

# Fieldbus —

## Part 5: Specification for Fieldbus access sublayer (Services and protocol)

ICS 25.040.40; 35.100

NO COPYING WITHOUT BSI PERMISSION EXCEPT AS PERMITTED BY COPYRIGHT LAW

**BSI**  
Documents

**AVA**

Document Sharing Hub

# Committees responsible for this Draft for Development

The preparation of this Draft for Development was entrusted to Technical Committee AMT/7, Monitoring and control aspects of advanced manufacturing technology, upon which the following bodies were represented:

- ABB Kent plc
- Cegelec Industrial Controls Ltd.
- Department of Trade and Industry
- Fisher-Rosemount Ltd.
- Foxboro Great Britain
- Institution of Electrical Engineers
- Measurement Technology Ltd.
- Process Automation and Computer Systems Ltd.
- Schneider Ltd.
- Siemens Automation plc

This Draft for Development, having been prepared under the direction of the Advanced Manufacturing Technology Assembly, was published under the authority of the Standards Board and comes into effect on 15 October 1996

© BSI 05-2000

The following BSI reference relates to the work on this Draft for Development: Committee reference AMT/7

ISBN 0 580 26487 4

## Amendments issued since publication

| Amd. No. | Date | Comments |
|----------|------|----------|
|          |      |          |
|          |      |          |
|          |      |          |
|          |      |          |
|          |      |          |

# Contents

|                                                                  | Page               |
|------------------------------------------------------------------|--------------------|
| Committees responsible                                           | Inside front cover |
| Foreword                                                         | v                  |
| 1 Scope                                                          | 1                  |
| 2 Overview                                                       | 1                  |
| 3 Architecture                                                   | 1                  |
| 3.1 FAS Interfaces                                               | 1                  |
| 3.2 Application Relationship                                     | 2                  |
| 3.3 AREP Characteristics                                         | 2                  |
| 3.3.1 AREP Context                                               | 2                  |
| 3.3.2 AREP Roles                                                 | 2                  |
| 3.3.3 AREP Cardinality                                           | 4                  |
| 3.3.4 AR Cardinality                                             | 4                  |
| 3.3.5 Conveyance Path                                            | 4                  |
| 3.3.6 Trigger Policy                                             | 5                  |
| 3.3.7 Conveyance Policy                                          | 5                  |
| 3.3.8 Identification of AREPs                                    | 5                  |
| 3.4 AR Establishment                                             | 5                  |
| 3.5 Underlying Communications Services                           | 6                  |
| 3.5.1 Connectionless and Connection-Oriented Services            | 6                  |
| 3.5.2 Buffer and Queue                                           | 6                  |
| 3.5.3 Cyclic and Acyclic Services                                | 6                  |
| 3.5.4 Quality of Service                                         | 6                  |
| 3.5.5 Segmentation                                               | 6                  |
| 3.5.6 Timeliness                                                 | 6                  |
| 4 FAS Model                                                      | 6                  |
| 4.1 Application Relationship End Point Classes                   | 6                  |
| 4.1.1 QUU (Queued User-triggered Unidirectional) AREP Class      | 6                  |
| 4.1.2 QUB (Queued User-triggered Bidirectional) AREP Class       | 7                  |
| 4.1.3 BNU (Buffered Network-Scheduled Unidirectional) AREP Class | 7                  |
| 5 Operational Services                                           | 7                  |
| 5.1 Conventions                                                  | 8                  |
| 5.1.1 Parameters                                                 | 8                  |
| 5.1.2 General conventions                                        | 8                  |
| 5.2 Associate (ASC)                                              | 8                  |
| 5.2.1 Argument                                                   | 9                  |
| 5.2.2 Result(+)                                                  | 9                  |
| 5.2.3 Result(−)                                                  | 9                  |
| 5.3 Data Transfer Confirmed (DTC)                                | 10                 |
| 5.3.1 Argument                                                   | 10                 |
| 5.3.2 Result                                                     | 10                 |
| 5.4 Data Transfer Unconfirmed (DTC)                              | 11                 |
| 5.4.1 Argument                                                   | 11                 |
| 5.5 Abort (ABT)                                                  | 12                 |
| 5.5.1 Argument                                                   | 12                 |
| 5.6 FAS-Compel (FCMP)                                            | 13                 |
| 5.6.1 Argument                                                   | 13                 |
| 5.6.2 Result                                                     | 13                 |
| 5.7 Get-Buffered-Message (GBM)                                   | 14                 |

|                                                                 | Page |
|-----------------------------------------------------------------|------|
| 5.7.1 Argument                                                  | 14   |
| 5.7.2 Result(+)                                                 | 14   |
| 5.7.3 Result(-)                                                 | 14   |
| 5.8 FAS-Status (FSTS)                                           | 15   |
| 5.8.1 Argument                                                  | 15   |
| 6 AREP Class Specifications                                     | 15   |
| 6.1 AREP Common Model                                           | 16   |
| 6.1.1 Attributes                                                | 16   |
| 6.1.2 FAS Services                                              | 16   |
| 6.2 Queued User-triggered Unidirectional AREP Formal Model      | 16   |
| 6.2.1 Attributes                                                | 16   |
| 6.2.2 FAS Services                                              | 16   |
| 6.3 Queued User-triggered Bidirectional AREP Formal Model       | 17   |
| 6.3.1 Attributes                                                | 17   |
| 6.3.2 FAS Services                                              | 17   |
| 6.4 Buffered Network-Scheduled Unidirectional AREP Formal Model | 17   |
| 6.4.1 Attributes                                                | 18   |
| 6.4.2 FAS Services                                              | 18   |
| 6.5 Mapping to Data Link Layer                                  | 18   |
| 6.5.1 DLL Mapping of QUU AREP Class                             | 18   |
| 6.5.2 DLL Mapping of QUB AREP Class                             | 20   |
| 6.5.3 DLL Mapping of BNU AREP Class                             | 23   |
| 7 FAS Protocol State Machines                                   | 26   |
| 7.1 Conventions Used for State Machines                         | 29   |
| 7.2 FAS Service Protocol Machine (FSPM)                         | 30   |
| 7.2.1 FSPM State Tables                                         | 30   |
| 7.3 AR Protocol Machine (ARPM)                                  | 32   |
| 7.3.1 AR Protocol Machine (ARPM) for QUU AREP                   | 32   |
| 7.3.2 AR Protocol Machine (ARPM) for QUB AREP                   | 33   |
| 7.3.3 AR Protocol Machine (ARPM) for BNU AREP                   | 39   |
| 7.4 DLL Mapping Protocol Machine (DMPM)                         | 42   |
| 7.4.1 DMPM States                                               | 43   |
| 7.4.2 DMPM State Table                                          | 43   |
| 7.5 Primitive Definitions                                       | 47   |
| 7.5.1 Primitives Exchanged between FAS user and FSPM            | 47   |
| 7.5.2 Parameters of FAX-User/FSPM Primitives                    | 48   |
| 7.5.3 Primitives Exchanged between ARPM and FSPM                | 49   |
| 7.5.4 Parameters of FSPM/ARPM Primitives                        | 50   |
| 7.5.5 Primitives Exchanged between DMPM and ARPM                | 51   |
| 7.5.6 Parameters of ARPM/DMPM Primitives                        | 52   |
| 7.5.7 Primitives Exchanged between Data Link Layer and DMPM     | 52   |
| 7.5.8 Parameters of DMPM/Data Link Layer Primitives             | 55   |
| 7.6 Functions                                                   | 56   |
| 7.6.1 Functions Used by FSPM                                    | 56   |
| 7.6.2 Functions Used by ARPM                                    | 56   |
| 7.6.3 Functions Used by DMPM                                    | 57   |
| 8 FAS-PDUs                                                      | 58   |
| 8.1 Coding of the FAS-PDU Types                                 | 58   |

|                                                                                                                                   | Page |
|-----------------------------------------------------------------------------------------------------------------------------------|------|
| 8.2 General Structure of an FAS-PDU                                                                                               | 58   |
| 8.2.1 FAS Header                                                                                                                  | 58   |
| 8.2.2 Protocol Identifier                                                                                                         | 58   |
| 8.2.3 Protocol Specific Part                                                                                                      | 58   |
| 8.3 FAS-DPU Type Specific Information                                                                                             | 59   |
| 8.3.1 ASC_ReqPDU Contents                                                                                                         | 59   |
| 8.3.2 ASC_RspPDU Contents                                                                                                         | 59   |
| 8.3.3 ASC_ErrPDU Contents                                                                                                         | 59   |
| 8.3.4 ABT_PDU Contents                                                                                                            | 59   |
| 8.3.5 DTC_ReqPDU Contents                                                                                                         | 59   |
| 8.3.6 DTC_RspPDU Contents                                                                                                         | 60   |
| 8.3.7 DTU_PDU Contents                                                                                                            | 60   |
| Annex A Time Sequence Diagrams of FAS Primitives                                                                                  | 61   |
| Annex B Data Link Layer Service Mapping                                                                                           | 65   |
| Annex C (normative) Reason codes                                                                                                  | 66   |
| Figure 1 — General DD 238 Architecture                                                                                            | 1    |
| Figure 2 — Execution of the Associate (ASC — service at the FAS user-FAS interface for Client/Server ARs)                         | 9    |
| Figure 3 — Execution of the Associate (ASC — service at the FAS user- FAS interface for Sink/Source and Publisher/Subscriber ARs) | 10   |
| Figure 4 — Execution of the Data Transfer Confirmed (DTC — service at the FAS user- FAS interface)                                | 11   |
| Figure 5 — Execution of the Data Transfer Unconfirmed (DTU — service at the FAS user- FAS interface)                              | 12   |
| Figure 6 — Execution of the Abort service at the FAS user- FAS interface                                                          | 13   |
| Figure 7 — Time Sequence Diagram for the FAS-Compel Service                                                                       | 13   |
| Figure 8 — Time Sequence Diagram for the Get Buffered Message Service                                                             | 15   |
| Figure 9 — FAS-Status Service                                                                                                     | 15   |
| Figure 10 — Relationships Among Protocol Machines and Adjacent Layers                                                             | 26   |
| Figure 11 — State Transition Diagram of FSPM                                                                                      | 30   |
| Figure 12 — State Transition Diagram of the QUU ARPM                                                                              | 32   |
| Figure 13 — State Transition Diagram of QUB ARPM                                                                                  | 34   |
| Figure 14 — State Transition Diagram of the BNU ARPM                                                                              | 40   |
| Figure 15 — State Transition Diagram of DMPM                                                                                      | 43   |
| Figure 16 — Time Sequence Diagram of Primitives and FAS-PDUs used with QUU AREP                                                   | 61   |
| Figure 17 — Time Sequence Diagram for Primitives and FAS-PDUs used with QUB AREP (1)                                              | 62   |
| Figure 18 — Time Sequence Diagram for Primitives and FAS-PDUs used with QUB AREP (2)                                              | 63   |
| Figure 19 — Time Sequence Diagram of Primitives and FAS-PDUs used with BNU AREP (1)                                               | 64   |
| Figure 20 — Time Sequence Diagram for Primitives and FAS-PDUs used with BNU AREP (2)                                              | 65   |
| Table 1 — Characteristics of AREP Roles                                                                                           | 4    |
| Table 2 — AREP Cardinality by AREP Type                                                                                           | 4    |
| Table 3 — AR Cardinality by AR Type                                                                                               | 4    |

|                                                                               | Page |
|-------------------------------------------------------------------------------|------|
| Table 4 — AREP Classes and Characteristics                                    | 6    |
| Table 5 — Associate (ASC)                                                     | 8    |
| Table 6 — Data Transfer Confirmed Service                                     | 10   |
| Table 7 — Data Transfer Unconfirmed Service                                   | 11   |
| Table 8 — Abort                                                               | 12   |
| Table 9 — FAS-Compel Service                                                  | 13   |
| Table 10 — Get-Buffered-Message Service                                       | 14   |
| Table 11 — FAS-Status                                                         | 15   |
| Table 12 — Primitives Sent from the FAS User to the FSPM                      | 27   |
| Table 13 — Primitives Sent from the FSPM to the FAS User                      | 27   |
| Table 14 — Primitives Sent from the FSPM to the ARPM                          | 28   |
| Table 15 — Primitives Sent from the ARPM to the FSPM                          | 28   |
| Table 16 — The Primitive Sent from the ARPM to DMPM                           | 28   |
| Table 17 — Primitives Sent from the DMPM to the ARPM                          | 28   |
| Table 18 — Primitives Sent from the DMPM to the Data Link Layer               | 28   |
| Table 19 — Primitives Sent from the Data Link Layer to the DMPM               | 29   |
| Table 20 — FSPM State Table — Sender Transactions                             | 30   |
| Table 21 — FSPM State Table — Receiver Transactions                           | 31   |
| Table 22 — QUU ARPM States                                                    | 32   |
| Table 23 — QUU ARPM State Table — Sender Transactions                         | 32   |
| Table 24 — QUU ARPM State Table — Receiver Transactions                       | 33   |
| Table 25 — QUB ARPM States                                                    | 33   |
| Table 26 — QUB ARPM State Table — Sender Transactions                         | 35   |
| Table 27 — QUB ARPM State Table — Receiver Transactions                       | 36   |
| Table 28 — BNU ARPM States                                                    | 39   |
| Table 29 — BNU ARPM State Table — Sender Transactions                         | 40   |
| Table 30 — BNU ARPM State Table — Receiver Transactions                       | 41   |
| Table 31 — DMPM State Descriptions                                            | 43   |
| Table 32 — DMPM State Table — Sender Transactions                             | 43   |
| Table 33 — DMPM State Table — Receiver Transactions                           | 45   |
| Table 34 — Primitives issued by FAS User to FSPM                              | 47   |
| Table 35 — Primitives issued by FSPM to FAS User                              | 48   |
| Table 36 — Primitives issued by FSPM to ARPM                                  | 49   |
| Table 37 — Primitives issued by ARPM to FSPM                                  | 49   |
| Table 38 — Parameters Used with Primitives Exchanged<br>between FSPM and ARPM | 50   |
| Table 39 — Primitives issued from ARPM to DMPM                                | 51   |
| Table 40 — Primitives issued by DMPM to ARPM                                  | 51   |
| Table 41 — Parameters Used with Primitives Exchanged<br>between ARPM and DMPM | 52   |
| Table 42 — Primitives Exchanged between Data Link Layer<br>and DMPM           | 53   |

# Foreword

This Part of DD 238 has been prepared by Technical Committee, AMT/7. It supersedes DD 236-6:1996, which is withdrawn.

Portions of this document are owned by Fieldbus Foundation and are being used with their expressed written permission.

This publication contains the following parts:

- *Part 1: Guide to system architecture*; (Equivalent to FF-800 Revision 1.0<sup>1)</sup>)
- *Part 2: Specification for 31.25 kbit/s — Physical layer profile*; (Equivalent to FF-816 Revision 1.0<sup>1)</sup>)
- *Part 3: Specification for Data link layer services subset*; (Equivalent to FF-821 Revision 1.0<sup>1)</sup>)
- *Part 4: Specification for Data link protocol*; (Equivalent to FF-822 Revision 1.0<sup>1)</sup>)
- *Part 5: Specification for Fieldbus access sublayer (Services and protocol)*; (Equivalent to FF-875 Revision 1.0<sup>1)</sup>)
- *Part 6: Fieldbus Message Specification*; (Equivalent to FF-870 Revision 1.0<sup>1)</sup>)
- *Part 7: Fieldbus management*;
- *Section 7.1: Specification for System management*; (Equivalent to FF-880 Revision 1.0<sup>1)</sup>)
- *Section 7.2: Specification for Network management*. (Equivalent to FF-801 Revision 1.0<sup>1)</sup>)

**This Draft for Development is not to be regarded as a British Standard.**

It is being issued in the Draft for Development series of publications and is of a provisional nature because the subject matter is still under development requiring wider exposure. It should be applied on this provisional basis, so that information and experience of its practical application may be obtained.

A review of this Draft for Development will be carried out not later than 2 years after its publication. Notification of the start of the review period, with a request for the submission of comments from users of this Draft for Development, will be made in an announcement in the appropriate issue of *BSI Update*.

According to the replies received, the responsible BSI Committee will judge whether the Draft for Development can be converted into a British Standard or what other action should be taken.

Observations which it is felt should receive attention before the official call for comments will be welcomed.

These should be sent to the Secretary of BSI Technical Committee AMT/7 at BSI, 389 Chiswick High Road, London, W4 4AL, giving the document reference and clause number and proposing, where possible, an appropriate revision of the text.

<sup>1)</sup> Available from the Fieldbus Foundation, 9390 Research Boulevard, Suite II-250, Austin, Texas 78759, USA.

### **Disclaimer of warranties**

This document is provided on an “as is” basis and may be subject to future additions, modifications, or corrections depending on the results of field trial testing. BSI and the Fieldbus Foundation hereby disclaims all warranties of any kind, express or implied, including any warranty of merchantability or fitness for a particular purpose, for this document. In no event will they be responsible for any loss or damage arising out of or resulting from any defect, error or omission in this document or from anyone’s use of or reliance on this document.

### **Summary of pages**

This document comprises a front cover, an inside front cover, pages i to vi, pages 1 to 66 and a back cover.

This standard has been updated (see copyright date) and may have had amendments incorporated. This will be indicated in the amendment table on the inside front cover.

This specification is intended to be read in conjunction with the DD 238 Fieldbus Message Specification (FMS), DD 238 Network Management Specification, and DD 238 Data Link Layer Services and Protocol specifications.

This specification first describes the FAS architecture and the model with an overview of an AR. It then describes the FAS services provided for the FAS service users. Several AR types are then specified by their attributes. This specification also defines how the DLL services are used to transfer the FAS-PDUs. Protocol machines for different AR types are provided to describe operations and behaviour of the FAS in detail. Finally, the structure of the FAS-PDUs are defined.

Figure 1 shows the relationship between the FAS and the other communication and application components of the DD 238 Fieldbus architecture.



The FAS provides services to its service users, such as the FMS or APs, and uses the services of the DLL as shown in the figure above. In addition, for management purposes, a local interface to the FAS is provided by the FAS Layer Management Entity (FASLME).

### 3.2 Application Relationship

In a distributed system, Application Processes (APs) communicate with each other by exchanging application layer messages across well-defined application layer communication channels.

These communication channels are modeled in the Fieldbus Application Layer as Application Relationships (ARs).

ARs are responsible for conveying messages between or among APs according to specific communication characteristics required by time-critical systems. When the FMS is the service user, the messages that are conveyed contain confirmed and unconfirmed service requests and confirmed service responses.

Each AR is established by connecting two or more Application Relationship End Points (AREPs). Characteristics of an AR are determined by the types of the AREPs involved in the AR. Each AR has only one AREP for each involved AP.

The characteristics of ARs are described in the remainder of Section 3. Different combinations of these characteristics lead to the definition of different classes of ARs in Section 4.

### 3.3 AREP Characteristics

#### 3.3.1 AREP Context

To ensure communications compatibility among or between AREPs, each AREP definition contains a set of compatibility related characteristics. These characteristics shall be configured appropriately for each AREP for the AR to operate properly.

AREP definitions also contain a set of characteristics that describe the operation of the AR. These characteristics, when combined with those used to specify compatibility, define the context of the AREP. The AREP context is used by the FAS and the Network Management Entity to manage the operation of the AREP and the conveyance of FAS-PDUs. The characteristics which comprise the AREP context are described next.

#### 3.3.2 AREP Roles

The role of an AREP describes how an AP interacts with other APs involved in the AR. Only AREPs with compatible roles can be related to form an AR. Each AREP role is defined by the characteristics of service requests and responses it is capable of conveying.

The following AREP roles are defined: Client, Server, Peer, Publisher, Subscriber, Source, and Sink.

#### Client, Server, and Peer AREPs

Peer AREPs are those that can operate either as a Client or a Server AREP, or simultaneously as both. In the following descriptions, Client AREPs include Peer AREPs acting as Clients, and Server AREPs include Peer AREPs acting as Servers.

The following characteristics apply to Client and Server AREPs with regard to AR establishment and termination:

- A Client/Server AR is established over a peer-to-peer Data Link Connection (DLC).
- A Client AREP is able to send a request to establish an AR to a single Server AREP and to receive its response.
- A Client AREP may be configured to send such a request to either a prespecified Server AREP, or to any Server AREPs. The latter is designated as “Free” Client.
- A Server AREP can receive a request to establish an AR from a single Client AREP and to respond to it.
- A Server AREP may be configured to receive such a request from either a prespecified Client AREP, or from any Client AREPs. The latter is designated as “Free” Server.
- Either the Client or the Server AREP may request to terminate the AREP.

The following characteristics apply to Client and Server AREPs for conveying messages on an established AR:

- A Client AREP can send confirmed data transfer service requests and receive their responses.
- A Client AREP is not capable of receiving or responding to confirmed data transfer requests.
- A Client AREP is not capable of sending unconfirmed data transfer service requests.
- A Server AREP is capable of receiving confirmed data transfer service requests and returning their responses.

- A Server AREP is not capable of sending confirmed data transfer service requests or receiving their responses.
- A Server AREP is not capable of receiving unconfirmed data transfer service requests.

### **Publisher and Subscriber AREPs**

The following characteristics apply to Publisher and Subscriber AREPs with regard to AR establishment and termination:

- A Publisher/Subscriber AR is established over a multi-peer DLC.
- A Publisher AREP does not send a request to Subscriber AREPs to establish an AR. However, a Publisher AREP requests its DLL entity to establish a multi-peer DLC for sending data. Subscribers are not known to the Publisher.
- A Subscriber AREP accepts a user request to establish an AR on a pre-specified DLCEP.
- A Subscriber AREPs does not receive a request from a Publisher AREP or send a request to a Publisher AREP to establish an AR. However, it requests its DLL entity to join the DLC that has been established by the Publisher to receive data from it.
- A Publisher AREP sends a request to Subscriber AREPs to terminate the AR.
- Subscriber AREPs receive the request from the Publisher AREP to terminate the AR.
- A Subscriber AREP does not send a request to the Publisher AREP to terminate the AR. However, it requests its DLL entity to stop participating in the DLC that has been used to receive data from the Publisher.

The following characteristics apply to Publisher and Subscriber AREPs for conveying messages on an established AR:

- A Publisher AREP can send unconfirmed data transfer service requests.
- A Publisher AREP is not capable of sending confirmed data transfer service requests.
- A Publisher AREP is not capable of receiving confirmed or unconfirmed data transfer service requests.
- A Subscriber AREP is capable of receiving only unconfirmed data transfer service requests.
- A Subscriber AREP is not permitted to send confirmed or unconfirmed data transfer service requests.

### **Source and Sink AREPs**

The following characteristics apply to Source and Sink AREPs with regard to AR establishment and termination:

- A Source/Sink AR is established between two or more Data Link Layer Service Access Points (DLSAPs) and does not use a DLC.
- A Source AREP accepts a user request to establish an AR. However, it does not interact with its local DLL entity during AR establishment.
- A Sink AREP accepts a user request to establish an AR. However, it does not interact with its local DLL entity during AR establishment.
- A Source AREP accepts a user request to terminate the AR, but does not send a request to Sink AREPs to terminate the AR. It performs a local AREP termination only.
- A Sink AREP accepts a user request to terminate the AR, but does not send a request to the Source AREP to terminate the AR. It performs a local AREP termination only.

The following characteristics apply to Source and Sink AREPs for conveying messages on an established AR:

- A Source AREP can send unconfirmed data transfer service requests.
- A Source AREP is not capable of sending confirmed data transfer service requests.
- A Source AREP is not capable of receiving confirmed or unconfirmed data transfer requests.
- A Sink AREP is capable of receiving only unconfirmed data transfer service requests.
- A Sink AREP is not permitted to send confirmed or unconfirmed data transfer service requests.
- Flow control and lost FAS-PDU detection are done by the FAS user.

The following table summarizes the capabilities of each AREP role. A Peer AREP has the capabilities of Client and Server AREPs at the same time and is not included in the table.

**Table 1 — Characteristics of AREP Roles**

| AREP Role  | Confirmed Service Requests | Confirmed Service Responses | Unconfirmed Service Requests |
|------------|----------------------------|-----------------------------|------------------------------|
| Client     | Issue                      | Receive                     | No                           |
| Server     | Receive                    | Issue                       | No                           |
| Publisher  | No                         | No                          | Issue                        |
| Subscriber | No                         | No                          | Receive                      |
| Source     | No                         | No                          | Issue                        |
| Sink       | No                         | No                          | Receive                      |

### 3.3.3 AREP Cardinality

The cardinality of an AREP indicates the number of remote AREPs with which it simultaneously communicates. The table below summarizes the cardinality of AREPs by its role.

**Table 2 — AREP Cardinality by AREP Type**

| AREP Role  | Permitted Cardinality | DLL Address Type          | Meaning                                                                               |
|------------|-----------------------|---------------------------|---------------------------------------------------------------------------------------|
| Client     | 1                     | Individual DLCEP          | A Client AREP shall be related to a single Server or Peer AREP in a single AR.        |
| Server     | 1                     | Individual DLCEP          | A Server AREP shall be related to a single Client or Peer AREP in a single AR.        |
| Peer       | 1                     | Individual DLCEP          | A Peer AREP shall be related to a single Client, Server, or Peer AREP in a single AR. |
| Publisher  | $\geq 0$              | Individual DLCEP          | A Publisher AREP shall be related to zero or more Subscriber AREPs in a single AR.    |
| Subscriber | 1                     | Individual DLCEP          | A Subscriber shall be related to a single Publisher AREP in a single AR.              |
| Source     | $\geq 0$              | Individual DLSAP          | A Source AREP shall be related to zero or more Sink AREPs in a single AR.             |
| Sink       | 1                     | Individual or Group DLSAP | A Sink AREP shall be related to a single Source AREP in a single AR.                  |

### 3.3.4 AR Cardinality

The cardinality of an AR indicates the number of AREPs communicating with each other in the AR. The table below summarizes the cardinality of ARs.

**Table 3 — AR Cardinality by AR Type**

| AR Type              | Permitted Cardinality | Meaning                               |
|----------------------|-----------------------|---------------------------------------|
| Client/Server        | 1-to-1                | 1 Client, 1 Server                    |
| Publisher/Subscriber | 1-to-Many             | 1 Publisher, zero or more Subscribers |
| Source/Sink          | 1-to-Many             | 1 Source, zero or more Sinks          |

### 3.3.5 Conveyance Path

A conveyance path of an AREP is a conceptual channel in an AR through which FAS-PDUs are transferred from one AREP to another. It is a one-way communication path between any two AREPs that send or receive FAS-PDUs.

AREPs that only send or receive FAS-PDUs are modeled with having one conveyance path, and those that do both are modeled with having two of them. An AR with a single conveyance path is called a uni-directional AR, and one with two conveyance paths is called a bi-directional AR.

### 3.3.6 Trigger Policy

Trigger policy of an AREP specifies when FAS-PDUs are transmitted by the local DLL. Two types of trigger policy are defined for the FAS.

The first type of trigger policy is referred to as user-triggered. User-triggered AREPs do not use the scheduling capability of the DLL. The DLL is configured so that FAS-PDUs are transmitted as soon as possible after they are received from the FAS. User-triggered AREPs do not support the timeliness aspects described later.

The second type of trigger policy is called network-scheduled. Network-scheduled AREPs use scheduling capability of the DLL. FAS-PDUs are transmitted to the network regardless of when they are received from the FAS user. The DLL is configured to transmit FAS-PDUs at a specified frequency. In addition, FAS-PDUs may be transmitted to the network sporadically when the local DLL is explicitly requested by the local or remote FAS users.

### 3.3.7 Conveyance Policy

Conveyance policy of an AREP indicates whether FAS-PDUs are transferred according to DL's buffer and queue models. These models describe the method of conveying FAS-PDUs from a sender to a receiver.

*NOTE Although this specification makes use of the buffer and queue models of the DLL, it does not specify whether additional buffers or queues are implemented in the FAS or not.*

In the case of an AREP with buffered conveyance policy, updates to its buffer replace its contents with a new FAS-PDU. Thus, unconveyed or undelivered FAS-PDUs that have been replaced in the buffer are lost. Additionally, a buffer may be read multiple times without destroying the FAS-PDU contained in it. Also, the same FAS-PDU may be sent multiple times if the buffer is not updated by the FAS. When this occurs, the receiving DLL entity indicates to the FAS that the receiving buffer has been updated, but the update was a duplicate of the previous data. The FAS will make this information available to its user so that the user may detect "staleness" of the received data.

In the case of an AREP with queued conveyance policy, it conveys FAS-PDUs using a first-in first-out (FIFO) queue. A queue entry is not overwritten. New FAS-PDUs are appended to a queue until it becomes full. The DLL sends FAS-PDUs in the order they are put into the queue. Similar, but reversed actions take place at a receiving AREP. If a queue is full, a new FAS-PDU will not be placed into the queue.

### 3.3.8 Identification of AREPs

An AP distinguishes an AR by its local identifier.

The remote AREP is known to the local AREP only through configuration of the remote DLSAP-address for Source and Sink AREPs, or through configuration of the remote DLCEP-address for Client, Server, Peer, and Subscriber AREPs. Publisher AREPs do not know the identity of Subscriber AREPs.

The only exception to this rule is for AREPs that are configured to be "Free." For "Free" Client and Server AREPs, the remote address is not configured, but is dynamically determined when the AR is established. For "Free" Sink and Source AREPs, the remote address is provided by an FAS-PDU.

## 3.4 AR Establishment

Each AR is established by connecting two or more AREPs of the same type. How messages are conveyed on an AR is dependent on the types of the AREPs involved in the AR. AR establishment can occur in one of three ways:

First, ARs can be pre-established. Pre-established means that the AREP context is created when an AP is connected to the network. In this case, communications among the APs involved in the AR take place without first having to explicitly establish the AR. Any AR may be defined to be pre-established.

*NOTE When pre-established ARs are used, the state machines that deal with AR establishment and release do not apply. However, those that deal with data transfer still apply. A means to bring the state machines to the data transfer phase is a local matter.*

Second, ARs can be pre-configured, but not established. Pre-configured means that the characteristics of the AR are known to each AREP, but that their contexts have not been synchronized for operation. In this case, the use of the FAS Associate service is required to activate the AR.

Third, ARs can require dynamic definition and establishment. In this case, AREP definitions have to be created for each of the AREPs involved in the AR. Network Management services may be used to remotely create AREP definitions. After creation, ARs can be established as though they were pre-configured.

Only Client/Server AR establishment results in the exchange of FAS-PDUs. During this exchange, DLCEP-addresses are used as the global identifiers for the Client/Server AREPs. These FAS-PDUs are transferred between node-local DLSAPs reserved for this purpose by the DD 238 Field No.

3.5 Underlying Communications Services

The FAS conveys FAS-PDUs using the capabilities of the DLL. This section briefly describes them for better understanding. Formal definitions of these characteristics are specified in the DD 238 Data Link Layer specification.

3.5.1 Connectionless and Connection-Oriented Services

The DLL supports AREPs by providing connectionless and connection-oriented mode of services.

3.5.2 Buffer and Queue

The DLL provides services to configure and control its buffers and queues. These services are used by the Network Management to configure AREPs.

3.5.3 Cyclic and Acyclic Services

The DLL supports AREPs by providing cyclic or acyclic services. Cyclic services are used to support timeliness requirements of AREPs.

User triggered AREPs are always acyclic. Network triggered AREPs are either cyclic or acyclic. A Cyclic AREP implies that scheduling is done by a scheduling mechanism of the DLL.

3.5.4 Quality of Service

The DLL may provide more than one quality of service. When this occurs, the AREP shall be configured to use the quality of service appropriate for the FAS-PDUs it conveys. Quality of service parameters for AREPs are contained in the Data Link Mapping section of this specification.

3.5.5 Segmentation

The FAS does not segment user data it conveys. However, if the DLL provides segmentation, the maximum length of data conveyed by the FAS in a single FAS-PDU will increase accordingly.

3.5.6 Timeliness

The FAS supports the timeliness parameters provided by the DLL. It does not control any of the parameters; rather, it passes through these parameters from the DLL to its user.

4 FAS Model

The FAS model assumes that one or more ARs exist between APs. Each AR connects one or more APs based on its cardinality.

4.1 Application Relationship End Point Classes

AREP classes are defined with a combination of characteristics to form different types of ARs. For purposes of interoperability, this specification defines three AREP classes, from which a basic set of standardized ARs can be derived. These classes are:

- QUU: Queued User-triggered Unidirectional AREP Class
- QUB: Queued User-triggered Bidirectional AREP Class
- BNU: Buffered Network-Scheduled Unidirectional AREP Class

The following table summarizes AREP classes and supported characteristics described in the Architecture section:

Table 4 — AREP Classes and Characteristics

|                     | AREP Classes   |                      |                       |
|---------------------|----------------|----------------------|-----------------------|
|                     | QUU            | QUB                  | BNU                   |
| Permitted Roles     | Source, Sink   | Client, Server, Peer | Publisher, Subscriber |
| Conveyance Paths    | 1              | 2                    | 1                     |
| Conveyance Policy   | Queued         | Queued               | Buffered              |
| Transmission Policy | User Triggered | User Triggered       | Network Scheduled     |

4.1.1 QUU (Queued User-triggered Unidirectional) AREP Class

This AREP class provides an AR that supports on-demand queued distribution of unconfirmed services from one AP to zero or more APs, such as one with Source/Sink interactions.

A Source AREP accepts an unconfirmed service request, incorporates it into the appropriate FAS-PDU, and submits the FAS-PDU to the DLL. The DLL is configured to provide queued, connectionless data transfer services according to the attribute definitions of the AREP that are specified in the later section. All the FAS-PDUs from the Source AREP are sent with the same quality of service provided by the DLL. A Sink AREP receives the FAS-PDU from the DLL and delivers the unconfirmed service indication in the order it is received to the FAS user associated with this AREP.

#### **4.1.2 QUB (Queued User-triggered Bidirectional) AREP Class**

This AREP class provides an AR that supports an on-demand exchange of confirmed services between two APs.

A Client AREP accepts a confirmed service request, incorporates it into the appropriate FAS-PDU, and submits the FAS-PDU to the DLL. The DLL is configured to provide queued, connection-oriented data transfer services according to the attribute definitions of the AREP that are specified in the later section. The communication characteristics defined for the AREP determines how the DLL is configured. All the FAS-PDUs from the Client AREP are sent with the same quality of service provided by the DLL. A Server AREP receives the FAS-PDU from the DLL and delivers the confirmed service indication in the order received to the AP associated with this AREP.

The Server AREP accepts a confirmed service response from its user, incorporates it into the appropriate FAS-PDU, and submits the FAS-PDU to the DLL. The DLL provides queue-oriented, connection-oriented data transfer services according to the attribute definitions of the AREP that are specified in the later section. All the FAS-PDUs from the Server AREP are sent with the same quality of service provided by the DLL. The Client AREP receives the FAS-PDU that contains a confirmed service response from the DLL, delivers the confirmed service confirmation to the AP associated with this AREP, and completes the confirmed service.

#### **4.1.3 BNU (Buffered Network-Scheduled Unidirectional) AREP Class**

This AREP class provides an AR that supports scheduled, buffered distribution of unconfirmed services to zero or more APs, such as one with Publisher/Subscriber interactions.

A Publisher AREP accepts an unconfirmed service request, incorporates it into the appropriate FAS-PDU, and submits the FAS-PDU to the DLL. The DLL provides buffered, connection-oriented data transfer services according to the attribute definitions of the AREP that are specified in the later section. The communication characteristics defined for the AREP determines how the DLL is configured. All the FAS-PDUs from the Publisher AREP are sent with the same quality of service provided by the DLL. A Subscriber AREP receives the FAS-PDU from the DLL and delivers the unconfirmed service indication in the order it is received to the AP associated with this AREP.

If the Publisher AREP receives another unconfirmed service request before the FAS-PDU that contains the previous service request is transmitted, the previous FAS-PDU will be replaced with the new one, and the previous FAS-PDU will be lost. Similarly, if the Subscriber AREP receives another FAS-PDU before the previous one is read by its user, the previous one will be replaced by the new one and will be lost.

If the Publisher AREP does not receive a new unconfirmed service request before the DLL is triggered to transmit the buffer content, the same FAS-PDU will be retransmitted. If the Subscriber AREP receives the same FAS-PDUs successively, it notifies the user that duplicate FAS-PDUs have been received.

## **5 Operational Services**

The FAS provides a set of services for its higher layer protocol. The Associate (ASC) and Abort (ABT) services control the establishment and release of ARs. The Data Transfer Confirmed (DTC) service provides a bi-directional exchange of higher layer services. It conveys confirmed higher layer services. The Data Transfer Unconfirmed (DTU) service provides a uni-directional data transfer and is used to convey unconfirmed higher layer services. The FAS-Compel (FCMP) service provides a means to generate “out of schedule” transmission from a scheduled buffer. The Get-Buffered-Message (GBM) service allows the FAS user to retrieve the contents of a buffer. Finally, the FAS-Status (FSTS) service reports several DLL events to the FAS user.

This section describes the FAS services, their parameters, and the service sequences at the interface between the FAS user and the FAS. The interface to the DLL is described later.

The following are the summary of FAS services at the interface to the FAS user:

- Associate (ASC) — establishment of an AR

- Abort (ABT) — release of an AR
- Data Transfer Confirmed (DTC) — data transfer confirmed by the user
- Data Transfer Unconfirmed (DTU) — unconfirmed data transfer
- FAS-Compel (FCMP) — a “send-buffer” request to the DLL
- Get-Buffered-Message (GBM) — retrieve an FAS-Service Data Unit (SDU) from the DLL
- FAS-Status (FSTS) — a reported event from the DLL

An FAS user which issues a service request to the FAS is called a “requester.” An FAS user which issues a service response to the FAS is called a “responder.” An FAS user which receives a service indication (for unconfirmed FAS user services) from the FAS is called the “receiver.”

## 5.1 Conventions

### 5.1.1 Parameters

The service primitives, used to represent service user/service provider interactions, convey parameters that indicate information available in the user/provider interaction.

This specification uses a tabular format to describe the component parameters of the FAS service primitives. Each table consists of up to five columns, containing the name of the service parameter, and one column each for the primitives used by the FAS service:

- the request primitive’s parameters,
- the indication primitive’s parameters,
- the response primitive’s parameters, and
- the confirm primitive’s parameters.

One parameter (or part of it) is listed in each row of each table. Under the appropriate service primitive columns, a code is used to specify the type of usage of the parameter on the primitive:

- “M” the parameter is mandatory for the primitive.
- “U” the parameter is a User option, and may or may not be provided depending on the dynamic usage of the FAS-user.
- “C” the parameter is conditional upon other parameters or upon the configuration of the AREP.
- “S” the parameter is a selection, and one of the parameters at the same indent level in the parameter column shall be selected.

If the above code is followed by “(=),” it indicates that the value of the parameter is equal to the parameter in the service primitive to its immediate left in the table.

### 5.1.2 General conventions

The service model, service primitives, and time-sequence diagrams used are entirely abstract descriptions; they do not represent a specification for implementation.

## 5.2 Associate (ASC)

The Associate (ASC) service is used to establish an AR and optionally transmit FAS user data.

**Table 5 — Associate (ASC)**

| Parameter Name       | .request | .indication | .response | .confirm |
|----------------------|----------|-------------|-----------|----------|
| Argument             | M        | M           |           |          |
| AREP Identifier      | M        | M           |           |          |
| Remote DLCEP Address | C        |             |           |          |
| Data                 | U        | U(=)        |           |          |
| Result(+)            |          |             | S         | S        |
| AREP Identifier      |          |             | M         | M        |
| Data                 |          |             | U         | U(=)     |
| Result(–)            |          |             | S         | S        |
| AREP Identifier      |          |             | M         | M        |
| Data                 |          |             | U         | U(=)     |

### 5.2.1 Argument

The Argument contains the parameters of the ASC.request and ASC.indication primitives.

#### AREP Identifier

This parameter contains sufficient information to locally identify the AR to be used to convey the service.

#### Data

This optional parameter contains user data.

#### Remote DLCEP Address

This conditional parameter is present if the RemoteAddressConfigurationType attribute of the associated AREP is Free. If it is present, this parameter carries the remote DLCEP address.

If the RemoteAddressConfigurationType attribute is Linked, this parameter is not present.

### 5.2.2 Result(+)

This parameter indicates that the remote user agreed to establish the AR.

#### AREP Identifier

This parameter contains sufficient information to locally identify the AR to be used to convey the service.

#### Data

This optional parameter contains user data.

### 5.2.3 Result(-)

This parameter indicates that the remote user rejected the Associate request.

#### AREP Identifier

This parameter contains sufficient information to locally identify the AR to be used to convey the service.

#### Data

This optional parameter contains user data.

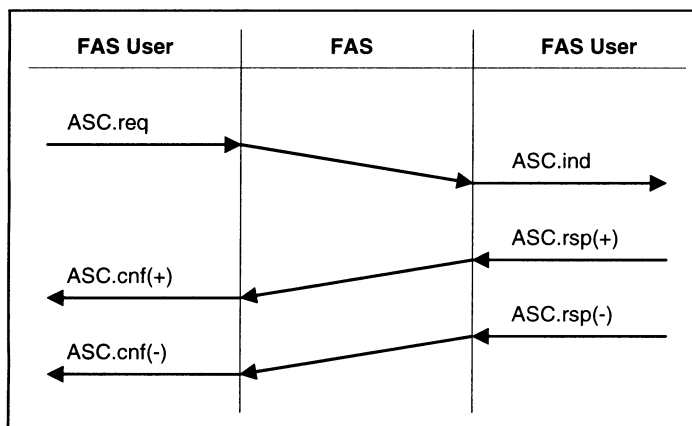
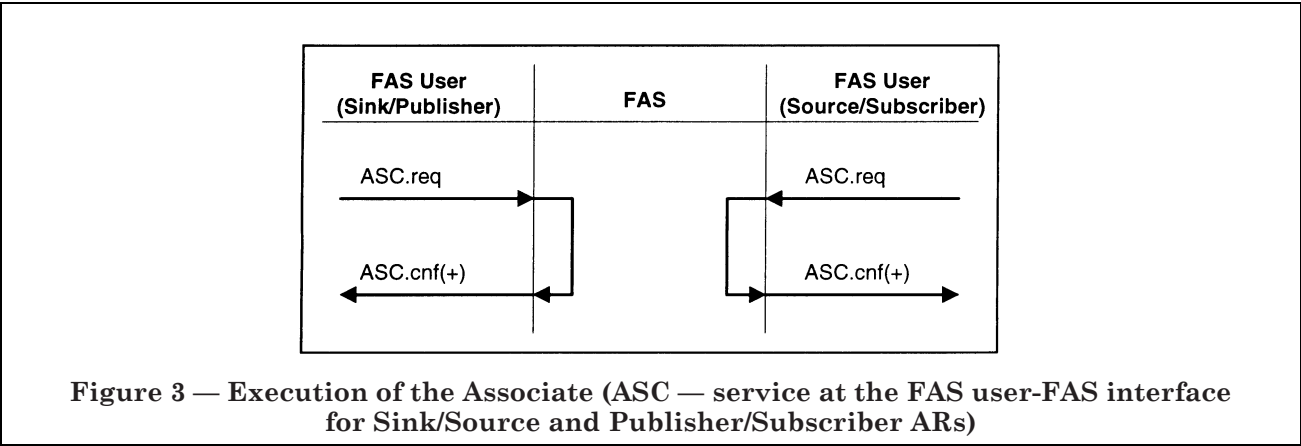


Figure 2 — Execution of the Associate (ASC — service at the FAS user-FAS interface for Client/Server ARs)



5.3 Data Transfer Confirmed (DTC)

The Data Transfer Confirmed service (DTC) is used to transmit confirmed FAS user service requests over an AR. An AR is a prerequisite for the execution of the DTC service.

Table 6 — Data Transfer Confirmed Service

| Parameter Name  | .request | .indication | .response | .confirm |
|-----------------|----------|-------------|-----------|----------|
| Argument        | M        | M           |           |          |
| AREP Identifier | M        | M           |           |          |
| Data            | M        | M(=)        |           |          |
| Result          |          |             | M         | M        |
| AREP Identifier |          |             | M         | M        |
| Data            |          |             | M         | M(=)     |

5.3.1 Argument

The argument contains the parameters of the DTC.request and DTC.indication primitives.

AREP Identifier

This parameter contains sufficient information to locally identify the AR to be used to convey the service.

Data

This parameter contains a confirmed user request.

5.3.2 Result

This parameter indicates that the DTC service has been executed.

AREP Identifier

This parameter contains sufficient information to locally identify the AR to be used to convey the service.

Data

This parameter contains a confirmed user response.

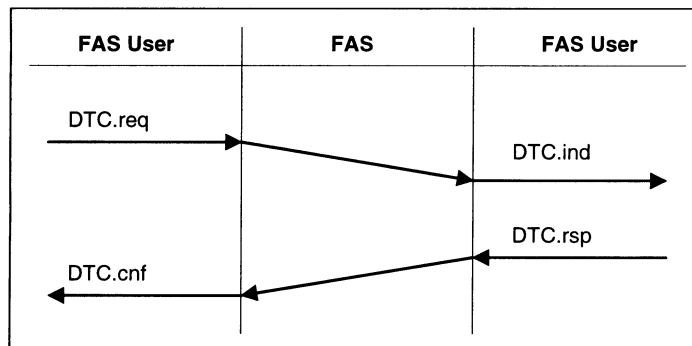


**Figure 4 — Execution of the Data Transfer Confirmed (DTC — service at the FAS user- FAS interface)**

#### 5.4 Data Transfer Unconfirmed (DTU)

The Data Transfer Unconfirmed service is used to transmit an unconfirmed FAS user service requests over a specified AR. An AR is a prerequisite for the execution of the DTU service.

**Table 7 — Data Transfer Unconfirmed Service**

| Parameter Name       | .request | .indication |
|----------------------|----------|-------------|
| Argument             | M        | M           |
| AREP Identifier      | M        | M           |
| Remote DLSAP Address | C        | C           |
| Duplicate FAS-SDU    |          | C           |
| Data                 | M        | M(=)        |
| Local Timeliness     |          | C           |
| Remote Timeliness    |          | C           |

##### 5.4.1 Argument

The argument contains the parameters of the DTU.request and DTU.indication primitives.

##### AREP Identifier

This parameter contains sufficient information to locally identify the AR to be used to convey the service.

##### Remote DLSAP Address

This conditional parameter is present if the ConfigurationType attribute of the associated AREP is Free. If it is present, this parameter carries the DLSAP address associated with the remote AREP. If the ConfigurationType attribute is Linked, this parameter is not present.

##### Duplicate FAS-SDU

This conditional parameter is present if the DuplicatePduDetectionSupported attribute is True. If it is present, it indicates whether or not the FAS-SDU is a duplicate of the FAS-SDU that has been received and delivered in a previous service indication.

This parameter takes a value of True if the same FAS-SDUs have been received successively. Otherwise, it takes a value of False.

##### Data

This parameter contains an unconfirmed service request.

##### Local Timeliness

This conditional parameter, if it is present and True, indicates that the FAS-SDU parameter has met the receiving timeliness criteria defined for the DLL. If its value is False, at least one of the timeliness criteria has not been met.

This parameter is present if the value of theSubscriberD1TimelinessClass attribute is other than None and is not present if it is None.

**Remote Timeliness**

This conditional parameter, if it is present and True, indicates that the FAS-SDU parameter has met the Publisher’s and transmitting DL’s timeliness criteria. If its value is False, at least one of the timeliness criteria has not been met.

This parameter is present if the value of thePublisherD1TimelinessClass attribute is other than None and is not present if it is None.

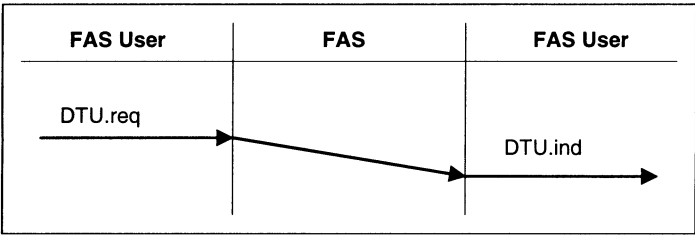


**Figure 5 — Execution of the Data Transfer Unconfirmed (DTU — service at the FAS user- FAS interface)**

**5.5 Abort (ABT)**

The Abort service releases the specified AR. Execution of the DTC and DTU services after an Abort request has been issued is not permitted until the AR is re-established. This service may be requested by the FAS user or internally generated by the FAS.

**Table 8 — Abort**

| Parameter Name    | .request | .indication |
|-------------------|----------|-------------|
| Argument          | M        | M           |
| AREP Identifier   | M        | M           |
| Locally Generated |          | M           |
| Identifier        | M        | M(=)        |
| Reason Code       | M        | M(=)        |
| Additional Detail | U        | U(=)        |

**5.5.1 Argument**

The argument contains the parameters of the ABT.request and ABT.indication primitives.

**AREP Identifier**

This parameter contains sufficient information to locally identify the AR to be used to convey the service. This parameter may use a key attribute of the AREP.

**Locally Generated**

This parameter indicates whether the Abort service was initiated at the local or remote AREP.

**Identifier**

This parameter indicates where the reason for the connection abort was identified (user, FAS user, FAS or DLL).

*NOTE In cases when the ABT.request or ABT.indication is generated by the FAS, the Additional Detail parameter does not exist. If the ABT.indication is generated by the local FAS, the Identifier and Reason Code parameters are set locally.*

**Reason Code**

This parameter specifies the reason for the AR abort.

**Additional Detail**

This optional parameter contains additional information about the connection release.

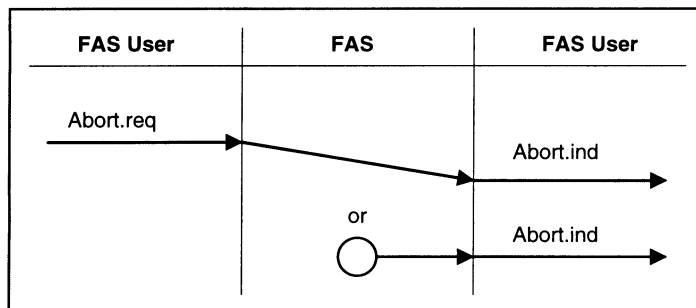


Figure 6 — Execution of the Abort service at the FAS user-FAS interface

### 5.6 FAS-Compel (FCMP)

The FAS-Compel service is a local confirmed service used by an FAS user to request the DLL to transmit the message which has been stored in the buffer. It is only used with the BNU AREP class. The structure of the component service primitives is shown below:

Table 9 — FAS-Compel Service

| Parameter Name  | .request | .confirm |
|-----------------|----------|----------|
| Argument        | M        |          |
| AREP Identifier | M        |          |
| Result          |          | M        |
| Status          |          | M        |

#### 5.6.1 Argument

The argument carries the parameters of the service request.

#### AREP Identifier

This parameter identifies the AREP to which the buffer that contains the message being delivered is attached.

#### 5.6.2 Result

This parameter carries the service confirmation.

#### Status

This parameter indicates the result of the service request. The following three status codes provided by the DLL are defined:

- success
- failure — inappropriate request
- failure — reason unspecified

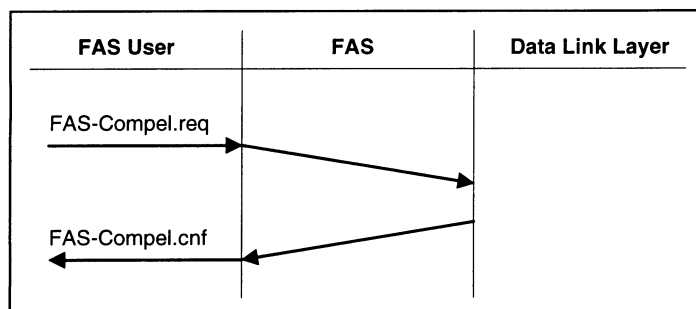


Figure 7 — Time Sequence Diagram for the FAS-Compel Service

### 5.7 Get-Buffered-Message (GBM)

The GetBuffered-Message service is a local confirmed service used by an FAS user to request the DLL to deliver the message which has been maintained in the buffer. It is used with the BNU AREP class only.

**Table 10 — Get-Buffered-Message Service**

| Parameter Name    | .request | .confirm |
|-------------------|----------|----------|
| Argument          | M        |          |
| AREP Identifier   | M        |          |
| Result(+)         |          | S        |
| AREP Identifier   |          | M        |
| Duplicate FAS-SDU |          | C        |
| Data              |          | M        |
| Local Timeliness  |          | C        |
| Remote Timeliness |          | C        |
| Result(–)         |          | S        |

*NOTE* The maximum outstanding requests for the Get-Buffered-Message service is always one; i.e., the FAS users need to wait for the service confirmation before they issue another service request.

#### 5.7.1 Argument

The argument carries the parameters of the service request.

##### AREP Identifier

This parameter identifies the AREP to which the buffer that contains the message is attached.

#### 5.7.2 Result(+)

The Result(+) parameter indicates that the service request succeeded.

##### AREP Identifier

This parameter identifies the AREP to which the buffer that contains the message is attached.

##### Duplicate FAS-SDU

This conditional parameter is present if the DuplicatePduDetectionSupported attribute is True. If it is present, it indicates whether or not the FAS-SDU is a duplicate of the FAS-SDU that was received and delivered in a previous service indication.

This parameter takes a value of True if the same FAS-SDUs have been received successively. Otherwise, it takes a value of False.

##### Data

This parameter contains an unconfirmed user request.

##### Local Timeliness

This conditional parameter, if it is present and True, indicates that the FAS-SDU parameter has met the receiving timeliness criteria defined for the DLL. If its value is False, at least one of the timeliness criteria has not been met.

This parameter is present if the value of theSubscriberD1TimelinessClass attribute is other than None and is not present if it is None.

##### Remote Timeliness

This conditional parameter, if it is present and True, indicates that the FAS-SDU parameter has met the Publisher's and transmitting DL's timeliness criteria. If its value is False, at least one of the timeliness criteria has not been met.

This parameter is present if the value of thePublisherD1TimelinessClass attribute is other than None and is not present if it is None.

#### 5.7.3 Result(–)

The Result(–) parameter indicates that the service request failed. When this parameter is returned, it means that the buffer attached to the specified AREP does not contain any FAS-PDUs.

### AREP Identifier

This parameter identifies the AREP to which a buffer is attached.

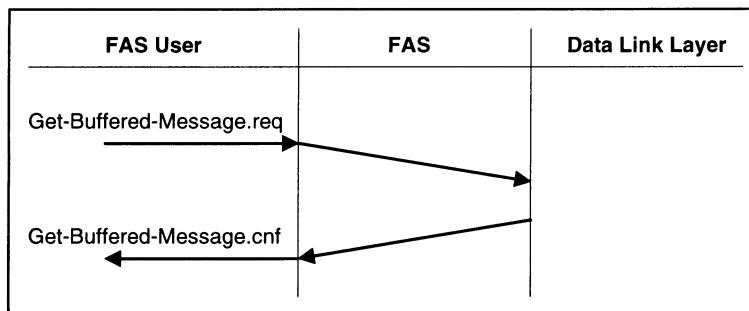


Figure 8 — Time Sequence Diagram for the Get Buffered Message Service

### 5.8 FAS-Status (FSTS)

The FAS-Status service provides the FAS user notification of a buffer-sent event from the DLL. It is used with the BNU AREP class only.

The structure of the component service primitives is shown below:

Table 11 — FAS-Status

| Parameter Name  | .indication |
|-----------------|-------------|
| Argument        | M           |
| AREP Identifier | M           |
| Reported Status | M           |

#### 5.8.1 Argument

This parameter carries the parameters of the service indication.

#### AREP Identifier

The argument specifies the AREP from which a DLL event is reported.

#### Reported Status

This parameter specifies the event which is reported from the DLL. The following event is specified:

Buffer-Sent: The DLL reported that the buffer content had just been sent.

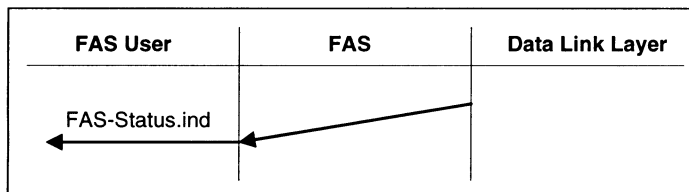


Figure 9 — FAS-Status Service

## 6 AREP Class Specifications

This section contains the AREP class definitions. Each definition is composed of attribute and service definitions for the class.

*NOTE All the AREP attribute values shall be specified prior to AREP operation.*