AERONAUTICAL TELECOMMUNICATION NETWORK PANEL

WORKING GROUP 3 (APPLICATIONS AND UPPER LAYERS)

Sub Group 2 (Air/Ground Applications)

Part 1: Context Management Application SARPs

Prepared by: Air/Ground Applications Subgroup

SUMMARY

This document is the draft of the Context Management Application SARPs for the ATN CNS/ATM-1 Package.

CONFIGURATION SHEET

Title : Draft Context Management SARPs for the ATN CNS/ATM-1 Package

Version : Proposed Version 3.0 (Brussels)~~2.0~~

Date: 22 April~~15 February~~ 1996

Contact: Airservices Australia, Eurocontrol, Federal Aviation Administration, Direction Generale de l'Aviation Civile, UK National Air Traffic Services

Status: Draft

Change History :

Version	Description	Affected Parts	Date
1.0	Banff Proposed Draft Context Management SARPs		6 Oct 1995
1.1	Input to ATNP/WG3 at Brisbane		5 Feb 1996
	Chapter 2		
	• Time accuracy clarified	§2.2.1	
	Chapter 3		
	• CM-end service added due to dialogue changes	§3.3.1, §3.8	
	• Maintain Dialogue parameter added to CM-logon service	§3.5.8	
	• CM-update service changes resulting from dialogue changes	§3.6	
	• CM-contact service changes resulting from dialogue changes	§3.7	
	• ~~CM-end service added~~	~~§3.8~~	
	Chapter 4		
	• The cmUpdate element was added for clarification purposes	§4.1, §5.3.2.4.1.1, §5.2.3.5.1.1, §5.3.3.3.1	
	• Abort reasons added due to dialogue changes	§4.1, §5.3.2.7, §5.3.2.8, §5.3.2.9, §5.3.3.10, §5.3.3.11, §5.3.3.12, all of §5.4	
	• Application name and version number clarified	§4.1	
	• Long TSAP clarifications, specifically adding ARS	§4.1	
	Chapter 5		
	• The cmUpdate element was added for clarification purposes	§5.3.2.4.1.1, §5.2.3.5.1.1, §5.3.3.3.1	
	• Abort reasons added due to dialogue changes	§5.3.2.7, §5.3.2.8, §5.3.2.9, §5.3.3.10, §5.3.3.11, §5.3.3.12, all of §5.4	
	• CM-update time sequence diagrams added	§5.1.4, Figures 5-3, 5-4	
	• Textual changes and new time sequence diagrams for dialogue changes	§5.1.5, Figures 5-5, 5-6, 5-7, 5-8, 5-9, 5-10 §5.3.2.2.2, §5.3.2.3.1.1, §5.3.2.5, §5.3.2.5.1.2, §5.3.2.6, §5.3.3.3.2 §5.3.3.4.2, §5.3.3.6, §5.3.3.9	

	• Addition of CM-end service text and time sequence diagrams	§5.1.6, Figure 5-11, §5.3.3.8	
	• Addition of CM-end timer	§5.2, Table 5-1	
	• Due to modified D-ABORT (new name, new variables) all of section 5.4 was deleted and rewritten.	§5.4	
	Chapter 6		
	• Addition of D-DATA, D-END	§6.2.1.1	
	• Addition of note for ATSC value references	§6.2.2.3.1	
	Chapter 7		
	• CM-logon request information clarified	§7.1.3.1	
	• Changes resulting from dialogue updates added	§7.2.2.4, , §7.2.2.8, §7.2.4.2, §7.2.5, §7.2.6	
	• Changes resulting from address/version number updates added	§7.2.2.3, §7.2.3.2	
	• CM-end service requirements added	§7.2.7	
	Appendix A		
	• State tables updated for dialogue changes	§A.2, Tables A-1 and A-2	
2.0	Output ~~from~~ Brisbane Draft SARPs		15 Feb 1996
	Chapter 1		
	• Addition of acronyms and definitions relating to long/short/actual TSAP addresses	§1.1.4.1, §1.1.4.2	
	• Changed reference of traffic type values to "[5]"	§1.1.5.1, §3.5.7.1, §3.6.5.1, §3.7.6.1, §3.9.7.1, §6.2.2.3.1	
	• Addition of Forward Function (FD4)	§1.2.2.1	
	• Addition of Forward Functionality	§1.1.2.1, §1.1.2.2, §1.1.4.2.1, §1.3.4	
	Chapter 2		
	• Timing requirements clarified	§2.2	
	Chapter 3		
	• Addition of CM-forward service	§3.3.1	
	• Insertion of new section (prior to CM-abort services) that provides the description of the CM-forward service	§3.9	
	Chapter 4		
	• Automatic tagging/removing of tags in ASN.1	§4.1	
	• Alphabetising ASN.1	§4.1	
	• Renaming CMDownlinkMessage to CMAircraftMessage, CMUplinkMessage to CMGroundMessage in order to allow for ground-ground message transfer	§5.3.2.5.1.1, §5.3.2.6.1, §5.3.2.6.2, §5.4.1.1, §5.4.2.1, §5.4.3.1, §5.4.3.2, §5.4.4.1, §5.4.4.2, §5.4.5.2, §5.3.3.6.1, §5.3.3.7.1, §5.3.3.7.2, §5.3.3.8.1, §5.3.3.8.2, §5.3.3.10.1	
	• Addition of appropriate ASN.1 for the CM-forward service	§5.4.5.1, §5.4.6.1 ~~§3.9.6.1~~, §5.3.3.1.2 §5.3.3.10.1	
	• Full definition of TSAP addresses, both long and short	§7.1.3.1, §7.1.3.2, §7.2.2.2, §7.2.2.5, §7.2.6.4	
	• Definition of logon unsuccessful response	§7.2.2.7	
	• Passing all version numbers/ addresses for each application	§7.1.3.1, §7.1.3.3,	

		§7.2.2.4, §7.2.3.2	
	• ICAO facility designator modified from 4 characters to 8	§3.5.2.1, §3.6.3.1, §3.7.3.1~~, §3.9.2.1, §3.9.4~~	
	Chapter 5		
	• Addition of CM-forward service	§5.1.7	
	• Addition of abort time sequence diagrams for ground-ground case	§5.1.10, §5.1.11, Figures 5-18, 5-19, 5-20, 5-21, 5-22	
	• Addition of CM-forward timer	§5.2, Table 5-1	
	• General re-ordering of section 5.3, so the dialogue service primitives are defined first followed by the CM service primitives (for both air and ground protocol descriptions)	§5.3	
	• Addition to D-START cnf and D-START ind requirements for inclusion of CM-forward	§5.3.3.2.1.3, §5.3.3.1.2	
	• Addition of CM-forward Request	§5.3.3.10	
	• State tables moved from Appendix A to section 5.5	§5.5	
	• Addition of FORWARD state to Table 5-2 along with additional forward events and forward timer	§5.5, Table 5-2	
	Chapter 6		
	• NULL removal from QOS table	§6.2.2.3.1, Table 6-1	
	• Addition of RER clarification (Note 2)	§6.2.2.3.1	
	Chapter 7		
	• Long/Short TSAP clarifications for Air and Ground user requirements	§7.1.3.1, §7.1.3.2, §7.2.2.2, §7.2.2.5, §7.2.6.4	
	• Addition of CM-forward service requirements	§7.2.6	
	• Passing all version numbers/ addresses for each application (CM-logon request CM-logon response and CM-forward)	§7.1.3.1, §7.2.2.4, §7.2.6.2	
	Appendix A		
	• State table moved to Chapter 5; appendix deleted	~~§~~Appendix A	
2.1	Input to ATNP/WG3 at Brussels		22 April 1996
	Chapter 1		
	• RCP acronym removed	§1.1.4.1, §2.1	
	• ARS definition added to clarify its relationship to the 24 bit ICAO airframe identifier	§1.1.4.2.1, §4.1	
	• TSAP definitions clarified by adding “address”	§1.1.4.2.1	
	• Document references updated	§1.1.5.1	
	• Descriptions generalized to accommodate ground-ground capability	§1.3.1.2, §1.3.1.5.1, §1.3.5.2.1	
	• Logon success/failure concept removed	§1.3.1.5.2, §3.5.6, §4.1, §7.2.2.4, §7.2.2.7	
	• CMForward is now the same in concept as CMLogon	§1.3.4.2.1, §4.1	
	Chapter 2		
	• Reference to RCP removed	§2.1	
	• Security requirement made into a note since there was no “shall”	§2.3	
	Chapter 3		
	• Notes clarified	§3.5.4.2, §3.5.6, §3.6, §3.7, §3.8, §3.9, §3.9.2, §3.9.3, §3.9.5.2, §3.10, §3.11	
	• Classes of Communication “I” and “J” added	§3.5.7.1, §3.6.5.1, §3.7.6.1, §3.9.7.1, Table	

	6-1
• Abstract value of “accepted” changed to “maintain”	§3.5.8.1
• Forward service changed to be confirmed instead of unconfirmed	§3.9.1, §4.1, §5.1.7, §5.3.3.2.1.3, §5.5
• Result parameter added for Forward service	§3.9.8
Chapter 4	
• Title clarified	§4
• Text clarifications	§4.1.1
• Automatic tag functionality deleted, tags added	§4.1.1
• CMForwardRequest now equals CMLogonRequest	§4.1.1
• TSAP addressing clarified	§4.1.1
• Response parameter removed from CMLogonResponse	§4.1.1
• Logon successful/not successful removed	§4.1.1
• Added commas to last items before extensibility markers	§4.1.1
Chapter 5	
• Figure 5-12 modified to allow for Forward service becoming confirmed	§5.1.7
• Section title made consistent with other SARPs	§5.2
• Timer use clarified	§5.2.1
• Timer requirement made a recommended practice	§5.2.2
• Text clarified	§5.3.1.1, §5.3.1.1.2, §5.3.3.1.3, §5.3.3.1.2, §5.3.3.2.12, §5.4.11, §5.4.2.1, §5.4.3.1
• Notes added to list available states	§5.3.2, §5.3.3
• Initial IDLE state requirement added	§5.3.2.1, §5.3.3.1
• Redundant “shalls” removed	§5.3.3.2.2.1.1, §5.3.3.2.1.1
• Result parameter functionality added for Forward service	§5.3.3.2.1.3, §5.4.3.2, §5.4.4.2, §5.4.5.1
• Timer text clarified	§5.4.1.1, §5.4.4.1, §5.4.5.2, §5.4.6.1
• State table modified to allow for confirmed Forward functionality	§5.5
Chapter 6	
• Deleted for clarification	§6.2.2.3
• “I” and “J” Classes of Communication added	Table 6-1
Chapter 7	
• General CM air and ground user requirements added	§7.1.1, §7.1.1.1, §7.2.1, §7.2.1.1, §7.2.1.2, §7.2.1.3
• Requirements changed to notes	§7.2.1, §7.2.2, §7.3.1, §7.3.2, §7.3.4, §7.3.5, §7.3.6
• Requirements combined/clarified	§7.2.2.1, §7.2.2.3
• Text clarified	§7.2.2.3, §7.2.3.2, §7.3.2.1, §7.3.6.2, §7.3.6.5
• Removal of logon successful indication; requirements combined for clarification	§7.3.2.4, §7.3.2.6, §7.3.2.7
• “Prohibited” requirement removed	§7.3.4.1

TABLE OF CONTENTS

1. APPLICATION OVERVIEW

1.1 Introduction

1.1.1 Purpose

1.1.1.1 The purpose of this document is to define draft Standards and Recommended Practices (SARPs) for the Context Management (CM) application through the use of the Aeronautical Telecommunication Network (ATN). This application provides addressing capability for data link applications. The ATN provides the media and protocols to conduct data link communications for the CM application.

1.1.1.2 Compliance with these standards is a means of assuring that the Air Traffic Control (ATC) system will perform its intended functions using data link, and that the CM application is implemented in a globally uniform and interpretable~~interoperable~~ manner.

1.1.2 Background

1.1.2.1 The CM application provides the capability to establish a logon between peer ATC ground systems and ATC ground and aircraft systems via the ATN. Once an appropriate connection is established, CM provides data link application information, the capability to log-on to another ground system, and the capability to update log-on information.

1.1.2.2 In the performance of this role, the CM application interacts with the following:

a) The ATN,
b) Other available data link applications (e.g., Controller Pilot Data Link Communication (CPDLC), Automatic Dependent Surveillance (ADS), and Flight Information Services (FIS)),
c) The ground ATC systems, and
d) The aircraft avionics system.

1.1.3 Structure of Document

1.1.3.1 Chapter 1: APPLICATION OVERVIEW contains the document's purpose and structure, a summary of the ADSP operational requirements (ORs) that relate to CM and maps these ORs to the functions of CM.

1.1.3.2 Chapter 2: GENERAL REQUIREMENTS contains performance, time accuracy, security, backwards compatibility, and error processing requirements.

1.1.3.3 Chapter 3: ABSTRACT SERVICE DEFINITION contains the description of the abstract service provided by the CM Application Service Element (CM-ASE).

1.1.3.4 Chapter 4: FORMAL DEFINITION OF MESSAGES contains the formal definition of messages exchanged by CM-ASEs using Abstract Syntax Notation Number One (ASN.1).

1.1.3.5 Chapter 5: PROTOCOL DEFINITION describes the exchanges of messages allowed by the CM protocol, as well as time constraints and CM-ASE protocol descriptions. This chapter also describes the CM protocol in terms of state tables.

1.1.3.6 Chapter 6: COMMUNICATION REQUIREMENTS contains the requirements that the CM application imposes on the underlying communication system.

1.1.3.7 Chapter 7 : CM USER REQUIREMENTS contains requirements imposed on the user of the CM-ASE service.

1.1.4 Explanation of Terms

1.1.4.1 Acronyms

1.1.4.1.1 The following abbreviations are used in this document:

ADS	Automatic Dependent Surveillance
ADSP	Automatic Dependent Surveillance Panel
AE	Application Entity
APDU	Application Protocol Data Unit
ARS	Administrative Region Selector
ASE	Application Service Element
ASN.1	Abstract Syntax Notation Number One
ATC	Air Traffic Control
ATM	Air Traffic Management
ATN	Aeronautical Telecommunication Network
ATS	Air Traffic Service
ATSC	Air Traffic Services Communication
CF	Control Function
CM	Context Management
CNS	Communication Navigation Surveillance
CNS/ATM	Communications Navigation Surveillance / Air Traffic Management
CPC	Controller Pilot Communications
CPDLC	Controller Pilot Data Link Communications
DS	Dialogue Service
FD	Functional Description
FDPS	Flight Data Processing System
FIS	Flight Information Services
IA5	International Alphabet Number 5
ICAO	International Civil Aviation Organization
ID	Identification
IDP	Initial Domain Part
IEC	International Electrotechnical Committee
IS	International Standard
ISO	International Organization for Standardization
ITU	International Telecommunications Union
LOC	Location Identifier
OR	Operational Requirement
PDU	Protocol Data Unit
PER	Packed Encoding Rules
QOS	Quality of Service
~~**RCP**~~	~~Required Communication Performance~~
RDP	Router Domain Part
RER	Residual Error Rate
SARPs	Standards and Recommended Practices
SEL	(Transport) Selector
SYS	System Identifier
TSAP	Transport Service Access Point
UTC	Coordinated Universal Time

1.1.4.2 General Definitions

1.1.4.2.1 For the purpose of this document the following definitions apply:

a) ***Active User:*** a user which is currently involved in a CM dialogue.
b) ***Actual TSAP address:*** the actual TSAP address is composed of the IDP and the long TSAP address.
c) ***AE Qualifier:*** that part of the AE title that uniquely identifies the particular application entity.
d) ***AE Title:*** a unique name for an application entity.

e) ***APDU:*** basic unit of information exchanged between two CM ASEs.
f) ***Application Entity:*** a model of those aspects of an application process that are significant from the viewpoint of accessing OSI capabilities.
g) ***Application Process:*** an element within an open system which performs information processing tasks for a particular application.
h) ***ARS:*** the Administrative Region Selector, which contains either the ICAO 24 bit airframe identifier or the ground system identifier.
i) ***ASE:*** an abstract module of a system providing service to other parts of the system.
j) ***CM AE abstract service interface:*** the abstract interface between the CM-users and the CM-service provider.
k) ***CM-ASE abstract service interface:*** the abstract interface through which the CM-ASE service are accessed.

Note.— In version 1 of the CM application, this interface coincides with the CM-AE abstract service interface.

k) ***CM-air-ASE:*** an abstract part of the aircraft system which performs the communication related functions of CM.
l) ***CM-air-user:*** the abstract part of the aircraft system which performs the non communication related functions of CM.
m) ***CM-CF:*** that abstract part of the application entity which performs the mapping between the CM-ASE service primitives and other elements within the CM application.
n) ***CM-ground-ASE:*** an abstract part of the ground system which performs the communication related functions of CM.
o) ***CM-ground-user:*** the abstract part of the ground system which performs the non communication related functions of CM.
p) ***CM service primitive:*** a function of a CM-AE that is not broken down further into sub-functions, and is presented as part of the CM-AE abstract service interface (i.e., request, indication, response, or confirmation).
q) ***CM service provider:*** the CM-service provider is composed of the ground and airborne CM AEs, all underlying data communication protocol entities and the physical media. As a consequence, it encompasses everything between the CM-AE service interfaces of the end-users of the CM application.
r) ***Context Management***: an independent service that meets ATSC addressing requirements. It provides the mechanism for aircraft and ATC ground systems to indicate availability to other ATN users and to convey the addresses to be employed.
s) ***Dialogue Service:*** the service which allows the CM-air-ASE to communicate CM-ground-ASE and vice-versa or allows a CM-ground-ASE to communicate with another CM-ground-ASE.
t) ***Long TSAP address:*** the long TSAP address is composed of the RDP and the short TSAP address.
u) ***Message:*** basic unit of information exchanged between ~~the~~ peer ~~airborne~~ CM Application~~s and the ground CM Application~~.
v) ***Residual Error Rate:*** is defined as the ratio of messages mis-delivered, non-delivered, or delivered with an error undetected by the system, over the total number of messages delivered to the system.
w) ***Short TSAP address:*** the short TSAP address is composed of the ARS, the LOC, the SYS and the SEL.

1.1.4.3 Conventions For Expressing Requirements

1.1.4.3.1 The following conventions apply for expressing requirements in this document:

a) ***shall*** - used to state a mandatory requirement.
b) ***should*** - used to state a recommended practice.

1.1.5 References

1.1.5.1 The following references are used in this document:

[1] Draft ICAO Manual of Air Traffic Services (ATS) Data Link Applications, 1 March 1996.~~Automatic Dependent Surveillance (ADS) and Air Traffic Services (ATS) Data Link Applications Guidance Material, June 1995.~~

[2] ISO/IEC IS 8825-2, ITU-T Recommendation X.691, Information Technology - ASN.1 Encoding Rules - Packed Encoding Rules (PER).

[3] ISO/IEC 8824-1, ITU-T Recommendation X.682, Information Technology - Abstract Syntax Notation One (ASN.1).

[4] Draft SARPs for ATN Upper Layers for CNS/ATM-1 Package, Version 2.0, ~~(as proposed in~~9 February 1996~~ 15 December 1995)~~.

[5] Reference Document for Traffic Types.

1.2 Application Functionality

1.2.1 This section lists the operational requirements (OR) as defined by the ADSP that CM addresses:

a) Establishment of data link between aircraft avionics and the FDPS, and
b) Registration of the communications and surveillance capabilities of an aircraft.

1.2.2 Mapping of Operational Requirements to Functional Descriptions

1.2.2.1 This section presents how each of the two CM related ORs listed above can be mapped to the functions described in Section 1.3. Table 1-1 shows which functional descriptions assist in achieving each OR and are labeled as follows (section reference in parentheses):

a) FD1: Logon (Section 1.3.1),
b) FD2: Update (Section 1.3.2),
c) FD3: Contact (Section 1.3.3),
d) FD4: Forward (Section 1.3.4), and
e) FD5: Registration (Section 1.3.5).

Operational Requirement	FD1	FD2	FD3	FD4	FD5
Establishment of data link between aircraft avionics and the FDPS	√	√	√	√	
Registration of the communications and surveillance capabilities of an aircraft					√

Table 1-1: OR to Functional Description Mapping

1.3 Functional Descriptions

1.3.1 FD1: Logon

1.3.1.1 Functional Description

1.3.1.2 The Logon function can only be air initiated. The aircraft system can use the logon function to provide an ~~aircraft~~ application name and version number for each air-only initiated application, and an ~~aircraft~~ application name, address, and version number for each application that the aircraft wishes to use that can be ground initiated, along with flight plan information as required by the ground system. In response, the ground provides an~~ ground~~ application name for each ground-only initiated requested application and an~~ ground~~ application name, address and version number for each requested application that can be air initiated and that the ground can support.

1.3.1.3 Up to a maximum of 256 applications can be supported.

1.3.1.4 Each time a logon is accomplished between a given aircraft and a ground system, the latest exchanged information replaces any previous information for each indicated application.

1.3.1.5 Message Descriptions

1.3.1.5.1 The CM Logon Request message provides required flight plan information, the aircraft's CM application name and address, and information for each ~~aircraft~~ application for which data link services are desired. For each application that can be ground initiated the aircraft must provide the application name, version number and address. For each application which is only air initiated the aircraft must provide the ~~aircraft~~ application name and version number.

1.3.1.5.2 The CM Logon Response message provides ~~an indication of success or failure of the logon, as well as~~ information for the logon-indicated air-initiated applications. For each desired air-initiated application the ground provides the application name, version number, and address.

1.3.2 FD2: Update

1.3.2.1 Functional Description

1.3.2.1.1 This function provides a method for the ground system to updated application information. This function assumes that the logon function has been accomplished.

1.3.2.2 Message Descriptions

1.3.2.2.1 The CM Update message can provide updated ground information for up to 256 applications. For each updated application the ground provides the application's name, version number and address.

1.3.3 FD3: Contact

1.3.3.1 Functional Description

1.3.3.1.1 This function provides a method for the ground system to request the aircraft system to initiate the logon function with a designated ground system. It is expected that the contact function will only be used when ground connectivity is not available between respective ground system applications. This function assumes that the logon function has been accomplished with the ground system initiating the contact function. The ground initiates this function with a contact request specifying which ground system the aircraft should logon with. The aircraft initiates a logon as specified above and indicates the success or lack thereof of the logon.

1.3.3.2 Message Descriptions

1.3.3.2.1 The CM Contact Request message provides the ground system CM application address that the initiating ground system is requesting the aircraft to logon with.

1.3.3.2.2 The CM Contact Response message provides the information indicating whether or not the requested contact was successful.

1.3.4 FD4: Forwarding

1.3.4.1 Functional Description

1.3.4.1.1 This function provides a method for a ground system to forward aircraft information received from the CM Logon function to another ground system. This function is initiated by a ground system having completed a successful logon, which can then forward the aircraft CM Logon information to other ground systems. It is a one-way forwarding of information (no response is permitted from the receiving CM-ground-user). The ground system receiving this CM information can then initiate a CM Update function to provide information to the aircraft for any air-initiated applications.

1.3.4.2 Message Descriptions

1.3.4.2.1 The CM Forward Request message contains the ~~flight plan information, the aircraft's CM application name, address, and application~~ information as provided in the initial logon.

1.3.5 FD5: Registration

1.3.5.1 Functional Description

1.3.5.1.1 This function provides a method for the air and ground CM applications to make available the application name, address, and version number for each application exchanged in the logon, update or forward functions to other applications or communications systems in the aircraft or on the ground.

1.3.5.2 Message Descriptions

1.3.5.2.1 There are no ~~aircraft system to/from ground system~~ message exchanges for this function.

2. GENERAL REQUIREMENTS

2.1 Performance Requirements

Note.— Systems developed to support Context Management functionality will be capable of meeting the ~~Required~~ c~~C~~ommunication requirements~~Performance (RCP)~~ appropriate for the phase of operation as specified in [5].

2.2 Time Accuracy Requirements

2.2.1 Absolute times which are sent as parameters over the data link shall be as accurate as the required resolution of the time parameter.

2.2.2 Absolute times shall be UTC.

2.2.3 Dates shall be expressed as UTC date (i.e., the day increments at 2359:59z).

2.2.4 Relative times shall be accurate to ± 0.1 second.

2.2.5 Where timestamps are used they shall consist of year, month, day and hour, minute, second.

2.3 Security Requirements

Note.— There are no internationally approved operational requirements relating to data link application security.

2.4 Backwards Compatibility Requirements

Note.— This document describes the version 1 of the CNS/ATM-1 Package CM application. Best efforts will be made to ensure that subsequent versions of this protocol are backwards compatible.

2.4.1 For CNS/ATM-1 the CM-air-ASE and CM-ground-ASE version numbers shall both be set to one.

2.5 Error Processing Requirements

2.5.1 s
the event of information input by the user being incompatible with that able to be processed by the system, the user shall be notified.

2.5.2 In the event of a user invoking a CM service primitive when the CM-ASE is not in a state specified in Chapter 5, the following shall occur:

a) the invocation is rejected, and
b) the user is notified.

3. THE ABSTRACT SERVICE

3.1 Introduction

Note 1.— This chapter defines the abstract service interface for the CM service. The CM-ASE abstract service is described in this chapter from the viewpoint of the CM-air-user, the CM-ground-user and the CM -service-provider.

Note 2.— This chapter defines the static behaviour (i.e., the format) of the FIS abstract service. Its dynamic behaviour (i.e., how it is used) is described in Chapter 7.

3.2 The CM Functional Model

Note 1.— Figure 3-1 shows the functional model of the CM Application. The functional modules identified in this model are the following :

a) the CM-user,
b) the CM Application Entity (CM-AE) service interface,
c) the CM-AE,
d) the CM Control Function (CM-CF),
e) the CM Application Service Element (CM-ASE) service interface,
f) the CM-ASE, and
g) the Dialogue Service (DS) interface.

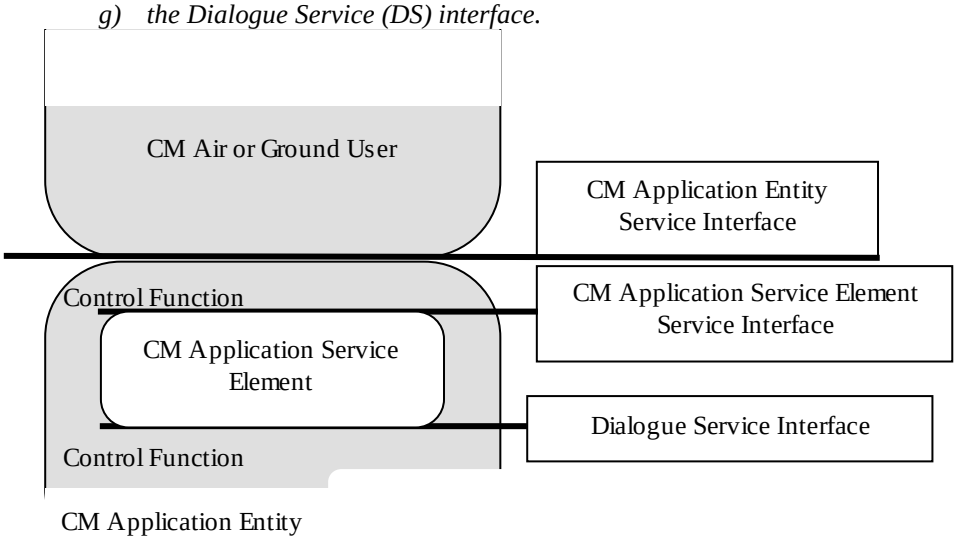

Figure 3-1: Functional Model of the CM Application

Note 2.— The CM-user represents the operational part of the CM system. This user does not perform the communication functions but relies on a communication service provided to it via the CM-AE through the CM-AE service interface. The individual actions at this interface are called CM-AE service primitives. Similarly, individual actions at other interfaces in the communication system are called service primitives at these interfaces.

Note 3.— The CM-AE consists of several elements including the CM-ASE and the CM-CF. The DS interface is made available by the CM-CF to the CM-ASE for communication with the peer CM-ASE.

Note 4.— The CM-ASE is the element in the communication system which executes the CM specific protocol. In other words, it takes care of the CM specific service primitive sequencing actions, message creation, timer management, error and exception handling.

Note 5.— The CM-ASE interfaces only with the CM-CF. This CM-CF is responsible for mapping service primitives received from one element (such as the CM-ASE and the CM-user) to other elements which interface with it. The part of the CM-CF which is relevant from the point of view of these SARPs, i.e. the part between the CM-user and the CM-ASE, will map CM-AE service primitives to CM-ASE service primitives transparently in the CNS/ATM-1 Package.

Note 6.— The DS interface is the interface between the CM-ASE and the part of CM-CF underneath the CM-ASE, and provides the dialogue service [4].

3.3 The CM-ASE Abstract Service

3.3.1 An implementation of either the CM ground based service or the CM air based service shall exhibit external behaviour consistent with having implemented a CM-ground-ASE, or CM-air-ASE respectively, with the following abstract service interface primitives, making them available to the CM-ground-user or CM-air-user respectively:

Note.— There is no requirement to implement the service in a CM product; however, it is necessary to implement the ground based and air basedsstem in sucha way that it will be impossible to detect (from the peer system) whether or not an interface has been built.

3.3.2 The CM-ASE abstract service shall consist of the following:

a) *CM-logon service* as defined in section 3.5,
b) *CM-update service* as defined in section 3.6,
c) *CM-contact service* as defined in section 3.7,
d) *CM-end service* as defined in section 3.8,
e) *CM-forward service* as defined in section 3.9,
f) *CM-user-abort service* as defined in section 3.10, and
g) *CM-provider-abort service* as defined in section 3.11.

3.4 Conventions

Note 1.— For a given primitive, the presence of each parameter is described by one of the following values in the parameter tables in Chapter 3.

a)	***blank***	*not present;*
b)	***C***	*conditional upon some predicate explained in the text;*
c)	***C(=)***	*conditional upon the value of the parameter to the left being present, and equal to that value;*
d)	***M***	*mandatory;*
e)	***M(=)***	*mandatory, and equal to the value of the parameter to the left;*
f)	***U***	*user option.*

Note 2.— The following abbreviations are used in this document:

a) ***Req*** *- request; data is input by CM-user initiating the service to its respective ASE,*
b) ***Ind*** *- indication; data is indicated by the receiving ASE to its respective CM-user,*
c) ***Rsp*** *- response; data is input by receiving CM-user to its respective ASE, and*
d) ***Cnf*** *- confirmation; data is confirmed by the initiating ASE to its respective CM-user.*

Note 3.— An unconfirmed service allows a message to be transmitted in one direction, without providing a corresponding response.

Note 4.— A confirmed service provides end-to-end confirmation that a message sent by one user was received by its peer user.

3.5 CM-logon Service

Note.— The CM-logon service allows the CM-air-user to initiate data link service. The CM-air-user provides information on each data link application for which it desires a data link service. The CM-ground-user responds indicating whether or not the CM-logon was successful, and if successful, includes information on each data link application it can support. It is a confirmed service.

3.5.1 The CM-logon service primitives shall contain the parameters as presented in Tables 3-1 and 3-2.

Parameter Name	Req	Ind	Rsp	Cnf
ICAO Facility Designator	M			
Aircraft Identifier	M	M(=)		
CM ASE Version Number		C		
Logon Request	M	M(=)		
Logon Response			M	M(=)
Class of Communication Service	U			
Maintain Dialogue			U	C(=)

Table 3-1: CM-logon Service Parameters
Air-ASE version Number ≤ Ground-ASE Version Number

Parameter Name	Req	Cnf
ICAO Facility Designator	M	
Aircraft Identifier	M	
CM ASE Version Number		M
Logon Request	M	
Class of Communication Service	U	

Table 3-2: CM-logon Service Parameters
Air-ASE version Number > Ground-ASE Version Number

3.5.2 ICAO Facility Designator

Note.— This parameter contains the addressed ground system's ICAO facility designator.

3.5.2.1 The *ICAO Facility Designator* parameter value shall conform to the abstract syntax eight-character ICAO facility designator.

3.5.3 Aircraft Identifier

Note.— This parameter contains 24-bit aircraft identifier of the aircraft initiating the CM-logon service.

3.5.3.1 The *Aircraft Identifier* parameter value shall conform to the abstract syntax 24-bit aircraft-id.

3.5.4 CM ASE Version Number

Note.— This parameter contains the version number of the CM-air-ASE.

3.5.4.1 Only if the CM-air-ASE version number is less than the CM-ground-ASE version number shall the CM-air-ASE version number be indicated to the CM-ground-user.

3.5.4.2 Only if the CM-air-ASE version number is greater than the CM-ground-ASE version number shall the CM-ground-ASE version number be confirmed to the CM-air-user.

Note 1.— If the CM-air-ASE version number is the same as the CM-ground-ASE version number, the Version Number parameter is not present in the~~an~~ indication ~~is not~~ given to the CM-ground-user, nor in the~~a~~ confirmation to the CM-air-user.

Note 2.— For CNS/ATM-1 the CM-air-ASE and CM-ground-ASE version numbers are both set to 1.

3.5.5 Logon Request

Note.— The logon request parameter contains the following data:

a) information for each data link application available on the aircraft, for which the aircraft requires data link service, and

b) aircraft flight plan information (e.g. flight id, aircraft destination and departure airport and time) as required by the addressed ground system.

3.5.5.1 The *Logon Request* parameter value shall conform to the ASN.1 abstract syntax CMLogonRequest.

3.5.5.2 Only if the CM-air-ASE version number is less than or equal to the CM-ground-ASE version number (i.e. the CM-ground-ASE can decode and present the information to the CM-ground user) shall the *Logon Request* parameter be indicated to the CM-ground-user.

3.5.6 Logon Response

Note.— This parameter contains information for each requested data link application for which the ground is able to provide data link service. ~~the following data:~~

~~an indication whether or not the logon was successful, and, if successful,~~
~~information for each requested data link application for which the ground is able to provide data link service~~.

3.5.6.1 The *Logon Response* parameter value shall conform to the ASN.1 abstract syntax CMLogonResponse.

3.5.6.2 If the *Logon Request* parameter was indicated to the CM-ground-user, the CM-ground-user shall provide the *Logon Response* parameter.

3.5.7 Class of Communication Service

Note.— This parameter contains the value of the required class of communication service if specified by the CM-air-user.

3.5.7.1 When this parameter is specified by the CM-air-user, the *Class of Communication Service* parameter value shall have one of the following abstract values: “A”, “B”, “C”, “D”, “E”, “F”, “G”, ~~or~~ “H”, “I”, or “J”.

Note 1.— Class of Communication Service parameter values are detailed in [5].

Note 2.— If not specified by the CM-air-user, this indicates that there is no routing preference.

3.5.8 Maintain Dialogue

Note 1.— This parameter is used to indicate whether or not the requested CM dialogue is to remain open after a Logon Response.

Note 2.— Whenever a CM dialogue is kept open by the CM-ground-user, it must later be explicitly closed by the CM-ground-user.

Note 3.— This parameter is only provided by the CM-ground-user when the CM-ground-user wishes to keep the CM dialogue open.

3.5.8.1 If provided by the CM-ground-user this parameter shall have the abstract value “maintain~~accepted~~”.

3.6 CM-update Service

Note.— The CM-update service allows~~can be used by~~ the CM-ground-user to transmit updated ground information for its applications to update previously coordinated CM-logon information. It is an unconfirmed service.

3.6.1 The CM-update service primitives shall contain the parameters as presented Table 3-3.

Parameter Name	Req	Ind
Aircraft Identifier	C	
ICAO Facility Designator	C	C(=)
Update Information	M	M(=)
Class of Communication Service	U	

Table 3-3: CM-update Service Parameters

3.6.2 Aircraft Identifier

Note.— This parameter contains the addressed aircraft's 24-bit aircraft identifier.

3.6.2.1 The *Aircraft Identifier* parameter value shall conform to the abstract syntax 24-bit aircraft-id.

3.6.2.2 If a CM dialogue does not exist when a CM-ground user invokes the CM-update service request, the CM-ground-user shall provide the *Aircraft Identifier* parameter value.

Note.— The CM-update service does not use this parameter when a CM dialogue exists.

3.6.3 ICAO Facility Designator

Note.— This parameter contains the ground system's ICAO facility designator.

3.6.3.1 The *ICAO Facility Designator* parameter value shall conform to the abstract syntax eight-character ICAO facility designator.

3.6.3.2 If a CM dialogue does not exist when a CM-ground user invokes the CM-update service request, the CM-ground-user shall provide the *ICAO Facility Designator* parameter value.

Note.— The CM-update service does not use this parameter when a CM dialogue exists.

3.6.4 Update Information

Note.— This parameter contains information on each updated data link application.

3.6.4.1 The *Update Information* parameter value shall conform to the ASN.1 abstract syntax CMUpdate.

3.6.5 Class of Communication Service

Note.— This parameter contains the value of the required class of communication service if specified by the CM-ground-user.

3.6.5.1 Where specified by the CM-ground-user, the *Class of Communication Service* parameter shall have one of the following abstract values: "A", "B", "C", "D", "E", "F", "G", ~~or~~ "H", "I", or "J".

Note 1.— Class of Communication Service parameter values are detailed in [5].

Note 2.— If not specified by the CM-ground-user, this indicates that there is no routing preference.

3.7 CM-contact Service

Note.— The CM-contact service allows~~can be used by~~ the CM-ASE-ground-user, after successful completion of a CM logon, to request that an aircraft logon with another ground system. It is a confirmed service.

3.7.1 The CM-contact service primitives shall contain the parameters as presented Table 3-4.

Parameter Name	Req	Ind	Rsp	Cnf
Aircraft Identifier	C			
ICAO Facility Designator	C	C(=)		
Contact Request	M	M(=)		
Contact Response			M	M(=)
Class of Communication Service	U			

Table 3-4: CM-contact Service Parameters

3.7.2 Aircraft Identifier

Note.— This parameter contains the addressed aircraft's 24-bit aircraft identifier.

3.7.2.1 The *Aircraft Identifier* parameter value shall conform to the abstract syntax 24-bit aircraft-id.

3.7.2.2 If a CM dialogue does not exist when a CM-ground user invokes the CM-contact service request, the CM-ground-user shall provide the *Aircraft Identifier* parameter value.

Note.— The CM-contact service does not use this parameter when a CM dialogue exists.

3.7.3 ICAO Facility Designator

Note.— This parameter contains the ground system's ICAO facility designator.

3.7.3.1 The *ICAO Facility Designator* parameter value shall conform to the abstract syntax eight-character ICAO facility designator.

3.7.3.2 If a CM dialogue does not exist when a CM-ground user invokes the CM-contact service request, the CM-ground-user shall provide the *ICAO Facility Designator* parameter value.

Note.— The CM-contact service does not use this parameter when a CM dialogue exists.

3.7.4 Contact Request

Note.— This parameter contains the ICAO facility designator for the ground system that the CM-ground-user requests the aircraft to contact.

3.7.4.1 The *Contact Request* parameter value shall conform to the ASN.1 abstract syntax CMContactRequest.

3.7.5 Contact Response

Note.— This parameter indicates success, or lack thereof, of the requested contact.

3.7.5.1 The *Contact Response* parameter value shall conform to the ASN.1 abstract syntax CMContactResponse.

3.7.6 Class of Communication Service

Note.— This parameter contains the value of the required class of communication service if specified by the CM-ground-user.

3.7.6.1 When this parameter is specified by the CM-ground-user, the *Class of Communication Service* parameter value shall have one of the following abstract values: "A", "B", "C", "D", "E", "F", "G", ~~or~~ "H", "I", or "J".

Note 1.— Class of Communication Service parameter values are detailed in [5].

Note 2.— If not specified by the CM-ground-user, this indicates that there is no routing preference.

3.8 CM-end Service

Note 1.— This service provides the capability for the CM-ground-user to terminate a CM dialogue. This service is only needed when the CM-ground-user maintains~~accepts a request for~~ a CM dialogue during the logon process. It is an unconfirmed service.

Note 2.— Only the CM-ground-user shall be capable of initiating the CM-end service.

3.8.1 The CM-end service primitives shall contain the parameters as presented Table 3-5.

Parameter Name	Req	Ind
none		

Table 3-5: CM-end Service Parameters

3.9 CM-forward Service

Note.— The CM-forward service allows~~can be used~~ by a CM-ground-user to forward data received in a CM-logon request to another CM-ground system. It is a confirmed service.

3.9.1 The CM-forward service primitives shall contain the parameters as presented in Tables 3-6 ~~and 3-7~~.

Parameter Name	Req	Ind	Cnf
Called ICAO Facility Designator	M	M	
Calling ICAO Facility Designator	M	M(=)	
CM ASE Version Number		C	C
Forward Request	M	M(=)	
Class of Communication Service	U		
Result		M	M

Table 3-6: CM-forward Service Parameters
~~Sending Ground-ASE version Number ≤ Receiving Ground-ASE Version Number~~

~~Parameter Name~~	~~Req~~	~~Cnf~~
~~Called ICAO Facility Designator~~	~~M~~	
~~Calling ICAO Facility Designator~~	~~M~~	
~~CM ASE Version Number~~		~~M~~
~~Forward Request~~	~~M~~	
~~Class of Communication Service~~	~~U~~	

~~Table 3-7: CM-forward Service Parameters~~
~~Sending Ground-ASE version Number > Receiving Ground-ASE Version Number~~

3.9.2 Called ICAO Facility Designator

Note.— This parameter contains the receiving~~called~~ ground system's ICAO facility designator.

3.9.2.1 The *Called ICAO Facility Designator* parameter value shall conform to the abstract syntax eight-character ICAO facility designator.

3.9.3 Calling ICAO Facility Designator

Note.— This parameter contains the sending~~initiating~~ ground system's ICAO facility designator.

3.9.4 The *Calling ICAO Facility Designator* parameter value shall conform to the abstract syntax eight-character ICAO facility designator.

3.9.5 CM ASE Version Number

Note.— This parameter contains the version number of the sending CM-ground-ASE.

3.9.5.1 Only if the sending CM-ground-ASE version number is less than the receiving CM-ground-ASE version number shall the sending CM-ground-ASE version number be indicated to the receiving CM-ground-user.

3.9.5.2 Only if the sending CM-ground-ASE version number is greater than the receiving CM-ground-ASE version number shall the receiving CM-ground-ASE version number be confirmed to the sending CM-ground-user.

Note 1.— If the sending CM-ground-ASE version number is the same as the receiving CM-ground-ASE version number, the Version Number paramter is not present in the~~an~~ indication ~~is not~~ given to the receiving CM-ground-user, nor in the ~~a~~ confirmation to the sending CM-ground-user.

Note 2.— For CNS/ATM-1 the sending CM-ground-ASE and receiving CM-ground-ASE version numbers are both set to 1.

3.9.6 Forward Request

Note.— This parameter contains information as provided in the CM Logon Request.

3.9.6.1 The *Forward Request* parameter value shall conform to the ASN.1 abstract syntax CMForwardRequest.

3.9.7 Class of Communication Service

Note.— This parameter contains the value of the required class of communication service if specified by the initiating CM-ground-user.

3.9.7.1 When this parameter is specified by the CM-ground-user, the *Class of Communication Service* parameter value shall have one of the following abstract values: "A", "B", "C", "D", "E", "F", "G", ~~or~~ "H", "I", or "J".

Note 1.— Class of Communication Service parameter values are detailed in [5].

Note 2.— If not specified by the CM-ground-user, this indicates that there is no routing preference

3.9.8 Result

Note.— This parameter indicates whether or not the information was forwarded as requested.

3.9.8.1 The *Result* parameter shall have the abstract value "success" or "failure".

Note.— When the sending CM-ground-ASE version number is less than or equal to the receiving CM-ground-ASE version number the Result parameter takes the abstract value "success". When the sending CM-ground-ASE version number is greater than the receiving CM-ground-ASE version number the Result parameter takes the abstract value "failure".

3.10 CM-user-abort Service

Note.— This service provides the capability for either the CM-air-user or a~~the~~ CM-ground-user to abort communication with its peer. This can be used for operational or technical reasons. It can be invoked at any time by an active user. Messages in transit may be lost during this operation. It is an unconfirmed service.

3.10.1 The CM-user-abort service primitives shall contain the parameters as presented Table 3-7~~8~~.

Parameter Name	Req	Ind
none		

Table 3-7~~8~~: CM-user-abort Service Parameters

3.11 CM-provider-abort Service

Note.— This service provides the capability for the CM-service provider to inform its users that it can no longer provide the CM service. Messages in transit may be lost during this operation.

3.11.1 The CM-provider-abort service primitives shall contain the parameters as presented Table 3-8~~9~~.

Parameter Name	Ind
Reason	M

Table 3-8~~9~~: CM-provider-abort Service Parameters

3.11.2 Reason

Note.— This parameter identifies the reason for the abort.

3.11.2.1 The *Reason* parameter value shall conform to the ASN.1 abstract syntax CMAbortReason.

4. FORMAL DEFINITIONS OF MESSAGES

4.1 CM ASN.1 Abstract Syntax

4.1.1 The abstract syntax of the CM protocol data units shall comply with the description contained in the ASN.1 module CMMessageSetVersion1 (conforming to [3]), as defined in this section.~~below:~~

```
CMMessageSetVersion1 DEFINITIONS ~~AUTOMATIC TAGS~~ ::=

BEGIN

-- ------------------------------------------------------------------------------
-- CM Message Structure
-- ------------------------------------------------------------------------------
--  Aircraft-generated messages

CMAircraftMessage ::=CHOICE
        {
        cmLogonRequest          [0]     CMLogonRequest,
        cmContactResponse       [1]     CMContactResponse,
        cmAbortReason           [2]     CMAbortReason,
        ...
        }

--  Ground-generated messages

CMGroundMessage ::=CHOICE
        {
        cmLogonResponse         [0]     CMLogonResponse,
        cmUpdate                [1]     CMUpdate,
        cmContactRequest        [2]     CMContactRequest,
        cmForwardRequest        [3]     CMForwardRequest,
        cmAbortReason           [4]     CMAbortReason,
        ...
        }

-- ------------------------------------------------------------------------------
-- CM Message Components
-- ------------------------------------------------------------------------------

AircraftFlightIdentification ::= IA5String (SIZE(2..8))

Airport ::= IA5String (SIZE(4))

APAddress ::= CHOICE
        {
        longTsap        [0]     LongTsap,
        shortTsap       [1]     ShortTsap
        }

APName ::= IA5String (SIZE(3))
```

```
APNameVersion ::= SEQUENCE
        {
        apName          APName,
        apVersion       VersionNumber
        }

APNameVersionAddress ::= SEQUENCE
        {
        apName                  APName,
        apVersion               VersionNumber,
        apAddress               APAddress
        }

CMAbortReason ::= ENUMERATED
        {
        timer-expired                           (0),
        undefined-error                         (1),
        invalid-PDU                             (2),
        not-permitted-PDU                       (3),
        dialogue-acceptance-not-permitted       (4),
        dialogue-end-not-accepted               (5),
        communication-service-error             (6),
        communication-service-failure           (7)
        }

CMContactRequest ::= SEQUENCE
        {
        icaoFacilityDesignation         ICAOFacilityDesignation,
        address                         LongTsap
        }

CMContactResponse ::= Response

CMForwardRequest ::= CMLogonRequestSEQUENCE
        {
        aircraftFlightIdentification    AircraftFlightIdentification,
        cMNameLongTSAP                  CMNameLongTSAP,
        groundInitiatedApplications     SEQUENCE SIZE (1..256) OF   APNameVersionAddress
                                                                    OPTIONAL,
        airOnlyInitiatedApplications    SEQUENCE SIZE (1..256) OF   APNameVersion
                                                                    OPTIONAL,
        airportDeparture                Airport                     OPTIONAL,
        airportDestination              Airport                     OPTIONAL,
        timeDepartureetd                Time                        OPTIONAL
        }

CMLogonRequest ::= SEQUENCE
        {
        aircraftFlightIdentification    [0]   AircraftFlightIdentification,
        cMNameLongTSAP                  [1]   CMNameLongTsapSAP,
        groundInitiatedApplications     [2]   SEQUENCE SIZE (1..256) OF   APNameVersionAddress
                                                                          OPTIONAL,
```

```
        airOnlyInitiatedApplications    [3]     SEQUENCE SIZE (1..256) OF   APNameVersion
                                                                            OPTIONAL,
        icaoFacilityDesignation         [4]     ICAOFacilityDesignation     OPTIONAL,
        airportDeparture                [5]     Airport                     OPTIONAL,
        airportDestination              [6]     Airport                     OPTIONAL,
        timeDepartureETD                [7]     Time                        OPTIONAL
        }

CMLogonResponse ::= SEQUENCE
        {
        response                                Response,
        airInitiatedApplications        [0]     SEQUENCE SIZE (1..256) OF   APNameVersionAddress
                                                                            OPTIONAL,
        groundOnlyInitiatedApplications [1]     SEQUENCE SIZE (1..256) OF   APNameVersion
                                                                            OPTIONAL
        }

~~CMNameLongTSAP :: = SEQUENCE~~
        ~~{~~
        ~~CMName        APName,~~
        ~~longTSAP      LongTsap~~
        ~~}~~

CMUpdate ::= CMLogonResponse

ICAOFacilityDesignation ::= IA5String (SIZE(8))

LongTsap ::= SEQUENCE
        {
        rDP             OCTET STRING (SIZE(5)),
        shortTsap       ShortTsap
        }

Response ::= ENUMERATED
        {
        contactsSuccess         (0),
        ~~logonNotSuccessful            (1),~~
        contactNotSuccessful    (12)
        }

ShortTsap ::= SEQUENCE
        {
        aRS             [0]     OCTET STRING (SIZE(3))          OPTIONAL,
        -- the aRS contains the ICAO 24 bit airframe identifier when the ShortTsap belongs to an aircraft
        -- or a ground address when the Short Tsap belongs to a ground system
        locSysTsel      [1]     OCTET STRING  (SIZE(9..11))
        }

Time ::= SEQUENCE
        {
        hours           Timehours,
        minutes         Timeminutes
        }
```

Timehours ::= INTEGER (0..23)
-- units = hour, range (0..23), resolution = 1 hour

Timeminutes ::= INTEGER (0..59)
-- units = minute, range (0..59), resolution = 1 minute

VersionNumber ::= INTEGER (0..255)

END

5. PROTOCOL DEFINITION

5.1 Sequence Rules

Note.— The following figures define the valid sequences of primitives that are possible to be invoked during the operation of the CM application. It shows the relationship in time between the service request and the resulting indication, and if applicable, the subsequent response and resulting confirmation.

5.1.1 With the exception of abort primitives, only the sequence of primitives described below shall be permitted.

Note.— Abort primitives may interrupt and terminate any of the normal message sequences outlined below.

5.1.2 With the exception of abort primitives, the CM-air-ASE and CM-ground-ASE shall process primitives in the order in which they are received.

Note.— This ensures that the CM-ASE will guarantee message sequencing, with the exception of aborts.

5.1.3 CM-logon Service

5.1.3.1 CM-Air-ASE Version Less Than Or Equal To The CM-Ground-ASE Version

Note.— The following sequence of messages, shown in Figure 5-1, occurs when the CM-logon service is initiated with a CM-air-ASE version number less than or equal to the CM-ground-ASE version number.

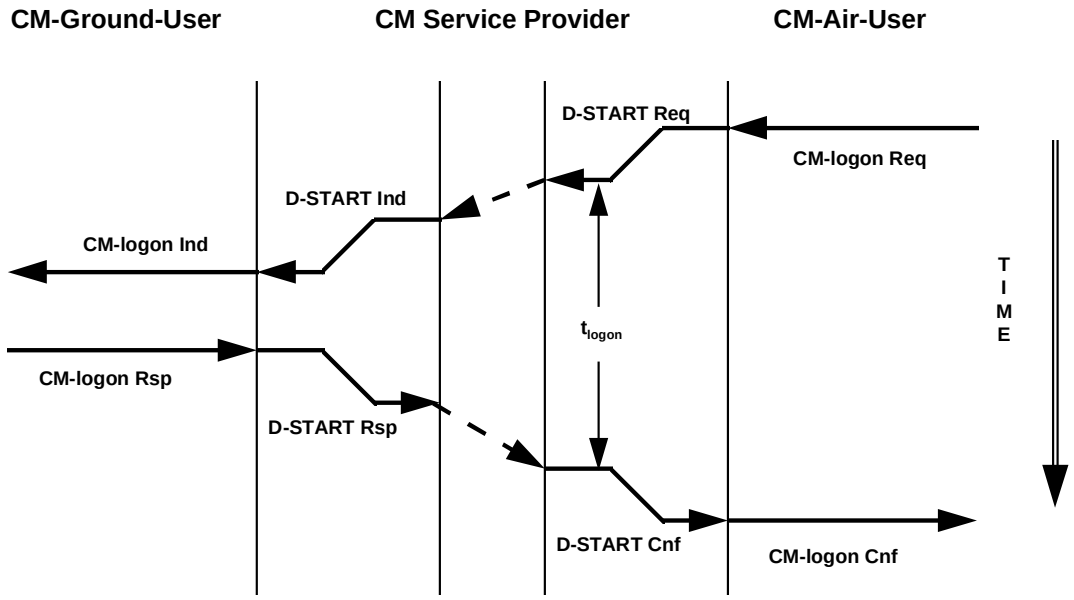

Figure 5-1: Sequence Diagram for CM-logon Service
CM-Air-ASE Version ≤ CM-Ground-ASE Version

5.1.3.2 CM-Air-ASE Version Greater Than The CM-Ground-ASE Version

Note.— The following sequence of messages, shown in Figure 5-2, occurs when the CM-logon service is initiated with a CM-air-ASE version number greater than the CM-ground-ASE version number.

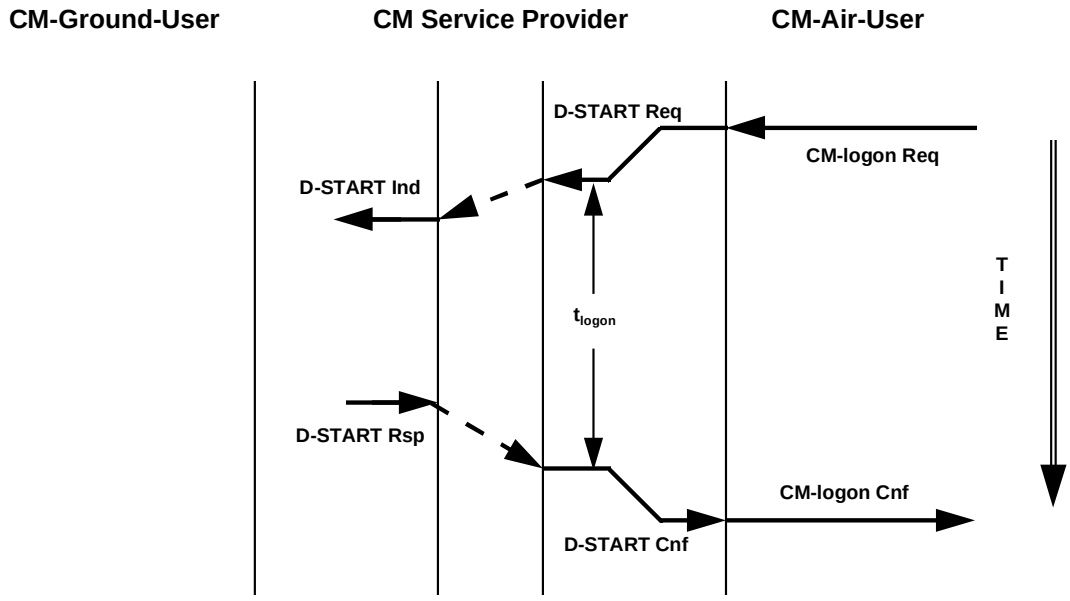

Figure 5-2: Sequence Diagram for CM-logon Service
CM-Air-ASE Version >CM-Ground-ASE Version

5.1.4 CM-update Service

Note 1.— The following sequence of messages, shown in Figure 5-3, occurs when the CM-update service is initiated when there is no CM dialogue in place.

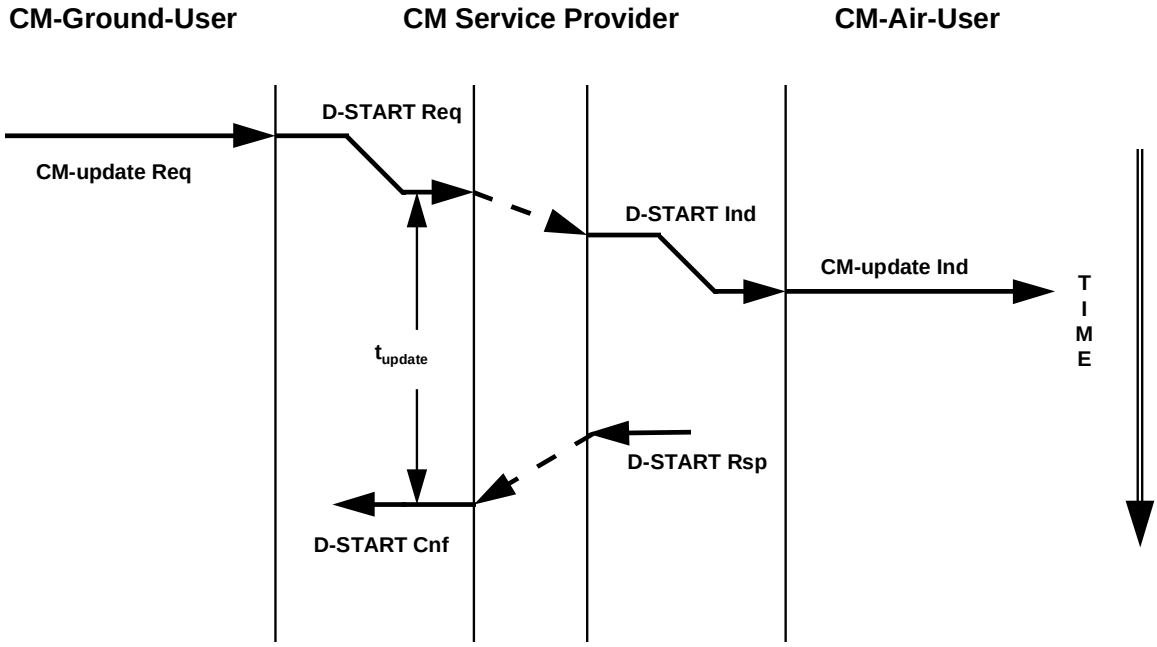

Figure 5-3: Sequence Diagram for CM-update Service
No Existing CM Dialogue

Note 2.— The following sequence of messages, shown in Figure 5-4, occurs when the CM-update service is initiated when there is a CM dialogue in place.

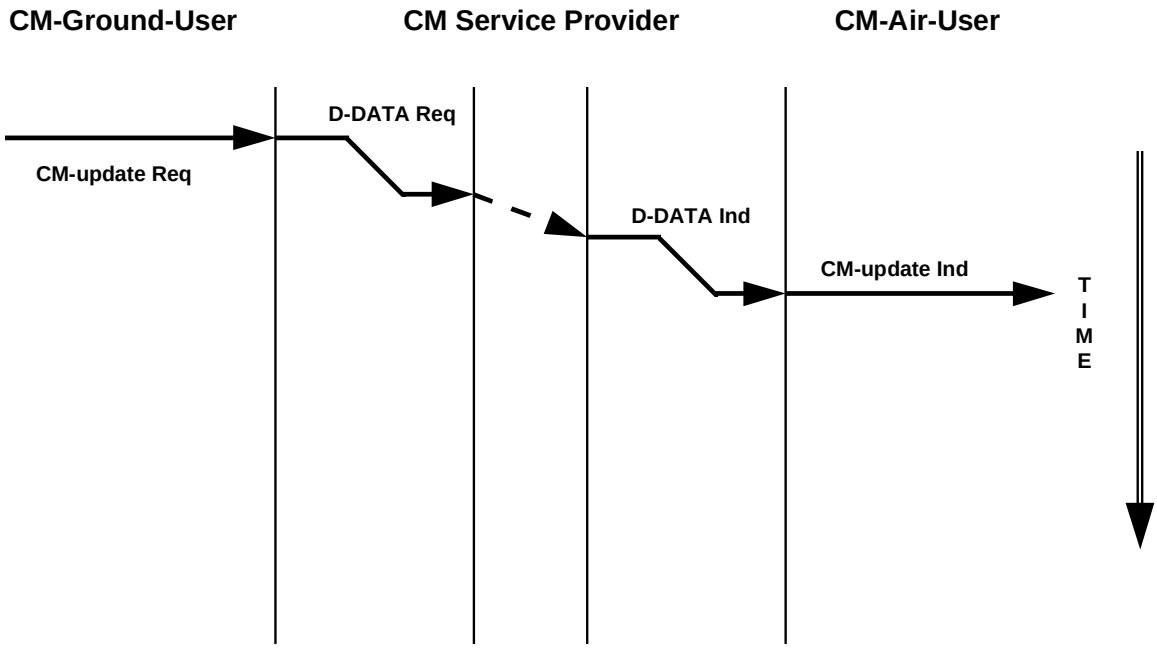

Figure 5-4: Sequence Diagram for CM-update Service
Existing CM Dialogue

5.1.5 CM-contact Service

5.1.5.1 CM-contact With CM-logon as Illustrated in Figure 5-1 with No Existing CM Dialogue

Note.— The following sequence of messages, shown in Figure 5-5, occurs when the CM-contact service is initiated with no existing CM dialogue in place and a CM-logon service occurs as indicated in Figure 5-1.

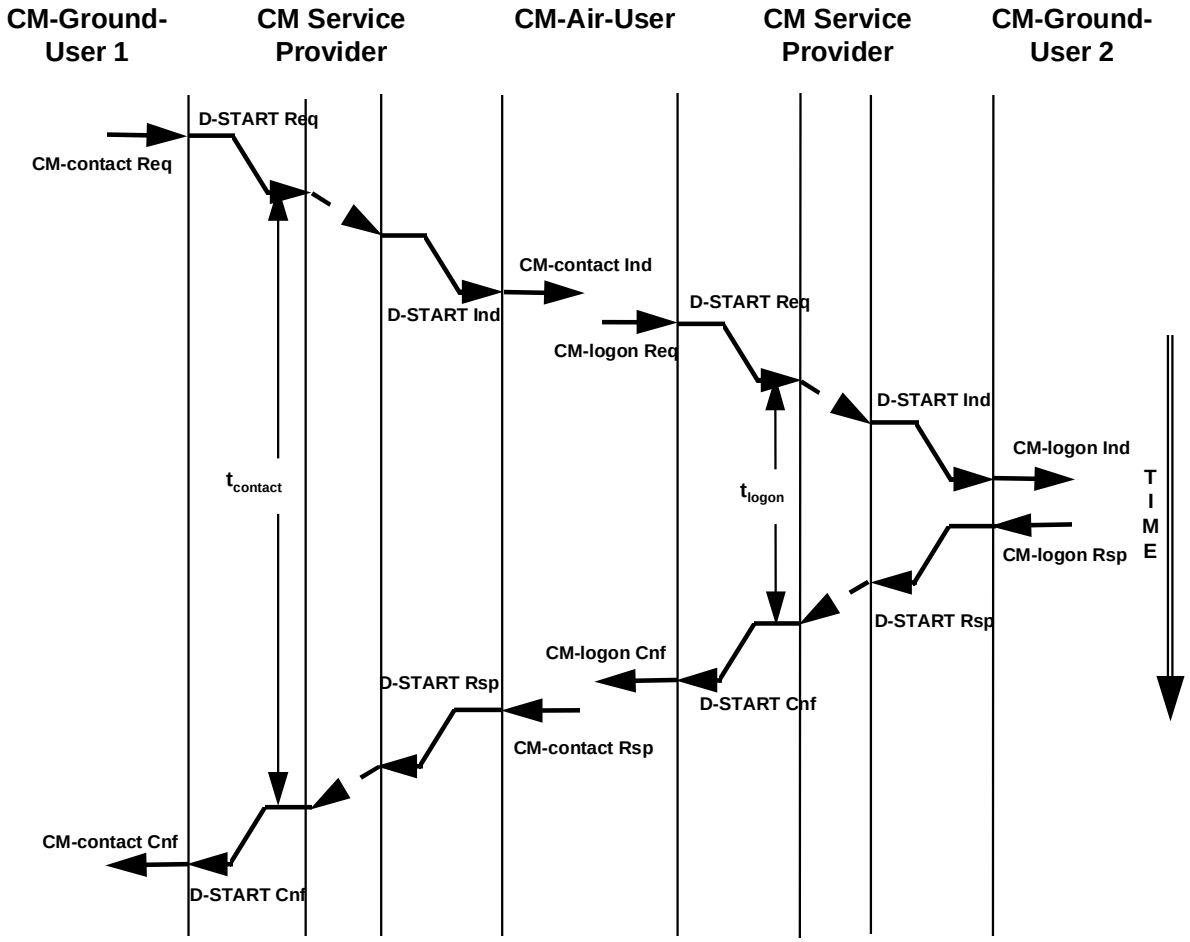

Figure 5-5: Sequence Diagram for CM-contact Service
No Existing CM Dialogue
With CM-logon Service as in Figure 5-1

5.1.5.2 CM-contact With CM-logon as Illustrated in Figure 5-1 with Existing CM Dialogue

Note.— The following sequence of messages, shown in Figure 5-6, occurs when the CM-contact service is initiated with existing CM dialogue and a CM-logon service occurs as indicated in Figure 5-1.

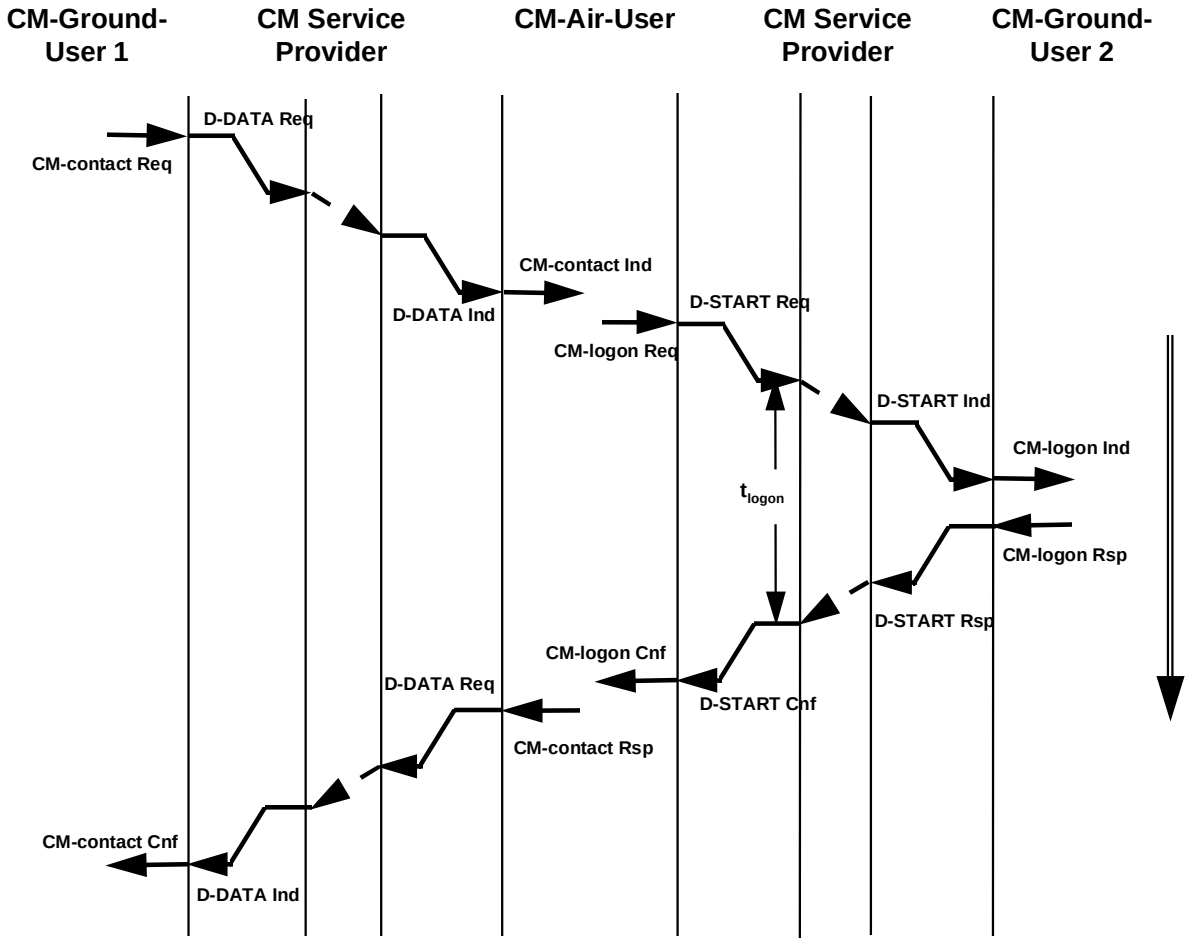

Figure 5-6: Sequence Diagram for CM-contact Service
With Existing CM Dialogue
With CM-logon Service as in Figure 5-1

5.1.5.3 CM-contact With CM-logon as Illustrated in Figure 5-2 No Existing CM Dialogue

Note 1.— The following sequence of messages, shown in Figure 5-7, occurs when the CM-contact service is initiated with no existing CM dialogue and a CM-logon service occurs as indicated in Figure 5-2.

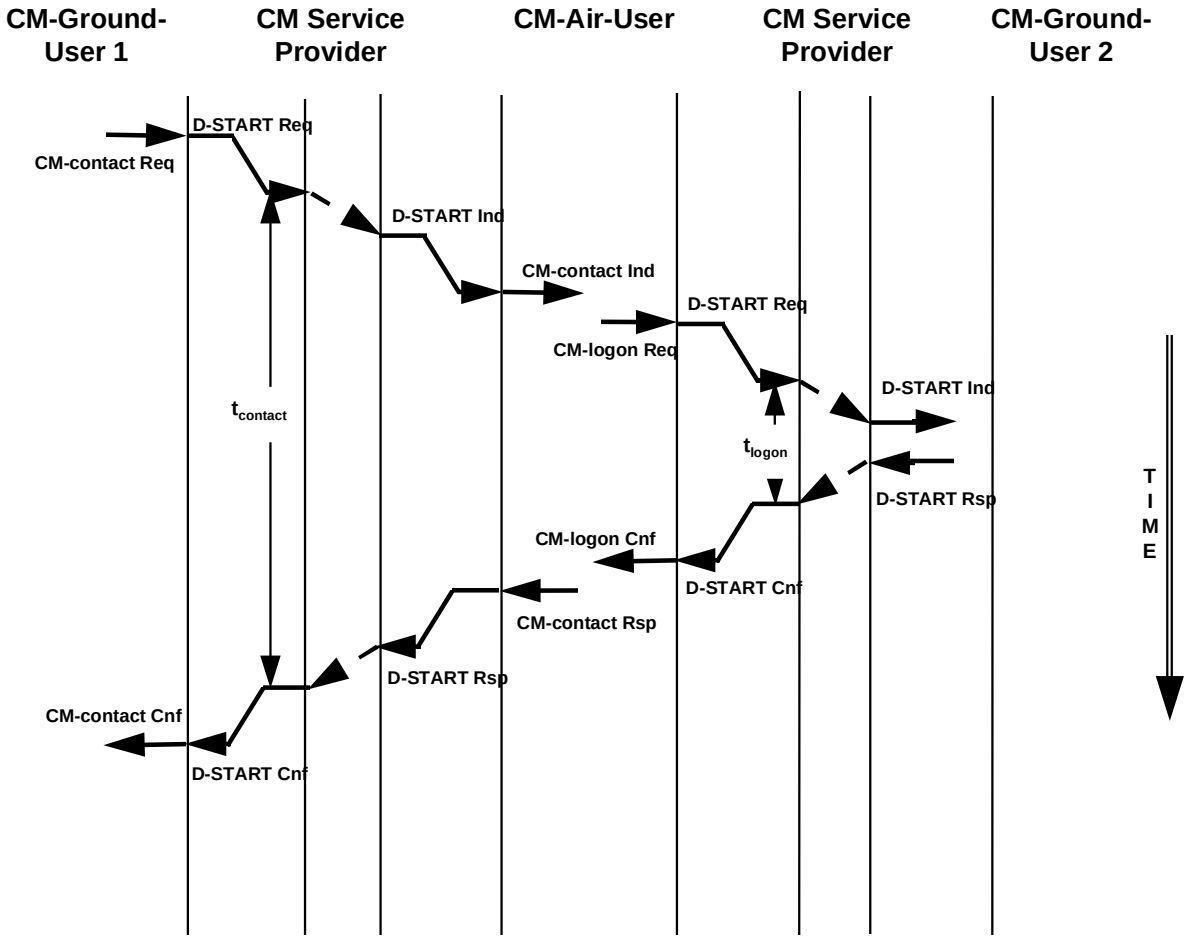

Figure 5-7: Sequence Diagram for CM-contact Service
No Existing CM Dialogue
With CM-logon Service as in Figure 5-2

Note 2.— This figure does not imply that only one CM-logon attempt is made for a given CM-Contact request.

5.1.5.4 CM-contact With CM-logon as Illustrated in Figure 5-2, Existing CM Dialogue

Note 1.— The following sequence of messages, shown in Figure 5-8, occurs when the CM-contact service is initiated with an existing CM dialogue and a CM-logon service occurs as indicated in Figure 5-2.

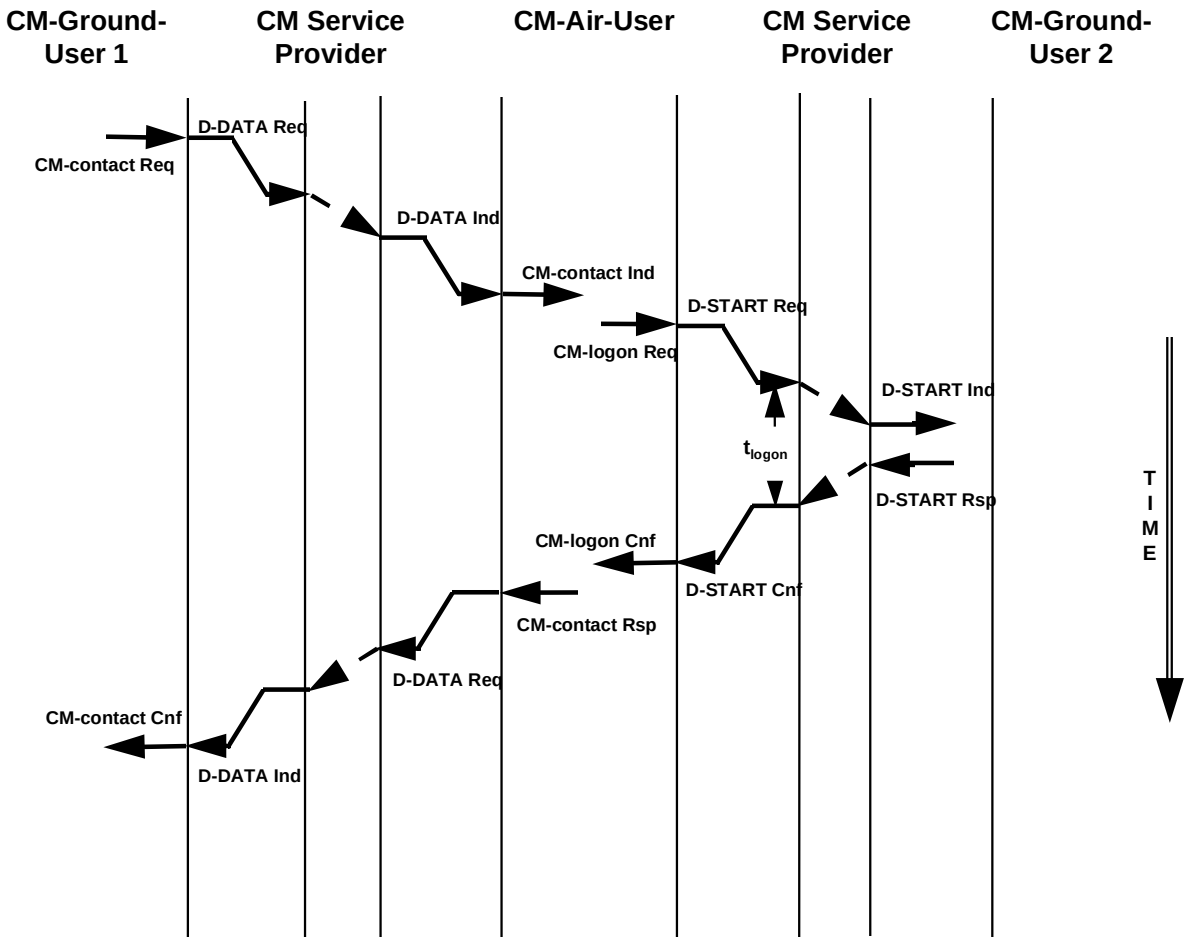

Figure 5-8: Sequence Diagram for CM-contact Service
Existing CM Dialogue
With CM-logon Service as in Figure 5-2

Note 2.—This figure does not imply that only one CM-logon attempt is made for a given CM-Contact request.

5.1.5.5 CM-contact Without CM-logon as Requested No Existing CM Dialogue

Note.— The following sequence of messages, shown in Figure 5-9, occurs when the CM-contact service is initiated with no existing CM dialogue and the CM-logon service is not initiated as requested.

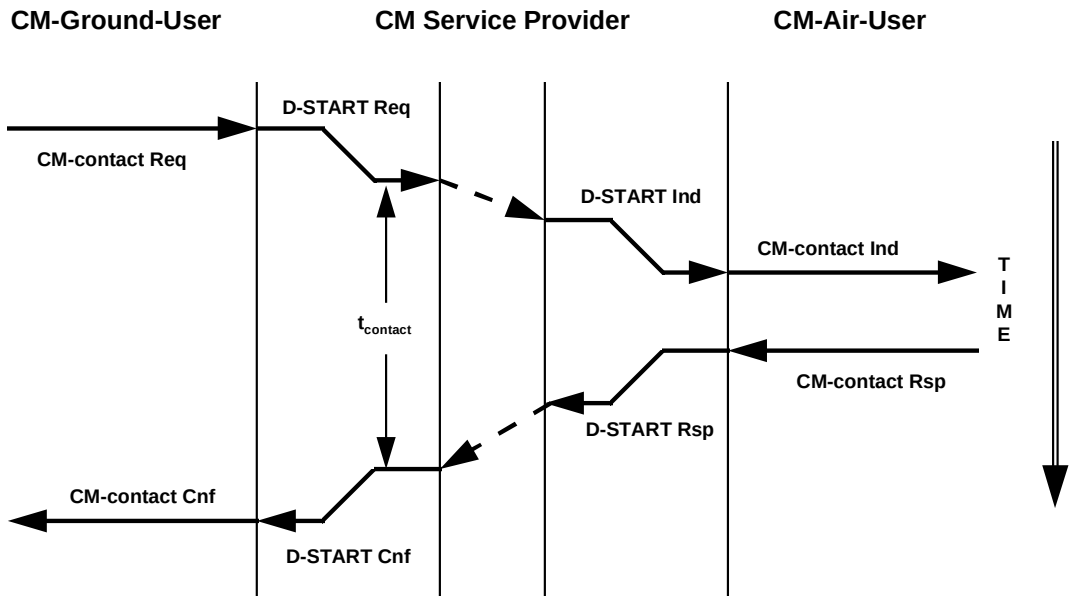

Figure 5-9: Sequence Diagram for CM-contact Service
No Existing CM Dialogue
Without CM-logon Service as Requested

5.1.5.6 CM-contact Without CM-logon as Requested With Existing CM Dialogue

Note.— The following sequence of messages, shown in Figure 5-10, occurs when the CM-contact service is initiated with existing CM dialogue and the CM-logon service is not initiated as requested.

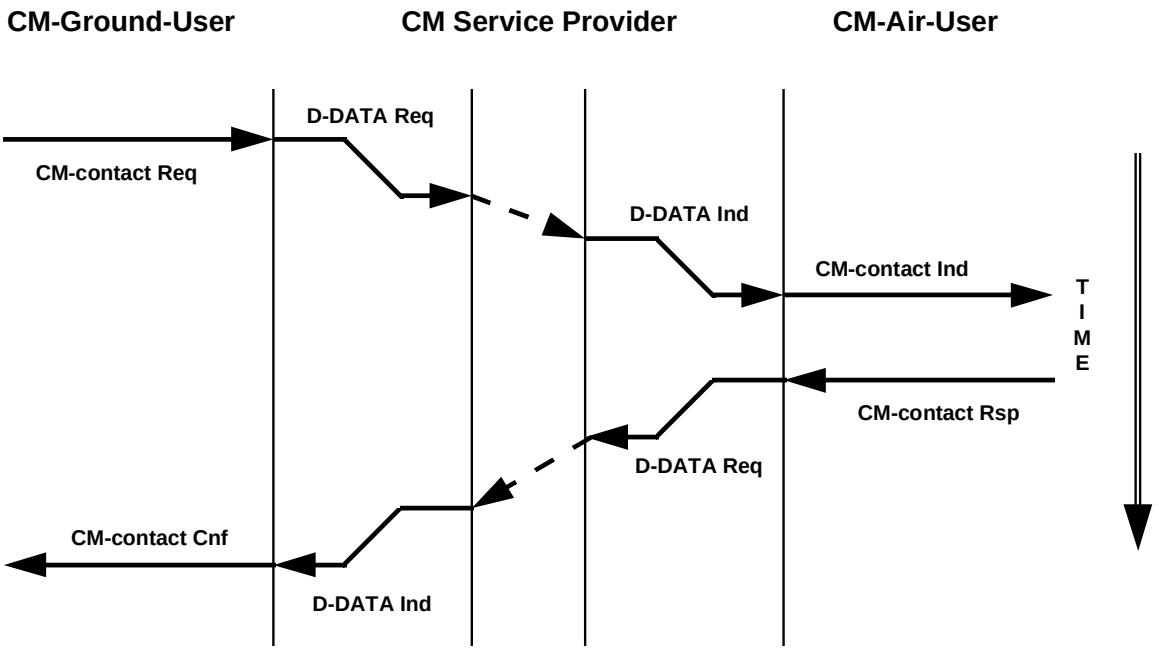

Figure 5-10: Sequence Diagram for CM-contact Service
With Existing CM Dialogue
Without CM-logon Service as Requested

5.1.6 CM-end Service

Note.— The following sequence of messages, shown in Figure 5-11, occurs when the CM-end service is initiated by the CM-ground-user.

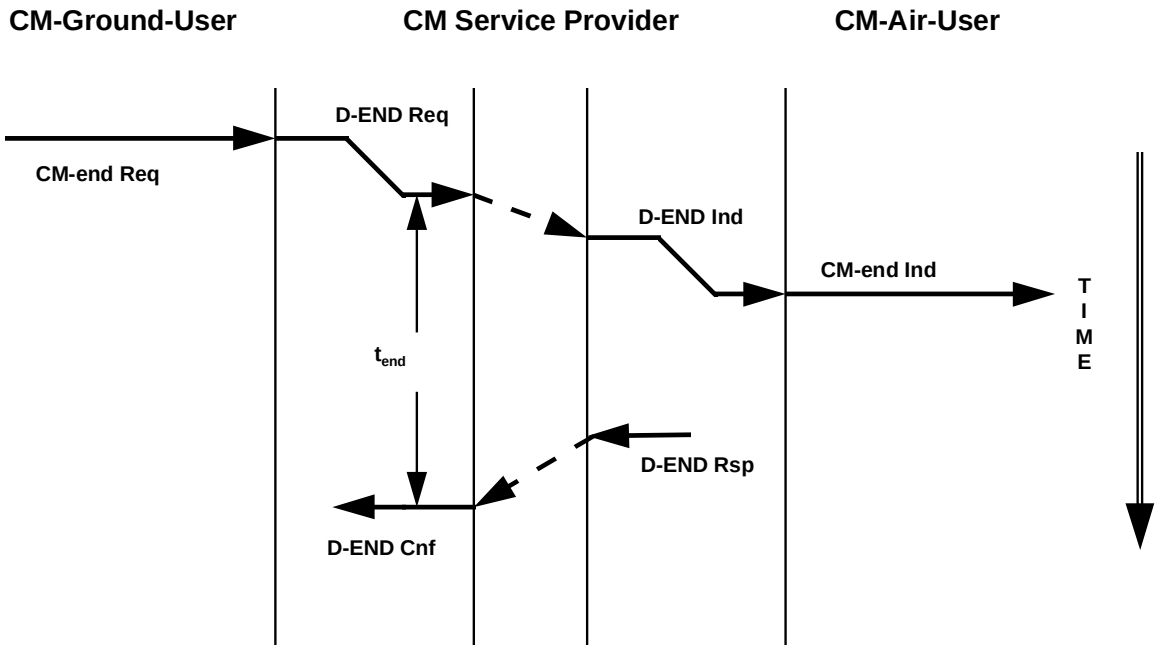

Figure 5-11: Sequence Diagram for CM-end Service

5.1.7 CM-forward Service (Ground-Ground)

5.1.7.1 Sending CM-Ground-ASE Version Less Than Or Equal To The Receiving CM-Ground-ASE Version

Note.— The following sequence of messages, shown in Figure 5-12, occurs when the CM-forward service is initiated with a sending CM-ground-ASE version number less than or equal to the receiving CM-ground-ASE version number.

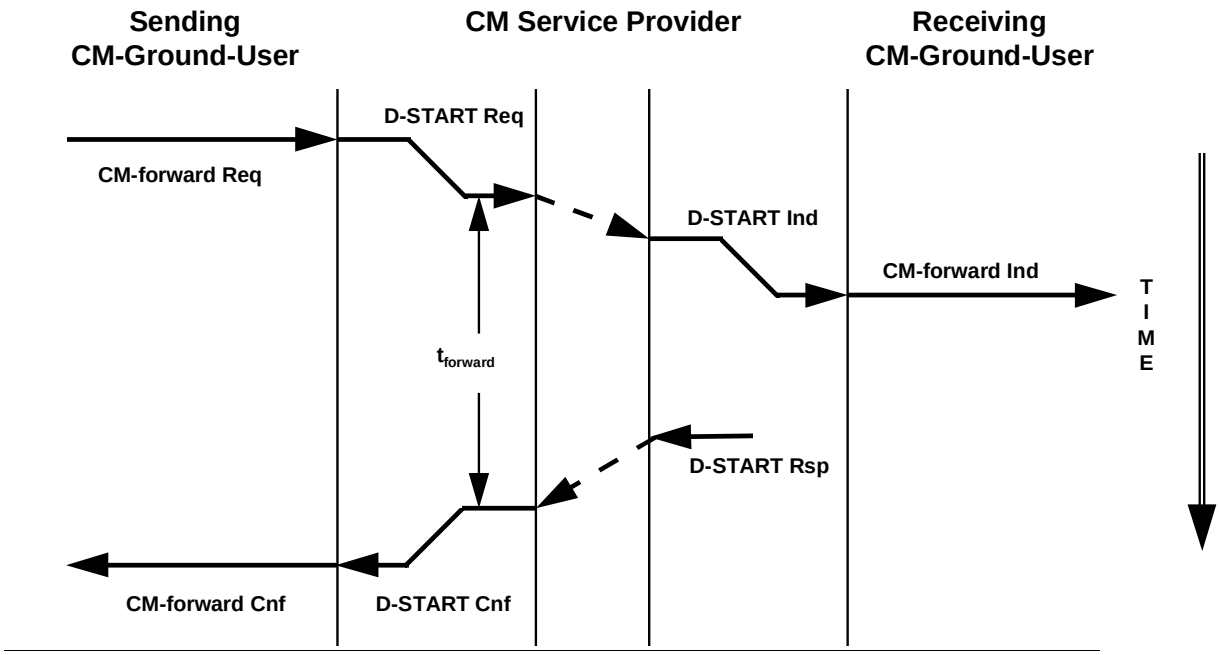

Figure 5-12: Sequence Diagram for CM-forward Service
Sending CM-Ground-ASE Version ≤ Receiving CM-Ground-ASE Version

5.1.7.2 Sending CM-Ground-ASE Version Greater Than The Receiving CM-Ground-ASE Version

Note.— The following sequence of messages, shown in Figure 5-13, occurs when the CM-forward service is initiated with a sending CM-ground-ASE version number greater than the receiving CM-ground-ASE version number.

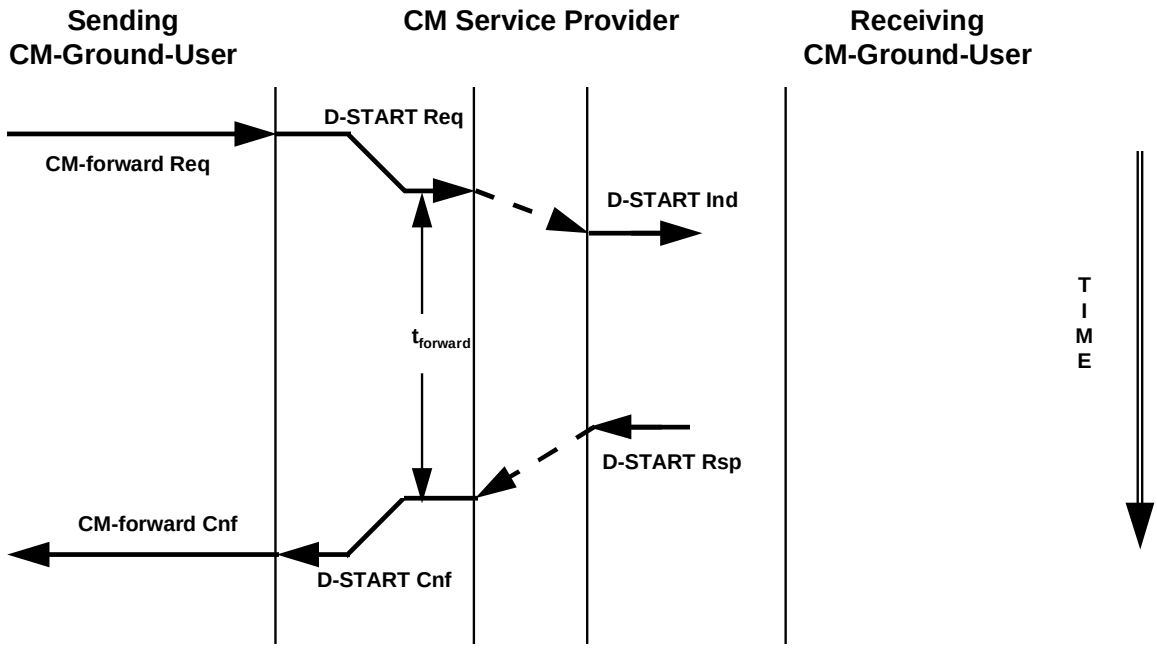

Figure 5-13: Sequence Diagram for CM-forward Service
Sending CM-Ground-ASE Version >Receiving CM-Ground-ASE Version

5.1.8 CM-user-abort Service

5.1.8.1 CM-user-abort Service; CM-Air-User Initiated

Note.— The following sequence of messages, shown in Figure 5-14, occurs when the CM-user-abort service is initiated by the CM-air-user.

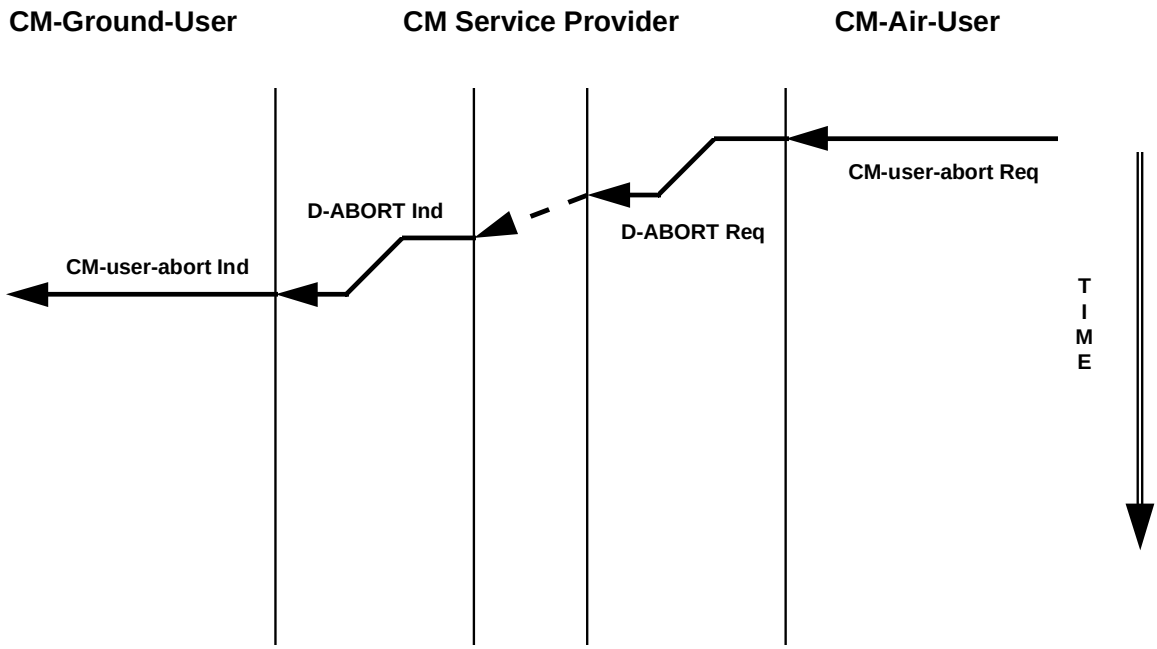

Figure 5-14: Sequence Diagram for CM-user-abort Service
CM-Air-User Initiated

5.1.8.2 CM-user-abort Service, CM-Ground-User Initiated

Note.— The following sequence of messages, shown in Figure 5-15, occurs when the CM-user-abort service is initiated by the CM-ground-user.

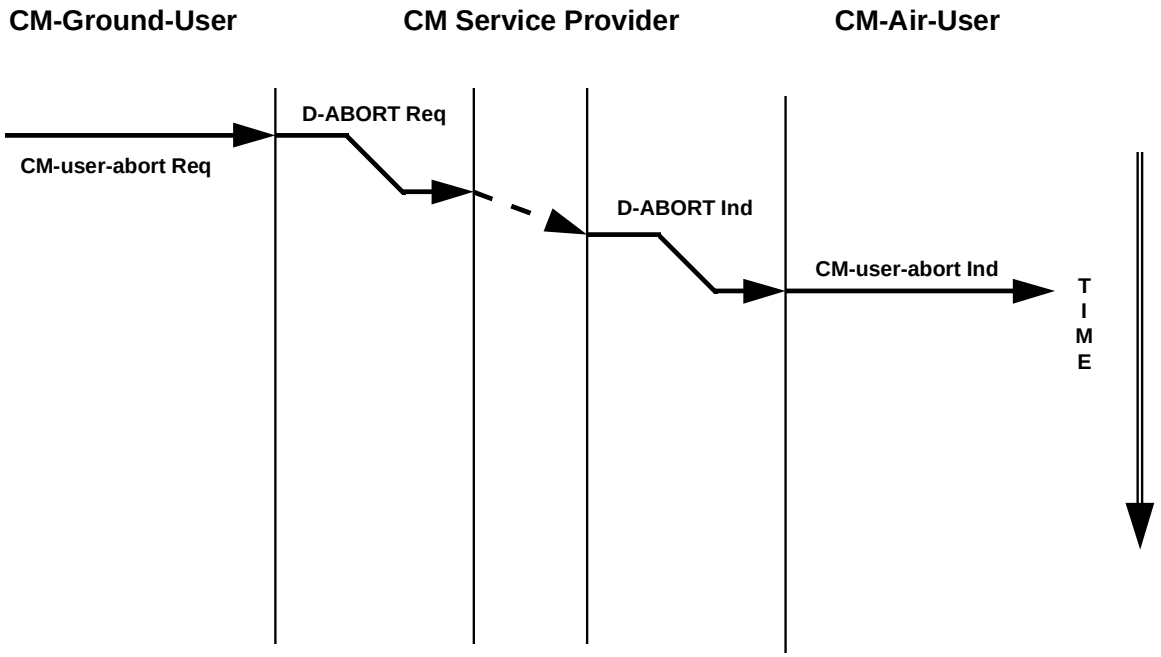

Figure 5-15: Sequence Diagram for CM-user-abort Service
CM-Ground-User Initiated

5.1.9 CM-provider-abort Service

5.1.9.1 Dialogue Service Abort

Note.— The following sequence of messages, shown in Figure 5-16, occurs when the dialogue service provider (below the level of the ASE) aborts.

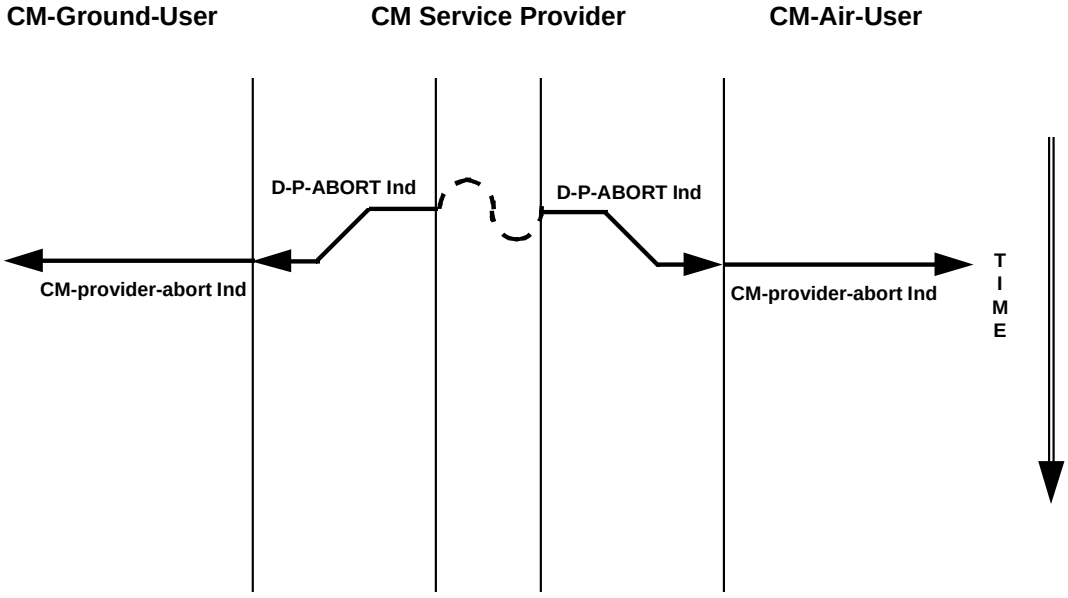

Figure 5-16: Sequence Diagram for CM-provider-abort Service: Dialogue Service Abort

5.1.9.2 CM-Air-ASE Abort

Note.— The following sequence of messages, shown in Figure 5-17, occurs when the CM-air-ASE aborts.

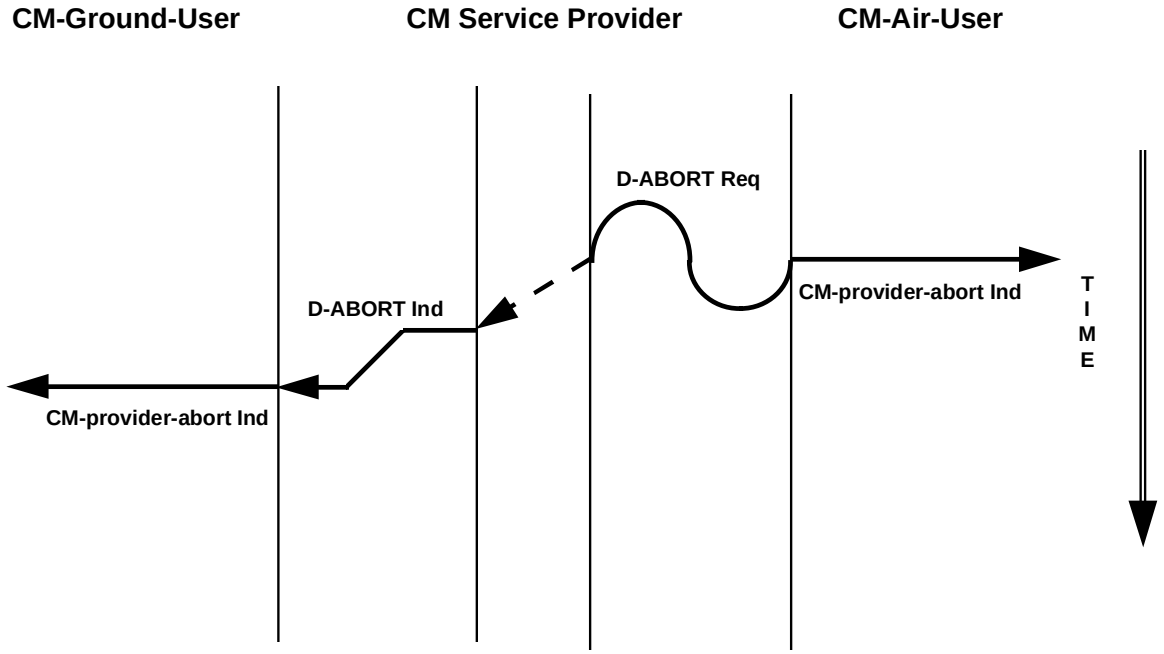

Figure 5-17: Sequence Diagram for CM-provider-abort Service: CM-Air-ASE Abort

5.1.9.3 CM-Ground-ASE Abort

Note.— The following sequence of messages, shown in Figure 5-18, occurs when the CM-ground-ASE aborts.

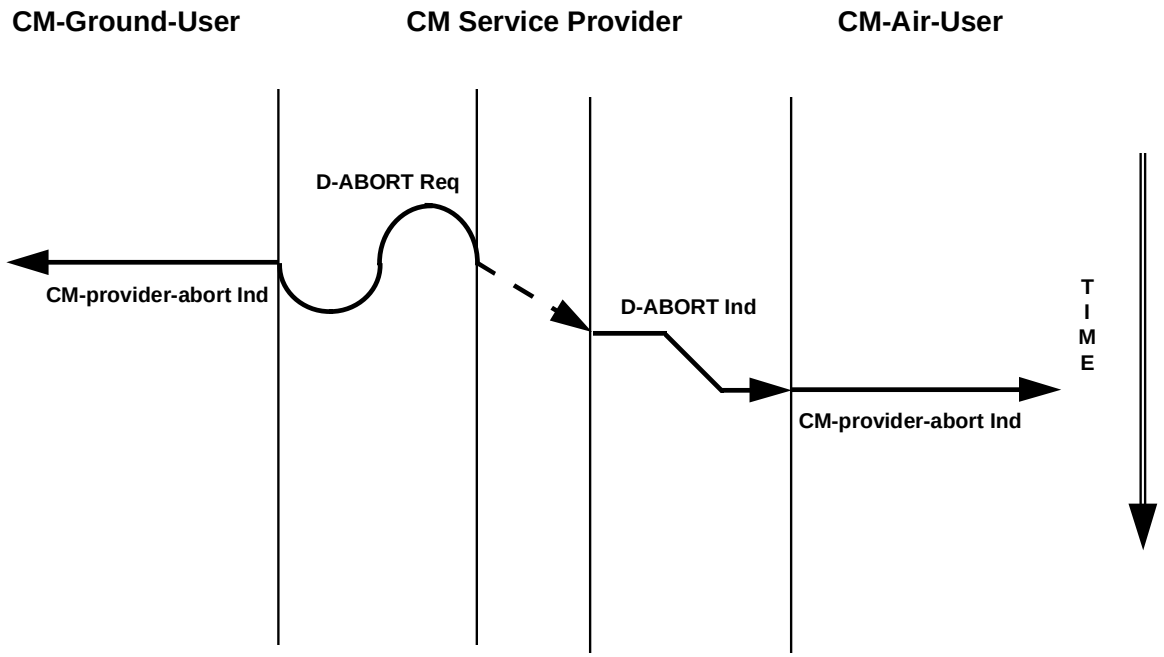

Figure 5-18: Sequence Diagram for CM-provider-abort Service: CM-Ground-ASE Abort

5.1.10 CM-user-abort Service (Ground-Ground)

5.1.10.1 CM-user-abort Service; Receiving CM-Ground-User Initiated

Note.— The following sequence of messages, shown in Figure 5-18, occurs when the CM-user-abort service is initiated by the receiving CM-ground-user.

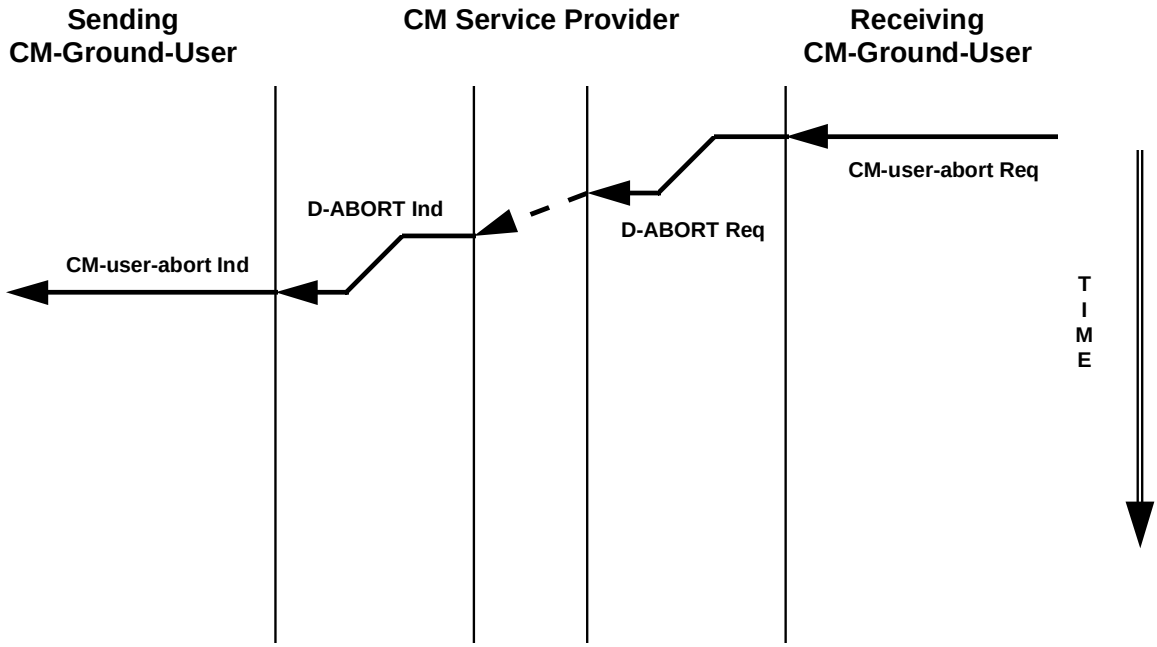

Figure 5-18: Sequence Diagram for CM-user-abort Service Receiving CM-Ground-User Initiated

5.1.10.2 CM-user-abort Service, Sending CM-Ground-User Initiated

Note.— The following sequence of messages, shown in Figure 5-19, occurs when the CM-user-abort service is initiated by the Sending CM-ground-user.

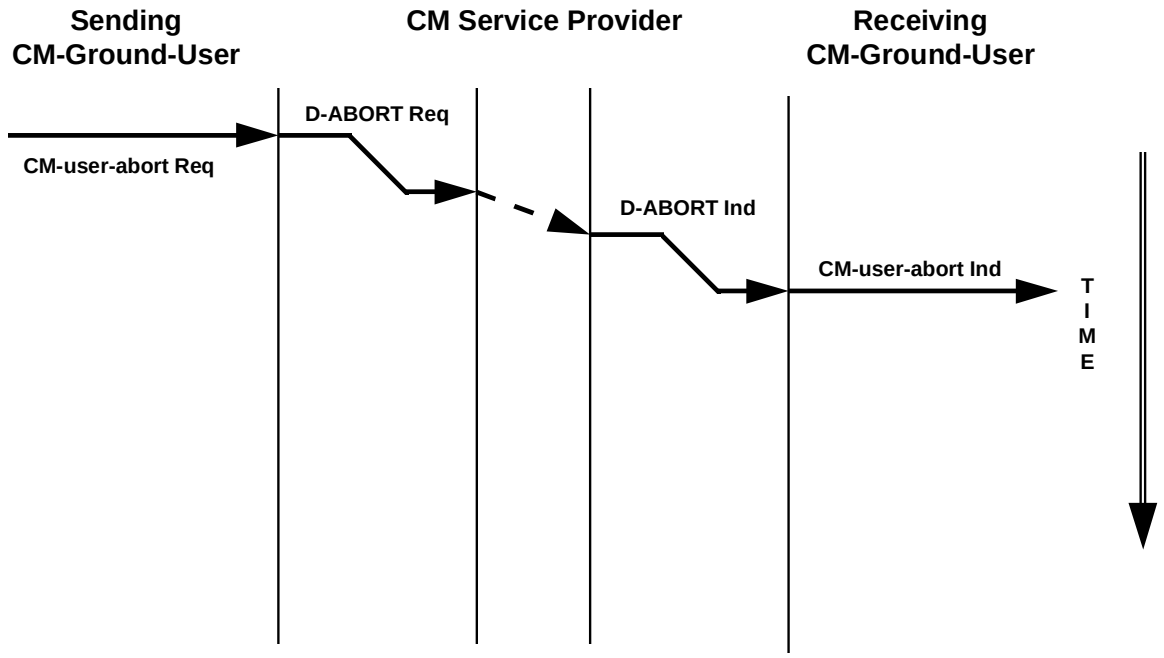

Figure 5-19: Sequence Diagram for CM-user-abort Service
Sending CM-Ground-User Initiated

5.1.11 CM-provider-abort Service (Ground-Ground)

5.1.11.1 Dialogue Service Abort

Note.— The following sequence of messages, shown in Figure 5-20, occurs when the dialogue service provider (below the level of the ASE) aborts.

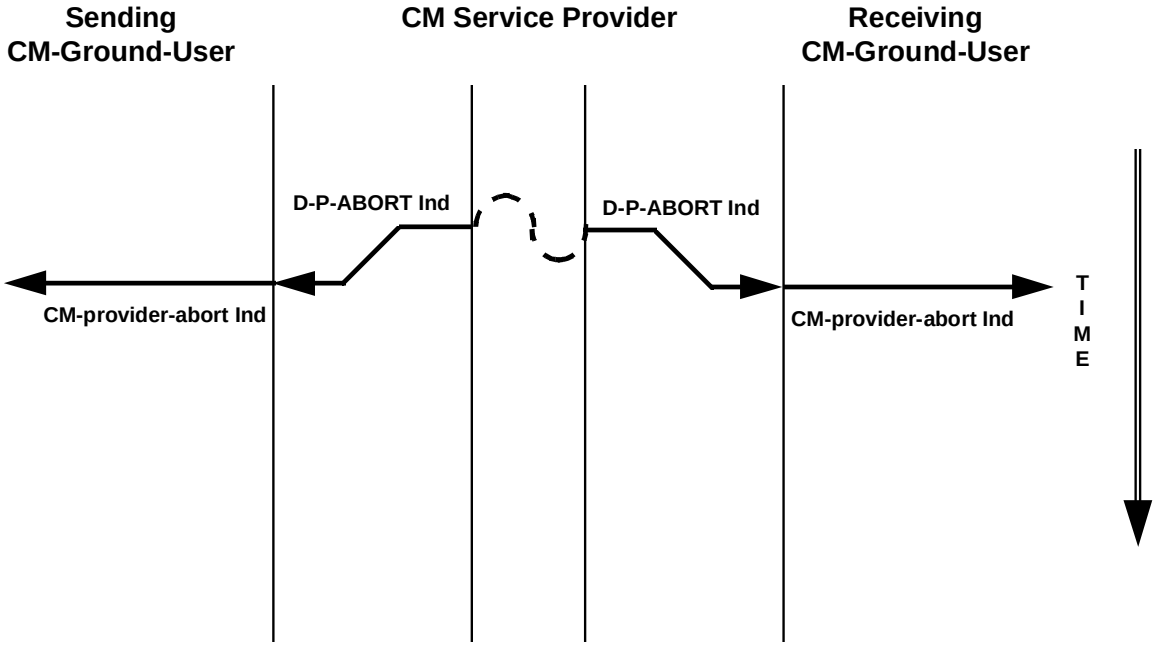

Figure 5-20: Sequence Diagram for CM-provider-abort Service: Dialogue Service Abort

5.1.11.2 Receiving CM-Ground-ASE Abort

Note.— The following sequence of messages, shown in Figure 5-21, occurs when the receiving CM-ground-ASE aborts.

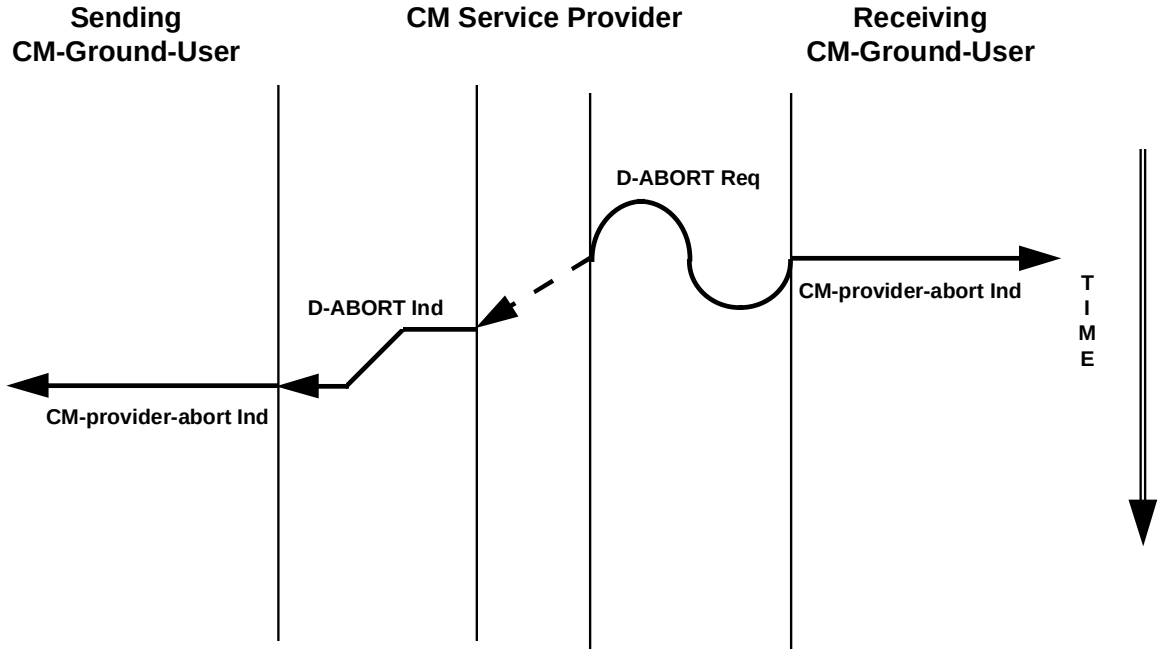

Figure 5-21: Sequence Diagram for CM-provider-abort Service: Receiving CM-Ground-ASE Abort

5.1.11.3 CM-Ground-ASE Abort

Note.— The following sequence of messages, shown in Figure 5-22, occurs when the Sending CM-ground-ASE aborts.

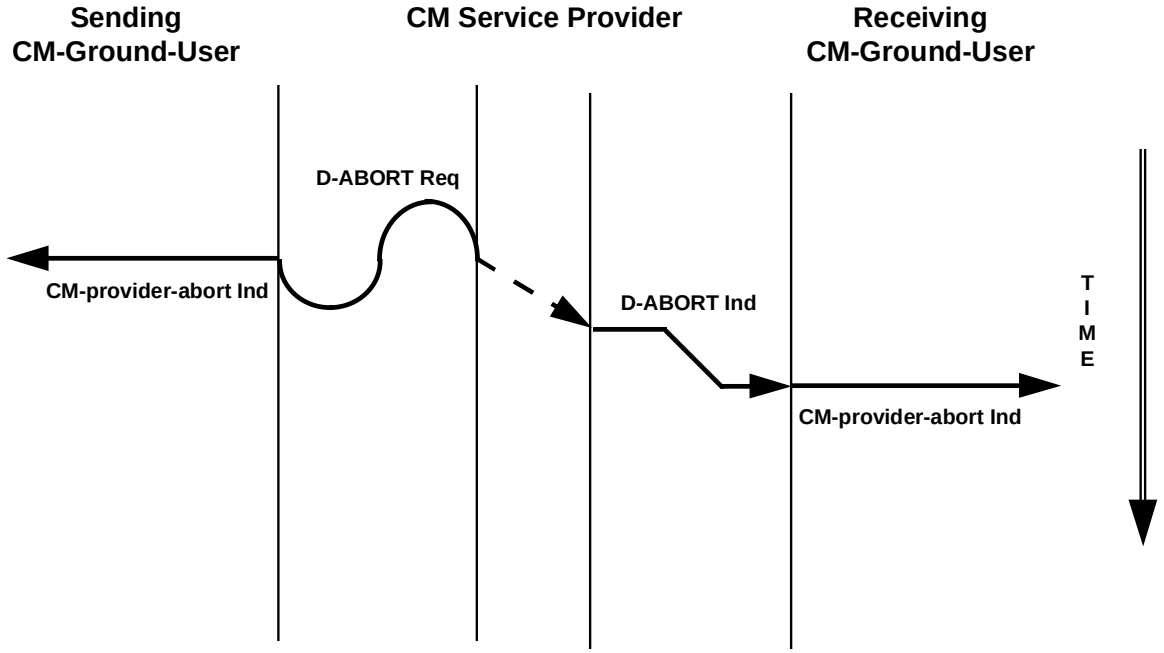

Figure 5-22: Sequence Diagram for CM-provider-abort Service: Sending CM-Ground-ASE Abort

5.2 CM Service ProviderApplication Timers

Note 1: — Table 5-1 lists the time constraints related to the CM application. Each time constraint requires a time to be set in the CM protocol machine.

5.2.1 A CM-ASE shall be capable of detecting when a timer expires.measure the time between the initial event and the corresponding final event

Note 1.— Table 5-1 lists the time constraints related to the CM application. Each time constraint requires a timer to be set in the CM protocol machine.

Note 2.— If the timer expires maximum time is exceeded before the final event has occurred, a CM-ASE shall takes the appropriate action as defined in section 5.4.1.

5.2.2 The timer valuesaction shouldshall be taken when the maximum time as indicated in Table 5-1 has expired.

CM Service	Timer	Timer Value	Timer Start Event	Timer Stop Event
CM-logon	t_{logon}	4 min	D-START request	D-START confirmation
CM-update	t_{update}	4 min	D-START request	D-START confirmation
CM-contact	$t_{contact}$	8 min	D-START request	D-START confirmation
CM-forward	$t_{forward}$	4 min	D-START request	D-START confirmation
CM-end	t_{end}	4 min	D-END request	D-END confirmation

Table 5-1: CM Service ProviderApplication Timers

5.3 CM-ASE Protocol Description

5.3.1 Introduction

Note.— This section presents requirements for CM-ASEs in specific states. Section 5.5 contains state tables for the CM-ASEs.

5.3.1.1 If no actions~~requirements~~ are described for a CM service primitive when a CM-ASE is in a specific state, then the invocation of that service primitive shall be prohibited while the CM-ASE is in that state.

5.3.1.2 Upon receipt of a PDU, ~~I~~if no actions~~requirements~~ are described for the arrival of that~~a~~ PDU when a CM-ASE is in a specific state, then exception handling procedures as described in section 5.4.4 shall apply.

5.3.2 CM-Air-ASE Protocol Description

Note.— The states defined for the CM-air-ASE are the following:

a) IDLE,
b) LOGON,
c) CONTACT,
d) DIALOGUE, and
e) CONTACT DIALOGUE.

5.3.2.1 On initiation, the CM-air-ASE shall be in the *IDLE* state.

5.3.2.2 D-START Indication

5.3.2.2.1 Upon receipt of a D-START indication, if the CM-air-ASE is in the *IDLE* state then:

5.3.2.2.1.1 If the APDU contained in the D-START *User Data* parameter is a [CMUpdate] APDU the CM-air-ASE shall:

a) invoke CM-update service indication with the following:
   1) the D-START *Calling Peer ID* parameter value as the CM-update *ICAO Facility Designator* parameter value, and
   2) the APDU contained in the D-START *User Data* parameter as the CM-update *Update Information* parameter value,
b) invoke D-START response with "Rejected (permanent)" provided as D-START *Result* parameter value, and
c) enter the *IDLE* state.

5.3.2.2.1.2 If the APDU contained in the D-START *User Data* parameter is a [CMContactRequest] APDU the CM-air-ASE shall:

a) invoke CM-contact service indication with the following:
   1) The D-START *Calling Peer ID* parameter value as the CM-contact *ICAO Facility Designator* parameter value, and
   2) The APDU contained in the D-START *User Data* parameter as the CM-contact *Contact Request* parameter value, and
b) enter the *CONTACT* state.

5.3.2.3 D-START Confirmation

5.3.2.3.1.1 Upon receipt of a D-START confirmation, if the CM-air-ASE is in the *LOGON* state and if the APDU contained in the D-START *User Data* parameter is a [CMLogonResponse] APDU the CM-air-ASE shall:

a) stop timer t_{logon},
b) if the abstract value of the D-START *Result* parameter is "rejected (permanent)" and the abstract value of the D-START *Reject Source* parameter is "DS user" then:
   1) if the version number of the CM-air-ASE is greater than the D-START *DS User Version Number* parameter value, ~~the CM-air-ASE shall~~ invoke CM-logon service confirmation with the *DS User Version Number* parameter value as the CM-logon *CM-ASE Version Number* parameter value, or

2) else ~~the CM-air-ASE shall~~ invoke CM-logon service confirmation with the APDU contained in the D-START *User Data* parameter as the CM-logon *Logon Response* parameter, and
3) enter the *IDLE* state.

c) if the abstract value of the D-START *Result* parameter is "accepted" then:
1) if the version number of the CM-air-ASE is greater than the D-START *DS User Version Number* parameter value, ~~the CM-air-ASE shall~~ invoke CM-logon service confirmation with the D-START *DS User Version Number* parameter value as the CM-logon *CM ASE Version Number* parameter value, or
2) else ~~the CM-air-ASE shall~~ invoke CM-logon service confirmation with the APDU contained in the D-START *User Data* parameter as the CM-logon *Logon Response* parameter value, and
3) enter the *DIALOGUE* state.

5.3.2.4 D-DATA Indication

5.3.2.4.1 Upon receipt of a D-DATA indication, if the CM-air-ASE is in the *DIALOGUE* state then:

5.3.2.4.1.1 If the APDU contained in the D-DATA *User Data* parameter is a [CMUpdate] APDU the CM-air-ASE shall:

a) invoke CM-update service indication with the APDU contained in the D-START *User Data* parameter as the CM-update service *Update Information* parameter value, and
b) remain in the *DIALOGUE* state.

5.3.2.4.1.2 If the APDU contained in the D-DATA *User Data* parameter is a [CMContactRequest] APDU the CM-air-ASE shall:

a) invoke CM-contact service indication with the APDU contained in the D-START *User Data* parameter as the CM-contact service *Contact Request* parameter value, and
b) enter the *CONTACT DIALOGUE* state.

5.3.2.5 D-END Indication

5.3.2.5.1.1 Upon receipt of a D-END indication, if the CM-air-ASE is in the *DIALOGUE* state then the CM-air-ASE shall:

a) invoke CM-end service indication,
b) invoke D-END response with the D-END *Result* parameter set to the abstract value "accepted", and
c) enter the *IDLE* state.

5.3.2.6 CM-logon Service Request

5.3.2.6.1 Upon receipt of a CM-logon service request:

5.3.2.6.1.1 If the CM-air-ASE is in the *IDLE* state the CM-air-ASE shall:

a) create a CMAircraftMessage APDU with a cmLogonRequest APDU-element based on the *Logon Request* parameter value,
b) invoke D-START request with the following:
1) the CM-logon *ICAO Facility Designator* parameter value as the D-START *Called Peer ID* parameter value,
2) the CM-logon *Aircraft Identifier* parameter value as the D-START *Calling Peer ID* parameter value,
3) the CM-air-ASE version number as the D-START *DS User Version Number* parameter value,
4) the D-START *Quality of Service* parameters set as follows:
i) if provided, the CM-logon service *Class of Communication* parameter value as the D-START *QOS Routing Class* parameter value,
ii) the abstract value of "flight regularity communications", as the D-START *QOS Priority* parameter value, and
iii) the abstract value of "low" as the D-START QOS *Residual Error Rate*, and

5) the CMAircraftMessage APDU as the D-START *User Data* parameter value,

c) start timer t_{logon}, and

d) enter the *LOGON* state.

5.3.2.7 CM-contact Service Response

5.3.2.7.1 Upon receipt of a CM-contact service response, if the CM-air-ASE is in the *CONTACT* state the CM-air-ASE shall:

a) create a CMAircraftMessage APDU with cmContactResponse APDU-element based on the CM-contact *Contact Response* parameter value,

b) invoke D-START response with the following:

1) the abstract value "rejected (permanent)" as D-START *Result* parameter value, and

2) the CMAircraftMessage APDU as the D-START *User Data* parameter value, and

c) enter the *IDLE* state.

5.3.2.7.2 Upon receipt of a CM-contact service response, if the CM-air-ASE is in the *CONTACT DIALOGUE* state the CM-air-ASE shall:

a) create a CMAircraftMessage APDU with a cmContactResponse APDU-element based on the CM-contact *Contact Response* parameter value,

b) invoke D-DATA request with the CMAircraftMessage APDU as the D-DATA *User Data* parameter value, and

c) enter the *DIALOGUE* state.

5.3.2.8 CM-user-abort Service Request

5.3.2.8.1 Upon receipt of a CM-user-abort service request, if the CM-air-ASE is not in the *IDLE* state the CM-air-ASE shall:

a) stop timer t_{logon}, if set,

b) invoke D-ABORT request with the D-ABORT *Originator* parameter set to the abstract value "user", and

c) enter the *IDLE* state.

5.3.2.9 D-ABORT Indication

5.3.2.9.1 Upon receipt of a D-ABORT indication, if the CM-air-ASE is not in the *IDLE* state the CM-air-ASE shall:

a) stop timer t_{logon}, if set

b) if the CM-air-user is an active user, then:

1) if the D-ABORT *Originator* parameter contains the abstract value "user" invoke CM-user-abort service indication, or

2) else invoke CM-provider-abort service indication with the APDU contained in the D-ABORT *User Data* parameter as the CM-provider-abort service *Reason* parameter value, and

c) enter *IDLE* state.

5.3.2.10 D-P-ABORT Indication

5.3.2.10.1 Upon receipt of a D-P-ABORT indication, if the CM-air-ASE is not in the *IDLE* state the CM-air-ASE shall:

a) stop timer t_{logon}, if set,

b) if the CM-air-user is an active user, invoke CM-provider-abort service indication with the CM-provider-abort *Reason* parameter set to the abstract value "communication-service-failure", and

c) enter the *IDLE* state.

5.3.3 CM-Ground-ASE Protocol Description

Note.— The states defined for the CM-ground-ASE are the following:

a) IDLE,
b) LOGON,
c) UPDATE,
d) CONTACT,
e) DIALOGUE,
f) CONTACT DIALOGUE,
g) END, and
h) FORWARD.

5.3.3.1 On initiation, the CM-ground-ASE shall be in the *IDLE* state.

5.3.3.2 D-START Indication

5.3.3.2.1 Upon receipt of a D-START indication, if the CM-ground-ASE is in the *IDLE* state and the APDU contained in the D-START *User Data* parameter is a [CMLogonRequest] APDU then:

5.3.3.2.1.1 If the *DS User Version Number* is greater than the CM-ground-ASE version number the CM-ground-ASE shall:

a) invoke D-START response with the following:
   1) the CM-ground-ASE version number as the D-START *DS User Version Number* parameter value, and
   2) the abstract value of "rejected (permanent)" as the D-START *Result* parameter value, and
b) enter the *IDLE* state.

5.3.3.2.1.2 If the *DS User Version Number* is less than the CM-ground-ASE version number the CM-ground-ASE shall:

a) invoke CM-logon service indication with the following:
   1) the D-START *Calling Peer ID* parameter value as the CM-logon service *Aircraft Identifier* parameter value,
   2) the D-START *DS User Version Number* parameter value as the CM-logon service *CM ASE Version Number* parameter value, and
   3) the APDU in the D-START *User Data* parameter as the CM-logon service *Logon Request* parameter value, and
b) enter the *LOGON* state.

5.3.3.2.1.3 If the~~Else (the~~ *DS User Version Number* is equal to CM-ground-ASE version number~~)~~ the CM-ground-ASE shall:

a) invoke CM-logon service indication with:
   1) the D-START *Calling Peer ID* parameter value as the CM-logon service *Aircraft Identifier* parameter value, and
   2) the APDU in the D-START *User Data* parameter as the CM-logon service *Logon Request* parameter value, and
b) enter the *LOGON* state

5.3.3.2.2 Upon receipt of a D-START indication, if the receiving CM-ground-ASE is in the *IDLE* state and the APDU contained in the D-START *User Data* parameter is a [CMForwardRequest] APDU then ~~the receiving CM-ground-ASE shall~~:

5.3.3.2.2.1 If the *DS User Version Number* is greater than the receiving CM-ground-ASE version number the receiving CM-ground-ASE shall:

a) invoke D-START response with the following:
   1) the receiving CM-ground-ASE version number as the D-START *DS User Version Number* parameter value, and
   2) the abstract value of "rejected (permanent)" as the D-START *Result* parameter value, and
b) enter the *IDLE* state.

5.3.3.2.2.2 If the *DS User Version Number* is less than the receiving CM-ground-ASE version number the receiving CM-ground-ASE shall:

a) invoke CM-forward service indication with the following:
   1) the D-START *Calling Peer ID* parameter value as the CM-forward service *Calling ICAO Facility Designator* parameter value,
   2) the D-START *DS User Version Number* parameter value as the CM-forward service *CM ASE Version Number* parameter value, and
   3) the APDU in the D-START *User Data* parameter as the CM-forward service *Forward Request* parameter value,
b) invoke D-START response with the abstract value of "rejected (permanent)" as the D-START *Result* parameter value, and
c) enter the *IDLE* state.

5.3.3.2.2.3 IfElse (the *DS User Version Number* is equal to receiving CM-ground-ASE version number) the receiving CM-ground-ASE shall:

a) invoke CM-forward service indication with:
   1) the D-START *Calling Peer ID* parameter value as the CM-forward service *Calling ICAO Facility Designator* parameter value, and
   2) the APDU in the D-START *User Data* parameter as the CM-forward service *Forward Request* parameter value,
b) invoke D-START response with the abstract value of "rejected (permanent)" as the D-START *Result* parameter value, and
c) enter the *IDLE* state.

5.3.3.3 D-START Confirmation

5.3.3.3.1 Upon receipt of a D-START confirmation:

5.3.3.3.1.1 If the CM-ground-ASE is in the *UPDATE* state and the D-START *User Data* parameter is not provided, the CM-ground-ASE shall:

a) stop timer t_{update}, and
b) if the abstract value of the D-START *Result* parameter is "rejected (permanent)" and the abstract value of the D-START *Reject Source* parameter is "DS user", the CM-ground-ASE shall enter the *IDLE* state.

5.3.3.3.1.2 If the CM-ground-ASE is in the *CONTACT* state and the APDU contained in the D-START *User Data* parameter is a [CMContactResponse] APDU, the CM-ground-ASE shall:

a) stop timer $t_{contact}$, and
b) if the abstract value of the D-START *Result* parameter is "rejected (permanent)" and the abstract value of the D-START *Reject Source* parameter is "DS user" then:
   1) invoke CM-contact service confirmation with the APDU in the D-START *User Data* parameter as the CM-contact *Contact Response* parameter value, and
   2) enter the *IDLE* state.

5.3.3.3.1.3 If the CM-ground-ASE is in the *FORWARD* state, and the D-START *User Data* parameter is not provided, the CM-ground-ASE shall:

a) stop timer $t_{forward}$, and
b) if the abstract value of the D-START *Result* parameter is "rejected (permanent)" and the abstract value of the D-START *Reject Source* parameter is "DS user" then:
   1) if the version number of the sending CM-ground-ASE is greater than the D-START *DS User Version Number* parameter value, the sending CM-ground-ASE shall invoke CM-forward service confirmation with the *DS User Version Number* parameter value as the CM-forward *CM-ASE Version Number* parameter value, and set the *Result* parameter abstract value to "failure",

2) ~~else (~~if the version number of the sending CM-ground-ASE is less than or equal to the version number of the receiving CM-ground-ASE~~)~~ the CM-ground-ASE shall invoke the CM-forward service confirmation with the *Result* parameter abstract value set to "success", and
3) enter the *IDLE* state.

5.3.3.4 D-DATA Indication

5.3.3.5 Upon receipt of a D-DATA indication if the CM-ground-ASE is in the *CONTACT DIALOGUE* state and the APDU contained in the D-DATA *User Data* parameter is a [CMContactResponse] APDU, the CM-ground-ASE shall:

a) invoke CM-contact service confirmation with the APDU contained in the D-DATA *User Data* parameter as the CM-contact *Contact Response* parameter value, and
b) enter the *DIALOGUE* state.

5.3.3.6 D-END Confirmation

5.3.3.6.1 Upon receipt of a D-END confirmation, if the CM-ground-ASE is in the *END* state and the abstract value of the D-END *Result* is "accepted", the CM-ground-ASE shall:

a) stop timer t_{end}, and
b) enter the *IDLE* state.

5.3.3.7 CM-logon Service Response

5.3.3.7.1 Upon receipt of a CM-logon service response, if the CM-ground-ASE is in the *LOGON* state the CM-ground-ASE shall:

a) create a CMGroundMessage APDU with a cmLogonResponse APDU-element based on the CM-logon *Logon Response* parameter value, and
b) invoke D-START response with the following:
   1) the CMGroundMessage APDU as the D-START *User Data* parameter value,
   2) if the CM-logon *Maintain Dialogue* parameter is provided by the CM-ground-user
      i) set the abstract value "accepted" as the D-START *Result* parameter value, and
      ii) enter the *DIALOGUE* state.
   3) if ~~else (~~the CM-logon *Maintain Dialogue* parameter is not provided by the CM-ground-user~~)~~
      i) set the abstract value "rejected (permanent)" as the D-START *Result* parameter value, and
      ii) enter the *IDLE* state.

5.3.3.8 CM-update Service Request

5.3.3.8.1 Upon receipt of a CM-update service request, if the CM-ground-ASE is in the *IDLE* state, the CM-ground-ASE shall:

a) create a CMGroundMessage APDU with a cmUpdate APDU-element based on the CM-update *Update Information* parameter value,
b) invoke D-START request with the following:
   1) the CM-update *Aircraft Identifier* parameter value as the D-START *Called Peer ID* parameter value,
   2) the CM-update *ICAO Facility Designator* parameter value as the D-START *Calling Peer ID* parameter value,
   3) set the D-START *Quality of Service* parameter as follows:
      i) *Class of Communication* from the CM-update request if provided by the CM-user as the D-START *QOS Routing Class* parameter value,
      ii) the abstract value of "flight regularity communications", as the D-START *QOS Priority* parameter value, and
      iii) the abstract value of "low" as the D-START QOS *Residual Error Rate,*
   4) the CMGroundMessage APDU as the D-START *User Data* parameter value,

c) set timer t_{update}, and
d) enter the *UPDATE* state.

5.3.3.8.2 Upon receipt of a CM-update service request, if the CM-ground-ASE is in the *DIALOGUE* state, the CM-ground-ASE shall:

a) create a CMGroundMessage APDU with a cmUpdate APDU-element based on the CM-update *Update Information* parameter value,
b) invoke D-DATA request with the CMGroundMessage APDU as the D-START *User Data* parameter value, and
c) remain in the *DIALOGUE* state.

5.3.3.9 CM-contact Service Request

5.3.3.9.1 Upon receipt of a CM-contact service request, if the CM-ground-ASE is in the *IDLE* state the CM-ground-ASE shall:

a) create a CMGroundMessage APDU with a cmContactRequest APDU-element based on the CM-contact *Contact Request* parameter value,
b) invoke D-START request with the following:
    1) the CM-contact *Aircraft Identifier* parameter value as the D-START *Called Peer ID* parameter value,
    2) the CM-contact *ICAO Facility Designator* parameter value as the D-START *Calling Peer ID* parameter value,
    3) set the D-START *Quality of Service* parameters as follows:
        i) *Class of Communication* from the CM-contact request if provided by the CM-user as the D-START *QOS Routing Class* parameter value,
        ii) The abstract value of "flight regularity communications", as the D-START *QOS Priority* parameter value, and
        iii) The abstract value of "low" as the D-START *QOS Residual Error Rate* parameter value,
    4) the CMGroundMessage APDU as the D-START *User Data* parameter value,
c) start timer $t_{contact}$, and
d) enter the *CONTACT* state.

5.3.3.9.2 Upon receipt of a CM-contact service request, if the CM-ground-ASE is in the *DIALOGUE* state the CM-ground-ASE shall:

a) create a CMGroundMessage APDU with a cmContactRequest APDU-element based on the CM-contact *Contact Request* parameter value,
b) invoke D-DATA request with the CMGroundMessage APDU as the D-START *User Data* parameter value, and
c) enter the *CONTACT DIALOGUE* state.

5.3.3.10 CM-end Service Request

5.3.3.10.1 Upon receipt of a CM-end service request, if the CM-ground-ASE is in the *DIALOGUE* state the CM-ground-ASE shall:

a) invoke D-END request,
b) start timer t_{end} , and
c) enter the *END* state.

5.3.3.11 CM-forward Service Request

5.3.3.11.1 Upon receipt of a CM-forward service request, if the CM-ground-ASE is in the *IDLE* state, the CM-ground-ASE shall:

a) create a CMGroundMessage APDU with a cmForwardRequest APDU-element based on the CM-forward service *Forward Request* parameter value,
b) invoke D-START request with the following:

1) the CM-forward *Called ICAO Facility Designator* parameter value as the D-START *Called Peer ID* parameter value,
2) the CM-forward *Calling ICAO Facility Designator* parameter value as the D-START *Calling Peer ID* parameter value,
3) the sending CM-ground-ASE version number as the D-START *DS User Version Number* parameter value,
4) set the D-START *Quality of Service* parameter as follows:
   i) *Class of Communication* from the CM-forward request if provided by the CM-user as the D-START *QOS Routing Class* parameter value,
   ii) the abstract value of "flight regularity communications" as the D-START *QOS Priority* parameter value, and
   iii) the abstract value of "low" as the D-START QOS *Residual Error Rate,*
5) the CMForwardMessage APDU as the D-START *User Data* parameter value,

c) set timer $t_{forward}$, and
d) enter the *FORWARD* state.

5.3.3.12 CM-user-abort Service Request

5.3.3.12.1 Upon receipt of a CM-user-abort service request, if the CM-ground-ASE is not in the *IDLE* state the CM-ground-ASE shall:

a) stop any timer, if set,
b) invoke D-ABORT request with the D-ABORT *Originator* parameter set to the abstract value "user", and
c) enter the *IDLE* state.

5.3.3.13 D-ABORT Indication

5.3.3.13.1.1 Upon receipt of a D-ABORT indication, if the CM-ground-ASE is not in the *IDLE* state the CM-ground-ASE shall:

a) stop any timer, if set,
b) if the CM-ground-user is an active user, then:
   1) if the D-ABORT *Originator* parameter contains the abstract value "user" invoke CM-user-abort service indication,
   2) else invoke CM-provider-abort service indication with the APDU contained in the D-ABORT *User Data* parameter as the CM-provider-abort service *Reason* parameter value, and
c) enter *IDLE* state.

5.3.3.14 D-P-ABORT Indication

5.3.3.14.1 Upon receipt of a D-P-ABORT indication, if the CM-ground-ASE is not in the *IDLE* state the CM-ground-ASE shall:

a) stop any timer, if set,
b) if the CM-ground-user is an active user, invoke CM-provider-abort service indication with the CM-provider-abort *Reason* parameter set to the abstract value "communication-service-failure", and
c) enter the *IDLE* state.

5.4 Exception Handling

5.4.1 Timer Expiration

5.4.1.1 If a CM-ASE detects that ~~any~~ timer has expired~~s~~, th~~at~~e CM-ASE shall:

a) interrupt any current activity,
b) if the CM-ASE is a CM-air-ASE, create a CMAircraftMessage APDU with a cmAbortReason [timer-expired] APDU message element,
c) if the CM-ASE is a CM-ground-ASE, create a CMGroundMessage APDU with a cmAbortReason [timer-expired] APDU message element,

d) invoke D-ABORT request with:
   1) the abstract value "provider" as the D-ABORT *Originator* parameter value, and
   2) the APDU as the D-ABORT *User Data* parameter value, and
e) if the CM-user is an active user, invoke CM-provider-abort service indication with the abstract value "timer-expired"~~APDU~~ as the CM-provider-abort *Reason* parameter value.

5.4.2 Unrecoverable System Error

5.4.2.1 If a CM-ASE has an unrecoverable system error, the CM-ASE shall:

a) if the CM-ASE is a CM-air-ASE, create a CMAircraftMessage APDU with a cmAbortReason [undefined-error] APDU message element,
b) if the CM-ASE is a CM-ground-ASE, create a CMGroundMessage APDU with a cmAbortReason [undefined-error] APDU message element,
c) invoke D-ABORT request with:
   1) the abstract value "provider" as the D-ABORT *Originator* parameter value, and
   2) the APDU as the D-ABORT *User Data* parameter value, and
d) if the CM-user is an active user, invoke CM-provider-abort service indication with the abstract value "undefined-error"~~APDU~~ as the CM-provider-abort *Reason* parameter value.

5.4.3 Invalid PDU

5.4.3.1 If the *User Data* parameter of a D-START indication or D-DATA indication does not contain a valid PDU, the CM-ASE shall:

a) if the CM-ASE is a CM-air-ASE, create a CMAircraftMessage APDU with a cmAbortReason [invalid-PDU] APDU message element,
b) if the CM-ASE is a CM-ground-ASE, create a CMGroundMessage APDU with a cmAbortReason [invalid-PDU] APDU message element,
c) invoke D-ABORT request with:
   1) the abstract value "provider" as the D-ABORT *Originator* parameter value, and
   2) the APDU as the D-ABORT *User Data* parameter value, and
d) if the CM-user is an active user, invoke CM-provider-abort service indication with the abstract value "invalid-PDU" as~~APDU as~~ the CM-provider-abort *Reason* parameter value.

5.4.3.2 If the *User Data* parameter of a D-START confirmation does not contain a valid PDU, the CM-ASE shall:

a) if the D-START *Result* parameter is set to the abstract value "accepted", then
   1) if the CM-ASE is a CM-air-ASE, create a CMAircraftMessage APDU with a cmAbortReason [invalid-PDU] APDU message element,
   2) if the CM-ASE is a CM-ground-ASE, create a CMGroundMessage APDU with a cmAbortReason [invalid-PDU] APDU message element,
   3) invoke D-ABORT request with:
      i) the abstract value "provider" as the D-ABORT *Originator* parameter value, and
      ii) the APDU as the D-ABORT *User Data* parameter value, and

~~if the D-START *Result* parameter is set to the abstract value "accepted", invoke D-ABORT request with:~~
~~the abstract value "provider" as the D-ABORT *Originator* parameter value, and~~
~~the APDU as the D-ABORT *User Data* parameter value, and~~

b) if the CM-user is an active user, invoke CM-provider-abort service indication with the abstract value "invalid-PDU" ~~APDU~~ as the CM-provider-abort *Reason* parameter value.

5.4.4 Not Permitted PDU

5.4.4.1 If the *User Data* parameter of a D-START indication or D-DATA indication is a valid PDU, but is not a permitted PDU as defined within section 5.3, the CM-ASE shall:

a) if the CM-ASE is a CM-air-ASE, create a CMAircraftMessage APDU with a cmAbortReason [not-permitted-PDU] APDU message element,

b) if the CM-ASE is a CM-ground-ASE, create a CMGroundMessage APDU with a cmAbortReason [not-permitted-PDU] APDU message element,
c) invoke D-ABORT request with:
   1) the abstract value "provider" as the D-ABORT *Originator* parameter value, and
   2) the APDU as the D-ABORT *User Data* parameter value, and
d) if the CM-user is an active user, invoke CM-provider-abort service indication with the abstract value "not-permitted-PDU" APDU as the CM-provider-abort *Reason* parameter value.

5.4.4.2 If the *User Data* parameter of a D-START confirmation is a valid PDU, but is not a permitted PDU as defined within section 5.3, the CM-ASE shall:

a) if the D-START *Result* parameter is set to the abstract value "accepted",
   1) if the CM-ASE is a CM-air-ASE, create a CMAircraftMessage APDU with a cmAbortReason [not-permitted-PDU] APDU message element,
   2) if the CM-ASE is a CM-ground-ASE, create a CMGroundMessage APDU with a cmAbortReason [not-permitted-PDU] APDU message element,
   3) invoke D-ABORT request with:
      i) the abstract value "provider" as the D-ABORT *Originator* parameter value, and
      ii) the APDU as the D-ABORT *User Data* parameter value, and

if the D-START *Result* parameter is set to the abstract value "accepted", invoke D-ABORT request with:
the abstract value "provider" as the D-ABORT *Originator* parameter value, and
the APDU as the D-ABORT *User Data* parameter value, and

b) if the CM-user is an active user, invoke CM-provider-abort service indication with the abstract value "not-permitted-PDU" APDU as the CM-provider-abort *Reason* parameter value.

5.4.5 D-START Confirmation *Result* or *Reject Source* Parameter Values Not as Expected

5.4.5.1 If the CM-ground-ASE receives a D-START confirmation with the D-START *Result* parameter having the abstract value of "accepted", the CM-ground-ASE shall:

a) create a CMGroundMessage APDU with a cmAbortReason [dialogue-acceptance-not-permitted] APDU message element,
b) invoke D-ABORT request with:
   1) the abstract value "provider" as the D-ABORT *Originator* parameter value, and
   2) the APDU as the D-ABORT *User Data* parameter value, and
c) if the CM-ground-user is an active user, invoke CM-provider-abort service indication with the abstract value "dialogue-acceptance-not-permitted" asAPDU as the CM-provider-abort *Reason* parameter value.

5.4.5.2 If the CM-ASE receives a D-START confirmation with the D-START *Result* parameter having the abstract value of "rejected (transient)" or if the D-START *Reject Source* parameter has the abstract value of "DS provider", and if the CM-user is an active user, the CM-ASE shall invoke CM-provider-abort service indication with the abstract value "communication-service-error" APDU as the CM-provider-abort *Reason* parameter value.:

if the CM-ASE is a CM-air-ASE, create a CMAircraftMessage APDU with a cmAbortReason [communication-service-error] APDU message element,
if the CM-ASE is a CM-ground-ASE, create a CMGroundMessage APDU with a cmAbortReason [communication-service-error] APDU message element, and

5.4.6 D-END Confirmation Not as Expected

5.4.6.1 If the CM-ground-ASE receives a D-END confirmation with the D-END *Result* parameter that does not have the abstract value of "accepted", the CM-ground-ASE shall:

a) create a CMGroundMessage APDU with a cmAbortReason [dialogue-end-not-accepted] APDU message element,
b) invoke D-ABORT request with:
   1) the abstract value "provider" as the D-ABORT *Originator* parameter value, and

2) the APDU as the D-ABORT *User Data* parameter value, and

c) if the CM-ground-user is an active user, invoke CM-provider-abort service indication with the abstract value "dialogue-end-not-accepted" ~~APDU~~ as the CM-provider-abort *Reason* parameter value.

5.5 CM ASE State Tables

5.5.1 Introduction

Note 1.— This section defines the state tables for the CM-air-ASE and the CM-ground-ASE.

Note 2.— If the state tables shown in this section conflict with the textual statements made elsewhere in this SARPs, the textual statements take precedence.

5.5.2 CM-Ground-ASE State Table

Note.— Table 5-2 reflects the possible states of the CM-Ground-ASE.

STATE ⇒ EVENT ⇓	IDLE	LOGON	UPDATE	CONTACT	DIALOGUE	CONTACT DIALOGUE	END	FORWARD
DIALOGUE Service Events								
D-START Indication Version Number is greater than the CM-ground-ASE version number	• D-START response ⇒*IDLE*	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	• D-START response ⇒*IDLE*
D-START Indication Version Number is less than or equal to the CM-ground-ASE version number, User Data = CMLogonRequest	• CM-logon indication ⇒*LOGON*	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur
D-START Indication Version Number is less than or equal to the CM-ground-ASE version number, User Data = CMForwardRequest	• CM-forward indication • D-START response ⇒*IDLE*	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur
D-START Confirmation *Result* "rejected (permanent)" and *Reject Source* "DS user"	cannot occur	cannot occur	• Stop timer t_{update} ⇒*IDLE*	• Stop timer $t_{contact}$ • CM-contact confirmation ⇒*IDLE*	cannot occur	cannot occur	cannot occur	• Stop timer $t_{forward}$ • ~~If sending ASE version > receiving ASE then~~ CM-forward confirmation ⇒*IDLE*
D-DATA Indication	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	• CM contact confirmation ⇒*DIALOGUE*	cannot occur	cannot occur
D-END Confirmation *Result* "accepted"	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	• stop timer t_{end} ⇒*IDLE*	cannot occur
CM-User Events								
CM-update Request	• D-START request • Start timer t_{update} ⇒*UPDATE*	not permitted	not permitted	not permitted	• D-DATA request ⇒*DIALOGUE*	not permitted	not permitted	not permitted
CM-contact Request	• D-START request • Start timer $t_{contact}$ ⇒*CONTACT*	not permitted	not permitted	not permitted	• D-DATA request ⇒*CONTACT DIALOGUE*	not permitted	not permitted	not permitted

CM-forward Request	• D-START request • Start timer $t_{forward}$ ⇒*FORWARD*	not permitted	not permitted	not permitted	not permitted	not permitted	not permitted	not permitted
CM-logon Response *Maintain Dialogue* not supplied by CM-ground user	not permitted	• D-START response ⇒*IDLE*	not permitted	not permitted	not permitted	not permitted	not permitted	not permitted
CM-logon Response *Maintain Dialogue* "accepted"	not permitted	• D-START response ⇒*DIALOGUE*	not permitted	not permitted	not permitted	not permitted	not permitted	not permitted
CM-end Request	not permitted	not permitted	not permitted	not permitted	• D-END request • Start timer t_{end} ⇒*END*	not permitted	not permitted	not permitted
ABORT Events								
CM-user-abort Request	not permitted	• D-ABORT request ⇒*IDLE*	• stop timer t_{update} • D-ABORT request ⇒*IDLE*	• stop timer $t_{contact}$ • D-ABORT request ⇒*IDLE*	• D-ABORT request ⇒*IDLE*	• D-ABORT request ⇒*IDLE*	• stop timer t_{end} • D-ABORT request ⇒*IDLE*	• stop timer $t_{forward}$ • D-ABORT request ⇒*IDLE*
D-ABORT Indication *Originator* is "provider"	cannot occur	• CM-provider-abort indication ⇒*IDLE*	• stop timer t_{update} • CM-provider-abort indication ⇒*IDLE*	• stop timer $t_{contact}$ • CM-provider-abort indication ⇒*IDLE*	• CM-provider-abort indication ⇒*IDLE*	• CM-provider-abort indication ⇒*IDLE*	• stop timer t_{end} • CM-provider-ABORT indication ⇒*IDLE*	• stop timer $t_{forward}$ • CM-provider-ABORT indication ⇒*IDLE*
D-ABORT Indication *Originator* is "user"	cannot occur	• CM-user-abort indication ⇒*IDLE*	• stop timer t_{update} • CM-user-abort indication ⇒*IDLE*	• stop timer $t_{contact}$ • CM-user-abort indication ⇒*IDLE*	• CM-user-abort indication ⇒*IDLE*	• CM-user-abort indication ⇒*IDLE*	• stop timer t_{end} • CM-user-ABORT indication ⇒*IDLE*	• stop timer $t_{forward}$ • CM-user-ABORT indication ⇒*IDLE*
D-P-ABORT indication	cannot occur	• CM-provider-abort indication ⇒*IDLE*	• stop timer t_{update} • CM-provider-abort indication ⇒*IDLE*	• stop timer $t_{contact}$ • CM-provider-abort indication ⇒*IDLE*	• CM-provider-abort indication ⇒*IDLE*	• CM-provider-abort indication ⇒*IDLE*	• stop timer t_{end} • CM-provider-abort indication ⇒*IDLE*	• stop timer $t_{forward}$ • CM-provider-abort indication ⇒*IDLE*
T_{update} Expires	cannot occur	cannot occur	• D-ABORT request • CM-provider-abort indication ⇒*IDLE*	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur

$T_{contact}$ Expires	cannot occur	cannot occur	cannot occur	• D-ABORT request • CM-provider-abort indication ⇒*IDLE*	cannot occur	cannot occur	cannot occur	cannot occur
T_{end} Expires	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	• D-ABORT request • CM-provider-abort indication ⇒*IDLE*	cannot occur
$T_{forward}$ Expires	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	cannot occur	• D-ABORT request • CM-provider-abort indication ⇒*IDLE*

Table 5-2: CM-Ground-ASE

5.5.3 CM-Air-ASE State Table

Note.— Table 5-3 reflects the possible states of the CM-Air-ASE.

STATE ⇒ EVENT ⇓	IDLE	LOGON	CONTACT	DIALOGUE	CONTACT DIALOGUE
DIALOGUE Service Events					
D-START Indication *User Data* CMUpdate	• CM-update indication • D-START response ⇒*IDLE*	cannot occur	cannot occur	cannot occur	cannot occur
D-START Indication *User Data* CMContactRequest	• CM-contact indication ⇒*CONTACT*	cannot occur	cannot occur	cannot occur	cannot occur
D-START Confirmation *Result* "rejected (permanent)" and *Reject Source* "DS user"	cannot occur	• Stop timer t_{logon} • CM-logon confirmation ⇒*IDLE*	cannot occur	cannot occur	cannot occur
D-START Confirmation *Result* "accepted"	cannot occur	• Stop timer t_{logon} • CM-logon confirmation ⇒*DIALOGUE*	cannot occur	cannot occur	cannot occur
D-DATA Indication *User Data* CMUpdate	cannot occur	cannot occur	cannot occur	• CM-update indication ⇒*DIALOGUE*	cannot occur
D-DATA Indication *User Data* CMContactRequest	cannot occur	cannot occur	cannot occur	• CM-contact indication ⇒*CONTACT DIALOGUE*	cannot occur
D-END Indication	cannot occur	cannot occur	cannot occur	• CM-end indication • D-END response ⇒*IDLE*	cannot occur
CM-User Events					
CM-contact Response	not permitted	not permitted	• D-START response ⇒*IDLE*	not permitted	• D-DATA request ⇒*DIALOGUE*
CM-logon Request	• D-START request • Start timer t_{logon} ⇒*LOGON*	not permitted	not permitted	not permitted	not permitted
ABORT Events					
CM-user-abort Request	not permitted	• stop timer t_{logon} • D-ABORT request ⇒*IDLE*	• D-ABORT request ⇒*IDLE*	• D-ABORT request ⇒*IDLE*	• D-ABORT request ⇒*IDLE*
D- ABORT Indication *Originator* is "provider"	cannot occur	• stop timer t_{logon} • CM-provider-abort indication ⇒*IDLE*	• CM-provider-abort indication ⇒*IDLE*	• CM-provider-abort indication ⇒*IDLE*	• CM-provider-abort indication ⇒*IDLE*
D-ABORT Indication *Originator* is "user"	cannot occur	• stop timer t_{logon} • CM-user-abort indication ⇒*IDLE*	• CM-user-abort indication ⇒*IDLE*	• CM-user-abort indication ⇒*IDLE*	• CM-user-abort indication ⇒*IDLE*
D-P-ABORT indication	cannot occur	• stop timer t_{logon} • CM-provider-abort indication ⇒*IDLE*	• CM-provider-abort indication ⇒*IDLE*	• CM-provider-abort indication ⇒*IDLE*	• CM-provider-abort indication ⇒*IDLE*
T_{logon} Expires	cannot occur	• D-ABORT request • CM-provider-abort indication ⇒*IDLE*	cannot occur	cannot occur	cannot occur

Table 5-3: CM-Air-ASE

6. COMMUNICATION REQUIREMENTS

6.1 Encoding Rules

6.1.1 The CM application shall use PER encoding as defined in reference [2], using the Basic Unaligned variant to encode/decode the ASN.1 message structure and content specified in Chapter 4 of this document, or a functionally equivalent means which provides the same result.

6.2 Dialogue Service Requirements

6.2.1 Primitive Requirements

6.2.1.1 Where dialogue service primitives, that is D-START, D-DATA, D-END, D-ABORT, and D-P-ABORT are described as being invoked in Chapter 5 of this document, the CM-ground-ASE and the CM-air-ASE shall exhibit external behavior consistent with the dialogue service, as described in reference [4], having been implemented and its primitives invoked.

6.2.2 ATN Quality-of-Service Requirements

6.2.2.1 The application service priority for CM shall be the abstract value of “flight regularity communications”.

6.2.2.2 The RER Quality of Service Parameter of the D-START shall be set to the abstract value of “low”.

Note 1.— The application service residual error rate (RER) for CM is required to be 10^{-6} or better.

Note 2.— The RER takes into account non-delivery for messages over a given dialogue.

~~Routing Policy~~

6.2.2.3 The CM-ASE shall map the CM-start service Class of Communication abstract values to the ATSC routing class abstract value part of the D-START QOS parameter as presented in Table 6-1.

Class of Communication Abstract Value	Routing Class Abstract Value
A	Traffic only follows Class A ATSC route(s)
B	Traffic only follows Class B ATSC route(s)
C	Traffic only follows Class C ATSC route(s)
D	Traffic only follows Class D ATSC route(s)
E	Traffic only follows Class E ATSC route(s)
F	Traffic only follows Class F ATSC route(s)
G	Traffic only follows Class G ATSC route(s)
H	Traffic only follows Class H ATSC route(s)
I	Traffic only follows Class I ATSC route(s)
J	Traffic only follows Class J ATSC route(s)

Table 6-1: Mapping Between Class of Communication and Routing Class Abstract Values

Note.— ATSC values are defined in [5].

6.3 CM AE Control Functions Requirements

The title of the CM application used as the AE-Qualifier for the CM application shall be the IA5 character string “CMA”.

7. CM USER REQUIREMENTS

Note.— Requirements imposed on CM-users concerning CM messsages and interfacing with the CM-ASEs are presented in this chapter.

7.1 CM-Air-User Requirements

7.1.1 General CM-Air-User Requirements

7.1.1.1 When a CM-air-user invokes the CM-logon service and requires a particular class of communication service, it shall set the class of communication service parameter to be the class of communication it requires.

Note.— When the CM-air-user invokes a CM-logon service and has no preference for the class of communication service to be used, the class of communication service parameter does not need to be provided.

7.1.2 The ASE Invocation/Aircraft Identifier Correlation

Note 1.— For each CM-air-ASE invocation, the CM-air-user ~~shall~~ establishes a correlation between a CM-air-ASE invocation and the ICAO facility designator.

Note 2.— Upon the initiation of a CM-logon service request, or upon receipt of a CM-update service indication or a CM-contact service indication, the ASE invocation correlation is~~shall be~~ based on the ICAO facility designator in the ICAO Facility Designator *primitive of the respective CM service.*

Note 3.— The correlation is~~shall be~~ maintained for the duration of the ASE invocation.

7.1.3 CM-logon Service Requirements

Note.— Only the CM-air-user is ~~shall be~~ permitted to initiate the CM-logon service.

7.1.3.1 When invoking the CM-logon service request the CM-air-user shall provide the following as part of the CMLogonRequest:

a) its CM ~~name,~~ long TSAP, ~~and version number,~~
b) information on each application for which it requires a data link service as follows:
   1) for air-only initiated services: application name and version number for all the versions that can be supported, and
   2) for applications that can be ground initiated: application name, version number, and address for all the versions that can be supported, and
c) flight information data as required by the ground system.

7.1.3.2 When invoking the CM-logon service request, if any RDP for a given application address is different than the CM RDP, the CM-air-user shall use the long TSAP for each application address provided.

~~When invoking the CM-logon service request the CM-air-user shall provide all of the versions and addresses, as required for ground-initiated applications, for each application the aircraft is capable of supporting.~~

7.1.3.3 Upon receipt of a CM-logon service confirmation, the CM-air-user shall make the information contained in the CMLogonResponse available to the other ~~CM-air~~ applications (i.e., ADS, CPDLC, and FIS), as well as to the dialogue service provider.

7.1.3.4 Upon the receipt of a CMLogonResponse from a CM-logon service confirmation from an ICAO ground facility designator for which CM information has previously been received, the CM-air-user shall only replace the previous information for which new logon information has been received.

7.1.4 CM-update Service Requirements

7.1.4.1 Upon the receipt of *Update Information* from a CM-update service indication from an ICAO ground facility designator for which CM information has previously been received, the CM-air-user shall only replace the previous information for which updated information has been received.

7.1.4.2 The CM-air-user shall make the updated information contained in the *Update Information* available to the other ~~CM-air-~~applications (i.e., ADS, CPDLC, and FIS), as well as to the dialogue service provider.

7.1.5 CM-contact Service Requirements

7.1.6 Upon receipt of a CM-contact indication, the CM-air-user shall invoke a CM-logon request with the indicated ground system within 0.5 seconds.

7.1.6.1 Upon receipt of a CM-logon confirmation when performing the CM-contact service, the CM-air-user shall invoke a CM-contact response within 0.5 seconds.

7.1.6.2 Upon receipt of a CM-contact service indication, the CM-air-user shall attempt to initiate a CM-logon service request with the indicated ICAO facility designator.

7.1.6.3 If a CM-logon service request is initiated, the CM-air-user shall comply with the CM-logon requirements as stated in the CM-logon service requirements section.

7.1.6.4 In addition to the above CM-logon service requirements, upon receipt of a CM-logon service response from the indicated ICAO facility designator, or if no CM-logon service request can be initiated, the CM-air-user shall invoke the CM-contact service response indicating the success or lack thereof of the CM-logon service request.

7.2 CM-Ground-User Requirements

7.2.1 General CM-Ground-User Requirements

7.2.1.1 A CM-ground-user shall invoke the CM-logon service, CM-update service, CM-contact service, and CM-end service only when communicating with a CM-air-user.

7.2.1.2 A CM-ground-user shall invoke the CM-forward service only when communicating with another CM-ground-user.

7.2.1.3 When a CM-ground-user invokes the CM-update service, CM-contact service, or CM-forward service and requires a particular class of communication service, it shall set the class of communication service parameter to be the class of communication it requires.

Note 1.— When the CM-ground-user invokes a CM-update service, CM-contact service, or CM-forward service and has no preference for the class of communication service to be used, the class of communication service parameter does not need to be provided.

Note 2.— When a CM-ground-user specifies the class of communication service parameter and the dialogue is in place, the class of communication parameter is ignored

7.2.2 The ASE Invocation/Aircraft Identifier Correlation

Note 1.— For each CM-ground-ASE invocation, the CM-ground-user ~~shall~~ establish~~es~~ a correlation between a CM-ground-ASE invocation and the aircraft 24 bit address

.

Note 2.— Upon the initiation a CM-update service request or CM-contact service request, or upon receipt of a CM-logon service indication the ASE invocation correlation is~~shall be~~ based on the 24-bit aircraft identification in the Aircraft Identifier *primitive of the respective CM service.*

Note 3.— The correlation is~~shall be~~ maintained for the duration of the ASE invocation.

7.2.3 CM-logon Service Requirements

7.2.3.1 Upon receipt of a CM-logon indication, the CM-ground-user shall invoke a CM-logon response within 0.5 seconds.

7.2.3.2 Upon receipt of a CM-logon service indication, the CM-ground-user shall make the aircraft application information contained in the *Logon Request* available to the other ~~CM-ground~~ applications (i.e., ADS, CPDLC, and FIS), as well as to the dialogue service provider.

7.2.3.3 Upon receipt of a CM-logon service indication, the CM-ground-user shall create the actual TSAP for each aircraft application information contained in the *Logon Request* based on the IDP and long TSAP for each application as defined in Chapter 4 of this document.

Note.— The actual TSAP is composed of the IDP and the long TSAP as in Chapter 4. The long TSAP = RDP + short TSAP. The short TSAP = ARS + LOC + SYS + SEL.

7.2.3.4 Upon the receipt of a *Logon Request* from a CM-logon service indication from a aircraft identifier for which CM information has previously been received and still being maintained, the CM-ground-user shall update the aircraft information accordingly.

7.2.3.5 Upon receipt of a CM-logon service indication, the CM-ground-user shall invoke a CM-logon response service with a CMLogonResponse containing:

~~an indication of whether or not the CM logon was successful,~~

a) application names, addresses, and version numbers for the requested applications that can be air-initiated for all versions that the ground and aircraft systems can support, and

b) application names and version numbers for the requested ground-only initiated applications that the ground system can support.

7.2.3.6 When invoking the CM-logon service response, if any RDP for a given application address is different than the CM RDP, the CM-ground-user shall use the long TSAP for each application address provided.

~~When invoking the CM-logon service response, the CM-ground-user shall provide all of the versions for each application the ground is capable of supporting.~~

~~When a CM-ground-user cannot support any of the air or ground applications requested the CM-ground-user shall indicate that the logon was unsuccesful.~~

7.2.3.7 If the CM-ground-user requires a CM dialogue to be maintained, the CM-ground-user shall set the CM-logon service response *Maintain Dialogue* parameter.

7.2.4 CM-update Service Requirements

7.2.4.1 Only the CM-ground-user shall be permitted to initiate the CM-update-service.

7.2.4.2 When invoking the CM-update service request, the CM-ground-user shall provide a CMUpdate containing application names, addresses, and version numbers for each of the data link applications being updated.

7.2.5 CM-contact Service Requirements

Note.— Only the CM-ground-user is~~shall be~~ permitted to initiate the CM-contact-service.

~~If a CM-dialogue exists when the CM-ground-user invokes the CM-contact service request, the CM-ground user shall be prohibited from requesting any other CM service primitive, except the CM-user-abort service request primitive, until it receives the CM-contact service confirmation.~~

7.2.5.1 When invoking the CM-contact service request, the CM-ground-user shall provide a CMContactRequest containing the ICAO facility designator of the ground facility that the ground requests the aircraft to contact.

7.2.6 CM-end Service Requirements

Note 1.— Only the CM-ground-user is ~~shall be~~ permitted to initiate the CM-end-service.

Note 2.— If the CM-ground-user establishes a CM dialogue with the CM-logon Maintain Dialogue *parameter set, the CM-ground user is~~shall be~~ responsible for closing the CM dialogue with the CM-end service.*

7.2.7 CM-forward Service Requirements

Note .— Only the CM-ground-user is ~~shall be~~ permitted to initiate the CM-forward-service.

7.2.7.1 When requesting the CM-forward service, the CM-ground-user shall provide all of the information from either a CM-logon request message or a CM-forward request message, whichever is the more recent.

7.2.7.2 Upon receipt of a CM-forward service indication, the CM-ground-user shall make the aircraft application information contained in the *Forward Request* available to the other ~~CM-ground~~ applications (i.e., ADS, CPDLC, and FIS), as well as to the dialogue service provider.

7.2.7.3 Upon receipt of a CM-forward service indication, the CM-ground-user shall create the actual TSAP for each aircraft application information contained in the *Forward Request* based on the IDP and long TSAP for each application as defined in Chapter 4 of this document.

Note.— The actual TSAP is composed of the IDP and the long TSAP as in Chapter 4. The long TSAP = RDP + short TSAP. The short TSAP = ARS + LOC + SYS + SEL.

7.2.7.4 Upon the receipt of a *Forward Request* from a CM-forward service indication concerning an aircraft identifier for which CM information has previously been received and is still being maintained, the CM-ground-user shall update the aircraft information accordingly.

7.2.7.5 Upon the receipt of a *Forward Request* from a CM-forward service indication ~~containing air-only initiated applications that are supported by the receiving CM-ground-user~~, the receiving CM-ground-user shall invoke a CM-update service request with the indicated aircraft~~containing the requested application addresses~~.