ATTACHMENT 3 ACARS TO IPS DIALOGUE SERVICE CONVERGENCE FUNCTION

Commented [OML1]: [P1-M22-02 – M.Olive] – Update AICF text/diagrams as necessary to reflect changes to accommodate port simplification. (Interdependency with ICAO WG-I.)

3.0 INTRODUCTION

This attachment specifies the ACARS to IPS Dialogue Service Convergence Function (AICF), including its interfaces and functional elements. The AICF adapts ACARS applications to the IPS Dialogue Service (IPS DS), which provides a mechanism for exchanging application messages over the IPS communications infrastructure.

3.1 AICF Overview

Figure 3-1 illustrates the ACARS message flow over the IPS dialogue service via the AICF, as well as the placement of the AICF within the upper layers between the ACARS application or peripheral and the IPS DS. The DTLS and UDP layers are shown for completeness.

COMMENTARY

In this attachment, any detail regarding the IPS DS, DTLS, and UDP layers is provided for illustrative purposes only. Normative information is available in the respective standards for those layers.

The ACARS application, or a peripheral (e.g., FMS), represents existing aircraft applications or systems that exchange messages with ground systems using the ACARS protocol stack. The application messages and protocols are specified in existing standards such as ARINC 620, ARINC 622, ARINC 623, etc. The ACARS application messages are accommodated by the AICF without any changes to the existing ACARS applications, systems, or specifications.

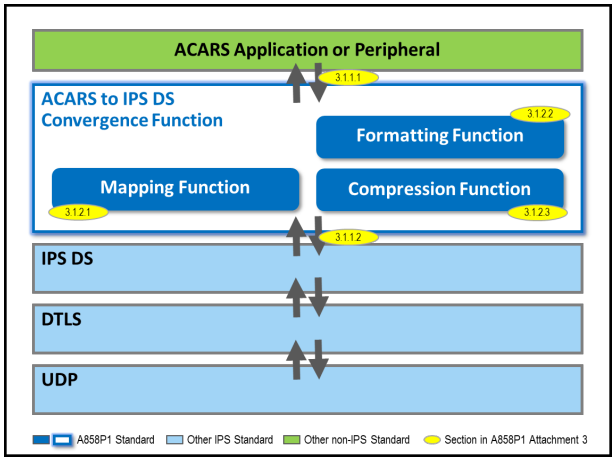

Figure 3-1 – AICF Placement within the Upper Layers

Each AICF function and interface is introduced in the following sub-sections, which are referenced in the figure by yellow ovals. Detailed specifications of downlink and uplink message processing are described in Sections 3.2 and 3.3, respectively, in this attachment.

3.1.1 AICF Interfaces

3.1.1.1 ACARS Message Interface

The detailed interface between the ACARS application/peripheral and the AICF is local and implementation-dependent. At a minimum, the interface must support a transport mechanism for exchanging the following ACARS message fields:

- ACARS message Label consisting of two characters.
- Optional ACARS Sub-label consisting of two characters preceded by a "#" character. The Sub-label is present when the ACARS application is hosted in a peripheral, e.g., ARINC 622 hosted in an FMS, and it uniquely identifies the peripheral.
- Optional Supplementary Address field which begins with the "/" character and terminates with a "." character and which may contain:
  - Optional Message Function Identifier (MFI) consisting of two characters immediately following the "/" character and followed by a space character. This field, which identifies flow types from an ACARS peripheral, is mandatory for ARINC 622 messages, and may be present for other ACARS applications as well.
  - One or more supplementary addresses containing three, four, or seven alpha-numeric characters, each of which is separated by a space character. This field is mandatory for ARINC 622 messages and contains either a four-character ATC Facility designator or a seven-character ATC Facility address.
- Application Text field, which contains the bit-oriented (e.g., ARINC 622) or character-oriented (e.g., ARINC 623) data generated/consumed by the ACARS application. If the application text includes a Cyclic Redundancy Check (CRC) to ensure end-to-end integrity, the CRC is preserved during the AICF processing.
- Flight Identifier (FlightID), which is six alpha-numeric characters consisting of a two-character airline identifier and a four-character flight number, and the Message Sequence Number (MSN), which consists of four alpha-numeric characters. Note that the FlightID and MSN are conveyed via this interface only for downlink messages; for uplink messages, the MSN is not included and the FlightID terminates in the AICF and is not presented to the ACARS application or peripheral.

3.1.1.2 IPS Dialogue Service Interface

The interface between the AICF and the IPS DS is the Dialogue Service (DS) interface per ICAO Doc. 9880, Part III. As specified in ICAO Doc. 9896, the IPS Dialogue Service appears as an instance of the dialogue service; therefore, reusing the same interface for the AICF facilitates commonality between B1/B2 application adaptation and ACARS application adaptation. The detailed implementation of the service interface is local and implementation-dependent.

For ACARS application adaptation, the AICF uses the following dialogue service primitives, which represent a subset of primitives supported by the IPS DS:

- D-START – a confirmed service used to establish the binding between communicating peer IPS DS entities

- D-DATA – an unconfirmed service used to exchange messages between peer IPS DS entities
- D-ABORT – an unconfirmed service used to terminate the binding between communicating peer IPS DS entities

In addition to the DS primitives and associated parameters, this interface also conveys control parameters used to identify the ACARS application type as well as the specific ACARS application.

3.1.2 AICF Functions

The AICF consists of three primary functions that operate on ACARS application messages: a Mapping Function, a Formatting Function, and a Compression Function.

3.1.2.1 Mapping Function

The AICF Mapping Function provides the mapping between ACARS application messages and IPS DS primitives and associated parameters. It also maintains the status of the dialogue (i.e., “open” or “closed”) for each application and end entity for which a binding is established using the D-START primitive; the specific mechanism for maintaining the dialogue status is local and implementation-dependent.

Section 3.4 in this attachment specifies the application-specific criteria for mapping to a dialogue service primitive and for setting the dialogue status.

3.1.2.2 Formatting Function

The AICF Formatting Function receives ACARS downlink messages via the ACARS Message Interface and assembles the relevant ACARS application message fields into Uncompressed Application Data, which serves as the input to the Compression Function. Conversely, in the uplink direction, this function parses the Uncompressed Application Data, which is the result of de-compression, into the ACARS application message fields.

Figure 3-2 illustrates the Uncompressed Application Data format, which is the concatenation of the ACARS message label, sub-label (if present), supplementary address field (if present), and the application text.

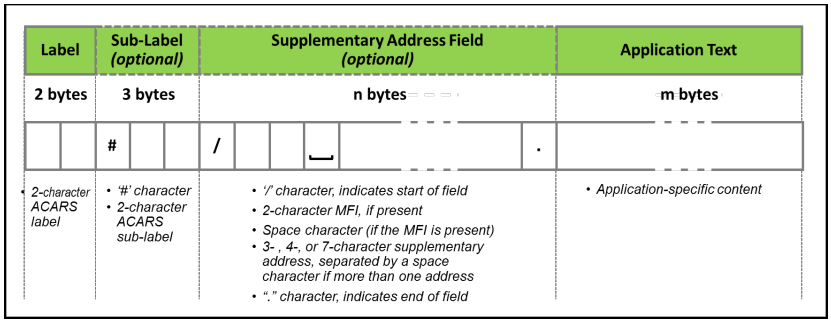

Figure 3-2 – Uncompressed Application Data Format

Note that this function assembles and parses the ACARS application message fields, but it does not in any way change or manipulate the content of the fields

themselves. In addition, since ACARS application messages exchanged using IPS are not encapsulated in the ARINC 618 air-ground protocol, downlink and uplink ACARS messages are not segmented into ARINC 618 ACARS blocks.

3.1.2.3 Compression Function

The AICF includes a Compression Function that applies a data compression algorithm to reduce the size of ACARS application messages exchanged over IPS. The input to the Compression Function (or output resulting from de-compression) is the Uncompressed Application Data. As shown in Figure 3-3, the output of the Compression Function (or input to de-compression) is a 1-byte Compression Parameter concatenated with the Compressed ACARS Message.

COMMENTARY

Some messages (e.g., small or encoded messages) may increase in size when compressed. The compression parameter allows the sending entity to determine compressibility and indicate the most efficient method of conveying the data, which may be with no compression.

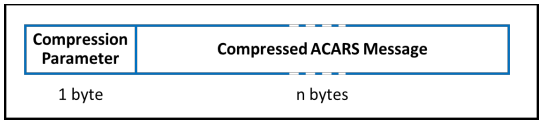

Figure 3-3 – Compressed Application Data Format

The format and field values of the Compression Parameter are shown in Table 3-1.

Table 3-1 – Compression Parameter Format and Field Values

Compression Parameter – 8 bits								
Reserved Field					Compression Algorithm Field			
(MSB) 8	7	6	5		4	3	2	(LSB) 1
0x0 thru 0xF	Reserved (future)				0x0	No compression		
					0x1	DEFLATE compression		
					0x2 thru 0xF	Reserved (future)		
By default, this field is set to 0x0								

In addition to “no compression,” Airborne IPS Systems, IPS Gateways, and Ground IPS Hosts that handle ACARS application messages shall support the DEFLATE algorithm, as a minimum. Reserved values in the compression parameter support the addition of other compression algorithms in the future.

3.1.3 IPS Dialogue Service Accommodation of the AICF

To facilitate commonality between B1/B2 application adaptation and ACARS application adaptation, the IPS Dialogue Service (IPS DS) per ICAO Doc. 9896 is used to convey ACARS application messages using the ATNPKT format; refer to Appendix A in this document for ATNPKT examples. The ATNPKT consists of a fixed part, which is always present, and a variable part, which contains optional fields depending on the dialogue service primitive and application data. The inclusion of optional fields in ATNPKT complies with the IPS DS mapping specified in ~~Table II-1-6 in~~ ICAO Doc. 9896 Part III, with the following exceptions:

Commented [OML2]: **[P1-M21-02 – M.Olive]** – Revisit relationship to mandatory/optional parameters in Doc, 9896 Table II-1-6 and clarify any exceptions for AICF.

Commented [OML3R2]: **Ed. Note** – Greg recommended pointing to the section but not to the specific table since that reference may change over time.

- The Called Peer ID, Calling Peer ID, and User Data fields are mandatory in any ATNPKT containing a D-START, D-STARTCNF, D-DATA, or D-ABORT primitive.
- The optional Content Version field, which specifies an ASN.1 syntax version associated with B1/B2 application messages, is not used for ACARS-based messages.
- The optional QoS parameter is not used for ACARS-based messages. The network layer utilizes the port number associated with specific ACARS ATS and AOC applications, as described in Section 3.2.2.1 and ~~Table 3-3~~Table 3-2, to assign message priority. This information is then used to set the differentiated service field in IP packets.

3.2 AICF Downlink Message Processing

Figure 3-4 illustrates the processing of downlink messages from an ACARS application or peripheral.

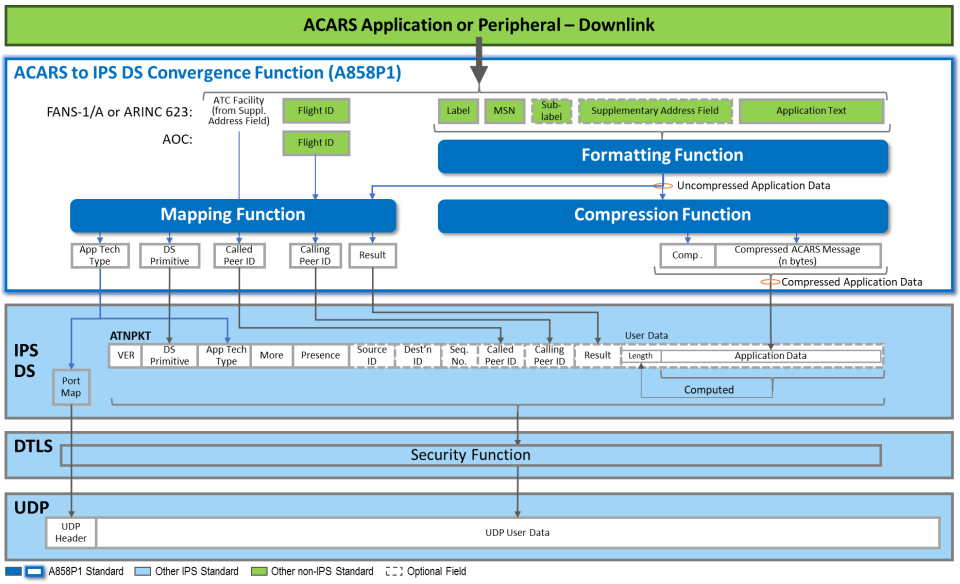

Figure 3-4 – AICF Downlink Processing

For downlink messages, the AICF Mapping Function uses information contained in the Uncompressed Application Data to determine values for the Application Technology Type, Dialogue Service primitive, Called Peer, Calling Peer, and Result parameters.

Other fixed and optional ATNPKT fields (e.g., Source ID and Destination ID) shown in the IPS DS block in Figure 3-4 are generated by the IPS DS for downlink messages, and they are not provided by the AICF via the IPS DS interface.

3.2.1 User Data

As specified in Section 3.1.2.2, the AICF Formatting Function assembles the ACARS application message fields to create Uncompressed Application Data, which serves as an input to both the AICF Mapping Function and Compression Function.

Compressed Application Data, which is the output of the AICF Compression Function specified in Section 3.1.2.3, is the User Data that is conveyed to the IPS DS.

3.2.2 Application Technology Type~~Control Parameters~~

~~3.2.2.1 Application Technology Type~~

The Application Technology Type field is an 8-bit value that indicates the type of application information carried in the IPS DS messages. The AICF communicates the Application Technology Type parameter to the IPS DS via a local, implementation-dependent interface.

For ACARS-based applications, ~~this parameter~~the three most significant bits of the 8-bit value indicate one of two general types: “ACARS ATS / IPS DS” for ARINC 622 (FANS-1/A) or ARINC 623 (character-based ATS) messages; and “ACARS AOC / IPS DS” for all other non-ATS ACARS messages. As shown in Table 3-2, these two types map directly to associated ports for ats-acars (5911) and aoc-acars (5913), respectively, which are registered with the Internet Assigned Numbers Authority (IANA) and specified in ICAO Doc. 9896. The least significant five bits of the 8-bit value identity the specific ACARS application. These bits are assigned by the AICF Mapping Function ~~assigns the Application Technology Type~~ based on the value of the ACARS message Label or the MFI (in the case of an ACARS peripheral) contained in the Uncompressed Application Data, as follows:

ACARS MU/CMU-hosted Application:

`Label/Address(es).Application_text`

ACARS Peripheral-hosted Application:

`Label#Sub-label/MFI<sp>Address(es).Application_text`

Table 3-2 – Application Technology Type

~~Value of ACARS MFI or Label~~	~~Application Technology Type~~	~~Application Technology Type Field Value (per ICAO Doc. 9896, Part II, Section 2.1)~~
~~Ax, Bx (i.e., the first character is ‘A’ or ‘B’)~~	~~“ACARS ATS / IPS DS”~~	~~b011~~
~~All other MFI / labels (i.e., the first character is not ‘A’ or ‘B’)~~	~~“ACARS AOC / IPS DS”~~	~~b101~~

Value of ACARS MFI or Label	ACARS Application	Application Technology Type [Note 1]		Associated Port Number and Service Name (per ICAO Doc. 9896, Part III)
		Binary (8 bits)	Hex	

A0 (uplink) B0 (downlink)	ARINC 622 AFN	101 "ACARS ATS/ IPS DS"	00000	0xA0	5911 ats-acars
AA (uplink) BA (downlink)	ARINC 622 CPDLC		00001	0xA1	
A6 (uplink) B6 (downlink)	ARINC 622 ADS-C		00010	0xA2	
AW (uplink) BW (downlink)	ARINC 622 ATS WIND		00011	0xA3	
Ax (uplink) Bx (downlink) (i.e., MFI / Labels starting with 'A' or 'B', except: A0, B0, A6, B6, AA, BA, AW, BW)	ARINC 623		00100	0xA4	
Reserved for future use	*Future*		00101 thru 11111	0xA5 thru 0xBF	
All other MFI / Labels (i.e., first character is not 'A' or 'B')	ACARS AOC	110 "ACARS AOC/ IPS DS"	00000	0xC0	5913 aoc-acars
Reserved for future use	*Future*		00001 thru 11111	0xC1 thru 0xDF	
Note 1: Refer to ICAO Doc. 9896 for the specification of all other Application Technology Type values (i.e., 0x00 thru 0x9F and 0xE0 thru 0xFF).					

3.2.2.2 Application Identifier

Commented [OML4]: Ed. Note - AppID, which was an abstract representation of the specific application, is no longer necessary since the application is now identified explicitly by the AppTechType.

The AICF Mapping Function also provides the IPS DS with an Application Identifier (AppID). The IPS DS uses the AppID to select the appropriate transport layer port number from the application-specific port numbers that are registered with the Internet Assigned Numbers Authority (IANA) and specified in ICAO Doc. 9896.

COMMENTARY

The use of AppID to communicate port information is an optional implementation construct; alternatively, the AICF mapping function could identify the port directly. However, for the purposes of specifying the AICF, the AppID serves as a convenient abstract representation in lieu of referencing detailed port numbers.

As shown in Table 3-3, the Mapping Function determines the specific application based on the value of the ACARS message Label or the MFI (in the case of an ACARS peripheral) contained in the Uncompressed Application Data.

Table 3-3 – Application Identifier

Value of ACARS MFI or Label	Application Identifier (AppID)	Associated Port Assignment (per ICAO Doc. 9896, Part II, Section 2.2)
A0 (uplink) B0 (downlink)	"ARINC 622 AFN"	5915
A6 (uplink)	"ARINC 622 ADS-C"	5917

~~B6 (downlink)~~		
~~AA (uplink)~~ ~~BA (downlink)~~	~~"ARINC 622 CPDLC"~~	~~5916~~
~~AW (uplink)~~ ~~BW (downlink)~~	~~"ARINC 622 ATS WIND"~~	~~5918~~
~~Ax (uplink)~~ ~~Bx (downlink)~~ ~~(i.e., MFI / Labels starting with 'A' or 'B', except: A0, B0, A6, B6, AA, BA, AW, BW)~~	~~"ARINC 623"~~	~~5919~~
~~All other MFI / Labels (i.e., the first character is not 'A' or 'B')~~	~~"AOC"~~	~~5914~~

~~The AICF may communicate the AppID to the IPS DS via the same local, implementation-dependent interface used to communicate the Application Technology Type.~~

3.2.3 Dialogue Service Parameters

3.2.3.1 Dialogue Service Primitive

For downlink messages, the dialogue service primitive is selected based on parameters and values contained in the Uncompressed Application Data, as well as the current state of the dialogue, in accordance with the application-specific criteria specified in Section 3.4 in this attachment.

3.2.3.2 Called and Calling Peer ID Parameters

For downlink messages, the Called Peer ID parameter identifies the intended ground IPS DS peer recipient, and the Calling Peer ID parameter identifies the airborne IPS DS peer originator that is sending the downlink.

For all ACARS ATS application downlink messages (ARINC 622 and ARINC 623), the Called Peer ID parameter contains the ATC Facility designator or address, which is the single 4- or 7-character supplementary address contained in the Uncompressed Application Data. The ATC Facility designator or address is located between the "/" character and "." character excluding any optional MFI and space <sp> character, as follows:

ACARS MU/CMU-hosted Application:

`Label/`**`ATC_Facility.`**`Application_text`

ACARS Peripheral-hosted Application:

`Label#Sub-label/MFI<sp>`**`ATC_Facility.`**`Application_text`

For all ACARS AOC application downlink messages, the Called Peer ID parameter is not included since the FlightID also contains the airline identifier. For all ACARS ATS and AOC downlink messages, the Calling Peer ID parameter contains the FlightID that is obtained via the ACARS Message Interface, as described in Section 3.1.1.1. The content and length of the Called and Calling Peer ID parameters for downlink messages are summarized in the following table.

Table 3-3 – Called and Calling Peer ID Parameter Content: Downlink

Application Technology Type	Called Peer ID		Calling Peer ID	
	Value	Length (bytes)	Value	Length (bytes)
0xA0 thru 0xBF "ACARS ATS/IPS DS"	ATC Facility designator or address	4 or 7	FlightID	6
0xC0 thru 0xDF "ACARS AOC/IPS DS"	Not included	--	FlightID	6

The Called Peer ID and Calling Peer ID parameters are mandatory when the dialogue service primitive is a D-START, D-STARTCNF, D-DATA, or D-ABORT.

3.2.3.3 Result Parameter

For downlink messages, the Result parameter indicates the airborne acceptance or rejection of a ground-initiated request to establish a dialogue for an application. The value of the Result parameter is per ICAO Doc. 9896, and the application-specific criteria for setting the parameter value is specified in Section 3.4 in this attachment. The AICF Mapping Function uses the Result value to set the status of the dialogue (i.e., "open" when Result is accepted or "closed" when Result is rejected).

The Result parameter is a mandatory parameter when the dialogue service primitive is a D-STARTCNF; otherwise, the parameter is not present for other primitives.

3.3 AICF Uplink Message Processing

Figure 3-5 illustrates the processing of uplink messages to an ACARS application or peripheral.

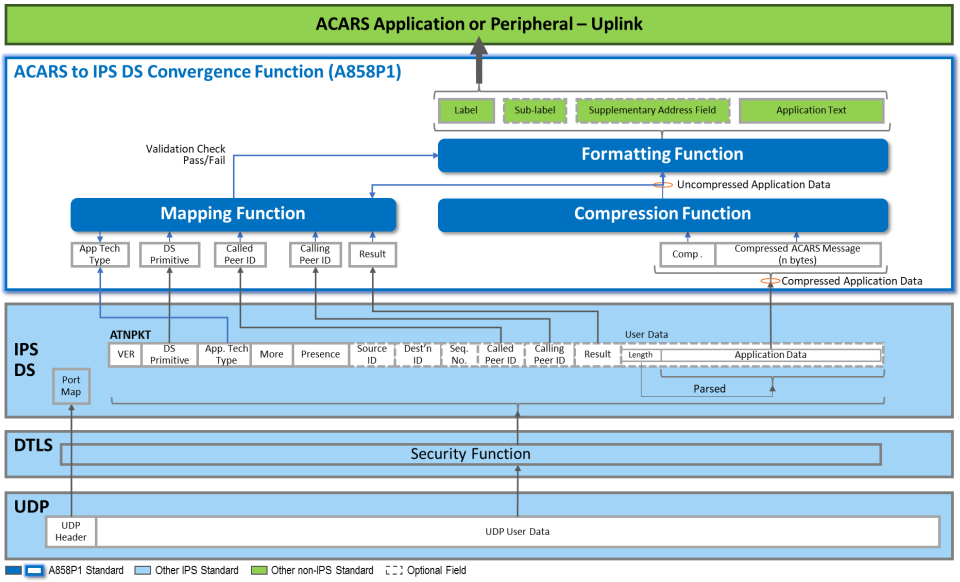

Figure 3-5 – AICF Uplink Processing

For uplink messages, the AICF Mapping Function uses received values for the Application Technology Type, Dialogue Service primitive, Called Peer ID, Calling Peer ID, and Result fields to associate uplink responses with downlink requests and to perform consistency checks (e.g., detect a malformed message).

Other fixed and optional ATNPKT fields (e.g., Source ID and Destination ID), as shown in the IPS DS block in Figure 3-5 are consumed by the IPS DS for uplink messages, and they are not presented to the AICF via the dialogue service interface.

3.3.1 User Data

The User Data received via the IPS DS interface is the Compressed Application Data, which is the input to the Compression Function. As specified in Section 3.1.2.3, the Compression Function applies the appropriate data decompression algorithm to recover the Uncompressed Application Data.

If received parameters in the Uncompressed Application Data are validated by the AICF Mapping Function, as described in the following sections, the Formatting Function parses and conveys the received ACARS application message fields to the ACARS application or peripheral as specified in Section 3.1.2.2.

3.3.2 Application Technology Type~~Control Parameters~~

~~#### 3.3.2.1 Application Technology Type and Application Identifier~~

Upon receipt of an uplink message, the IPS DS communicates the Application Technology Type ~~and AppID~~ information to the AICF via a local, implementation-dependent interface. Once the Uncompressed Application Data is recovered, the AICF Mapping Function validates that the ACARS message Label or the MFI (in the case of an ACARS peripheral) in the received message is consistent with the Application Technology Type ~~and AppID~~, based on the values in Table 3-2 ~~and Table 3-3~~. If the consistency check is successful, then the AICF Mapping Function indicates success to the Formatting Function. If the consistency check fails, then the received message is considered malformed and the AICF discards it; recovery is relegated to the application layer, which detects when an expected response is not received before expiration of a message timer.

3.3.3 Dialogue Service Parameters

3.3.3.1 Dialogue Service Primitive

Upon receipt on an uplink message, the AICF Mapping Function validates the dialogue service primitive based on received parameters and the current state of the dialogue associated with the end entity sending the message.

Section 3.4 in this attachment specifies the application-specific criteria for identifying the expected uplink dialogue service primitive and for setting the dialogue status.

3.3.3.2 Called and Calling Peer ID

For uplink messages, the Called Peer ID parameter identifies the intended airborne IPS DS peer recipient, and the Calling Peer ID parameter identifies the ground IPS DS peer originator that is sending the uplink.

For all ACARS ATS and AOC uplink messages, the Called Peer ID parameter contains the aircraft FlightID. For all ACARS ATS application uplink messages (ARINC 622 and ARINC 623), the Calling Peer ID parameter contains the ATC

Facility designator or address. For all ACARS AOC application uplink messages, the Calling Peer ID parameter is not included since the FlightID also contains the airline identifier. The following table summarizes the content and length of the Called and Calling Peer ID parameters for uplink messages.

Table 3-4 – Called and Calling Peer ID Parameter Content: Uplink

Application Technology Type	Called Peer ID		Calling Peer ID	
	Value	Length (bytes)	Value	Length (bytes)
0xA0 thru 0xBF "ACARS ATS/IPS DS"	FlightID	6	ATC Facility designator or address	4 or 7
0xC0 thru 0xDF "ACARS AOC/IPS DS"	FlightID	6	No included	--

The Called Peer ID and Calling Peer ID parameters are mandatory when the dialogue service primitive is a D-START, D-STARTCNF, D-DATA, or D-ABORT. Both fields are consumed by the AICF and are not transferred to the ACARS application or peripheral. The AICF uses the information contained in the fields to perform the following consistency checks:

- For all uplink messages, verify that the length and format of the received parameter values are consistent with expected values (e.g., the value of the received FlightID matches the aircraft-local flight identifier value)
- For ATS uplink messages, verify that the received ATC Facility designator or address corresponds to the ATC Facility designator or address associated with an open dialogue

If the consistency check is successful, then the AICF Mapping Function indicates success to the Formatting Function. If the consistency check fails, then the received message is considered malformed and the AICF discards it; recovery is relegated to the application layer, which detects when an expected response is not received before expiration of a message timer.

3.3.3.3 Result Parameter

For uplink messages, the Result parameter indicates the ground acceptance or rejection of an air-initiated request to establish a dialogue for an application. The value of the Result parameter is per ICAO Doc. 9896, and the application-specific criteria for setting the parameter value is specified in Section 3.4 in this attachment. The AICF Mapping Function within the AICF uses the Result value to set the status of the dialogue (i.e., "open" when Result is accepted or "closed" when Result is rejected).

The Result parameter is a mandatory parameter when the dialogue service primitive is a D-STARTCNF; otherwise, the parameter is not present for other primitives.

3.4 Application-specific DS Primitive Mapping

This section specifies the parameters and values contained in ACARS-based messages that are used to select the dialogue service primitive. Two tables are included for each ACARS-based application: one for primitive mapping when the entity initiating the request does not have an existing dialogue, and one for primitive mapping once a dialogue is established. Each table includes the following information:

- **Procedure** – application-specific procedure (e.g., connection request)
- **Request** – Application message sent by an initiating entity
  - **Message** – Values that specify a specific application message
  - **UL/DL** – Indication of whether the message is an uplink (to aircraft) or downlink (from aircraft)
  - **DS Primitive** – dialogue service primitive for the specific application message
- **Response** – Application message sent by a responding entity
  - The sub-columns are defined the same as for Request
- **Dialogue Status** – status of the application-specific dialogue between the aircraft and a ground entity at the completion of the request-response sequence

The ACARS-based applications addressed in this section include:

- Section 3.4.1 – ARINC 622 – ATS Data Link Applications, including AFN, CPDLC, ADS-C, and ATS WIND
- Section 3.4.2 – ARINC 623 – Character-oriented ATS
- Section 3.4.3 – AOC

Each of these sections also describes the application-specific criteria for determining and setting the open/closed status of the dialogue.

3.4.1 ARINC 622 – ATS Data Link Applications

3.4.1.1 AFN Application

The Uncompressed Application Data is an AFN application message when ~~AppID equals "ARINC 622 AFN" per the criteria in~~ the Application Technology Type field equals 0xA0 per Table 3-2 in this attachment.

In each of the following tables, the columns labeled "A622 Message" specify the three-character Imbedded Message Identified (IMI) and Message Type Identifier (MTI) values that are contained in the application text and which identify a specific AFN application message, including:

- FN_CON (MTI = FPO) – AFN Contact message
- FN_AK (MTI = FAK) – AFN Acknowledge message
- FN_CAD (MTI = FCA) – AFN Contact Advisory message
- FN_RESP (MTI = FRP) – AFN Response message
- FN_COMP (MTI = FCP) – AFN Complete message

The IMI and MTI values are used to select the dialogue service primitive in concert with the current state of the dialogue, which is associated with a specific ground center (i.e., the ATC Facility designator or address included in the Called or Calling Peer ID parameter). Some AFN messages include a one-byte reason code after the MTI, and the value of this code is used to determine the dialogue status upon completion of the request-response sequence.

The following table specifies the primitive mapping when a dialogue does not exist, meaning that the dialogue status is "AFN-CLOSED" for the entity initiating the request. "AFN-CLOSED" is the initial state when the AICF is initialized.

Table 3-5 – DS Primitive Mapping for AFN Application: No Existing Dialogue

Procedure	Request			Response			Dialogue Status	Notes
	A622 Message	UL/ DL	DS Primitive	A622 Message	UL/ DL	DS Primitive		
Contact (Logon)	IMI = AFN MTI = FPO	DL	**D-START**	IMI = AFN MTI = FAK Reason = 0	UL	**D-STARTCNF** Result = Accepted	AFN-OPEN	1, 2
				IMI = AFN MTI = FAK Reason ≠ 0	UL	**D-STARTCNF** Result = Rejected	AFN-CLOSED	
Note 1: Initial AFN logon to a center when no dialogue with that center exists. Note 2: When a ground center has an existing (i.e., residual) AFN dialogue with the aircraft that is initiating a new AFN dialogue using D-START, then the ground center supplants the existing dialogue with the new dialogue.								

The following table specifies the primitive mapping when a dialogue exists, meaning that the dialogue status is “AFN-OPEN” for the entity initiating the request.

Table 3-6 – DS Primitive Mapping for AFN Application: Existing Dialogue

Procedure	Request			Response			Dialogue Status	Notes
	A622 Message	UL/ DL	DS Primitive	A622 Message	UL/ DL	DS Primitive		
Contact (Logon)	IMI = AFN MTI = FPO	DL	**D-DATA**	IMI = AFN MTI = FAK Reason = 0	UL	**D-DATA**	AFN-OPEN	1
				IMI = AFN MTI = FAK Reason ≠ 0	UL	**D-DATA**	AFN-CLOSED	
Address Forwarding	IMI = AFN MTI = FCA	UL	**D-DATA**	IMI = AFN MTI = FRP	DL	**D-DATA**	(no change)	
	IMI = AFN MTI = FCP Reason = 0	DL	**D-DATA**	None	--	**--**	AFN-CLOSED	
	IMI = AFN MTI = FCP Reason ≠ 0	DL	**D-DATA**	None	--	**--**	(no change)	2
Note 1: AFN logon to a center when there is an existing dialogue with that center, e.g., when the pilot enters a new flight number for a multi-leg flight. Note 2: If the result of the procedure not successful, then the AFN dialogue remains open.								

Once a dialogue for AFN messages is opened between an aircraft and a specific ground center, the dialogue remains open until one of the request-response sequences results in “AFN-CLOSED” dialogue status. In addition, the dialogue may be closed silently[1] and the dialogue status set to “AFN-CLOSED” upon the termination of a flight (e.g., weight-on-wheels and forward door open). The discrete inputs used to determine end-of-flight is implementation-dependent.

[1] Silently means that a dialogue is closed locally by the aircraft.

3.4.1.2 CPDLC Application

The Uncompressed Application Data is a CPDLC application message when ~~AppID equals "ARINC 622 CPDLC" per the criteria in~~the Application Technology Type field equals 0xA1 per Table 3-2 in this attachment.

In each of the following tables, the column labeled "A622 Message" specifies the three-character IMI value that is contained in the application text and which identifies a specific CPDLC application message. The IMI value is used to select the dialogue service primitive in concert with the current state of the dialogue, which is associated with a specific ground center (i.e., the ATC Facility designator or address included in the Called or Calling Peer ID parameter). The connection confirmation (CCx) and disconnect request (DRx) messages are used to set the dialogue status upon completion of the request-response sequences for connection initiation and termination, respectively.

The following table specifies the primitive mapping when a dialogue does not exist, meaning that the dialogue status is "CPDLC-CLOSED" for the entity initiating the request. "CPDLC-CLOSED" is the initial state when the AICF is initialized.

Table 3-7 – DS Primitive Mapping for CPDLC Application: No Existing Dialogue

Procedure	Request			Response			Dialogue Status	Notes
	A622 Message	UL/ DL	DS Primitive	A622 Message	UL/ DL	DS Primitive		
Connection Request	IMI = CRx	UL	**D-START**	IMI = CCx	DL	**D-STARTCNF** Result = Accepted	CPDLC-OPEN	1, 2, 3
				IMI = DRx	DL	**D-STARTCNF** Result = Rejected	CPDLC-CLOSED	
Note 1: In the IMI, the value of the third character 'x' is the version number of the message, e.g., CR1. Note 2: A successful AFN logon is required prior to an initial CPDLC connection request. After a successful AFN login, the aircraft will accept a CPDLC connection request from other centers within the same administrative domain, which is the recommended implementation per RTCA DO-258A/ EUROCAE ED-100A. Note 3: When the aircraft has an existing CPDLC dialogue with a specific center, a subsequent connection request (using D-START) from the same center is treated as a new CPDLC dialogue that supplants the existing dialogue.								

The following table specifies the primitive mapping when a dialogue exists, meaning that the dialogue status is "CPDLC-OPEN" for the entity initiating the request.

Table 3-8 – DS Primitive Mapping for CPDLC Application: Existing Dialogue

Procedure	Request			Response			Dialogue Status	Notes
	A622 Message	UL/ DL	DS Primitive	A622 Message	UL/ DL	DS Primitive		
Connection Request	IMI = CRx	UL	**D-START**	IMI = CCx	DL	**D-STARTCNF** Result = Accepted	CPDLC-OPEN	1, 2
				IMI = DRx	DL	**D-STARTCNF** Result = Rejected	CPDLC-CLOSED	

Uplink Message (UMxxx)	IMI = ATx	UL	**D-DATA**	Depends upon the uplink message	DL	**D-DATA**	(no change)	1
Downlink Message (DMxxx)	IMI = ATx	DL	**D-DATA**	Depends upon the downlink message	UL	**D-DATA**	(no change)	1
Connection Termination	IMI = ATx (UM161 end service)	UL	**D-DATA**	IMI = DRx	DL	**D-ABORT**	CPDLC-CLOSED	1, 3
	IMI = DRx (ground initiated)	UL	**D-ABORT**	None	--	**--**	CPDLC-CLOSED AFN-CLOSED	1, 4
	IMI = DRx (aircraft initiated)	DL	**D-ABORT**	None	--	--	CPDLC-CLOSED AFN-CLOSED	1, 5

Note 1: In the IMI, the value of the third character 'x' is the version number of the message, e.g., CR1.
Note 2: When the aircraft has an existing CPDLC dialogue with a specific center, a subsequent connection request (using D-START) from the same center is treated as a new CPDLC dialogue that supplants the existing dialogue.
Note 3: Ground-initiated end service uplink message UM161, which occurs after transfer from the current data authority (CDA) to next data authority (NDA), closes the CPDLC dialogue for the CDA when the disconnect request (DRx) message is sent.
Note 4: Ground-initiated disconnect request closes both the CPDLC and associated AFN dialogues when the disconnect request (DRx) message is sent. No impact on any open ADS-C dialogues. Since some avionics implementations do not support DRx uplinks, ICAO Doc. 10037 specifies using an ATx uplink containing a UM161 End Service message to terminate a CPDLC connection (refer to the prior row in this table).
Note 5: Aircraft-initiated (e.g., by pilot) disconnect request closes both the CPDLC and associated AFN dialogues when the disconnect request (DRx) message is sent. No impact on open ADS-C dialogues.

Once a dialogue for CPDLC messages is opened between an aircraft and a specific ground center, the dialogue remains open until one of the request-response sequences results in "CPDLC-CLOSED" dialogue status. In addition, the dialogue may be closed silently and the dialogue status set to "CPDLC-CLOSED" upon the termination of a flight (e.g., weight-on-wheels and forward door open). The discrete inputs used to determine end-of-flight is implementation-dependent.

3.4.1.3 ADS-C Application

The Uncompressed Application Data is an ADS-C application message when ~~AppID equals "ARINC 622 ADS-C" per the criteria in~~the Application Technology Type field equals 0xA2 per Table 3-2 in this attachment.

In each of the following tables, the column labeled "A622 Message" specifies the three-character IMI value that is contained in the application text and which identifies ADS-C application messages. The IMI value is used to select the dialogue service primitive in concert with the current state of the dialogue, which is associated with a specific ground center (i.e., the ATC Facility designator or address included in the Called or Calling Peer ID parameter). The disconnect (DIS) message

is used to set the dialogue status upon completion of the request-response sequence for connection termination.

The following table specifies the primitive mapping when a dialogue does not exist, meaning that the dialogue status is “ADS-CLOSED” for the entity initiating the request. “ADS-CLOSED” is the initial state when the AICF is initialized.

Table 3-9 – DS Primitive Mapping for ADS-C: No Existing Dialogue

Procedure	Request			Response			Dialogue Status	Notes
	A622 Message	UL/ DL	DS Primitive	A622 Message	UL/ DL	DS Primitive		
Initial Contract Establishment	IMI = ADS (any contract request)	UL	**D-START**	IMI = ADS (ACK or NAK tag)	DL	**D-STARTCNF** Result = Accepted	ADS-OPEN	1, 2
				IMI = DIS	DL	**D-STARTCNF** Result = Rejected	ADS-CLOSED	3
Note 1: Establish an ADS-C dialogue, which is independent of AFN and CPDLC dialogues. Note 2: When an aircraft has an existing (i.e., residual) ADS-C dialogue with the center that is initiating a new ADS-C dialogue using D-START, then the aircraft supplants the existing dialogue with the new dialogue. Note 3: When the ground attempts to establish an ADS-C contract but the aircraft ADS-C function is disabled, the aircraft response is a downlink disconnect request.								

The following table specifies the primitive mapping when a dialogue exists, meaning that the dialogue status is “ADS-OPEN” for the entity initiating the request.

Table 3-10 – DS Primitive Mapping for ADS-C Application: Existing Dialogue

Procedure	Request			Response			Dialogue Status	Notes
	A622 Message	UL/ DL	DS Primitive	A622 Message	UL/ DL	DS Primitive		
Contract Establishment	IMI = ADS (any contract request)	UL	**D-DATA**	IMI = ADS (ACK or NAK tag)	DL	**D-DATA**	(no change)	1
ADS-C Report	IMI = ADS (report tag + data)	DL	**D-DATA**	None	--	**--**	(no change)	
Cancel Contract(s) (Ground-initiated)	IMI = ADS (cancel contract)	UL	**D-DATA**	IMI = ADS (ACK or NAK tag)	DL	**D-DATA**	(no change)	2
	IMI = ADS (cancel all contracts and terminate connection)	UL	**D-DATA**	IMI = DIS	DL	**D-ABORT**	ADS-CLOSED	3
Disconnect (Aircraft-initiated)	IMI = DIS	DL	**D-ABORT**	None	--	**--**	ADS-CLOSED	4
Note 1: Establish an additional ADS-C contract(s) when there is an existing dialogue with the center requesting the contract.								

Note 2: Cancelling a specific contract does not close the dialogue, which allows other existing ADS-C contracts to be maintained and new ADS-C contracts to be established.
Note 3: Cancelling all contracts terminates the connection, which closes the ADS-C dialogue.
Note 4: An aircraft initiated disconnect may be the result of pilot action, three consecutive ADS-C negative acknowledgements (NAKs), or expiration of the ADS-C application inactivity timer.

Once a dialogue for ADS-C messages is opened between an aircraft and a specific ground center, the dialogue remains open until one of the request-response sequences results in “ADS-CLOSED” dialogue status.

3.4.1.4 ATS WIND Application

The Uncompressed Application Data is an ATS WIND application message when ~~AppID equals “ARINC 622 ATS WIND” per the criteria in~~the Application Technology Type field equals 0xA3 per Table 3-2 in this attachment.

In each of the following tables, the column labeled “A622 Message” specifies the three-character IMI value that is contained in the application text and which identifies a specific ATS WIND message. The IMI value is used to select the dialogue service primitive in concert with the current state of the dialogue, which is associated with a specific ground center (i.e., the ATC Facility designator or address included in the Called or Calling Peer ID parameter).

The following table specifies the primitive mapping when a dialogue does not exist, meaning that the dialogue status is “WIND-CLOSED” for the entity initiating the request. “WIND-CLOSED” is the initial state when the AICF is initialized.

Table 3-11 – DS Primitive Mapping for ATS WIND Application: No Existing Dialogue

Procedure	Request			Response			Dialogue Status	Notes
	A622 Message	UL/ DL	DS Primitive	A622 Message	UL/ DL	DS Primitive		
Uplink Wind	IMI = PWF or PWI	UL	**D-START**	IMI = RES	DL	**D-STARTCNF** Result = Accepted	WIND-OPEN	1
				IMI = REJ	DL	**D-STARTCNF** Result = Rejected	WIND-CLOSED	
Note 1: When the aircraft has an existing (i.e., residual) ATS WIND dialogue with the center that is initiating a new ATS WIND dialogue, then the aircraft supplants the existing dialogue with the new dialogue.								

The following table specifies the primitive mapping when a dialogue exists, meaning that the dialogue status is “WIND-OPEN” for the entity initiating the request.

Table 3-12 – DS Primitive Mapping for ATS WIND Application: Existing Dialogue

Procedure	Request			Response			Dialogue Status	Notes
	A622 Message	UL/ DL	DS Primitive	A622 Message	UL/ DL	DS Primitive		
Uplink Wind	IMI = PWF or PWI	UL	**D-DATA**	IMI = RES	DL	**D-DATA**	(no change)	
				IMI = REJ	DL	**D-ABORT**	WIND-CLOSED	1

Note 1: A rejection (REJ) downlink is sent when one or more errors is detected in the uplink, which closes the ATS WIND dialogue.

Once a dialogue for ATS WIND messages is opened between an aircraft and a specific ground center, the dialogue remains open until one of the request-response sequences results in “WIND-CLOSED” dialogue status. In addition, the dialogue may be closed silently and the dialogue status set to “WIND-CLOSED” upon the termination of a flight (e.g., weight-on-wheels and forward door open). The discrete inputs used to determine end-of-flight is implementation-dependent.

3.4.2 ARINC 623 – Character-oriented ATS

The Uncompressed Application Data is an ARINC 623 application message when ~~AppID equals “ARINC 623” per the criteria in~~the Application Technology Type field equals 0xA4 per Table 3-2 in this attachment.

In each of the following tables, the column labeled “A623 Message” specifies the three-character IMI value that is contained in the application text and which identifies a specific ARINC 623 message. The IMI value is used to select the dialogue service primitive in concert with the current state of the dialogue, which is associated with a specific ground center (i.e., the ATC Facility designator or address included in the Called or Calling Peer ID parameter).

The following table specifies the primitive mapping when a dialogue does not exist, meaning that the dialogue status is “623-CLOSED” for the entity initiating the request. “623-CLOSED” is the initial state when the AICF is initialized.

Table 3-13 – DS Primitive Mapping for ARINC 623 Messages: No Existing Dialogue

Procedure	Request			Response			Dialogue Status	Notes
	A623 Message	UL/ DL	DS Primitive	A623 Message	UL/ DL	DS Primitive		
Pushback Clearance	IMI = PCx	DL	**D-START**	IMI = PCx or FSx	UL	**D-STARTCNF** Result = Accepted	623-OPEN	1, 2, 3
Taxi Clearance	IMI = ETx	DL	**D-START**	IMI = ETx or FSx	UL	**D-STARTCNF** Result = Accepted	623-OPEN	1, 2, 3
Departure Clearance	IMI = DCx	DL	**D-START**	IMI = DCx or FSx	UL	**D-STARTCNF** Result = Accepted	623-OPEN	1, 2, 3
Oceanic Clearance	IMI = OCx	DL	**D-START**	IMI = OCx or FSx	UL	**D-STARTCNF** Result = Accepted	623-OPEN	1, 2, 3
Automatic Terminal Information Service	IMI = TIx	DL	**D-START**	IMI = TIx	UL	**D-STARTCNF** Result = Accepted	623-OPEN	1, 2
Terminal Weather	IMI = TWx	DL	**D-START**	IMI = TWx	UL	**D-STARTCNF** Result = Accepted	623-OPEN	1, 2

Note 1: In the IMI, the value of the third character ‘x’ is the version number of the message.
Note 2: When a ground center has an existing (i.e., residual) ARINC 623 dialogue with the aircraft that is initiating a hew ARINC 623 dialogue, then the ground center supplants the existing dialogue with the new dialogue.

Note 3: The uplink response may be a Flight Service message (IMI = FSx), which the ground ATC may use to indicate: the status of the request; the need to standby while processing is completed; or the need to revert to voice in the event of an error with the request.

The following table specifies the primitive mapping when a dialogue exists, meaning that the dialogue status is "623-OPEN" for the entity initiating the request.

Table 3-14 – DS Primitive Mapping for ARINC 623 Messages: Existing Dialogue

Procedure	Request			Response			Dialogue Status	Notes
	A623 Message	UL/ DL	DS Primitive	A623 Message	UL/ DL	DS Primitive		
Pushback Clearance	IMI = PCx	DL	**D-DATA**	IMI = PCx or FSx	UL	**D-DATA**	(no change)	1, 2
Taxi Clearance	IMI = ETx	DL	**D-DATA**	IMI = ETx or FSx	UL	**D-DATA**	(no change)	1, 2
Departure Clearance	IMI = DCx	DL	**D-DATA**	IMI = DCx or FSx	UL	**D-DATA**	(no change)	1, 2
Oceanic Clearance	IMI = OCx	DL	**D-DATA**	IMI = OCx or FSx	UL	**D-DATA**	(no change)	1, 2
Automatic Terminal Information Service	IMI = TIx	DL	**D-DATA**	IMI = TIx	UL	**D-DATA**	(no change)	1
Terminal Weather	IMI = TWx	DL	**D-DATA**	IMI = TWx	UL	**D-DATA**	(no change)	1

Note 1: In the IMI, the value of the third character 'x' is the version number of the message.
Note 2: The uplink response may be a Flight Service message (IMI = FSx), which the ground ATC may use to indicate: the status of the request; the need to standby while processing is completed; or the need to revert to voice in the event of an error with the request.

Once a dialogue for ARINC 623 messages is opened between an aircraft and a specific ground center, the dialogue remains open until a subsequent D-START sequence restarts the dialogue. In addition, the dialogue may be closed silently and the dialogue status set to "623-CLOSED" upon the termination of a flight (e.g., weight-on-wheels and forward door open). The discrete inputs used to determine end-of-flight is implementation-dependent.

In addition, to minimize the number of open dialogues, the Airborne IPS System may be designed to maintain a single ARINC 623 dialogue such that a dialogue with each new ground center silently closes a prior dialogue with a previous ground center. For example, a dialogue may be opened initially at a departure airport when the flight crew requests weather information and/or pushback/taxi clearance. During flight, an open ARINC 623 dialogue is closed silently each time the flight crew requests weather information from a different airport. Upon arrival, an open ARINC 623 dialogue is closed silently when the flight crew requests weather information and/or taxi clearance from the arrival airport.

3.4.3 AOC

The Uncompressed Application Data is an AOC message when ~~AppID equals "AOC" per the criteria in~~the Application Technology Type field equals 0xC0 per Table 3-2 in this attachment. In each of the following tables, the presence of an AOC Label or MFI is used to select the dialogue service primitive in concert with the current state of the dialogue.

The following table specifies the primitive mapping when a dialogue does not exist, meaning that the dialogue status is "AOC-CLOSED" for the entity initiating the request. "AOC-CLOSED" is the initial state when the AICF is initialized.

Table 3-15 – DS Primitive Mapping for AOC Messages: No Existing Dialogue

Procedure	Request			Response			Dialogue Status	Notes
	AOC Message	UL/ DL	DS Primitive	AOC Message	UL/ DL	DS Primitive		
Any AOC UL or DL message	Any	Any	**D-START**	Any	Any	**D-STARTCNF** Result = Accepted	AOC-OPEN	1
Note 1: The first AOC message is sent in a D-START to initiate the dialogue. This occurs upon AICF initialization or whenever the aircraft FlightID changes.								

The following table specifies the primitive mapping when a dialogue exists, meaning that the dialogue status is "AOC-OPEN" for the entity initiating the request.

Table 3-16 – DS Primitive Mapping for AOC Messages: Existing Dialogue

Procedure	Request			Response			Dialogue Status	Notes
	AOC Message	UL/ DL	DS Primitive	AOC Message	UL/ DL	DS Primitive		
Any AOC UL or DL message	Any	Any	**D-DATA**	Any	Any	**D-DATA**	(no change)	

Once a dialogue for AOC messages is opened, the dialogue remains open until the FlightID changes and a subsequent D-START sequence restarts the dialogue. In addition, the dialogue may be closed silently and the dialogue status set to "AOC-CLOSED" upon expiration of an application inactivity timer or upon the termination of a flight (e.g., weight-on-wheels and forward door open), whichever occurs first. The discrete inputs used to determine end-of-flight is implementation-dependent.