIFY state, then stop the timer t_{2NC};

5) start the timer t_C; and

6) set the variable vs_1 = notify and vs_2 = *Msg Number*.

b) If the AIDC-ASE protocol machine is any other state:

1) invoke an AIDC-pvd-abrt indication primitive with the *AbortReason* parameter set to the abstract value: “protocolerror”;

2) invoke an AIDC-ABORT request primitive with the *AbortReason* set to abstract value “protocolerror”; and

3) enter the IDLE state.

3.2.6.1.6 AIDC-DATA Indication Primitive with an AIDC-nfy-apdu

3.2.6.1.6.1 When Invoked

3.2.6.1.6.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-nfy-apdu when the AIDC-ASE protocol machine is in the IDLE state, or the NOTIFY state.

3.2.6.1.6.2 Action Upon Invocation

3.2.6.1.6.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-nfy-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, and the *Msg Number* parameter from the AIDC-nfy-apdu;

b) invoke an AIDC-nfy indication primitive with the extracted parameters in a) above as the AIDC-nfy indication primitive *User Data* parameter, and the AIDC-nfy indication *Msg Number* parameter;

c) if running stop the timer t_{1IN};

d) if running, stop the timer t_{2R};

e) if in the NOTIFY state, then stop the timer t_{1NC}; and

f) set the variable vr_1 = notify and vr_2 = *Msg Number*.

3.2.6.1.7 AIDC-crd-start Request Primitive

3.2.6.1.7.1 When Invoked

3.2.6.1.7.1.1 It shall be valid to invoke the AIDC-crd-start request primitive when the AIDC-ASE protocol machine is in the IDLE state, or the NOTIFY state, or the COORDINATED state, or the TRANSFERRED state.

3.2.6.1.7.2 Action Upon Invocation

3.2.6.1.7.2.1 Upon the receipt of an AIDC-crd-start request primitive the AIDC-ASE shall:

a) create an AIDC-crd-start-apdu based on the *User Data* parameter, and the *Msg Number* parameter;

b) invoke an AIDC-DATA request primitive with the AIDC-crd-start-apdu as the *AIDC Data* parameter;

c) if running, stop timer t_{2IN};

d) if in the NOTIFY state, stop timer T_{1CT};

e) if in the COORDINATED state, then:

1) if running, stop timer t_{1CT};

2) if running, stop timer t_{2CT};

f) if in the TRANSFERRED state, then stop the timer t_{TE};

g) start the timer t_C; and

h) set the variable vs_1 = back and vs_2 = *Msg Number*.

3.2.6.1.8 AIDC-DATA Indication Primitive with an AIDC-crd-start-apdu

3.2.6.1.8.1 When Invoked

3.2.6.1.8.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-crd-start-apdu when the AIDC-ASE protocol machine is in the IDLE state, or the NOTIFY state, or the COORDINATED state, or the TRANSFERRED state.

3.2.6.1.8.2 Action Upon Invocation

3.2.6.1.8.2.1 Upon the receipt of a AIDC-DATA indication primitive with an AIDC-crd-start-apdu the AIDC ASE shall:

a) extract the *User Data* parameter, and the *Msg Number* parameter from the AIDC-crd-start-apdu;

b) invoke an AIDC-crd-start indication primitive with the extracted parameters in a) above as the AIDC-crd-start indication *Msg Number* parameter and the AIDC-crd-start indication primitive *User Data* parameter;

c) if running, stop timer t_{1IN};

d) if running, stop the timer t_{2R};

e) if in the NOTIFY state, then stop the timer t_{1NC};

f) if in the COORDINATED state, then:

1) if running, stop timer t_{2CT};

2) if running, stop the timer t_{1CT};

g) if in the TRANSFERRED state, then stop the timer t_{TE}; and

h) set the variable vr_1 =back and vr_2 = *Msg Number*.

3.2.6.1.9 AIDC-crd-end Request Primitive

3.2.6.1.9.1 When Invoked

3.2.6.1.9.1.1 It shall be valid to invoke the AIDC-crd-end request primitive when the AIDC-ASE protocol machine is in the NEGOTIATING state, or the RE-NEGOTIATING state, or the BACKWARD COORDINATING state.

3.2.6.1.9.2 Action Upon Invocation

3.2.6.1.9.2.1 Upon the receipt of an AIDC-crd-end request primitive the AIDC-ASE shall:

a) create an AIDC-crd-end-apdu based on the *User Data* parameter, the *Result* parameter and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-crd-end-apdu as the *AIDC Data* parameter;

c) start the timer t_C; and

d) if in the NEGOTIATING state, then

1) set the variable vs_1 = coord-end;

2) if the AIDC-crd-end *Result* parameter has the value “accept” then set the variable vse=accept; and

3) if the AIDC-crd-end *Result* parameter has the value “reject” then set the variable vse=reject.

e) if in the RE-NEGOTIATING state, set the variable vs_1 = coord-end;

f) if in the BACKWARD COORDINATING state, set the variable vs_1 = back-end; and

g) set the variable vs_2 = *Msg Number*.

3.2.6.1.10 AIDC-DATA Indication Primitive with an AIDC-crd-end-apdu

3.2.6.1.10.1 When Invoked

3.2.6.1.10.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-crd-end-apdu when the AIDC-ASE protocol machine is in the NEGOTIATING state, or the RE-NEGOTIATING state, or the BACKWARD COORDINATING state.

3.2.6.1.10.2 Action Upon Invocation

3.2.6.1.10.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-crd-end-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, the *Result* parameter and the *Msg Number* parameter from the AIDC-crd-end-apdu;

b) invoke an AIDC-crd-end indication primitive with the extracted parameters in a) above as the AIDC-crd-end indication *User Data* parameter, the AIDC-crd-end indication *Result* parameter and the AIDC-crd-end indication primitive *Msg Number* parameter respectively;

c) if running, stop the timer t_{1R};

d) if running, stop the timer t_{2R};

e) if running, stop the timer t_S; and

f) if in the NEGOTIATING state, then:

1) set the variable vr_1 = coord-end;

2) if the AIDC-crd-end *Result* parameter has the value “accept” then set the variable vre = accept; and

g) if in the RE-NEGOTIATING state, set the variable vs_1 = coord-end;

h) if in the BACKWARD COORDINATING state, set the variable vr_1 = back-end; and

i) set the variable vr_2 = *Msg Number*.

3.2.6.1.11 AIDC-crd-ngtt Request Primitive

3.2.6.1.11.1 When Invoked

3.2.6.1.11.1.1 It shall be valid to invoke the AIDC-crd-ngtt request primitive when the AIDC-ASE protocol machine is in the NEGOTIATING state, or the RE-NEGOTIATING, or the BACKWARD-COORDINATING state.

3.2.6.1.11.2 Action Upon Invocation

3.2.6.1.11.2.1 Upon the receipt of an AIDC-crd-ngtt request primitive the AIDC-ASE shall:

a) create an AIDC-crd-ngtt-apdu based on the *User Data* parameter, and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-crd-ngtt-apdu as the *AIDC Data* parameter

c) start the timer t_C; and

d) set the variable vs_1 = coord-negot, and vs_2 = *Msg Number*.

3.2.6.1.12 AIDC-DATA Indication Primitive with an AIDC-crd-ngtt-apdu

3.2.6.1.12.1 When Invoked

3.2.6.1.12.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-crd-ngtt-apdu when the AIDC-ASE protocol machine is in the NEGOTIATING state, or the RE-NEGOTIATING state, or the BACKWARD-COORDINATING state.

3.2.6.1.12.2 Action Upon Invocation

3.2.6.1.12.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-crd-ngtt-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, and the *Msg Number* parameter from the AIDC-crd-ngtt-apdu; and

b) invoke an AIDC-crd-ngtt indication primitive with the extracted parameters in a) above as the AIDC-crd-ngtt indication *User Data* parameter, and the AIDC-crd-ngtt indication primitive *Msg Number* parameter respectively;

c) if running, stop the timer t_{1R};

d) if running, stop the timer t_{2R};

e) if running, stop the timer t_S; and

f) set the variable vr_1 = coord-negot and vr_2 = *Msg Number*.

3.2.6.1.13 AIDC-crd-stndby Request Primitive

3.2.6.1.13.1 When Invoked

3.2.6.1.13.1.1 It shall be valid to invoke the AIDC-crd-stndby request primitive when the AIDC-ASE protocol machine is in the NEGOTIATING state, or the RE-NEGOTIATING state, or the BACKWARD-COORDINATING state.

3.2.6.1.13.2 Action Upon Invocation

3.2.6.1.13.2.1 Upon the receipt of an AIDC-crd-stndby request primitive the AIDC-ASE shall:

a) create an AIDC-crd-stndby-apdu based on the *User Data* parameter, and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-crd-stndby-apdu as the *AIDC Data* parameter;

c) start the timer t_C; and

d) set the variable vs_1 = coord-stndby and vs_2 = *Msg Number*.

3.2.6.1.14 AIDC-DATA Indication Primitive with an AIDC-crd-stndby-apdu

3.2.6.1.14.1 When Invoked

3.2.6.1.14.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-crd-stndby-apdu when the AIDC-ASE protocol machine is in the NEGOTIATING state, or the RE-NEGOTIATING state, or the BACKWARD-COORDINATING state.

3.2.6.1.14.2 Action Upon Invocation

3.2.6.1.14.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-crd-stndby-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, and the *Msg Number* parameter from the AIDC-crd-stndby-apdu;

b) invoke an AIDC-crd-stndby indication primitive with the extracted parameters in a) above as the AIDC-crd-stndby indication *User Data* parameter, and the AIDC-crd-stndby indication primitive *Msg Number* parameter respectively;

c) if running, stop the timer t_{1R};

d) if running, stop the timer t_{2R}; and

e) set the variable vs_1 = coord-stndby and vr_2 = *Msg Number*.

3.2.6.1.15 AIDC-tfr-init Request Primitive

3.2.6.1.15.1 When Invoked

3.2.6.1.15.1.1 It shall be valid to invoke the AIDC-tfr-init request primitive when the AIDC-ASE protocol machine is in the COORDINATED state.

3.2.6.1.15.2 Action Upon Invocation

3.2.6.1.15.2.1 Upon the receipt of an AIDC-tfr-init request primitive the AIDC-ASE shall:

a) create an AIDC-tfr-init-apdu based on the *User Data* parameter, and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-tfr-init-apdu as the *AIDC Data* parameter;

c) if running, stop the timer t_{1CT};

d) if running, stop the timer t_{2CT};

e) start the timer t_C; and

f) set the variable vs_1 = trns-init, and vs_2 = *Msg Number*.

3.2.6.1.16 AIDC-DATA Indication Primitive with an AIDC-tfr-init-apdu

3.2.6.1.16.1 When Invoked

3.2.6.1.16.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-tfr-init-apdu when the AIDC-ASE protocol machine is in the COORDINATED state.

3.2.6.1.16.2 Action Upon Invocation

3.2.6.1.16.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-tfr-init-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, and the *Msg Number* parameter from the AIDC-tfr-init-apdu;

b) invoke an AIDC-tfr-init indication primitive with the extracted parameters in a) above as the AIDC-tfr-init indication *User Data* parameter, and the AIDC-tfr-init indication primitive *Msg Number* parameter respectively;

c) if running, stop the timer t_{1CT};

d) if running, stop the timer t_{2CT}; and

e) set the variable vr_1 = trns-init and vr_2 = *Msg Number*.

3.2.6.1.17 AIDC-tfr-rqst Request Primitive

3.2.6.1.17.1 When Invoked

3.2.6.1.17.1.1 It shall be valid to invoke the AIDC-tfr-rqst request primitive when the AIDC-ASE protocol machine is in the COORDINATED state.

3.2.6.1.17.2 Action Upon Invocation

3.2.6.1.17.2.1 Upon the receipt of an AIDC-tfr-rqst request primitive the AIDC-ASE shall:

a) create an AIDC-tfr-rqst-apdu based on the *User Data* parameter, and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-tfr-rqst-apdu as the *AIDC Data* parameter;

c) start the timer t_C; and

d) set the variable vs_2 = *Msg Number*.

3.2.6.1.18 AIDC-DATA Indication Primitive with an AIDC-tfr-rqst-apdu

3.2.6.1.18.1 When Invoked

3.2.6.1.18.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-tfr-rqst-apdu when the AIDC-ASE protocol machine is in the COORDINATED state.

3.2.6.1.18.2 Action Upon Invocation

3.2.6.1.18.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-tfr-rqst-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, and the *Msg Number* parameter from the AIDC-tfr-rqst-apdu;

b) invoke an AIDC-tfr-rqst indication primitive with the extracted parameters in a) above as the AIDC-tfr-rqst indication *User Data* parameter, and the AIDC-tfr-rqst indication primitive *Msg Number* parameter respectively; and

c) set the variable vr_2 = *Msg Number*.

3.2.6.1.19 AIDC-tfr-prpsl Request Primitive

3.2.6.1.19.1 When Invoked

3.2.6.1.19.1.1 It shall be valid to invoke the AIDC-tfr-prpsl request primitive when the AIDC-ASE protocol machine is in the PRE-TRANSFER state.

3.2.6.1.19.2 Action Upon Invocation

3.2.6.1.19.2.1 Upon the receipt of an AIDC-tfr-prpsl request primitive the AIDC-ASE shall:

a) create an AIDC-tfr-prpsl-apdu based on the *User Data* parameter, and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-tfr-prpsl-apdu as the *AIDC Data* parameter;

c) start the timer t_C;

d) if P1, then set the variable vs_1 = trns-prpsl; and

e) set the variable vs_2 = *Msg Number*.

3.2.6.1.20 AIDC-DATA Indication Primitive with an AIDC-tfr-prpsl-apdu

3.2.6.1.20.1 When Invoked

3.2.6.1.20.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-tfr-prpsl-apdu when the AIDC-ASE protocol machine is in the PRE-TRANSFER state.

3.2.6.1.20.2 Action Upon Invocation

3.2.6.1.20.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-tfr-prpsl-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, and the *Msg Number* parameter from the AIDC-tfr-prpsl-apdu;

b) invoke an AIDC-tfr-prpsl indication primitive with the extracted parameters in a) above as the AIDC-tfr-prpsl indication *User Data* parameter, and the AIDC-tfr-prpsl indication primitive *Msg Number* parameter respectively;

c) if P1, then set the variable vr_1 = trns-prpsl; and

d) set the variable vr_2 = *Msg Number*.

3.2.6.1.21 AIDC-tfr-accept Request Primitive

3.2.6.1.21.1 When Invoked

3.2.6.1.21.1.1 It shall be valid to invoke the AIDC-tfr-accept request primitive if the predicate P1 is true and when the AIDC-ASE protocol machine is in the PRE-TRANSFER state.

3.2.6.1.21.2 Action Upon Invocation

3.2.6.1.21.2.1 Upon the receipt of an AIDC-tfr-accept request primitive the AIDC-ASE shall:

a) create an AIDC-tfr-accept-apdu based on the *User Data* parameter, and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-tfr-accept-apdu as the *AIDC Data* parameter;

c) start the timer t_C; and

d) set the variable vs_2 = *Msg Number*.

3.2.6.1.22 AIDC-DATA Indication Primitive with an AIDC-tfr-accept-apdu

3.2.6.1.22.1 When Invoked

3.2.6.1.22.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-tfr-accept-apdu if the predicate P1 is true and when the AIDC-ASE protocol machine is in the PRE-TRANSFER state.

3.2.6.1.22.2 Action Upon Invocation

3.2.6.1.22.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-tfr-accept-apdu the AIDC-ASE shall:

a) if running stop timer t_{1R};

b) extract the *User Data* parameter, and the *Msg Number* parameter from the AIDC-tfr-accept-apdu;

c) invoke an AIDC-tfr-accept indication primitive with the extracted parameters in a) above as the AIDC-tfr-accept indication *User Data* parameter, and the AIDC-tfr-accept indication primitive *Msg Number* parameter respectively; and

d) set the variable vr_2 = *Msg Number*.

3.2.6.1.23 AIDC-tfr-comm Request Primitive

3.2.6.1.23.1 When Invoked

3.2.6.1.23.1.1 It shall be valid to invoke the AIDC-tfr-comm request primitive when the AIDC-ASE protocol machine is in the PRE-TRANSFER state.

3.2.6.1.23.2 Action Upon Invocation

3.2.6.1.23.2.1 Upon the receipt of an AIDC-tfr-comm request primitive the AIDC-ASE shall:

a) create an AIDC-tfr-comm-apdu based on the *User Data* parameter, and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-tfr-comm-apdu as the *AIDC Data* parameter;

c) start the timer t_C; and

d) set the variable vs_1 = trns-comm and vs_2 = *Msg Number*.

3.2.6.1.24 AIDC-DATA Indication Primitive with an AIDC-tfr-comm-apdu

3.2.6.1.24.1 When Invoked

3.2.6.1.24.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-tfr-comm-apdu when the AIDC-ASE protocol machine is in the PRE-TRANSFER state.

3.2.6.1.24.2 Action Upon Invocation

3.2.6.1.24.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-tfr-comm-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, and the *Msg Number* parameter from the AIDC-tfr-comm-apdu;

b) invoke an AIDC-tfr-comm indication primitive with the extracted parameters in a) above as the AIDC-tfr-comm indication *User Data* parameter, and the AIDC-tfr-comm indication primitive *Msg Number* parameter respectively;

c) if running, stop the timer t_{2R};

d) if running, stop the timer t_{3R}; and

e) set the variable vr_1 = trns-comm and vr_2 = *Msg Number*.

3.2.6.1.25 AIDC-tfr-comm-assm Request Primitive

3.2.6.1.25.1 When Invoked

3.2.6.1.25.1.1 It shall be valid to invoke the AIDC-tfr-comm-assm request primitive when the AIDC-ASE protocol machine is in the PRE-TRANSFER state, or the TRANSFERRING state.

3.2.6.1.25.2 Action Upon Invocation

3.2.6.1.25.2.1 Upon the receipt of an AIDC-tfr-comm-assm request primitive the AIDC-ASE shall:

a) create an AIDC-tfr-comm-assm-apdu based on the *User Data* parameter, and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-tfr-comm-assm-apdu as the *AIDC Data* parameter;

c) start the timer t_C;

d) if running, stop the timer t_{3R}; and

e) set the variable vs_1 = trns-assm and vs_2 = *Msg Number*.

3.2.6.1.26 AIDC-DATA Indication Primitive with an AIDC-tfr-comm-assm-apdu

3.2.6.1.26.1 When Invoked

3.2.6.1.26.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-tfr-comm-assm-apdu when the AIDC-ASE protocol machine is in the PRE-TRANSFER state, or the TRANSFERRING state.

3.2.6.1.26.2 Action Upon Invocation

3.2.6.1.26.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-tfr-comm-assm-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, and the *Msg Number* parameter from the AIDC-tfr-comm-assm-apdu;

b) invoke an AIDC-tfr-comm-assm indication primitive with the extracted parameters in a) above as the AIDC-tfr-comm-assm indication *User Data* parameter, and the AIDC-tfr-comm-assm indication primitive *Msg Number* parameter respectively;

c) if running, stop the timer t_{1R};

d) if running, stop the timer t_{2R}; and

e) set the variable vr_1 = trns-assm and vr_2 = *Msg Number*.

3.2.6.1.27 AIDC-tfr-cntrl Request Primitive

3.2.6.1.27.1 When Invoked

3.2.6.1.27.1.1 It shall be valid to invoke the AIDC-tfr-cntrl request primitive when the AIDC-ASE protocol machine is in the COORDINATED state.

3.2.6.1.27.2 Action Upon Invocation

3.2.6.1.27.2.1 Upon the receipt of an AIDC-tfr-cntrl request primitive the AIDC-ASE shall:

a) create an AIDC-tfr-cntrl-req-apdu based on the *User Data* parameter, and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-tfr-cntrl-req-apdu as the *AIDC Data* parameter;

c) if running, stop the timer t_{1CT};

d) if running, stop the timer t_{2CT};

e) start the timer t_C; and

f) set the variable vs_1 = trns-start vs_2 = *Msg Number*.

3.2.6.1.28 AIDC-DATA Indication Primitive with an AIDC-tfr-cntrl-req-apdu

3.2.6.1.28.1 When Invoked

3.2.6.1.28.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-tfr-cntrl-Req-apdu when the AIDC-ASE protocol machine is in the COORDINATED state.

3.2.6.1.28.2 Action Upon Invocation

3.2.6.1.28.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-tfr-cntrl-req-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, and the *Msg Number* parameter from the AIDC-tfr-cntrl-req-apdu;

b) invoke an AIDC-tfr-cntrl indication primitive with the extracted parameters in a) above as the AIDC-tfr-cntrl indication *User Data* parameter, and the AIDC-tfr-cntrl indication primitive *Msg Number* parameter respectively;

c) if running, stop the timer t_{1CT};

d) if running, stop the timer t_{2CT};

e) if running, stop the timer t_{2R}; and

f) set the variable $vr_1 = t_{rns}$-start and vr_2 = *Msg Number*.

3.2.6.1.29 AIDC-tfr-cntrl Response Primitive

3.2.6.1.29.1 When Invoked

3.2.6.1.29.1.1 It shall be valid to invoke the AIDC-tfr-cntrl response primitive when the AIDC-ASE protocol machine is in the TRANSFERRING state.

3.2.6.1.29.2 Action Upon Invocation

3.2.6.1.29.2.1 Upon the receipt of an AIDC-tfr-cntrl response primitive the AIDC-ASE shall:

a) create an AIDC-tfr-cntrl-rsp-apdu based on the *User Data* parameter, the *Result* parameter and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-tfr-cntrl-rsp-apdu as the *AIDC Data* parameter;

c) start the timer t_C;

d) if the *Result* parameter has the abstract value:

1) "accepted"; set vs_1 = trns-accept;

2) "rejected"; set vs_1 = trns-reject.

e) set the variable vs_2 = *Msg Number*;

3.2.6.1.30 AIDC-DATA Indication with an AIDC-tfr-cntrl-rsp-apdu

3.2.6.1.30.1 When Invoked

3.2.6.1.30.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-tfr-cntrl-rsp-apdu when the AIDC-ASE protocol machine is in the TRANSFERRING state.

3.2.6.1.30.2 Action Upon Invocation

3.2.6.1.30.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-tfr-cntrl-rsp-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, the *Result* parameter, and the *Msg Number* parameter from the AIDC-tfr-cntrl-rsp-apdu;

b) invoke an AIDC-tfr-cntrl confirmation primitive with the extracted parameters in a) above as the AIDC-tfr-cntrl confirmation *User Data* parameter, and the AIDC-tfr-cntrl indication primitive *Msg Number* parameter respectively;

c) if running, stop the timer t_{1R};

d) if running, stop the timer t_{2R};

e) if the *Result* parameter has the abstract value:

1) "accepted"; set vr_1 = trns-accept;

2) "rejected"; set vr_1 = trns-reject; and

f) set the variable vr_2 = *Msg Number*.

3.2.6.1.31 AIDC-inf-tfr Request Primitive

3.2.6.1.31.1 When Invoked

3.2.6.1.31.1.1 It shall be valid to invoke the AIDC-inf-tfr request primitive when the AIDC-ASE protocol machine is in the IDLE state, or in the NOTIFY state, or the NEGOTIATING state, or the RE-NEGOTIATING state, or the COORDINATED state, or the PRE-TRANSFER state, or the TRANSFERRING state, or the TRANSFERRED state, or the BACKWARD-COORDINATING state.

3.2.6.1.31.2 Action Upon Invocation

3.2.6.1.31.2.1 Upon the receipt of an AIDC-inf-tfr request primitive the AIDC-ASE shall:

a) create an AIDC-inf-tfr-apdu based on the *User Data* parameter, and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-info-tfr-apdu as the *AIDC Data* parameter;

c) if running, stop the timer t_{2IN};

d) start the timer t_C; and

e) set the variables vs_1 = info-trans and vs_2 = *Msg Number*

3.2.6.1.32 AIDC-DATA Indication Primitive with an AIDC-inf-tfr-apdu

3.2.6.1.32.1 When Invoked

3.2.6.1.32.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-inf-tfr-apdu when the AIDC-ASE protocol machine is in the IDLE state, or the NOTIFY state, or the NEGOTIATING state, or the RE-NEGOTIATING state, or the COORDINATED state, or the PRE-TRANSFER state, or the TRANSFERRING state, or the TRANSFERRED state or the BACKWARD-COORDINATING state.

3.2.6.1.32.2 Action Upon Invocation

3.2.6.1.32.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-inf-tfr-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, and the *Msg Number* parameter from the AIDC-inf-tfr-apdu;

b) invoke an AIDC-inf-tfr indication primitive with the extracted parameters in a) above as the AIDC-inf-tfr indication *User Data* parameter, and the AIDC-inf-tfr indication primitive *Msg Number* parameter respectively;

c) if running, stop the timer t_{1IN}; and

d) set the variables vr_1 = info-trans and vr_2 = *Msg Number*.

3.2.6.1.33 AIDC-ucf Request Primitive

3.2.6.1.33.1 When Invoked

3.2.6.1.33.1.1 It shall be valid to invoke the AIDC-ucf request primitive when the AIDC-ASE protocol machine is in the IDLE state, or the NOTIFY state, or the NEGOTIATING state, or the RE-NEGOTIATING state, or the COORDINATED state, or the PRE-TRANSFER state, or the TRANSFERRING state, or the TRANSFERRED state, or the BACKWARD-COORDINATING state.

3.2.6.1.33.2 Action Upon Receipt

3.2.6.1.33.2.1 Upon the receipt of an AIDC-ucf request primitive, the AIDC-ASE shall:

a) create an AIDC-ucf-apdu based on the *Result* parameter, the *Reason* parameter if present, and the *Reference ID* parameter;

b) if vr_2 = *Reference ID*, then:

1) invoke an AIDC-DATA request primitive with the AIDC-ucf-apdu as the *User Data* parameter;

2) if the AIDC-ucf request primitive *Result* parameter has the abstract value "accepted" then:

i) if in the IDLE state:

A) if the variable vr_1 = notify, then;

I) start the timer t_{1NC}; and

II) enter the NOTIFY state.

B) if the variable vr_1 = coord-start, then enter the NEGOTIATING state.

C) if the variable vr_1 = info-trans, then start the timer t_{1NC};

ii) if in the NOTIFY state:

A) if the variable vr_1 = notify, then;

I) start the timer t_{1NC}; and

II) enter the NOTIFY state.

B) if the variable vr_1 = coord-start, then enter the NEGOTIATING state;

C) if the variable vr_1 = end, then enter the IDLE state

iii) if in the NEGOTIATING state:

A) if the variable vr_1 = coord-end, then:

I) if variable vre = accept, then:

(a) start the timer t_{1CT};

(b) set the variable vre = NULL; and

(c) enter the COORDINATED state.

II) if variable vre = reject, then:

(a) start the timer t_{1NC};

(b) set the variable vre = NULL; and

(c) enter the NOTIFY state.

B) if the variable vr_1 = coord-stndby, then start the timer t_S;

C) if the variable vr_1 = end, then enter the IDLE state.

iv) if in the COORDINATED state:

A) if the variable vr1 = coord-start, then:

I) enter RE-NEGOTIATING state.

B) if the variable vr_1 = trns-init, then:

I) start the timer t_{3R}; and

II) enter PRE-TRANSFER state.

C) if the variable vr_1 = trns-start, then enter the TRANSFERRING state.

D) if the variable vr_1 = end, then:

I) stop all timers; and

II) enter the IDLE state.

v) if in the RE-NEGOTIATING state:

A) if the variable vr_1 = coord-end, then:

I) start the timer t_{1CT}; and

II) enter the COORDINATED state.

B) if the variable vr_1 = coord-stnby, then start the timer t_S;

C) if the variable vr_1 = end, then enter the IDLE state.

vi) if in the PRE-TRANSFER state:

A) if the variable vr_1 = trns-comm, then enter the TRANSFERRING state.

B) if the variable vr_1 = trns-assm, then:

I) start the timer t_{TE}; and

II) enter the TRANSFERRED state.

vii) if in the TRANSFERRING state:

A) if the variable vr_1 = trns-assm, then:

I) start the timer t_{TE}; and

II) enter the TRANSFERRED state.

B) if the variable vr_1 = trns-accept, then:

I) start the timer t_{TE}; and

II) enter the TRANSFERRED state.

C) if the variable vr_1 = trns-reject, then:

I) start the timer t_{1CT}; and

II) enter the COORDINATED state.

viii) if in the TRANSFERRED state:

A) if the variable vr_1 = back, then enter the BACKWARD COORDINATING state.

B) if the variable vr_1 = end, then enter the IDLE state.

ix) if in the BACKWARD-COORDINATING state:

A) if the variable vr_1 = back-end, then enter the TRANSFERRED state.

3) if the AIDC-ucf request primitive *Result* parameter has the abstract value "rejected", then:

i) if in the IDLE state, then:

A) if vr_1 = notify start t_{1NC};

B) if vr_1 = coord-start timer t_{1NC};

C) if vr_1 = info-trans start timer t_{1IN};

D) set the variable vr_1 = NULL, vr_2 = NULL and vre = NULL; and

E) remain in the current state.

ii) if in the NOTIFY state, then:

A) start timer t_{1NC};

B) set the variable vr_1 = NULL, vr_2 = NULL, and vre=NULL; and

C) remain in the same state.

iii) if in the COORDINATED state then:

A) start t_{1CT};

B) set the variable vr_1 = NULL, vr_2 = NULL, and vre=NULL; and

C) remain in the current state.

iv) if in any other state, then:

A) if the variable vr_1!=NULL, then start the timer t_{2R};

B) set the variable vr_1 = NULL, vr_2 = NULL, and vre=NULL; and

C) remain in the current state.

c) if the variable vr_2!= *Reference ID* then:

1) invoke an AIDC-pvd-abrt indication;

2) invoke an AIDC-ABORT request;

3) stop all timers; and

4) enter the IDLE state.

3.2.6.1.34 AIDC-DATA Indication Primitive with an AIDC-ucf-apdu

3.2.6.1.34.1 When Invoked

3.2.6.1.34.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-ucf-apdu when the AIDC-ASE protocol machine is in the IDLE state, or the NOTIFY state, or the NEGOTIATING state, or the RE-NEGOTIATING state, or the COORDINATED state, or the PRE-TRANSFER state, or the TRANSFERRING state, or the TRANSFERRED state, or the BACKWARD-COORDINATED state.

3.2.6.1.34.2 Action Upon Invocation

3.2.6.1.34.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-ucf-apdu the AIDC-ASE shall:

a) stop timer t_c;

b) extract the *Result* parameter, the *Reason* parameter if present, and the *Reference ID* parameter from the AIDC-ucf-apdu;

c) if the variable vs_2 = *Reference ID*, invoke an AIDC-ucf indication primitive with the extracted parameters in a) above as the AIDC-ucf indication primitive *Result* parameter, the AIDC-ucf indication primitive *Reason* parameter, and the AIDC-ucf indication primitive *Reference ID* parameter;

1) if the *Response* parameter of the AIDC-ucf request primitive has the abstract value "accepted":

i) if in the IDLE state, or the NOTIFY state:

A) if the variable vs_1 = notify, then:

I) start the timer t_{2NC}; and

II) enter the NOTIFY state.

B) if the variable vs_1 = coord-start, then:

I) start the timer t_{1R}; and

II) enter the NEGOTIATING state.

C) if the variable vs_1 = info-trans, then start the timer t_{2IN}.

D) if the variable vs_1 = end, then enter IDLE state;

ii) if in the NEGOTIATING state:

A) if the variable vs_1 = coord-negot, then start the timer t_{1R}.

B) if the variable vs_1 = coord-end, then:

I) if variable vse = accept, then:

(a) start the timer t_{2CT}; and

(b) set the variable vse = NULL; and

(c) enter the COORDINATED state.

II) if variable vse = reject, then:

(a) start the timer t_{2NC};

(b) set the variable vse = NULL; and

(c) enter the NOTIFY state.

C) if the variable vs_1 = end, then enter IDLE state.

iii) if in the COORDINATED state:

A) if the variable vs_1 = coord-start, then:

I) start the timer t_{1R}; and

II) enter the RE-NEGOTIATING state.

B) if the variable vs_1 = trns-init, then enter the PRE-TRANSFER state.

C) if the variable vs_1 = trns-start, then enter the TRANSFERRING state.

D) if the variable vs_1 = end, then:

I) stop all timers;

II) start timer t_{2NC}; and

III) enter the IDLE state.

iv) if in the RE-NEGOTIATING state:

A) if the variable vs1 = coord-negot, then start the timer t_{1R};

B) if the variable vs1 = coord-end, then:

I) start the timer t_{2CT};

II) enter the COORDINATED state.

C) if the variable vs_1 = end, then enter IDLE state.

v) if in the PRE-TRANSFER state:

A) if the variable vs1 = trns-prpsl, then:

I) start timer t_{1R}; and

II) remain in the current state.

B) if the variable vs_1 = trns-comm, then enter the TRANSFERRING state.

C) if the variable vs_1 = trns-assm, then enter the TRANSFERRED state.

vi) if in the TRANSFERRING state:

A) if the variable vs_1 = trns-assm, then enter the TRANSFERRED state.

B) if the variable vs_1 = trns-accept, then enter the TRANSFERRED state.

C) if the variable vs_1 = trns-reject, then:

I) start the timer t_{2CT}; and

II) enter the COORDINATED state.

vii) if in the TRANSFERRED state:

A) if the variable vs_1 = back, then enter the state BACKWARD COORDINATING

B) if the variable vs_1 = end, then enter IDLE state.

viii) if in the BACKWARD-COORDINATING state:

A) if the variable vs_1 = back-end, then enter the TRANSFERRED state.

2) if the *Response* parameter of the AIDC-ucf request primitive has the abstract value: rejected, then:

i) if in the IDLE state then:

A) if vr_1 = notify start timer t_{2NC};

B) if vr_1 = coord-start start timer t_{2NC};

C) if vr_1 = info-trans start timer t_{2IN};

D) set the variable vs_1 = NULL, vs_2 = NULL and vse=NULL; and

E) remain in the current state.

ii) if in the NOTIFY state then:

A) start timer t_{2NC};

B) set the variable vs_1 = NULL; vs_2 = NULL, and vse=NULL; and

C) remain in the current state.

iii) if in the COORDINATED state then:

A) start timer t_{2CT};

B) set the variable vs_1 = NULL, vs_2 = NULL, and vse=NULL; and

C) remain in the current state.

iv) if in any other state then:

A) stop all timers;

B) set the variable vs_1 = NULL, vs_2 = NULL, and vse=NULL; and

C) remain in the current state.

d) if the variable vs_2 *Reference ID*, then:

1) invoke an AIDC-pvd-abrt indication;

2) invoke an AIDC-ABORT request;

3) stop all timers; and

4) enter the IDLE state.

3.2.6.1.35 AIDC-end Request Primitive

3.2.6.1.35.1 When Invoked

3.2.6.1.35.1.1 It shall be valid to invoke the AIDC-end request primitive when the AIDC-ASE protocol machine is in the NOTIFY state, or the COORDINATED state, or the TRANSFERRED state.

3.2.6.1.35.2 Action Upon Invocation

3.2.6.1.35.2.1 Upon the receipt of an AIDC-end request primitive the AIDC-ASE shall:

a) create an AIDC-end-apdu based on the *User Data* parameter, if present, and the *Msg Number* parameter;

b) invoke a AIDC-DATA request primitive with the AIDC-end-apdu as the *AIDC Data* parameter;

c) if in the NOTIFY state, stop the timer t_{2NC};

d) if in the COORDINATED state, then:

1) if running, stop the timer t_{1CT};

2) if running, stop the timer t_{2CT};

e) start the timer t_C; and

f) set the variable vs_1 = end and vs_2 = *Msg Number*.

3.2.6.1.36 AIDC-DATA Request with an AIDC-end-apdu

3.2.6.1.36.1 When Invoked

3.2.6.1.36.1.1 It shall be valid to invoke the AIDC-DATA indication primitive with an AIDC-end-apdu when the AIDC-ASE protocol machine is in the IDLE state, or the NOTIFY state, or the COORDINATED state, or the TRANSFERRED state.

3.2.6.1.36.2 Action Upon Invocation

3.2.6.1.36.2.1 Upon the receipt of an AIDC-DATA indication primitive with an AIDC-end-apdu the AIDC-ASE shall:

a) extract the *User Data* parameter, if present, and the *Msg Number* parameter from the AIDC-end-apdu;

b) invoke an AIDC-end indication primitive with the extracted parameters in a) above as the AIDC-end indication *User Data* parameter, and the AIDC-end indication primitive *Msg Number* parameter respectively;

c) if in the IDLE state, stop the timer t_{2R};

d) if in the NOTIFY state, stop the timer t_{1NC};

e) if in the COORDINATED state:

1) if running, stop the timer t_{1CT};

2) if running, stop the timer t_{2CT};

f) if in the TRANSFERRED state, stop the timer t_{TE}; and

g) set the variable vr_1 = end and vr_2 = *Msg Number*.

3.2.6.1.37 AIDC-usr-abrt Request Primitive

3.2.6.1.37.1 When Invoked

3.2.6.1.37.1.1 It shall be valid to invoke the AIDC-usr-abrt request primitive when the AIDC-ASE protocol machine is in the NOTIFY state, or the NEGOTIATING state, or the RE-NEGOTIATING state, or the COORDINATED state, or the PRE-TRANSFER state, or the TRANSFERRING state, or the TRANSFERRED state, or the BACKWARD-COORDINATING state.

3.2.6.1.37.2 Action Upon Invocation

3.2.6.1.37.2.1 Upon the receipt of an AIDC-usr-abrt request primitive the AIDC-ASE shall:

a) invoke an AIDC-ABORT request primitive;

b) stop all timers; and

c) enter the IDLE state.

3.2.6.1.38 AIDC-ABORT Indication Primitive

3.2.6.1.38.1 When Invoked

3.2.6.1.38.1.1 It shall be valid to invoke the AIDC-ABORT indication primitive when the AIDC-ASE protocol machine is in the NOTIFY state, or the NEGOTIATING state, or the RE-NEGOTIATING state, or the COORDINATED state, or the PRE-TRANSFER state, or the TRANSFERRING state, or the TRANSFERRED state, or the BACKWARD-COORDINATING state.

3.2.6.1.38.2 Action Upon Invocation

3.2.6.1.38.2.1 Upon the receipt of a AIDC-ABORT indication primitive the AIDC-ASE shall:

a) invoke an AIDC-usr-abrt indication primitive;

b) stop all timers; and

c) enter the IDLE state.

3.2.6.1.39 AIDC-P-ABORT indication

3.2.6.1.39.1 When Invoked

3.2.6.1.39.1.1 It shall be valid to invoke the AIDC-P-ABORT indication primitive when the AIDC-ASE protocol machine is in the NOTIFY state, or the NEGOTIATING state, or the RE-NEGOTIATING state, or the COORDINATED state, or the PRE-TRANSFER state, or the TRANSFERRING state, or the TRANSFERRED state, or the BACKWARD-COORDINATING state.

3.2.6.1.39.2 Action Upon Invocation

3.2.6.1.39.2.1 Upon the receipt of a AIDC-P-ABORT indication primitive the AIDC-ASE shall:

a) invoke an AIDC-pvd-abrt indication primitive with the *AbortReason* parameter set to the abstract value: communications service failure;

b) stop all timers; and

c) enter the IDLE state.

3.2.6.2 Exception Handling

3.2.6.2.1 Timer Expiration

3.2.6.2.1.1 If a timer expires the AIDC-ASE shall:

a) interrupt any current activity;

b) invoke AIDC-ABORT request primitive; and

c) invoke an AIDC-pvd-abrt indication primitive with the AIDC-pvd-abrt *AbortReason* parameter set to the abstract value: timer expired.

3.2.6.2.2 Irrecoverable System Error

3.2.6.2.2.1 If an AIDC-ASE has an irrecoverable system error, the AIDC-ASE shall:

a) interrupt any current activity;

b) invoke AIDC-ABORT request primitive; and

c) invoke an AIDC-pvd-abrt indication with the AIDC-pvd-abrt *AbortReason* parameter set to the abstract value: undefined-error.

3.2.6.2.3 Invalid APDU

3.2.6.2.3.1 If a APDU received is determined to be invalid, the AIDC-ASE shall:

a) invoke an AIDC-ABORT request primitive; and

b) invoke an AIDC-pvd-abrt indication primitive with the *AbortReason* parameter set to the abstract value: invalid APDU; and

c) enter the IDLE state.

3.2.6.3 AIDC Application Timers

Note.— Table 3.2.6-1 lists the time constraints related to the AIDC application. Each time constraint requires a timer to be set in the AIDC protocol machine.

3.2.6.3.1 An AIDC-ASE shall measure the time between the initial event and the corresponding final event for each of the timers in Table 3.2.6-1.

3.2.6.3.1.1 An implementation of the AIDC-ASE shall provide a means for configuring each of the timer values in Table 3.2.6.1.

Note.— The exact means used to configure timer values is a local matter.

3.2.6.3.2 If the maximum time is exceeded before the final event has occurred, an AIDC-ASE shall take appropriate action as defined in 3.2.6.2.1.

3.2.6.3.2.1 **Recommendation.—***The actions defined in 3.2.6.2.1 should be taken when the maximum time as indicated in Table 3.2.6-1 has expired.*

Table 3.2.6-1: AIDC-ASE Timers

Purpose	Timer	Timer Value	Timer Start Event	Timer Stop Event (Primitive invocation)
User Confirmation for all AIDC services except Abort services	t_C	2 min	AIDC-DATA Req with AIDC-ASE-apdu	AIDC-DATA Ind with AIDC-ucf-apdu
Monitors time for transition between an initial invocation of the Info-transfer service primitive in the IDLE state and a subsequent Regime	t_{1IN}	see 3.2.6.3.1.1	+ve AIDC-ucf Ind in response to AIDC-inf-tfr Req	•AIDC-nfy Ind •AIDC-crd-start Ind

Purpose	Timer	Timer Value	Timer Start Event	Timer Stop Event (Primitive invocation)
Monitors timer for transition between an initial invocation of the Info-transfer service primitive in the IDLE state and a subsequent Regime	t_{2IN}	see 3.2.6.3.1.1	+ve AIDC-ucf-IND in response to AIDC-inf-tfr-Req	•AIDC-nfy Req •AIDC-crd-start Req
Monitors time for transition between Notifying Regime and Coordinating Regime.	t_{1NC}	see 3.2.6.3.1.1 TBD	+ve AIDC-ucf Req in response to AIDC-nfy Ind	•AIDC-nfy Ind •AIDC-crd-start Ind •AIDC-end Ind
Monitors time for transition between Notifying Regime and Coordinating Regime. Complementary timer for t_{1NC}.	t_{2NC}	see 3.2.6.3.1.1	+ve AIDC-ucf Ind in response to AIDC-nfy Req	•AIDC-nfy Req •AIDC-crd-start Req •AIDC-end Req
Response monitoring timer used with the following services: •AIDC-crd-start •AIDC-crd-ngtt •AIDC-tfr-prpsl (if P1 true) •AIDC-tfr-comm •AIDC-tfr-cntrl	t_{1R}	see 3.2.6.3.1.1	+ve AIDC-ucf Ind in response to: •AIDC-crd-start Req •AIDC-crd-ngtt Req •AIDC-tfr-prpsl Req (if P1 true) •AIDC-tfr-comm Req •AIDC-tfr-cntrl Req	•AIDC-crd-ngtt Ind •AIDC-crd-stndby Ind •AIDC-crd-end Ind •AIDC-tfr-accept Ind •AIDC-tfr-comm-assm Ind •AIDC-tfr-cntrl Cnf
Response monitoring timer after -ve User Confirmation for all services except: •AIDC-tfr-rqst •AIDC-tfr-prpsl •AIDC-tfr-accept •AIDC-info-transfer •AIDC-usr-abrt •AIDC-pvd-abrt	t_{2R}	see 3.2.6.3.1.1	-ve AIDC-ucf Req	AIDC-ASE service Ind primitive except for the following services: •AIDC-tfr-rqst •AIDC-tfr-prpsl •AIDC-tfr-accept •AIDC-inf-tfr •AIDC-usr-abrt •AIDC-pvd-abrt
Standby timer associated with AIDC-crd-stndby service.	t_S	see 3.2.6.3.1.1	+ve AIDC-ucf Req in response to AIDC-crd-stndby Ind	•AIDC-crd-ngtt Ind •AIDC-crd-end Ind

Purpose	Timer	Timer Value	Timer Start Event	Timer Stop Event (Primitive invocation)
Monitors time for transition between Coordinating Regime and Transferring Regime.	t_{1CT}	see 3.2.6.3.1.1	+ve AIDC-ucf Req in response to AIDC-crd-end Ind	•AIDC-tfr-init Ind or Req •AIDC-tfr-cntrl Ind or Req •AIDC-end Ind or Req • AIDC-crd-start Ind or Req
Monitors time for transition between Coordinating Regime and Transferring Regime. Complementary timer for t_{1CT}.	t_{2CT}	see 3.2.6.3.1.1	+ve AIDC-ucf Ind in response to AIDC-crd-end Req	•AIDC-tfr-init Req or Ind •AIDC-tfr-cntrl Req or Ind •AIDC-end Req or Ind • AIDC-crd-start Ind or Req
Response monitoring timer used with the AIDC-tfr-init service	t_{3R}	see 3.2.6.3.1.1	+ve AIDC-ucf Req in response to AIDC-tfr-init Ind	•AIDC-tfr-comm Ind •AIDC-tfr-comm-assm-Req
End timer to monitor time after completion of Transferring Regime to the invocation of the AIDC-end service or commencement of backward coordination.	t_{TE}	see 3.2.6.3.1.1	+ve AIDC-ucf Req in response to AIDC-tfr-comm-assm Ind or AIDC-tfr-cntrl Cnf	•AIDC-crd-start Ind or Req •AIDC-end Ind

3.2.6.4 State Table

3.2.6.4.1 The AIDC-ASE shall behave in accordance with the following state table, which show diagrammatically the state transitions and actions performed by the AIDC-ASE in response to incoming events. Incoming events are shown in the first column of the state table. Each cell in the state table shows:

a) optionally, one or more variables, denoted “vrN”, or “vsN”, where N is either an integer or the character “e”. The variables are defined as required and take on a value within the state table.

b) the new state that the AIDC-ASE shall enter after the action has been performed

c) the action, if any, which the AIDC-ASE shall perform.

3.2.6.4.2 Blank cells indicate error conditions. The error handling shall result in the association being aborted, if one exists, and a notification being given to the AIDC-User.

3.2.6.4.3 In the event of a conflict between the actions implied by the state table and the text in the above, the text shall take precedence.

Note 1.— Variables

vr1-variable holding the last received event type for saving

vr2-variable holding the last received Msg Number for saving

vre-variable holding the last receive Result parameter of the AIDC-crd-end service for saving

vs1-variable holding the last sent event type for saving

vs2-variable holding the last sent Msg Number for saving

vse-variable holding the last send Result parameter of the AIDC-crd-end service for saving

Note 2.— Predicates

c1 - Logical Confirmation result = Accept

c2 - vr2 = Reference ID

c3 - vs2 = Reference ID

c4 - vr1 = notify

c5 - vs1 = notify

c6 - vr1 = coord-start

c7 - vs1 = coord-start

c8 - vr1 = coord-negot

c9 - vs1 = coord-negot

c10 - vr1 = coord-stndby

c11 - vs1 = coord-stndby

c12 - vr1 = coord-end

c13 - vs1 = coord-end

c14 - vr1 = trns-init

c15 - vs1 = trns-init

c16 - vr1 = trns-start

c17 - vs1 = trns-start

c18 - vr1 = trns-accept

c19 - vs1 = trns-accept

c20 - vr1 = trns-reject

c21 - vs1 = trns-reject

c22 - vr1 = trns-comm

c23 - vs1 = trns-comm

c24 - vr1 = trns-assm

c25 - vs1 = trns-assm

c26 - vr1 = back

c27 - vs1 = back

c28 - vr1 = back-end

c29 - vs1 = back-end

c30 - vr1 = end

c31 - vs1 = end

c32 - vr1 = info-trans

c33 - vs1 = info-trans

c34 - AIDC-crd-end Req/Ind Result = accept

c35 - AIDC-crd-end Req/Ind Result = reject

c36 - AIDC-tfr-cntrl Rsp/Cnf Result = accepted

c37 - AIDC-tfr-cntrl Rsp/Cnf Result = rejected

c38 - vr1 = NULL

c39 - vre = accept

c40 - vre = reject

c41 - vse = accept

c42 - vse = reject

c43 - vs1 = trns-prpsl

Note 3.— Where a predicate is shown in the state table with a preceding exclamation mark (!), this indicates that the predicate is not true.

Table 3.2.6-4: State Table

State⇒ Event⇓	**IDLE**	**NOTIFY**	**NEGOTIATING**	**RE-NEGOTIATING**	**COORDINATED**
AIDC-nfy REQ	• vs1= notify • vs2=Msg Number • AIDC-DATA REQ with AIDC-nfy-apdu • start t_C • stop t_{2IN} ⇒ IDLE	• vs1= notify • vs2=Msg Number • AIDC-DATA REQ with AIDC-nfy-apdu • stop t_{2NC} • stop t_{2IN} • start t_C ⇒ NOTIFY			
rcv AIDC-nfy-apdu	• vr1= notify • vr2=Msg Number • AIDC-nfy IND • stop t_{2R} • stop t_{1IN} ⇒ IDLE	• vr1= notify vr2=Msg Number • AIDC-nfy IND • stop t_{1NC} • stop t_{2R} • stop t_{1IN} ⇒ NOTIFY			
AIDC-crd-start REQ	• vs1= back • vs2=Msg Number • start t_C • stop t_{2IN} • AIDC-DATA REQ with AIDC-crd-start-apdu ⇒ IDLE	• vs1= back • vs2=Msg Number • stop t_{2IN} • stop t_{1CT} • start t_C • AIDC-DATA REQ with AIDC-crd-start-apdu ⇒ NOTIFY			• vs1= back • vs2=Msg Number • stop t_{2IN} • stop t_{1CT} • stop t_{2CT} • start t_C • AIDC-DATA REQ with AIDC-crd-start-apdu ⇒ COORDINATED
rcv AIDC-crd-start-apdu	• vr1= back • vr2=Msg Number • AIDC-crd-start IND • stop t_{1IN} • stop t_{2R} ⇒ IDLE	• vr1= back • vr2=Msg Number • AIDC-crd-start IND • stop t_{1NC} • stop t_{1IN} • stop t_{2R} ⇒ NOTIFY			• vr1= back • vr2=Msg Number • AIDC-crd-start IND • stop t_{1CT} • stop t_{2CT} • stop t_{1IN} • stop t_{2R} ⇒ COORDINATED

State⇒ Event⇓	**IDLE**	**NOTIFY**	**NEGOTIATING**	**RE-NEGOTIATING**	**COORDINATED**
AIDC-crd-end REQ			• vs1= coord-end vs2=Msg Number • start t_C • AIDC-DATA REQ with AIDC-crd-end-apdu • if c34 vse=accept • if c35 vse=reject ⇒ NEGOTIATING	• vs1= coord-end • vs2=Msg Number • start t_C • AIDC-DATA REQ with AIDC-crd-end-apdu ⇒ RE-NEGOTIATING	
rcv AIDC-crd-end-apdu			• vr1 = coord-end • vr2 = Msg Number • AIDC-crd-end IND • stop t_{1R} • stop t_{2R} • stop t_s • if c34 vre=accept ⇒ NEGOTIATING	• vs1 = coord-end • vr2 = Msg Number • AIDC-crd-end IND • stop t_{1R} • stop t_{2R} • stop t_s ⇒ RE-NEGOTIATING	
AIDC-crd-ngtt REQ			• vs1 = coord-negot • vs2 = Msg Number • AIDC-DATA with AIDC-crd-ngtt-apdu • start t_C ⇒ NEGOTIATING	• vs1 = coord-negot • vs2 = Msg Number • AIDC-DATA with AIDC-crd-ngtt-apdu • start t_C ⇒ RE-NEGOTIATING	
rcv AIDC-crd-ngtt-apdu			• vr1 = coord-negot • vr2 = coord-negot • AIDC-crd-ngtt IND • stop t_{1R} • stop t_{2R} • stop t_S ⇒ NEGOTIATING	• vr1 = coord-negot • vr2 = coord-negot • AIDC-crd-ngtt IND • stop t_{1R} • stop t_{2R} • stop t_S ⇒ RE-NEGOTIATING	
AIDC-crd-stndby REQ			• vs1 = coord-stndby • vs2 = Msg Number • AIDC-DATA REQ with AIDC-crd-stndby-apdu • start t_C ⇒ NEGOTIATING	• vs1 = coord-stndby • vs2 = Msg Number • AIDC-DATA REQ with AIDC-crd-stndby-apdu • start t_C ⇒ RE-NEGOTIATING	

State⇒ Event⇓	**IDLE**	**NOTIFY**	**NEGOTIATING**	**RE-NEGOTIATING**	**COORDINATED**
rcv AIDC-crd-stndby-apdu			• vs1?? = coord-stndby • vr2 = Msg Number • AIDC-crd-stndby IND • stop t_{1R} • stop t_{2R} ⇒ NEGOTIATING	• vs1?? = coord-stndby • vr2 = Msg Number • AIDC-crd-stndby IND • stop t_{1R} • stop t_{2R} ⇒ RE-NEGOTIATING	
AIDC-tfr-init REQ					• vs1 = trns-init • vs2 = Msg Number • AIDC-DATA REQ with AIDC-tfr-init-apdu • stop t_{1CT} • stop t_{2CT} • start t_C ⇒ COORDINATED
rcv AIDC-tfr-init -apdu					• vr1 = trns-init • vr2 = Msg Number • AIDC-tfr-init IND • stop t_{1CT} • stop t_{2CT} ⇒ COORDINATED
AIDC-tfr-rqst REQ					• vs2 = Msg Number • AIDC-DATA REQ with AIDC-tfr-rqst-apdu • start t_C ⇒ COORDINATED
rcv AIDC-tfr-rqst-apdu					• vr2 = Msg Number • AIDC-tfr-rqst IND ⇒ COORDINATED

State⇒ Event⇓	**IDLE**	**NOTIFY**	**NEGOTIATING**	**RE-NEGOTIATING**	**COORDINATED**
AIDC-tfr-cntrl REQ					• vs1 = trns-start • vs2 = Msg Number • AIDC-DATA REQ with AIDC-tfr-cntrl-Req-apdu • stop t_{1CT} • stop t_{2CT} • start t_C ⇒ COORDINATED
rcv AIDC-tfr-cntrl-req-apdu					• vr1 = trns-start • vr2 = Msg Number • AIDC-tfr-cntrl IND • stop t_{1CT} • stop t_{2CT} • stop t_{2R} ⇒ COORDINATED
AIDC-inf-tfr REQ	• vs1 = info-trans • vs2 = Msg Number • AIDC-DATA REQ with AIDC-inf-tfr-apdu • start t_C • stop t_{2IN} ⇒ IDLE	• vs1 = info-trans • vs2 = Msg Number • AIDC-DATA REQ with AIDC-inf-tfr-apdu • start t_C • stop t_{2IN} ⇒ NOTIFY	• vs1 = info-trans • vs2 = Msg Number • AIDC-DATA REQ with AIDC-inf-tfr-apdu • start t_C • stop t_{2IN} ⇒ NEGOTIATING		• vs1 = info-trans • vs2 = Msg Number • AIDC-DATA REQ with AIDC-inf-tfr-apdu • start t_C • stop t_{2IN} ⇒ COORDINATED
rcv AIDC-inf-tfr-apdu	• vr1 = info-trans • vr2 = Msg Number • AIDC-inf-tfr IND • stop t_{1IN} ⇒ IDLE	• vr1 = info-trans • vr2 = Msg Number • AIDC-inf-tfr IND • stop t_{1IN} ⇒ NOTIFY	• vr1 = info-trans • vr2 = Msg Number • AIDC-inf-tfr IND • stop t_{1IN} ⇒ NEGOTIATING		• vr1 = info-trans • vr2 = Msg Number • AIDC-inf-tfr IND • stop t_{1IN} ⇒ COORDINATED

State⇒ Event⇓	**IDLE**	**NOTIFY**	**NEGOTIATING**	**RE-NEGOTIATING**	**COORDINATED**
AIDC-end REQ		• vs1 = end • vs2 = Msg Number • AIDC-DATA REQ with AIDC-end-apdu • stop t_{2NC} • start t_C ⇒ NOTIFY			• vs1 = end • vs2 = Msg Number • AIDC-DATA REQ with AIDC-end-apdu • stop t_{1CT} • stop t_{2CT} • start t_C ⇒ COORDINATED
rcv AIDC-end -apdu	• vr1 = end • vr2 = Msg Number • AIDC-end IND • stop t_{2R} ⇒ IDLE	• vr1 = end • vr2 = Msg Number • AIDC-end IND • stop t_{1NC} ⇒ NOTIFY			• vr1 = end • vr2 = Msg Number AIDC-end IND • stop t_{1CT} • stop t_{2CT} ⇒ COORDINATED
AIDC-usr-abrt REQ	• AIDC-ABORT REQ • stop all timers ⇒ IDLE	• AIDC-ABORT REQ • stop all timers ⇒ IDLE	• AIDC-ABORT REQ • stop all timers ⇒ IDLE	• AIDC-ABORT REQ • stop all timers ⇒ IDLE	• AIDC-ABORT REQ • stop all timers ⇒ IDLE
AIDC-ABORT IND		• AIDC-usr-abrt IND • stop all timers ⇒ IDLE	• AIDC-usr-abrt IND • stop all timers ⇒ IDLE	• AIDC-usr-abrt IND • stop all timers ⇒ IDLE	• AIDC-usr-abrt IND • stop all timers ⇒ IDLE
AIDC-P-ABORT IND		• AIDC-pvd-abrt IND • stop all timers ⇒ IDLE	• AIDC-pvd-abrt IND • stop all timers ⇒ IDLE	• AIDC-pvd-abrt IND • stop all timers ⇒ IDLE	• AIDC-pvd-abrt IND • stop all timers ⇒ IDLE

State⇒ Event⇓	**IDLE**	**NOTIFY**	**NEGOTIATING**	**RE-NEGOTIATING**	**COORDINATED**

AIDC-ucf REQ	if c2 {
	• AIDC-DATA REQ with AIDC-ucf-apdu
	• if c1 & c4 {
	• start t_{1NC}
	⇒ NOTIFY}
	• if c1 & c6
	⇒ NEGOTIATING
	• if c1 & c32 {
	• start t_{1NC}
	⇒ IDLE }
	• if !c1 {
	• if c4 start t_{1NC}
	• if c6 start t_{1NC}
	• if c32 start t_{1IN}
	• vr1 = NULL
	• vr2 = NULL
	• vre = NULL
	⇒ IDLE}
	}
	if !c2 {
	• AIDC-pvd-abrt IND
	• AIDC-ABORT REQ
	• stop all timers
	⇒ IDLE}
		if c2 {
		• AIDC-DATA REQ with AIDC-ucf-apdu
		• if c1 & c4 {
		• start t_{1NC}
		⇒ NOTIFY}
		• if c1 & c6
		⇒ NEGOTIATING
		• if c1 & c30
		⇒ IDLE
		• if !c1 {
		• start t_{1NC}
		• vr1 = NULL
		• vr2 = NULL
		• vre = NULL
		⇒ NOTIFY}
		}
		if !c2 {
		• AIDC-pvd-abrt IND
		• AIDC-ABORT REQ
		• stop all timers
		⇒ IDLE}
			if c2 {
			• AIDC-DATA REQ with AIDC-ucf-apdu
			• if c1 & c12 & c39 {
			• start t_{1CT}
			• vre = NULL
			⇒ COORDINATED}
			• if c1 & c12 & c40 {
			• start t_{1NC}
			• vre = NULL
			⇒ NOTIFY}
			• if c1 & c10 {
			• start t_S
			⇒ NEGOTIATING}
			• if c1 & c30
			⇒ IDLE
			• if !c1 {
			• if !c38 start t_{2R}
			• vr1 = NULL
			• vr2 = NULL
			• vre = NULL
			⇒ NEGOTIATING}
			}
			if !c2 {
			• AIDC-pvd-abrt IND
			• AIDC-ABORT REQ
			• stop all timers
			⇒ IDLE}
				if c2 {
				• AIDC-DATA REQ with AIDC-ucf-apdu
				• if c1 & c12 {
				• start t_{1CT}
				⇒ COORDINATED}
				• if c1 & c10 {
				• start t_S
				⇒ RE-NEGOTIATING}
				• if c1 & c30
				⇒ IDLE
				• if !c1 {
				• if !c38 start t_{2R}
				• vr1 = NULL
				• vr2 = NULL
				• vre = NULL
				⇒ RE-NEGOTIATING}
				}
				if !c2 {
				• AIDC-pvd-abrt IND
				• AIDC-ABORT REQ
				• stop all timers
				⇒ IDLE}
					if c2 {
					• AIDC-DATA REQ with AIDC-ucf-apdu
					• if c1 & c6
					⇒ RENEGOTIATING
					• if c1 & c14 {
					• start t_{3R}
					⇒ PRE-TRANSFER }
					• if c1 & c16
					⇒ TRANSFERRING
					• if c1 & c30 {
					• stop all timers
					⇒ IDLE }
					• if !c1 {
					• start t_{1CT}
					• vr1 = NULL
					• vr2 = NULL
					• vre = NULL
					⇒ COORDINATED}
					if !c2 {
					• AIDC-pvd-abrt IND
					• AIDC-ABORT REQ
					• stop all timers
					⇒ IDLE}

State⇒ Event⇓	**IDLE**	**NOTIFY**	**NEGOTIATING**	**RE-NEGOTIATING**	**COORDINATED**

rcv AIDC-ucf REQ pdu	if c3 {
	• AIDC-ucf IND
	• if c1 & c5 {
	• start t_{2NC}
	⇒ NOTIFY }
	• if c1 & c7 {
	• start t_{1R}
	⇒ NEGOTIATING}
	• if c1 & c33 {
	• start t_{2IN}
	⇒ IDLE }
	• if c1 & c31
	⇒ IDLE
	• if !c1 {
	• if c4 start t_{2NC}
	• if c6 start t_{2NC}
	• if c32 start t_{2IN}
	• vs1=NULL
	• vs2=NULL
	• vse=NULL
	⇒ IDLE }
	}
	if !c3 {
	• AIDC-pvd-abrt IND
	• AIDC-ABORT REQ
	• stop all timers
	⇒ IDLE }
		if c3 {
		• AIDC-ucf IND
		• if c1 & c5 {
		• start t_{2NC}
		⇒ NOTIFY }
		• if c1 & c7 {
		• start t_{1R}
		⇒ NEGOTIATING}
		• if c1 & c33 {
		• start t_{2IN}
		⇒ NOTIFY }
		• if c1 & c31
		⇒ IDLE
		• if !c1 {
		• start t_{2NC}
		• vs1=NULL
		• vs2=NULL
		• vse=NULL
		⇒ NOTIFY }
		}
		if !c3 {
		• AIDC-pvd-abrt IND
		• AIDC-ABORT REQ
		• stop all timers
		⇒ IDLE }
			if c3 {
			• AIDC-ucf IND
			• if c1 & c9 {
			• start t_{1R}
			⇒ NEGOTIATING }
			• if c1 & c13 & c41 {
			• start t_{2CT}
			• vse = NULL
			⇒ COORDINATED}
			• if c1 & c13 & c42 {
			• start t_{2NC}
			• vse = NULL
			⇒ NOTIFY}
			• if c1 & c31
			⇒ IDLE
			• if !c1 {
			• stop all timers
			• vs1=NULL
			• vs2=NULL
			• vse=NULL
			⇒ NEGOTIATING }
			}
			if !c3 {
			• AIDC-pvd-abrt IND
			• AIDC-ABORT REQ
			• stop all timers
			⇒ IDLE }
				if c3 {
				• AIDC-ucf IND
				• if c1 & c9 {
				• start t_{1R}
				⇒ RE-NEGOTIATING }
				• if c1 & c13 {
				• start t_{2CT}
				⇒ COORDINATED}
				• if c1 & c31
				⇒ IDLE
				• if !c1 {
				• stop all timers
				• vs1=NULL
				• vs2=NULL
				• vse=NULL
				⇒ RE-NEGOTIATING }
				}
				if !c3 {
				• AIDC-pvd-abrt IND
				• AIDC-ABORT REQ
				• stop all timers
				⇒ IDLE }
					if c3 {
					• AIDC-ucf IND
					• if c1 & c7 {
					• start t_{1R}
					⇒ RE-NEGOTIATING }
					• if c1 & c15
					⇒ PRE-TRANSFER
					• if c1 & c17
					⇒ TRANSFERRING
					• if c1 & c31 {
					• stop all timers
					• start t_{2NC}
					⇒ IDLE }
					• if !c1 {
					• start t_{2CT}
					• vs1=NULL
					• vs2=NULL
					• vse=NULL
					⇒ COORDINATED }
					}
					if !c3 {
					• AIDC-pvd-abrt IND
					• AIDC-ABORT REQ
					• stop all timers
					⇒ IDLE }

State⇒ Event⇓	**PRE-TRANSFER**	**TRANSFERRING**	**TRANSFERRED**	**BACKWARD COORDINATING**
AIDC-crd-start REQ			• vs1 = back • vs2 = Msg Number • AIDC-DATA REQ with AIDC-crd-start-apdu • start t_C • stop t_{2IN} • stop t_{TE} ⇒ TRANSFERRED	
rcv AIDC-crd-start-apdu			• vr1 = back • vr2 = Msg Number • AIDC-crd-start IND • stop t_{1IN} • stop t_{2R} • stop t_{TE} ⇒ TRANSFERRED	
AIDC-crd-end REQ				• vs1 = back-end • vs2 = Msg Number • AIDC-DATA REQ with AIDC-crd-end-apdu • start t_C ⇒ BACKWARD COORDINATING
rcv AIDC-crd-end-apdu				• vr1 = back-end • vr2 = Msg Number • AIDC-crd-end IND • stop t_{1R} • stop t_{2R} • stop t_S ⇒ BACKWARD COORDINATING
AIDC-crd-ngtt REQ				• vs1 = coord-negot • vs2 = Msg Number • AIDC-DATA REQ with AIDC-crd-ngtt-apdu • start t_C ⇒ BACKWARD-COORDINATING

State⇒ Event⇓	**PRE-TRANSFER**	**TRANSFERRING**	**TRANSFERRED**	**BACKWARD COORDINATING**
rcv AIDC-crd-ngtt-apdu				• vr1 = coord-negot • vr2 = Msg Number • AIDC-crd-ngtt IND • stop t_{1R} • stop t_{2R} • stop t_S ⇒ BACKWARD-COORDINATING
AIDC-crd-stndby REQ				• vs1 = coord-stndby • vs2 = Msg Number • AIDC-DATA REQ with AIDC-crd-stndby-apdu • start t_C ⇒ BACKWARD COORDINATING
rcv AIDC-crd-stnby -apdu				• vs1?? = coord-stndby • vr2 = Msg Number • stop t_{1R} • stop t_{2R} ⇒ BACKWARD-COORDINATING
AIDC-tfr-prpsl REQ	• if P1, vs1 = trns-prpsl • vs2 = Msg Number • AIDC-DATA REQ with AIDC-tfr-prpsl-apdu • start t_C ⇒ PRE-TRANSFER			
rcv AIDC-tfr-prpsl-apdu	• If P1, vr1 = trns-prpsl • vr2 = Msg Number • AIDC_tfr-prpsl IND ⇒ PRE-TRANSFER			
AIDC-tfr-accept REQ	if P1 { • vs2 = Msg Number • AIDC-DATA REQ with AIDC-tfr-accept-apdu • start t_C ⇒ PRE-TRANSFER}			
rcv AIDC-tfr-accept-apdu	if P1 { • stop t_{1R} • AIDC-tfr-accept IND • vr2 = Msg Number ⇒ PRE-TRANSFER}			

State⇒ Event⇓	**PRE-TRANSFER**	**TRANSFERRING**	**TRANSFERRED**	**BACKWARD COORDINATING**
AIDC-tfr-comm REQ	• vs1 = trns-comm • vs2 = Msg Number • AIDC-DATA REQ with AIDC-tfr-comm-apdu • start t_C ⇒ PRE-TRANSFER			
rcv AIDC-tfr-comm apdu	• vr1 = trns-comm • vr2 = Msg Number • AIDC-tfr-comm IND • stop t_{2R} • stop t_{3R} ⇒ PRE-TRANSFER			
AIDC-tfr-comm-assm REQ	• vs1 = trns-assm • vs2 = Msg Number • AIDC-DATA REQ with AIDC-tfr-comm-assm-apdu • start t_C • stop t_{3R} ⇒ PRE-TRANSFER	• vs1 = trns-assm • vs2 = Msg Number • AIDC-DATA REQ with AIDC-tfr-comm-assm-apdu • start t_C • stop t_{3R} ⇒ TRANSFERRING		
rcv AIDC-tfr-comm-assm apdu	• vr1 = tnsf-assm • vr2 = Msg Number • AIDC-tfr-comm-assm IND • stop t_{1R} • stop t_{2R} ⇒ PRE-TRANSFER	• vr1 = tnsf-assm • vr2 = Msg Number • AIDC-tfr-comm-assm IND • stop t_{1R} • stop t_{2R} ⇒ TRANSFERRING		
AIDC-tfr-cntrl RSP		• if c36 vs1 = trns-accept • if c37 vs1 = trns-reject • vs2 = Msg Number • AIDC-DATA REQ with AIDC-tfr-cntrl-Rsp-apdu • start t_C • TRANSFERRING		
rcv AIDC-tfr-cntrl-Rsp-apdu		• if c36 vr1 = trns-accept • if c37 vr1 = trns-reject • vr2 = Msg Number • AIDC-tfr-cntrl CNF • stop t_{1R} • stop t_{2R} ⇒ TRANSFERRING		

State⇒ Event⇓	**PRE-TRANSFER**	**TRANSFERRING**	**TRANSFERRED**	**BACKWARD COORDINATING**
AIDC-inf-tfr REQ	• vs1 = info-trans • vs2 = Msg Number • AIDC-DATA REQ with AIDC-inf-tfr-apdu • start t_C • stop t_{2IN} ⇒ PRE-TRANSFER	• vs1 = info-trans • vs2 = Msg Number • AIDC-DATA REQ with AIDC-inf-tfr-apdu • start t_C • stop t_{2IN} ⇒ TRANSFERRING	• vs1 = info-trans • vs2 = Msg Number • AIDC-DATA REQ with AIDC-inf-tfr-apdu • start t_C • stop t_{2IN} ⇒ TRANSFERRED	
rcv AIDC-inf-tfr-apdu	• vr1 = info-trans • vr2 = Msg Number • AIDC-inf-tfr IND • stop t_{1IN} ⇒ PRE-TRANSFER	• vr1 = info-trans • vr2 = Msg Number • AIDC-inf-tfr IND • stop t_{1IN} ⇒ TRANSFERRING	• vr1 = info-trans • vr2 = Msg Number • AIDC-inf-tfr IND • stop t_{1IN} ⇒ TRANSFERRED	
AIDC-end REQ			• vs1 = end • vs2 = Msg Number • AIDC-DATA REQ with AIDC-end-apdu • start t_C ⇒ TRANSFERRED	
rcv AIDC-end REQ pdu			• vr1 = end • vr2 = Msg Number • AIDC-end IND • stop t_{TE} ⇒ TRANSFERRED	
AIDC-usr-abrt REQ	• AIDC-ABORT REQ • stop all timers ⇒ IDLE	• AIDC-ABORT REQ • stop all timers ⇒ IDLE	• AIDC-ABORT REQ • stop all timers ⇒ IDLE	• AIDC-ABORT REQ • stop all timers ⇒ IDLE
AIDC-ABORT IND	• AIDC-usr-abrt IND • stop all timers ⇒ IDLE	• AIDC-usr-abrt IND • stop all timers ⇒ IDLE	• AIDC-usr-abrt IND • stop all timers ⇒ IDLE	• AIDC-usr-abrt IND • stop all timers ⇒ IDLE
AIDC-P-ABORT IND	• AIDC-pvd-abrt IND • stop all timers ⇒ IDLE	• AIDC-pvd-abrt IND • stop all timers ⇒ IDLE	• AIDC-pvd-abrt IND • stop all timers ⇒ IDLE	• AIDC-pvd-abrt IND • stop all timers ⇒ IDLE

State⇒ Event⇓	**PRE-TRANSFER**	**TRANSFERRING**	**TRANSFERRED**	**BACKWARD COORDINATING**

AIDC-ucf REQ	if c2 {
	• AIDC-DATA REQ with AIDC-ucf-apdu
	• if c1 & c22
	⇒ TRANSFERRING
	• if c1 & c24 {
	• start t_{TE}
	⇒ TRANSFERRED }
	• if !c1 {
	• if !c38 start t_{2R}
	• vr1 = NULL
	• vr2 = NULL
	• vre = NULL
	⇒ PRE-TRANSFER}
	}
	if !c2 {
	• AIDC-pvd-abrt IND
	• AIDC-ABORT REQ
	• stop all timers
	• IDLE }
		if c2 {
		• AIDC-DATA REQ with AIDC-ucf-apdu
		• if c1 & c24 {
		• start t_{TE}
		⇒ TRANSFERRED }
		• if c1 & c18 {
		• start t_{TE}
		⇒ TRANSFERRED }
		• if c1 & c20 {
		• start t_{1CT}
		⇒ COORDINATED }
		• if !c1 {
		• if !c38 start t_{2R}
		• vr1 = NULL
		• vr2 = NULL
		• vre = NULL
		⇒ PRE-TRANSFER}
		}
		if !c2 {
		• AIDC-pvd-abrt IND
		• AIDC-ABORT REQ
		• stop all timers
		• IDLE }
			if c2 {
			• AIDC-DATA REQ with AIDC-ucf-apdu
			• if c1 & c26
			⇒ BACKWARD COORDINATING
			• if c1 & c30
			⇒ IDLE
			• if !c1 {
			• if !c38 start t_{2R}
			• vr1 = NULL
			• vr2 = NULL
			• vre = NULL
			⇒ PRE-TRANSFER}
			}
			if !c2 {
			• AIDC-pvd-abrt IND
			• AIDC-ABORT REQ
			• stop all timers
			• IDLE }
				if c2 {
				• AIDC-DATA REQ with AIDC-ucf-apdu
				• if c1 & c28
				⇒ TRANSFERRED
				• if !c1 {
				• if !c38 start t_{2R}
				• vr1 = NULL
				• vr2 = NULL
				• vre = NULL
				⇒ PRE-TRANSFER}
				}
				if !c2 {
				• AIDC-pvd-abrt IND
				• AIDC-ABORT REQ
				• stop all timers
				• IDLE }

State⇒ Event⇓	**PRE-TRANSFER**	**TRANSFERRING**	**TRANSFERRED**	**BACKWARD COORDINATING**

rcv AIDC-ucf-REQ pdu	if c3 {
	• AIDC-ucf IND
	• if c1 & c43 {
	• start t_{1R}
	⇒ PRE-TRANSFER }
	• if c1 & c23
	⇒ TRANSFERRING
	• if c1 & c25
	⇒ TRANSFERRED
	• if !c1 {
	• stop all timers
	• vs1=NULL
	• vs2=NULL
	• vse=NULL
	⇒ PRE-TRANSFER }
	}
	if !c3 {
	• AIDC-pvd-abrt IND
	• AIDC-ABORT REQ
	• stop all timers
	⇒ IDLE }
		if c3 {
		• AIDC-ucf IND
		• if c1 & c25
		⇒ TRANSFERRED
		• if c1 & c19
		⇒ TRANSFERRED
		• if c1 & c21 {
		• start t_{2CT}
		⇒ COORDINATED }
		• if !c1 {
		• stop all timers
		• vs1=NULL
		• vs2=NULL
		• vse=NULL
		⇒ TRANSFERRING}
		}
		if !c3 {
		• AIDC-pvd-abrt IND
		• AIDC-ABORT REQ
		• stop all timers
		⇒ IDLE }
			if c3 {
			• AIDC-ucf IND
			• if c1 & c27
			⇒ BACKWARD COORDINATING
			• if c1 & c31
			⇒ IDLE
			• if !c1 {
			• stop all timers
			• vs1=NULL
			• vs2=NULL
			• vse=NULL
			⇒ TRANSFERRED}
			}
			if !c3 {
			• AIDC-pvd-abrt IND
			• AIDC-ABORT REQ
			• stop all timers
			⇒ IDLE }
				if c3 {
				• AIDC-ucf IND
				• if c1 & c29
				⇒ TRANSFERRED
				• if !c1 {
				• stop all timers
				• vs1=NULL
				• vs2=NULL
				• vse=NULL
				⇒ BACKWARD COORDINATING}
				}
				if !c3 {
				• AIDC-pvd-abrt IND
				• AIDC-ABORT REQ
				• stop all timers
				⇒ IDLE }

3.2.7 AIDC FORMAL DEFINITIONS

Note.—The following defines the ASN.1 [ISO/IEC 8824-1] abstract syntax for the AIDC-AE.

3.2.7.1 AIDC ASN.1 Abstract Syntax

3.2.7.1.1 Each AIDC-APDU shall conform to the abstract syntax as defined below:

```
AIDCMessageSetVersion1 DEFINITIONS ::=

BEGIN

-- ------------------------------------------------------------------------------------------------------------------
-- AIDC-APDU
-- ------------------------------------------------------------------------------------------------------------------

AIDC-APDU ::= CHOICE
        {
        aidc-ucf-apdu               [0]   AIDC-ucf-apdu,
        aidc-nfy-apdu               [1]   AIDC-nfy-apdu,
        aidc-crd-start-apdu         [2]   AIDC-crd-start-apdu,
        aidc-crd-end-apdu           [3]   AIDC-crd-end-apdu,
        aidc-crd-ngtt-apdu          [4]   AIDC-crd-ngtt-apdu,
        aidc-crd-stndby-apdu        [5]   AIDC-crd-stndby-apdu,
        aidc-tfr-init-apdu          [6]   AIDC-tfr-init-apdu,
        aidc-tfr-rqst-apdu          [7]   AIDC-tfr-rqst-apdu,
        aidc-tfr-prpsl-apdu         [8]   AIDC-tfr-prpsl-apdu,
        aidc-tfr-accept-apdu        [9]   AIDC-tfr-accept-apdu,
        aidc-tfr-cntrl-req-apdu     [10]  AIDC-tfr-cntrl-Req-apdu,
        aidc-tfr-cntrl-rsp-apdu     [11]  AIDC-tfr-cntrl-Rsp-apdu,
        aidc-tfr-comm-apdu          [12]  AIDC-tfr-comm-apdu,
        aidc-tfr-comm-assm-apdu     [13]  AIDC-tfr-comm-assm-apdu,
        aidc-inf-tfr-apdu           [14]  AIDC-inf-tfr-apdu,
        aidc-end-apdu               [15]  AIDC-end-apdu,
        ...
        }
```

```
-- ------------------------------------------------------------------------------------------------------------------------
-- AIDC-apdu
-- ------------------------------------------------------------------------------------------------------------------------

AIDC-ucf-apdu ::=  SEQUENCE
        {
        result                                  [0]     Result,
        reason                                  [1]     ApplicationErrorData            OPTIONAL,
        referenceid                             [2]     MessageNumber
        }

AIDC-nfy-apdu ::=  SEQUENCE
        {
        calledICAOFacilityDesignation           [0]     ICAOFacilityDesignation,
        callingICAOFacilityDesignation [1]      ICAOFacilityDesignation         OPTIONAL,
        notify                                  [2]     Notify,
        msgnumber                               [3]     MessageNumber
        }

AIDC-crd-start-apdu ::=  SEQUENCE
        {
        calledICAOFacilityDesignation           [0]     ICAOFacilityDesignation,
        callingICAOFacilityDesignation [1]      ICAOFacilityDesignation         OPTIONAL,
        startdata                               [2]     Startdata,
        msgnumber                               [3]     MessageNumber
        }

Startdata ::= CHOICE
        {
        coordinateinitial                       [0]     CoordinateInitial,
        coordinateupdate                        [1]     CoordinateUpdate
        }

AIDC-crd-end-apdu ::=  SEQUENCE
        {
        enddata                                 [0]     Enddata,
        result                                  [1]     Result,
        msgnumber                               [2]     MessageNumber
        }

Enddata ::= CHOICE
        {
        coordinateaccept                        [0]     CoordinateAccept,
        coordinatereject                [1]     CoordinateReject
        }
```

```
AIDC-crd-ngtt-apdu ::=  SEQUENCE
        {
        coordinatenegotiate             [0]     CoordinateNegotiate,
        msgnumber                       [1]     MessageNumber
        }

AIDC-crd-stndby-apdu ::=  SEQUENCE
        {
        coordinatestandby               [0]     CoordinateStandby,
        msgnumber                       [1]     MessageNumber
        }

AIDC-tfr-init-apdu ::=  SEQUENCE
        {
        transferinitiate                [0]     TransferInitiate,
        msgnumber                       [1]     MessageNumber
        }

AIDC-tfr-rqst-apdu ::=  SEQUENCE
        {
        transferrequest                 [0]     TransferRequest,
        msgnumber                       [1]     MessageNumber
        }

AIDC-tfr-prpsl-apdu ::=  SEQUENCE
        {
        transferconditionsproposal      [0]     TransferConditionsProposal,
        msgnumber                       [1]     MessageNumber
        }

AIDC-tfr-accept-apdu ::=  SEQUENCE
        {
        transferconditionsaccept        [0]     TransferConditionsAccept,
        msgnumber                       [1]     MessageNumber
        }


AIDC-tfr-cntrl-Req-apdu ::=  SEQUENCE
        {
        transfercontrol                 [0]     TransferControl,
        msgnumber                       [1]     MessageNumber
        }
```

```
AIDC-tfr-cntrl-Rsp-apdu ::=  SEQUENCE
        {
        transfercontroldata             [0]     TransferControlData,
        result                          [1]     Result,
        msgnumber                       [2]     MessageNumber
        }

TransferControlData ::= CHOICE
        {
        transfercontrolassume           [0]     TransferControlAssume,
        transfercontrolreject           [1]     TransferControlReject
        }

AIDC-tfr-comm-apdu ::=  SEQUENCE
        {
        transfercomm                    [0]     TransferComm,
        msgnumber                       [1]     MessageNumber
        }

AIDC-tfr-comm-assm-apdu ::=  SEQUENCE
        {
        transfercommassume              [0]     TransferCommAssume,
        msgnumber                       [1]     MessageNumber
        }

AIDC-inf-tfr-apdu ::=  SEQUENCE
        {
        infodata                        [0]     InfoData,
        msgnumber                       [1]     MessageNumber
        }

InfoData ::= CHOICE
        {
        generalexecutivedata            [0]     GeneralExecutiveData,
        generalpoint                    [1]     GeneralPoint,
        surveillancegeneral             [2]     SurveillanceGeneral,
        generalfreetext                 [3]     GeneralFreeText,
        emergencyfreetext               [4]     EmergencyFreeText,
        ...
        }

AIDC-end-apdu ::= SEQUENCE
        {
        cancel                          [0]     Cancel          OPTIONAL,
        msgnumber                       [1]     MessageNumber
        }
```

```
MessageType ::= ENUMERATED
        {
        aidc-ucf-apdu                   (0),
        aidc-nfy-apdu                   (1),
        aidc-crd-start-apdu             (2),
        aidc-crd-end-apdu               (3),
        aidc-crd-ngtt-apdu              (4),
        aidc-crd-stndby-apdu            (5),
        aidc-tfr-init-apdu              (6),
        aidc-tfr-rqst-apdu              (7),
        aidc-tfr-prpsl-apdu             (8),
        aidc-tfr-accept-apdu            (9),
        aidc-tfr-cntrl-req-apdu         (10),
        aidc-tfr-cntrl-rsp-apdu         (11),
        aidc-tfr-comm-apdu              (12),
        aidc-tfr-comm-assm-apdu         (13),
        aidc-inf-tfr-apdu               (14),
        aidc-end-apdu                   (15),
        ...
        }
-- ------------------------------------------------------------------------------------------------------------------------
-- AIDC MESSAGE DEFINITIONS
-- ------------------------------------------------------------------------------------------------------------------------

Notify ::= SEQUENCE
        {
        flightID                [0]     FlightID,
        departure               [1]     DepartureAirportTime,
        destination             [2]     DestinationAirport              OPTIONAL,
        flightRuleFlightType    [3]     FlightRuleFlightType            OPTIONAL,
        beaconCode              [4]     BeaconCode                      OPTIONAL,
        aircraftNumberType      [5]     AircraftNumberType,
        cnsEquipment            [6]     CNSEquipment            OPTIONAL,
        boundaryEstimate        [7]     BoundaryEstimate,
        route                   [8]     Route                           OPTIONAL,
        otherInfo               [9]     OtherInformation                OPTIONAL,
        timestamp               [10]    YMDHMS,
        ...
        }

CoordinateInitial ::= SEQUENCE
        {
        flightID                [0]     FlightID,
        departure               [1]     DepartureAirportTime,
        destination             [2]     DestinationAirport              OPTIONAL,
        flightRuleFlightType    [3]     FlightRuleFlightType            OPTIONAL,
```

```
    beaconCode              [4]   BeaconCode              OPTIONAL,
    aircraftNumberType      [5]   AircraftNumberType,
    cnsEquipment            [6]   CNSEquipment        OPTIONAL,
    boundaryEstimate        [7]   BoundaryEstimate,
    route                   [8]   Route    OPTIONAL,
    otherInfo               [9]   OtherInformation    OPTIONAL,
    timestamp               [10]  YMDHMS,
    ...
    }

CoordinateUpdate ::= SEQUENCE
    {
    flightID                [0]   FlightID,
    departure               [1]   DepartureAirportTime,
    destination             [2]   DestinationAirport  OPTIONAL,
    beaconCode              [3]   BeaconCode          OPTIONAL,
    boundaryEstimate        [4]   BoundaryEstimate,
    route                   [5]   Route               OPTIONAL,
    timestamp               [6]   YMDHMS,
    ...
    }

CoordinateNegotiate ::=  SEQUENCE
    {
    flightID                [0]   FlightID,
    departure               [1]   DepartureAirportTime,
    destination             [2]   DestinationAirport  OPTIONAL,
    boundaryEstimate        [3]   BoundaryEstimate,
    route                   [4]   Route               OPTIONAL,
    timestamp               [5]   YMDHMS,
    ...
    }

CoordinateStandby ::= SEQUENCE
    {
    flightID                [0]   FlightID,
    departure               [1]   DepartureAirportTime,
    destination             [2]   DestinationAirport  OPTIONAL,
    timestamp               [3]   YMDHMS,
    ...
    }

CoordinateAccept ::= SEQUENCE
    {
    flightID                [0]   FlightID,
    departure               [1]   DepartureAirportTime,
```

```
        destination                 [2]     DestinationAirport      OPTIONAL,
        frequency                   [3]     Frequency               OPTIONAL,
        timestamp                   [4]     YMDHMS,
        ...
        }

CoordinateReject ::= SEQUENCE
        {
        flightID                    [0]     FlightID,
        departure                   [1]     DepartureAirportTime,
        destination                 [2]     DestinationAirport      OPTIONAL,
        timestamp                   [3]     YMDHMS,
        ...
        }

TransferInitiate ::=  SEQUENCE
        {
        flightID                    [0]     FlightID,
        departure                   [1]     DepartureAirportTime,
        destination                 [2]     DestinationAirport      OPTIONAL,
        executiveData               [3]     ExecutiveData           OPTIONAL,
        trackData                   [4]     TrackData               OPTIONAL,
        timestamp                   [5]     YMDHMS,
        ...
        }

TransferConditionsProposal ::=  SEQUENCE
        {
        flightID                    [0]     FlightID,
        departure                   [1]     DepartureAirportTime,
        destination                 [2]     DestinationAirport      OPTIONAL,
        executiveData               [3]     ExecutiveData           OPTIONAL,
        timestamp                   [4]     YMDHMS,
        ...
        }

TransferConditionsAccept ::=  SEQUENCE
        {
        flightID                    [0]     FlightID,
        departure                   [1]     DepartureAirportTime,
        destination                 [2]     DestinationAirport      OPTIONAL,
        frequency                   [3]     Frequency               OPTIONAL,
        timestamp                   [4]     YMDHMS,
        ...
        }
```

TransferRequest ::= SEQUENCE

```
{
flightID                [0]  FlightID,
departure               [1]  DepartureAirportTime,
destination             [2]  DestinationAirport     OPTIONAL,
frequency               [3]  Frequency              OPTIONAL,
timestamp               [4]  YMDHMS,
...
}
```

TransferControl ::= SEQUENCE

```
{
flightID                [0]  FlightID,
departure               [1]  DepartureAirportTime,
destination             [2]  DestinationAirport     OPTIONAL,
executiveData           [3]  ExecutiveData          OPTIONAL,
timestamp               [4]  YMDHMS,
...
}
```

TransferControlAssume ::= SEQUENCE

```
{
flightID                [0]  FlightID,
departure               [1]  DepartureAirportTime,
destination             [2]  DestinationAirport     OPTIONAL,
timestamp               [3]  YMDHMS,
...
}
```

TransferControlReject ::= SEQUENCE

```
{
flightID                [0]  FlightID,
departure               [1]  DepartureAirportTime,
destination             [2]  DestinationAirport     OPTIONAL,
timestamp               [3]  YMDHMS,
...
}
```

```
TransferComm ::= SEQUENCE
        {
        flightID                        [0]     FlightID,
        departure                       [1]     DepartureAirportTime,
        destination                     [2]     DestinationAirport          OPTIONAL,
        executiveData                   [3]     ExecutiveData               OPTIONAL,
        releaseIndicator        [4]     ReleaseIndicator            OPTIONAL,
        timestamp                       [5]     YMDHMS,
        ...
        }

TransferCommAssume ::= SEQUENCE
        {
        flightID                        [0]     FlightID,
        departure                       [1]     DepartureAirportTime,
        destination                     [2]     DestinationAirport          OPTIONAL,
        timestamp                       [3]     YMDHMS,
        ...
        }

SurveillanceGeneral ::=         SEQUENCE
        {
        flightID                        [0]     FlightID,
        departure                       [1]     DepartureAirportTime        OPTIONAL,
        destination                     [2]     DestinationAirport          OPTIONAL,
        trackData                       [3]     TrackData,
        timestamp                       [4]     YMDHMS,
        ...
        }

GeneralPoint ::= SEQUENCE
        {
        functionalAddress               [0]     FunctionalAddress           OPTIONAL,
        flightID                        [1]     FlightID,
        departure                       [2]     DepartureAirportTime        OPTIONAL,
        destination                     [3]     DestinationAirport          OPTIONAL,
        flightRuleFlightType            [4]     FlightRuleFlightType,
        beaconCode                      [5]     BeaconCode                  OPTIONAL,
        aircraftNumberType              [6]     AircraftNumberType,
        cnsEquipment                    [7]     CNSEquipment,
        boundaryEstimate                [8]     BoundaryEstimate            OPTIONAL,
        route                           [9]     Route                       OPTIONAL,
        otherInfo                       [10]    OtherInformation            OPTIONAL,
        timestamp                       [11]    YMDHMS,
        ...
        }
```

```
GeneralExecutiveData ::=  SEQUENCE
        {
        flightID                        [0]     FlightID,
        frequency                       [1]     Frequency,
        executivedata                   [2]     ExecutiveData,
        timestamp                       [4]     YMDHMS,
        ...
        }

EmergencyFreeText ::=  SEQUENCE
        {
        functionalAddress               [0]     FunctionalAddress       OPTIONAL,
        flightID                        [1]     FlightID,
        freeText                        [2]     FreeText,
        timestamp                       [3]     YMDHMS,
        ...
        }

GeneralFreeText ::=  SEQUENCE
        {
        functionalAddress               [0]     FunctionalAddress       OPTIONAL,
        flightID                        [1]     FlightID,
        freeText                        [2]     FreeText,
        timestamp                       [3]     YMDHMS,
        ...
        }

Cancel ::=  SEQUENCE
        {
        flightID                        [0]     FlightID,
        departure                       [1]     DepartureAirportTime,
        destination                     [2]     DestinationAirport      OPTIONAL,
        boundaryEstimate                [3]     BoundaryEstimate        OPTIONAL,
        otherinfo                       [4]     OtherInformation        OPTIONAL,
        timestamp                       [5]     YMDHMS,
        ...
        }

-- ------------------------------------------------------------------------------------------------------------------------
-- AIDC MESSAGE ELEMENTS
-- ------------------------------------------------------------------------------------------------------------------------

ProviderAbortReason ::= ENUMERATED
        {
        protocolerror                   (0),
        timerexpired                    (1),
```

```
        undefinederror                  (2),
        providererror                   (3),
        rejectedpermanent               (4),
        rejectedtransient               (5),
        ...
        }

AircraftNumberType ::=  SEQUENCE
        {
        numberOfAircraft        [0]     NumberOfAircraft        OPTIONAL,
        aircraftType            [1]     AircraftType,
        wakeTurbulenceCategory  [2]     WakeTurbulenceCategory  OPTIONAL
        }

AircraftIdentification ::=  IA5String (SIZE(2..7))

AircraftType ::=  IA5String (SIZE(2..4))

AircraftAddress ::=  BIT STRING (SIZE(24))

Airport ::=  IA5String (SIZE(4))

DepartureAirportTime ::=  SEQUENCE
        {
        airport         [0]     Airport,
        time            [1]     Time            OPTIONAL
        }

ATSRouteDesignator ::=  IA5String (SIZE(2..7))

Level ::=  CHOICE
        {
        levelFeet               [0]     LevelFeet,
        levelMetre              [1]     LevelMetre,
        levelFlightLevel        [2]     LevelFlightLevel,
        levelFlightLevelMetric  [3]     LevelFlightLevelMetric
        }

LevelFeet ::=  INTEGER (-60..7000)
        -- unit = Feet, Range (-600..70000), resolution = 10

LevelMetre ::=  INTEGER (-30..25000)
        -- unit =metre, Range (-30..25000), resolution = 1

LevelFlightLevel ::=  INTEGER (30..700)
        -- unit = Level (100 feet), Range (30..700), resolution = 1
```

```
LevelFlightLevelMetric ::=  INTEGER (100..2500)
        -- unit = Level (10 metres), Range (100..2500), resolution = 1

ApplicationErrorData ::=  SEQUENCE
        {
        messageType                     [0]     MessageType,
        componentType                   [1]     ComponentType,
        errorCode                       [2]     ErrorCode,
        errorData                       [3]     ErrorData               OPTIONAL
        }

ATWLevel ::=  SEQUENCE
        {
        atw                             [0]     ATWLevelTolerance,
        level                           [1]     Level
        }

ATWLevelTolerance ::=  ENUMERATED
        {
        at                              (0),
        atorabove                       (1),
        atorbelow                       (2)
        }

BeaconCode ::=  SEQUENCE SIZE (4) OF BeaconCodeOctalDigit

BeaconCodeOctalDigit ::=  INTEGER (0..7)

BoundaryEstimate ::=  SEQUENCE
        {
        boundaryFix                     [0]     Position,
        crossingTime                    [1]     Time,
        crossingLevel                   [2]     Level,
        atwLevel                        [3]     ATWLevel                OPTIONAL
        }

CNSEquipment ::=  SEQUENCE
        {
        comNavEquipmentStatus           [0]     SEQUENCE SIZE (0..24) OF
                                                ComNavEquipmentStatus   OPTIONAL,
        ssrEquipmentAvailable           [1]     SSREquipmentAvailable,
        adsAvailable                    [2]     BOOLEAN,
        acasAvailable                   [3]     BOOLEAN,
        dataLink                        [4]     SEQUENCE SIZE (0..4) OF DataLink
        }
```

```
ComNavEquipmentStatus ::=  ENUMERATED
        {
        aloranA                         (0),
        cloranC                         (1),
        ddme                            (2),
        edecca                          (3),
        fadf                            (4),
        ggnss                           (5),
        hhfRtf                          (6),
        iinertialNavigation             (7),
        lils                            (8),
        momega                          (9),
        ovor                            (10),
        pdoppler                        (11),
        rrnavRouteEquipment             (12),
        ttacan                          (13),
        uuhfRTF                         (14),
        vvhfRTF                         (15),
        ...
        }

DataLink ::=  ENUMERATED
        {
        hf                              (0),
        modeS                           (1),
        satcom                          (2),
        vhf                             (3)
        }

Date ::=  SEQUENCE
        {
        year                            [0]     Year,
        month                           [1]     Month,
        day                             [2]     Day
        }

Day ::=  INTEGER (1..31)
        -- unit = Day, Range (1..31), resolution = 1

Degrees ::=  CHOICE
        {
        degreesMagnetic                 [0]     DegreesMagnetic,
        degreesTrue                     [1]     DegreesTrue
        }

DegreesMagnetic ::=  INTEGER (1..360)
```

```
        -- unit = degree, Range (1..360), resolution = 1

DegreeMinutes ::=  INTEGER (0..5999)
        -- unit = Minute, Range (0..59.99), resolution = 0.01

DegreeSeconds ::=  INTEGER (0..59)
        -- unit = Second, Range (0..59), resolution = 1

DegreesTrue ::=  INTEGER (1..360)
        -- unit = degree, Range (1..360), resolution = 1

DestinationAirport ::= Airport

DirectRouting ::=  SEQUENCE
        {
        fix2                    [0]     Position                OPTIONAL,
        fix1                    [1]     Position
        }

Distance ::=  CHOICE
        {
        distanceNM              [0]     DistanceNM,
        distancekm              [1]     Distancekm
        }

Distancekm ::=  INTEGER (0..2000)
        -- unit = kilometre, Range (0..2000), resolution = 1

DistanceNM ::=  INTEGER (0..1000)
        -- unit = Nautical Mile, Range (0..1000), resolution = 1

ExecutiveData ::=  SEQUENCE
        {
        speed                   [0]     Speed                   OPTIONAL,
        level                   [1]     Level                   OPTIONAL,
        heading                 [2]     DegreesMagnetic         OPTIONAL,
        vertRate                [3]     VerticalChange      OPTIONAL,
        directRouting           [4]     DirectRouting           OPTIONAL
        }

FixName ::=  IA5String (SIZE(1..5))

FlightID ::= SEQUENCE
        {
        aircraftIdentification  [0]     AircraftIdentification,
        selcal                  [1]     Selcal                  OPTIONAL,
```

```
        registration                    [2]     Registration            OPTIONAL,
        airframeID                      [3]     AircraftAddress         OPTIONAL
        }

FlightRule ::=  ENUMERATED
        {
        ifr                             (0),
        vfr                             (1),
        ifrfirst                        (2),
        vfrfirst                        (3)
        }

FlightRuleFlightType ::=  SEQUENCE
        {
        flightRule                      [0]     FlightRule,
        flightType                      [1]     FlightType
        }

FlightType ::=  ENUMERATED
        {
        scheduledAirTransport                   (0),
        nonScheduledAirTransport                (1),
        generalAviation         (2),
        military                                (3),
        otherFlights                            (4)
        }

FreeText ::=  IA5String (SIZE(1..256))

Frequency ::=  CHOICE
        {
        frequencyHF                     [0]     FrequencyHF,
        frequencyVHFChannel             [1]     FrequencyVHFChannel,
        frequencyUHF                    [2]     FrequencyUHF,
        frequencySatChannel             [3]     FrequencySatChannel
        }

FrequencyHF ::=  INTEGER (2850..28000)
        -- unit = Kilohertz, Range (2850..28000), resolution = 1

FrequencyVHFChannel ::= INTEGER (23600..27398)
        -- unit: VHF Channel, Range: 118.000..136.990), resolution = 0.005

FrequencyUHF ::=  INTEGER (9000..15999)
        -- unit = Megahertz, Range (225.000..399.975), resolution = 0.025
```

```
FrequencySatChannel ::=  NumericString (SIZE(12))
        -- FrequencySatChannel corresponds to a 12-digit telephone number

FunctionalAddress ::= IA5String (SIZE(1..18))

ICAOFacilityDesignation ::=  IA5String (SIZE(4..8))

Latitude ::=  SEQUENCE
        {
        latitudeDegrees         [0]     LatitudeDegrees,
        latitudeMinutes         [1]     DegreeMinutes           OPTIONAL,
        latitudeSeconds         [2]     DegreeSeconds           OPTIONAL,
        latitudeDirection               [3]     LatitudeDirection
        }

LatitudeDegrees ::=  INTEGER (0..90000)
        -- unit = Degree, Range (0..90), resolution = 0.001

LatitudeDirection ::= ENUMERATED
        {
        north                   (0),
        south                   (1)
        }

LatitudeLongitude ::=  SEQUENCE
        {
        latitude                [0]     Latitude,
        longitude               [1]     Longitude
        }

Longitude ::= SEQUENCE
        {
        longitudeDegrees        [0]     LongitudeDegrees,
        longitudeMinutes        [1]     DegreeMinutes           OPTIONAL,
        longitudeSeconds        [2]     DegreeSeconds           OPTIONAL,
        longitudeDirection      [3]     LongitudeDirection
        }

LongitudeDegrees ::=  INTEGER (0..180000)
        -- unit = Degree, Range (0..180), resolution = 0.001

LongitudeDirection ::= ENUMERATED
        {
        east                    (0),
        west                    (1)
        }
```

MessageNumber ::= INTEGER (0..999999)

Month ::= INTEGER (1..12)
-- unit = Month, Range (1..12), resolution = 1

Navaid ::= IA5String (SIZE(1..4))

NumberOfAircraft ::= INTEGER (1..2)

OtherInformation ::= FreeText

PlaceBearing ::= SEQUENCE
{
fixName [0] FixName,
latitudeLongitude [1] LatitudeLongitude OPTIONAL,
degrees [2] Degrees
}

PlaceBearingDistance ::= SEQUENCE
{
placeBearing [0] PlaceBearing,
distance [1] Distance
}

PlaceBearingPlaceBearing ::= SEQUENCE SIZE (2) OF PlaceBearing

Position ::= CHOICE
{
fixName [0] FixName,
navaid [1] Navaid,
airport [2] Airport,
latitudeLongitude [3] LatitudeLongitude,
placeBearingDistance [4] PlaceBearingDistance
}

PublishedIdentifier ::= SEQUENCE
{
fixName [0] FixName,
latitudeLongitude [1] LatitudeLongitude
}

Registration ::= IA5String (SIZE(7))

```
ReleaseIndicator ::= ENUMERATED
        {
        climb                           (0),
        descent                         (1),
        turns                           (2),
        allActions                      (3)
        }

Result ::= ENUMERATED
        {
        accepted                        (0),
        rejected                        (1),
        ...
        }

Route ::= SEQUENCE
        {
        SEQUENCE SIZE (1..128) OF RouteInformation,
        position                        [0]     Position,
        time                            [1]     Time,
        level                           [2]     Level,
        speedGround                     [3]     SpeedGround,
        trueTrackAngle          [4]     TrueTrackAngle
        }

RouteInformation ::= CHOICE
        {
        publishedIdentifier             [0]     PublishedIdentifier,
        latitudeLongitude               [1]     LatitudeLongitude,
        placeBearingPlaceBearing        [2]     PlaceBearingPlaceBearing,
        placeBearingDistance            [3]     PlaceBearingDistance,
        aTSRouteDesignator              [4]     ATSRouteDesignator,
        trackDetail                     [5]     TrackDetail
        }

Selcal ::= IA5String (SIZE(4))

Speed ::= CHOICE
        {
        speedGround                     [0]     SpeedGround,
        speedGroundMetric               [1]     SpeedGroundMetric,
        speedMach                       [2]     SpeedMach,
        speedIndicated                  [3]     SpeedIndicated,
        speedIndicatedMetric            [4]     SpeedIndicatedMetric,
        speedTrue                       [5]     SpeedTrue,
        speedTrueMetric                 [6]     SpeedTrueMetric
```

```
        }

SpeedGround ::=  INTEGER (-50..2000)
        -- unit = Knots, Range (-50..2000), resolution = 1

SpeedGroundMetric ::=  INTEGER (-100..4000)
        -- unit = kilometre/hour, Range (-100..4000), resolution = 1

SpeedIndicated ::=  INTEGER (0..400)
        -- unit = Knots, Range (0..400), resolution = 1

SpeedIndicatedMetric ::=  INTEGER (0..800)
        -- unit =kilometre/hour, Range (0..800), resolution = 1

SpeedMach ::=  INTEGER (500..4000)
        -- unit = Mach, Range (0.5..4.0), resolution = 0.001

SpeedTrue ::=  INTEGER (0..2000)
        -- unit = Knots, Range (0..2000), resolution = 1

SpeedTrueMetric ::=  INTEGER (0..4000)
        -- unit =kilometre/hour, Range (0..4000), resolution = 1

SSREquipmentAvailable ::=  ENUMERATED
        {
        nnil                            (0),
        atransponderModeA               (1),
        ctransponderModeAandC           (2),
        xatransponderModeS              (3),
        ptransponderModeSPA             (4),
        itransponderModeSID             (5),
        satransponderModeSPAID          (6),
        ...
        }
        -- PA: Pressure Level; ID: Aircraft Identification

Time ::=  SEQUENCE
        {
        hours                           [0]     TimeHours,
        minutes                         [1]     TimeMinutes
        }

Timehhmmss ::=  SEQUENCE
        {
        hoursminute                     Time,
        seconds                         TimeSeconds
```

```
        }

TimeHours ::=  INTEGER (0..23)
        -- unit = Hour, Range (0..23), resolution = 1

TimeMinutes ::=  INTEGER (0..59)
        -- unit = Minute, Range (0..59), resolution = 1

TimeSeconds ::= INTEGER(0..59)
        -- unit= Second, Range (0..59), resolution = 1

TrackData ::=  SEQUENCE
        {
        position                        [0]     Position,
        time                            [1]     Time,
        level                           [2]     Level,
        speedGround                     [3]     SpeedGround,
        trueTrackAngle          [4]     TrueTrackAngle
        }

TrackDetail ::=  SEQUENCE
        {
        trackName                       [0]     TrackName,
        latitudeLongitude               [1]     LatitudeLongitude
        }

TrackName ::=  IA5String (SIZE(1..6))

TrueTrackAngle ::= Degrees

VerticalChange ::=  SEQUENCE
        {
        direction                       [0]     VerticalDirection,
        rate                            [1]     VerticalRate
        }

VerticalDirection ::=  ENUMERATED
        {
        up                              (0),
        down                            (1)
        }
```

```
VerticalRate ::=  CHOICE
        {
        verticalRateEnglish             [0]     VerticalRateEnglish,
        verticalRateMetric              [1]     VerticalRateMetric
        }

VerticalRateEnglish ::=  INTEGER (0..3000)
        -- unit = Feet/Minute, Range (0..30000), resolution = 10

VerticalRateMetric ::=  INTEGER (0..1000)
        -- unit =metre/Minute, Range (0..1000), resolution = 1

WakeTurbulenceCategory ::=  ENUMERATED
        {
        high                            (0),
        medium                          (1),
        low                             (2)
        }

Year ::=  INTEGER (1996..2095)
        -- unit = Year, Range (1996..2095), resolution = 1

YMDHMS ::= SEQUENCE
        {
        date                            [0]     Date,
        timehhmmss                      [1]     Timehhmmss
        }


-- ------------------------------------------------------------------------------------------------------------------------
-- AIDC ERROR-RELATED TYPES
-- ------------------------------------------------------------------------------------------------------------------------

ComponentType ::=  ENUMERATED
        {
        ctUnknown                       (0),
        ctNotApplicable                 (1),
        ctAircraftNumberType            (2),
        ctBeaconCode                    (3),
        ctBoundaryEstimate              (4),
        ctCNSEquipment                  (5),
        ctDepartureAirportTime          (6),
        ctDestinationAirport            (7),
        ctExecutiveData                 (8),
        ctFlightID                      (9),
        ctFlightRuleFlightType          (10),
```

```
        ctFreeText                          (11),
        ctFrequency                         (12),
        ctFunctionalAddress                 (13),
        ctReleaseIndicator                  (14),
        ctRoute                             (15),
        ctTrackData                         (16),
        ctUnrecognised              (255),
        ...
        }

ErrorCode ::= ENUMERATED
        {
        invalidNumberOfAircraft             (0),
        invalidAircraftType                 (1),
        invalidWakeTurbulenceCategory       (2),
        invalidBeaconCodeOctalDigit         (3),
        invalidFixName                      (4),
        invalidNavaid                       (5),
        invalidAirport                      (6),
        invalidLatitude                     (7),
        invalidLongitude                    (8),
        invalidTime                         (9),
        invalidLevelFeet                    (10),
        invalidLevelMetre                   (11),
        invalidLevelFlightLevel     (12),
        invalidLevelFlightLevelMetric       (13),
        invalidATWLevelTolerance            (14),
        invalidComNavEquipmentStatus        (15),
        invalidSSREquipmentAvailable (16),
        invalidDataLink                     (17),
        invalidSpeedGround                  (18),
        invalidSpeedGroundMetric            (19),
        invalidSpeedMach                    (20),
        invalidSpeedIndicated               (21),
        invalidSpeedIndicatedMetric         (22),
        invalidSpeedTrue                    (23),
        invalidSpeedTrueMetric              (24),
        invalidVerticalDirection    (25),
        invalidVerticalRateEnglish          (26),
        invalidVerticalRateMetric           (27),
        invalidAircraftIdentification       (28),
        invalidSelcal                       (29),
        invalidRegistration                 (30),
        invalidAircraftAddress              (31),
        invalidFlightRule                   (32),
        invalidFlightType                   (33),
```

```
        invalidFrequencyHF                          (34),
        invalidFrequencyVHFChannel                  (35),
        invalidFrequencyUHF                         (36),
        invalidFrequencySatChannel                  (37),
        invalidFunctionalAddress                    (38),
        invalidReleaseIndicator                     (39),
        invalidDistancekm                           (40),
        invalidDistanceNM                           (41),
        invalidATSRouteDesignator                   (42),
        invalidTrackName                            (43),
        invalidmsgnumber                            (250),
        invalidreferenceid                          (251),
        invalidcallingICAOFacilityDesignation       (252),
        invalidcalledICAOFacilityDesignation        (253),
        invalidtimestamp                            (254),
        unknown                                     (255),
        ...
        }
```

ErrorData ::= BIT STRING (SIZE(1..256))

END

3.2.8 COMMUNICATION REQUIREMENTS

3.2.8.1 Encoding Rules

3.2.8.1.1 The AIDC application shall use PER as defined in ISO/IEC 8825-2, using the Basic Unaligned variant to encode/decode the ASN.1 message structure and content specified in 3.2.7, or a functionally equivalent means which provides the same result.

3.2.8.2 Quality-of-Service Requirements

3.2.8.2.1 Routing Policy

3.2.8.2.1.1 Routing Class shall be conveyed by local means, using the values for Security Tag Value specified in Table 5.6-1.

3.2.8.2.1.2 For the AIDC application no routing class is specified, thus the value corresponding to "ATSC: No Traffic Type Policy Preference" shall be conveyed.

Note.— It is stated in 5.2.7.3.1, "The mechanism by which the connection initiator determines the appropriate ATN Security Label is a local matter. For example, it may be identified by an extension to the transport service interface, be implicit in the choice of a given TSAP, or be identified using a Systems Management function."

3.2.8.2.2 Priority

3.2.8.2.2.1 The AIDC application service priority shall have the value corresponding to "normal priority flight safety messages".

3.2.8.2.2.2 Priority shall map to the session connection priority component of the A-ASSOCIATE request primitive Quality of Service parameter (see Table 3.2.5-1), using the values for Transport Layer Priority specified in Table 1.2 (see .3.8).

Note.— Although transport priority and network priority are semantically independent of each other, 5.5.1.2 states that the TS-user specifies the Application Service Priority, which in turn is mapped into the resulting CLNP PDUs according to Table 1.2, which defines the fixed relationship between transport priority and the network priority.

3.2.8.2.3 Residual Error Rate

3.2.8.2.3.1 The required AIDC application service RER shall have the value corresponding to "low".

3.2.8.2.3.2 Thus the residual error rate component of the A-ASSOCIATE request primitive Quality of Service parameter (see Table 3.2.5-1) shall be set to zero.

Note.— 5.5.1.2 states that the transport service user specifies the required residual error rate to determine whether or not the transport checksum is required.

3.2.9 AIDC-USER REQUIREMENTS

Note.— The following identifies the requirements imposed upon the AIDC-User by the AIDC-AE.

3.2.9.1 Inter-Operability

3.2.9.1.1 To achieve inter-operability for the implementation of AIDC between ATS units, or ATS regions on a global basis, the following requirements shall be satisfied:

a) a common AIDC message set and associated services is agreed upon by the ATS units involved,

b) agreement, by the ATS units involved, as to what flight related conditions dictate the invocation of the AIDC services,

c) agreement as to the timing associated with the use of the AIDC services,

d) agreement as to the predicate P1 as defined in 3.2.6.1.4.

3.2.9.2 Message Handling

3.2.9.2.1 Message Priorities

3.2.9.2.1.1 In cases where message queueing is implemented by the AIDC-User, an application specific priority scheme shall be implemented.

3.2.9.2.1.1.1 Under this scheme the AIDC messages received shall be assigned one of the following priorities:

a) Normal;

b) Urgent; or

c) Distress.

3.2.9.2.1.1.2 Assignment of priorities:

a) General freetext messages shall be assigned the priority of “Normal”.

b) Emergency freetext messages shall be assigned the priority of “Distress”.

c) Surveillance data transfer messages shall be assigned the priority of “Urgent”.

d) All other AIDC messages shall be assigned the priority “Normal”.

3.2.9.2.1.1.3 The AIDC-User shall process messages with a priority of Distress first, followed by messages with a priority of Urgent and then messages with a priority of Normal.

3.2.9.3 Operational Timers

3.2.9.3.1 The AIDC-User shall manage a number of timers associated with the sending of AIDC messages. These timers are described below:

a) a Confirmation Timer, used to detect the failure of the AIDC-User to receive the user confirmation for a previously sent message;

b) a Response Timer, used to detect the failure of the AIDC-User to receive the appropriate message in response to a previously sent coordination or transfer message;

c) a Monitor Timer, used to detect the failure of the AIDC-User to receive another expected AIDC message; and

d) a Standby Timer, used to extend the time before a response to a coordination message is expected.

3.2.9.4 AIDC-AE Specific Requirements

3.2.9.4.1 Management of AIDC-AE Instantiations

3.2.9.4.1.1 As each instantiation of the AIDC-AE is on a flight by flight basis, the AIDC-User shall manage these instantiations by some local means.

3.2.9.4.2 The User-confirmation Service

3.2.9.4.2.1 Upon the receipt of an AIDC message, the AIDC-User shall validate the semantics of the message and use the User-confirmation service to indicate to the peer AIDC-User whether the message has been accepted or rejected.

3.2.9.4.2.2 In order for the AIDC-AE to correlate user service primitives with the User-confirmation primitives, the AIDC-User shall generate and manage a set of unique identifiers.

3.2.9.4.2.3 The identifiers received by an AIDC-User in indication service primitives, shall be used as the *Referenced Number* parameter of the User-confirmation request primitive.

3.2.9.4.3 Coordination Messages

Note.— The Coordinate-start service is used for the initial flight coordination and the updating of the coordination conditions for a flight. The passing of either the CoordinateInitial or CoordinateUpdate message when using the service is under user control and procedures.

3.2.9.4.3.1 If for some reason the D-ATSU AIDC-User cannot accept the type of message received, it shall signal that condition with a negative User-confirmation

3.2.9.5 Error Processing Requirements

3.2.9.5.1 In the event of information input by the user being incompatible with that able to be processed by the system, the user shall be notified.

3.2.10 SEQUENCE DIAGRAMS

3.2.10.1 Sequence Diagrams

3.2.10.1.1 On the invocation of an AIDC-User request primitives or on the receipt of AIDC-ASE primitives from the underlying communication service provider, the AIDC-AE shall ensure that the appropriate sequencing of primitives shown in the following figures, is enforced.

Note 1.— The sequence diagrams, shown below, do not mandate the user sequencing of primitives. Nor do they cover all possible sequences.

Note 2.— The following figures, show the sequence of AIDC-User primitives and the CF mapping of those primitives to/from the primitives of the AIDC-ASE.

Note 3.— As the User-confirmation service is common to all services, except the abort services, the invocation of the primitives are shown as a dotted line.

Note 4.— The "+" symbol preceding the User-confirmation primitive indicates that the user has validated and accepted the contents or semantics of the received message.

Note 5.— The "-" symbol preceding the User-confirmation primitive indicates that the user has rejected the contents of the received message.

Note 6.— The "|" symbol represents an OR statement.

Note 7.— Primitives shown with dashed lines identify the User-Confirmation service primitives.

Note 8.— The dotted arrows under the column headed "ATN Service Provider" simply identifies transition of data between AIDC-AE peers across the ATN internet.

Note 9.— Dotted extentions to the columns identifies that the sequence shown is either the continuation of a sequence of service invocations or other service invocations are to follow.

Note 10.— The timers shown in the diagrams are technical timers and are defined in 3.2.6 of this document.

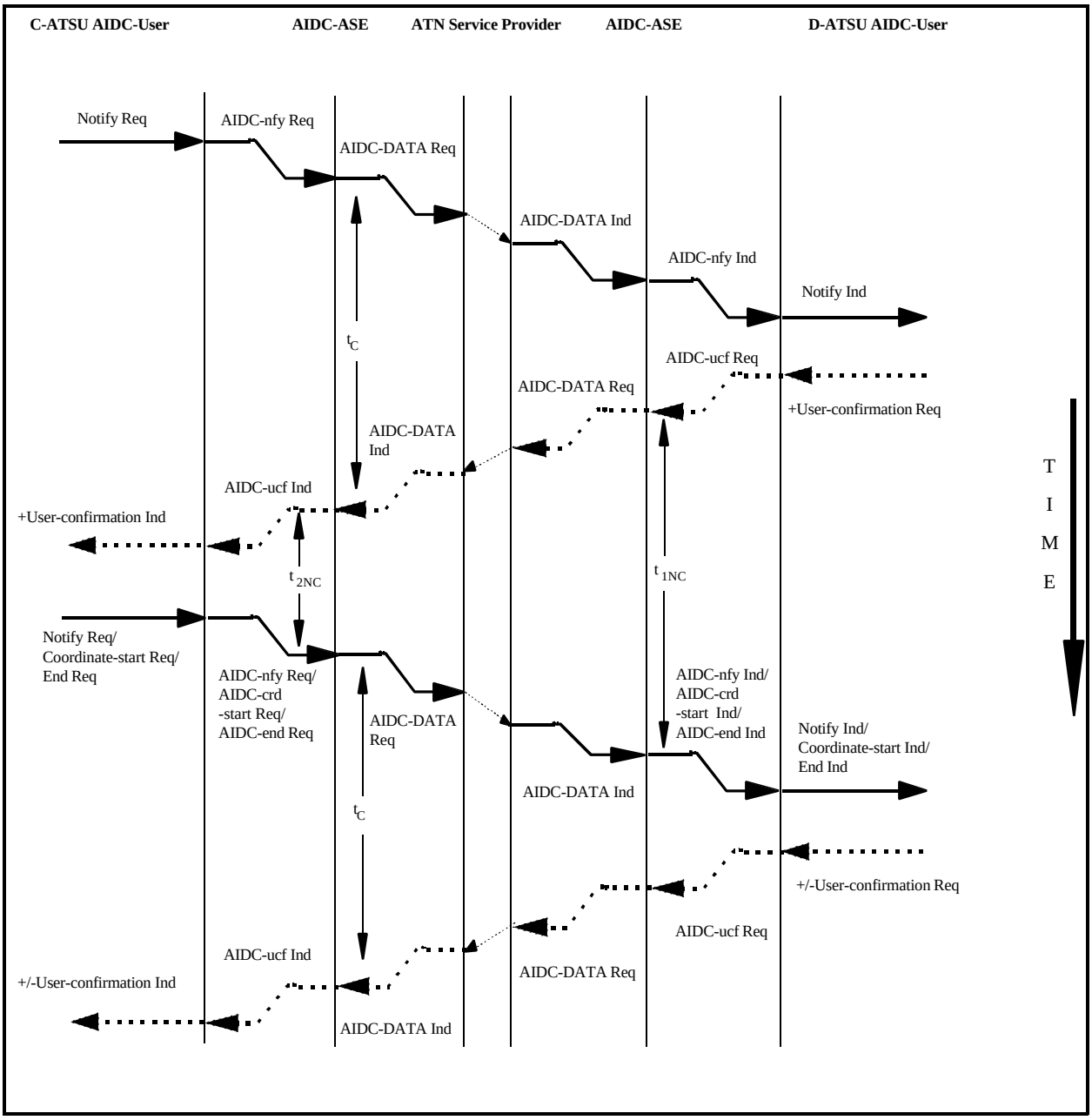

Figure 3.2.10-1: Sequence Diagram showing the entry into the Notifying Regime through the invocation of the Notify service

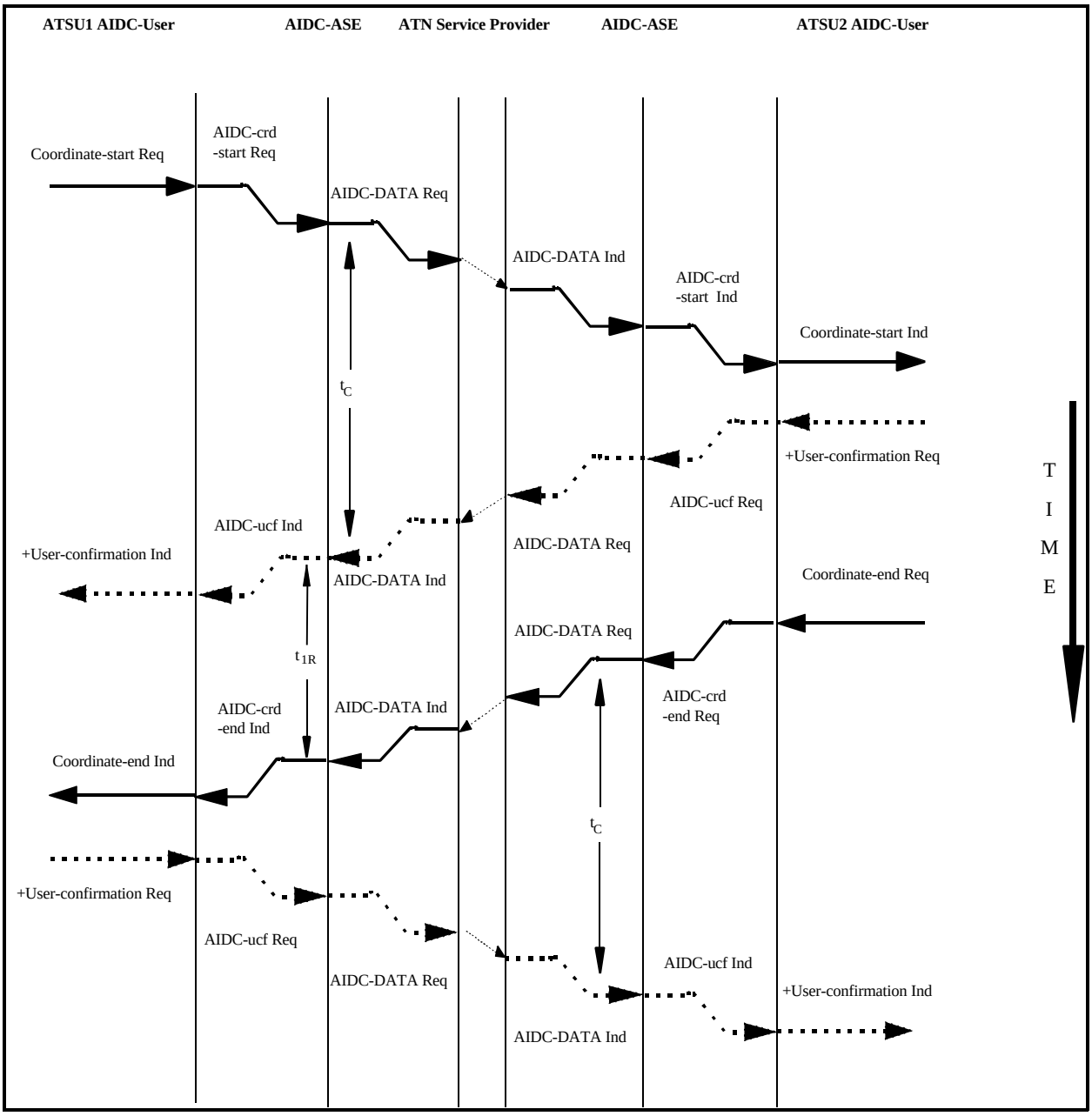

Figure 3.2.10-2: Sequence Diagram showing the entry and exit of the Coordination Regime through the invocation of the Coordinate-start Service and the Cordinate-end Service respectively

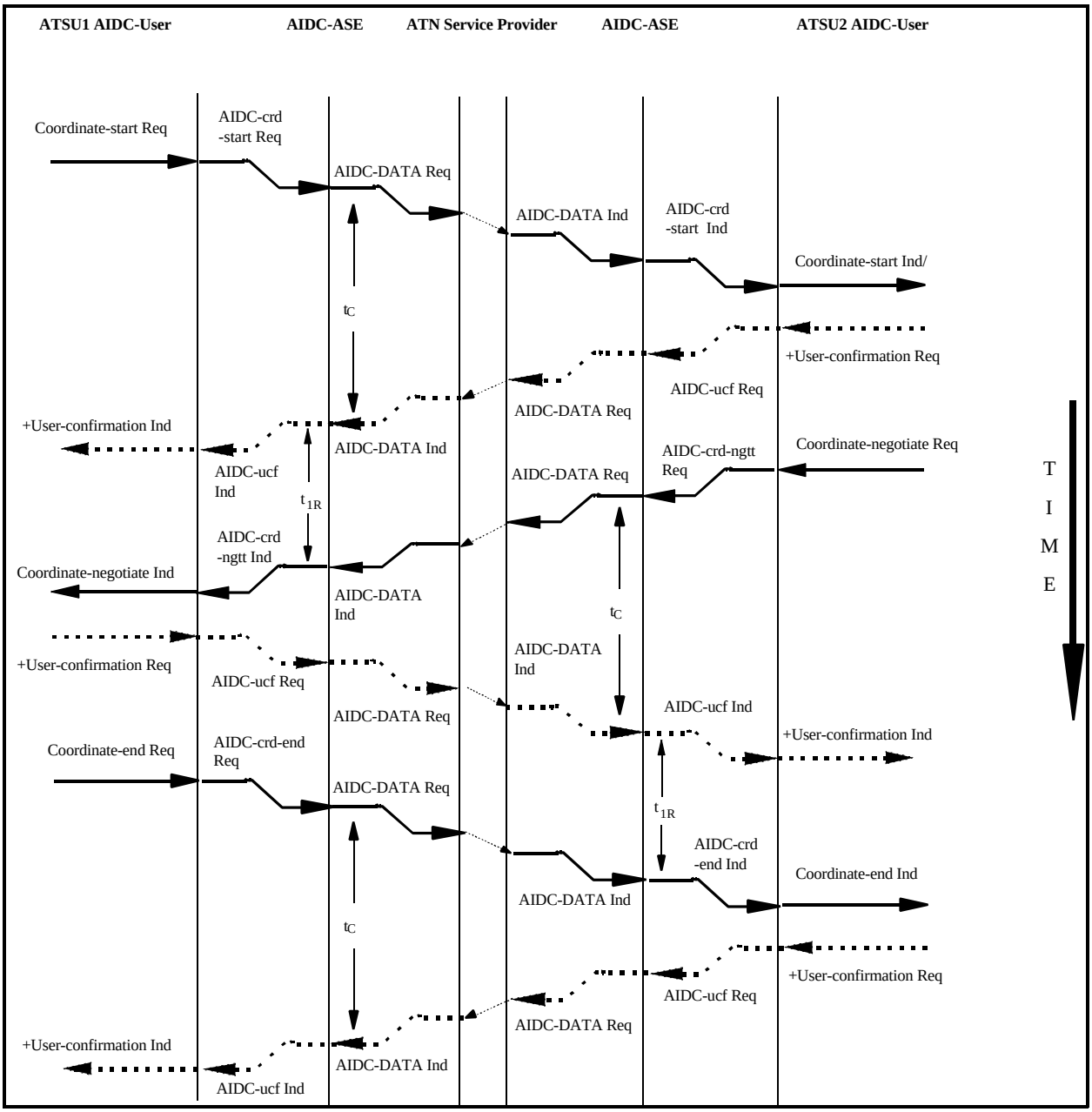

Figure 3.2.10-3: Sequence Diagram showing the invocation of the Coordinate-negotiate Service

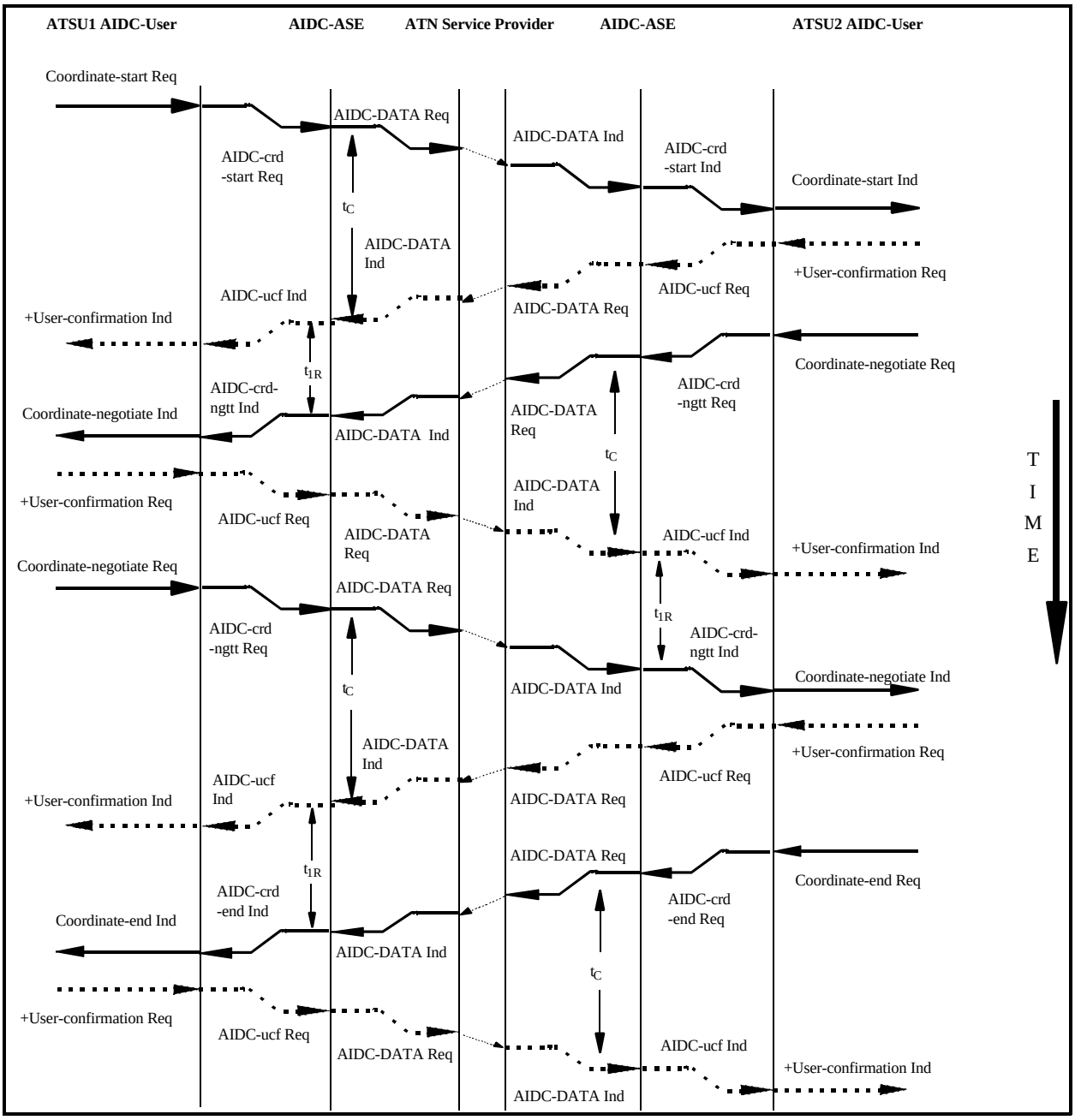

Figure 3.2.10-4: Sequence Diagram showing multiple invocations of the Coordinate-negotiate service

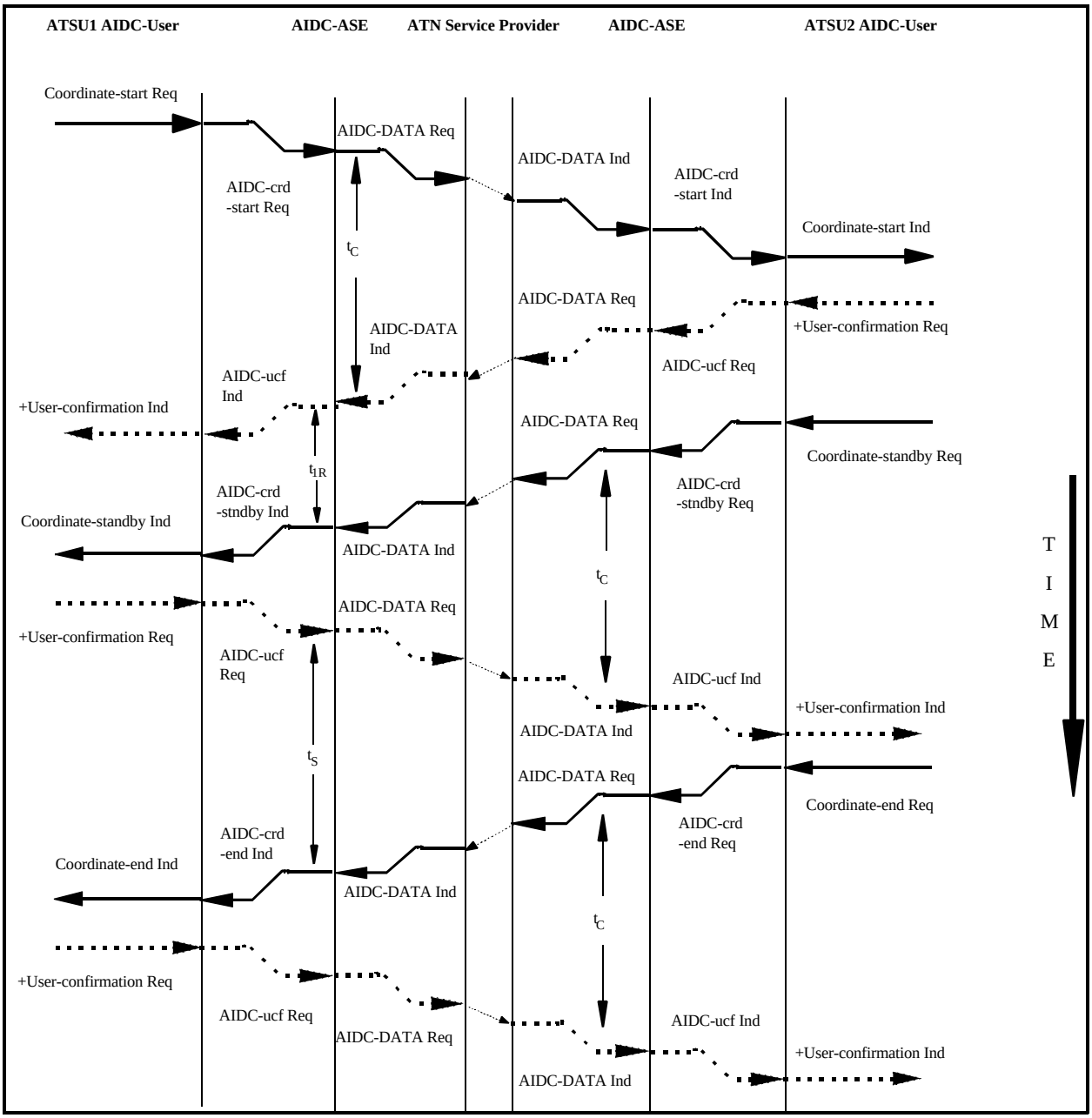

Figure 3.2.10-5: Sequence Diagram showing the invocation of the Coordinate-standby service

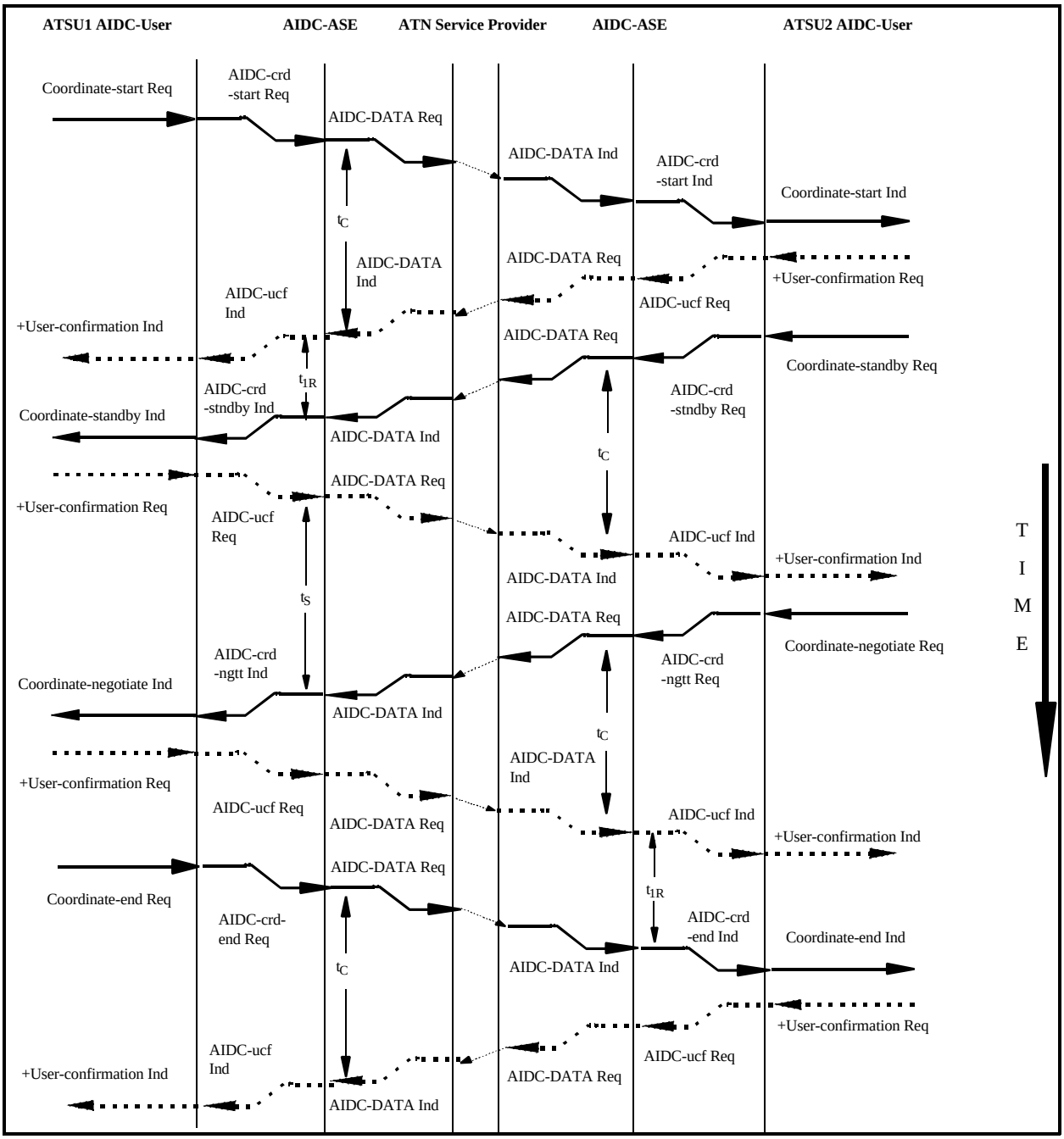

Figure 3.2.10-6: Sequence Diagram showing the invocation of the Coordinate-standby service followed by the invocation of the Coordinate-negotiate service

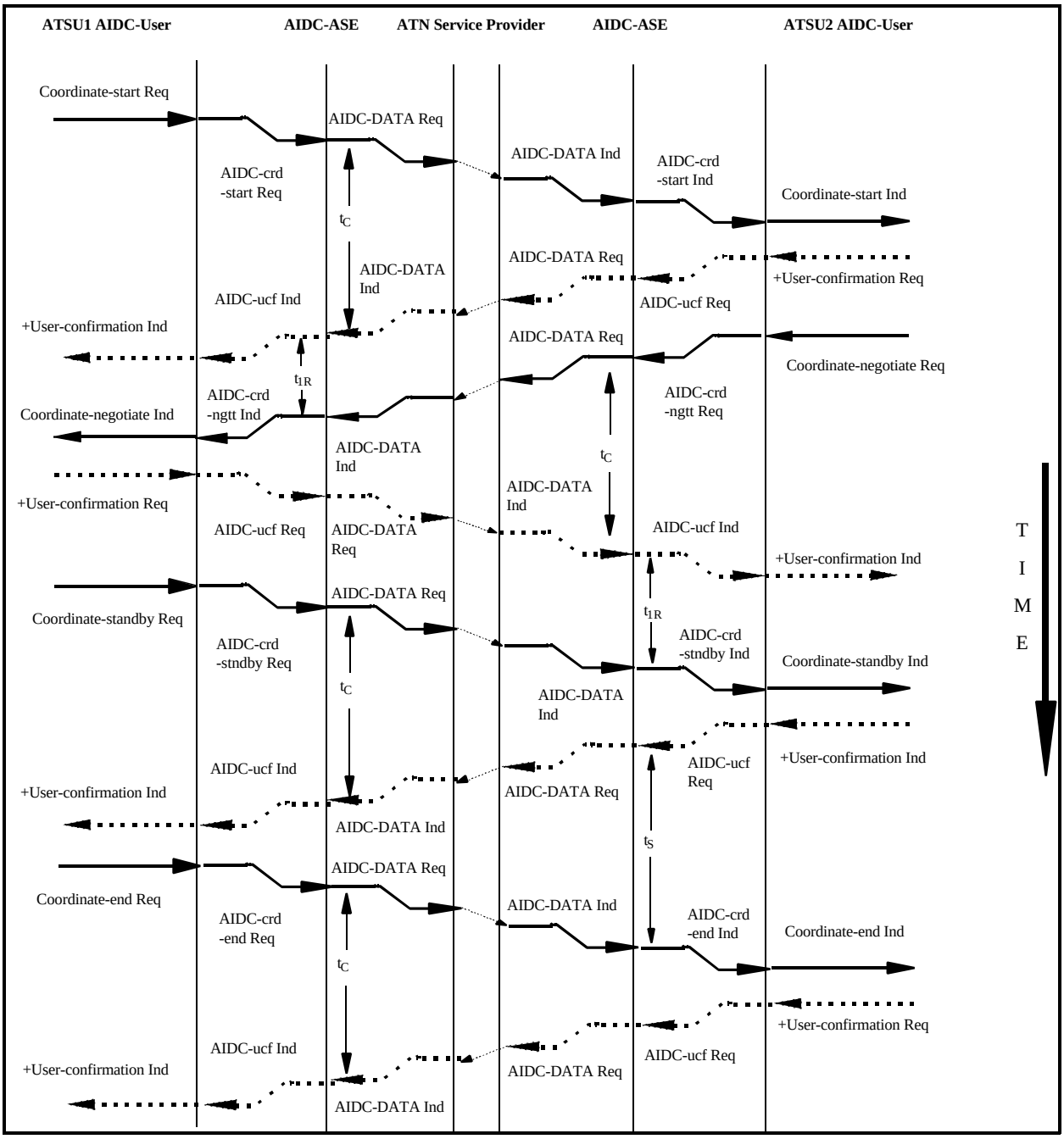

Figure 3.2.10-7: Sequence Diagram showing the invocation of the Coordinate-standby service followed by the Coordinate-negotiate service

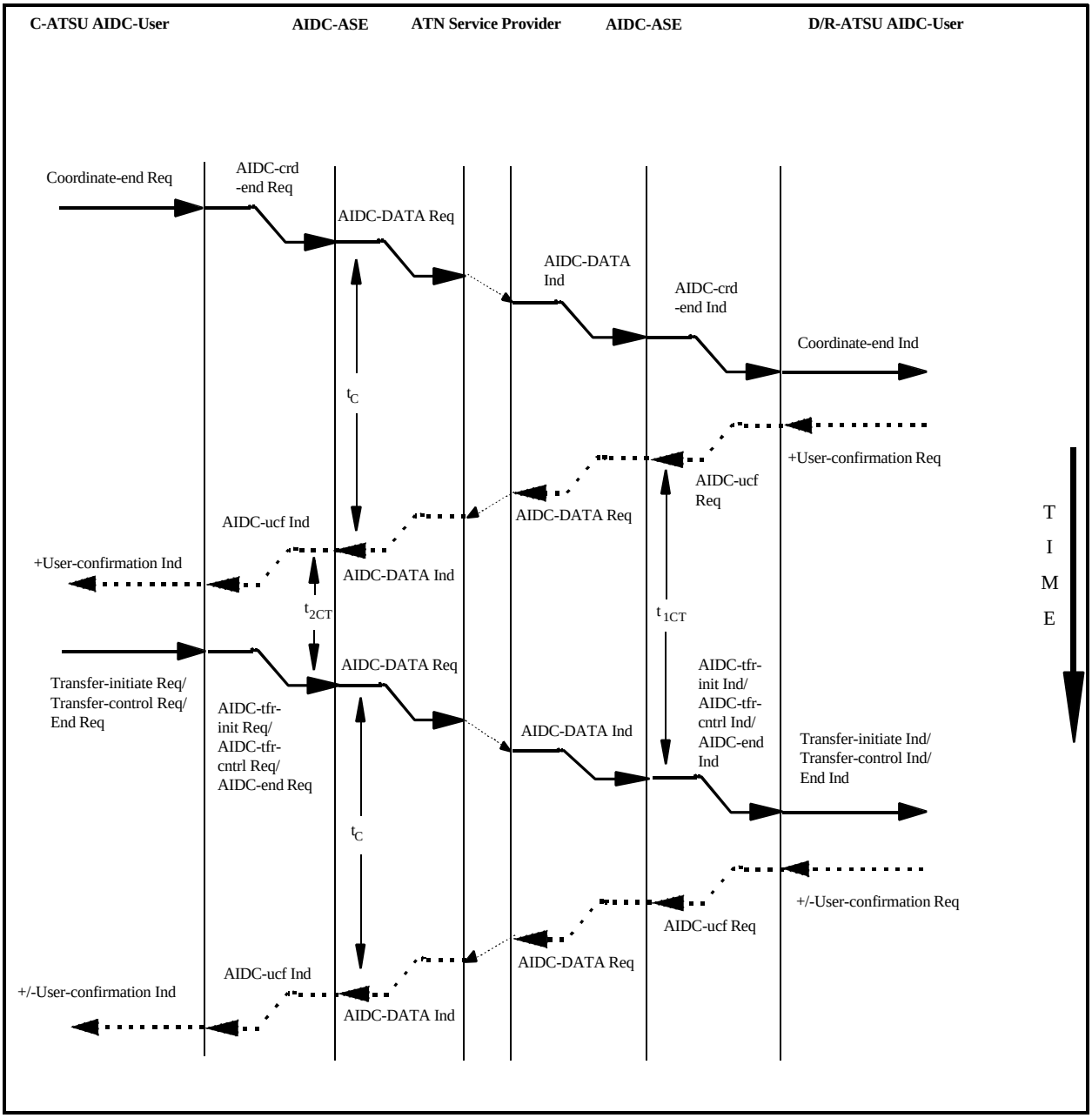

Figure 3.2.10-8: Sequence Diagram showing the start of the Transferring Regime through the invocation of the Transfer-initiate service

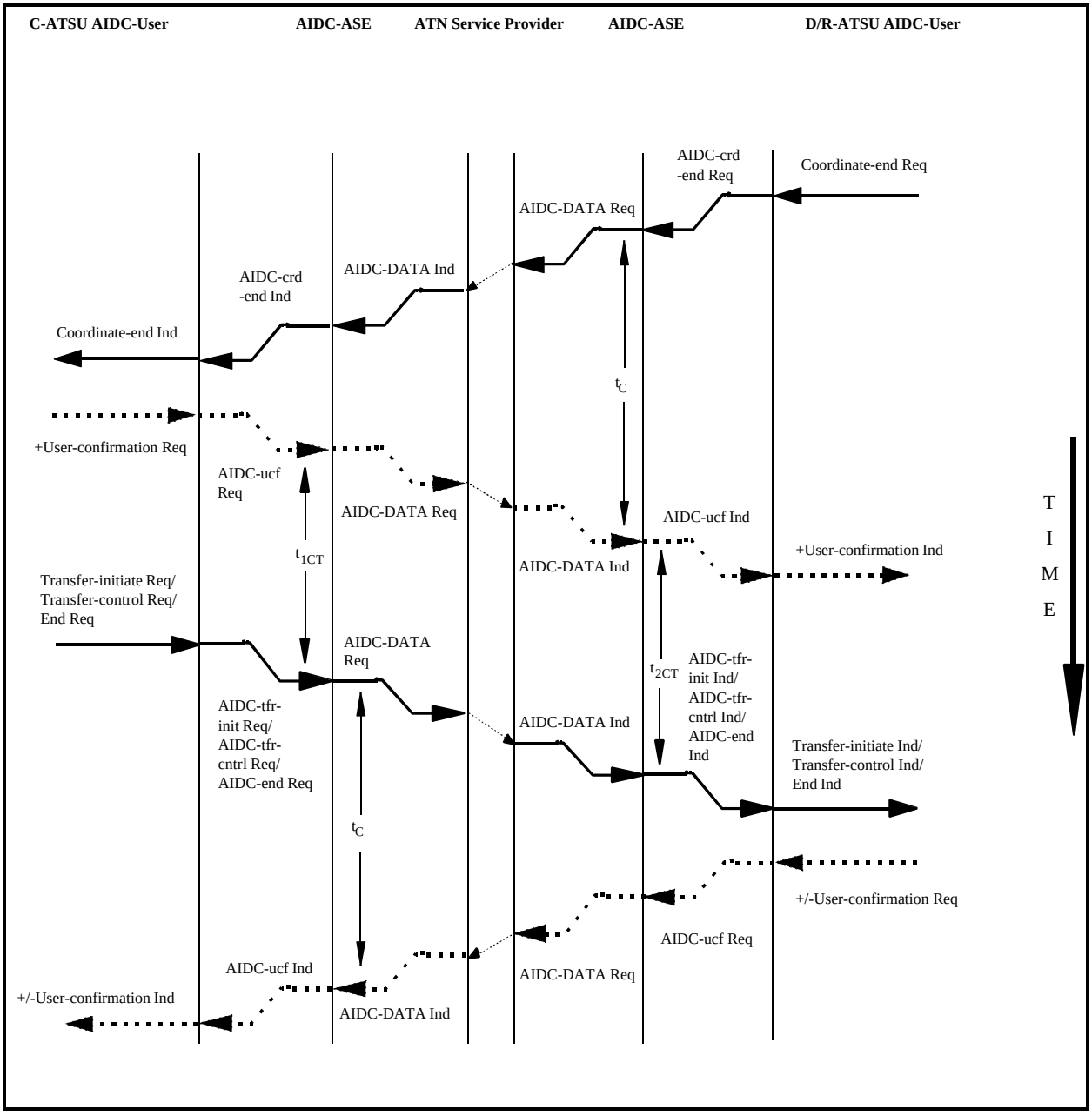

Figure 3.2.10-9: Sequence Diagram showing the start of the Transferring Regime through the invocation of the Transfer-initiate service

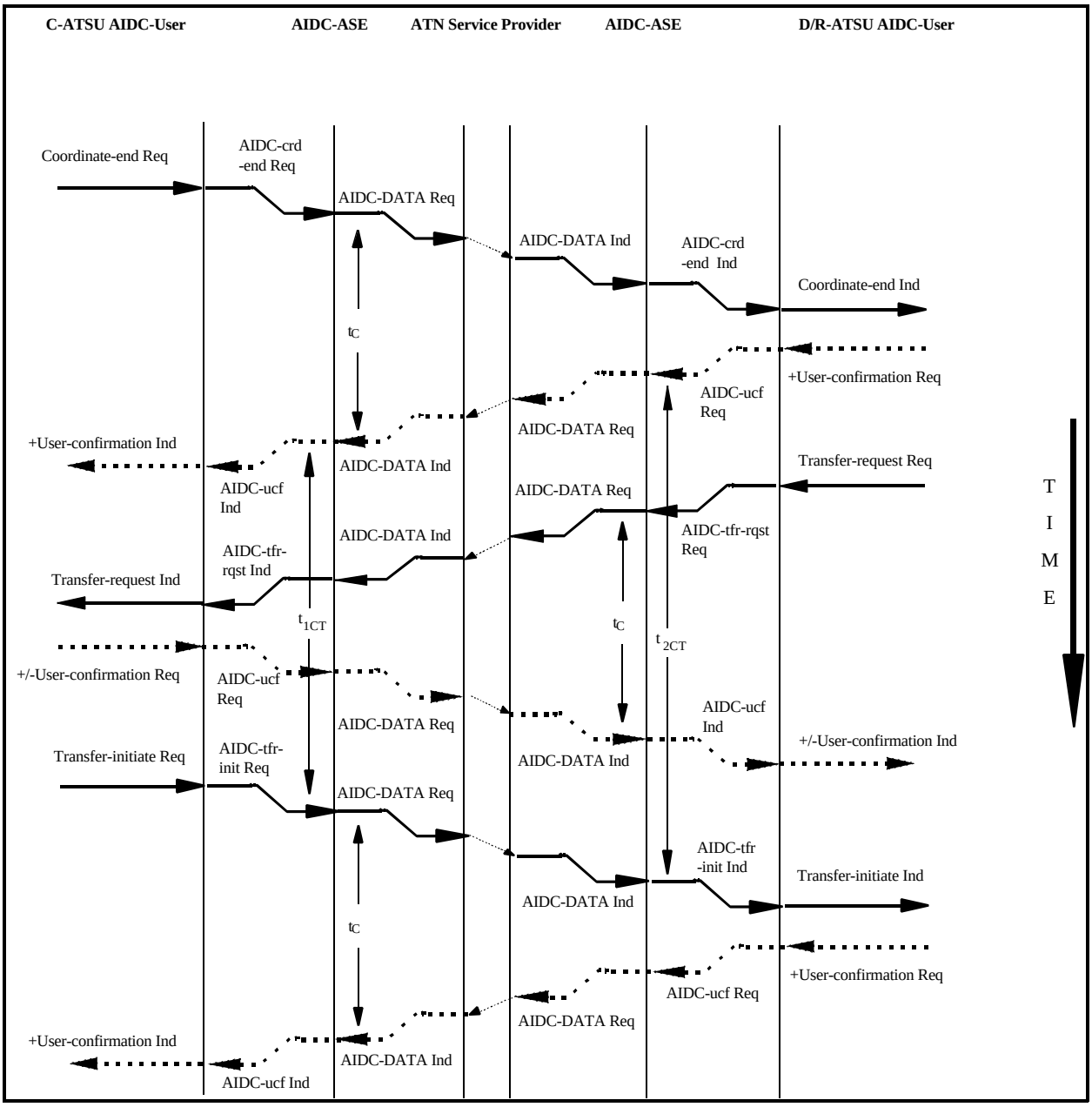

Figure 3.2.10-10: Sequence Diagram showing the invocation of the Transfer-request service

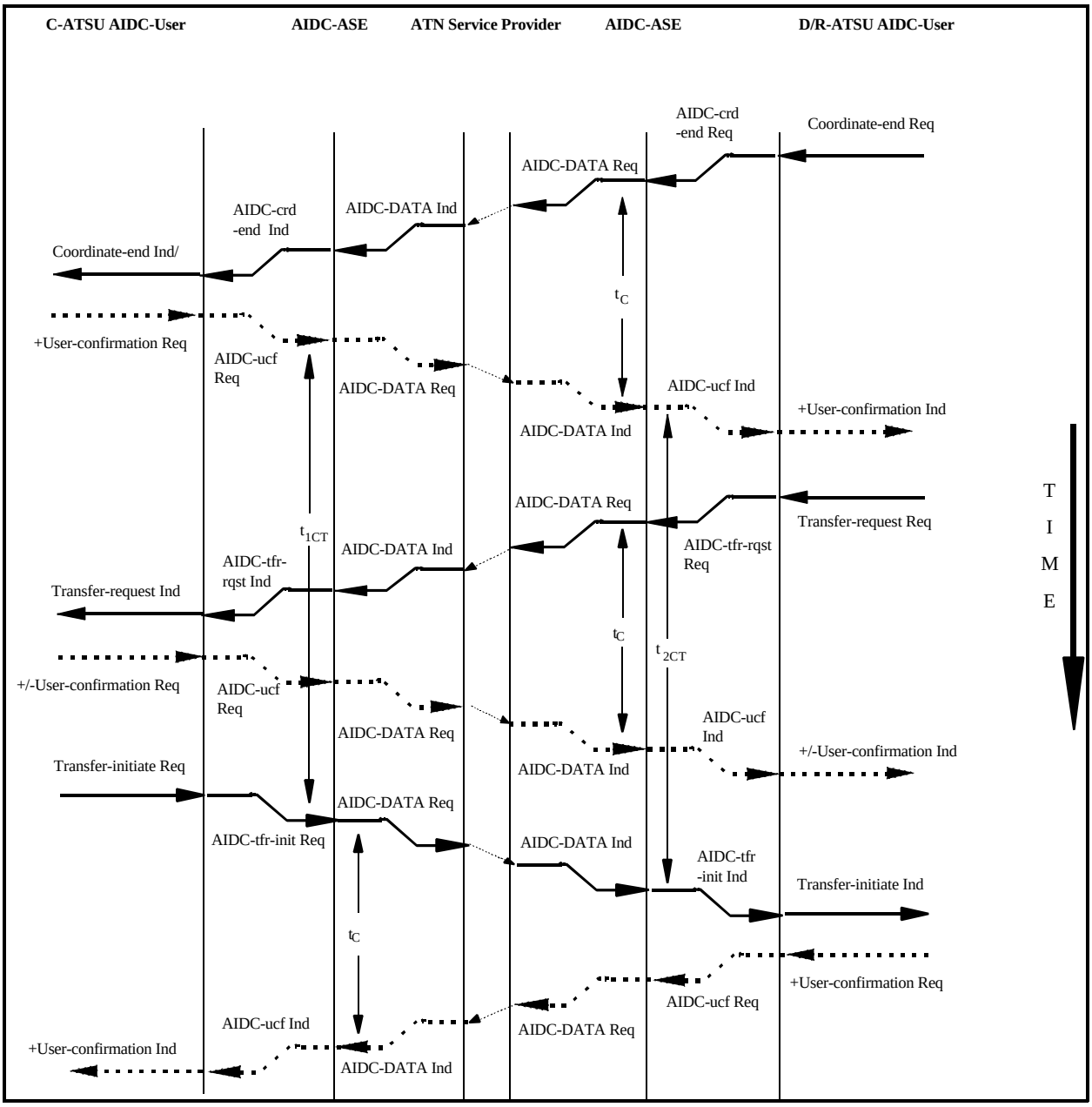

Figure 3.2.10-11: Sequence Diagram showing the invocation of the Transfer-request service

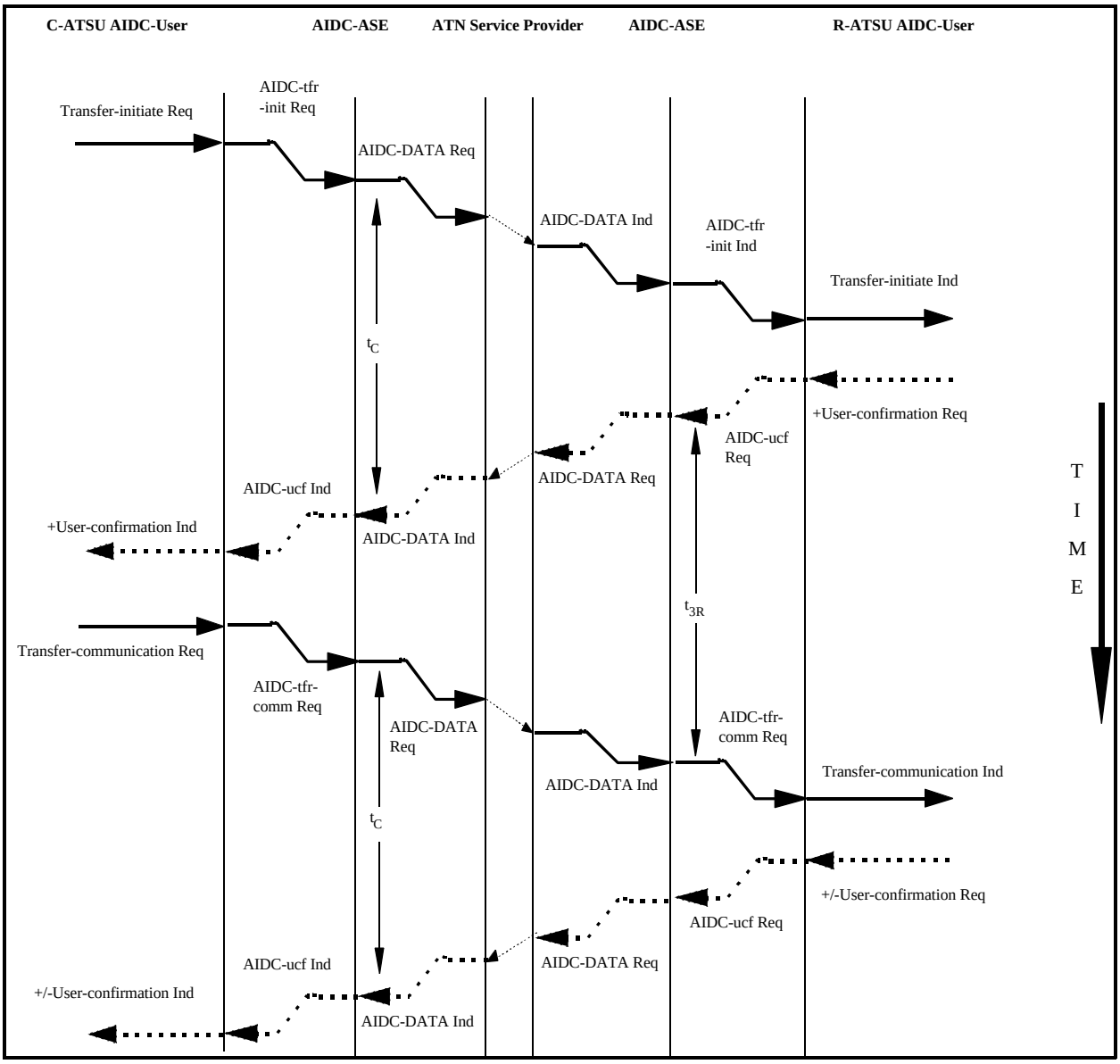

Figure 3.2.10-12: Sequence Diagram showing the invocaton of the Transfer-communication service after the Transfer-initiate service

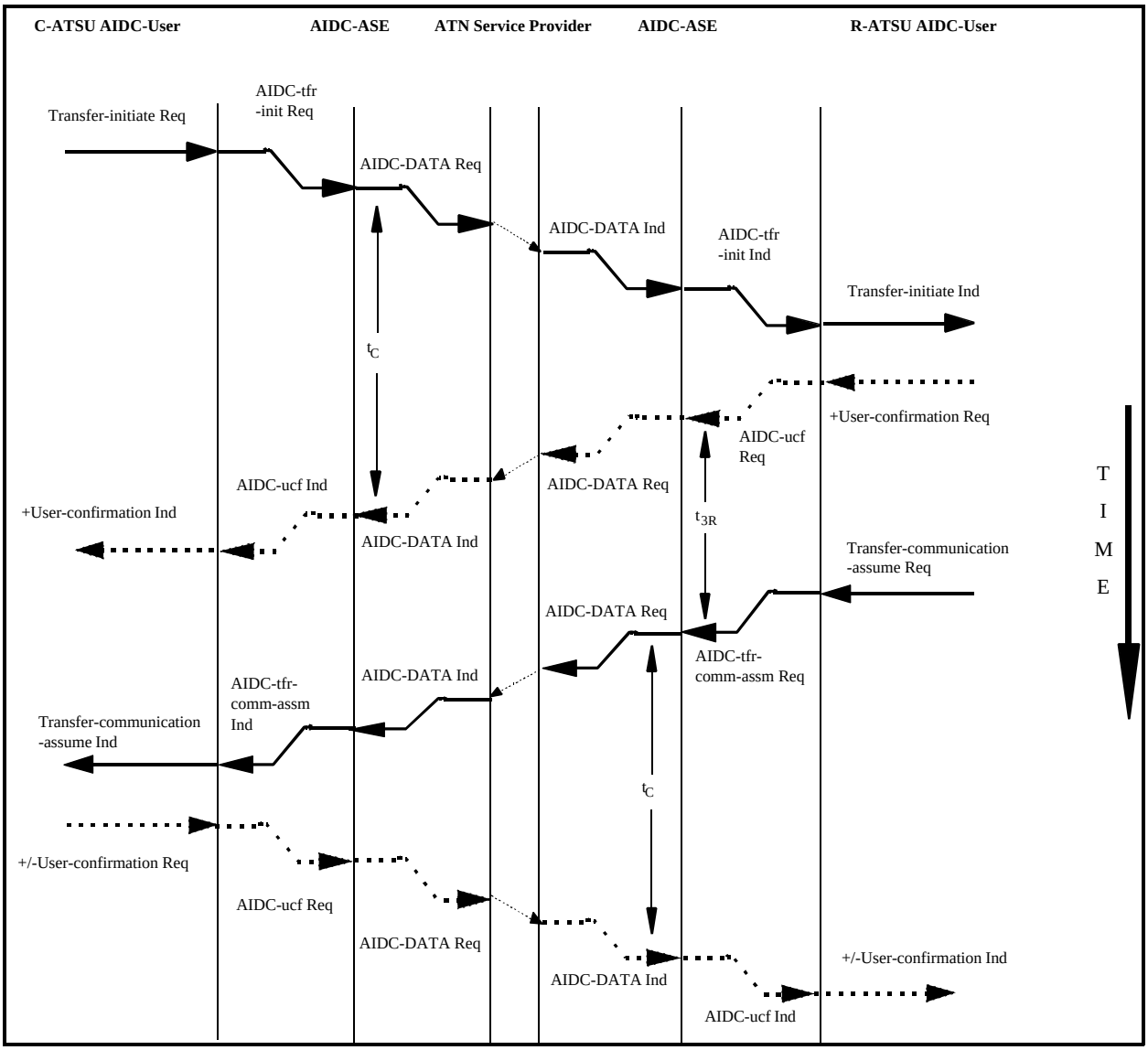

Figure 3.2.10-13: Sequence Diagram showing the invocation of the Transfer-communication-assume service after the Transfer-initiate service

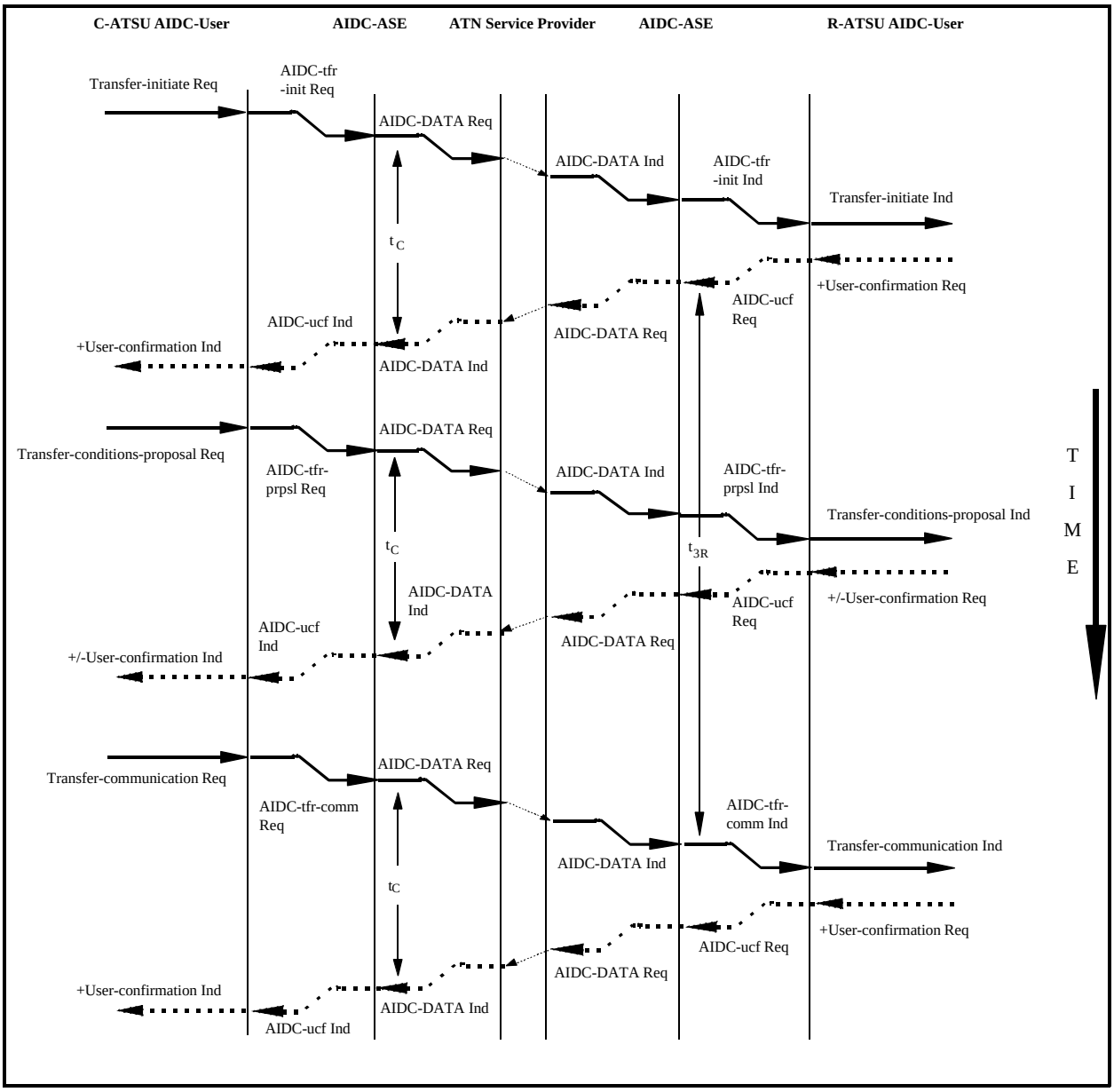

Figure 3.2.10-14: Sequence Diagram showing the invocation of the Transfer-conditions-proposal service, with predicate P1 false, followed by Transfer-communication service

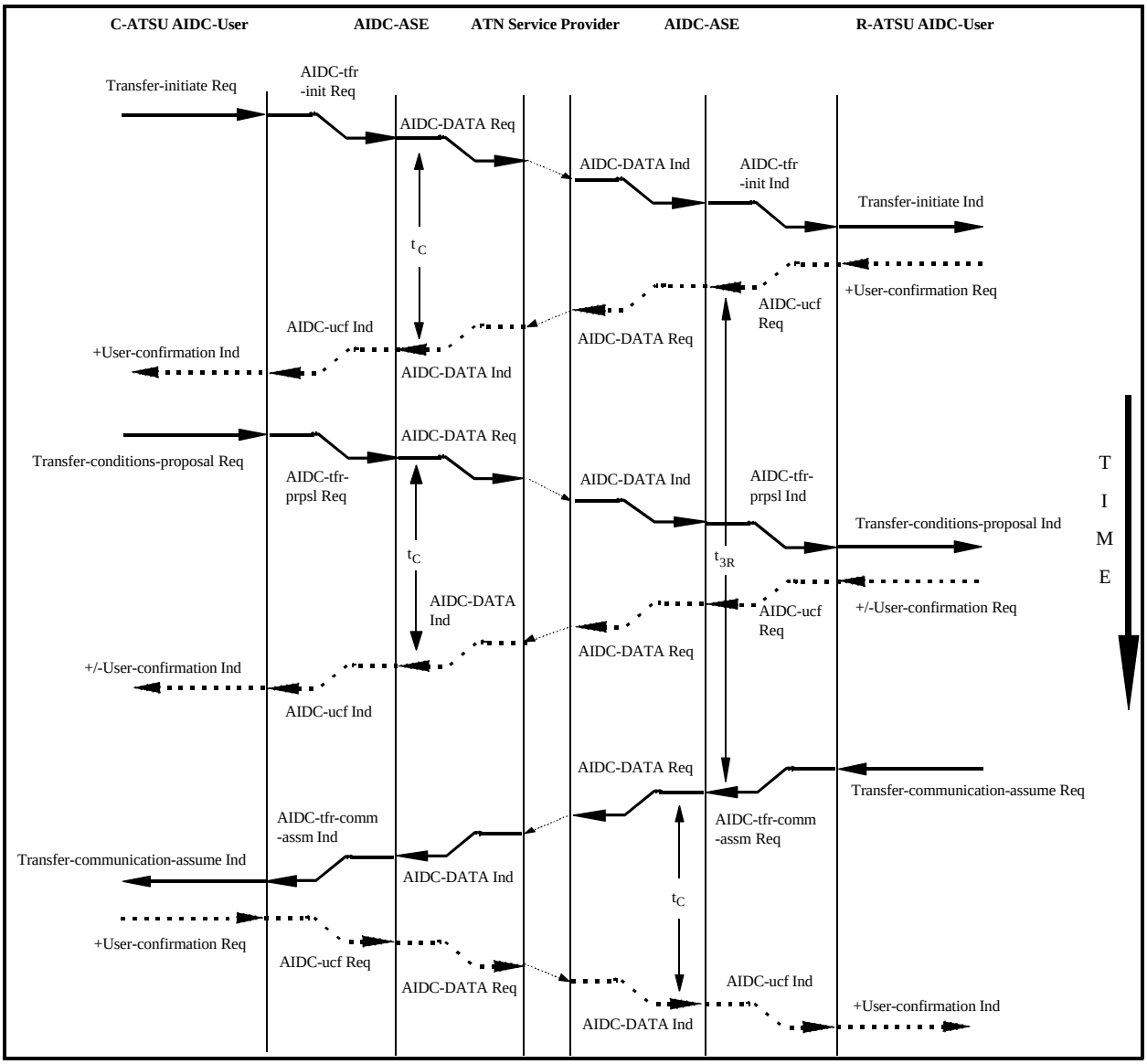

Figure 3.2.10-15: Sequence Diagram showing the invocation of the Transfer-conditions-proposal service, with the predicate P1 false, followed by the invocation of the Transfer-communication-assume service

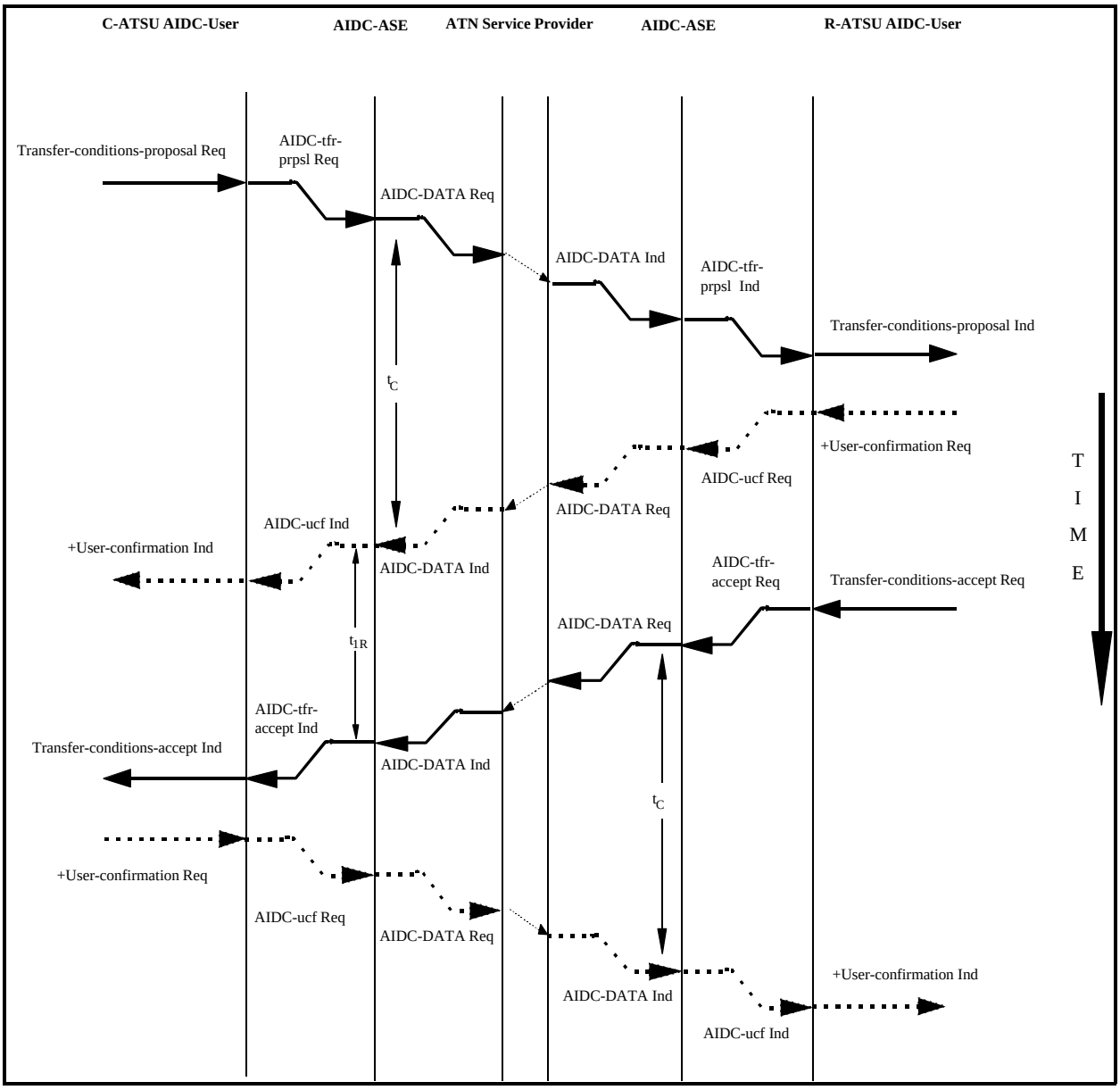

Figure 3.2.10-16: Sequence Diagram showing the invocation of the Transfer-conditions-proposal service and the Transfer-conditions-accept service with the predicate P1 true

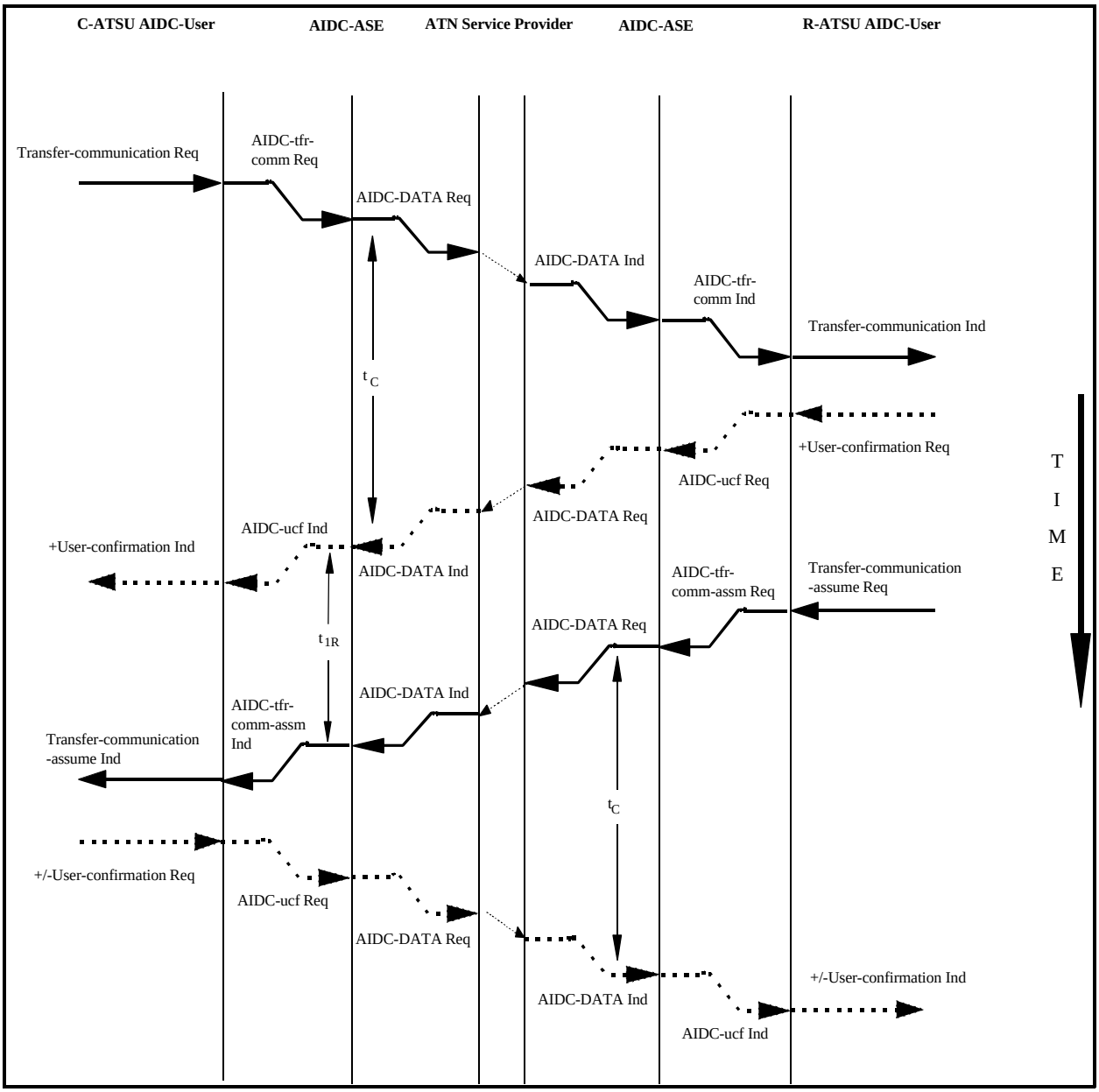

Figure 3.2.10-17: Sequence Diagram showing the invocation of the Transfer-communications service with the invocation of the Transfer-communications-assume service

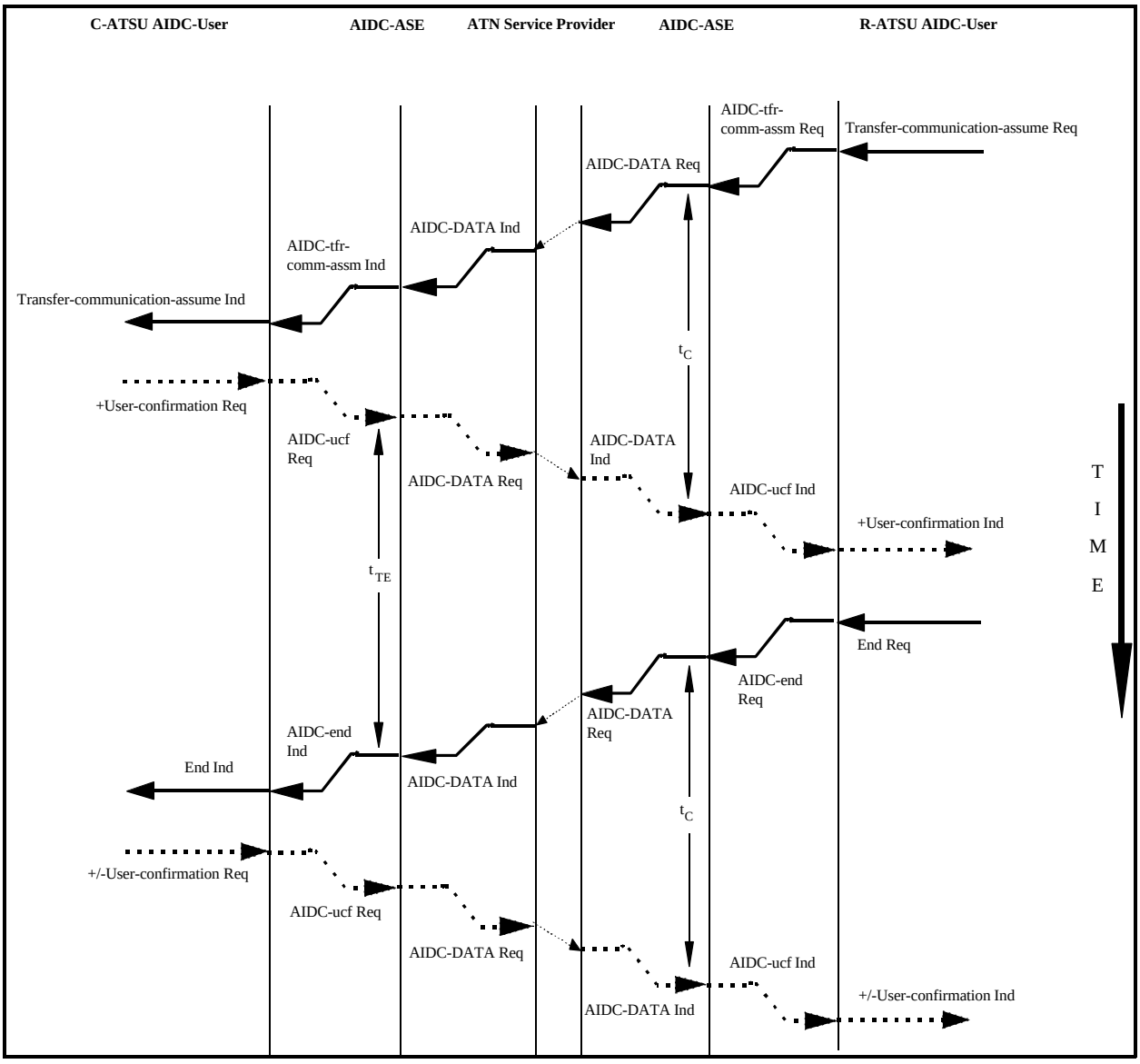

Figure 3.2.10-18: Sequence Diagram showing the end of the AIDC service through the invocation of the End service after the Transfer-communication-assume service

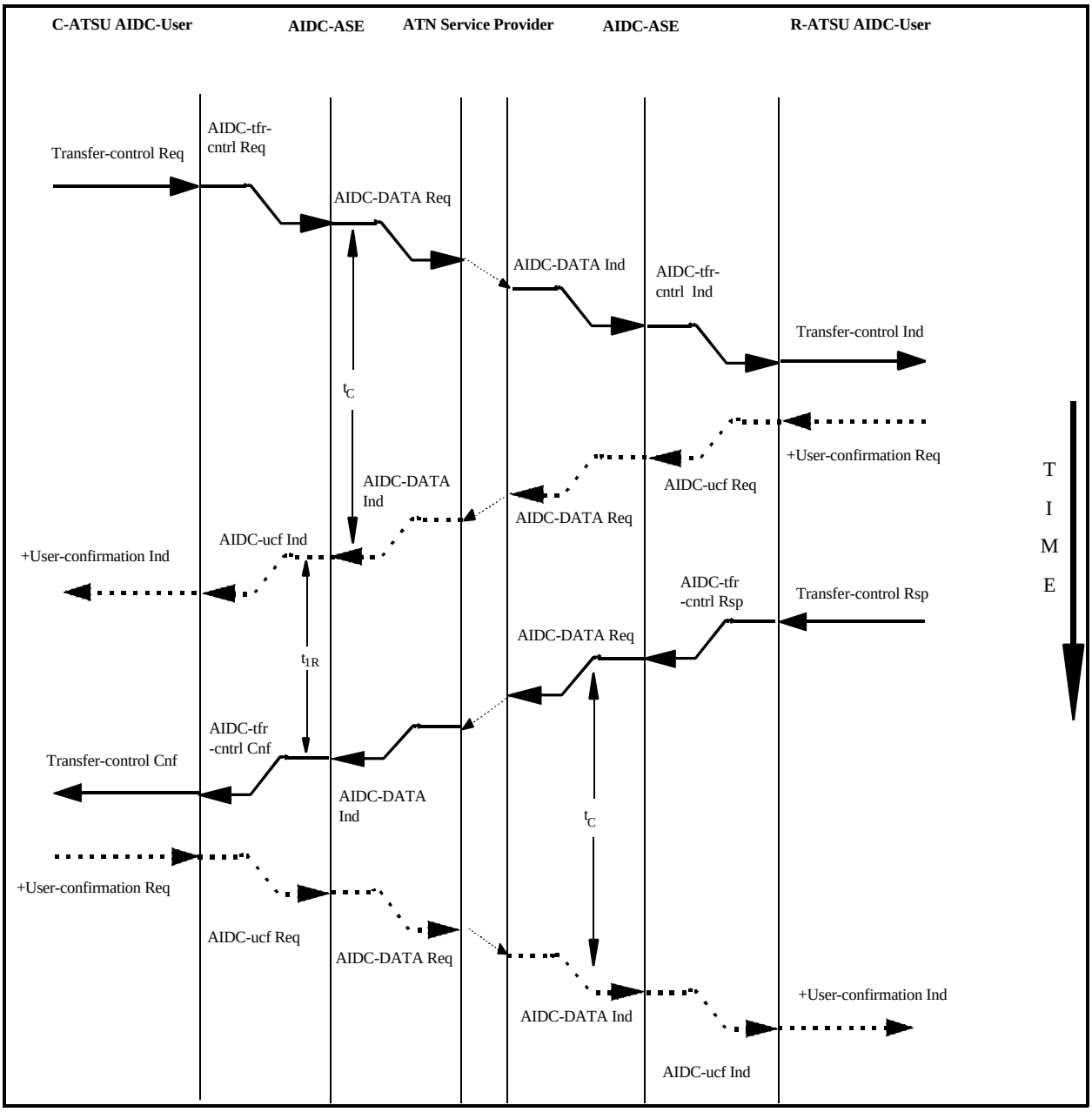

Figure 3.2.10-19: Sequence Diagram showing the start and end of the Transferring Regime through the invocation of the Transfer-control service

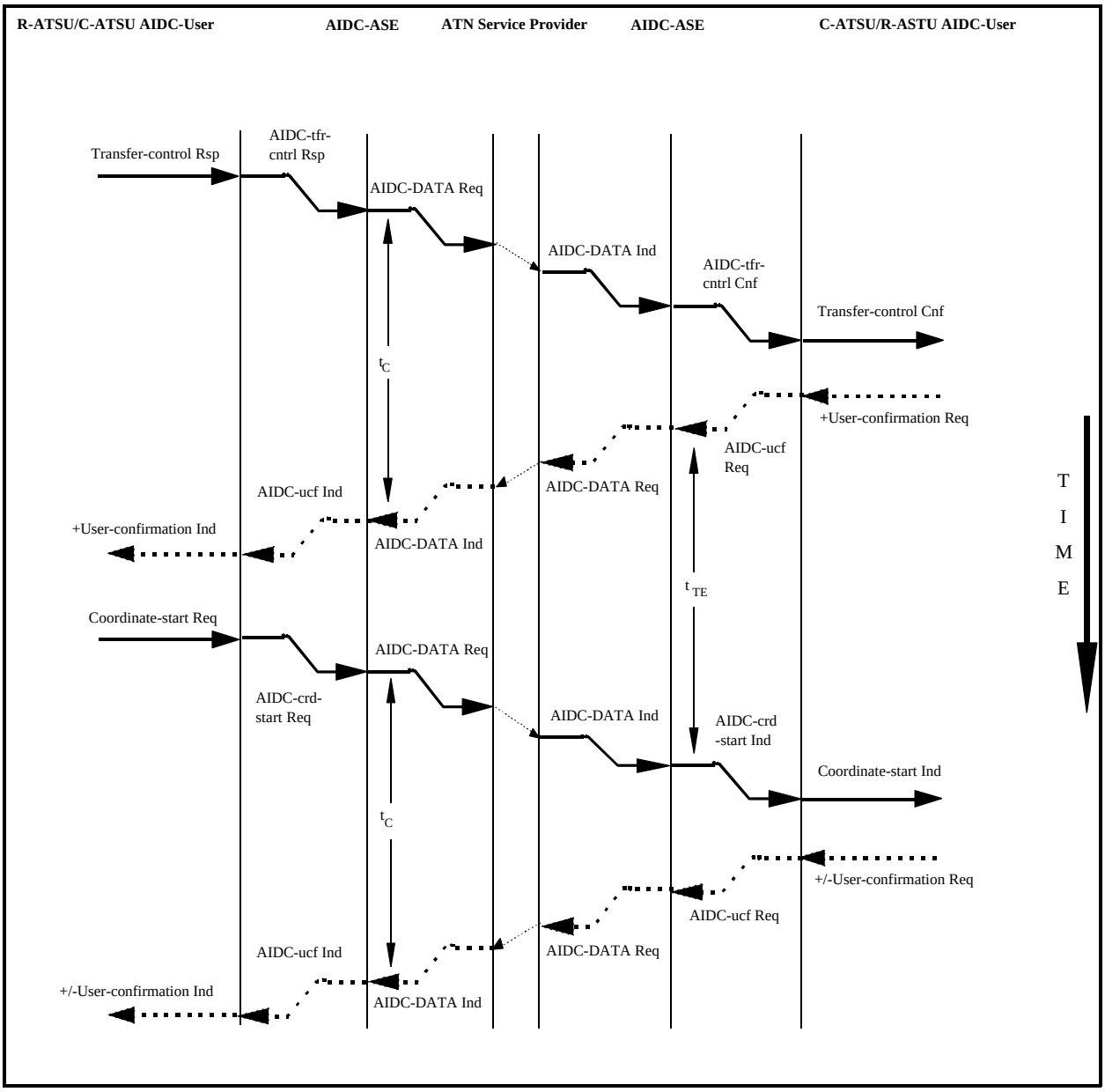

Figure 3.2.10-20: Sequence Diagram showing the re-entry into the Coordinating Regime, after the end of the Transferring Regime, through the invocation of the Coordinate-start service

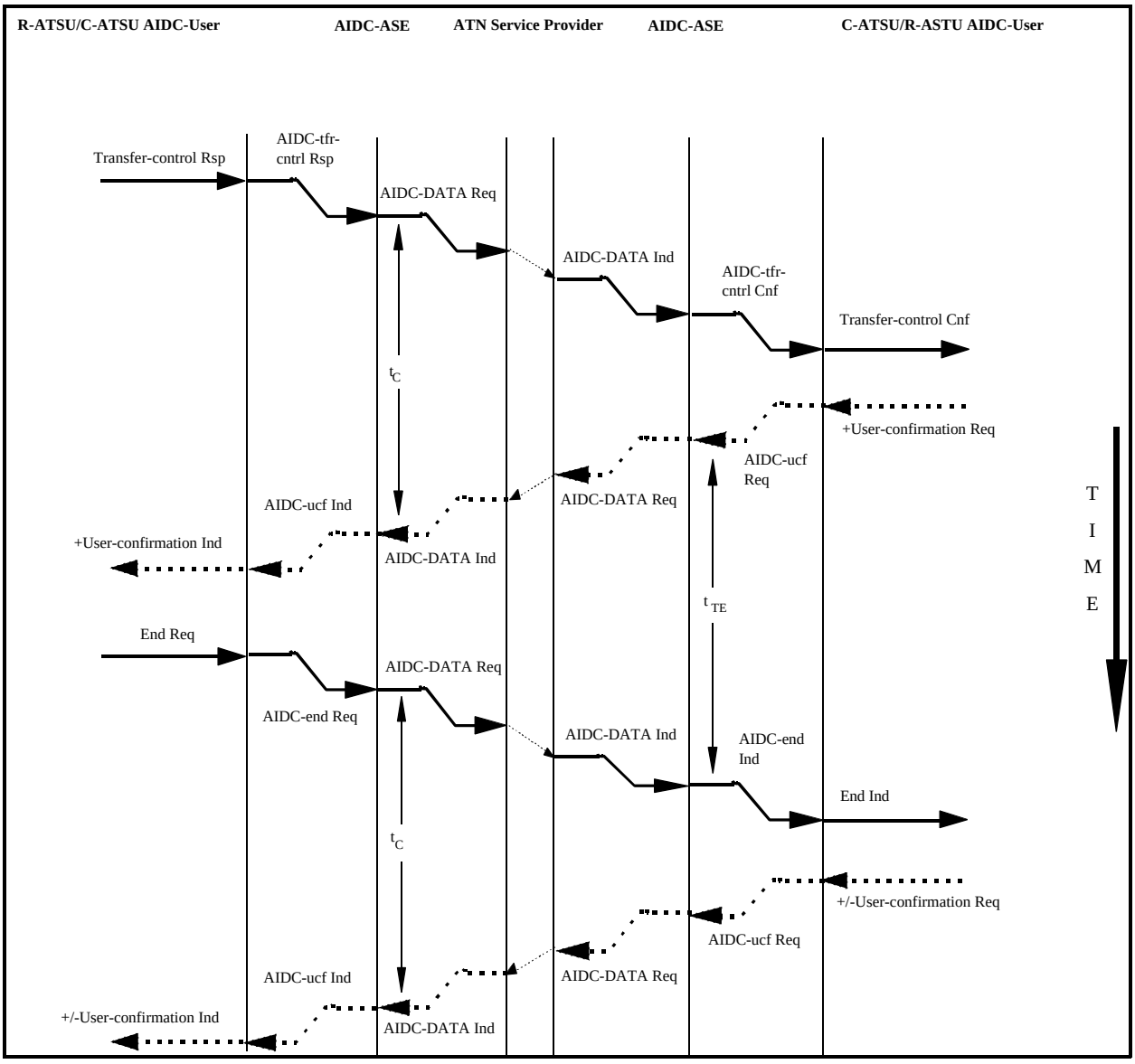

Figure 3.2.10-21: Sequence Diagram showing the end of the AIDC service after the Transfer-control service

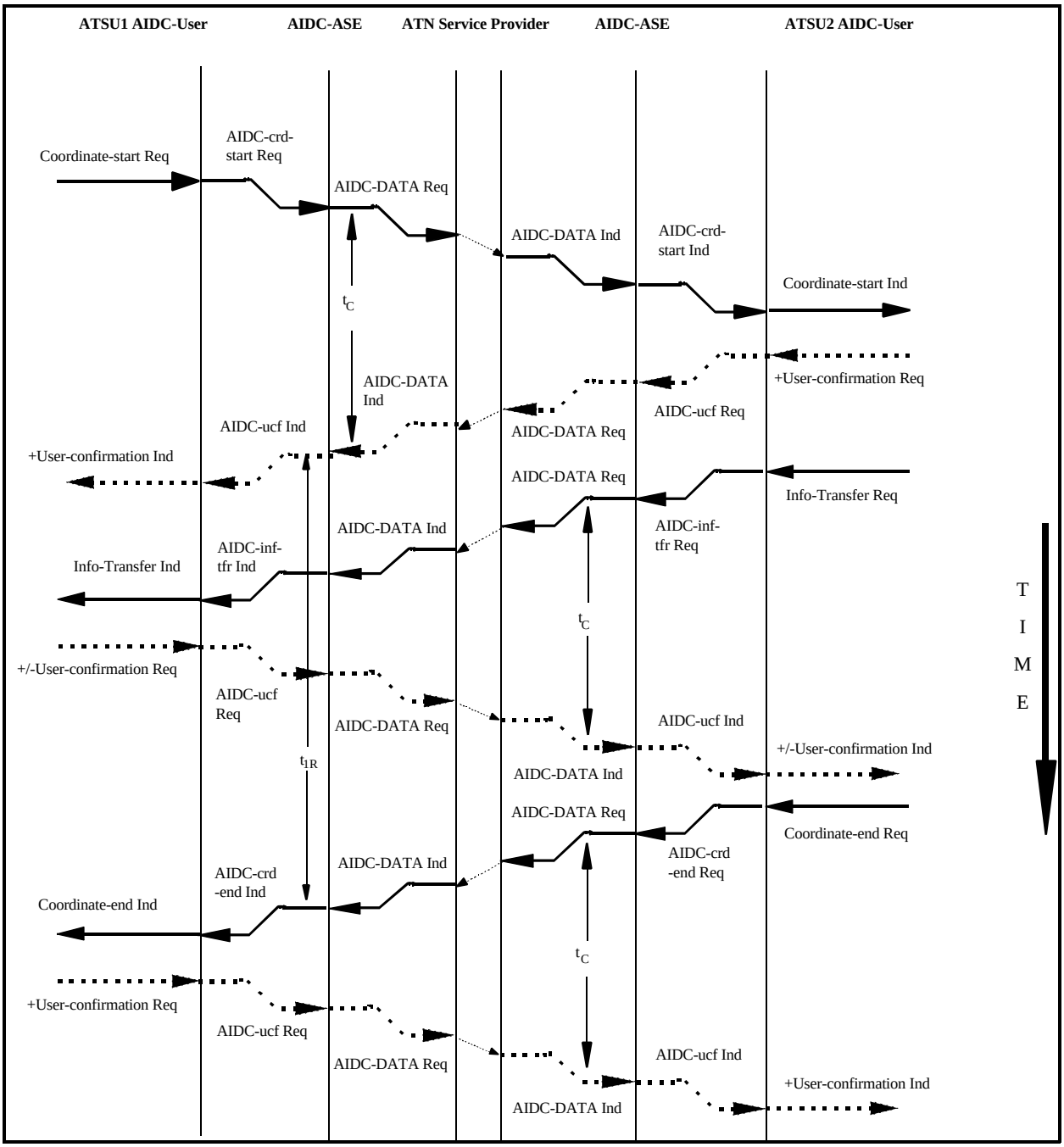

Figure 3.2.10-22: Sequence Diagram showing the invocation of the Info-transfer service in the Coordinating Regime

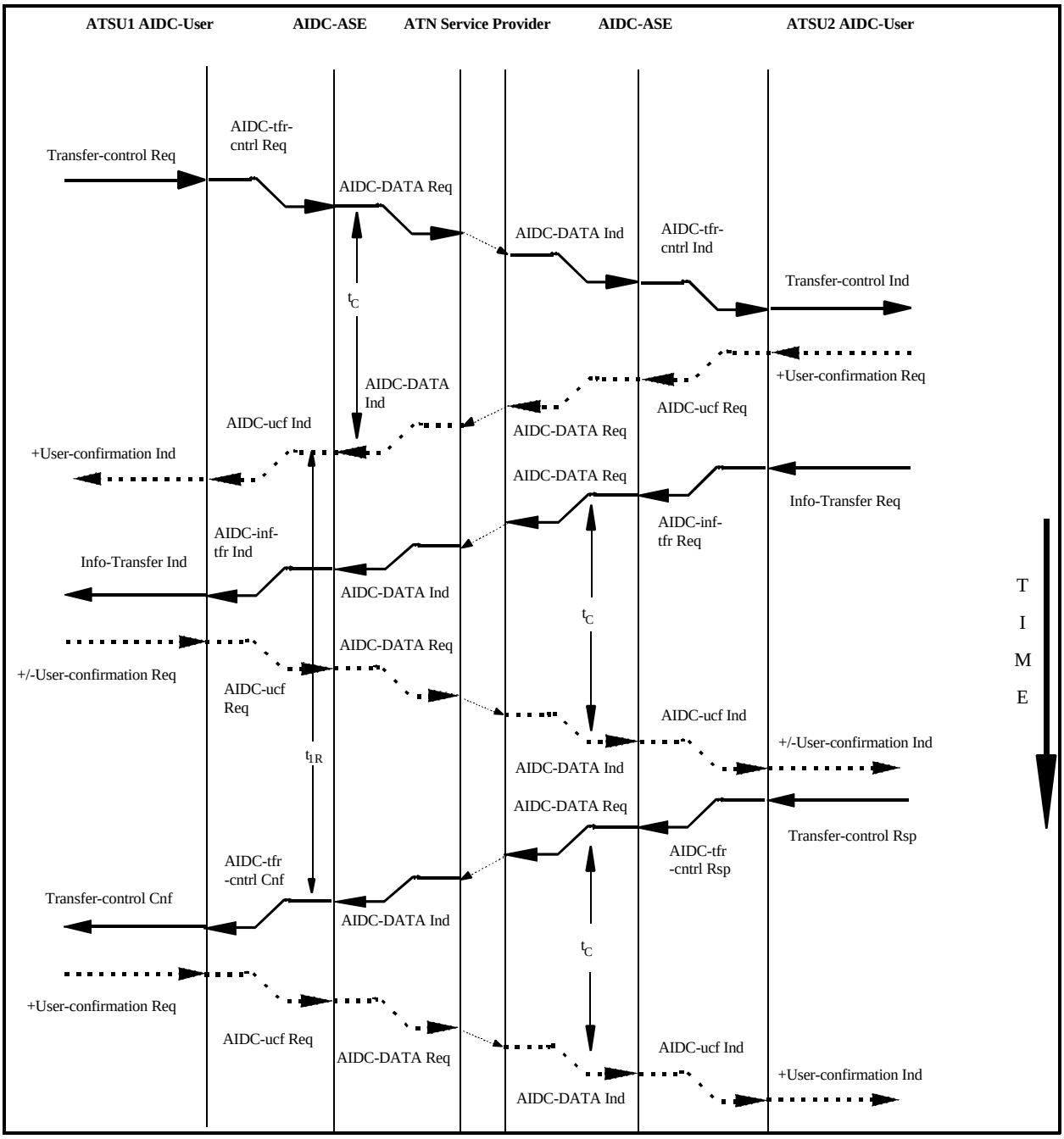

Figure 3.2.10-23: Sequence Diagram showing the invocation of the Info-transfer service in the Transferrring Regime

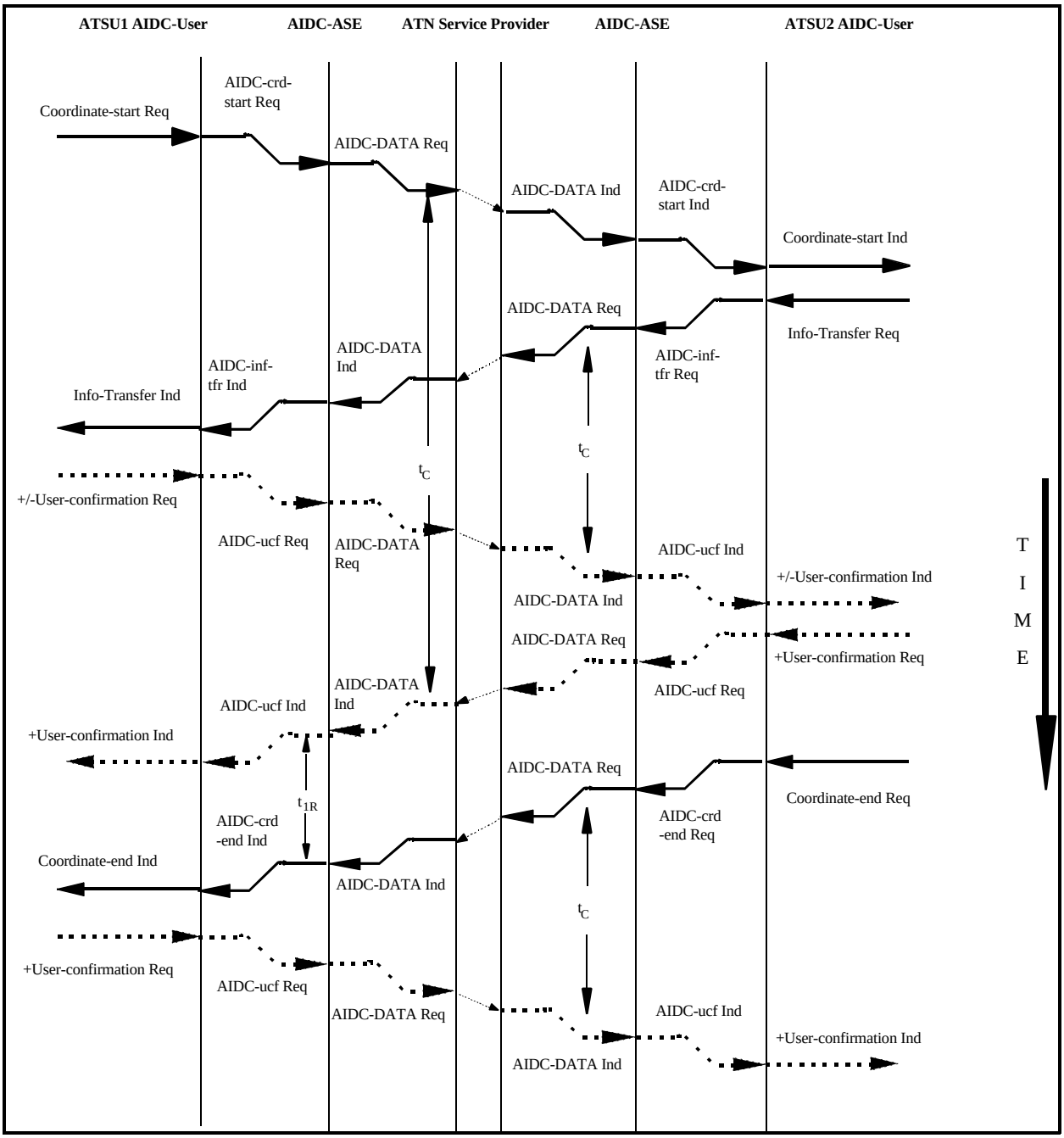

Figure 3.2.10-24: Sequence Diagram showing the invocation of the Info-transfer service before the User Confirmation of the Coordinate-start service

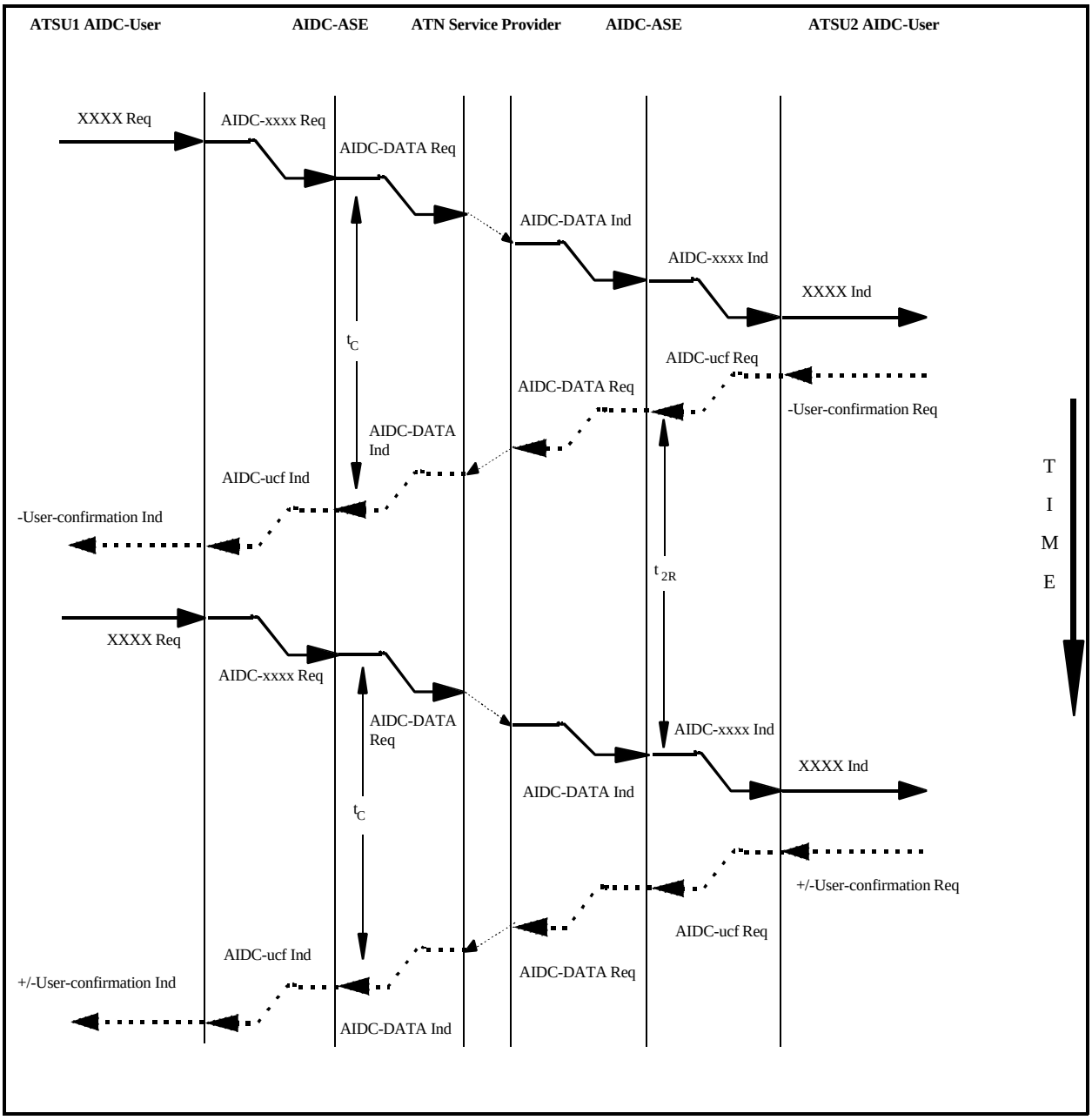

Figure 3.2.10-25: Sequence Diagram showing the sequence for a negative User Confirmation for all services except the Transfer-request, Transfer-conditions-proposal, Transfer-conditions-accept, and Info-Transfer services

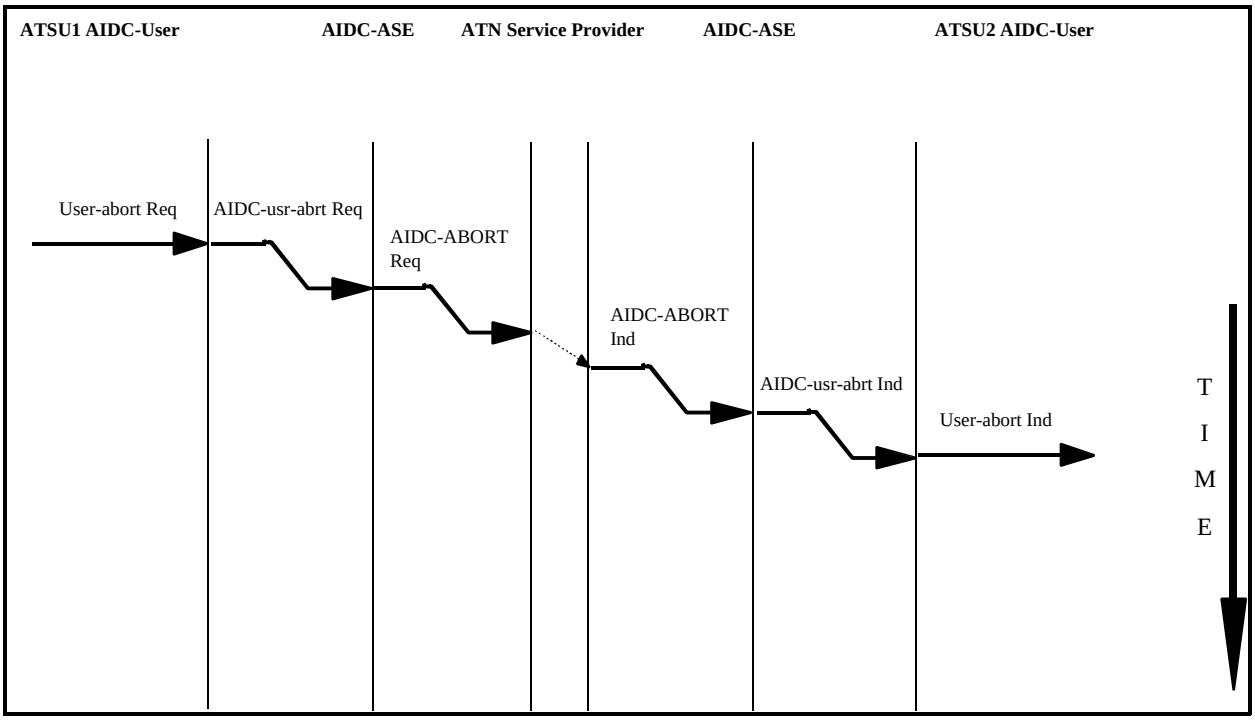

Figure 3.2.10-26: Sequence Diagram for the User-abort service

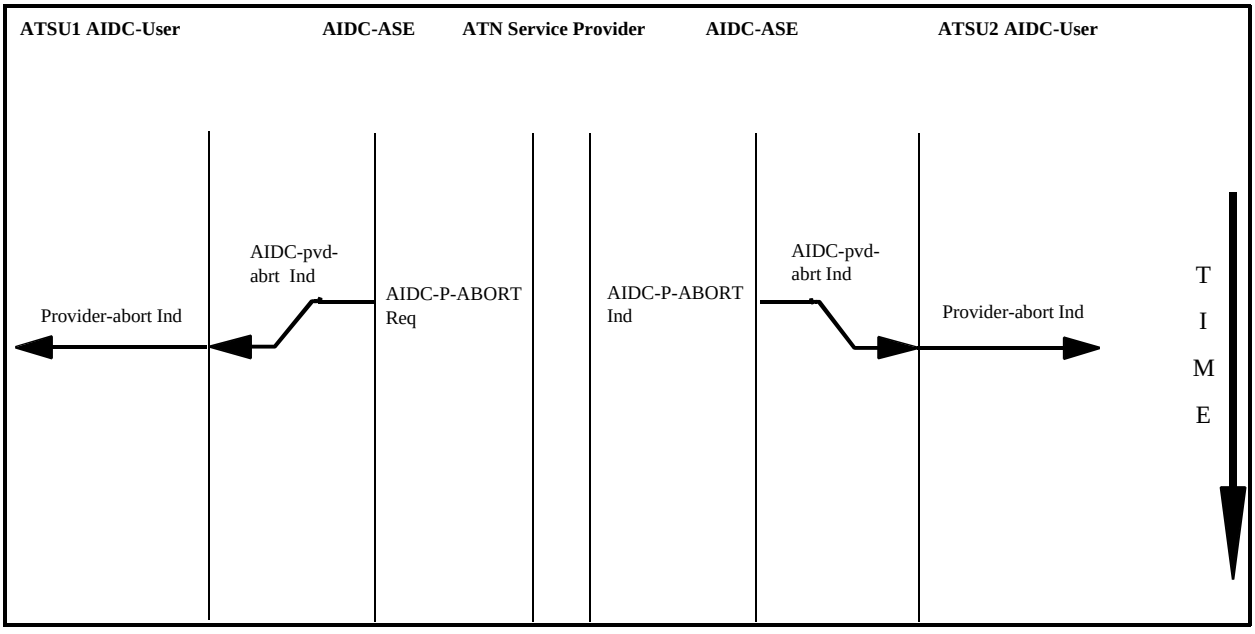

Figure 3.2.10-27: Sequence Diagram showing the invocation of the Provider-abort service invoked by the ATN service provider

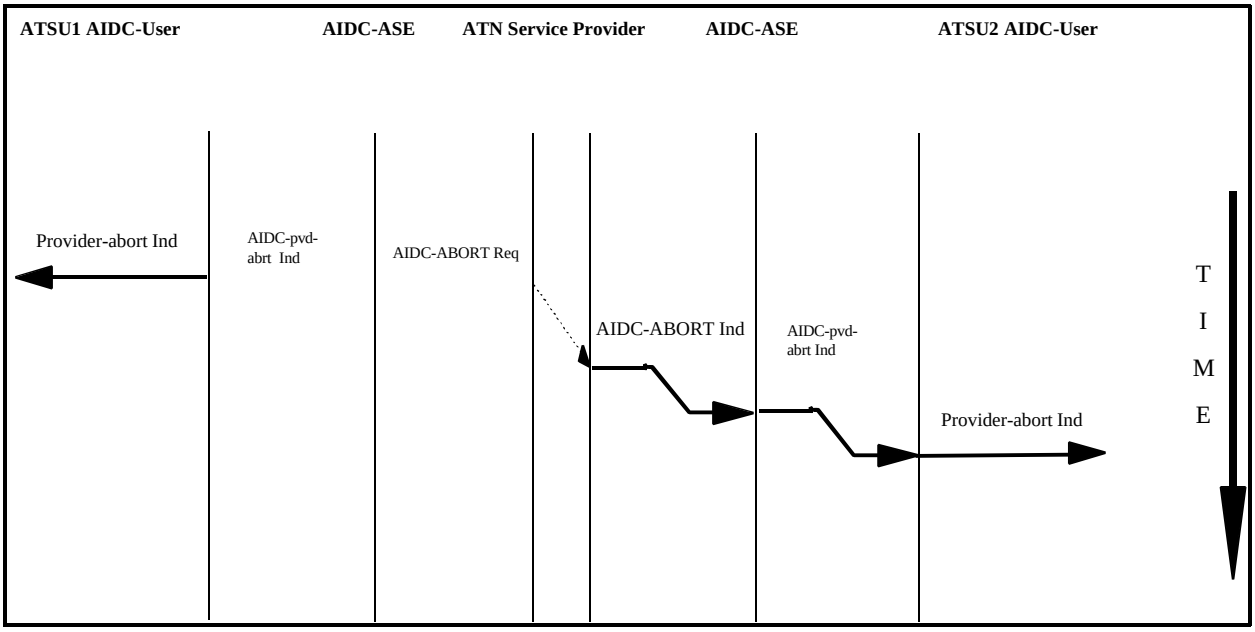

Figure 3.2.10-28: Sequence Diagram showing the service invocation sequence when the AIDC-ASE aborts