Data Compression Over Frame Relay Implementation Agreement

FRF.9

Frame Relay Forum Technical Committee

January 22, 1996

Note: The user's attention is called to the possibility that implementation of the frame relay implementation agreement contained herein may require the use of inventions covered by patent rights held by third parties. By publication of this frame relay implementation agreement, the Frame Relay Forum makes no representations that the implementation of the specification will not infringe on any third party rights. The Frame Relay Forum takes no position with respect to any claim that has been or may be asserted by any third party, the validity of any patent rights related to such claims, or the extent to which a license to use such rights may or may not be available.

For additional information contact:

The Frame Relay Forum

Worldwide Headquarters

303 Vintage Park Drive

Foster City, CA 94404-1138

Telephone: +415-578-6980

Fax: +415-525-0182

frf@sbexpos.com

Editor:

Don Cantwell
Motorola, Inc.
20 Cabot Boulevard
Mansfield, MA USA 02048-1193
Tel: 508-261-5525
FAX: 508-337-7173
Internet: LDC001@email.mot.com

Table of Contents

List of Figures

List of Tables

Revision History

First Release of Data Compression over Frame Relay
Implementation Agreement. FRF.9 . . . . January 22, 1996

1 INTRODUCTION

1.1 Purpose

This document is the Data Compression over Frame Relay Implementation Agreement. The agreements herein were reached in the Frame Relay Forum, and are based on the relevant frame relay standards referenced in Section 1.7. They document agreements reached among vendors and suppliers of frame relay products and services regarding the options to be implemented.

This agreement may be submitted to different bodies involved in ratification of implementation agreements and conformance testing to facilitate multi-vendor interoperation.

1.2 Overview

This document describes the encapsulation of the Data Compression Protocol (DCP) over frame relay, the Data Compression Protocol format and operation, and the default Data Compression Function Definition (DCFD). It applies to un-numbered information (UI) frames encapsulated using Q.933 Annex E [9] and FRF.3.1 [3]. It addresses data compression on both permanent virtual connections (PVC) and switched virtual connections (SVC) as qualified in Section 1.6. It is compatible with the network interworking function described in FRF.5 [4] and may be used on frame relay connections that are interworked with ATM using FRF.5. The use of data compression on frame relay connections that are interworked with ATM using service interworking (FRF.8 [5]) is defined in Annex B.

DCP is logically decomposed into two sublayers: the DCP Control sublayer, and the DCP Function sublayer.

The DCP Control sublayer provides the following services:

- encapsulation of encoded user data and negotiation primitives within DCP protocol data units (DCP PDU’s) for transport between DCP peers.
- negotiation of DCP configuration options, including: negotiation of optional DCP formats and procedures, and negotiation of optional DCP Functions (algorithms) and parameters.
- synchronization of the sender and receiver peers, including: detection of loss of synchronization and signalling for resynchronization between DCP peers.
- anti-expansion protection, including: compressed/uncompressed mode signalling from the encoder to the peer decoder.
- identification of DCP contexts.

The DCP Function sublayer provides the following services:

- encoding of user data into compressed user data according to one or more of a variety of public or proprietary algorithms.
- decoding of compressed user data into uncompressed user data.

DCP supports the negotiation of optional public or proprietary DCP Functions. Each DCP Function sublayer is defined by a separate DCFD document that, in combination with this document, is sufficient to assure interoperability of DCP among manufacturers providing the same DCP Function.

A DCFD includes: encoding and decoding procedures, extensions to the DCP PDU defined in this document, loss-of-synchronization detection, and resynchronization procedures, etc. See Section 5 for a detailed description of the contents of a DCFD. A default DCFD is provided by Annex A of this IA. Support of Annex A shall be required by an implementation compliant with this IA.

1.3 Terminology

- **Must, Shall, or Mandatory** - the item is an absolute requirement of this implementation agreement.
- **Should** - the item is highly desirable.
- **May or Optiona**l - the item is not compulsory, and may be followed or ignored according to the needs of the implementor.
- **Not Applicable** - the item is outside the scope of this implementation agreement.

1.4 Definitions

- Anti-expansion - A means to inhibit the expansion of user data due to compression encoding.
- Data Compression Context - A vocabulary and other information for error detection and synchronization, created and maintained by DCP peers to encode/decode user data.
- Data Compression Context Identifier - An identifier that addresses a DCP data PDU to a particular data compression context.
- Data Compression Function - An entity that performs the data compression encoding, decoding, error detection, synchronization, and negotiation.
- Data Compression Function Definition - A specification that describes the format and procedures used by a data compression function to transport user data and control primitives.
- Data Compression Protocol - The protocol defined in this IA to carry user data and control primitives.
- DCP Control Protocol - The protocol used for DCP negotiation.
- DCP Header - The first octet of the Data Compression Protocol.
- DCP Payload - The data carried by a DCP data PDU as defined in a DCFD.
- Decoder - An entity that decompresses user data.
- Encoder - An entity that compresses user data.
- History Buffer - The type of vocabulary used for X3.241 compression.
- Mode-1 - A simplified negotiation mode for DCP.
- Mode-2 - A full negotiation mode for DCP.
- Receiver - An entity that implements a data compression decoder and the receiving side of the Data Compression Protocol.
- Sender - An entity that implements a data compression encoder and the sending side of the Data Compression Protocol.

1.5 Acronym List

- CCP - Compression Control Protocol
- C/D - Control/data
- C/U - Compressed/uncompressed
- DC - Data Compression
- DCCI - Data compression context identifier
- DCFD - Data Compression Function Definition
- DCP - Data Compression Protocol
- DCPCP - DCP Control Protocol
- E - Extension
- IA - Implementation Agreement
- IPI - Initial Protocol Identifier
- LCB - Longitudinal Check Byte
- LCP - Link Control Protocol
- NLPID - Network Layer Protocol Identifier
- OUI - Organization Unique Identifier
- PDU - Protocol Data Unit
- PVC - Permanent Virtual Connection
- R-A - Reset acknowledge
- R-R - Reset request
- SVC - Switched Virtual Connection

1.6 Qualifications

This IA only applies to Q.933 Annex E frames that use an Un-numbered Information (UI) control field. This IA does not cover frames that use a Number Information (I) control field.

The frame relay signalling messages of Q.933 currently do not support signalling of DCP. A SVC is established using the multiprotocol encapsulation codepoint (ISO/IEC DTR9577). The data compression usage is then negotiated using DCP. This implies that following DCP negotiation where compression is disabled, a connection that requires the use of data compression may be released, or may remain active with compression disabled. If the connection remains active with compression disabled, the traffic parameters requested for the compressed call may not be sufficient. In the future, the appropriate codepoints and cause codes must be added to Q.933 to signal the use of DCP.

Note: The Data Compression Protocol formats and procedures in this agreement are currently undergoing standardization within the ANSI and ITU bodies. Changes to these formats and procedures may result from the standardization process.

1.7 Relevant Standards

[1] ANSI/IEEE Std 802-1990, IEEE Standards for Local and Metropolitan Area Networks: Overview and Architecture

[2] ANSI X3.241-1994 Data Compression Method - adaptive coding with sliding window for information interchange

[3] FRF.3.1 - Multiprotocol Encapsulation Implementation Agreement

[4] FRF.5 - Frame Relay / ATM PVC Network Interworking Implementation Agreement

[5] FRF.8 - Frame Relay / ATM PVC Service Interworking Implementation Agreement

[6] ISO/IEC/TR 9577:1992 Information processing systems - Telecommunications and information exchange between systems - Protocol identification in the network layer

[7] ITU Q.922 - Digital subscriber signalling system no 1 (DSS 1) - ISDN Data Link Specification for Frame Mode Bearer Services

[8] ITU Q.931 - Digital subscriber signalling system no 1 (DSS 1) - User-Network Interface Layer 3 Specification for Basic Call Control

[9] ITU Q.933 - Digital subscriber signalling system no 1 (DSS 1) - Signalling Specification for Frame Mode, Basic Call Control

[10] RFC1661 - “The Point-to-Point Protocol (PPP)”

[11] TIA/EIA 655 - Support for Terminal Adaption and Data Compression in Data Circuit-Terminating Equipment (DCE) with provisions for negotiation of parameters.

2 DATA COMPRESSION ENCAPSULATION

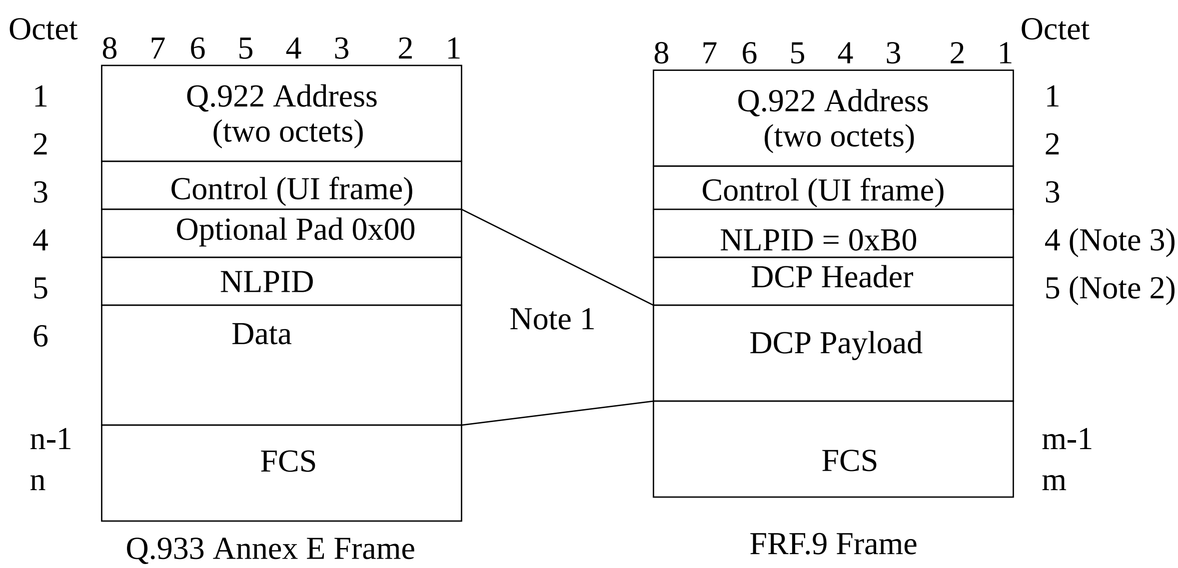
This section describes the encapsulation method for the Data Compression Protocol (DCP) with Multiprotocol Encapsulation over a Frame Relay network using Q.933 Annex E.

DCP shall be encapsulated within a Q.922 Annex A [7] frame with a Network Layer Protocol Identifier (NLPID) equal to Data Compression (0xB0) indicating that a DCP PDU follows. A DCP PDU is the combination of a DCP Header and DCP Payload or DCPCP PDU as defined in Section 3. The DCP layer coexists with the current Q.933 Annex E as another NLPID. The same virtual connection can concurrently transport multi-protocol encapsulated data that uses the DCP layer (NLPID equal to 0xB0) as well as multiprotocol encapsulated data that does not use the DCP layer (NLPID not equal to 0xB0). The ordering of frames sent on a virtual connection with and without the DCP layer and using the same IPI must be maintained.

Throughout this IA, the octets are sent in ascending order; inside an octet, bit 1 is the first bit to be sent.

2.1 Encapsulation of DCP Data PDUs

Figure 1 shows the frame format for a Q.922 Annex A frame carrying a DCP data PDU that contains Q.933 Annex E data. A DCP data PDU consists of a DCP Header and a DCP Payload. Only Q.933 Annex E frames with the control field set to UI are supported by this IA.



Note 1: All octets following a UI control field and prior to the FCS octets in a Q.933 Annex E frame are encoded/decoded as the DCP payload as specified in this IA.

Note 2: The DCP Header starts on octet 5 and the length of the DCP Header is defined in Section 3.

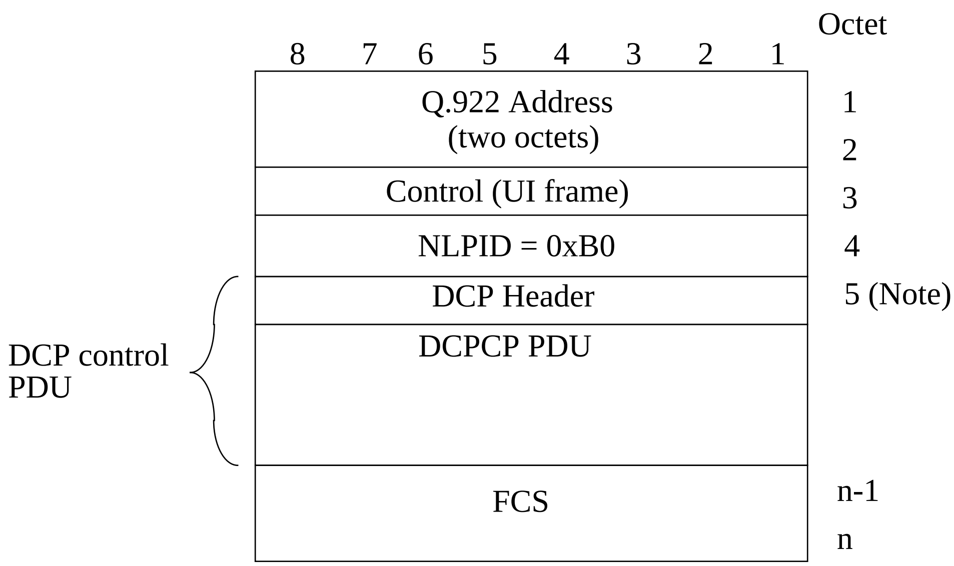
Note 3: Octet 4 of the FRF.9 frame is the IPI as defined in [6]. The NLPID in the DCP Payload of the FRF.9 frame is also an IPI as defined in Note 4 of Table 2 in [6].

Figure 1 - Encapsulation of a DCP data PDU that Contains Q.933 Annex E Data

Following the DCP Header is the DCP Payload as defined in Section 3.1. The DCP Payload, when decoded, begins with an NLPID as defined in ISO/IEC/TR 9577 [6] preceded by an optional Pad, in the first octet. As defined in section E.3 of Q.933 Annex E, the different NLPID values identify various encapsulation protocols. The Data following the NLPID in the DCP Payload is defined in sections E.4.1 and E.5 of Q.933 Annex E exclusive of the FCS.

2.2 Encapsulation of DCP Control PDUs

Figure 2 shows the frame format for a Q.922 Annex A frame carrying a DCP control PDU. A DCP control PDU consists of a DCP Header and a DCP Control Protocol (DCPCP) PDU. The DCP control PDU is defined in Section 3.2.



Note: The length of the DCP Header is defined in Section 3.

Figure 2 - Encapsulation of a DCP control PDU

3 DCP PROTOCOL DATA UNIT FORMAT

The DCP PDU is used by a DCP entity to communicate data or control information to the remote peer DCP entity. The most significant (earliest sent) octet(s) of the DCP PDU must be the DCP Header. The C/D bit of the DCP Header signals whether the DCP PDU is for control or for data.

3.1 DCP Data PDU Format

A DCP data PDU encapsulates compressed or uncompressed user data for transport to the peer DCP entity. Exactly one user data PDU is mapped into exactly one DCP data PDU. DCP does not specify a maximum size of a DCP data PDU. The maximum size is determined by the application using DCP and is dependent on the maximum size of a user data PDU, the DC Function expansion characteristics, and any anti-expansion procedures in use.

The DCP data PDU format is defined in Figure 3. The DCP Header is one or three octets in length depending on the length of the DCCI. The DCFD Data must be an integer number of octets.

	8	7	6	5	4	3	2	1
DCP Header	E	C/U	R-A	R-R	Reserved			C/D
	DCCI (zero or two octets)							
DCP Payload	DCFD Data (zero to n octets)							

E	0 for extension 1 for no extension
C/U	0 for uncompressed mode 1 for compressed mode
R-A	0 for no reset acknowledge 1 for reset acknowledge
R-R	0 for no reset request 1 for reset request
C/D	0 for DCP data PDU's
Reserved	Reserved for future use. Set to 0.
DCCI	Zero or two-octet DC Context Identifier
DCFD Data	Information defined in the DCFD including user data

Note: The specification of the E bit is defined in section 4.5.1 of Q.931 [8] and, as currently specified, the E bit is always 1 for DCP data PDUs.

Figure 3 - DCP Data PDU Format

The DCP Header for DCP data PDUs conveys per-PDU information: synchronization signalling, anti-expansion signalling, and an optional data compression context identifier (DCCI). Additional fields (e.g., sequence number) may be defined as part of the DCFD Data in each of the DCFDs.

3.1.1 Anti-expansion Signalling Format

Anti-expansion signalling (C/U) may be provided from the encoder to the decoder in one direction of the DC connection to indicate if the associated DCP Payload in this DCP data PDU is compressed or not. Anti-expansion signalling applies only to the protocol instance for the DCP context identified by the DCCI, if provided. The use of the C/U bit is defined in a DCFD. Manufacturers may choose to provide an alternative anti-expansion signalling mechanism according to a DCFD.

3.1.2 Synchronization Signalling Format

Synchronization signalling should be provided between DCP peers via the R-R and R-A bits in the DCP data PDU Header. They apply only to the synchronization protocol instance for the DCP context identified by the DCCI, if provided. R-R and R-A may be signalled in a DCP Header that accompanies a DCP Payload; they may also be signalled via a DCP Header without an attached DCP Payload. Separate R-R and R-A signals are provided to allow independent resynchronization of either or both directions of a DCP connection. The use of the R-R and R-A bits is defined in a DCFD. Manufacturers may choose to provide an alternative synchronization signalling mechanism according to a DCFD.

3.1.3 DCP Context Identification Signalling Format

There is exactly one DCP instance per frame relay virtual connection. The DCCI field allows for the identification of multiple data compression contexts within a DCP instance. This is used to multiplex multiple data compression contexts over each frame relay virtual connection. The data compression sender maps user data into a data compression context, which is identified by a DCCI. The data compression receiver uses this DCCI to address the specified context for the decoding of the user data and delivers the decoded user data to the upper layers. The information within the user data is sufficient for the upper layers to properly address the user data. The DCCI field of the DCP Header is either zero or two octets in length with the most significant octet first. The determination of the length of the DCCI field is DCFD specific.

3.1.4 DCP Payload for DCP Data PDUs

The contents of the DCP Payload is defined according to the DCFD. The DCP Payload must be an integer number of octets. Each DCFD must specify any padding or encoder flush procedures necessary to assure this.

3.2 DCP Control PDU Format

DCP Control Protocol (DCPCP) is used to enable, disable, and optionally configure DCP. DCPCP has two modes of operation: Mode-1 and Mode-2. Mode-2 provides full negotiation capabilities to enable, disable, and configure DCP using the Point to Point Protocol (PPP) Link Control Protocol (LCP) [10] negotiation procedures. Mode-1 uses a subset of the Mode-2 negotiation primitives with simplified procedures to enable and disable DCP with the default DCFD and default parameter values. Mode-1 operation is required; Mode-2 operation is optional.

Each DCP control PDU encapsulates exactly one control primitive for transport to the peer DCP entity. The DCP control PDU format is defined in Figure 4.

The length of a DCP Header for DCP control PDU's is one octet. The length of the DCPCP PDU varies depending on the control primitive. The DCP control PDU is sent in uncompressed mode.

	8	7	6	5	4	3	2	1
DCP Header	E	C/U	R-A	R-R	Reserved			C/D
DCPCP PDU	Control Primitive							

E	0 for extension 1 for no extension
C/U	0 for uncompressed mode
R-A	0 for no reset acknowledge
R-R	0 for no reset request
C/D	1 for DCP control PDU's
Reserved	Reserved for future use. Set to 0.
Control Primitive	Control information defined in this section

Note 1: The DCCI is not present for DCPCP PDUs. DCP control PDUs are not applicable on a data compression context basis.
Note 2: As currently specified, the E bit is always 1 for DCP control PDUs.

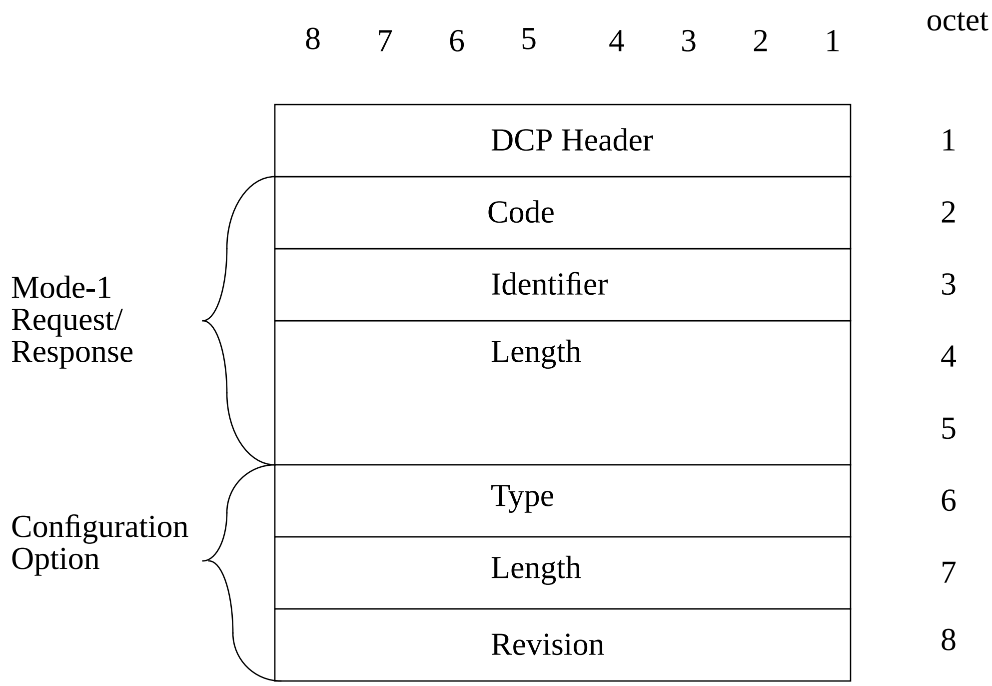
Figure 4 - DCP Control PDU Format

3.2.1 DCPCP PDUs

DCPCP PDUs use the same formats as the PPP LCP as defined in RFC 1661, section 5 [10]. DCPCP Mode-1 uses a subset of the DCPCP PDU formats (Configure-Request and Configure-Ack with the Mode-1 Configuration Option only).

3.2.1.1 Mode-1 Formats

The messages Mode-1 Request and Mode-1 Response, which are used in the Mode-1 Procedures are defined in Section 3.2.1.1.1 and Section 3.2.1.1.2. For reference, a description of the formats is shown in Figure 5. The only difference between Mode-1 Request and Response is that the Code field has the value “1” for the Request and “2” for the Response.



Note: The DCP Header is shown for reference.

Figure 5 - Mode-1 message formats

3.2.1.1.1 Mode-1 Request

The Mode-1 Request message is a DCPCP Configure-Request packet with the Code field set to “1”, the Identifier field set to “0”, the Length field set to “7”, and containing the Mode-1 Configuration Option as defined in Section 3.2.2.1.

3.2.1.1.2 Mode-1 Response

The Mode-1 Response message is a DCPCP Configure-Ack packet with the Code field set to “2”, the Identifier field set to “0”, the Length field set to “7”, and containing the Mode-1 Configuration Option as defined in Section 3.2.2.1.

3.2.1.2 Mode-2 Formats

Mode-2 formats are the same as the LCP packet formats defined in section 5 of RFC 1661 [10], with a unique set of Configurations options.

The LCP packets with codes 1 through 7 are required. The other LCP packets specified in RFC 1661 are optional.

3.2.2 DCPCP Configuration Options

DCPCP Configuration Options allow negotiation of DCFDs and their parameters. DCPCP uses the same Configuration Option format defined for LCP [10], with a separate set of Options.

Configuration Options 0-254 in this protocol indicate DCFDs that the receiver is willing or able to use to decode data sent by the sender. Configuration Option 255 is reserved for future use. As a result, it is to be expected that systems may offer to accept several algorithms, and negotiate a single one that will be used. Configuration Option 254 indicates that the receiver will support the default DCFD and the default parameter values.

Vendors may want to use proprietary DCFDs. This IA has made a mechanism available to negotiate these without encumbering a numbering authority with proprietary number requests.

The LCP option negotiation procedure is used. If multiple configuration options are listed in a single Configure-Request, they shall be listed in order of decreasing preference. If an option is unrecognized, a Configure-Reject must be sent. If all protocols the sender implements are Configure-Rejected by the receiver, then an empty Configure-Request/Ack pair are exchanged to indicate that no compression is enabled in that direction on the link. In order to provide proper synchronization capabilities, DCP data PDUs containing only a DCP Header with R-R set to 1 must be sent by a sender and processed by a receiver that have no compression enabled.

If an option in a Configure-Request is recognized by the local peer, but not acceptable due to values in the request (or optional parameters not in the request), a Configure-Nak must be sent to the remote peer. The Configure-Nak must contain only those options that will be acceptable. A new Configure-Request should be sent by the remote peer with only the single preferred option, adjusted to accommodate the options listed in the Configure-Nak.

The DCPCP Option code points are currently assigned to match up with those of TIA/EIA 655 [11]. Current values used for DCP are assigned as follows:

DCPCP Option	DCFD name
0	OUI
23	X3.241 - LZS-DCP (X3.241-1994)
254	Mode-1

3.2.2.1 Mode-1 Configuration Option

The simplified DCPCP Mode-1 negotiation uses a single Configuration Option. A summary of the Mode-1 Configuration Option format is shown below. The fields are sent in ascending order.

8 7 6 5 4 3 2 1	octet
Type	1
Length	2
Revision	3

Figure 6 - Mode 1 Configuration Option Format

3.2.2.1.1 Type

The Type field shall be one octet in length and shall be set to 254.

3.2.2.1.2 Length

The Length field shall be one octet in length and shall be set to 3.

3.2.2.1.3 Revision

The Revision field shall be one octet in length and contains the revision number. The current revision is 1.

3.2.2.2 Public DCFD Configuration Options

Configuration Options provide a way to negotiate the use of a publicly defined DCFD. All publicly defined DCFDs use a common configuration option format; the Values field is specific to the DCFD and is defined as a part of the DCFD.

A summary of the DCFD Configuration Option format is shown below. The fields are sent in ascending order.

8 7 6 5 4 3 2 1	octet
Type	1
Length	2
Values	3 . . n

Figure 7 - DCFD Configuration Option Format

3.2.2.2.1 Type

The Type field shall be one octet in length and is set to 1 to 254 as specified in the CCP Option Type field of TIA/EIA 655, Annex B [11].

3.2.2.2.2 Length

The Length field shall be one octet in length and is set to the number of octets in the Values field plus 2.

3.2.2.2.3 Values

The Values field shall be zero or more octets in length, and contains additional data as defined by the DCFD.

3.2.2.3 Proprietary DCFD OUI Configuration Option

This Configuration Option provides a way to negotiate the use of a proprietary DCFD. Since the first matching DCPCP configuration option will be used, any known OUI compression options should be sent first, before the common options are used.

Before accepting this option, an implementation must verify that the Organization Unique Identifier identifies a proprietary algorithm that the implementation can decode, and that any vendor specific negotiation values are fully understood.

A summary of the Proprietary Data Compression OUI Configuration Option format is shown in Figure 8. The fields are sent in ascending order.

8 7 6 5 4 3 2 1	octet
Type	1
Length	2
OUI	3 4 5
Subtype	6
Values	7 . . n

Figure 8 - OUI Configuration Option Format

3.2.2.3.1 Type

The Type field shall be one octet in length and is set to 0 for OUI.

3.2.2.3.2 Length

The Length field shall be one octet in length and is set to the number of octets in the Values field plus 6.

3.2.2.3.3 OUI

The OUI field shall be three octets in length and is the vendor's IEEE Organization Unique Identifier (OUI) assigned to the vendor by IEEE 802 [1]. This identifies the option as being proprietary to the indicated vendor. The bits within the octet are in canonical order, and the most significant octet is sent first.

3.2.2.3.4 Subtype

The Subtype field shall be one octet in length and is specific to each OUI. The purpose of this field is to select between multiple proprietary DCFDs under the vendor’s OUI. Each OUI implements its own values.

3.2.2.3.5 Values

The Values field shall be zero or more octets, and contains additional data as determined by the vendor's compression protocol. The purpose of this field is the same as the Values Field in Public DCFD Configuration Options (Section 3.2.2.2). Vendors should use the public DCFDs as a guide for implementing this field.

4 DCP PROCEDURES

This section contains the procedures used by the Data Compression Protocol.

4.1 Anti-Expansion Procedure

Anti-expansion procedures may be provided by the sender to reduce the offered load to a network if the DCP Function expands the uncompressed user data. They may also be used to avoid exceeding a maximum PDU size restriction of an intervening protocol layer when the DCP Function expands the uncompressed user data. The sender must set C/U = 1 when the compression encoding has been performed on the user data. The sender must set C/U = 0 when the compression encoding has not been performed on the user data.

Anti-expansion procedures must be provided by the receiver. When C/U = 1, the decoder must decode the DCP Payload. When C/U = 0, the decoder must not decode the DCP Payload. Each DCFD must specify any requirements of the decoder with respect to the synchronization of the data compression context during anti-expansion procedures (See Section 5).

4.2 Synchronization Procedure

Synchronization procedures should be provided to recover from a loss of synchronization between DCP peers. Frame relay does not assure reliable transport of DCP PDU's. DCP Function decoders commonly do not recover from decompressing dropped, erroneous, or mis-ordered PDUs and propagate errors catastrophically until they are reset to an initial state.

Each DCFD must specify a means of synchronization which may be one or more of the following:

- a means to detect loss of synchronization (e.g., by sequence number or LCB),

- a means to recover from loss of synchronization using the synchronization signalling provided by the DCP Header (e.g., reset every frame or reset on loss of synchronization), or
- the use of an intervening protocol layer that provides reliable transport (e.g., LAPB error recovery and sequential delivery).

When a local DCP receiver detects or receives an indication of loss of synchronization in the remote-to-local direction of the DCP connection, it shall generate a R-R signal set to “1” on a new empty DCP data PDU or on the next DCP data PDU containing user data destined for the remote DCP peer. The DCCI in that DCP Header, if provided, shall correspond to the local-to-remote direction of the same DCP context. Once a R-R set to “1” has been generated, any DCP data PDU's received in the remote-to-local direction of that DCP context that contain compressed user data (C/U=1) shall be discarded until a R-A set to “1” is received for that context. The R-R signal set to “1” may be repeated to increase reliability.

When a remote DCP receiver detects a R-R set to “1” in the local-to-remote direction of the DCP context, it shall reset its encoder to the initial state and shall generate a R-A signal set to “1” on a new empty DCP data PDU or on the next DCP data PDU containing user data destined for the local DCP peer. The DCCI in that DCP Header, if provided, shall correspond to the remote-to-local direction of the same DCP context.

When a local DCP receiver receives a R-A signal set to “1” in the remote-to-local direction of the DCP context, it may, according to the DCFD, reset its state for that context to the initial state. The initial state of the DCP Function should be specified in the DCFD. The local DCP receiver must decode any user data in the DCP data PDU containing the R-A set to “1” and all subsequent DCP data PDUs until another loss of synchronization is detected.

Note: For transport of HDLC derivative protocols (e.g. IEEE 802.2) and other time sensitive protocols, a R-R set to “1” should be sent immediately upon detection of a receive failure. Furthermore, additional R-Rs set to “1” should be sent every second until a R-A set to “1” is received.

4.3 DCP Control Protocol Procedures

This section describes the Mode-1 and Mode-2 DCPCP procedures. The data compression protocol is designed to accommodate the selection and use of a single Data Compression Function from a menu of candidate Data Compression Functions. The formats and procedures for the DCP Function sublayer shall be defined in a DCFD.

4.3.1 Mode-1

DCPCP Mode-1 provides a simple negotiation protocol to enable DCP with the default DCFD and default parameter values as specified in Annex A. Once DCP is successfully enabled, a Mode-1 only implementation may disable DCP by forcing the virtual connection to the inactive state, or by sending a Mode-1 request and not sending a Mode-1 response.

4.3.1.1 Mode-1 Procedures

DCPCP Mode-1 consists of three phases: Disabled, Initialization, and Operation. The Disabled phase is entered upon powerup or when a frame relay virtual connection is released. The Initialization phase is entered upon frame relay virtual connection establishment and when DCP is enabled. The Operation phase is entered upon the successful completion of the Initialization phase. Unsuccessful completion of the Initialization phase causes DCP to enter the Disabled Phase. DCP data PDUs are transferred only when DCPCP Mode-1 is in the Operation phase. DCP control PDUs may be transferred in any phase.

The Mode-1 Initialization phase shall be entered when a frame relay virtual connection to a peer is established, DCP is administratively enabled (by the user), or a Mode-1 Request is received. A signal for PVC establishment is obtained via link management Procedures (e.g. Q.933 Annex A) when there is a transition from a PVC status of “inactive” to “active” and/or the presence of both “active” and “new” for a PVC. A signal for SVC establishment is obtained via Q.933 [9] call control procedures when a call transitions to the Active state (U10 or N10). When the Mode-1 Initialization phase is entered, the handshake procedure shall be initiated. Each time the handshake procedure is initiated, it shall operate as follows:

> First, the entity shall send a Mode-1 Request to its peer. Next, the entity shall start a handshake completion timer to expire P-Retry-Time seconds after the Mode-1 Request was sent. Upon receiving a Mode-1 Request, the entity shall send a Mode-1 Response. When a Mode-1 Response has been both sent and received, the handshake procedure is complete and the entity shall enter the Mode-1 Operation phase. If the handshake completion timer expires before the handshake procedure is completed, the handshake procedure shall be re-initiated. If the handshake procedure is re-initiated P-Retry-Count times without leaving the Mode-1 Initialization phase, the entity shall terminate the handshake procedure and enter the Disabled phase.

The Mode-1 Disabled phase shall be entered when a frame relay virtual connection to a peer is released. A frame relay virtual connection is released when a signal for PVC inactive is obtained via link management Procedures (e.g. Q.933 Annex A) when there is a transition from a PVC status of “active” to “inactive” or a signal for SVC release is obtained via Q.933 [9] call control procedures when a call transitions out of the Active state (U10 or N10). While in the Mode-1 Operation phase, if a Mode-1 Request is received, the entity shall terminate transfer of DCP data PDUs and enter the Mode-1 Initialization phase. Mode-1 phase transitions are shown in Table 1.

Table 1 Mode-1 Phase Transitions

Stimulus \ Phase	Disabled	Initialization	Operation
Frame relay virtual connection established	Initialization	Initialization	Initialization
Frame relay virtual connection released	Disabled	Disabled	Disabled
Mode-1 request received	Initialization	Initialization	Initialization
Handshake complete	n/a	Operation	n/a
Handshake unsuccessful	n/a	Disabled	n/a

Note: The handshake procedure should be active only when in the Initialization phase.

4.3.1.2 Mode-1 Parameters

The parameters listed in Table 2 are used in the Mode-1 Initialization phase and should be user configurable.

Table 2 Mode-1 parameters

Parameter Name	Range	Suggested Value
P-Retry-Time	N/A	3 seconds
P-Retry-Count	3 or greater	10

4.3.2 Mode-2

Mode-2 is optional and provides the capability to enable and disable DCP, and to negotiate DCFDs and their associated parameters.

4.3.2.1 Mode-2 Procedures

Mode-2 shall use the finite-state automaton described in sections 3 and 4 of RFC 1661 [10] with the following exceptions:

1. If the Mode-2 negotiate DCPCP finite-state automaton enters the Closed state because of negotiation time out (P-Revert-Time seconds), the entity shall enter the Mode-1 Initialization phase.
2. An entity may abandon Mode-2 and enter the Mode-1 initialization phase at any time.

3. If an entity operating in Mode-2 receives a Mode-1 Request at any time, it shall enter the Mode-1 Initialization phase.
4. If an entity that supports Mode-2 is currently in Mode-1 and it receives a Mode-2 Configure-Request, it shall begin Mode-2 negotiation.

Note: The (lower layer) Up/Down events for the automaton should be generated by the PVC and SVC active signals as described in Section 4.3.1.1.

Before any DCP data PDUs may be communicated, DCPCP must reach the Opened state.

4.3.2.2 Mode-2 Parameters

In addition to the Parameters necessary for the LCP negotiation mechanism [10], the parameters listed in Table 2 and Table 3 are required for fallback operation with Mode-1.

Table 3 Mode-2 parameters

Parameter Name	Range	Suggested Value
P-Revert-Time	N/A	30 seconds

5 DCFD REQUIREMENTS

DCFDs must specify the data compression encoder/decoder, data compression options, and anything else that effects interoperability in sufficient detail so that implementations can interoperate. Each DCFD must specify:

- A description of the data compression encoder/decoder including the encoder/decoder method, an end of frame detection method, and the initial state of the encoder/decoder.
- A description of the synchronization procedures including any reliance on a reliable transport protocol, a means of detecting loss of synchronization, the use of R-R/R-A in the DCP Header, and any other method used for synchronization.
- A description of the anti-expansion procedures including the use of C/U in the DCP Header, any other anti-expansion procedure, and a description of any encoder or decoder procedures (e.g. resets, vocabulary updates) during anti-expansion mode.
- A means for identifying the data compression context including the use of a DCCI or any other means of identifying the data compression context.
- A description of any additional header or trailer octets including the use of the E bit in the DCP header and a description of each field in any additional header or trailer octets.
- A means to ensure octet alignment at the end of a frame including any padding fields or flushing procedures.
- A definition and description of any additional parameters specific to the DCFD required for negotiation including a definition of the negotiation syntax, a description of each parameter, and a description of the valid parameter values.
- A description and definition of anything else that effects interoperability.

ANNEX A

Default Data Compression Function Definition

A.1 Introduction

This annex specifies a default DCFD which is required to ensure data compression interoperability. The default DCFD requires the use of the ANSI X3.241-1994 [2] compression algorithm with the protocol described in this annex. This combination of algorithm and protocol is referred to as “LZS-DCP”.

Note: This is the only DCFD specified in this document, although others may be defined in the future.

A.2 Operational Formats

The format of the DCP PDU for this DCFD using the ANSI X3.241-1994 data compression algorithm shall be as shown in Figure A.1. Octets are sent in ascending order.

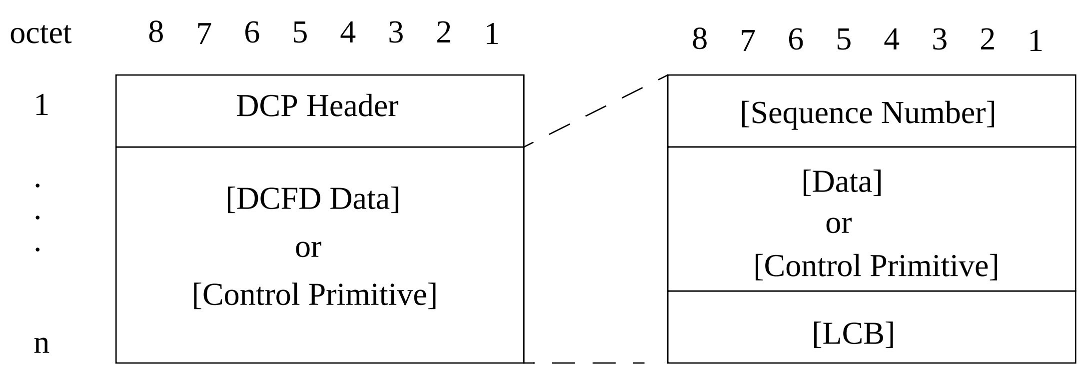


Figure A.1 - Format of DCP PDU

Note: The names of the optional fields are shown enclosed in brackets.

A.2.1 DCP-Header

The format of the DCP-Header shall be as shown in Section 3.

A.2.1.1 E - Extension Bit (bit 8)

This bit shall be set to “1”.

A.2.1.2 C/U - Compressed/Uncompressed Bit (bit 7)

This bit is defined as specified in Section 3.

A.2.1.3 R-A - Reset-Ack (bit 6)

The R-A bit shall be used to inform the receiver peer that the history buffer specified by the DCCI in the DCP data PDU was in the cleared state just before the data contained in the DCP data PDU was processed by the compression transformation (Section A.3.1.1.1). This bit shall be set to a value of “1” to indicate a Reset-Ack.

A.2.1.4 R-R - Reset-Request (bit 5)

The R-R bit shall be used to request that the sender peer clear the history buffer specified by the DCCI in the DCP data PDU. This bit shall be set to a value of “1” to indicate a Reset-Request.

A.2.1.5 Res - Reserved (bits 4,3,2)

These bits are defined as specified in Section 3.

A.2.1.6 C/D - Control/Data (bit 1)

This bit is defined as specified in Section 3.

A.2.1.7 DCCI

The DCCI field shall have a length of 0 octets when P-Context-Count equals “0” or “1”. If P-Context-Count is 2 or more this field shall be 2 octets in length, most significant octet first.

The DCCI field shall contain the number of the data compression context for which this DCP data PDU applies. The number shall range from 1 to P-Context-Count.

The DCCI field size shall match in both directions on the link. If P-Context-Count is 2 or more on one direction of the link, the DCCI field shall be present in all DCP data PDUs in both directions, and it shall be 2 octets in length.

A.2.2 Sequence Number

The sequence number field shall be one octet in length. When P-Check-Mode = “Sequence Number” or “Sequence Number + LCB”, the sequence number field shall be present in all DCP data PDUs that contain a data field.

A.2.3 Data

The data field shall contain a single user data PDU when the C/D bit in the DCP Header is set to “0” in either compressed or uncompressed form, depending on the state of the C/U bit. The length of this field shall always be an integer number of octets. This field is required in all DCP data PDUs that do not have the R-R or R-A bits set to “1”.

If the C/U bit is set to “1”, the form of the data field shall be one block of compressed data as defined in 3.2 of X3.241-1994, with the following exceptions: 1) the end marker may be followed with additional octets containing only zeros; 2) if the final octet in the block of compressed data has a value of “0”, then it may be removed from the data field.

Note: The block of compressed data begins with a raw byte token (3.6, 5.3 of X3.241-1994) or string token (3.9, 5.4 of X3.241-1994) and ends with an end marker (3.2, 5.7 X3.241-1994). There is only one end marker per block of compressed data.

If the C/U bit is set to “0”, the data field shall contain the uncompressed form of the user data PDU.

A.2.4 Control Primitive

The Control Primitive field shall be as defined in Section 3.2 when the C/D bit in the DCP Header is set to “1”. There shall be no Sequence Number field or LCB field when the C/D bit is set to “1”.

A.2.5 Longitudinal Check Byte (LCB)

The LCB field shall be one octet in length, and if present shall be the last octet in the DCP Payload. When P-Check-Mode = “LCB” or “Sequence Number + LCB”, this field shall be present in all DCP data PDUs where the data field contains compressed data (C/U = “1”). This field shall not be present in DCP Payloads where the data field contains uncompressed data (C/U = “0”). This field shall contain the result of the LCB calculation, in accordance with Section A.3.2.2.2.

A.3 Procedures for ANSI X3.241-1994 compression algorithm using DCP-Header

The LZS-DCP compression protocol supports both connection-less and connection oriented compression. Multiple DCP connections are supported through the use of multiple contexts.

A.3.1 Operation

The operation of the LZS-DCP compression protocol shall depend on the values of the parameters listed in Section A.4.

A.3.1.1 Sender

This section describes the compression transformation and encapsulation procedure for the sender.

A.3.1.1.1 Compression Transformation

The compression transformation shall result in either compressed or uncompressed data. When a user data PDU arrives, it shall be assigned to a particular context and processed according to ANSI X3.241-1994 to form compressed data or used as is to form uncompressed data. Prior to the compression transformation, if a Reset-Request is outstanding for the context to be used, the history buffer shall be cleared. In performing the transformation, if P-Process-Mode = “Compressed-Only”, the history buffer shall only be updated if the result is compressed data. If P-Process-Mode = “Compressed+Uncompressed” the history buffer shall be updated when either compressed data or uncompressed data is produced. Uncompressed data may be sent at any time. Uncompressed data shall be sent if the compression transformation causes enough expansion to cause the frame size to exceed the maximum frame relay information field size.

If P-Process-Mode = “Compressed-Only” and the encoder has modified the history buffer before sending an uncompressed DCP data PDU, the history buffer shall be cleared or restored to its previous state before the next user data PDU is processed.

Note: The maximum expansion of the ANSI X3.241-1994 algorithm is 12.5%.

A.3.1.1.2 Encapsulation Procedure

The output of the compression transformation shall be placed in the data field of the DCP data PDU. The C/U bit shall be set according to whether the data field contains compressed or uncompressed data. If the sequence number field is present according the value of P-Check-Mode, the sequence number value shall be placed in the sequence number field for the applicable context and the sequence number value shall be incremented. If the data field contains compressed data, and P-Check-Mode is set accordingly, the LCB field shall be present and its value shall be computed as specified in Section A.3.2.2.2

If the history buffer is in the clear state (the history buffer contains no data bytes) prior to performing the compression transformation, the resulting compressed or uncompressed DCP data PDU shall be sent with the R-A bit set to “1”.

A.3.1.2 Receiver

When a DCP data PDU is received from the peer entity, the R-R and R-A bits shall be checked. If the R-R bit is set to “1”, the local sending entity shall be signalled that a Reset-Request has been received for the context specified by the DCCI field. If the R-A bit is set to “1”, any outstanding receive failure for the specified context shall be cleared. If no receive failure is outstanding, and the sequence number field is present, its value shall be checked according to Section A.3.2.2.1. If a receive failure has occurred, it shall be handled according to Section A.3.3 and the remainder of the DCP data PDU shall be discarded. If no receive failure is detected, the data shall be assigned to the indicated decompression context and processed according to P-Process-Mode.

If the C/U bit is set to “1”, a single octet containing the value “0” shall be appended to the data field and the resulting compressed data block shall be decoded according to ANSI X3.241-1994. If the LCB field is present on the received DCP data PDU, an LCB for the uncompressed data shall be computed and checked against the received LCB according to Section A.3.2.2.2. If a receive failure has occurred, it shall be handled according to Section A.3.3.

If the C/U bit is set to "0" and P-Process-Mode = "Compressed+Uncompressed", the specified decompression history buffer shall be updated with the received data. If the C/U bit is set to "0" and P-Process-Mode = "Compressed-Only", the specified decompression history buffer shall not be modified.

A.3.2 Parameter Specific Procedures

This section describes procedures used for the P-Context-Count, P-Check-Mode, and P-Process-Mode parameters.

A.3.2.1 P-Context-Count

The P-Context-Count parameter determines the number of contexts to be maintained for the data compression protocol. Each context represents a logical connection between the DCP peers. Each context shall contain an X3.241 history buffer and additional information for error detection and synchronization procedures. The DCP peers use error detection and synchronization procedures to ensure that both the encoder and decoder copies of each history buffer are identical.

Setting P-Context-Count to the value "0" indicates that the compression is to be on a connectionless basis. In this case, a single context shall be used and the history buffer shall be cleared at the beginning of every DCP data PDU. The sender shall set the R-A bit to "1" on all outgoing DCP data PDUs.

When P-Context-Count is set to the value "1", a single context shall be maintained by each of the DCP peers.

When P-Context-Count is greater than "1", P-Context-Count separate contexts shall be maintained by the receiver. The sender peer may use up to P-Context-Count separate contexts. The DCCI field shall be included on all DCP data PDUs in both directions in this case. Its size shall be determined by the P-Context-Count as described in Section A.2.1.7.

A.3.2.2 P-Check-Mode

The P-Check-Mode parameter determines what type of error checking information is added to the outgoing DCP data PDU. The error check information is included in the sequence number and LCB fields.

To maintain connection integrity, P-Check-Mode = "None" should not be used when P-Context-Count is not equal to "0".

A.3.2.2.1 Sequence Numbers

The value of the sequence number field (the sequence number of the DCP data PDU) shall begin with "1" and increment modulo 256 on successive DCP data PDUs that contain data fields. This number shall be relative to the context used.

On receipt of a DCP data PDU with the R-A bit set to "0", if the sequence number of the DCP data PDU is any number other than (N+1) mod 256, where N is the sequence number of the last DCP data PDU received for the same context, or an initial value of "0", a receive failure for that context has occurred. The receive failure shall be handled according to the synchronization procedure in Section A.3.3.

The sequence number shall not be reset by the sender when a packet containing a Reset-Ack set to “1” is sent. The receiver shall resynchronize its sequence number reference for the indicated context when a DCP data PDU containing a Reset-Ack set to “1” is received.

A.3.2.2.2 LCB

The LCB octet shall be the Exclusive-OR of FF(hex) and each octet of the uncompressed user data (prior to the compression transformation).

On receipt, the receiver shall compute the Exclusive-OR of FF(hex) and each octet of the decoded user data. If this value does not match the received LCB in the DCP data PDU, then a receive failure for that context has occurred. The receive failure shall be handled according to the synchronization procedure in Section A.3.3.

A.3.2.3 P-Process-Mode

The P-Process-Mode parameter determines how DCP data PDUs containing uncompressed data are handled by the encoder/decoder peers. When P-Process-Mode = “Compressed-Only”, uncompressed DCP data PDUs do not modify the contents of the history buffers. When P-Process-Mode = “Compressed+Uncompressed”, both the encoder and decoder update their history buffers with the uncompressed data.

A.3.3 Synchronization Procedure

The DCP-Header includes R-R (Reset-Request) and R-A (Reset-Ack) bits in order to provide a mechanism for indicating a receive failure in one direction of a DCP connection without affecting traffic in the other direction. A receive failure shall be determined using the procedures specified in Section A.3.2.2, according to the value of P-Check-Mode.

On the occurrence of a receive failure, an implementation shall send a DCP data PDU with the R-R bit set to “1” (a Reset-Request) and with the DCCI matching that of the context that had the failure. The data field may be present if user data is waiting to be transported for that context, or the R-R bit may be set in a DCP data PDU sent without sequence number, data, or LCB fields. Once a receive failure has occurred, the data field in any subsequent DCP data PDUs received for that context shall be discarded until a DCP data PDU containing a Reset-Ack set to “1” is received. It shall be the responsibility of the receiver to ensure the reliability of the reset request/ acknowledge mechanism. This may require the transmission of additional Reset-Requests set to “1” before a Reset-Ack set to “1” is received.

Upon reception of a DCP data PDU containing a Reset-Request set to “1”, the encoder’s history buffer shall be cleared, which shall cause a Reset-Ack set to “1” to be sent with the next DCP data PDU for the context specified in the DCCI (Section A.3.1.1.2).

Note: The DCP data PDU is also delivered to the local receiver as well, for normal processing as described in Section A.3.1.2.

On receipt of a Reset-Ack set to “1”, the decoder history buffer may be cleared. After clearing, any compressed or uncompressed data contained in the DCP data PDU is processed.

Note: The clearing of the decoder history buffer is not required due to the characteristics of the X3.241 compression algorithm. Any data received will not reference the old history buffer contents.

Reset-Requests and Reset-Acks shall be specific to the DCCI specified in the DCP data PDU containing them.

A.4 Parameters for ANSI X3.241-1994 compression algorithm using DCP-Header

The parameters and values for this annex shall be as listed in Table A.1. Support of these default values is mandatory for this DCFD.

Table A.1 Data compression parameters specific to the LZS-DCP compression type

Parameter Name	Default Value	Other Values
P-Context-Count	1	[0 .. 65535]
P-Check-Mode	Sequence Number + LCB	None, Sequence Number, LCB
P-Process-Mode	Compressed-Only	Compressed+Uncompressed

A.5 LZS-DCP configuration option format

The configuration option format for the LZS-DCP DCFD shall be as shown in Figure A.2.

8 7 6 5 4 3 2 1	octet
Type	1
Length	2
Context Count	3
	4
Check Mode	5
Process Mode	6

Figure A.2 - LZS-DCP configuration option format

A.5.1 Type

The Type field shall be 1 octet in length and shall specify the compression type for which the configuration option applies. For type LZS-DCP, the Type field shall be 23.

A.5.2 Length

The Length field shall be 1 octet in length and shall indicate the length of the configuration option. For this configuration option, its value shall be “6”.

A.5.3 Context Count

The Context Count field shall be two octets, most significant octet first, and shall be used to specify the value of P-Context-Count.

A.5.4 Check Mode

The Check Mode field shall be one octet in length and shall be used to set the value of P-Check-Mode. The values shall be defined as follows:

0	P-Check-Mode = “None”
1	P-Check-Mode = “LCB”
2	P-Check-Mode = “Sequence Number”
3	P-Check-Mode = “Sequence Number + LCB”

A.5.5 Process Mode

The Process Mode field shall be one octet in length. It shall be used to set P-Process-Mode. The values shall be defined as follows:

0	P-Process-Mode = “Compressed-Only”
1	P-Process-Mode = “Compressed+Uncompressed”

ANNEX B

Data Compression with ATM Service Interworking

Data compression over frame relay may be used with FR/ATM Service interworked connections in the following manner: The Q.933 Annex E data compression header (NLPID = xB0) is mapped to the RFC 1483 LLC/NLPID format (LLC/NLPID = x FE FE 03 B0) in the service interworking function and passed via an ATM connection to a second service interworking function or an ATM end system. If an ATM end station receives an RFC 1483 formatted frame with the above LLC/NLPID encoding and it does not implement the FRF.9 Data Compression Protocol and Q.933 Annex E, it discards the frame. This causes the DCP negotiation procedures in the originating frame relay device to time-out and disable data compression for the connection.

The following text applies to FRF.8 Section 5.3.1, **Encapsulation Mapping in Translation Mode**.

New text:

5.3.1.4 FRF.9 Data Compression Protocol PDUs

Figure B.1 shows the translation between the FR Q.922 PDU payload header and the ATM AAL5 CPCS-PDU payload header that shall be performed by the IWF for FRF.9 Data Compression Protocol (DCP) PDUs.

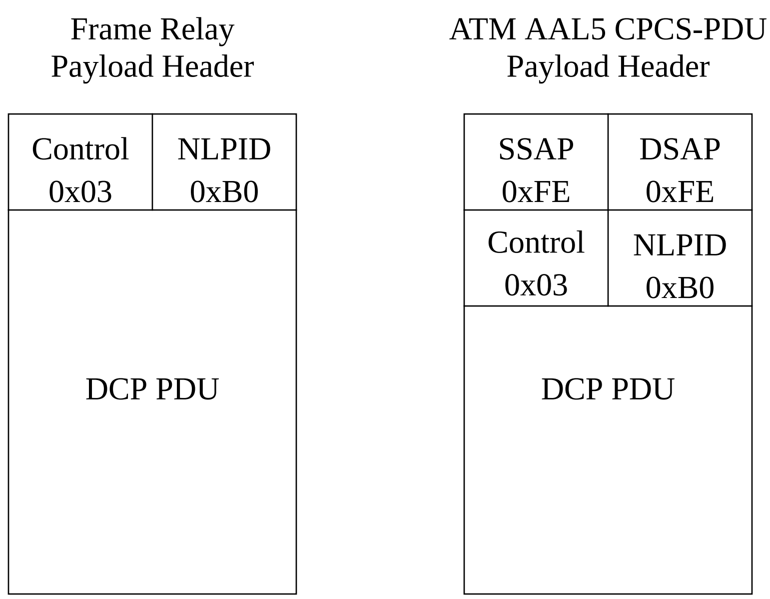


Figure B.1 - FR/ATM Payload Header Translation for FRF.9 DCP

ANNEX C - Informational

Example of the Use of Multiple Histories

The use of multiple histories over a single frame relay virtual connection is an option supported by this Implementation Agreement. The use of this option can increase compression ratios for certain applications. An analysis of the user data may be required to determine if the use of multiple histories increases compression ratios.

Data compression functions rely on redundancy in data streams to achieve compression. The more redundancy in a data stream, the higher the compression ratio.

In a multiprotocol environment, there may be many independent data streams multiplexed over a single frame relay virtual connection. The aggregation of these independent data streams, when multiplexed into a single stream, may have less redundancy than each individual data stream. Therefore, the compression ratio of the aggregate stream using a single history may be lower than the compression ratio when each independent data stream is compressed using a separate history.

An implementation of this Implementation Agreement may choose to use a separate history for each individual, non-aggregated, data stream. Note that it is the data compression sender that may choose to assign individual data streams to each history, without the knowledge of the remote data compression receiver, using any criteria that the sender finds convenient. The remote receiver decompresses the data received in each history and the decompressed Q.933 Annex E PDU is delivered to the upper protocol layer. The information contained in this decompressed Q.933 Annex E PDU contains sufficient information to deliver the PDU to the specified protocol entity.

Also note that the histories of the local sender to remote receiver are independent from the histories of the remote sender to local receiver. This implies that the criteria used by the local sender to assign individual data streams to a history is independent of the criteria that the remote sender uses. It also implies that one sender may choose to assign individual data streams to separate histories and the other sender may choose to use a single history for the aggregate data stream.

For example, the local sender may choose to assign a different history for each NLPID value contained within a Q.933 Annex E PDU. The remote receiver has no knowledge of this other than the number of histories that it must support for this virtual connection. When a PDU arrives at the local sender with NLPID = 0xCC the local sender assigns the PDU to the history with DCCI value 1. When a PDU arrives at the local sender with NLPID = 0x80 the local sender assigns the PDU to the history with DCCI value 2 and so on.

The remote receiver decodes each PDU using the history for the DCCI value contained in the DCP data PDU. It delivers all decoded PDUs to the next protocol layer which uses the NLPID in the decoded PDU to deliver it to the specified protocol entity.

The remote sender in this example may choose to use a single history for the aggregate data stream. When a PDU arrives at the remote sender with any NLPID value the remote sender assigns the PDU to the history with DCCI value 1.

The local receiver decodes each PDU using the history for the DCCI value contained in the DCP data PDU. It will only receive DCP data PDUs with a DCP payload on the history with DCCI value 1. It may receive DCP data PDUs with no DCP payload for any DCCI value when the remote receiver losses synchronization and sets the Reset-Request bit in the DCP Header. The local receiver delivers all decoded PDUs to the next protocol layer which uses the NLPID in the decoded PDU to deliver it to the specified protocol entity.