Copies of this document may be purchased from:
Global Engineering, 15 Inverness Way East,
Englewood, CO 80112-5704
Phone: (800) 854-7179 or (303) 792-2181 Fax: (303) 792-2192

FIBRE CHANNEL

FABRIC LOOP ATTACHMENT
(FC-FLA)

REV 2.7

NCITS working draft proposed
Technical Report

August 12, 1997

Secretariat: Information Technology Industry Council

POINTS OF CONTACT:

Roger Cummings (T11 Chairman)
DISTRIBUTED PROCESSING TECHNOLOGY
140 Candace Drive
Maitland, FL 32751
Phone: (407) 830-5522 x348
Fax: (407) 260-5366
E-Mail: cummings_roger@dpt.com

Ed Grivna (T11 Vice Chairman)
Cypress Semiconductor
2401 East 86th Street
Bloomington, MN 55425
(612) 851-5046
Fax: (612) 851-5087
E-Mail: elg@cypress.com

I. Dal Allan
(Fibre Channel Working Group Chairman)
ENDL
14426 Black Walnut Court
Saratoga, CA 95070
(408) 867-6630
Fax: (408) 867-2115
E-Mail: dal.allan@mcimail.com

Jeffrey Stai (Technical Editor)
Brocade Communications Systems, Inc.
15707 Rockfield Boulevard, Suite 215
Irvine, CA 92618
(714) 455-2908
Fax: (714) 455-9287
E-mail: stai@brocadecomm.com

Editor’s Notes, revision 2.7:

- Incorporate comment responses from first public review period.
- Note: many many of the change bars resulted from editorial input from ANSI. One particular culprit was a minor adjustment to the “hex” format from: hex’F00D’ to: hex ‘F00D’.

Editor’s Notes, revision 2.6:

- Incorporate comment responses from first letter ballot by T11.

Editor’s Notes, revision 2.5 (as always, check change bars for “little” edits!):

- Revision submitted for first letter ballot by X3T11.
- Added Name Service requirements to Annex A and well-know address stuff.
- Completed definition and description of use of Loop Fabric Address.
- Established OPNfr as the only mechanism for delivery of multicast and broadcast to the loop.
- Clarification of RSCN issues.
- 6.3 and 6.5 were swapped so that the order of subjects matched clause 5.
- Fixes to 5.6, particularly the flow figure.
- Annex B: clarify Streamed Sequence rules.
- Deleted some unneeded reference documents (clause 2).
- Earlier revs stated that LIP(y,x) could be used for ULP-specific hard reset. Turns out this violated FC-AL.

Editor’s Notes, revision 2.4 (as always, check change bars for “little” edits!):

- Noted that OPNfr is under consideration to be the only method for the FL_Port of delivering broadcast/multicast to loop.
- The SCSI and IP stuff is moved to a normative annex (Annex B).
- Time for NL_Port to wait to receive FAN is E_D_TOV. No ACC for FAN.
- Clarified login to Well-Known Addresses.
- SCN is replaced with RSCN; also many changes to both RSCN and SCR in Annex A, including adding an ACC for the RSCN, and some basic filter spec for SCR.
- Aligned hex and binary notation to be same as FC-PLDA. (This caused the most change bars!)
- Added “Examples” annex, deleted “Fabric Use” annex. Examples will be filled in “soon”.

Editor’s Notes, revision 2.3:

- Fixed LFA definition and added Name Server definition. Added note indicating <u>table</u> “Notes” (only) are NORMATIVE.
- Clarified table headings in many places; changed “Recipient” to “Responder” to be consistent with FC-PH usage in all places.
- Expanded frame retransmission rules.
- Added initialization flow diagrams. Also note that FAN wait for delivery and FAN ACC are pending.
- Sanitized discovery in several places, also made PLOGI in Class 2 more allowable. SCSI Target discovery rewritten.
- New definitions for E_D_TOV and R_A_TOV. Note that R_A_TOV is NON-ZERO for ALL CASES in this document!
- In Annex A, added <u>proposed</u> SCR ELS, <u>proposed</u> revised wording for SCN, clarified LPC definition.
- Added Annex for FLA examples, deleted most text from BB_Credit annex. Renumbered annexes.

Editor’s Notes, revisions 1.0, 1.1, 2.0, 2.1, 2.2: (deleted) draft proposed NCITS Technical Report

Fibre Channel — Fabric Loop Attachment (FC-FLA)

Secretariat

Information Technology Industry Council

Approved ,199

American National Standards Institute, Inc.

Abstract

This report selects and restricts logical options from the Fibre Channel Physical and Signalling, Fibre Channel Protocol for SCSI, Fibre Channel Arbitrated Loop, Fibre Channel Switch, and Small Computer Systems Interface standards, such that any device complying with this report should interoperate. This report addresses options for devices that are both loop-attached to the fabric and direct-attach to the fabric.

NCITS Technical Report Series

This Technical Report is one in a series produced by the American National Standards Committee, NCITS, Information Technology. The secretariat for NCITS is held by the Information Technology Industry Council (ITI), 1250 Eye Street, NW Suite 200, Washington DC 20005.

As a by-product of the standards development process and the resources of knowledge devoted to it, NCITS from time to time produces Technical Reports. Such Technical Reports are not standards, nor are they intended to be used as such.

NCITS Technical Reports are produced in some cases to disseminate the technical and logical concepts reflected in standards already published or under development. In other cases, they derive from studies in areas where it is found premature to develop a standard due to a still changing technology, or inappropriate to develop a rigorous standard due to the existence of a number of viable options, the choice of which depends on the user's particular requirements. These Technical Reports, thus, provide guidelines, the use of which can result in greater consistency and coherence of information processing systems.

When the draft Technical Report is completed, the Technical Committee approval process is the same as for a draft standard. Processing by NCITS is also similar to that for a draft standard.

PATENT STATEMENT

CAUTION: The developers of this Technical Report have requested that holder's of patents that may be required for the implementation of the Technical Report, disclose such patents to the publisher. However, neither the developers nor the publisher have undertaken a patent search in order to identify which, if any, patents may apply to this Technical Report.

As of the date of publication of this Technical Report and following calls for the identification of patents that may be required for the implementation of the Technical Report, no such claims have been made. No further patent search is conducted by the developer or the publisher in respect to any Technical Report it processes. No representation is made or implied that licenses are not required to avoid infringement in the use of this Technical Report.

Published by

American National Standards Institute
11 W. 42nd Street, New York, New York 10036

Printed in the United States of America

INSERT CODE HERE

Contents

1 Introduction and scope 1

2 Normative references 1

2.1 Approved references 2

2.2 References under development 2

2.3 Other references 2

3 Definitions and conventions 3

3.1 Definitions 3

3.2 Editorial conventions 4

3.2.1 Binary notation 5

3.2.2 Hexadecimal notation 5

3.3 Abbreviations, acronyms, and symbols 5

3.3.1 Acronyms and abbreviations 5

3.4 Applicability and use of this document 6

4 Structure and concepts 9

4.1 Summary of Private Loop versus Public Loop behaviors 9

4.2 Private Loop and Public Loop addressing 10

4.3 FL_Port and Fabric operation 11

4.3.1 Collection of frames 12

4.3.2 Delivery of frames 12

4.3.3 Open and close behavior 13

4.3.4 Services provided by the Fabric 13

4.3.5 Loop Fabric Address 14

4.3.6 Broadcast and multicast 14

4.4 NL_Port operation 15

4.5 N_Port operation with NL_Ports 15

4.6 Document structure 15

4.7 Compliance levels 16

5 NL_Port common feature sets 17

5.1 FLOGI request parameters for NL_Ports 18

5.2 PLOGI features and parameters for NL_Ports 19

5.2.1 PLOGI Common Service Parameters for NL_Ports 20

5.2.2 PLOGI Class 2 Service Parameters 21

5.2.3 PLOGI Class 3 Service Parameters 22

5.3 Other NL_Port FC-PH Features 23

5.4 P_RJT and F_BSY/P_BSY frames 25

5.5 FC-AL features 26

5.6 NL_Port initialization features 28

5.7 Extended Link Services 32

5.8 Well-known address usage by NL_Ports 34

6 Fabric and FL_Port feature sets 36

6.1 FLOGI ACC features and parameters to NL_Ports 37

6.2 Well-known addresses 39

6.2.1 Features for N_Ports at well-known addresses 41

6.2.2 PLOGI Common Service Parameters for N_Ports at well-known addresses 42

6.2.3 PLOGI Class 2 Service Parameters for N_Ports at well-known addresses 43

6.2.4 PLOGI Class 3 Service Parameters for N_Ports at well-known addresses 44

6.2.5 Other FC-PH Features for N_Ports at well-known addresses 45

6.2.6 Extended Link Services for N_Ports at well-known addresses 47

6.3 Other Fabric features 47

6.4 Fabric_Port_Reject (F_RJT) frames 48

6.4.1 F_RJT frame delimiters 48

6.4.2 Generation of F_RJT frames 48

6.5 FC-AL features for FL_Ports 50

6.6 FL_Port initialization features 52

7 N_Port feature sets 54

7.1 PLOGI features and parameters for N_Ports 55

7.1.1 PLOGI Common Service Parameters for N_Ports 56

7.1.2 PLOGI Class 2 Service Parameters 57

7.1.3 PLOGI Class 3 Service Parameters 58

7.2 Other N_Port FC-PH features 59

7.3 P_RJT and F_BSY/P_BSY frames 61

7.4 N_Port initialization features 62

7.5 Extended Link Services 64

7.6 Well-known address usage by N_Ports 66

8 Timers on a Public Loop 68

8.1 Loop Initialization Sequence Hold Time (LIS_HOLD_TIME) 68

8.2 Error Detect Timeout (E_D_TOV) 68

8.3 Resource Allocation Timeout (R_A_TOV) 68

8.4 Resource Recovery Timer (RR_TOV) 68

8.5 ULP Timer (ULP_TOV) 68

Annex A 69

A.1 Extended Link Services 69

A.1.1 Fabric Address Notification (FAN) 69

A.1.2 Loop Initialize (LINIT) 70

A.1.3 Loop Port Control (LPC) 72

A.1.4 Loop Status (LSTS) 74

A.1.5 Registered State Change Notification (RSCN) 77

A.1.6 State Change Registration (SCR) 81

Annex B 85

B.1 Introduction 85

B.2 SCSI-FCP as an FC-FLA ULP . . . 85
B.2.1 How FC-FLA Supersedes FC-PLDA . . . 85
B.2.2 Summary of Additional Features for FC-FLA SCSI Operation . . . 88
B.3 IP as an FC-FLA ULP . . . 89
Annex C . . . 91
C.1 Introduction . . . 91
C.2 Use of Registered State Change Notification to Assist I/O Probing in Switched Mass Storage Topologies . . . 91
C.2.1 Background Information . . . 91
C.2.2 Specific Requirements for a Solution . . . 91
C.2.3 Additional Tools Required by System Components . . . 92
C.2.4 Example Usage of RSCN and SCR ELS . . . 92
C.3 Name Service Usage . . . 93
C.3.1 Initial Condition of Name Service Following First Loop Initialization . . . 94
C.3.2 Implicit Registration of N_Ports and NL_Ports from FLOGI Parameters . . . 94
C.3.3 Explicit Registration of FC-4 Types . . . 95
C.3.4 Querying the Name Service . . . 96
C.3.5 Name Service Request Sequences . . . 99
C.4 FC-FLA Loop Initialization Examples . . . 109
C.4.1 Power-up Initialization Example . . . 110
C.4.2 Changed NL_Port Example . . . 112
Annex D . . . 115
D.1 Introduction . . . 115
D.2 OPN and CLS latencies . . . 115
Annex E . . . 117
E.1 Introduction . . . 117
E.2 Frame Collection Methods . . . 117
E.3 Frame Collection Delivery Triggers . . . 118

Figures

1. Private Loop and Public Loop device coexistence 10
2. NL_Port initialization flow 28
3. FL_Port initialization flow 52
4. N_Port initialization flow 62
5. Loop Initialization Example 109

Tables

1. Public NL_Port behavior 9
2. Private Loop and Public Loop addressing 11
3. FLOGI features and parameters for NL_Ports 18
4. PLOGI features for NL_Ports 19
5. PLOGI Common Service Parameters for NL_Ports 20
6. PLOGI Class 2 Service Parameters for NL_Ports 21
7. PLOGI Class 3 Service Parameters for NL_Ports 22
8. Other FC-PH features for NL_Ports 23
9. FC-AL features for NL_Ports 26
10. Extended Link Services for NL_Ports 32
11. Well-known address support for NL_Ports 34
12. FLOGI ACC features and parameters for FL_Ports 37
13. Well-known address support for Fabrics 39
14. Features for N_Ports at well-known addresses 41
15. PLOGI Common Service Parameters for well-known addresses 42
16. PLOGI Class 2 Service Parameters for well-known addresses 43
17. PLOGI Class 3 Service Parameters for well-known addresses 44
18. Other FC-PH Features for well-known addresses 45
19. Other Fabric features 47
20. F_RJT reason codes 48
21. FC-AL features for FL_Ports 50
22. PLOGI features for N_Ports 55
23. PLOGI Common Service Parameters for N_Ports 56
24. PLOGI Class 2 Service Parameters for N_Ports 57
25. PLOGI Class 3 Service Parameters for N_Ports 58
26. Other FC-PH features for N_Ports 59

27. Extended Link Services for N_Ports 64

28. Well-known address support for N_Ports 66

A.1. FAN Payload 70

A.2. LINIT Payload 71

A.3. Initialization Function 71

A.4. LINIT Accept Payload 72

A.5. LINIT Status 72

A.6. LPC Payload 73

A.7. Port Control Function 73

A.8. LPC Accept Payload 74

A.9. LPC Status 74

A.10. LSTS Payload 75

A.11. LSTS Accept Payload 75

A.12. FC-FLA Compliance Level 76

A.13. Loop State 76

A.14. RSCN Payload 79

A.15. Generic Affected N_Port ID page 79

A.16. Address Format 80

A.17. Port Address Format
- Affected N_Port ID page 80

A.18. Area Address Format
- Affected N_Port ID page 81

A.19. Domain Address Format
- Affected N_Port ID page 81

A.20. RSCN Accept Payload 81

A.21. SCR Payload 82

A.22. Registration Function 82

A.23. SCR Accept Payload 83

B.1. FC-PLDA Superseded by FC-FLA 85

C.1. Name Service - Initial State 94

C.2. Name Service - Implicit FLOGI Registration 94

C.3. Example FLOGI Request Payload, NL_Port “A” 95

C.4. Name Service - Explicit FC-4 Registration 95

C.5. Example RFC-4 Request Payload, N_Port “B” 96

C.6. Example RFC-4 Request Payload, N_Port “B” 96

C.7. Example GP_ID4 Request Payload, Nx_Ports 97

C.8. Example GP_ID4 Response Payload, Nx_Ports 97

C.9. Example GP_ID4 Request Payload, LFAs 98

C.10. Example GP_ID4 Response Payload, LFAs 98

C.11. Example GFC-4 Request Payload 99

C.12. Example GFC-4 Response Payload 99

C.13. RFC-4 Request Header and Payload 100

C.14. RFC-4 Response Header and Payload 102

C.15. GP_ID4 Request Header and Payload 103

C.16. GP_ID4 Response Header and Payload 104

C.17. GPT Request Header and Payload 106

C.18. GPT Response Header and Payload 108

C.19. Power-up Initialization Activities 110

C.20. NL_Port Swap Activities 112

E.1. Frame Collection Algorithms 117

draft proposed NCITS Technical Report
for Information Technology—

Fibre Channel —
Fabric Loop Attachment (FC-FLA)

1 Introduction and scope

This Technical Report specifies Fibre Channel options between Public NL_Ports and FL_Ports, between Public NL_Ports and other Public NL_Ports (whether on the same or different loops), and between Public NL_Ports and N_Ports. This report also provides some direction for SCSI and IP options for communication across a fabric between NL_Ports and N_Ports.

This report is intended to serve as an implementation guide whose primary objective is to maximize the likelihood of interoperability between conforming implementations. This report prohibits or requires features that are optional, and prohibits the use of some non-optional features in the referenced ANSI standards.

A second objective of this Technical Report is to simplify implementations and their associated documentation, testing, and support requirements. This means that there will be some optional features which are not mutually exclusive, but are still prohibited or required solely for the purpose of this simplification.

Internal characteristics of conformant implementations are not defined by this Technical Report. This Technical Report incorporates features from the standards described below. Where needed, changes are or have been proposed to the appropriate ANSI NCITS technical committees to ensure this Technical Report remains a strict subset of ANSI standards.

2 Normative references

The following standards contain provisions which, through reference in the text, constitute provisions of this standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed below.

Copies of the following documents can be obtained from ANSI: Approved ANSI standards, approved and draft international and regional standards (ISO, IEC, CEN/CENELEC), and approved foreign standards (including BSI, JIS, and DIN). For further information, contact ANSI Customer Service Department at 212-642-4900 (phone), 212-302-1286 (fax) or via the World Wide Web at http://www.ansi.org.

Additional availability contact information is provided below as needed.

2.1 Approved references

[1] ANSI X3.230-1994, *Information Technology - Fibre Channel Physical and Signaling Interface (FC-PH)*

[2] ANSI X3.272-1996, *Information Technology - Fibre Channel Arbitrated Loop (FC-AL)*

[3] ANSI X3.131-1994, *Information Technology - Small Computer Systems Interface - 2 (SCSI-2)*

[4] ANSI X3.269-1996, *Information Technology - Fibre Channel Protocol for SCSI (FCP)*

[5] ANSI X3.270-1996, *Information Technology - SCSI-3 Architecture Model (SAM)*

[6] ANSI X3.288-1996, *Information Technology - Fibre Channel - Generic Services (FC-GS)*

[7] ANSI X3.289-1996, *Information Technology - Fibre Channel - Fabric Generic (FC-FG)*

[8] ANSI X3.297-1997, *Information Technology - Fibre Channel - Physical and Signalling Interface-2 (FC-PH-2)*

[9] ANSI X3.230-1994/AM1-1996, *Information Technology - Fibre Channel Physical and Signaling Interface Ammendment 1 (FC-PH-AM1)*

2.2 References under development

At the time of publication, the following referenced standards were still under development. For information on the current status of the document, or regarding availability, contact the relevant standards body or other organization as indicated.

NOTE – For more information on the current status of a document, contact NCITS at the address listed in the front matter. To obtain copies of this document, contact Global Engineering at the address listed in the front matter, or NCITS.

[10] ANSI X3.303-199x, *Fibre Channel - Physical and Signalling Interface-3 (FC-PH-3)*, T11/Project 1119D/Rev 9.2

[11] ANSI X3.xxx-199x, *Fibre Channel Arbitrated Loop (FC-AL-2),* T11/Project 1133D/Rev 5.6

[12] ANSI X3.xxx-199x, *Fibre Channel - Generic Services-2 (FC-GS-2)*, T11/Project 1134D/Rev 0.1

[13] ANSI X3.xxx-199x, *Fibre Channel - Switched Fabric (FC-SW)*, T11/Project 959D/Rev 3.2

[14] ANSI TR.xxx-199x, *Fibre Channel - Private Loop Direct Attach (FC-PLDA)*, T11/Project 1162-DT/Rev 1.11

2.3 Other references

All of the following profiles are available from the Fibre Channel Association (FCA), 12407 MoPac Expressway North 100-357, P. O. Box 9700, Austin, TX 78758-9700; (800) 272-4618 (phone); or via e-mail, FCA-Info@amcc.com.

[15] FCSI-101*, FCSI Common FC-PH Feature Sets Used in Multiple Profiles*, Rev 3.1

[16] FCSI-201*, FCSI SCSI Profile*, Rev 2.2

[17] FCSI-202, *FCSI IP Profile*, Rev 2.1

[18] *FCA N_Port to F_Port Interoperability Profile,* Rev 1.0

[19] *FCA Internet Protocol Profile*, Rev XX

3 Definitions and conventions

For FC-FLA, the following definitions, conventions, abbreviations, acronyms, and symbols apply.

3.1 Definitions

3.1.1 Byte: A group of eight bits.

3.1.2 Collation: The action by an FL_Port of collecting frames for delivery at a later time in a single Loop Tenancy to a given NL_Port.

3.1.3 Directory Server: The entity at well-known address hex 'FFFFFC'. See reference [1].

3.1.4 Directory Service Application: A Fibre Channel Service provided by the Directory Server. This Service is indicated by an FCS_Type of hex 'FC'. See reference [6].

3.1.5 F_Port: As defined in FC-PH (see reference [1]). In this report, an F_Port is assumed to always refer to a port to which non-loop N_Ports are attached to a fabric, and does not include FL_Ports.

3.1.6 Fabric F_Port: The entity at the well-known address hex 'FF FF FE'. See reference [1].

3.1.7 FL_Port: An L_Port that is able to perform the function of an F_Port, attached via a link to one or more NL_Ports in an Arbitrated Loop topology. The AL_PA of an FL_Port is hex '00'. In this report, an FL_Port is assumed to always refer to a port to which NL_Ports are attached to a fabric, and does not include F_Ports.

3.1.8 Frame Bundling: A process whereby the Fabric collects frames for a source and destination pair before attempting to deliver the collected frames within a single Loop Tenancy to the destination NL_Port.

3.1.9 Frame Piling: A process whereby the Fabric collects frames for a destination from any source before attempting to deliver the collected frames within a single Loop Tenancy to the destination NL_Port.

3.1.10 L_Port: A port which contains Arbitrated Loop functions associated with the Arbitrated Loop topology.

3.1.11 Local Loop: The Physical Loop to which an L_Port is directly attached.

3.1.12 Local Port: Any L_Port connected to the Local Loop. A Local Port can only be reached via an OPN of the Device's AL_PA.

3.1.13 Loop Device: A device with at least one NL_Port.

3.1.14 Loop Fabric Address: A 24-bit ID used to address a loop for purposes of loop management. The format of this address is Domain || Area || hex '00'. See annex A.

3.1.15 Loop Tenancy: The period of time beginning when an L_Port send or receives OPN and ending when an L_Port has both transmitted CLS and received CLS.

3.1.16 N_Port: As defined in FC-PH. In this report, an N_Port is assumed to always refer to a direct fabric-attached port, and does not include NL_Ports.

3.1.17 NL_Port: An L_Port that is able to perform the function of an N_Port, attached via a link to one or more NL_Ports and zero or more FL_Ports in an Arbitrated Loop topology. In this report, an NL_Port is assumed to always refer to a loop-attached port, and does not include N_Ports.

3.1.18 Name Service: A specific Fibre Channel Service provided by the Directory Server. This Service is indicated by an FCS_Subtype of hex '02'. See reference [6] and reference [12].

3.1.19 OPN Originator: The L_Port on an Arbitrated Loop that enters the OPEN state.

3.1.20 OPN Recipient: The L_Port on an Arbitrated Loop that enters the OPENED state.

3.1.21 Physical Loop: An Arbitrated Loop as defined in the FC-AL standard. It is useful to note that a Physical Loop can be thought of as a set of devices and the interconnects between them arranged such that any appropriate FC-PH or FC-AL frame or primitive may be sent from one device, pass through every device once, and then return to the sending device. What is important is the loop functionality, rather than the nature of the physical interconnect and routing between the devices.

3.1.22 Private Loop device: As defined in FC-PLDA.

3.1.23 Private NL_Port: An NL_Port that is observing the rules of Private Loop behavior (see clause 4).

3.1.24 Public Loop device: As defined in FC-PLDA.

3.1.25 Public NL_Port: An NL_Port that can observe the rules of either Public Loop or Private Loop behavior (see clause 4). A Public NL_Port may have open Exchanges with both Private and Public NL_Ports concurrently.

3.1.26 Query: Any of the "Get" service requests provided by the Name Service (see annex A).

3.1.27 Remote Loop: The Physical Loop separate from the Local Loop to which the other communicating NL_Port is directly attached. A Remote Loop can only be reached across the Fabric.

3.1.28 Remote Port: Any NL_Port not connected to the Local Loop, or any N_Port. A Remote Port can only be reached across the Fabric. An NL_Port can reach a Remote Port only via an OPN of AL_PA hex '00'. All N_Ports and NL_Ports are Remote Ports relative to a given N_Port.

3.1.29 X.500 Service: A specific Fibre Channel Service provided by the Directory Server. This Service is indicated by an FCS_Subtype of hex '01'. See reference [6].

3.2 Editorial conventions

In this Technical Report, a number of conditions, mechanisms, sequences, parameters, events, states, or similar terms that do not have their normal English meaning are printed with the following conventions:

– the first letter of each word in uppercase and the rest lowercase (e.g., Exchange, Class, etc.).

– a term consisting of multiple words, with the first letter of each word in uppercase and the rest lowercase, and each word separated from the other by an underscore (_) character. A word may consist of an acronym or abbreviation which would be printed in uppercase. (e.g., NL_Port, Transfer_Length, etc.).

All terms and words not conforming to the conventions noted above have the normal technical English meanings.

Numbered items in this Technical Report do not represent any priority. Any priority is explicitly indicated.

In all of the figures, tables, and text of this standard, the most significant bit of a binary quantity is shown on the left side. Exceptions to this convention are indicated in the appropriate sections.

The term “shall” is used to indicate a mandatory rule. If such a rule is not followed, the results are unpredictable unless indicated otherwise.

The fields or control bits that are not applicable shall be set as required by the appropriate standard.

If a field or a control bit in a frame is specified as not meaningful, the entity that receives the frame shall not check that field or control bit.

In several tables within this document, there is a column on the right side of the table labelled “Notes”. These notes are NORMATIVE and shall be considered requirements of this document.

In the event of conflict between the text, tables, and figures in this document, the following precedence shall be used: text, tables, figures.

3.2.1 Binary notation

Binary notation may be used to represent some fields. Single bit fields are represented using the binary values 0 and 1. For multiple bit fields, the binary value is enclosed in single quotation marks followed by the letter b. For example, a four-byte Process_Associator field containing a binary value may be represented as ‘00000000 11111111 10011000 11111010’b.

3.2.2 Hexadecimal notation

Hexadecimal notation may be used to represent some fields. When this is done, the value is enclosed in single quotation marks and preceded by the word hex. For example, a four-byte Process_Associator field containing a binary value of ‘00000000 11111111 10011000 11111010’b is shown in hexadecimal format as hex ’00 FF 98 FA’.

3.3 Abbreviations, acronyms, and symbols

Abbreviations and acronyms applicable to this Technical Report are listed. Definitions of several of these items are included in 3.1. Abbreviations and acronyms are commonly used terms defined in referenced standards (e.g., LIP is defined in FC-AL).

3.3.1 Acronyms and abbreviations

A_W_TOV	Arbitration_Wait_Timeout value
BLS	Basic Link Service
E_D_TOV	Error_Detect_Timeout value
ELS	Extended Link Service

FAN	Fabric Address Notification
FCA	Fibre Channel Association
FC-AL	Fibre Channel Arbitrated Loop, reference [2]
FC-AL-2	Fibre Channel Arbitrated Loop-2, reference [11]
FC-FG	Fibre Channel - Fabric Generic, reference [7]
FC-GS	Fibre Channel - Generic Services, reference [6]
FC-GS-2	Fibre Channel - Generic Services-2, reference [12]
FCLC	Fibre Channel Loop Community
FC-PH	Fibre Channel Physical and Signaling Interface, reference [1]
FC-PH-2	Fibre Channel Physical and Signaling Interface-2, reference [8]
FC-PH-3	Fibre Channel Physical and Signaling Interface-3, reference [10]
FC-PLDA	Fibre Channel - Private Loop Direct Attach, reference [14]
FC-SW	Fibre Channel - Switched Fabric, reference [13]
FCP	Fibre Channel Protocol, reference [4]
FCSI	Fibre Channel Systems Initiative
GP_ID4	Get Native Port Identifiers based on Port_Type Name Service
GPT	Get Port_Type Name Service
IP	Internet Protocol
IU	Information Unit
LAN	Local Area Network
LFA	Loop Fabric Address
LINIT	Loop Initialize Extended Link Service
LIS_HOLD_TIME	Loop Initialization Sequence Hold Time
LLC	Logical Link Control
LPC	Loop Port Control Extended Link Service
LSTS	Loop Status Extended Link Service
MAC	Media Access Control
NFS	Network File System or Network File Server
R_A_TOV	Resource_Allocation_Timeout value
RFC-4	Register FC-4 Types Name Service
RR_TOV	Resource_Recovery_Timeout value
RSCN	Registered State Change Notification Extended Link Service
SCR	State Change Registration Extended Link Service
SCSI	Small Computer System Interface, reference [3]
SI	Sequence Initiative
ULP	Upper Level Protocol
ULP_TOV	Upper_Level_Protocol_Timeout value
WAN	Wide Area Network
WWN	World Wide Name

3.3.2 Symbols

Unless indicated otherwise, the following symbols have the listed meaning.

||	concatenation

3.4 Applicability and use of this document

Since the nature of this document is a profile, the usual definitions of the following words do not apply! Please read these definitions carefully!

Prohibited: If a feature is Prohibited, it means that it shall not be used between compliant implementations. An implementation may use the feature to communicate with non-compliant implementations.

This document does not prohibit the implementation of features, only their use between compliant implementations. However, interoperability is not guaranteed if Prohibited features are used.

Required: If a feature or parameter value is Required, it means that it shall be used between compliant implementations. Compliant implementations are required to implement the feature. An implementation may use the feature or other features to communicate with non-compliant implementations. Interoperability is not guaranteed if Required features are not implemented.

Allowed: If a feature or parameter value is Allowed, it means that it may be used between compliant implementations. Compliant implementations are not required to implement the feature, but if they do, the feature shall be implemented as described in this document. Typically, the potential user of a feature may determine if the potential recipient supports that feature via a Required discovery process.

Invocable: If a feature or parameter value is Invocable, it means that it may be used between compliant implementations. Compliant implementations are required to implement the feature, and make available the use of the feature. Invocable is different than Allowable or Required in that an originator may invoke the feature if needed, but the originator is not required to invoke it, and may never need to. Typically, an Invocable feature is Required for implementation by the recipient of the feature.

The tables in the following clauses list features described in the various standards specific to the operations described in the clause. These tables indicate whether the feature is Required, Prohibited, Invocable, or Allowed for compliance with this report; or whether a parameter is Required to be a particular value for compliance with this report. **Features or parameters that are not listed do not affect the interoperability of Public Loop devices**.

The following legend is used for table entries in these clauses:

'P' the implementation is Prohibited from using the specified feature
'R' the implementation is Required to support the specified feature
'A' use of the specified feature is Allowed
'I' the implementation may Invoke the specified feature
'n' the parameter shall be set to this value
'X' this parameter has no required value; any value is allowed
'-' this parameter or feature is not meaningful

4 Structure and concepts

This clause provides an overview of a fabric-attach loop topology environment that allows NL_Ports and N_Ports to communicate across a Fabric without regard to the details of the topology. This environment also provides for efficient use of the loop topology by the FL_Port.

4.1 Summary of Private Loop versus Public Loop behaviors

The definition of Public NL_Port behavior is as follows (see FC-PLDA for a similar table defining Private NL_Port behavior):

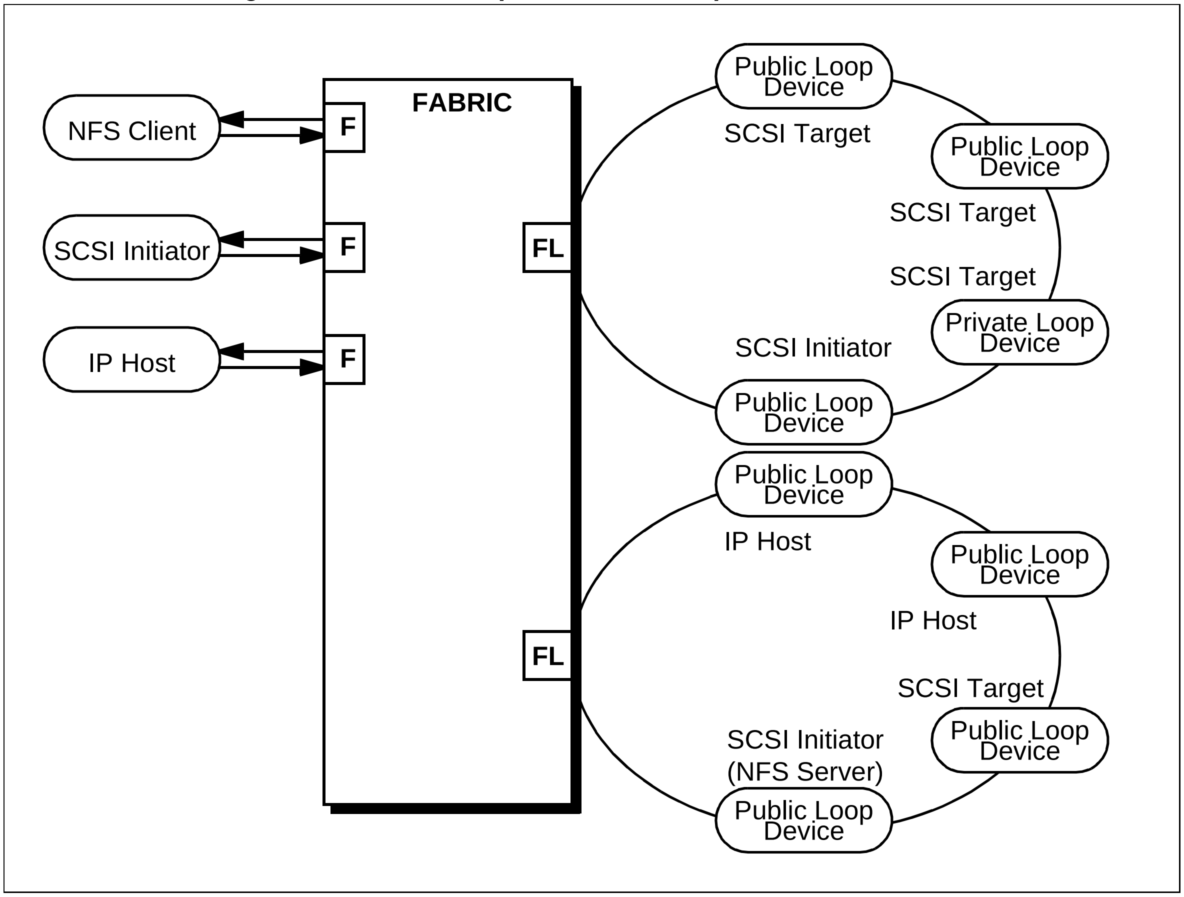
Table 1 – Public NL_Port behavior

Behavior	Public NL_Port
Domain + Area of device's NL_Port ID=hex '0000'	Prohibited
NL_Port may Open AL_PA=hex '00'	Required
NL_Port may respond to AL_PA=hex '00' (NL_Port operating as an F/NL_Port)	Prohibited
FL_Port may Open NL_Port	Required
F/NL_Port may Open NL_Port	Prohibited
Public NL_Port may Open any Local NL_Port	Allowed
Private NL_Port may Open any Local NL_Port	Allowed

A port which **only** exhibits Private Loop behavior is called a Private NL_Port. A port that exhibits Public behavior is called a Public NL_Port, even though it may communicate with Private NL_Ports. For example, a Public NL_Port in Figure 1 may be an NFS server which communicates with NFS Clients residing directly on the fabric using IP, and with SCSI Targets that are Local Ports on the same loop using FCP.

Devices with only private NL_Ports are called private loop devices. A private loop device is Prohibited from providing or requesting fabric services. Devices with at least one Public NL_Port are called public loop devices.

Figure 1 – Private Loop and Public Loop device coexistence



4.2 Private Loop and Public Loop addressing

A 24-bit NL_Port native port address identifier is divided into three 8 bit fields. In order of bit significance these are: Domain, Area, and Port. The AL_PA of an NL_Port corresponds to its Port address. No two ports on the same Local Loop shall share the same AL_PA. All Public NL_Ports on the same Local Loop shall have the same Domain || Area.

NOTE – The FC-PLDA report states that “the Domain || Area of a private NL_Port is zero, and that a private NL_Port shall accept frames from NL_Ports which have any value for their Domain || Area in the S_ID of the frame header”. This does not imply that private NL_Ports should expect to receive frames across a Fabric.

The following table summarizes how S_ID and D_ID and OPNs shall be created when sending frames. The S_ID and D_ID are created by the sender of the frame, and the sender performs an OPN on the source loop. The FL_Port performs the OPN on the destination loop; this OPN is received by the frame recipient. If both source and destination are on the same Local Loop, the OPN shall be performed by the frame sender only.

Table 2 – Private Loop and Public Loop addressing

Frame sent from (note 1)	Frame received by (note 1)	OPN AL_PD		S_ID (note 1)		D_ID (note 1)	
		source loop	dest. loop	23:8	7:0	23:8	7:0
Public NL_Port	Fabric-Attached N_Port	hex ’00’	n/a	Local D&A	Source AL_PA	N_Port ID	
Fabric-Attached N_Port	Public NL_Port	n/a	D_ID 7:0	N_Port ID		Remote D&A	Dest. AL_PA
Local Public NL_Port	Remote Public NL_Port	hex ’00’	D_ID 7:0	Local D&A	Source AL_PA	Remote D&A	Dest. AL_PA
Remote Public NL_Port	Local Public NL_Port	hex ’00’	D_ID 7:0	Remote D&A	Source AL_PA	Local D&A	Dest. AL_PA
Local Public NL_Port	Local Public NL_Port	D_ID 7:0 **(note 2)**		Local D&A	Source AL_PA	Local D&A	Dest. AL_PA
Local Public NL_Port	Local Private NL_Port	D_ID 7:0		Local D&A	Source AL_PA	hex ’0000’	Dest. AL_PA
Local Private NL_Port	Local Public NL_Port	D_ID 7:0		hex ’0000’	Source AL_PA	Local D&A	Dest. AL_PA
Local Private NL_Port	Local Private NL_Port	D_ID 7:0		hex ’0000’	Source AL_PA	hex ’0000’	Dest. AL_PA

NOTES:

1 “D&A” refers to the Domain and Area; “Local” means the Domain and Area on the Local Loop, “Remote” means the Domain and Area of the Remote Loop.

2 The behavior of an FL_Port when it receives a unicast frame from a Local NL_Port destined for another Local NL_Port is not defined by this report.

4.3 FL_Port and Fabric operation

The FL_Port is the gateway to the Fabric for the Public NL_Ports on its Local Loop, allowing NL_Ports to act as if they were N_Ports and interact with other N_Ports and NL_Ports attached to the fabric.

NOTE – For purposes of this report we restrict the meaning of N_Port to “direct Fabric-attached N_Port”. If a device is loop-attached, it is an NL_Port. When we refer to a characteristic that applies to either, we will explicitly refer to both. Similarly, we restrict the meaning of F_Port to “fabric port to which an N_Port (as defined above) is attached”. If NL_Ports are attached to the fabric port, it is an FL_Port. See 3.1.5, 3.1.7, 3.1.16, and 3.1.17.

An FL_Port has a responsibility to use the loop efficiently; delivering one frame per Loop Tenancy in many cases is probably not very efficient. Therefore, the FL_Port may choose to collect (collate) frames using an appropriate algorithm, and to attempt to deliver the collected frames to the destination within a single Loop Tenancy. Note that collecting a number of frames at the FL_Port for delivery to an NL_Port does not imply that the NL_Port has buffer credit available to receive the frames; rather, the FL_Port can only attempt to deliver the collected frames to the extent that the NL_Port can accept them in a given Loop Tenancy.

Note also that FL_Ports may choose to not collect frames as described above. The ability to collect frames in this manner is not a requirement for compliance to this report.

The FL_Port also manages the initialization and control of the Local Loop to the extent that an N_Port can interoperate with the NL_Ports on the Local Loop as if those NL_Ports were direct fabric-attached devices. The FL_Port performs LIPs and AL_PA assignments transparently to the rest of the fabric. N_Ports and NL_Ports may do discovery of other N_Ports and NL_Ports via the Name Service.

The fabric may provide optional Extended Link Services so that an N_Port may exercise more control over a Remote Loop. These include services to get the status of a loop, to force the initialization of a loop, and to control the bypass of loop devices. See 4.3.4.

4.3.1 Collection of frames

Frames may be collected using a variety of different algorithms. This subclause will provide a survey of the most interesting methods identified during the development of this report. Other slightly less interesting methods are cataloged in Annex E.

Frame Piling is the most general case, and most closely resembles classic N_Port to N_Port behavior, with some added latency introduced by the accumulation process and other loop-induced latencies. Frames are accumulated for an NL_Port destination from all sources and delivered to the destination in its next turn.

Frame Bundling is a special case of Frame Piling that accumulates all frames from a single source, independent of SEQ_ID or X_ID. Frame Bundling can be useful in situations where frame multiplexing can cause a reduction in performance at the receiving end, provided that the NL_Port destination login with each source allows only one concurrent Sequence.

4.3.2 Delivery of frames

The FL_Port, after accumulating frames, then delivers the frames to the destination. There is a tradeoff between the latency induced by accumulation and the inefficiency induced by delivering too few frames per Loop Tenancy. Also, the FL_Port ensures that frames get delivered in a timely manner, even if the preferred “trigger” is not achieved. The following lists some interesting trigger mechanisms for frame delivery that were identified during development of this document:

- Frame delivery may be triggered when some number of frames have been accumulated using a collation algorithm. The disadvantage of this trigger is that the source may send fewer frames than the trigger level and have no more to send. This issue can be overcome by coupling this trigger with another more deterministic trigger.

- Delivery can be triggered when an end of sequence is detected.

- Delivery can be triggered when the accumulated frames have become “old”. A time stamp can be recorded for each collection of frames. When the time stamp passes an age threshold, delivery is triggered.

- Delivery can be triggered if the fabric is becoming congested, and by delivering the accumulated frames, the fabric can relieve the congestion without losing or busying frames.

- Delivery can be triggered based on the contents of the frame header or payload.

– Delivery can be triggered whenever the loop is not busy and the FL_Port has frames to send. "Loop not busy" is defined as no NL_Ports actively arbitrating for the loop.

4.3.3 Open and close behavior

The FL_Port delivers frames by arbitrating for the loop, opening the destination NL_Port, transmitting the frames, and closing the loop. As allowed by FC-AL, the FL_Port may arbitrate unfairly, meaning that the FL_Port may enter the ARBITRATING state at any time without regard to access fairness. When the loop is released by an NL_Port, the FL_Port will always win when it is arbitrating.

Once the FL_Port has control of the loop, it may retain control to deliver frames to more than one destination NL_Port by using the TRANSFER state. When the FL_Port finishes the CLS with a destination NL_Port, it holds onto control of the loop and opens another destination NL_Port. The FL_Port should weigh this control against the need of other ports to use the loop. One possible strategy is to retain control until an NL_Port is observed to be arbitrating, and then release the loop after completing some additional amount of work. In the case of the FAN ELS, the FL_Port is Required to deliver frames to all attached Public NL_Ports using the TRANSFER state.

If an FL_Port sends an OPN, and receives that same OPN, the FL_Port shall assume the NL_Port has ceased to exist and shall perform an implicit logout of the NL_Port. If the FL_Port receives an immediate CLS from the NL_Port in response to the OPN, and the FL_Port is operating with that NL_Port with a Login BB_Credit of zero, it shall assume that NL_Port is busy, and reschedule the frame(s) for delivery at a later time, subject to the usual timeout limits.

4.3.4 Services provided by the Fabric

The fabric shall provide support for the following additional Extended Link Service (documented in Annex A):

– Fabric Address Notification (FAN): Sent by an FL_Port to inform all attached Public NL_Ports of the current fabric address information.

The fabric may provide support for the following additional Extended Link Services (documented in Annex A):

– Registered State Change Notification (RSCN): Sent when a change occurs at an N_Port or NL_Port.

– State Change Registration (SCR): Allows an N_Port or NL_Ports to indicate that it would like to receive the Registered State Change Notification (RSCN) ELS.

A fabric may provide additional Extended Link Services that allow an N_Port or NL_Port to manage and control a Remote Loop (documented in Annex A):

– Loop Initialize (LINIT): Allows an N_Port or NL_Port to request that a LIP (of any type) be sourced by the FL_Port on a Remote Loop.

– Loop Port Control (LPC): Allows an N_Port or NL_Port to request that LPEs and LPBs (of any type) be sourced by the FL_Port on a Remote Loop.

– Loop Status (LSTS): Allows an N_Port or NL_Port to determine the current condition of a Remote Loop.

The fabric shall provide support for the following FC Common Services Required for operation on or with a Public Loop, and shall be implemented by FL_Ports and N_Ports and NL_Ports that comply with this report:

– Name Service Register FC-4 Type (RFC-4): Used by N_Ports and NL_Ports to inform the Name Service of the FC-4 TYPE(s) supported.

– Name Service Get Native Port Identifiers for Port Type (GP_ID4): Used by N_Ports and NL_Ports to get a list of N_Ports and NL_Ports that have completed FLOGI.

– Name Service Get Port Type (GPT): Used by N_Ports and NL_Ports to determine the Port_Type of a given port.

Other Name Service services may be supported by the Fabric, and attempted by N_Ports and NL_Ports. A fabric that does not provide a requested registration service shall return an FS_RJT with a Reason Code of “Command not supported” and an Explanation Code of “No additional explanation”. Annex C contains examples of Name Service services.

4.3.5 Loop Fabric Address

The Loop Fabric Address (LFA, see 3.1.14) is an identifier used to address a loop for purposes of loop management. The LFA is used as the destination ID (D_ID) in the LINIT, LPC, and LSTS request Sequences, and is used as the source ID (S_ID) in the response Sequences. No other Sequences shall be directed to an LFA.

Additional requirements for LFA for each ELS are detailed in Annex A.

An N_Port or NL_Port may discover an LFA in one of several ways:

– the N_Port or NL_Port may query the Name Service using the GPT request for the FL_Port Port_Type;

– the N_Port or NL_Port may assume that any identifier of the form “(ddaa)||(hex ’00’)” is an LFA (and if it is not, the receiving port may respond with an LS_RJT response Sequence);

– the N_Port or NL_Port may calculate the LFA of a loop by computing the logical bitwise AND of the native port address of an NL_Port on the loop with hex ’FFFF00’;

– the N_Port or NL_Port has implicit knowledge of the LFA in a manner not defined by this document.

4.3.6 Broadcast and multicast

As defined by FC-AL, an FL_Port shall not receive frames delivered to the Local Loop by an NL_Port using OPNfr or OPNyr. An FL_Port may receive broadcast and multicast frames from an attached NL_Port during a Loop Tenancy initiated by OPNyx or OPNyy, and shall deliver these frames to the addressed destination group, including the Local Loop if any ports on the Local Loop are part of the addressed destination group.

An FL_Port shall deliver broadcast or multicast frames to its Local Loop via the OPNfr Primitive Signal, as defined in FC-AL. The use of OPNyr by the FL_Port is prohibited.

4.4 NL_Port operation

An NL_Port attached to a Public Loop is expected to be either:

- a Private NL_Port, which communicates only with other NL_Ports on the Local Loop, and does not communicate with the FL_Port; or,
- a Public NL_Port, which may communicate with other NL_Ports on the Local Loop, and may communicate with the FL_Port for purposes of communicating with Remote Ports.

An NL_Port attached to a Public Loop may not be an F/NL_Port (an NL_Port which is able to respond to AL_PA hex '00'), since that would conflict with the duties of the FL_Port. The operation of a Private NL_Port is covered in FC-PLDA (see reference [14]).

A Public NL_Port is expected to behave as if it were a direct-fabric-attached N_Port to the extent that it is possible. The fabric and FL_Ports hide much of the "loop-ness" of the NL_Port from Remote Ports. The Public NL_Port also does its share by performing the appropriate logins and registrations with fabric services. If a Remote Port wishes to deal with the loop nature of the NL_Port, it may via use of the various ELS noted in 4.3.4; use of these additional services is not Required for normal operation.

An NL_Port also has the responsibility to use the loop efficiently, and to release the loop when it has no more frames to send. The exact methods by which an NL_Port decides when to OPN and when to CLS are not addressed by this document.

NL_Ports shall be able to receive frames in a multiplexed manner, meaning that frames from one Sequence may be mixed with frames from other Sequences, and possibly from other sources, even within the same Loop Tenancy. An NL_Port may, however, expect frames to be delivered by the Fabric in-order from a single source, meaning that they are delivered to the destination in the order they were received from the source (this is the definition of "in-order" in FC-PH). This is guaranteed even if Class 2 frames are busied by the Fabric; the source is Prohibited from retransmitting Class 2 frames to which a BSY response is received, if the retransmit would result in an out-of-order delivery. Sequences may be re-tried if the frame cannot be retransmitted.

NL_Ports shall accept the FAN ELS request Sequence and shall attempt the RFC-4 Common Services request Sequence (see 4.3.4).

4.5 N_Port operation with NL_Ports

The intent of this Technical Report is to provide a Fabric environment such that a direct fabric-attached N_Port need not be aware that the other N_Port with which it is communicating is in fact a loop-attached NL_Port. Some NL_Port behaviors defined in this document are also applied to N_Ports in those cases where the behavior is necessary for interoperability.

Some tools have been provided within this report that can facilitate the management of Public NL_Ports in error recovery and initialization. These tools should be considered optional enhancements, and are not Required for inter-operation of compliant implementations.

4.6 Document structure

This report is structured such that the features of NL_Ports, FL_Ports, and N_Ports are separate from the features of the Upper Level Protocols. This structure will allow the use of compliant ports with ULPs not covered by this report. This has been done to increase the extensibility of Public Loop implementations.

4.7 Compliance levels

FC-FLA has multiple compliance levels. All devices that comply with this report at least meet compliance level “A” (referred to as FLA-A). Some devices provide additional features and therefore meet compliance level “B” (referred to as FLA-B). Features Required for FLA-B are noted where they are defined.

NOTE – In some cases, mixing FLA-A and FLA-B devices may cause all devices on a loop to operate as FLA-A devices, because an FLA-A device is not able to propagate the FLA-B behavior.

5 NL_Port common feature sets

The tables in this clause list features described in the Fibre Channel Physical and Signaling Interface standard (FC-PH) and in the Fibre Channel Arbitrated Loop (FC-AL), specific to the operation of NL_Ports. These tables indicate whether the feature is Required, Prohibited, Allowed, or Invocable for compliance with this Technical Report. Features that are not listed do not affect interoperability of public loop devices.

PLEASE READ 3.4 for the exact definition of Prohibited, Required, Allowed, and Invocable, and for the legend for table entries used in this clause.

5.1 FLOGI request parameters for NL_Ports

This clause specifies the features used by an NL_Port when sending a FLOGI request. Table 3 lists the features used and parameters requested by NL_Ports with usage defined by this document. In this table, “NL_Port Originator” refers to an NL_Port issuing a FLOGI request to an FL_Port. See 6.1 for the requirements for the FLOGI ACC response.

Table 3 – FLOGI features and parameters for NL_Ports

Feature/Parameter	NL_Port Originator	Notes
FLOGI S_ID	hex ’0000’||AL_PA]	1
Class of service for FLOGI	3	
FC-PH Version Highest Version Lowest Version	 X hex ’20’	 5
BB_Credit (min)	0	6
Valid Vendor Version Level	0	2
N_Port/F_Port	0	
Alternate BB_Credit Management	1	
Maximum BB Receive Data Field Size (min)	256	
FLOGI Payload Length	X	3
Supported Classes of Service: Class 1 Class 2 Class 3 Class 4 and up	 P A R P	
Sequential Delivery bit Class 2 Class 3	 1 1	 4 4
Priority/Preemption in Class 2	0	

NOTES:

1 The lower 8-bits of the S_ID is set to the AL_PA value of the NL_Port, which the NL_Port received during loop initialization. The fabric retains this value for the lower 8-bits of the assigned ID, and uses this value to generate the AL_PD when it returns the FLOGI ACC.

2 Profile versions are communicated via the Report Node Capabilities (RNC) ELS. The Valid Vendor Version Level shall not be set to 1 for the purposes of communicating support of this Technical Report.

3 The NL_Port is Allowed to set this bit to 1 to receive a 256 byte FLOGI ACC payload that contains the Services Availability field.

4 While FC-PH may require that the frames travel a “fixed route” across the fabric, this document does not particularly care how the function is implemented, as long as it is performed correctly.

5 The value hex ’20’ indicates FC-PH-3.

6 An NL_Port is Required to support a value of 0 or greater for this parameter.

5.2 PLOGI features and parameters for NL_Ports

This clause specifies the features and parameters sent by an NL_Port to an NL_Port or N_Port for a PLOGI request or PLOGI ACC. Table 4 lists the features supported by NL_Ports with usage defined by this document. In this table, “NL_Port Originator” refers to an NL_Port issuing a PLOGI request to another N_Port or NL_Port, and “NL_Port Responder” refers to an NL_Port issuing a PLOGI ACC to an N_Port or NL_Port that sent a PLOGI request.

Table 4 – PLOGI features for NL_Ports

Feature	NL_Port Originator	NL_Port Responder	Notes
Class of service for PLOGI (NL_Port Originator) and PLOGI ACC (NL_Port Responder) Class 1 Class 2 Class 3 Class 4 and up	 P I R P	 P A R P	1
Supported Classes of Service: Class 1 Class 2 Class 3 Class 4 and up	 P A R P	 P A R P	2

NOTES:

1 If an NL_Port receives a PLOGI in a Class of Service other than Class 3, and the NL_Port does not support PLOGI in that Class, it shall return a P_RJT with an Action Code of “Rejected frame may be retryable” and a Reason Code of “Class not supported”. The NL_Port shall tolerate this behavior; the expectation is that an older N_Port is trying Class 2 first before trying Class 3.

2 An NL_Port shall not advertise support for a Class of Service that is not supported by the fabric (as indicated by FLOGI ACC) when sending a PLOGI or PLOGI ACC to any N_Port or NL_Port that is not attached to the Local Loop.

5.2.1 PLOGI Common Service Parameters for NL_Ports

Table 5 lists PLOGI Common Service Parameters for NL_Ports with usage defined by this document. In this table, “NL_Port Originator or Responder” refers to either an NL_Port issuing a PLOGI request to another N_Port or NL_Port, or to an NL_Port issuing a PLOGI ACC to an N_Port or NL_Port that sent a PLOGI request.

Table 5 – PLOGI Common Service Parameters for NL_Ports

Common Service Parameter	NL_Port Originator or Responder	Notes
FC-PH Version Highest Version Lowest Version	 X hex '20'	
BB_Credit (min)	0	4
Continuously Increasing Relative Offset	1	
Random Relative Offset	X	5
Valid Vendor Version Level	0	1
N_Port/F_Port	0	
Alternate BB_Credit Management	1	
E_D_TOV Resolution	0	2
Payload Length	0	
BB Receive Data Field Size (min)	256	3
Total Concurrent Sequences (min)	1	
Relative Offset by Information Category	X	5
E_D_TOV	e_d_tov	2

NOTES:

1 Profile versions are communicated via the Report Node Capabilities (RNC) ELS. The Valid Vendor Version Level shall not be set to 1 for the purposes of communicating support of this Technical Report.

2 In FC-AL, E_D_TOV is meaningful only when performing PLOGI with other Local Ports. It is not meaningful when performing PLOGI with Remote Ports. See 8.2 for rules regarding E_D_TOV.

3 In FC-AL, BB Receive Data Field SIze is meaningful only when performing PLOGI with other Local Ports. It is not meaningful when performing PLOGI with Remote Ports.

4 In FC-AL, BB_Credit is meaningful only when performing PLOGI with other Local Ports. It is not meaningful when performing PLOGI with Remote Ports. An NL_Port is Required to support a value of 0 or greater for this parameter.

5 The value for this parameter is determined by the ULP. See Annex B. In general, if a specific ULP supported by the NL_Port requires this feature, the corresponding bit should be set to one.

5.2.2 PLOGI Class 2 Service Parameters

Table 6 lists PLOGI Class 2 Service Parameters with usage defined by this document. In this table, “NL_Port Originator or Responder” refers to either an NL_Port issuing a PLOGI request to another N_Port or NL_Port, or to an NL_Port issuing a PLOGI ACC to an N_Port or NL_Port that sent a PLOGI request.

Table 6 – PLOGI Class 2 Service Parameters for NL_Ports

Class 2 Service Parameter	NL_Port Originator or Responder	Notes
Priority/Preemption	0	
X_ID Reassignment	‘00’b	
Initial Process Associator	‘00’b	
ACK Initiator Capability ACK_0 ACK_N	 X 0	
ACK Generation Assistance	X	1
ACK Recipient Capability ACK_0 ACK_N	 X 0	
Initiator Data compression capable	0	
Initiator Data compression History buffer size	-	
Initiator Data Encryption capable	0	
X_ID Interlock	X	
Recipient Error Policy Supported	‘00’b	
Recipient Categories per Sequence	‘00’b	
Recipient Data compression capable	0	
Recipient Data compression History buffer size	-	
Recipient Data Encryption capable	0	
Maximum Receive data field size (min)	256	
Concurrent Sequences (min)	1	
End to End Credit (min)	1	
Open Sequences per Exchange (min)	1	

NOTES:

1 ACK generation assistance is useful for ACK_0 generation in Class 2. ACK generation assistance is Allowed by this document.

5.2.3 PLOGI Class 3 Service Parameters

Table 7 lists PLOGI Class 3 Service Parameters with usage defined by this document. In this table, “NL_Port Originator or Responder” refers to either an NL_Port issuing a PLOGI request to another N_Port or NL_Port, or to an NL_Port issuing a PLOGI ACC to an N_Port or NL_Port that sent a PLOGI request.

Table 7 – PLOGI Class 3 Service Parameters for NL_Ports

Class 3 Service Parameter	NL_Port Originator or Responder	Notes
Initial Process Associator	‘00’b	
Initiator Data compression capable	0	
Initiator Data compression History buffer size	-	
Initiator Data Encryption capable	0	
Recipient Error Policy Supported	’00’b	
Recipient Categories per Sequence	’00’b	
Recipient Data compression capable	0	
Recipient Data compression History buffer size	-	
Recipient Data Encryption capable	0	
Maximum Receive data field size (min)	256	
Concurrent Sequences (min)	1	
Open Sequences per Exchange (min)	1	
NOTES: none		

5.3 Other NL_Port FC-PH Features

Table 8 lists other FC-PH features not covered in previous clauses for NL_Ports with usage defined by this document. In this table, “NL_Port Originator” refers to an NL_Port sending a Frame or Sequence to another N_Port or NL_Port, and “NL_Port Responder” refers to an NL_Port receiving a Frame or Sequence from another N_Port or NL_Port.

Table 8 – Other FC-PH features for NL_Ports

Feature	NL_Port Originator	NL_Port Responder	Notes
X_ID reassignment	P	A	
X_ID invalidation	P	A	
R_CTL Routing Bits			
FC-4 Device_Data frame	R	R	
Extended Link_Data frame	R	R	
Basic Link_ Data frame	R	R	
Link_Control frame	R	R	1
TYPE field set to any value	A	A	
Link Control frames			1
ACK_0	A	A	
ACK_1	R	R	
ACK_N	P	P	
F_BSY	P	R	
F_RJT	P	R	
P_BSY	I	R	
P_RJT	I	R	
LCR	R	R	
NTY	P	P	
END	P	P	
Basic Link Services			
ABTS	I	R	6
BA_ACC	R	R	
BA_RJT	R	R	
NOP	P	P	
RMC	P	P	
PRMT	P	P	
Nonzero Continue Sequence Condition values	P	-	
Ignore nonzero Continue Sequence values	-	R	
X_ID Interlock in Class 2	R	A	5
ACK Assist (ACK_Form in F_CTL) in Class 2			2
ACK_Form = no assist	A	A	
ACK_Form = ACK_1 expected	A	A	
ACK_Form = ACK_N expected	P	P	
ACK_Form = ACK_0 expected	A	A	
Sequence Retransmission (in F_CTL)	P	P	
Abort Sequence Condition ≠ ’00’b	A	A	
Optional Headers (all)	A	A	8
Payload size when frame is not the last frame of a sequence	(size MOD 4) == 0	(size MOD 4) == 0	3

Table 8 – Other FC-PH features for NL_Ports (continued)

Feature	NL_Port Originator	NL_Port Responder	Notes
Node Name Format Registered Format non-Registered Format	 I P	 R P	 4
N_Port Name Format Registered Format non-Registered Format	 I P	 R P	 4
Hunt Groups	P	P	
Multicast and Broadcast	A	A	7
Dedicated Simplex	P	P	

NOTES:

1 Link Control frames are Required only for Class 2. A Class 3-only NL_Port shall be able to return a P_RJT in Class 2.

2 Values of ’01’b and ‘11’b are Allowed based on the result of login. A value of ’10’b (for ACK_N) is Prohibited by this profile.

3 As per FC-PH, the size of a frame that is not the last frame of an Information Category shall be a multiple of four. A frame length of zero is Allowed.

4 All 64-bits of the Node Name and N_Port Name shall be significant. The Node Name and N_Port Name shall use a format that contains a worldwide unique identifier issued by a recognized registration authority. The IEEE, IEEE Extended, and IEEE Registered name formats are examples of acceptable formats. The Node Name and N_Port Name need not be related to each other in any way.

5 If the Sequence Recipient requires X_ID Interlock, the Sequence Initiator shall provide it.

6 Note that this means that, unlike FC-PLDA, the SCSI Target may invoke ABTS, and the SCSI Initiator shall accept it.

7 The use of broadcast and multicast is specific to the ULP. See also 5.5.

8 The use of optional headers is specific to the ULP.

5.4 P_RJT and F_BSY/P_BSY frames

Compliant NL_Ports operating in Class 2 may return a P_RJT in response to any received Class 2 frame, for any of the reason codes defined in FC-PH, FC-PH-2, or FC-PH-3. Compliant NL_Ports operating in Class 2 may receive a P_RJT in response to any Class 2 frame sent by the NL_Port, for any of the reason codes defined in FC-PH, FC-PH-2, and FC-PH-3.

Compliant NL_Ports operating in Class 2 may return a P_BSY in response to any received Class 2 frame, for any of the reason codes defined in FC-PH, FC-PH-2, and FC-PH-3. Compliant NL_Ports operating in Class 2 may receive a P_BSY or F_BSY in response to any Class 2 frame sent by the NL_Port, for any of the reason codes defined in FC-PH, FC-PH-2, and FC-PH-3.

An NL_Port is Allowed to retransmit individual frames in response to P_RJT, F_BSY, or P_BSY, if one of the following criteria is met:

- the frame is the first frame of the first Sequence of an Exchange, and X_ID Interlock is Required;
- the frame is the only frame of a single frame Sequence.

Otherwise, an NL_Port that receives a P_RJT, F_BSY, or P_BSY in response to a Class 2 frame shall abort the Sequence (via ABTS) and retransmit the Sequence or abort the Exchange, as Allowed or Required by the ULP.

An NL_Port that sends a P_RJT or P_BSY shall expect the Sequence and/or Exchange to be aborted by the other NL_Port or N_Port. An NL_Port that sends a P_RJT or P_BSY shall expect the frame to be re-transmitted if the frame meets one of the above criteria. The Sequence may be retransmitted after the ABTS protocol is complete, if Allowed or Required by the ULP.

NL_Port response to F_RJT is defined in 6.4.

5.5 FC-AL features

Table 9 lists Fibre Channel Arbitrated Loop features for NL_Ports with usage defined by this document. In this table, “NL_Port Originator” refers to an NL_Port sending an FC-AL feature to an FL_Port or another Local NL_Port, and “NL_Port Responder” refers to an NL_Port receiving an FC-AL feature from an FL_Port or another Local NL_Port. See 6.5 for FC-AL requirements for FL_Ports.

Table 9 – FC-AL features for NL_Ports

Feature	NL_Port Originator	NL_Port Responder	Notes
Open Full Duplex (OPNyx)			
Open NL_Port on Local Loop	I	R	
Open FL_Port	I	-	
Open Half Duplex (OPNyy)			
Open NL_Port on Local Loop	I	R	
Open FL_Port	I	-	
Send frames to FL_Port for more than one D_ID in a single Loop Tenancy	I	-	
Receive frames from FL_Port for more than one S_ID in a single Loop Tenancy	-	R	
Unfairness	I	R	1
Transfer mode (use of TRANSFER loop state)	I	R	1
LILP/LIRP			
FLA-A Compliance	A	A	2
FLA-B Compliance	R	R	
Broadcast and Multicast via Broadcast Replicate (OPNfr)			
To NL_Ports (only) on the Local Loop	A	-	3
From NL_Ports (only) on the Local Loop	-	A	3
To Remote Ports	P	-	11
From Remote Ports	-	A	12
Broadcast and Multicast via Selective Replicate (OPNyr)			
To NL_Ports (only) on the Local Loop	A	-	3
From NL_Ports (only) on the Local Loop	-	A	3
To Remote Ports	P	-	11
From Remote Ports	-	P	12
Alternate BB_Credit model	R	R	4
Login_BB_Credit=0	R	R	5
Login_BB_Credit>0	A	A	5

Table 9 – FC-AL features for NL_Ports (continued)

Feature	NL_Port Originator	NL_Port Responder	Notes
LPEyx/LPByx/LPEfx	A	A	6
LIP (F7, F7) and (F7, AL_PS) [initializing] (F8, F7) and (F8, AL_PS) [loop failure] (AL_PD, AL_PS) [selective hard reset]	 I I I	 R R R	 7 8 9
MRK	P	R	10

NOTES:

1 NL_Ports shall tolerate the use of Unfairness and Transfer Mode by other L_Ports.

2 Some legacy devices that do not implement LILP and LIRP would otherwise be compliant with this profile.

3 NL_Ports may use Broadcast Replicate (OPNfr) or Selective Replicate (OPNyr) to communicate on a Local Loop. As required by FC-AL, the FL_Port shall not receive frames associated with OPNfr or OPNyr. Therefore, frames originated on a Local Loop by an NL_Port using OPNfr or OPNyr shall not propagate beyond the Local Loop. Use of broadcast and multicast is specific to the ULP. The method by which an NL_Port may discover the ability of other NL_Ports to receive OPNfr or OPNyr frames is not defined by this document.

4 Alternate BB_Credit management is mandatory in FC-AL.

5 NL_Ports shall interoperate with NL_Ports and FL_Ports which advertise any Login_BB_Credit. A recipient of frames may login with Login_BB_Credit>0, but the originator of frames is not Required to take advantage of it.

6 LPEfx is useful for resetting the bypass circuits of NL_Ports which have been bypassed and have lost their AL_PAs due to a LIP (and therefore cannot be enabled using an addressed LPEyx).

7 This LIP may be issued by an NL_Port to request an AL_PA if it has none, or if it is unable to win arbitration within A_W_TOV. NL_Port response to a LIP is described in 5.6.

8 This LIP may be issued by an NL_Port which detects a Link Failure, as defined by FC-PH.

9 An NL_Port or FL_Port may invoke this form of LIP to reset an NL_Port in a ULP-specific manner. In the absence of a ULP-defined reset, a Public NL_Port shall respond to this form of LIP in the same manner as LIP(F7,AL_PS).

10 An NL_Port that receives MRK shall attempt to forward the MRK. FC-AL states that an L_Port may remove a MRK if necessary for clock skew management.

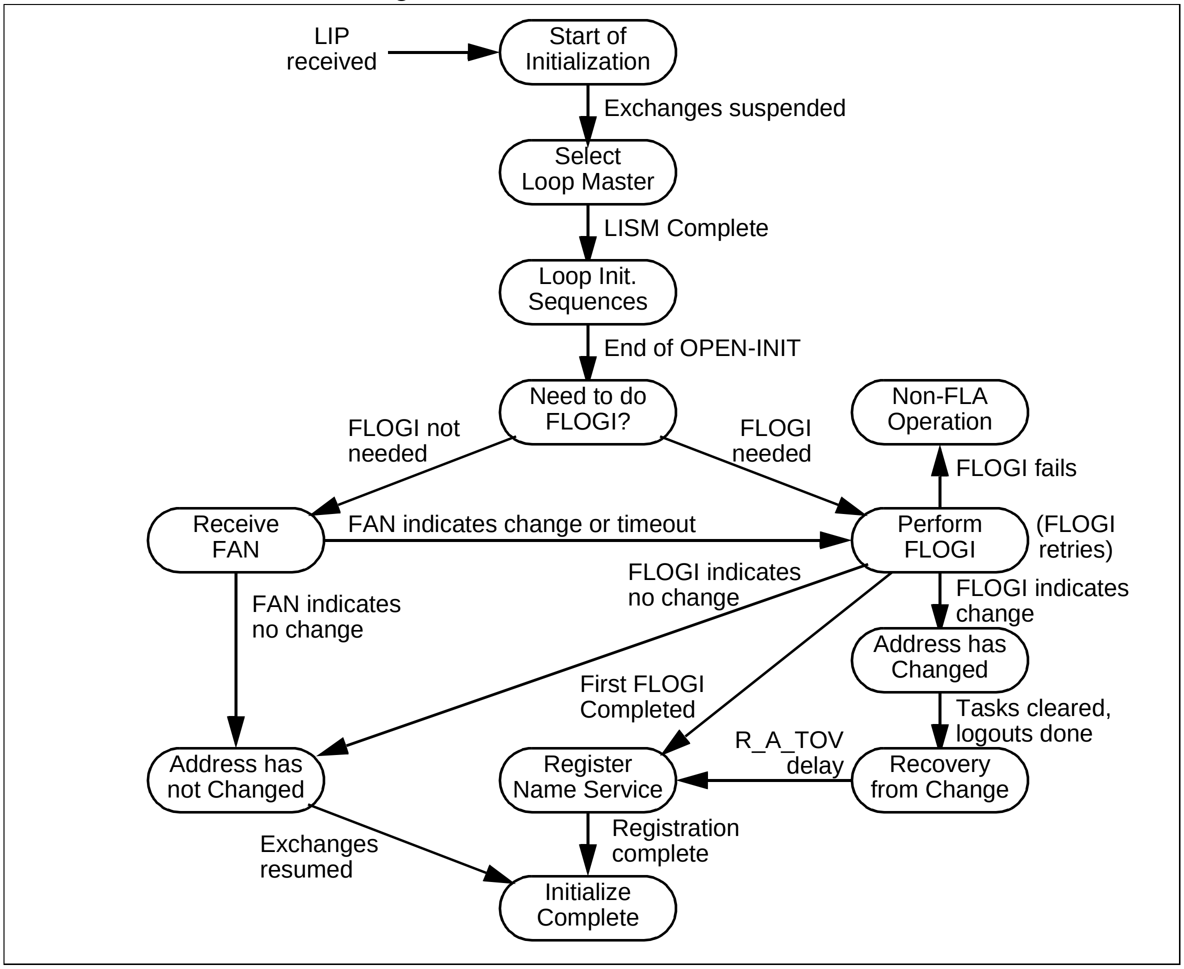
11 As required by FC-AL, an NL_Port with broadcast or multicast frames intended for Remote Ports shall open the FL_Port via OPNyx or OPNyy (or be opened by the FL_Port via OPNyx) and send the frames to the FL_Port for forwarding by the fabric. Note that this could cause the frames to be delivered to the Local Loop if any NL_Ports on the Local Loop are part of the broadcast or multicast group.

12 The only method Allowed by this document of delivering broadcast or multicast from the FL_Port to the attached NL_Ports is via an OPNfr sent by the FL_Port. The FL_Port is not Required to perform a discovery process to determine whether the attached NL_Ports are able to receive broadcast or multicast frames via OPNfr. Use of broadcast and multicast, and the number of frames that may be sent following OPNfr, are both specific to the ULP.

5.6 NL_Port initialization features

NL_Ports shall initialize as detailed below. Figure 2 shows a schematic of the process to illustrate the flow. If the figure is different than the text, the text shall apply.

Figure 2 – NL_Port initialization flow



a) **Start of Initialization**. Whenever an NL_Port receives any LIP, it shall begin this initialization process. The NL_Port shall suspend all open Exchanges with all other N_Ports and NL_Ports, both on Remote Loops and the Local Loop. The NL_Port shall proceed to **step (b)**.

b) **Selection of Loop Master**. At the start of Loop Initialization (as defined in FC-AL clause 10), the NL_Port shall compete to become Loop Master as defined in FC-AL. (The NL_Port is expected to lose to an FL_Port, as required by FC-AL. If it doesn't then the loop does not conform to this report; i.e, it may be a Private Loop.) The NL_Port shall proceed to **step (c)**.

c) **Loop Initialization Sequences**. If the NL_Port had an AL_PA prior to the LIP, and the NL_Port had completed FLOGI, then the NL_Port shall attempt to acquire its Fabric-assigned AL_PA during the LIFA Loop Initialization Sequence. If the NL_Port had an AL_PA prior to the LIP, and the NL_Port had not completed FLOGI, then the NL_Port shall attempt to acquire its previously-assigned AL_PA during the LIPA Loop Initialization Sequence. Otherwise, or if the NL_Port could not acquire an AL_PA during LIFA or LIPA, the NL_Port shall attempt to acquire an

AL_PA during the LIHA or LISA Loop Initialization Sequences, as appropriate. The NL_Port shall proceed to **step (d)**.

d) **Need to do FLOGI**. Upon completion of Loop Initialization, the NL_Port shall implicitly log out with the Fabric and perform FLOGI as described in **step (f)** with the Fabric F_Port (well-known address hex 'FFFFFE') if one or more of the following is true:

– the L_bit was set to one in at least the LISA Sequence during Loop Initialization;

– the NL_Port did not acquire the AL_PA it had prior to the Loop Initialization;

– the N_Port_Name or Node_Name of the NL_Port changed to a different value than it had prior to the Loop Initialization;

– the NL_Port did not have an AL_PA prior to the Loop Initialization;

– the NL_Port had not completed FLOGI prior to the Loop Initialization.

Otherwise, the NL_Port shall proceed to **step (e)**.

e) **Receive FAN**. If FLOGI is not required as described in **step (d)**, the NL_Port shall wait a minimum of E_D_TOV to receive the FAN (Fabric Address Notification) ELS. The NL_Port shall discard all frames received except ADISC and PLOGI requests until the FAN is received. The NL_Port shall either receive any ADISC and/or PLOGI requests and delay reply until the FAN is received, or the NL_Port shall respond to ADISC and PLOGI request Sequences with an LS_RJT reply Sequence, with a Reason code of “Unable to perform Command Request at this time”. After receiving the FAN ELS, if the NL_Port determines that the FL_Port has the same address, F_Port_Name and Fabric_Name that the FL_Port had before initialization, the NL_Port shall proceed to **step (j)**. Otherwise, the NL_Port shall implicitly log out with the Fabric and perform FLOGI as described in **step (f)**. If the NL_Port does not receive a FAN ELS within E_D_TOV of the completion of Loop Initialization Protocol, it shall also proceed to **step (f)**.

NOTE – The behavior described above in which frames (regardless of Class of Service) are discarded prior to FAN being received will normally not happen if the FL_Port is present and sending FAN as described in 6.6. In that case, the FAN will always be the first frame received following loop initialization.

f) **Perform FLOGI**. If FLOGI is required as described in **step (d)** or **step (e)**, the NL_Port shall perform the following.

1) The NL_Port shall attempt to send FLOGI (see 5.1) to the FL_Port by opening AL_PA hex '00' via OPNyx, initiating the FLOGI request Sequence, and originating a CLS. The NL_Port may send R_RDY if it is able to receive the FLOGI ACC in the same Loop Tenancy.

2) The NL_Port shall ignore any FAN ELS received while attempting to perform FLOGI. The NL_Port shall either receive any ADISC and/or PLOGI requests and delay reply until the FLOGI is complete, or the NL_Port shall respond to ADISC and PLOGI request Sequences with an LS_RJT reply Sequence, with a Reason code of “Unable to perform Command Request at this time”.

3) If the NL_Port detects that the FLOGI request attempt fails (e.g., by receiving the OPN sent), or if the FLOGI ACC Sequence is not received within E_D_TOV of the FLOGI request, it may either:

   i) retry the attempt in **step (1)** after R_A_TOV; or,

   ii) proceed to **step (l)**; or,

   iii) wait for a future Loop Initialization in which the L_bit is set to one in the LISA Sequence.

4) If the FLOGI is completed successfully, and the NL_Port determines that it has the same AL_PA, address (ID), N_Port_Name and Node_Name that it had before initialization, and the NL_Port also determines that the FL_Port has the same address, F_Port_Name and Fabric_Name that the FL_Port had before initialization, the NL_Port shall proceed to **step (j)**. If the FLOGI is completed successfully, and if the NL_Port did not have a prior AL_PA or had not completed FLOGI prior to loop initialization (as noted in **step (d)**), the NL_Port shall proceed to **step (i)**. Otherwise, the NL_Port shall proceed to **step (g)**.

g) **Address has changed**. An NL_Port performing this step has determined that its own addressing information and/or that of the FL_Port have changed, or that it did not have complete addressing information, prior to the initialization. The NL_Port shall discard all pending Exchanges in a manner consistent with the ULP, implicitly LOGO with all other N_Ports and NL_Ports, and perform ULP-specific actions to clear pending tasks. The NL_Port shall proceed to **step (h)**.

h) **Recovery from address change**. The NL_Port shall wait for R_A_TOV before originating any new Exchanges. During this time, the NL_Port shall discard all frames received except for the PLOGI and ADISC request Sequences, and LOGO ACC reply Sequences. The NL_Port shall respond to all PLOGI request Sequences with an LS_RJT reply Sequence, with a Reason code of “Unable to perform Command Request at this time”. The NL_Port shall respond to all ADISC request Sequences with a LOGO request Sequence to the N_Port or NL_Port that initiated the ADISC request Sequence. The NL_Port shall proceed to **step (i)**.

i) **Registration**. The NL_Port shall perform PLOGI explicitly with the Directory Server (well-known address hex ’FFFFFC’) and shall attempt an RFC-4 request with the Name Service. If the attempted RFC-4 fails, the NL_Port is not Required to retry the attempt. The NL_Port may at this time perform any other Directory Server registration requests it may wish to do. The NL_Port shall proceed to **step (k)**.

j) **Address has not changed**. An NL_Port performing this step has determined that its own addressing information and that of the FL_Port have not changed as a result of the initialization. The NL_Port shall resume all suspended Exchanges with Remote Ports and with Local Ports that are known to be Public NL_Ports. If the NL_Port has any suspended Exchanges with Private Loop devices on the Local Loop, the NL_Port may perform the Private Loop authentication described in FC-PLDA as either the originator or recipient, as appropriate. The NL_Port shall proceed to **step (k)**.

k) **Completion of NL_Port initialization**. This completes initialization for the NL_Port. The NL_Port may proceed to originate and respond to Exchanges, and login with other N_Ports and NL_Ports as needed.

l) **Fall-back to non-FC-FLA operation**. The NL_Port is not connected to a loop that contains an FL_Port, and therefore is not connected to an FC-FLA compliant loop. The NL_Port may choose to complete initialization based on the rules described in FC-PLDA.

An NL_Port that wishes to login with an NL_Port or N_Port shall perform the following steps:

a) **Discovery**. As required by the ULP, the NL_Port may perform a discovery process to locate N_Ports and NL_Ports, and perform PLOGI with those N_Ports and/or NL_Ports. The GP_ID4 service provided by the Name Service may be used as part of this process.

b) **NL_Port Login**. If the NL_Port wishes to use Class 2 to communicate with an N_Port or NL_Port, and FLOGI with the FL_Port establishes that the FL_Port supports Class 2, the NL_Port shall PLOGI with that port in Class 2. If the PLOGI ACC is received successfully, the NL_Port may use Class 2 with the other port; if the fabric indicates that a Fabric Path is Not Available (see 6.4), then Class 2 is not available with that N_Port/NL_Port. If the NL_Port only requires Class 3, then PLOGI shall be performed in Class 3.

5.7 Extended Link Services

Table 10 lists Extended Link Services with usage defined by this document. Devices that receive requests for Extended Link Services that are unsupported shall return LS_RJT with a reason code of “Invalid Command Code”. In this table, “NL_Port Originator” refers to an NL_Port sending an Extended Link Service request to another N_Port or NL_Port, and “NL_Port Responder” refers to an NL_Port receiving an Extended Link Service request from another N_Port or NL_Port.

Table 10 – Extended Link Services for NL_Ports

Extended Link Service	NL_Port Originator	NL_Port Responder	Notes
ABTX	A	A	
ADISC	I	R	1
ADVC	P	P	
ECHO	P	P	
ESTC	P	P	
ESTS	P	P	
FACT	P	P	
FAN	P	R	1
FDACT	P	P	
FDISC	I	P	2
FLOGI	R	P	2
GAID	P	P	
LINIT	I	P	3
LOGO	R	R	
LPC	I	P	3
LSTS	I	P	3
NACT	P	P	
NDACT	P	P	
PDISC	I	R	1
PLOGI	R	R	
PRLI	A	A	5
PRLO	A	A	5
QoSR	P	P	
RCS	P	P	
RES	A	A	7
RLS	A	A	4
RNC	R	R	
RRQ	R	R	
RSCN	A	A	6
RSI	A	A	7
RSS	A	A	7
RTV	P	P	
RVCS	P	P	
SCN	A	A	
SCR	A	A	6

Table 10 – Extended Link Services for NL_Ports (continued)

Extended Link Service	NL_Port Originator	NL_Port Responder	Notes
TEST	P	P	
TPLS	P	P	
TPRLO	A	A	5

NOTES:

1 FAN is a new ELS defined for this report (see Annex A) and is the preferred method for Public NL_Ports to authenticate addresses following loop initialization. ADISC is the preferred method to authenticate addresses with Private NL_Ports following loop initialization.

2 Public NL_Ports do not receive FDISC or FLOGI requests.

3 These are new optional ELS provided for loop management (see Annex A).

4 Not all fields in the LESB may be supported (see vendor-specific documentation). RLS can be a diagnostic tool for isolating link degradation in loop hub or backplane topologies. Using RLS, diagnostic software can use a combination of self-directed test frames, LPB, and LESB polling to determine which device or link segment is causing link errors.

5 Support for PRLI, PRLO and TPRLO is ULP-dependent. See the specific ULP standards and profiles.

6 These are new optional ELS to allow registration and notification of state changes (see Annex A).

7 These are used by Stream devices that conform to FC-PLDA.

5.8 Well-known address usage by NL_Ports

Table 11 lists the well-known address requirements for NL_Ports with usage defined by this document. In this table, “NL_Port Originator” refers to an NL_Port sending a request Sequence to an N_Port at a well-known address, and “NL_Port Responder” refers to an NL_Port receiving a request Sequence from an N_Port at a well-known address.

Table 11 – Well-known address support for NL_Ports

Parameter	NL_Port Originator	NL_Port Responder	Notes
Well-Known Address hex ’FFFFFF’ (Broadcast)	A	A	4, 5
Well-Known Address hex ’FFFFFE’ (“Fabric F_Port”)	R	R	1, 5
Extended Link Services:			
ABTS	I	R	
FAN	P	R	
FDISC	I	P	
FLOGI	R	P	
RRQ	I	R	
All other Extended Link Services	P	P	
Well-Known Address hex ’FFFFFD’ (“Fabric Controller”)	R	R	
Extended Link Services:			
ECHO	A	A	
PLOGI	A	P	
RSCN	I	A	
SCN	A	A	
SCR	I	P	
TEST	A	A	
All other Extended Link Services	P	P	
Well-Known Address hex ’FFFFFC’ (“Directory Server”)	R	P	
Extended Link Services:			
ECHO	A	A	
LOGO	I	P	
PDISC	I	P	
PLOGI	R	P	
RLS	A	A	
RNC	R	R	
RRQ	R	R	
TEST	A	A	
All other Extended Link Services	P	P	
Name Service			2
RFC-4	R	P	6
GPT	I	P	
GP_ID4, Port_Type = “Nx_Port”	I	P	
GP_ID4, other Port_Types	A	P	
All other services	A	P	
X.500 Service	A	P	
ARP Service	A	P	
Well-Known Address hex ’FFFFFB’ (Time Server)	P	P	
Well-Known Address hex ’FFFFFA’ (Management Server)	P	P	
Well-Known Address hex ’FFFFF9’ (Quality of Service Facilitator)	P	P	3

Table 11 – Well-known address support for NL_Ports (continued)

Parameter	NL_Port Originator	NL_Port Responder	Notes
Well-Known Address hex ’FFFFF8’ (Alias Server)	A	A	7
Well-Known Address hex ’FFFFF7’ (Security Key Distribution Server)	P	P	
Well-Known Address hex ’FFFFF6’ (Clock Synchronization Server)	P	P	

NOTES:

1 This is the address to which an N_Port or NL_Port sends a FLOGI.

2 The Name Service is defined in FC-GS-2 (see reference [12]). See Annex C for examples of Name Services.

3 This service is currently defined only for Class 4 service requests.

4 Broadcast to NL_Ports shall be achieved via Open Broadcast Replicate (OPNfr). Use of broadcast is specific to the ULP. See 5.5.

5 PLOGI shall not be sent to these addresses.

6 An NL_Port is required to attempt this request. See 5.6.

7 The Alias Server may be used to support Multicast and Broadcast.

6 Fabric and FL_Port feature sets

The tables in this clause list features described in the Fibre Channel Physical and Signaling Interface standard (FC-PH) and in the Fibre Channel Arbitrated Loop (FC-AL), specific to the operation of Fabrics and FL_Ports. These tables indicate whether the feature is Required, Prohibited, Allowed, or Invocable for compliance with this specification. Features that are not listed do not affect interoperability of public loop devices.

PLEASE READ 3.4 for the exact definition of Prohibited, Required, Allowed, and Invocable, and for the legend for table entries used in this clause.

6.1 FLOGI ACC features and parameters to NL_Ports

This clause specifies the FLOGI ACC features and parameters returned by an FL_Port to an NL_Port to complete login. Table 12 lists the features used and parameters returned by FL_Ports with usage defined by this document. In this table, “FL_Port Responder” refers to an FL_Port issuing a FLOGI ACC to an NL_Port. See 5.1 for the requirements for the FLOGI request.

Table 12 – FLOGI ACC features and parameters for FL_Ports

Feature/Parameter	FL_Port Responder	Notes
Class of service for FLOGI and FLOGI ACC Class 3 Other Classes	 R A	1
Returned D_ID in FLOGI frame header	ddaa||AL_PA	2
Maximum size of FLOGI ACC frame	256	
FC-PH Version Highest Version Lowest Version	 X hex ’20’	
BB_Credit (min)	0	11
Valid Vendor Version Level	0	3
N_Port/F_Port	1	
Alternate BB_Credit Management	1	
E_D_TOV Resolution	0	5
FLOGI Payload Length	from request	8
Maximum BB Receive Data Field Size (min)	256	4
R_A_TOV	r_a_tov	5
E_D_TOV	e_d_tov	5
Fabric Name Format Registered Format non-Registered Format	 R P	12
Fabric_Port Name Format Registered Format non-Registered Format	 R P	12
Supported Classes of Service: Class 1 Class 2 Class 3 Class 4 and up	 P A R P	6

Table 12 – FLOGI ACC features and parameters for FL_Ports (continued)

Feature/Parameter	FL_Port Responder	Notes
Sequential Delivery bit		
Class 2	1	7
Class 3	1	7
Priority/Preemption in Class 2	0	
Services Availability field		9
"Directory Server" bit	1	10
All other bits	0	

NOTES:

1 If a fabric receives a request for FLOGI to hex 'FFFFFE' in a Class of Service other than Class 3 that it does not support, it shall issue a F_RJT. The F_RJT shall begin with SOFnX (where 'X' is the Class of Service of the received FLOGI) and end with EOFt. The Action Code shall be "Rejected frame is retryable", and the Reason Code shall be "Class not supported by entity at hex 'FFFFFE'". The fabric shall respond to the FLOGI with a reply Sequence in the same Class of Service as the FLOGI request Sequence.

2 The fabric shall build the assigned ID by using the requestor's Port_ID (AL_PA) for the low order byte (the AL_PA was transferred in the FLOGI request as the low order byte of the S_ID). The fabric may freely assign the Domain_ID (dd) and Area_ID (aa). The value of 'ddaa' shall be the same for all Public NL_Ports on a given loop. The fabric shall return the value of the low order byte of the S_ID of the FLOGI request in the low order byte of the D_ID

3 Profile versions are communicated via the Report Node Capabilities (RNC) ELS. The Valid Vendor Version Level shall not be set to 1 for the purposes of communicating support of this Technical Report.

4 Public loop devices shall be capable of receiving 256-byte frames. FC-AL initialization requires NL_Ports to be able to receive 132-byte frames. There is no restriction on the *maximum maximum* frame size (up to the limits specified by FC-PH).

5 See clause 8 on page 68 for the timer values.

6 Any SOFc1 frame received by an FL_Port shall be rejected. The F_RJT shall begin with SOFn1 and end with EOFdt. The Action Code shall be "Rejected frame is not retryable", and the Reason Code shall be "Class not supported". All SOFn1 and SOFi1 frames shall be discarded by the FL_Port.

7 While FC-PH may require that the frames travel a "fixed route" across the fabric, this document does not particularly care how the function is implemented, as long as it is performed correctly.

8 The FLOGI Payload Length bit shall be set to the same value as in the FLOGI request from the NL_Port. If this bit is set to 1, the 256 byte FLOGI ACC payload that contains the Services Availability field shall be returned.

9 This field is returned if the Payload Length bit is set to 1.

10 Setting this bit indicates the Directory Server, and the Required Name Service services are available. Additional services may also be available.

11 An FL_Port is Required to support a value of 0 or greater for this parameter.

12 All 64-bits of the Fabric Name and Fabric_Port Name shall be significant. The Fabric Name and Fabric_Port Name shall use a format that contains a worldwide unique identifier issued by a recognized registration authority. The IEEE, IEEE Extended, and IEEE Registered name formats are examples of acceptable formats. The Fabric Name and Fabric_Port Name need not be related to each other in any way.

6.2 Well-known addresses

This clause specifies the well-known addresses that shall be provided by the fabric, and the N_Port characteristics of those addresses. All services provided by well-known addresses are either directly performed by a fabric element, or by some other entity known to the fabric as a whole. This report does not make the distinction; rather, it mandates that NL_Ports and N_Ports have access to Required addresses and services, and that this access be available in some manner through the FL_Port or F_Port.

Table 13 lists the well-known address requirements for fabrics with usage defined by this document. In this table, “Fabric Originator” refers to an N_Port at a well-known address sending a request Sequence to an N_Port or NL_Port, and “Fabric Responder” refers to an N_Port at a well-known address receiving a request Sequence from an N_Port or NL_Port.

Table 13 – Well-known address support for Fabrics

Parameter	Fabric Originator	Fabric Responder	Notes
Well-Known Address hex ’FFFFFF’ (Broadcast)	A	A	4, 5
Well-Known Address hex ’FFFFFE’ (“Fabric F_Port”)	R	R	1, 5
Extended Link Services:			
FAN (see Annex A)	R	P	8
FDISC	P	R	
FLOGI	P	R	
RRQ	R	R	
All other Extended Link Services	P	P	
Well-Known Address hex ’FFFFFD’ (“Fabric Controller”)	R	R	
Extended Link Services:			
ECHO	A	A	
PLOGI	P	A	
RSCN	R	R	10
SCN	A	A	
SCR	P	R	10
TEST	A	A	
All other Extended Link Services	P	P	
Well-Known Address hex ’FFFFFC’ (“Directory Server”)	P	R	
Extended Link Services:			
ECHO	A	A	
LOGO	P	R	
PDISC	P	R	
PLOGI	P	R	
RLS	A	A	9
RNC	R	R	
RRQ	R	R	
TEST	A	A	
All other Extended Link Services	P	P	
Name Service:			2
RFC-4	P	A	
GPT	P	R	6
GP_ID4, Port_Type = “Nx_Port”	P	R	
GP_ID4, other Port_Types	P	A	
All other Name Service services	P	A	
X.500 Service	P	A	
ARP Service	P	A	

Table 13 – Well-known address support for Fabrics (continued)

Parameter	Fabric Originator	Fabric Responder	Notes
Well-Known Address hex ’FFFFFB’ (Time Server)	P	P	
Well-Known Address hex ’FFFFFA’ (Management Server)	P	P	
Well-Known Address hex ’FFFFF9’ (Quality of Service Facilitator)	P	P	3
Well-Known Address hex ’FFFFF8’ (Alias Server)	A	A	7
Well-Known Address hex ’FFFFF7’ (Security Key Distribution Server)	P	P	
Well-Known Address hex ’FFFFF6’ (Clock Synchronization Server)	P	P	

NOTES:

1 This is the address to which an N_Port or NL_Port sends a FLOGI.

2 The Name Service is defined in FC-GS-2 (see reference [12]). See Annex C for examples of Name Services.

3 This service is currently defined only for Class 4 service requests.

4 Broadcast to NL_Ports shall be achieved via Open Broadcast Replicate (OPNfr). Use of broadcast is specific to the ULP. See 6.5.

5 PLOGI shall not be sent to these addresses.

6 At a minimum, a fabric shall implicitly register the Native Port Identifier and Port_Type of any N_Port and NL_Port that completes FLOGI. A fabric shall also implicitly register the Native Port Identifier and Port_Type of all FL_Ports, so that Loop Fabric Addresses may be determined. The fabric shall also implicitly register, upon completion of FLOGI, the Port_Name, Node_Name, and Supported Classes of Service, if it supports the query commands for those objects.

7 The Alias Server may be used to support Multicast and Broadcast.

8 FAN is a new ELS defined for this report (see Annex A) and is the preferred method for Public NL_Ports to authenticate addressing following loop initialization.

9 Not all fields in the LESB may be supported (see vendor-specific documentation). RLS can be a diagnostic tool for isolating link degradation in loop hub or backplane topologies. Using RLS, diagnostic software can use a combination of self-directed test frames, LPB, and LESB polling to determine which device or link segment is causing link errors.

10 These are new optional ELS to allow registration and notification of state changes (see Annex A).

6.2.1 Features for N_Ports at well-known addresses

This clause specifies the supported features for an N_Port residing at a well-known address. Table 14 lists the features with usage defined by this document. In this table, “N_Port (Fabric) Originator” refers to an N_Port at a well-known address sending a Sequence to an N_Port or NL_Port, and “N_Port (Fabric) Responder” refers to an N_Port at a well-known address receiving a Sequence from an N_Port or NL_Port.

Table 14 – Features for N_Ports at well-known addresses

Parameter	N_Port (Fabric) Originator	N_Port (Fabric) Responder	Notes
Class of service for PLOGI (N_Port Originator) and PLOGI ACC (N_Port Responder)			1, 3
Class 1	P	P	
Class 2	P	A	
Class 3	P	R	
Class 4 and up	P	P	
Supported Classes of Service:			1
Class 1	P	P	
Class 2	A	A	
Class 3	R	R	
Class 4 and up	P	P	
Implicit Login with N_Port at Well-Known Addresses			2
address hex ’FFFFFF’	-	R	
address hex ’FFFFFE’	-	R	
address hex ’FFFFFD’	-	A	
All other well-known addresses	-	P	

NOTES:

1 If an N_Port at a well-known address receives a request for service (other than FLOGI to hex ’FFFFFE’) in an unsupported Class of Service other than Class 3, it shall issue a P_RJT (not F_RJT). The P_RJT shall begin with SOFnX and end with EOFt. The Action Code shall be “Rejected frame is not retryable”, and the Reason Code shall be “Class not supported”.

2 Explicit login is Required before accessing services at all Well-Known addresses except hex ’FFFFFF’, hex ’FFFFFE’, and hex ’FFFFFD’. Explicit PLOGI is Prohibited to Well-Known addresses hex ’FFFFFF’, hex ’FFFFFE’. Implicit or explicit PLOGI may be performed with Well-Known address hex ’FFFFFD’.

3 N_Ports at well-known addresses shall not initiate PLOGI.

6.2.2 PLOGI Common Service Parameters for N_Ports at well-known addresses

Table 15 lists PLOGI Common Service Parameters for N_Ports at well-known addresses with usage defined by this document. In this table, “N_Port (Fabric) Responder” refers to an N_Port at a well-known address issuing a PLOGI ACC to an N_Port or NL_Port that sent a PLOGI request.

Table 15 – PLOGI Common Service Parameters for well-known addresses

Common Service Parameter	N_Port (Fabric) Responder	Notes
FC-PH Version Highest Version Lowest Version	 X hex ’20’	
BB_Credit (min)	-	3
Continuously Increasing Relative Offset	1	
Random Relative Offset	0	
Valid Vendor Version Level	0	1
N_Port/F_Port	0	
Alternate BB_Credit Management	0	2
E_D_TOV Resolution	-	3
Payload Length	0	
BB Receive Data Field Size (min)	-	3
Total Concurrent Sequences (min)	1	
Relative Offset by Information Category	X	
E_D_TOV	-	3

NOTES:

1 Profile versions are communicated via the Report Node Capabilities (RNC) ELS. The Valid Vendor Version Level shall not be set to 1 for the purposes of communicating support of this Technical Report.

2 Since the well-known address is accessed via the fabric, Alternate BB_Credit Management does not apply.

3 Since these N_Ports are by nature Remote Ports, these parameters are not meaningful in PLOGI.

4 The value for this parameter is determined by the ULP. See Annex B. In general, if a specific ULP supported by the N_Port at a well-known address requires this feature, the corresponding bit should be set to one.

6.2.3 PLOGI Class 2 Service Parameters for N_Ports at well-known addresses

Table 16 lists PLOGI Class 2 Service Parameters for N_Ports at well-known addresses with usage defined by this document. In this table, "N_Port (Fabric) Responder" refers to an N_Port at a well-known address issuing a PLOGI ACC to an N_Port or NL_Port that sent a PLOGI request.

Table 16 – PLOGI Class 2 Service Parameters for well-known addresses

Class 2 Service Parameter	N_Port (Fabric) Responder	Notes
Priority/Preemption	0	
X_ID Reassignment	'00'b	
Initial Process Associator	'00'b	
ACK Initiator Capability ACK_0 ACK_N	 X 0	
ACK Generation Assistance	X	1
ACK Recipient Capability ACK_0 ACK_N	 X 0	
Initiator Data compression capable	0	
Initiator Data compression History buffer size	-	
Initiator Data Encryption capable	0	
X_ID Interlock	X	
Recipient Error Policy Supported	'00'b	
Recipient Categories per Sequence	'00'b	
Recipient Data compression capable	0	
Recipient Data compression History buffer size	-	
Recipient Data Encryption capable	0	
Maximum Receive data field size (min)	256	
Concurrent Sequences (min)	1	
End to End Credit (min)	1	
Open Sequences per Exchange (min)	1	

NOTES:

1 ACK generation assistance is useful for ACK_0 generation in Class 2. ACK generation assistance is Allowed by this document.

6.2.4 PLOGI Class 3 Service Parameters for N_Ports at well-known addresses

Table 17 lists Class 3 Service Parameters for N_Ports at well-known addresses with usage defined by this document. In this table, “N_Port (Fabric) Responder” refers to an N_Port at a well-known address issuing a PLOGI ACC to an N_Port or NL_Port that sent a PLOGI request.

Table 17 – PLOGI Class 3 Service Parameters for well-known addresses

Class 3 Service Parameter	N_Port (Fabric) Responder	Notes
Initial Process Associator	’00’b	
Initiator Data compression capable	0	
Initiator Data compression History buffer size	-	
Initiator Data Encryption capable	0	
Recipient Error Policy Supported	’00’b	
Recipient Categories per Sequence	’00’b	
Recipient Data compression capable	0	
Recipient Data compression History buffer size	-	
Recipient Data Encryption capable	0	
Maximum Receive data field size (min)	256	
Concurrent Sequences (min)	1	
Open Sequences per Exchange (min)	1	
NOTES: (none)		

6.2.5 Other FC-PH Features for N_Ports at well-known addresses

Table 18 lists other FC-PH features not covered in previous clauses for N_Ports at well-known addresses with usage defined by this document. In this table, “N_Port (Fabric) Originator” refers to an N_Port at a well-known address sending a Frame or Sequence to an N_Port or NL_Port, and “N_Port (Fabric) Responder” refers to an N_Port at a well-known address receiving a Frame or Sequence from an N_Port or NL_Port.

Table 18 – Other FC-PH Features for well-known addresses

Feature	N_Port (Fabric) Originator	N_Port (Fabric) Responder	Notes
X_ID reassignment	P	P	
X_ID invalidation	P	P	
R_CTL Routing Bits			
FC-4 Device_Data frame	R	R	
Extended Link_Data frame	R	R	
Basic Link_ Data frame	R	R	
Link_Control frame	R	R	1
TYPE field set to:			
hex ’00’ (Basic Link Services)	R	R	
hex ’01’ (Extended Link Services)	R	R	
hex ’20’ (Fibre Channel Services)	R	R	
all other values	A	A	
Link Control frames			1
ACK_0	A	A	
ACK_1	R	R	
ACK_N	P	P	
F_BSY	P	R	
F_RJT	P	R	
P_BSY	I	R	
P_RJT	I	R	
LCR	R	R	
NTY	P	P	
END	P	P	
Basic Link Services			
ABTS	I	R	6
BA_ACC	R	R	
BA_RJT	R	R	
NOP	P	P	
RMC	P	P	
PRMT	P	P	
Nonzero Continue Sequence Condition values	P	-	
Ignore nonzero Continue Sequence values	-	R	
X_ID Interlock in Class 2	R	A	5
ACK Assist (ACK_Form in F_CTL) in Class 2			2
ACK_Form = no assist	A	A	
ACK_Form = ACK_1 expected	A	A	
ACK_Form = ACK_N expected	P	P	
ACK_Form = ACK_0 expected	A	A	
Sequence Retransmission (in F_CTL)	P	P	

Table 18 – Other FC-PH Features for well-known addresses (continued)

Feature	N_Port (Fabric) Originator	N_Port (Fabric) Responder	Notes
Abort Sequence Condition ≠ ’00’b	A	A	
Optional Headers (all)	P	P	
Payload size when frame is not the last frame of a sequence	(size MOD 4) == 0	(size MOD 4) == 0	3
Node Name Format Registered Format non-Registered Format	 I P	 R P	4
N_Port Name Format Registered Format non-Registered Format	 I P	 R P	4
Hunt Groups	P	P	
Multicast and Broadcast	P	P	8
Dedicated Simplex	P	P	

NOTES:

1 Link Control frames are Required only for Class 2. A Class 3-only N_Port shall be able to return a P_RJT in Class 2.

2 Values of ’01’b and ’11’b are Allowed based on the result of login. A value of ’10’b (for ACK_N) is Prohibited by this profile.

3 As per FC-PH, the size of a frame that is not the last frame of an Information Category shall be a multiple of four. A frame length of zero is Allowed.

4 All 64-bits of the Node Name and N_Port Name shall be significant. The Node Name and N_Port Name shall use a format that contains a worldwide unique identifier issued by a recognized registration authority. The IEEE, IEEE Extended, and IEEE Registered name formats are examples of acceptable formats. The Node Name and N_Port Name need not be related to each other in any way.

5 If the Sequence Recipient requires X_ID Interlock, the Sequence Initiator shall provide it.

6 Note that this means that, unlike FC-PLDA, the SCSI Initiator shall accept ABTS.

7 This is a placeholder for a format to be administered by the FCLC.

8 The use of broadcast and multicast to or from well-known addresses is prohibited.

6.2.6 Extended Link Services for N_Ports at well-known addresses

See Table 13 on page 39 for Extended Link Service support for each well-known address.

6.3 Other Fabric features

Table 19 lists other fabric features with usage defined by this document.

Table 19 – Other Fabric features

Feature	Fabric	Notes
Sequential Delivery to NL_Ports or N_Ports:		
Class 2	R	1
Class 3	R	1

NOTES:

1 Actually, the fabric shall forward deliverable frames to an NL_Port in the same order that they were received from the source port. If the source port sends frames out of Sequence order, the fabric shall not re-order them. Also, gaps occur if the fabric returns F_BSY or F_RJT to a frame, or if the destination returns P_BSY or P_RJT to a frame, or if a frame is invalid. This is the required behavior described in FC-PH, 23.6.7.2, bit 27. While FC-PH may require that the frames travel a “fixed route” across the fabric, this document does not particularly care how the function is implemented, as long as it is performed correctly.

6.4 Fabric_Port_Reject (F_RJT) frames

This clause describes the generation of F_RJT frames. The content is drawn substantially from the N_Port-to-F_Port Interoperability profile v1.0. See reference [18].

6.4.1 F_RJT frame delimiters

An F_RJT in response to an SOFc1 frame shall begin with SOFn1 and shall end with EOFdt. An F_RJT in response to a Class 2 frame shall begin with SOFn2 and shall end with EOFn. An F_RJT shall not be sent in Class 3. An F_RJT to any other Class of Service recognized by the FL_Port shall begin with SOFnX (where X is the Class) and shall end with EOFn.

6.4.2 Generation of F_RJT frames

Table 20 summarizes the F_RJT codes that may be generated by an FL_Port compliant with this report, the situations in which these codes may be generated, and the recovery action that shall be performed by the NL_Port. This table contains all Reason Codes applicable to Class 2; Reason Codes that are applicable to other Classes of Service are not listed here and shall not be generated by compliant FL_Ports.

Table 20 – F_RJT reason codes

Reason Code	Generated by Fabric in response to		N_Port/NL_Port Recovery Action	Notes
	before or during FLOGI	after FLOGI		
Invalid D_ID	Prohibited	The D_ID specified has not been assigned or is otherwise not valid	Abnormally terminate Sequence	1, 9
Invalid S_ID	a) Could not grant S_ID requested by NL_Port; OR b) NL_Port not logged into fabric	S_ID not equal to identifier assigned during FLOGI	Perform FLOGI	
N_Port or NL_Port not available, temporarily	Prohibited	Link recovery or loop initialization in progress at destination N_Port	Abnormally terminate Sequence	1, 2
N_Port or NL_Port not available, permanently	Prohibited	N_Port/NL_Port or F_Port/FL_Port off line, powered down, or nonexistent at destination	Abnormally terminate Sequence	1, 2
Class not supported	Prohibited	a) Class of Service not supported by FL_Port b) Broadcast frame (D_ID=hex ’FFFFFF’)received in Class other than Class 3	Not defined	3, 4
Delimiter usage error	Prohibited	Prohibited	none	5
TYPE not supported	Prohibited	Prohibited	none	5

Table 20 – F_RJT reason codes (continued)

Reason Code	Generated by Fabric in response to: before or during FLOGI	Generated by Fabric in response to: after FLOGI	N_Port/NL_Port Recovery Action	Notes
Incorrect Length	Class 3 FLOGI frame Data_Field size is larger than 256 bytes	Frame in a supported Class Data_Field size is larger than Maximum BB Receive Data Field Size	Not defined	4
Login required	NL_Port not logged into fabric	NL_Port no longer logged into fabric	Perform FLOGI	
Fabric path not available	Prohibited	Class 2 path to destination N_Port or NL_Port does not exist	Perform PLOGI again for Class 3 only	6
Class of Service not supported by entity at hex 'FFFFFE'	FLOGI was not sent in a supported Class	BLS or ELS not sent in a supported Class	Perform FLOGI, BLS, or ELS in Class 3	7
Invalid CS_CTL field	Prohibited	Prohibited	none	5, 8
Invalid Class of Service	Prohibited	Prohibited	none	5

NOTES:

1 The NL_Port shall release all local resources associated with the Sequence.

2 The NL_Port shall perform the appropriate ULP recovery, such as ABTS.

3 See “Class of Service not supported by entity at hex 'FFFFFE'” for the “before or during FLOGI” case.

4 This is a severe error that indicates a major inconsistency between the NL_Port and the FL_Port. The NL_Port recovery action is not defined by this document.

5 This Reason Code shall not be generated by an FL_Port.

6 This can occur when a fabric element that supports only Class 3 is used to join other fabric elements that support Classes 2 and 3. See 5.6.

7 The FL_Port is Allowed to accept FLOGI in Class 2, but NL_Ports are Prohibited from originating FLOGI in Class 2.

8 Non-zero CS_CTL bits are ignored, as required by FC-PH.

9 What likely happened is that the intended N_Port or NL_Port has had its ID changed. The Name Service may be used to find the N_Port or NL_Port and re-login with it.

6.5 FC-AL features for FL_Ports

Table 21 lists Fibre Channel Arbitrated Loop features for FL_Ports with usage defined by this document. In this table, “FL_Port Originator” refers to an FL_Port sending an FC-AL feature to an NL_Port, and “FL_Port Responder” refers to an FL_Port receiving an FC-AL feature from an NL_Port.

Table 21 – FC-AL features for FL_Ports

Feature	FL_Port Originator	FL_Port Recipient	Notes
Open Full Duplex	R	R	
Open Half Duplex	P	R	
Send frames to NL_Port for more than one S_ID in a single Loop Tenancy	I	-	
Receive frames from NL_Port for more than one D_ID in a single Loop Tenancy	-	R	
Unfairness	I	R	1
Transfer mode (use of TRANSFER loop state)	I	R	1
LILP/LIRP FLA-A Compliance FLA-B Compliance	 A R	 A R	 2
Broadcast and Multicast via Broadcast Replicate (OPNfr) To NL_Ports on the Local Loop From NL_Ports on the Local Loop	 A -	 - P	 11, 12 3, 11
Broadcast and Multicast via Selective Replicate (OPNyr) To NL_Ports on the Local Loop From NL_Ports on the Local Loop	 P -	 - P	 12 3
Alternate BB_Credit model	R	R	4
Login_BB_Credit=0	R	R	5
Login_BB_Credit>0	A	A	5

Table 21 – FC-AL features for FL_Ports (continued)

Feature	FL_Port Originator	FL_Port Recipient	Notes
LPEyx/LPByx/LPEfx	I	P	6
LIP			
(F7, F7) and (F7, AL_PS) [initializing]	I	R	7
(F8, F7) and (F8, AL_PS) [loop failure]	I	R	8
(AL_PD, AL_PS) [selective hard reset]	I	R	9
MRK	P	R	10

NOTES:

1 FL_Ports shall tolerate the use of Unfairness and Transfer Mode by other L_Ports. See 4.3.3 for some additional notes on the behavior of FL_Ports.

2 Some legacy devices that do not implement LILP and LIRP would otherwise be compliant with this profile.

3 NL_Ports may use Broadcast Replicate (OPNfr) or Selective Replicate (OPNyr) to communicate on a Local Loop. As required by FC-AL, the FL_Port shall not receive frames associated with OPNfr or OPNyr. Therefore, frames originated on a Local Loop by an NL_Port using OPNfr or OPNyr shall not propagate beyond the Local Loop. Use of broadcast and multicast is specific to the ULP.

4 Alternate BB_Credit management is mandatory in FC-AL

5 FL_Ports shall interoperate with NL_Ports which advertise any Login_BB_Credit. A recipient of frames may login with Login_BB_Credit>0, but the originator of frames is not Required to take advantage of it.

6 LPEfx is useful for resetting the bypass circuits of NL_Ports which have been bypassed and have lost their AL_PAs due to a LIP (and therefore cannot be enabled using an addressed LPEyx).

7 This LIP may be issued by an FL_Port to request an AL_PA if it has none, or if it is unable to win arbitration within A_W_TOV. FL_Port response to a LIP is described in 6.6.

8 This LIP may be issued by an FL_Port which detects a Link Failure, as defined by FC-PH.

9 An NL_Port or FL_Port may invoke this form of LIP to reset an NL_Port in a ULP-specific manner. An FL_Port shall respond to this form of LIP in the same manner as LIP(F7,AL_PS).

10 An FL_Port that receives MRK shall attempt to forward the MRK. FC-AL states that an L_Port may remove a MRK if necessary for clock skew management.

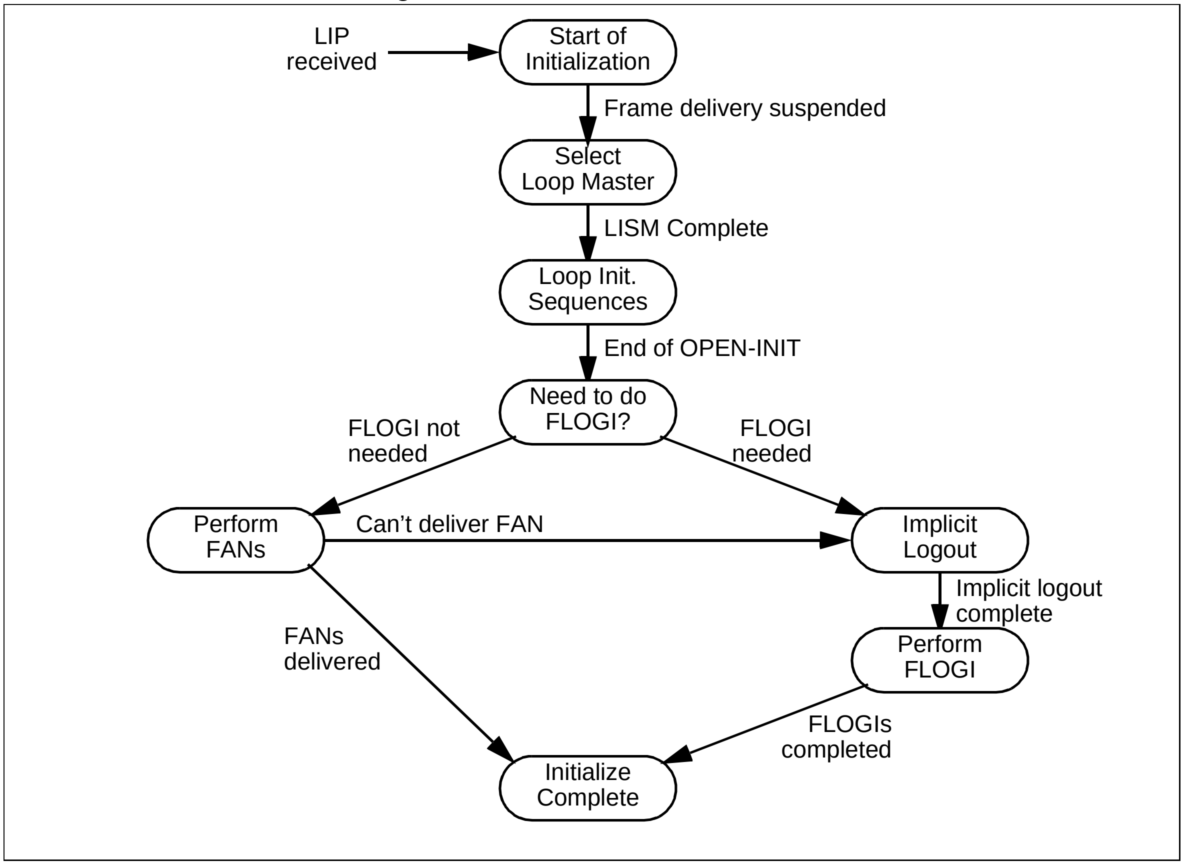
11 As required by FC-AL, an NL_Port with broadcast or multicast frames intended for Remote Ports shall open the FL_Port via OPNyx or OPNyy (or be opened by the FL_Port via OPNyx) and send the frames to the FL_Port for forwarding by the fabric. Note that this could cause the frames to be delivered to the Local Loop if any NL_Ports on the Local Loop are part of the broadcast or multicast group.

12 The only method Allowed by this document of delivering broadcast or multicast from the FL_Port to the attached NL_Ports is via an OPNfr sent by the FL_Port. The FL_Port is not Required to perform a discovery process to determine whether the attached NL_Ports are able to receive broadcast or multicast frames via OPNfr. Use of broadcast and multicast, and the number of frames that may be sent following OPNfr, are both specific to the ULP.

6.6 FL_Port initialization features

FL_Ports shall initialize as detailed below. Figure 3 shows a schematic of the process to illustrate the flow. If the figure is different than the text, the text shall apply.

Figure 3 – FL_Port initialization flow



a) **Start of Initialization**. Whenever an FL_Port receives any LIP, it shall begin initialization. The FL_Port shall suspend delivery of frames to destination NL_Ports on the Local Loop. The FL_Port shall proceed to **step (b)**.

b) **Selection of Loop Master**. At the start of Loop Initialization (as defined in FC-AL clause 10), the FL_Port shall compete to become Loop Master as defined in FC-AL. The FL_Port shall proceed to **step (c)**.

NOTE – The FL_Port is expected to win, as described by FC-AL. If it doesn't, then there is more than one FL_Port on the loop.

c) **Loop Initialization Sequences**. The FL_Port shall perform the LIFA, LIPA, LIHA, and LISA Loop Initialization Sequences. It may also perform LIRP and LILP if allowed by the Loop Initialization protocol as defined in FC-AL. The FL_Port shall set the L_bit in at least the LISA Sequence if the fabric is being initialized for the first time, or if the F_Port_Name or Fabric_Name of the FL_Port changed to a different value than it had prior to the Loop Initialization, or for the same reasons that would cause an F_Port within the fabric to generate an OLS (e.g., when it loses state information for the attached loop). The FL_Port shall proceed to **step (d)**.

d) **Need to do FLOGI**. After completion of Loop Initialization, if the L_bit was set to one in the LISA Sequence during Loop Initialization, the FL_Port shall proceed to **step (e)**. Otherwise, the FL_Port shall proceed to **step (g)**.

e) **Perform implicit logout**. This step is performed if the FL_Port required all NL_Ports to perform FLOGI. The FL_Port shall implicitly logout all NL_Ports attached to the Local Loop. The FL_Port shall also communicate to the Name Service (well-known address hex 'FFFFFC') information that causes all NL_Ports attached to the Local Loop to be de-registered. The FL_Port shall proceed with **step (f)**.

f) **Perform FLOGI**. The Fabric may (but is not Required to) discard frames destined for NL_Ports attached to this FL_Port. The Fabric shall consider that an NL_Port is a Private NL_Port until it completes FLOGI. The Fabric shall consider FLOGI complete with an NL_Port when the FL_Port successfully transmits the FLOGI ACC to the NL_Port (see Note below). The FL_Port shall attempt to send FLOGI ACC to the NL_Port by opening the AL_PA of the NL_Port full duplex via OPN(AL_PD, hex '00'), initiating the FLOGI accept Sequence, and originating a CLS. The FL_Port may initiate the FLOGI accept Sequence if already open full duplex with the NL_Port, before forwarding the CLS sent by the NL_Port following the FLOGI request Sequence. The FL_Port shall also communicate to the Name Service (well-known address hex 'FFFFFC') information from the FLOGI that causes the following objects to be registered: Native Port Identifier and Port Type. The FL_Port may communicate to the Name Service information from the FLOGI that causes the following objects to be registered: Port_Name, Node_Name, and Supported Classes of Service. The FL_Port shall proceed to **step (h)**.

NOTE – The FLOGI completion behavior described above is a deviation from FC-PH. The requirement in FC-PH is to consider FLOGI complete when the F_Port receives the R_RDY for the FLOGI ACC. Since Alternate BB_Credit Management does not require credit balancing before CLS, the R_RDY cannot be relied upon.

g) **Perform FAN**. This step is performed if the FL_Port did not require all NL_Ports to perform FLOGI. The FL_Port shall initiate a FAN (Fabric Address Notification) ELS to each AL_PA on the Local Loop which had an NL_Port attached that had completed FLOGI prior to the initialization. The FL_Port shall begin sending FAN immediately after the completion of the Loop Initialization Protocol, and before any other node is able to send a frame on the loop. The FAN ELS shall be delivered to each NL_Port without allowing any other node to send a frame until all FAN ELS have been sent. The Fabric shall not change the entries within the Name Service for any NL_Ports to which the FL_Port successfully sends the FAN. The Fabric shall not discard frames destined for NL_Ports to which the FL_Port successfully delivers the FAN, except as required due to normal fabric congestion management. If the FL_Port is unable to deliver the FAN to an NL_Port, the fabric may implicitly logout the NL_Port and perform **step (e)** and **step (f)** on the NL_Port. The FL_Port shall proceed to **step (h)**.

NOTE – The FL_Port meets the requirement to send FAN without allowing other activity by arbitrating immediately after the CLS is sent following the OPEN-INIT state, winning, and entering the OPEN state to send the FAN to the first NL_Port. Subsequent FANs are sent by sending CLS, entering the TRANSFER state, and sending OPN for the next NL_Port. If the FL_Port is unable to enter the TRANSFER state, it may instead arbitrate unfairly and win the loop to send subsequent FANs without releasing control of the loop. This process is repeated until all FANs have been sent.

h) **Initialize complete**. The FL_Port has completed the initialization. The FL_Port shall resume delivery of frames to destination NL_Ports on the Local Loop. The FL_Port shall accept FLOGI from attached NL_Ports. The FL_Port shall accept requests to all well-known addresses.

7 N_Port feature sets

The tables in this clause list features described in the Fibre Channel Physical and Signaling Interface standard (FC-PH) specific to the operation of N_Ports that are communicating across a fabric with Public NL_Ports. These tables indicate whether the feature is Required, Prohibited, Allowed, or Invocable for compliance with this specification. Features that are not listed do not affect interoperability of N_Ports with Public NL_Ports.

PLEASE READ 3.4 for the exact definition of Prohibited, Required, Allowed, and Invocable, and for the legend for table entries used in this clause.

7.1 PLOGI features and parameters for N_Ports

This clause specifies the parameters sent by an N_Port to an NL_Port to request a login. Table 22 lists the login request parameters requested by N_Ports with usage defined by this document. In this table, “N_Port Originator” refers to an N_Port issuing a PLOGI request to a Public NL_Port, and “N_Port Responder” refers to an N_Port issuing a PLOGI ACC to a Public NL_Port that sent a PLOGI request.

Table 22 – PLOGI features for N_Ports

Feature	N_Port Originator	N_Port Responder	Notes
Class of service for PLOGI (N_Port Originator) and PLOGI ACC (N_Port Responder)			1, 2
Class 1	P	P	
Class 2	A	A	
Class 3	R	R	
Class 4 and up	P	P	
Supported Classes of Service:			3
Class 1	P	P	
Class 2	A	A	
Class 3	R	R	
Class 4 and up	P	P	

NOTES:

1 If an N_Port follows the convention of attempting PLOGI first in Class 2 (Class 1 presumably being skipped because the fabric does not support it), it shall tolerate a rejection by the receiving NL_Port and proceed to attempt PLOGI in Class 3. Likewise, the NL_Port shall also be tolerant of this behavior.

2 If an N_Port receives a PLOGI in a Class of Service other than Class 3, and the N_Port does not support PLOGI in that Class, it shall return a P_RJT with an Action Code of “Rejected frame may be retryable” and a Reason Code of “Class not supported”. The N_Port shall tolerate this behavior; the expectation is that an older N_Port is trying Class 2 first before trying Class 3.

3 An N_Port shall not advertise support for a Class of Service that is not supported by the fabric (as indicated by FLOGI ACC) when sending a PLOGI or PLOGI ACC to any N_Port or NL_Port.

7.1.1 PLOGI Common Service Parameters for N_Ports

Table 23 lists N_Port Common Service Parameters for N_Ports logging in with NL_Ports with usage defined by this document. In this table, “N_Port Originator or Responder” refers to either an N_Port issuing a PLOGI request to a Public NL_Port, or to an N_Port issuing a PLOGI ACC to a Public NL_Port that sent a PLOGI request.

Table 23 – PLOGI Common Service Parameters for N_Ports

Common Service Parameter	N_Port Originator or Responder	Notes
FC-PH Version Highest Version Lowest Version	 X 20 hex	
BB_Credit (min)	-	2
Continuously Increasing Relative Offset	1	
Random Relative Offset	X	3
Valid Vendor Version Level	0	1
N_Port/F_Port	0	
Alternate BB_Credit Management	-	2
E_D_TOV Resolution	-	2
Payload Length	0	
BB Receive Data Field Size (min)	-	2
Total Concurrent Sequences (min)	1	
Relative Offset by Information Category	R	
E_D_TOV	-	2

NOTES:

1 Profile versions are communicated via the Report Node Capabilities (RNC) ELS. The Valid Vendor Version Level shall not be set to 1 for the purposes of communicating support of this Technical Report.

2 This field does not apply to a direct fabric-attached device.

3 The value for this parameter is determined by the ULP. See Annex B.

7.1.2 PLOGI Class 2 Service Parameters

Table 24 lists Class 2 Service Parameters with usage defined by this document. In this table, “N_Port Originator or Responder” refers to either an N_Port issuing a PLOGI request to a Public NL_Port, or to an N_Port issuing a PLOGI ACC to a Public NL_Port that sent a PLOGI request.

Table 24 – PLOGI Class 2 Service Parameters for N_Ports

Class 2 Service Parameter	N_Port Originator or Responder	Notes
Priority/Preemption	0	
X_ID Reassignment	’00’b	
Initial Process Associator	’00’b	
ACK Initiator Capability ACK_0 ACK_N	 X 0	
ACK Generation Assistance	X	1
ACK Recipient Capability ACK_0 ACK_N	 X 0	
Initiator Data compression capable	0	
Initiator Data compression History buffer size	-	
Initiator Data Encryption capable	0	
X_ID Interlock	X	
Recipient Error Policy Supported	’00’b	
Recipient Categories per Sequence	’00’b	
Recipient Data compression capable	0	
Recipient Data compression History buffer size	-	
Recipient Data Encryption capable	0	
Maximum Receive data field size (min)	256	
Concurrent Sequences (min)	1	
End to End Credit (min)	1	
Open Sequences per Exchange (min)	1	

NOTES:

1 ACK generation assistance is useful for ACK_0 generation in Class 2. ACK generation assistance is Allowed by this document.

7.1.3 PLOGI Class 3 Service Parameters

Table 25 lists Class 3 Service Parameters with usage defined by this document. In this table, “N_Port Originator or Responder” refers to either an N_Port issuing a PLOGI request to a Public NL_Port, or to an N_Port issuing a PLOGI ACC to a Public NL_Port that sent a PLOGI request.

Table 25 – PLOGI Class 3 Service Parameters for N_Ports

Class 3 Service Parameter	N_Port Originator or Responder	Notes
Initial Process Associator	’00’b	
Initiator Data compression capable	0	
Initiator Data compression History buffer size	-	
Initiator Data Encryption capable	0	
Recipient Error Policy Supported	’00’b	
Recipient Categories per Sequence	’00’b	
Recipient Data compression capable	0	
Recipient Data compression History buffer size	-	
Recipient Data Encryption capable	0	
Maximum Receive data field size (min)	256	
Concurrent Sequences (min)	1	
Open Sequences per Exchange (min)	1	
NOTES: (none)		

7.2 Other N_Port FC-PH features

Table 26 lists other FC-PH features not covered in previous clauses for N_Ports with usage defined by this document. In this table, “N_Port Originator” refers to an N_Port sending a Frame or Sequence to a Public NL_Port, and “N_Port Responder” refers to an N_Port receiving a Frame or Sequence from a Public NL_Port.

Table 26 – Other FC-PH features for N_Ports

Feature	N_Port Originator	N_Port Responder	Notes
X_ID reassignment	P	A	
X_ID invalidation	P	A	
R_CTL Routing Bits			
FC-4 Device_Data frame	R	R	
Extended Link_Data frame	R	R	
Basic Link_ Data frame	R	R	
Link_Control frame	R	R	1
TYPE field set to any value	A	A	
Link Control frames			1
ACK_0	A	A	
ACK_1	R	R	
ACK_N	P	P	
F_BSY	P	R	
F_RJT	P	R	
P_BSY	I	R	
P_RJT	I	R	
LCR	R	R	
NTY	P	P	
END	P	P	
Basic Link Services			
ABTS	I	R	6
BA_ACC	R	R	
BA_RJT	R	R	
NOP	P	P	
RMC	P	P	
PRMT	P	P	
Nonzero Continue Sequence Condition values	P	-	
Ignore nonzero Continue Sequence values	-	R	
X_ID Interlock in Class 2	R	A	5
ACK Assist (ACK_Form in F_CTL) in Class 2			2
ACK_Form = no assist	A	A	
ACK_Form = ACK_1 expected	A	A	
ACK_Form = ACK_N expected	P	P	
ACK_Form = ACK_0 expected	A	A	
Sequence Retransmission (in F_CTL)	P	P	
Abort Sequence Condition ≠ ’00’b	A	A	
Optional Headers (all)	A	A	9
Payload size when frame is not the last frame of a sequence	(size MOD 4) == 0	(size MOD 4) == 0	3

Please read the definitions of Prohibited, Required, Allowed, and Invocable in clause 3.4 on page 6!

Table 26 – Other FC-PH features for N_Ports (continued)

Feature	N_Port Originator	N_Port Responder	Notes
Node Name Format Registered Format non-Registered Format	 I P	 R P	 4
N_Port Name Format Registered Format non-Registered Format	 I P	 R P	 4
Hunt Groups	P	P	
Multicast and Broadcast	A	A	8
Dedicated Simplex	P	P	

NOTES:

1 Link Control frames are Required only for Class 2. A Class 3-only N_Port shall be able to return a P_RJT in Class 2.

2 Values of ’01’b and ’11’b are Allowed based on the result of login. A value of ’10’b (for ACK_N) is Prohibited by this profile.

3 As per FC-PH, the size of a frame that is not the last frame of an Information Category shall be a multiple of four. A frame length of zero is Allowed.

4 All 64-bits of the Node Name and N_Port Name shall be significant. The Node Name and N_Port Name shall use a format that contains a worldwide unique identifier issued by a recognized registration authority. The IEEE, IEEE Extended, and IEEE Registered name formats are examples of acceptable formats. The Node Name and N_Port Name need not be related to each other in any way.

5 If the Sequence Recipient requires X_ID Interlock, the Sequence Initiator shall provide it.

6 Note that this means that, unlike FC-PLDA, the SCSI Target may invoke ABTS, and the SCSI Initiator shall accept it.

7 This is a placeholder for a format to be administered by the FCLC.

8 The use of broadcast and multicast is specific to the ULP.

9 The use of optional headers is specific to the ULP.

7.3 P_RJT and F_BSY/P_BSY frames

Compliant N_Ports operating in Class 2 may return a P_RJT in response to any received Class 2 frame, for any of the reason codes defined in FC-PH, FC-PH-2, and FC-PH-3. Compliant N_Ports operating in Class 2 may receive a P_RJT in response to any Class 2 frame sent by the N_Port, for any of the reason codes defined in FC-PH, FC-PH-2, and FC-PH-3.

Compliant N_Ports operating in Class 2 may return a P_BSY in response to any received Class 2 frame, for any of the reason codes defined in FC-PH, FC-PH-2, and FC-PH-3. Compliant N_Ports operating in Class 2 may receive a P_BSY or F_BSY in response to any Class 2 frame sent by the N_Port, for any of the reason codes defined in FC-PH, FC-PH-2, and FC-PH-3.

An N_Port is Allowed to retransmit individual frames in response to P_RJT, F_BSY, or P_BSY, if one of the following criteria is met:

- the frame is the first frame of the first Sequence of an Exchange, and X_ID Interlock is Required;

- the frame is the only frame of a single frame Sequence.

Otherwise, an N_Port that receives a P_RJT, F_BSY, or P_BSY in response to a Class 2 frame shall abort the Sequence (via ABTS) and retransmit the Sequence or abort the Exchange, as Allowed or Required by the ULP.

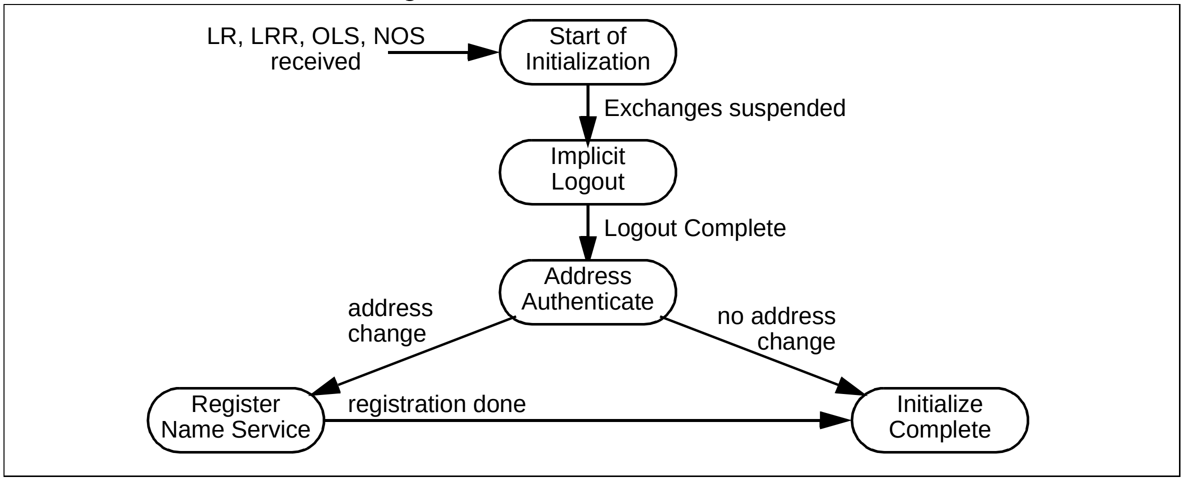
An N_Port that sends a P_RJT or P_BSY shall expect the Sequence and/or Exchange to be aborted by the other NL_Port or N_Port. An N_Port that sends a P_RJT or P_BSY shall expect the frame to be re-transmitted if the frame meets one of the above criteria. The Sequence may be retransmitted after the ABTS protocol is complete, if Allowed or Required by the ULP.

N_Port response to F_RJT is defined in 6.4.

7.4 N_Port initialization features

N_Ports shall initialize as detailed below. Figure 2 shows a schematic of the process to illustrate the flow. If the figure is different than the text, the text shall apply.

Figure 4 – N_Port initialization flow



a) **Start of Initialization**. Whenever an N_Port receives an LR, LRR, OLS, or NOS Primitive Sequence, it shall begin this initialization process. The N_Port shall suspend all open Exchanges with all other N_Ports and NL_Ports. (The behavior of the N_Port if it receives any other Primitive Sequence is not addressed by this document.) The N_Port shall proceed to **step (b)**.

b) **Implicit logout**. If an OLS or NOS is received, or if LR, LRR, OLS, or NOS is received and the N_Port had not completed FLOGI prior to the start of initialization, the N_Port shall be implicitly logged out from the fabric. The N_Port shall proceed to **step (c)**.

c) **Address authentication**. After completion of Link Initialization, the N_Port shall perform FLOGI if it was implicitly logged out from the fabric in **step (b)**. If FLOGI is not required, the N_Port may perform FLOGI for internally generated reasons, otherwise the N_Port shall perform FDISC. After the FLOGI or FDISC is completed successfully (the N_Port has received FLOGI ACC or FDISC ACC):

   1) If the N_Port has the same N_Port_ID, N_Port_Name and Node_Name that it had before initialization, and the N_Port also determines that the F_Port has the same address, F_Port_Name and Fabric_Name that the FL_Port had before initialization, the N_Port shall resume all suspended Exchanges with all other N_Ports and NL_Ports. The N_Port shall proceed to **step (e)**.

   2) If the N_Port does not have the same N_Port_ID, N_Port_Name or Node_Name that it had before initialization, or the N_Port determines that the F_Port does not have the same address, F_Port_Name or Fabric_Name that the FL_Port had before initialization, the N_Port shall discard all pending Exchanges in a manner consistent with the ULP, implicitly LOGO with all other N_Ports and NL_Ports, and perform ULP-specific actions to clear pending tasks. The N_Port shall wait for R_A_TOV before originating any new Exchanges. During this time, the N_Port shall discard all frames received except for the PLOGI and ADISC request Sequences, and LOGO ACC reply Sequences. The N_Port shall respond to all PLOGI request Sequences with an LS_RJT reply Sequence, with a Reason code of “Unable to perform Command Request at this time”. The N_Port shall respond to all ADISC request Se-

quences with a LOGO request Sequence to the N_Port or NL_Port that initiated the PLOGI or ADISC request Sequence. The N_Port shall proceed to **step (d)**.

d) **Registration**. The N_Port shall perform PLOGI explicitly with the Directory Server (well-known address hex 'FFFFFC') and attempt an RFC-4 request with the Name Service. If the attempted RFC-4 fails, the N_Port is not Required to retry the attempt. The N_Port may at this time perform any other Directory Server registration requests it may wish to do. The N_Port shall proceed to **step (e)**.

e) **Initialization complete**. This completes initialization for the N_Port. The N_Port may proceed to originate and respond to Exchanges, and login with other N_Ports and NL_Ports as needed.

An N_Port that wishes to login with an NL_Port or N_Port shall perform the following steps:

a) **Discovery**. As required by the ULP, the N_Port may perform a discovery process to locate N_Ports and NL_Ports, and perform PLOGI with those N_Ports and/or NL_Ports. The GP_ID4 service provided by the Name Service may be used as part of this process.

b) **NL_Port Login**. If the N_Port wishes to use Class 2 to communicate with an N_Port or NL_Port, and FLOGI with the F_Port establishes that the F_Port supports Class 2, the N_Port shall PLOGI with that port in Class 2. If the PLOGI ACC is received successfully, the N_Port may use Class 2 with the other port; if the fabric indicates that a Fabric Path is Not Available (see 6.4), then Class 2 is not available with that N_Port/NL_Port. If the N_Port only requires Class 3, then PLOGI shall be performed in Class 3.

7.5 Extended Link Services

Table 27 lists Extended Link Services with usage defined by this document. Devices that receive requests for Extended Link Services that are unsupported shall return LS_RJT with a reason code “Invalid Command Code”. In this table, “N_Port Originator” refers to an N_Port sending an Extended Link Service request to a Public NL_Port, and “N_Port Responder” refers to an N_Port receiving an Extended Link Service request from a Public NL_Port.

Table 27 – Extended Link Services for N_Ports

Feature	N_Port Originator	N_Port Responder	Notes
ABTX	P	P	
ADISC	I	R	1
ADVC	P	P	
ECHO	A	A	
ESTC	P	P	
ESTS	P	P	
FACT	P	P	
FAN	P	P	3
FDACT	P	P	
FDISC	I	P	2
FLOGI	R	P	2
GAID	P	P	
LINIT	I	P	3
LOGO	R	R	
LPC	I	P	3
LSTS	I	P	3
NACT	P	P	
NDACT	P	P	
PDISC	I	R	1
PLOGI	R	R	
PRLI	A	A	5
PRLO	A	A	5
QoSR	P	P	
RCS	P	P	
RES	P	P	
RLS	I	A	4
RNC (previously called RVU)	R	R	
RRQ	R	R	
RSCN	A	A	6
RSI	P	P	
RSS	P	P	
RTV	P	P	
RVCS	P	P	
SCN	A	A	
SCR	A	A	6

Table 27 – Extended Link Services for N_Ports (continued)

Feature	N_Port Originator	N_Port Responder	Notes
TEST	A	A	
TPLS	P	P	
TPRLO	A	A	5

NOTES:

1 Address authentication for N_Ports is described in 7.4.

2 N_Ports do not receive FDISC or FLOGI requests.

3 These are new ELS provided for loop management (see Annex A).

4 Not all fields in the LESB may be supported (see vendor-specific documentation).

5 Support for PRLI, PRLO and TPRLO is ULP-dependent. See the specific ULP standards and profiles.

6 These are new optional ELS to allow registration and notification of state changes (see Annex A).

7.6 Well-known address usage by N_Ports

Table 11 lists the well-known address requirements for N_Ports with usage defined by this document. In this table, “N_Port Originator” refers to an N_Port sending a request Sequence to an N_Port at a well-known address, and “N_Port Responder” refers to an N_Port receiving a request Sequence from an N_Port at a well-known address.

Table 28 – Well-known address support for N_Ports

Parameter	N_Port Originator	N_Port Responder	Notes
Well-Known Address hex ’FFFFFF’ (Broadcast)	A	A	4, 7
Well-Known Address hex ’FFFFFE’ (“Fabric F_Port”)	R	R	1, 4
Extended Link Services:			
ABTS	I	R	
FAN	P	P	
FDISC	I	P	
FLOGI	R	P	
RRQ	I	R	
All other Extended Link Services	P	P	
Well-Known Address hex ’FFFFFD’ (“Fabric Controller”)	R	R	
Extended Link Services:			
ECHO	A	A	
PLOGI	A	P	
RSCN	I	A	
SCN	A	A	
SCR	I	P	
TEST	A	A	
All other Extended Link Services	P	P	
Well-Known Address hex ’FFFFFC’ (“Directory Server”)	R	P	
Extended Link Services:			
ECHO	A	A	
LOGO	I	P	
PDISC	I	P	
PLOGI	R	P	
RLS	A	A	
RNC	R	R	
RRQ	R	R	
TEST	A	A	
All other Extended Link Services	P	P	
Name Service			2
RFC-4	R	P	5
GPT	I	P	
GP_ID4, Port_Type = “Nx_Port”	I	P	
GP_ID4, other Port_Types	A	P	
All other services	A	P	
X.500 Service	A	P	
ARP Service	A	P	
Well-Known Address hex ’FFFFFB’ (Time Server)	P	P	
Well-Known Address hex ’FFFFFA’ (Management Server)	P	P	
Well-Known Address hex ’FFFFF9’ (Quality of Service Facilitator)	P	P	3

Table 28 – Well-known address support for N_Ports (continued)

Parameter	N_Port Originator	N_Port Responder	Notes
Well-Known Address hex ’FFFFF8’ (Alias Server)	A	A	6
Well-Known Address hex ’FFFFF7’ (Security Key Distribution Server)	P	P	
Well-Known Address hex ’FFFFF6’ (Clock Synchronization Server)	P	P	

NOTES:

1 This is the address to which an N_Port or NL_Port sends a FLOGI.

2 The Name Service is defined in FC-GS-2 (see reference [12]). See Annex C for examples of Name Services.

3 This service is currently defined only for Class 4 service requests.

4 PLOGI is not needed and shall not be used prior to using the services at these addresses.

5 An NL_Port is required to attempt this request. See 7.4.

6 The Alias Server may be used to support Multicast and Broadcast.

7 Use of broadcast is specific to the ULP.

8 Timers on a Public Loop

This clause defines the times and timeout values used for public loop implementations. Throughout this clause, references are made to FC-PLDA. In general, the requirements stated in FC-PLDA also apply to FC-FLA compliant devices. Within this clause, only the differences from FC-PLDA are noted.

8.1 Loop Initialization Sequence Hold Time (LIS_HOLD_TIME)

The requirements stated in FC-PLDA apply.

8.2 Error Detect Timeout (E_D_TOV)

Prior to the completion of FLOGI, NL_Ports shall use the value of E_D_TOV Required by FC-PLDA. Prior to the completion of FLOGI, N_Ports shall use their default value of E_D_TOV. After the completion of FLOGI, NL_Ports and N_Ports shall use the value of E_D_TOV specified by the fabric in the FLOGI ACC. The recommended value for E_D_TOV in the FLOGI ACC is 2 seconds.

8.3 Resource Allocation Timeout (R_A_TOV)

Prior to the completion of FLOGI, NL_Ports shall use the value of R_A_TOV Required by FC-PLDA. Prior to the completion of FLOGI, N_Ports shall use their default value of R_A_TOV. After the completion of FLOGI, NL_Ports and N_Ports shall use the value of R_A_TOV specified by the fabric in the FLOGI ACC; this value shall be used for all purposes (there is no split definition as in FC-PLDA). The recommended value for R_A_TOV in the FLOGI ACC is 10 seconds.

8.4 Resource Recovery Timer (RR_TOV)

The requirements stated in FC-PLDA apply.

8.5 ULP Timer (ULP_TOV)

The requirements stated in FC-PLDA apply for SCSI-FCP NL_Ports and N_Ports. No requirement is made for IP NL_Ports and N_Ports, or for any other ULP.

Annex A
(normative)

Additional Services for Public Loop Operation

This annex defines additional proposed Extended Link Services that are deemed useful for the support of Arbitrated Loops that form part of a Fabric. It is expected that they will incorporated in a future standard when and if they are approved. The statements of support (Required, Allowed, etc.) within this annex presume acceptance of these services.

A.1 Extended Link Services

The following new Extended Link Service is Required for operation on a Public Loop, and shall be implemented by all FL_Ports and NL_Ports that comply with this report:

– Fabric Address Notification (FAN): Informs all NL_Ports of the current address of the FL_Port.

The following Extended Link Services are Allowed for operation on or with a Public Loop, and may be implemented by FL_Ports and NL_Ports that comply with this report:

– Loop Initialize (LINIT): Enables a requestor to initiate Loop Initialization.

– Loop Port Control (LPC): Enables a requestor to bypass or enable Loop Devices on a given Loop.

– Loop Status (LSTS): Enables a requestor to ascertain the state of a given Loop and its Loop Devices.

– Registered State Change Notification (RSCN): Sent when a change occurs at an N_Port or NL_Port.

– State Change Registration (SCR): Allows an N_Port or NL_Ports to indicate that it would like to receive the Registered State Change Notification (RSCN) ELS.

A.1.1 Fabric Address Notification (FAN)

The Fabric Address Notification Extended Link Service is sent by the Fabric F_Port (well-known address hex ’FFFFFE’) to all known previously logged in (FLOGI) and attached ports following an initialization event. (This initialization event is typically Loop Initialization on an Arbitrated Loop, though other events that may cause a port to change its ID may also be considered.) The Fabric F_Port shall report the Fabric_Port_Name and Fabric_Name as they were reported in the prior login (FLOGI), and shall report the current Loop Fabric Address if the fabric attach is an FL_Port. The Fabric F_Port shall send this service using the default login parameters; i.e., the parameters that are in effect prior to a FLOGI request.

The attached ports may use this information to authenticate active Exchanges and operating parameters (e.g., login BB_Credit).

The Fabric F_Port shall report identical information to all attached ports. If the information changes in any way before the Fabric F_Port is able to send the service to all attached ports, the Fabric F_Port shall initiate a new initialization event.

The attached ports shall not initiate an Accept Reply Sequence to this Extended Link Service. A Service Reject may be initiated if the port does not recognize or support this service.

Protocol:

Fabric Address Notification request Sequence
No Reply Sequence

Format: FT–1

Addressing: The S_ID is the Fabric F_Port (hex 'FFFFFE'). The D_ID is the N_Port or NL_Port receiving the notification.

Payload: The format of the FAN request Payload is shown in table A.1.

Table A.1 – FAN Payload

Item	Size Bytes
hex '60000000'	4
Reserved	1
Loop Fabric Address	3
Fabric_Port_Name	8
Fabric_Name	8

Loop Fabric Address: The Loop Fabric Address (see 3.1.14) of the Local Loop.

Fabric_Port_Name: As defined in FC-PH.

Fabric_Name: As defined in FC-PH.

A.1.2 Loop Initialize (LINIT)

The Loop Initialize Extended Link Service requests the initialization of Loop Initialization on a designated loop. The FL_Port connected to the loop addressed by the Loop Fabric Address shall originate a LIP of the type specified in the Payload and enter the INITIALIZING Arbitrated Loop state. The Accept Reply Sequence shall not be initiated before the FL_Port has transitioned from the OPEN-INIT to MONITORING state, and has completed sending any FAN ELS.

Protocol:

Loop Initialize request Sequence
Accept (ACC) Reply Sequence

Format: FT–1

Addressing: The S_ID designates the N_Port or NL_Port requesting Loop Initialization of the loop. The D_ID field shall be the Loop Fabric Address (LFA) of the loop to be initialized.

Payload: The format of the LINIT request Payload is shown in table A.2.

Table A.2 – LINIT Payload

Item	Size Bytes
hex ‘70000000’	4
Reserved	1
Initialization Function	1
LIP Byte 3	1
LIP Byte 4	1

Initialization Function: This field defines modifications of the initialization to be performed. The format of the Initialization Function is shown in table A.3.

Table A.3 – Initialization Function

Function	Value
Normal Initialization - The fabric determines the best method by which to complete the initialization.	0
Force Login - The L_bit shall be set in the LISA Sequence to force all affected NL_Ports to perform FLOGI.	1
Reserved	2 - 255

LIP Byte 3: The 3rd character of the LIP Primitive Sequence to be originated.

LIP Byte 4: The 4th character of the LIP Primitive Sequence to be originated.

NOTE – LIP Byte 3 and LIP Byte 4 should only be set to values permitted by FC-AL or FC-AL-2.

Reply Extended Link Service Sequence:

Service Reject (LS_RJT)
  Signifies the rejection of the LINIT command
Accept (ACC)
  Signifies acceptance of the LINIT request and initiation of Loop Initialization.
– Accept Payload

The format of the Accept Payload is shown in table A.4.

Table A.4 – LINIT Accept Payload

Item	Size Bytes
hex ‘02000000’	4
Reserved	3
Status	1

Status: The format of the Status field is shown in table A.5.

Table A.5 – LINIT Status

State	Value
Reserved	0
Success - The requested function was completed.	1
Failure - The requested function could not be completed.	2
Reserved	3 - 255

A.1.3 Loop Port Control (LPC)

The Loop Port Control Extended Link Service requests the Loop Bypass and/or Loop Enable of specific L_Ports on a designated loop. The FL_Port connected to the loop addressed by the Loop Fabric Address shall gain access to the loop in the manner selected by the requester, and originate LPB and/or LPE Primitive Sequences to the AL_PAs specified in the Payload.

The Accept Reply Sequence shall not be initiated before the FL_Port has either originated all requested LPBs and LPEs and left the loop in the state selected by the requester, or has detected a failure to deliver any LPB or LPE.

Protocol:

Loop Port Control request Sequence
Accept (ACC) Reply Sequence

Format: FT–1

Addressing: The S_ID designates the N_Port or NL_Port requesting the bypass and/or enable of Loop Devices on a loop. The D_ID field shall be the Loop Fabric Address (LFA) of the loop with attached L_Ports to be bypassed and/or enabled.

Payload: The format of the LPC request Payload is shown in table A.6.

Table A.6 – LPC Payload

Item	Size Bytes
hex '71000000'	4
Reserved	3
Port Control Function	1
Loop Devices to Bypass (LPB) (AL_PA bit map)	16
Loop Devices to Enable (LPE) (AL_PA bit map)	16

Port Control Function: A code that indicates how to perform the bypass function. The format of the Port Control Function is shown in table A.7.

Table A.7 – Port Control Function

Function	Value
Reserved	0
Normal Bypass	1
Initialize and Bypass	2
Reserved	3 - 255

The "Normal Bypass" function requests the FL_Port to Arbitrate for the loop, win, send OPN(0,0), receive OPN(0,0), and then issue the requested LPB and LPE Primitive Sequences as defined in FC-AL. After all LPB and LPE Primitive Sequences have been delivered the FL_Port shall issue a CLS and return to the MONITORING state. If the FL_Port is unable to successfully deliver any requested LPB or LPE Primitive Sequence within two AL_TIME delays (transmit the Primitive Sequence and receive it back), the FL_Port shall attempt to return to the MONITORING state and indicate Failure status.

The "Initialize and Bypass" function requests the FL_Port to enter the INITIALIZING state (send at least 12 LIP(F7,F7) Primitive Sequences), and then issue the requested LPB and LPE Primitive Sequences as defined in FC-AL. After all LPB and LPE Primitive Sequences have been delivered the FL_Port shall re-enter the INITIALIZING state (send at least 12 LIP(F7,F7) Primitive Sequences), and then attempt to complete Loop Initialization. If the FL_Port is unable to successfully deliver any requested LPB or LPE Primitive Sequence within two AL_TIME delays (transmit the Primitive Sequence and receive it back), the FL_Port shall continue to attempt to complete Loop Initialization and indicate Failure status.

Loop Devices to Bypass: The format of this field follows the AL_PA bit mapped format defined in FC–AL (see FC–AL table 15). If a bit is set to one in this field, then an LPByx Primitive Sequence shall be originated to the corresponding AL_PA by the FL_Port.

Loop Devices to Enable: The format of this field follows the AL_PA bit mapped format defined in FC–AL (see FC–AL table 15). If a bit is set to one in this field, then an LPEyx Primitive Sequence shall be originated to the corresponding AL_PA by the FL_Port. If Word 0 bit 31 is set to one then the LPEfx Primitive Sequence shall be originated by the FL_Port.

Reply Extended Link Service Sequence:

Service Reject (LS_RJT)
Signifies the rejection of the LPC command
Accept (ACC)
Signifies acceptance of the LPC request.
– Accept Payload

The format of the Accept Payload is shown in table A.8.

Table A.8 – LPC Accept Payload

Item	Size Bytes
hex ‘02000000’	4
Reserved	3
Status	1

Status: The format of the Status field is shown in table A.9.

Table A.9 – LPC Status

State	Value
Reserved	0
Success - The requested function was completed.	1
Failure - The requested function could not be completed.	2
Reserved	3 - 255

A.1.4 Loop Status (LSTS)

A Loop Status Extended Link Service requests the Fabric Controller to report on the state of the specified Loop.

Protocol:

Loop Status request Sequence
Reply Sequence

Format: FT–1

Addressing: The S_ID is the N_Port or NL_Port requesting status for the specified loop. The D_ID field shall be the Loop Fabric Address (LFA) of the loop for which status is being requested.

Payload: The format of the LSTS Payload is shown in table A.10.

Table A.10 – LSTS Payload

Item	Size Bytes
hex '72000000'	4

Reply Extended Link Service Sequence:

Service Reject (LS_RJT)
Signifies the rejection of the LSTS command
Accept (ACC)
Supplies the requested status for the identified Loop.
– Accept Payload

The format of the Accept Payload is shown in table A.11.

Table A.11 – LSTS Accept Payload

Item	Size Bytes
hex '02000000'	4
Reserved	1
Failed Receiver	1
FC-FLA Compliance Level	1
Loop State	1
Current Public Loop Devices bit map	16
Current Private Loop Devices bit map	16
AL_PA position map	128

Failed Receiver: The AL_PA of the L_Port that detected the loop failure. This field is valid only if the Loop State indicates a Loop Failure. This field shall be set to hex ’00’ if the failure was detected by the FL_Port. This field shall be set to hex ’F7’ if the detector of the failure cannot be determined.

NOTE – If the failure is not detected by the FL_Port, this field will typically be set to the “x” value of a received LIP(hex ’F8’, x).

FC-FLA Compliance Level: The Compliance Level of the loop, as detected by the FL_Port. The format of the FC-FLA Compliance Level field is shown in table A.12. See 4.7 for the meaning of the various compliance levels.

Table A.12 – FC-FLA Compliance Level

Level	Value
Reserved	0
FC-FLA Level A	1
FC-FLA Level B	2
Reserved	3 - 255

Loop State: The format of the Loop State field is shown in table A.13.

Table A.13 – Loop State

State	Value
Reserved	0
Online - The loop is not performing Loop Initialization, and no Failure has been detected.	1
Loop Failure - A Loop Failure has been detected. The AL_PA that detected the failure is reported in the Failed Receiver field.	2
Initialization Failure - The FL_Port has not been able to complete Loop Initialization.	3
Initializing - The loop is currently performing Loop Initialization.	4
Reserved	5 - 255

Current Public Loop Devices: The format of this field follows the AL_PA bit mapped format defined in FC–AL (see FC–AL table 15). If a bit is set to one in this field, then a Loop Device that has performed FLOGI is present at the identified location. This field is valid only when the value in the Loop State field is Online.

Current Private Loop Devices: The format of this field follows the AL_PA bit mapped format defined in FC–AL (see FC–AL table 15). If a bit is set to one in this field, then a Loop Device is present at the identified location. This field is valid only when the value in the Loop State field is Online.

AL_PA position map: The format of this field follows the AL_PA position map format defined in FC-AL for the LILP Sequence. If byte 0 of word 0 has a value of zero, then no AL_PA position map is available. This field is valid only when the value in the Loop State field is Online.

A.1.5 Registered State Change Notification (RSCN)

A Registered State Change Notification (RSCN) Extended Link Service request shall be sent to registered N_Ports and NL_Ports when an event occurs which may have affected the state of an N_Port or NL_Port, or the ULP state within the N_Port or NL_Port. The term “state” is used here to refer to any condition of an N_Port or NL_Port that is considered important enough to notify other N_Ports and NL_Ports of a change in that state. The definition of “important” is specific to an N_Port or NL_Port implementation, but can include the Login state or Link state.

NOTE – The State Change Notification (SCN) ELS defined in FC-PH-2 may be used to alert N_Ports and NL_Ports to state changes in Remote Ports. The FC-PH-2 definition does not adequately define the process by which a device registers to receive SCNs from the fabric, and does not explicitly address loop state changes. This text defines a registered and filtered version of SCN that provides for positive registration and de-registration. The function of this ELS is similar to the original SCN.

RSCN is intended to provide a timely indication of changes in nodes to avoid the considerable traffic that polling can generate. RSCN may be used to detect a failed node and release resources tied up by the failed node. RSCN may also be used to notify interested nodes of new devices coming online, and of changes within an online node that affect the operation of the system (e.g., more storage has become available). The sender of the RSCN request may coalesce several events into a single report.

The normal use of an RSCN request is for the fabric to notify registered N_Ports and NL_Ports of changes detected by the fabric. The RSCN request is issued by the Fabric Controller (hex ’FFFFFD’) to the registered N_Ports and NL_Ports. An RSCN request may also be issued by an N_Port or NL_Port to the Fabric Controller to indicate changes of state within the N_Port or NL_Port that are not otherwise detectable by the fabric. In either case, the payload of an RSCN request includes a list containing the addresses of the Affected N_Ports and NL_Ports. Note that while it is not explicitly prohibited, the sending of RSCN between two N_Ports or NL_Ports, neither of which are the Fabric Controller, is outside the scope of this document.

A.1.5.1 RSCN issued by the Fabric Controller

The Fabric Controller shall issue an RSCN request to all registered N_Ports and NL_Ports for an Affected N_Port or NL_Port when an event is detected by the fabric. The Fabric Controller shall ensure that any Fabric-provided resources (such as the Name Service) have received updates to reflect changes caused by the event, prior to issuing the RSCN for the event. An “event” may include any of the following:

- an implicit fabric Logout of the Affected N_Port or NL_Port, including loss of signal, NOS, and OLS, or when the fabric receives a FLOGI from a port that had already completed FLOGI;
- a loop initialization of the Affected NL_Port, and the L_bit was set in the LISA Sequence;
- a fabric Login from an Affected N_Port or NL_Port not previously logged in;
- the fabric path between the Affected N_Port or NL_Port and any other N_Port or NL_Port has

changed (such as: a change to the fabric routing tables that affects the ability of the fabric to deliver frames in order, or an E_Port initialization or failure);

- any other fabric-detected state change of the Affected N_Port or NL_Port;
- an RSCN request is issued by the Affected N_Port or NL_Port to the Fabric Controller.

A registered N_Port or NL_Port that receives an RSCN request may perform any operation it deems necessary to determine the nature of the state change. These operations include the PDISC ELS, the ADISC ELS, a query to the Name Service, or a ULP query. The fabric may accumulate Affected N_Port or NL_Port addresses for subsequent delivery to reduce the volume of RSCN traffic.

A.1.5.2 RSCN issued by the Affected N_Port or NL_Port

An N_Port or NL_Port shall issue an RSCN request to the Fabric Controller when an event is detected by the N_Port or NL_Port. An “event” may include any of the following:

- a failure within the Affected N_Port or NL_Port;
- any other important state change of the Affected N_Port or NL_Port.

A.1.5.3 RSCN initiative

An Affected N_Port or NL_Port shall issue one RSCN request for all state changes that occur prior to the initiation of the RSCN request; multiple RSCN requests shall not be queued for initiation. An RSCN request shall be considered initiated when the SOF of the first frame of the RSCN request Sequence has been transmitted.

A.1.5.4 RSCN registration

The Fabric Controller shall only issue RSCN requests to N_Ports and NL_Ports that have registered to be notified of state changes in other N_Ports and NL_Ports. This registration shall be performed via the State Change Registration (SCR) Extended Link Service. An N_Port or NL_Port may issue an RSCN to the Fabric Controller without having completed SCR with the Fabric Controller.

Protocol:

Registered State Change Notification (RSCN) request Sequence
Accept (ACC) Reply Sequence

Format: FT–1

Addressing: In the first case (the fabric notifies a registered N_Port or NL_Port), the S_ID is the Fabric Controller, hex ’FFFFFD; the D_ID is the address of the registered N_Port or NL_Port destination. In the second case (an Affected N_Port or NL_Port notifies the fabric of a change), the S_ID designates either the N_Port or NL_Port indicating a state change to the Fabric Controller; the D_ID is the Fabric Controller, hex ’FFFFFD.

Payload: The format of the RSCN request Payload is shown in table A.14.

Table A.14 – RSCN Payload

Item	Size Bytes
hex '61' = RSCN Command code	1
hex '04' = Page Length	1
Payload Length	2
Affected N_Port ID pages	4-max

Page Length: The length in bytes of an Affected N_Port ID page. This value is fixed at hex '04'.

Payload Length: The length in bytes of the entire payload, inclusive of the word 0. This value shall be a multiple of 4. The minimum value of this field is 4. The maximum value of this field is 256.

Affected N_Port ID page: Each Affected N_Port ID page contains the ID of an Affected N_Port or NL_Port. The RSCN payload may contain zero or more of these pages. The generic format of the Affected N_Port ID page is shown in table A.15.

Table A.15 – Generic Affected N_Port ID page

Item	Size Bytes
Address Format	1
Affected N_Port ID	3

Address Format: The format of the Address Format field is shown in table A.16.

Table A.16 – Address Format

Format	Value
Port Address - Bytes 1, 2, and 3 of the Affected N_Port ID are valid, and indicate a single N_Port or NL_Port address. This format is illustrated in table A.17.	0
Area Address Group - Bytes 1 and 2 of the Affected N_Port ID are valid, and indicates a group of addresses that encompass an Area of N_Port or NL_Port addresses. Byte 3 shall be zero. This format is illustrated in table A.18.	1
Domain Address Group - Byte 1 of the Affected N_Port ID is valid, and indicates a group of addresses that encompass a Domain of N_Port and/or NL_Port addresses. Bytes 2 and 3 shall be zero. This format is illustrated in table A.19.	2
Reserved	3 - 255

Affected N_Port ID: As defined by the Address Format. If the Address Format indicates an Area or Domain Address Group, all N_Port or NL_Port IDs within the indicated Group are Affected. The following three tables detail the various address formats.

Table A.17 – Port Address Format - Affected N_Port ID page

Item	Size Bytes
Address Format = hex ’00’	1
Affected N_Port ID	3

Table A.18 – Area Address Format - Affected N_Port ID page

Item	Size Bytes
Address Format = hex '01'	1
Affected N_Port Domain	1
Affected N_Port Area	1
Reserved = hex '00'	1

Table A.19 – Domain Address Format - Affected N_Port ID page

Item	Size Bytes
Address Format = hex '02'	1
Affected N_Port Domain	1
Reserved = hex '00'	2

Reply Extended Link Service Sequence:

Service Reject (LS_RJT)
Signifies the rejection of the RSCN command
Accept (ACC)
Signifies acceptance of the RSCN request.
– Accept Payload

The format of the Accept Payload is shown in table A.20.

Table A.20 – RSCN Accept Payload

Item	Size Bytes
hex '02000000'	4

A.1.6 State Change Registration (SCR)

The State Change Registration (SCR) Extended Link Service requests the Fabric Controller to add the N_Port or NL_Port to the list of N_Ports and NL_Ports registered to receive the Registered State Change Notification (RSCN) Extended Link Service.

Protocol:

State Change Registration request Sequence
Accept (ACC) Reply Sequence

Format: FT–1

Addressing: The S_ID designates the N_Port or NL_Port requesting registration for State Change Notification. The D_ID field is the Fabric Controller, hex‘FFFFFD‘.

Payload: The format of the SCR request Payload is shown in table A.21.

Table A.21 – SCR Payload

Item	Size Bytes
hex ‘62000000’	4
Reserved	3
Registration Function	1

Registration Function: The format of the Registration Function field is shown in table A.22.

Table A.22 – Registration Function

Function	Value
Reserved	0
Fabric Detected registration - Register to receive all RSCN requests issued by the Fabric Controller for events detected by the fabric.	1
N_Port Detected registration - Register to receive all RSCN requests issued by the Fabric Controller for events detected by the Affected N_Port or NL_Port.	2
Full registration - Register to receive all RSCN requests issued by the Fabric Controller. The RSCN request shall return all Affected N_Port ID pages.	3
Reserved	4 - 254
Clear registration - Remove any current RSCN registrations.	255

Reply Extended Link Service Sequence:

Service Reject (LS_RJT)
Signifies the rejection of the SCR command
Accept (ACC)
Signifies acceptance of the SCR request and registration for RSCN.
– Accept Payload

The format of the Accept Payload is shown in table A.23.

Table A.23 – SCR Accept Payload

Item	Size Bytes
hex ‘02000000’	4

Annex B
(normative)

ULP Operation by FC-FLA Devices

B.1 Introduction

This annex defines SCSI-FCP and IP behavior for FC-FLA compliant devices.

B.2 SCSI-FCP as an FC-FLA ULP

The following defines the mapping of SCSI-FCP features as a ULP for an FC-FLA device. SCSI-FCP operation is well-defined by FC-PLDA, and shall therefore be the basis of SCSI-FCP operation by FC-FLA devices.

B.2.1 How FC-FLA Supersedes FC-PLDA

In general, FC-FLA features supersede similar FC-PLDA features. The following table enumerates the features of FC-FLA that shall supersede FC-PLDA. FC-PLDA clauses (e.g., 5.8) which are introductory in nature are not listed. If the "Superseding FC-FLA Clause" is listed as a number in brackets (e.g., "[1]"), then the corresponding note at the end of the table defines the behavior. If the "Superseding FC-FLA Clause" is listed as "none", then the corresponding FC-PLDA clause shall be used as is.

Table B.1 – FC-PLDA Superseded by FC-FLA

FC-PLDA Clause	Superseding FC-FLA Clause
1	1
2	2
3	3
4	4
5.1	5.6
5.2	5.2.1, [1]
5.3	5.2.3
5.4	5.3
5.5	5.3
5.6	5.7
5.7	5.6
5.8.1	none
5.8.2	none
5.8.3.1	[2]
5.8.3.2	[2], [3]
5.8.4	none
6.1	5.5

Table B.1 – FC-PLDA Superseded by FC-FLA (continued)

FC-PLDA Clause	Superseding FC-FLA Clause
6.2.1	[4]
6.2.2	[4]
6.2.3	[5]
6.3.1	5.5
6.3.2	5.5
6.3.3	5.5
6.3.4	5.6
7	8
7.1	8.1
7.2	8.2
7.3	8.3
7.4	8.4
7.5	8.5
8.1	none
8.2	none
8.2.1	none
8.2.2	none
8.2.3	none
8.2.3.1	none
8.2.3.2	none
8.2.4	none
8.2.4.1	none
8.2.4.2	none
8.3	none
8.4	none
8.5	none
8.6	[6]
9.1	none
9.2	[7]
9.3.1	[7]
9.3.2	[2]
9.3.3	none
9.3.4	none
9.3.5	none
9.3.6	none

Table B.1 – FC-PLDA Superseded by FC-FLA (continued)

FC-PLDA Clause	Superseding FC-FLA Clause
9.3.7	none
9.4	[7]
9.5	none
9.6	[7]
9.7	5.6
10.1	none
10.2	none
10.3	[8]
10.4	none
10.4.1	5.6
10.4.2	5.6
10.5	none
11.1	[9]
11.2	none
11.3	none
11.4	none
11.4.1	none

Table B.1 – FC-PLDA Superseded by FC-FLA (continued)

FC-PLDA Clause	Superseding FC-FLA Clause
12.1	[9]
12.2	none
12.3	none
12.4	none

Notes:

1 The FC-FLA clause supersedes, with the exception that the Random Relative Offset parameter shall be set as defined in FC-PLDA. The Random Relative Offset setting for devices that support ULPs other than FC-PLDA, or that support more than one ULP, is not defined by this document

2 The clause is used as defined, with the exception that R_A_TOV, as defined in 8.3 of this document, shall instead apply.

3 The clause is used as defined, with the exception that normal FC-PH rules for Streamed Sequences shall instead apply, except as follows: If the communicating ports are operating on an in-order fabric in Class 3, a Sequence may be considered to be closed either when the Sequence following changes Information Category; or, when the Sequence ends with a transfer of Sequence Initiative, and the Sequence Initiator receives a frame for the Sequence following in the same Exchange from the Sequence Recipient.

4 The clause is used as defined, with the exception that normal FC-AL rules for Alternate BB_Credit Management shall instead apply.

5 Annex D of this document shall instead apply. (This clause of FC-PLDA is not considered required for interoperability.)

6 The clause is used as defined for Class 3 operation. For Class 2 operation, the normal FC-PH rules for Sequence delivery shall instead apply.

7 The clause is used as defined, with the exception that a SCSI Target is Allowed to invoke the ABTS protocol.

8 The clause is used as defined, with the exception that discovery of SCSI Targets on a Remote Loop (or a Remote N_Port) may be performed via the Simple Name Server, RSCN, or other means. Also, the function performed by ADISC/PDISC in FC-PLDA is performed by FAN in FC-FLA.

9 The clause is used as defined, with the exception that Class 2 may be used by FC-FLA compliant devices, if implemented.

B.2.2 Summary of Additional Features for FC-FLA SCSI Operation

The following summarizes the features that must be added to all FC-PLDA-compliant devices to bring them up to FC-FLA Class 3 operation. This summary is provided as a convenience; any conflicts with the list in B.2.1 shall be resolved in favor of B.2.1.

B.2.2.1 Additional Features for Both SCSI Initiators and SCSI Targets

The following features must be added to all FC-PLDA-compliant devices.

a) The ability to open and send the FLOGI request to the L_Port at AL_PA = x'00', and to be opened and receive the FLOGI ACC from the L_Port at AL_PA = x'00'.

b) The ability to authenticate Exchanges (and detect changes to its own native address identifier) following the FC-AL Loop Initialization Protocol by receiving the FAN ELS from the L_Port at AL_PA = x'00', instead of using PDISC/ADISC.

c) The ability to claim a fabric-assigned AL_PA during the FC-AL Loop Initialization Protocol by setting the corresponding bit in the LIFA Sequence.

d) The ability to register the FC-4 Object with the Simple Name Server.

e) The ability to open the L_Port at AL_PA = x'00' and send frames to one or more Remote Ports before closing.

f) The ability to be opened by the L_Port at AL_PA = x'00' and receive frames from one or more Remote Ports before closing.

g) Some timer values and usages are different between FC-PLDA and FC-FLA.

B.2.2.2 Additional Features for SCSI Initiators

The following additional features must be added to all FC-PLDA-compliant SCSI Initiators.

a) The ability to properly accept and respond to the ABTS BLS.

B.2.2.3 Additional Features for SCSI Targets

No additional features must be added to FC-PLDA-compliant SCSI Targets.

B.3 IP as an FC-FLA ULP

There is an active profile effort to develop an IP profile that will operate on FC-FLA compliant devices. FC-FLA provides tools that may be used by any protocol, including IP. The IP profile will define the use of these tools for IP.

Annex C
(informative)

Examples of Public Loop Device Behavior

C.1 Introduction

Public Loop Devices can make use of several behaviors that are either new, or perhaps not previously used by mainstream Fibre Channel devices. This annex provides examples to illustrate these behaviors.

C.2 Use of Registered State Change Notification to Assist I/O Probing in Switched Mass Storage Topologies

This clause describes the rationale for the various new tools described within this document. It also gives an example description of a flow that may be used by hosts and storage devices for discovery.

C.2.1 Background Information

Existing mass storage environments rely on a technique commonly referred to as “I/O probing” to determine the presence of devices on the SCSI bus. In this environment, the host will initiate a ‘probe’ command to a specific address to determine if a device exists at the probed address; typically a timeout will occur if there is no device at the probed address. I/O probing of the mass storage link enables the host to build or validate a higher level SCSI LUN map. Examples of when I/O probing is performed include:

- discovery of the mass storage configuration at initialization;
- update of the mass storage configuration map after a failure is detected;
- an operator requests a configuration update;
- periodic validation of the host map of the mass storage configuration.

Although this probing technique can be used in private loop (FC-PLDA) environments, it becomes untenable for the potentially large address space supported by switched mass storage topologies with multiple hosts. For example, it could take several hours for a single host to complete a probe of the entire 2^{24} address space. Therefore, new FC tools are needed to assist host I/O probing algorithms in switched mass storage topologies, which may include public loops. The goals of these tools include:

- significantly reduce the time to determine and update the host mass storage configuration map;
- minimize the impact to implementations of host I/O drivers. switch fabrics, and storage devices;
- ensure the solution is scalable to large, multi-switch topologies.

C.2.2 Specific Requirements for a Solution

The minimum set of functionality provided by these new tools includes:

– a gathering mechanism to determine all of the currently active addresses on the Fibre Channel fabric, where “active” refers to nodes that are known to have completed FLOGI with the fabric;

– an update mechanism that notifies a host whenever a change in the currently active fabric address space is detected;

– a registration/de-registration mechanism to allow only interested hosts to receive notifications (storage devices generally will not register for this service);

– a device-initiated notification to allow peripheral controllers to notify the host of an internal configuration change or failure that affects the SCSI LUN map but not the active FC address map.

C.2.3 Additional Tools Required by System Components

The additional protocol functionality required are shown below by system component.

– Switched fabric requirements:

  – extended format FLOGI, as defined in FC-PH-3;

  – a minimal subset of the Name Service that is able to provide a list of nodes that have completed FLOGI, as defined in FC-GS-2;

  – a Fabric Controller that support RSCN and SCR protocols, as defined in this document.

– Host mass storage driver:

  – extended format FLOGI, as defined in FC-PH-3;

  – registration and de-registration for RSCN with the Fabric Controller using SCR ELS, as defined in this document;

  – reception and processing of RSCN ELS notifications from the Fabric Controller, as defined in this document;

  – ability to perform a Get Native Port Identifiers (GP_ID4) with the Name Service, as defined in FC-GS-2;

  – ability to perform a Register FC-4 Types (RFC-4) with the Name Service, as defined in FC-GS-2.

– Peripheral storage device:

  – extended format FLOGI, as defined in FC-PH-3;

  – ability to perform a Register FC-4 Types (RFC-4) with the Name Service, as defined in FC-GS-2;

  – optionally, to initiate RSCN to the Fabric Controller, as defined in this document.

C.2.4 Example Usage of RSCN and SCR ELS

The following is simple example of how the host mass storage driver and storage device would use these new FC tools implement the I/O probing technique. The example assumes the host is directly attached to a switch and the storage devices are attached to the switch via public loops; note, however, that the protocol is topology independent.

Host mass storage driver:

a) The host completes link initialization and detects that it is directly attached to an F-Port.

b) The host completes the extended format F_LOGI with well-known address hex 'FFFFFE' (the Fabric Login Server), and discovers that the Name Service exists.

c) The host attempts to complete Name Service registration duties as required by this document.

d) The host completes the SCR protocol with well-known address hex 'FFFFFD' (the Fabric Controller) to register for state change notification.

NOTE – The host may receive an RSCN ELS any time after step (d). Recovery action by the host is not specified but may include a PDISC of the address(es) indicated in the RSCN payload.

e) The host completes the Get Native Port Identifiers (GP_ID4) with the Name Service at well-known address hex 'FFFFFC' (the Directory Service Facilitator) to get the list of all active Native Port addresses.

f) The host attempts PLOGI with active addresses, and discovers which addresses are SCSI-FCP Targets via the usual methods described in FC-PLDA (see Annex B).

g) The host builds an initial mass storage LUN configuration image.

NOTE – The host may choose to un-subscribe from notification at any time by completing an SCR deregistration with the Fabric Controller

Public Loop device:

a) The device completes link initialization and acquires an AL_PA.

b) The device completes extended format FLOGI with well-known address hex 'FFFFFE' (the Fabric Login Server), determines its full 24 bit Native Port address, and discovers the presence of a server at 0xFFFFFC (Directory Service Facilitator).

c) The device attempts to complete Name Service registration duties as required by this document.

NOTE – An intelligent mass storage device, such as a RAID controller, may choose to send a RSCN ELS to well-known address hex 'FFFFFD' (the Fabric Controller) to signify a failure or internal LUN configuration change to interested hosts.

C.3 Name Service Usage

FC-FLA provides for the availability of a Name Service as an alternate method of performing discovery of available N_Ports and NL_Ports. This clause shows examples of how the Name Service builds its database and services requests. For purposes of these examples, a simple Fabric with one entity within the Fabric performing Name Service duties is shown. One FL_Port with 3 attached Public NL_Ports and 1 Private NL_Port is shown, along with 2 other N_Ports attached to F_Ports. All numbers shown are in hexadecimal format to keep table clutter down. All request and response Sequences are "idealized" for clarity; specific payload examples are shown in C.3.5.

C.3.1 Initial Condition of Name Service Following First Loop Initialization

The following table shows the initial state of the Name Service database for this example, immediately following Loop Initialization and Link Initialization, but before any frames have been transmitted by any port (other than Loop Initialization Sequences). Note that while the NL_Ports know their AL_PAs, none of the ports know their full ID.

Table C.1 – Name Service - Initial State

Object	NL_Port A	NL_Port B	NL_Port C	NL_Port D	N_Port A	N_Port B
Native ID	null	null	null	null	null	null
Port_Name	null	null	null	null	null	null
Node_Name	null	null	null	null	null	null
Class of Service	null	null	null	null	null	null
FC-4 Types	null	null	null	null	null	null
Port_Type	NL_Port	NL_Port	NL_Port	NL_Port	N_Port	N_Port

After initialization, the Port Type is known, since the Fabric knew what kind of initialization was performed and completed successfully; Loop, with the LIP protocol and the various Loop Initialization Sequences, or Link, with the OLS/LR/LRR protocol. If the port could not be initialized, the Port Type would be set to “null”. The Native ID object of all N_Ports and NL_Ports are set to “null” since the ID has not yet been assigned via FLOGI. All other objects are also set to “null”.

C.3.2 Implicit Registration of N_Ports and NL_Ports from FLOGI Parameters

The following table shows the Name Service database after all N_Ports and Public NL_Ports have completed FLOGI.

Table C.2 – Name Service - Implicit FLOGI Registration

Object	NL_Port A	NL_Port B	NL_Port C	NL_Port D	N_Port A	N_Port B
Native ID	010101	010102	010104	null	010201	010301
Port_Name	PN_A	PN_B	PN_C	null	PN_E	PN_F
Node_Name	NN_A	NN_B	NN_C	null	NN_E	NN_F
Class of Service	2,3	3	3	null	2,3	2,3
FC-4 Types	null	null	null	null	null	null
Port_Type	NL_Port	NL_Port	NL_Port	NL_Port	N_Port	N_Port

FLOGI contains or establishes most of the information required by the Name Service to create and maintain the database. The Native ID is assigned by the Fabric and returned to the N_Ports and NL_Ports in the FLOGI Accept payload. In this example, all N_Ports and NL_Ports are given a Domain_ID = hex ’01’. The Public NL_Ports share a common Area_ID = hex ’01’; the N_Ports each are assigned there own Area_ID in this example. The N_Ports are assigned a Port_ID of hex ’01’, while the NL_Ports receive a Port_ID equal to their established AL_PA. The Port_Name and

Node_Name are taken directly from the FLOGI request payload, as are the supported Classes of Service. The following table illustrates the FLOGI request payload sent by NL_Port “A”. Those items that are passed to the Name Service are indicated in **boldface**.

Table C.3 – Example FLOGI Request Payload, NL_Port “A”

Words	Contents
0-3	Common Service Parameters
5-6	**Port_Name = “PN_A”**
7-8	**Node_Name = “NN_A”**
9-12	Class 1 Service Parameters
13-16	Class 2 Service Parameters (**Class 2 valid bit = 1**)
17-20	Class 3 Service Parameters (**Class 3 valid bit = 1**)

Note also that NL_Port “D” has not changed because FLOGI has not yet been attempted by that port. Since it is a Private NL_Port, it never will attempt FLOGI, and so the Name Service entry does not change. Since the Name Service has no knowledge as to the Private or Public nature of an NL_Port, it retains this entry in its database in anticipation of a future FLOGI.

C.3.3 Explicit Registration of FC-4 Types

The following table shows the Name Service database after all N_Ports and Public NL_Ports have completed registration of FC-4 types via the Name Service RFC-4 request.

Table C.4 – Name Service - Explicit FC-4 Registration

Object	NL_Port A	NL_Port B	NL_Port C	NL_Port D	N_Port A	N_Port B
Native ID	010101	010102	010104	null	010201	010301
Port_Name	PN_A	PN_B	PN_C	null	PN_E	PN_F
Node_Name	NN_A	NN_B	NN_C	null	NN_E	NN_F
Class of Service	2,3	3	3	null	2,3	2,3
FC-4 Types	SCSI, IP	SCSI	SCSI	null	SCSI	SCSI, IP
Port_Type	NL_Port	NL_Port	NL_Port	NL_Port	N_Port	N_Port

The RFC-4 request payload contains an 8-word bit map which indicates which TYPE codes are supported; each bit corresponds to a standard FC-PH TYPE code value. For example, an N_Port or NL_Port which supported only SCSI-FCP would set bit 8 of the first word (corresponding to TYPE code hex ’08’), hex ‘00000100’, and the other 7 words to zero. The following table illustrates the RFC-4 request payload sent by N_Port “B” to indicate support for both SCSI-FCP and IP.

Table C.5 – Example RFC-4 Request Payload, N_Port “B”

<table>
<tr><th>Word(s)</th><th colspan="3">Contents</th></tr>
<tr><td>0</td><td>FC_CT rev = hex ’01’</td><td colspan="2">zero</td></tr>
<tr><td>1</td><td colspan="3">FC_CT Name Service codes = hex ’FC020000’</td></tr>
<tr><td>2</td><td colspan="2">Command Code RFC-4 = hex ’0217’</td><td>zero</td></tr>
<tr><td>3</td><td colspan="3">zero</td></tr>
<tr><td>4</td><td>zero</td><td colspan="2">ID of N_Port “B” = hex ’010301’</td></tr>
<tr><td>5</td><td colspan="3">SCSI-FCP & IP supported = hex ’00000120’</td></tr>
<tr><td>6-12</td><td colspan="3">zero</td></tr>
</table>

The RFC-4 response payload is a simple Accept response with no additional content.

Table C.6 – Example RFC-4 Request Payload, N_Port “B”

<table>
<tr><th>Word(s)</th><th colspan="3">Contents</th></tr>
<tr><td>0</td><td>FC_CT rev = hex ’01’</td><td colspan="2">zero</td></tr>
<tr><td>1</td><td colspan="3">FC_CT Name Service codes = hex ’FC020000’</td></tr>
<tr><td>2</td><td colspan="2">Command Code FS_ACC = hex ’8002’</td><td>zero</td></tr>
<tr><td>3</td><td colspan="3">no error = hex ’00000000’</td></tr>
</table>

A detailed example of the RFC-4 format is shown in C.3.5.1.

C.3.4 Querying the Name Service

An N_Port or NL_Port may query the Name Service at any time to request information from the data-base. The Name Service will always make its best attempt to return the most current information.

As described in C.2, one common use of Name Service queries is to simply request a list of all ports that have completed FLOGI. This is done by issuing the Get Native Port Identifiers for Port_Type (GP_ID4) request to the Name Service. The following table illustrates the GP_ID4 request payload sent by N_Port “A” to get a list of all logged in N_Ports and NL_Ports. The “Port_Type” field of the request is set to hex ’7F’, which requests all types of Nx_Ports.

Table C.7 – Example GP_ID4 Request Payload, Nx_Ports

Word(s)	Contents		
0	FC_CT rev = hex ’01’	zero	
1	FC_CT Name Service codes = hex ’FC020000’		
2	Command Code GP_ID4 = hex ’01A1’		zero
3	zero		
4	Port_Type = hex ’7F’	zero	

The following table illustrates the GP_ID4 response payload sent to N_Port “A”.

Table C.8 – Example GP_ID4 Response Payload, Nx_Ports

Word(s)	Contents		
0	FC_CT rev = hex ’01’	zero	
1	FC_CT Name Service codes = hex ’FC020000’		
2	Command Code FS_ACC = hex ’8002’		zero
3	no error = hex ’00000000’		
4	hex ’00’	hex ’010101’	
5	hex ’00’	hex ’010102’	
6	hex ’00’	hex ’010104’	
7	hex ’00’	hex ’010201’	
8	hex ’80’	hex ’010301’	

The list contains the Native IDs of all N_Ports and NL_Ports that have completed FLOGI and have therefore been assigned a Native ID. NL_Port “D” is not returned since it has no Native ID assigned. The first byte of the last entry on the list is set to hex ’80’.

To get a list of all Loop Fabric Addresses, just ask for a different Port_Type. The following table illustrates the GP_ID4 request payload sent by N_Port "A" to get a list of all LFAs. The "Port_Type" field of the request is set to hex '7F', which requests all types of Nx_Ports.

Table C.9 – Example GP_ID4 Request Payload, LFAs

Word(s)	Contents		
0	FC_CT rev = hex '01'	zero	
1	FC_CT Name Service codes = hex 'FC020000'		
2	Command Code GP_ID4 = hex '01A1'		zero
3	zero		
4	Port_Type = hex '82'	zero	

The following table illustrates the GP_ID4 response payload sent to N_Port "A".

Table C.10 – Example GP_ID4 Response Payload, LFAs

Word(s)	Contents		
0	FC_CT rev = hex '01'	zero	
1	FC_CT Name Service codes = hex 'FC020000'		
2	Command Code FS_ACC = hex '8002'		zero
3	no error = hex '00000000'		
4	hex '80'	hex '010100'	

The list contains the LFA of the one loop attached to the Fabric. Since it is the only entry on the list, the first byte is set to hex '80'.

Once the N_Port or NL_Port has received a list of logged in N_Ports and NL_Ports, it can continue to query the Name Service for more information. For example, it may request the supported FC-4 types.

The following table illustrates the Get FC-4 Types (GFC-4) request payload sent by N_Port “A” to get a list of all FC-4 types supported by NL_Port “B”.

Table C.11 – Example GFC-4 Request Payload

Word(s)	Contents	
0	FC_CT rev = hex ’01’	zero
1	FC_CT Name Service codes = hex ’FC020000’	
2	Command Code GFC-4 = hex ’0117’	zero
3	zero	
4	zero	NL_Port “B” ID = hex ’010102’

The following table illustrates the GFC-4 response payload sent to N_Port “A”.

Table C.12 – Example GFC-4 Response Payload

Word(s)	Contents	
0	FC_CT rev = hex ’01’	zero
1	FC_CT Name Service codes = hex ’FC020000’	
2	Command Code FS_ACC = hex ’8002’	zero
3	no error = hex ’00000000’	
4	SCSI-FCP supported = hex ’00000100’	
5-11	zero	

The list contains the same bit map format as was sent by NL_Port “B” when it performed the RFC-4 request. This example also illustrates that the Name Service is a simple store-and-recall process. It performs no complex operations; it simply stores what it is asked to store and retrieves what it is asked to retrieve.

C.3.5 Name Service Request Sequences

This clause shows examples of Name Service request and response frames (Sequences) as they would be constructed by a device that complies with this report.

C.3.5.1 Name Service Register FC-4 Type (RFC-4)

The Register FC-4 Type (RFC-4) Common FC Service requests the Name Service to record which FC-4 types are supported by the N_Port or NL_Port. The requirements for this service are defined in FC-GS-2. The intent here is to show the specific Frame contents for this Name Service request as used by an FC-FLA compliant device.

The following table shows the format of a single Frame Sequence used by an FC-FLA compliant N_Port or NL_Port to perform the RFC-4 request.

Table C.13 – RFC-4 Request Header and Payload

Word	Byte 0 Bits 31-24	Byte 1 Bits 23-16	Byte 2 Bits 15-8	Byte 3 Bits 7-0
0	hex '02' (R_CTL: FC-4 Device Data, Unsolicited Control)	hex 'FFFFFC' (D_ID: Directory Server)		
1	hex '00' (CS_CTL)	Port ID of requesting N_Port or NL_Port (S_ID)		
2	hex '20' (TYPE: Fibre Channel Services)	hex '290000' (F_CTL: First Sequence of Exchange, Last Frame of Sequence, Transfer Sequence Initiative)		
3	SEQ_ID	hex '00' (DF_CTL)	hex '0000' (SEQ_CNT)	
4	OX_ID		hex 'FFFF' (RX_ID)	
5	hex '00000000' (Parameter)			
6	hex '01' (FC-CT revision)	hex '000000' (IN_ID; used by servers)		
7	hex 'FC' (FCS_Type: Directory Service Application)	hex '02' (FCS_Subtype: Name Service)	hex '00' (Options: single Exchange)	hex '00' (Reserved)
8	hex '0217' (Command Code: RFC-4		hex '0000' (Reserved)	
9	hex '00' (Reserved)	hex '00' (Reason Code)	hex '00' (Explanation Code)	hex '00' (Vendor Unique)

Table C.13 – RFC-4 Request Header and Payload (continued)

Word	Byte 0 Bits 31-24	Byte 1 Bits 23-16	Byte 2 Bits 15-8	Byte 3 Bits 7-0
10	hex ’00’ (Reserved)	S_ID of sending N_Port or NL_Port		
11	hex ’00’	hex ’00’	hex ’01’ (SCSI-FCP bit, for example)	hex ’00’
12	hex ’00’	hex ’00’	hex ’00’	hex ’00’
13	hex ’00’	hex ’00’	hex ’00’	hex ’00’
14	hex ’00’	hex ’00’	hex ’00’	hex ’00’
15	hex ’00’	hex ’00’	hex ’00’	hex ’00’
16	hex ’00’	hex ’00’	hex ’00’	hex ’00’
17	hex ’00’	hex ’00’	hex ’00’	hex ’00’
18	hex ’00’	hex ’00’	hex ’00’	hex ’00’

Words 0-5 contain the frame header of a Fibre Channel Common Services request. The Name Service requires that all requests be made as a single new Exchange, and that the response be returned and the Exchange closed before the next request is issued.

Words 6-9 contain the FC_CT header for the CT_IU. The first byte of word 6 contains the FC_CT revision code; bits 23-0 are a reserved space that may be used to carry the S_ID of the original requestor between different servers. Word 7 indicates that the request is directed to the Name Service, and indicates via the Option byte that this service request will be completed before the next request is sent. Word 8 indicates the command to perform. Word 9 is used only for reject responses and is set to zero.

Words 10-18 contain the request payload. Word 10 contains the Native Port Identifier of the port registering its FC-4 types. Words 11-18 contains a bit map of supported FC-4 Types. Word 0, bit 0 indicates TYPE code hex ’00’, and so on. The map here shows the bit position for SCSI-FCP TYPE code hex ’08’, as an example only.

The following table shows the format of a single Frame Sequence used by an FC-FLA compliant fabric to respond to the RFC-4 request.

Table C.14 – RFC-4 Response Header and Payload

Word	Byte 0 Bits 31-24	Byte 1 Bits 23-16	Byte 2 Bits 15-8	Byte 3 Bits 7-0
0	hex '03' (R_CTL: FC-4 Device Data, Solicited Control)	Port ID of requesting N_Port or NL_Port (D_ID)		
1	hex '00' (CS_CTL)	hex 'FFFFFC' (S_ID: Directory Server)		
2	hex '20' (TYPE: Fibre Channel Services)	hex '980000' (F_CTL: Exchange Responder, Last Sequence of Exchange, Last Frame of Sequence)		
3	SEQ_ID	hex '00' (DF_CTL)	hex '0000' (SEQ_CNT)	
4	OX_ID		RX_ID	
5	hex '00000000' (Parameter)			
6	hex '01' (FC-CT revision)	hex 'XXXXXX' (IN_ID; used by servers; may have been used)		
7	hex 'FC' (FCS_Type: Directory Service Application)	hex '02' (FCS_Subtype: Name Service)	hex '00' (Options: single Exchange)	hex '00' (Reserved)
8	hex '8002' (Command Code: FS_ACC)		hex '0000' (Reserved)	
9	hex '00' (Reserved)	hex '00' (Reason Code)	hex '00' (Explanation Code)	hex '00' (Vendor Unique)

Words 0-5 contain the frame header of a Fibre Channel Common Services response. The Name Service requires that all requests be made as a single new Exchange, and that the response be returned and the Exchange closed before the next request is issued.

Words 6-9 contain the FC_CT header for the CT_IU. The first byte of word 6 contains the FC_CT revision code; bits 23-0 are a reserved space that may be used to carry the S_ID of the original requestor between different servers; this space may be non-zero if it was used. Word 7 indicates that the response is from the Name Service, and indicates via the Option byte that the service must be completed before the next request is sent. Word 8 indicates the Command Code of an FC_CT Accept (FS_ACC) response. Word 9 is zero since the response is an Accept response.

If the response is a reject (FS_RJT) instead of an Accept, then the Command Code in word 8 would be set to hex '8001', and the Reason Code and Explanation Code would be set accordingly.

C.3.5.2 Name Service Get Native Port Identifiers for Port Type (GP_ID4)

The Get Native Port Identifiers for Port Type (GP_ID4) Common FC Service requests the Name Service to return a list of Native Port Identifiers of the specified Port Type. The requirements for this service are defined in FC-GS-2. The intent here is to show the specific Frame contents for this Name Service.

The following table shows the format of a single Frame Sequence used by an FC-FLA compliant N_Port or NL_Port to perform the GP_ID4 request.

Table C.15 – GP_ID4 Request Header and Payload

Word	Byte 0 Bits 31-24	Byte 1 Bits 23-16	Byte 2 Bits 15-8	Byte 3 Bits 7-0
0	hex '02' (R_CTL: FC-4 Device Data, Unsolicited Control)	hex 'FFFFFC' (D_ID: Directory Server)		
1	hex '00' (CS_CTL)	Port ID of requesting N_Port or NL_Port (S_ID)		
2	hex '20' (TYPE: Fibre Channel Services)	hex '290000' (F_CTL: First Sequence of Exchange, Last Frame of Sequence, Transfer Sequence Initiative)		
3	SEQ_ID	hex '00' (DF_CTL)	SEQ_CNT	
4	OX_ID		RX_ID	
5	hex '00000000' (Parameter)			
6	hex '01' (FC-CT revision)	hex '000000' (IN_ID; used by servers)		
7	hex 'FC' (FCS_Type: Directory Service Application)	hex '02' (FCS_Subtype: Name Service)	hex '00' (Options: single Exchange)	hex '00' (Reserved)
8	hex '01A1' (Command Code: GP_ID4)		hex '0000' (Reserved)	
9	hex '00' (Reserved)	hex '00' (Reason Code)	hex '00' (Explanation Code)	hex '00' (Vendor Unique)
10	hex '7F' (Port Type code for all Nx_Ports)	hex '000000' (Reserved)		

Words 0-5 contain the frame header of a Fibre Channel Common Services request. The Name Service requires that all requests be made as a single new Exchange, and that the response be returned and the Exchange closed before the next request is issued.

Words 6-9 contain the FC_CT header for the CT_IU. The first byte of word 6 contains the FC_CT revision code; bits 23-0 are a reserved space that may be used to carry the S_ID of the original requestor between different servers. Word 7 indicates that the request is directed to the Name Service, and indicates via the Option byte that the service will be completed before the next request is sent. Word 8 indicates the command to perform. Word 9 is used only for reject responses and is set to zero.

Word 10 contains the Port Type code that will cause a list of all N_Ports and NL_Ports to be returned. The Name Service at least supports the Port Type code for Nx_Port. The Name Service may support other code values. A fabric that receives a request for an unimplemented Port Type code should return an FS_RJT with a Reason Code of “Command not supported” and an Explanation Code of “No additional explanation”.

NOTE – If a list of N_Ports only is desired, this code should be set to hex ’01’. If a list of NL_Ports only is desired, this code should be set to hex ’02’.

The following table shows the format of a single Frame Sequence used by an FC-FLA compliant fabric to respond to the GP_ID4 request.

Table C.16 – GP_ID4 Response Header and Payload

Word	Byte 0 Bits 31-24	Byte 1 Bits 23-16	Byte 2 Bits 15-8	Byte 3 Bits 7-0
0	hex ’03’ (R_CTL: FC-4 Device Data, Solicited Control)	Port ID of requesting N_Port or NL_Port (D_ID)		
1	hex ’00’ (CS_CTL)	hex ’FFFFFC’ (S_ID: Directory Server)		
2	hex ’20’ (TYPE: Fibre Channel Services)	hex ’980000’ (F_CTL: Exchange Responder, Last Sequence of Exchange, Last Frame of Sequence)		
3	SEQ_ID	hex ’00’ (DF_CTL)	SEQ_CNT	
4	OX_ID		RX_ID	
5	hex ’00000000’ (Parameter)			

Table C.16 – GP_ID4 Response Header and Payload (continued)

Word	Byte 0 Bits 31-24	Byte 1 Bits 23-16	Byte 2 Bits 15-8	Byte 3 Bits 7-0
6	hex '01' (FC-CT revision)	hex 'XXXXXX' (IN_ID; used by servers; may have been used)		
7	hex 'FC' (FCS_Type: Directory Service Application)	hex '02' (FCS_Subtype: Name Service)	hex '00' (Options: single Exchange)	hex '00' (Reserved)
8	hex '8002' (Command Code: FS_ACC)		hex '0000' (Reserved)	
9	hex '00' (Reserved)	hex '00' (Reason Code)	hex '00' (Explanation Code)	hex '00' (Vendor Unique)
10	hex '00' (Control: Not the last item)	Native Port Identifier #1 of N_Port or NL_Port		
11	hex '00' (Control: Not the last item)	Native Port Identifier #2 of N_Port or NL_Port		
12	hex '00' (Control: Not the last item)	Native Port Identifier #3 of N_Port or NL_Port		
. . .				
N+9	hex '80' (Control: This is the last item)	Native Port Identifier #N of N_Port or NL_Port		

Words 0-5 contain the frame header of a Fibre Channel Common Services response. The Name Service requires that all requests be made as a single new Exchange, and that the response be returned and the Exchange closed before the next request is issued.

Words 6-9 contain the FC_CT header for the CT_IU. The first byte of word 6 contains the FC_CT revision code; bits 23-0 are a reserved space that may be used to carry the S_ID of the original requestor between different servers; this space may be non-zero if it was used. Word 7 indicates that the response is from the Name Service, and indicates via the Option byte that the service must be completed before the next request is sent. Word 8 indicates the Command Code of an FC_CT Accept (FS_ACC) response. Word 9 is zero since the response is an Accept response.

Words 10 and beyond contain the list of N_Port and NL_Port Native Port Identifiers. Byte 0 of each word is set to hex '00' if this Identifier is not the last entry. Byte 0 of each word is set to hex '80' if this

Identifier is the last entry. Note that the GP_ID4 response Sequence may exceed the capacity of a single frame.

If the response is a reject (FS_RJT) instead of an Accept, then the Command Code in word 8 would be set to hex ’8001’, the list of Native Port Identifiers is not returned, and the Reason Code and Explanation Code would be set accordingly.

C.3.5.3 Name Service Get Port Type (GPT)

The Get Port Type (GPT) Common FC Service requests the Name Service to return the Port Type of the specified Native Port Identifier. The requirements for this service are defined in FC-GS-2. The intent here is to show the specific Frame contents for this Name Service.

The following table shows the format of a single Frame Sequence used by an FC-FLA compliant N_Port or NL_Port to perform the GPT request.

Table C.17 – GPT Request Header and Payload

Word	Byte 0 Bits 31-24	Byte 1 Bits 23-16	Byte 2 Bits 15-8	Byte 3 Bits 7-0
0	hex ’02’ (R_CTL: FC-4 Device Data, Unsolicited Control)	hex ’FFFFFC’ (D_ID: Directory Server)		
1	hex ’00’ (CS_CTL)	Port ID of requesting N_Port or NL_Port (S_ID)		
2	hex ’20’ (TYPE: Fibre Channel Services)	hex ’290000’ (F_CTL: First Sequence of Exchange, Last Frame of Sequence, Transfer Sequence Initiative)		
3	SEQ_ID	hex ’00’ (DF_CTL)	SEQ_CNT	
4	OX_ID		RX_ID	
5	hex ’00000000’ (Parameter)			

Table C.17 – GPT Request Header and Payload (continued)

Word	Byte 0 Bits 31-24	Byte 1 Bits 23-16	Byte 2 Bits 15-8	Byte 3 Bits 7-0
6	hex ’01’ (FC-CT revision)	hex ’000000’ (IN_ID; used by servers)		
7	hex ’FC’ (FCS_Type: Directory Service Application)	hex ’02’ (FCS_Subtype: Name Service)	hex ’00’ (Options: single Exchange)	hex ’00’ (Reserved)
8	hex ’011A’ (Command Code: GPT)		hex ’0000’ (Reserved)	
9	hex ’00’ (Reserved)	hex ’00’ (Reason Code)	hex ’00’ (Explanation Code)	hex ’00’ (Vendor Unique)
10	hex ’00’ (Reserved)	Native Port Identifier		

Words 0-5 contain the frame header of a Fibre Channel Common Services request. The Name Service requires that all requests be made as a single new Exchange, and that the response be returned and the Exchange closed before the next request is issued.

Words 6-9 contain the FC_CT header for the CT_IU. The first byte of word 6 contains the FC_CT revision code; bits 23-0 are a reserved space that may be used to carry the S_ID of the original requestor between different servers. Word 7 indicates that the request is directed to the Name Service, and indicates via the Option byte that the service will be completed before the next request is sent. Word 8 indicates the command to perform. Word 9 is used only for reject responses and is set to zero.

Word 10 contains the Native Port Identifier (Port ID) for which the Port Type is being requested. A fabric that receives a request for an invalid Native Port Identifier should return an FS_RJT with a Reason Code of “Unable to perform command request” and an Explanation Code of “Unacceptable Native Port Identifier”.

The following table shows the format of a single Frame Sequence used by an FC-FLA compliant fabric to respond to the GPT request.

Table C.18 – GPT Response Header and Payload

Word	Byte 0 Bits 31-24	Byte 1 Bits 23-16	Byte 2 Bits 15-8	Byte 3 Bits 7-0
0	hex ’03’ (R_CTL: FC-4 Device Data, Solicited Control)	Port ID of requesting N_Port or NL_Port (D_ID)		
1	hex ’00’ (CS_CTL)	hex ’FFFFFC’ (S_ID: Directory Server)		
2	hex ’20’ (TYPE: Fibre Channel Services)	hex ’980000’ (F_CTL: Exchange Responder, Last Sequence of Exchange, Last Frame of Sequence)		
3	SEQ_ID	hex ’00’ (DF_CTL)	SEQ_CNT	
4	OX_ID		RX_ID	
5	hex ’00000000’ (Parameter)			
6	hex ’01’ (FC-CT revision)	hex ’XXXXXX’ (IN_ID; used by servers; may have been used)		
7	hex ’FC’ (FCS_Type: Directory Service Application)	hex ’02’ (FCS_Subtype: Name Service)	hex ’00’ (Options: single Exchange)	hex ’00’ (Reserved)
8	hex ’8002’ (Command Code: FS_ACC)		hex ’0000’ (Reserved)	
9	hex ’00’ (Reserved)	hex ’00’ (Reason Code)	hex ’00’ (Explanation Code)	hex ’00’ (Vendor Unique)
10	Port Type	hex ’000000’ (Reserved)		

Words 0-5 contain the frame header of a Fibre Channel Common Services response. The Name Service requires that all requests be made as a single new Exchange, and that the response be returned and the Exchange closed before the next request is issued.

Words 6-9 contain the FC_CT header for the CT_IU. The first byte of word 6 contains the FC_CT revision code; bits 23-0 are a reserved space that may be used to carry the S_ID of the original requestor between different servers; this space may be non-zero if it was used. Word 7 indicates that the response is from the Name Service, and indicates via the Option byte that the service must be completed before the next request is sent. Word 8 indicates the Command Code of an FC_CT Accept (FS_ACC) response. Word 9 is zero since the response is an Accept response.

Word 10 contains the Port Type of the specified Native Port Identifier. A value of hex ’01’ indicates an N_Port, which is a port connected to an F_Port as defined by this document. A value of hex ’02’ indicates an NL_Port, which is a port connected to an FL_Port as defined by this document. A value of hex ’82’ indicates an FL_Port, which indicates the Native Port Identifier is the Loop Fabric Address of the associated loop.

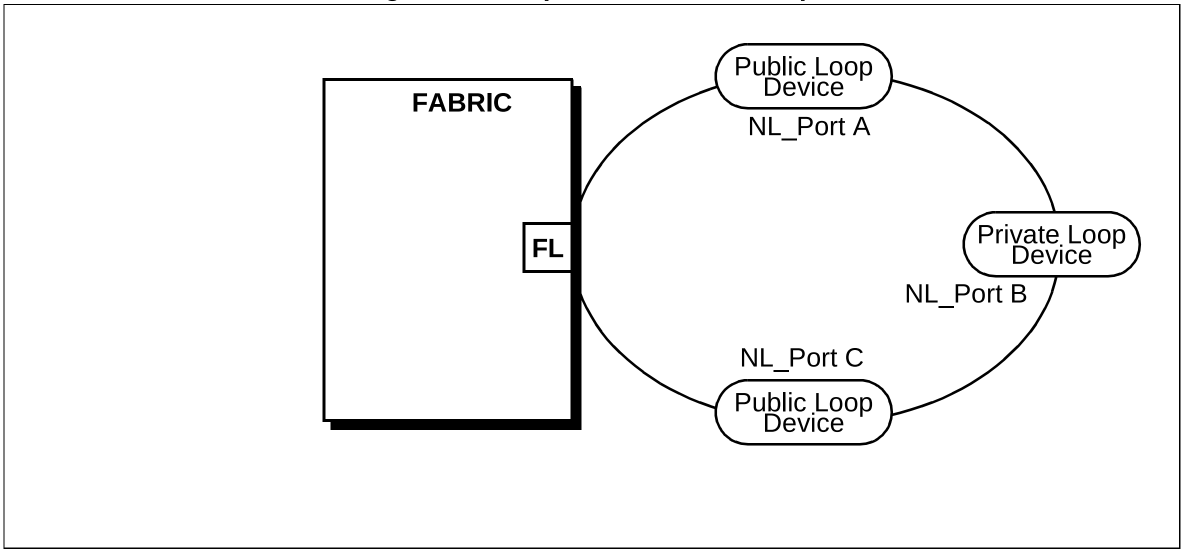
NOTE – Other Port Type codes may be reported, but their meaning has not been defined by this document.

If the response is a reject (FS_RJT) instead of an Accept, then the Command Code in word 8 would be set to hex ’8001’, the Port Type is not returned, and the Reason Code and Explanation Code would be set accordingly.

C.4 FC-FLA Loop Initialization Examples

This clause contains examples of loop initialization for Public Loop devices. Refer to 5.6 and 6.6 for specific requirements for initialization by NL_Ports and FL_Ports. The tables in these examples show the sender and receiver of frames during the initialization process. The following figure shows the loop used to illustrate the examples.

Figure 5 – Loop Initialization Example



C.4.1 Power-up Initialization Example

This example shows the steps taken when a Fabric and an attached loop are first powered-on, or otherwise brought on-line for the first time.

Table C.19 – Power-up Initialization Activities

Activity	FL_Port	NL_Port A	NL_Port B	NL_Port C
1. Enter OPEN-INIT loop state	Send LIPs then fill	Send LIPs then fill	Send LIPs then fill	Send LIPs then fill
2. Select Loop Master	Send LISM until own LISM received	Echo LISM from FL_Port	Echo LISM from FL_Port	Echo LISM from FL_Port
3. Send LISx frames and LIRx frames	Send LISx frames with L_bit set to one, LIRx frames, then CLS	Set bit in LISx frame when appropriate, process LIRx frames, then CLS	Set bit in LISx frame when appropriate, process LIRx frames, then CLS	Set bit in LISx frame when appropriate, process LIRx frames, then CLS
4. Perform FLOGI		Send FLOGI request to FL_Port		
	Receive FLOGI request and send FLOGI ACC			
				Send FLOGI request to FL_Port
	Receive FLOGI request and send FLOGI ACC			

Table C.19 – Power-up Initialization Activities (continued)

Activity	FL_Port	NL_Port A	NL_Port B	NL_Port C
5. Perform RFC-4		Send RFC-4 request to Name Service		
	Receive RFC-4 request and send RFC-4 FS_ACC			
				Send RFC-4 request to Name Service
	Receive RFC-4 request and send RFC-4 FS_ACC			
6. Perform Discovery & PLOGI		Send GP_ID4 request to Name Service		
	Receive GP_ID4 request and send GP_ID4 FS_ACC			
		Send PLOGI to discovered port(s)	Perform PLDA discovery	

In Activity #1, all ports are sending LIP and then clearing the loop by sending the current fill word, as described in FC-AL. Similarly, Activity #2 and Activity #3 proceed as described in FC-AL. In Activity #2, the FL_Port wins because it is the only FL_Port on the loop; the NL_Ports echo the FL_Port LISM as soon as they recognize it. The L_bit is set in at least the LISA Sequence to alert the NL_Ports to the need to perform FLOGI; the NL_Ports would do it anyway since none had completed FLOGI prior to this initialization.

Once all Loop Initialization Sequences have completed and the FL_Port has sent CLS to exit the OPEN-INIT state, the NL_Ports may proceed with Activity #4 to log in with the FL_Port and thus gain access to the Fabric. The NL_Ports know they must perform FLOGI because the L_bit was set in one or more LISx Sequences. Each port in turn arbitrates for the bus and sends FLOGI. The FL_Port receives FLOGI and returns the appropriate FLOGI Accept reply, which includes the FL_Port's Service Parameters, and also contains the ID of the NL_Port, which is constructed from the NL_Port's AL_PA and the Domain || Area of the loop assigned by the Fabric. Note that NL_Port "B" is doing nothing in this case since it is a Private NL_Port.

In Activity #5, the NL_Ports are fulfilling their role in the discovery process by registering their supported FC-4 types with the Name Service. Since the Fabric in this case supports the Name Service, the NL_Ports receive an FS_ACC reply to their request.

In Activity #6, the NL_Ports may then proceed to attempt to discover other appropriate N_Ports and NL_Ports with which to begin operations. NL_Port "A" issues a Name Service request to the Name Service via the FL_Port to get a list of logged in ports. NL_Port "A" may then choose to request the

FC-4 types supported by an N_Port or NL_Port on the list by a Name Service request, or it may query the N_Port or NL_Port directly. See C.3 for specific examples of Name Service requests.

Note that NL_Port “B”, since it is a Private NL_Port, is proceeding with the discovery procedure described in FC-PLDA.

C.4.2 Changed NL_Port Example

This example shows the steps taken when an NL_Port is swapped and the loop re-initializes. In this case, NL_Port “C” is replaced with NL_Port “D”.

Table C.20 – NL_Port Swap Activities

Activity	FL_Port	NL_Port A	NL_Port B	NL_Port D
1. Enter OPEN-INIT loop state	Send LIPs then fill	Send LIPs then fill	Send LIPs then fill	Send LIPs then fill
2. Select Loop Master	Send LISM until own LISM received	Echo LISM from FL_Port	Echo LISM from FL_Port	Echo LISM from FL_Port
3. Send LISx frames and LIRx frames	Send LISx frames with L_bit zero, LIRx frames, then CLS	Set bit in LISx frame when appropriate, process LIRx frames, then CLS	Set bit in LISx frame when appropriate, process LIRx frames, then CLS	Set bit in LISx frame when appropriate, process LIRx frames, then CLS
4. Perform FAN	Send FAN frame to NL_Port A	Receive FAN frame and resume normal operation		
	Send FAN frame to NL_Port C			Receive FAN frame and discard
5. Perform FLOGI			(perform PLDA LIP recovery)	Send FLOGI request to FL_Port
	Receive FLOGI request and send FLOGI ACC			
6. Perform RFC-4			(perform PLDA LIP recovery)	Send RFC-4 request to Name Service
	Receive RFC-4 request and send RFC-4 FS_ACC			
7. Perform Discovery & PLOGI			(perform PLDA LIP recovery)	as needed

In Activity #1, all ports are sending LIP and then clearing the loop by sending the current fill word, as described in FC-AL. Similarly, Activity #2 and Activity #3 proceed as described in FC-AL. In

Activity #2, the FL_Port wins because it is the only FL_Port on the loop; the NL_Ports echo the FL_Port LISM as soon as they recognize it. The L_bit is zero in all LISx Sequences because the FL_Port has detected no changes that would warrant setting the L_bit to one.

Once all Loop Initialization Sequences have completed and the FL_Port has sent CLS to exit the OPEN-INIT state, the FL_Port proceeds with Activity #4 and sends FAN (see A.1.1) to all NL_Ports that have completed FLOGI. NL_Port “A” gets a FAN because it completed FLOGI during the previous initialization. NL_Port “B” does not receive a FAN because it is a Private NL_Port and never performed FLOGI. NL_Port “D” receives a FAN because it resides at the same AL_PA as NL_Port “C” used to reside; NL_Port “D” discards the FAN and waits for a chance to perform FLOGI.

NL_Port “D” performs Activity #5 to log in with the FL_Port and thus gain access to the Fabric. The FL_Port receives FLOGI and realizes that the NL_Port at that AL_PA has changed, and then returns the appropriate FLOGI Accept reply, which includes the FL_Port’s Service Parameters, and also contains the ID of the NL_Port, which is constructed from the NL_Port’s AL_PA and the Domain || Area of the loop assigned by the Fabric.

In Activity #6, the NL_Port “D” registers its supported FC-4 types with the Name Service, as described above.

In Activity #7, NL_Port “D” may then proceed to attempt to discover other appropriate N_Ports and NL_Ports with which to begin operations.

Note that NL_Port “B”, since it is a Private NL_Port, may begin the LIP recovery procedure described in FC-PLDA anytime after the FL_Port has completed FAN delivery.

Annex D
(informative)

BB_Credit Management for Public Loop Devices

D.1 Introduction

Public Loop Devices use “Alternate BB_Credit Management”, as described very well in FC-AL [reference 2]. Additional material may also be found in FC-PLDA [reference 14]. What follows in this annex are some additional explicative notes that do not appear in those references.

D.2 OPN and CLS latencies

When Login_BB_Credit=0, CLS latency is minimized but a latency exists on every OPN while waiting for an R_RDY before frames can be transmitted.

When Login_BB_Credit>0 at an NL_Port, that NL_Port should neither originate nor forward a CLS until it can guarantee that the number of available receive buffers upon receipt of an immediate, subsequent OPN will be greater than or equal to its Login_BB_Credit. To ensure this, one strategy that may be employed is for the NL_Port to never transition from RECEIVED CLOSE to MONITORING, or from XMITTED CLOSE to MONITORING, or from TRANSFER to MONITORING, unless it has Login_BB_Credit buffers available.

When Login_BB_Credit>0, OPN latency is minimized but CLS latency may increase while waiting for Login_BB_Credit buffers to become available. It is possible to reduce CLS latency when Login_BB_Credit>0 by increasing the available receive buffers within a Sequence Recipient. For example, a Sequence Recipient may choose to have:

Available_Rx_Buffers=Available BB_Credit (of current Sequence Initiator) + Login_BB_Credit

at all times during a Loop Tenancy with a Sequence Initiator. This would allow the Sequence Recipient as a CLS Recipient to forward a CLS immediately without having to wait for Login_BB_Credit to become available. A Sequence Recipient controls the Available BB_Credit variable in a Sequence Initiator through the return of R_RDYs.

Some ports may be tempted to reduce CLS latency in another fashion. To prevent buffer overruns, a CLS Recipient is only required to have Login_BB_Credit buffers available before the next OPN is received, not necessarily before the CLS is forwarded. Ports that initiate or forward a CLS before Login_BB_Credit is available and rely on a (topology-specific) delay between the time a CLS is transmitted or forwarded and the time an OPN is received do so at their own risk and with methods not defined in this document.

Annex E
(informative)

Other Frame Collection and Delivery Methods

E.1 Introduction

Several methods of frame collection and frame delivery triggers were identified during the development of this report. Some methods that were deemed more interesting are defined in the main body of this report. Other methods that were deemed less interesting are defined in this annex.

E.2 Frame Collection Methods

Frames may be collected using a variety of different algorithms. This clause will provide a survey of all interesting methods identified during the development of this report; some are defined in 4.3.1. The following table summarizes all methods:

Table E.1 – Frame Collection Algorithms

Collection Method	Collect Frames based on...			
	D_ID	S_ID	SEQ_ID	OPN-CLS at Source
Sequence Collation	•	•	• (1)	
Tenancy Collation	•	•		• (2)
Frame Bundling	•	•		
Frame Piling	•			

NOTES:

1 SEQ_ID is unique for each D_ID/S_ID pair, as per FC-PH clause 18.6.

2 N_Ports cannot source Tenancy Collation because there is currently no way for an N_Port to define the OPN/CLS boundaries.

Sequence Collation is interesting in that it allows the fabric to present complete sequences to the destination, if the destination requires it or is helped by it. This requires more buffering by the fabric, possibly resulting in greater latency. The length of Sequence that can be collected is limited by the size of the available buffer. Also, in Class 2 with ACK_1, it may be impossible to collate a Sequence.

Tenancy Collation sounds interesting at first glance but is probably not truly useful. The general consensus was that an L_Port should be constructed to be independent of Loop Tenancy behavior, mostly because a CLS can occur at any time. Also, the burden for the fabric is great when it must track OPN-CLS behavior across fabric elements; and, if the source generates frames for multiple destinations in one Loop Tenancy, the fabric must create a separate OPN-CLS indication at each destination.

Frame Piling is described in clause 4.3.1.

Frame Bundling is described in clause 4.3.1.

E.3 Frame Collection Delivery Triggers

This clause lists some interesting trigger mechanisms for frame delivery not already documented in clause 4.3.2:

– Delivery can be triggered when one or more Sequences have been accumulated. (Sequence Collation)

– Delivery can be triggered when a complete Loop Tenancy has been accumulated. (Tenancy Collation)

– It was noted that one interesting type of frame content to use as a trigger was to deliver a Basic Link Service or Extended Link Service when the frame is accumulated. This is based on the observation that BLS and ELS Sequences are used for initialization and error recovery, and should be passed along without delay. Also in many instances, after sending a BLS or ELS, the source will send no more frames to that destination until the BLS or ELS has completed.

A

ABTX 32, 64
acronyms 5
address
  Domain 10
  Native Address Identifier 10
Addressing 10
ADISC 32, 64
ADVC 32, 64
AL_PA 10
Allowed 7
Alternate BB_Credit 26, 50
Area 10

B

BB_Credit 18, 20, 37, 42, 56
  Alternate management 18, 20, 37, 42, 56
Broadcast 14

C

Collation 3
Continue Sequence Condition 23, 45, 59

D

Data Compression 21, 22, 43, 44, 57, 58
Directory Server 3
Directory Service Application 3
Domain 10

E

E_D_TOV. See Timers, Error Detect
ECHO 32, 64
ESTC 32, 64
ESTS 32, 64
Exchange
  X_ID Invalidation 23, 45, 59
  X_ID Reassignment 23, 45, 59
Extended Link Services 13

F

Fabric F_Port 3
Fabric Operation 11
FL_Port 11
FLOGI 32, 64
Frame Bundling 3, 12
Frame Piling 3, 12

H

Headers
    Optional 23, 46, 59

I

Invokable 7

L

L_Port 3
LILP/LIRP 26, 50
Link_Control frame 23, 45, 59
Local Device 3
Local Loop 3
LOGO 32, 64
Loop Fabric Address 14
Loop Tenancy 3, 4

M

Multicast 14
Multicast/Selective Replicate 26, 50

N

N_Port Name 24, 37, 46, 60
N_Port Operation 15
Name Service 4, 14
Native Address Identifier 10
NL_Port Operation 15
Node Name 24, 46, 60
Normative references 1
notation 5

O

Open Full Duplex 26, 50
Open Half Duplex 26, 50
OPN Originator 4
OPN Recipient 4

P

PDISC 32, 64
Physical Loop 4
PLOGI 32, 64
Port 10
Private loop device 4
Private NL_Port 4, 9, 15
PRLI 32, 64
PRLO 32, 64
Process_Associator 21, 22, 43, 44, 57, 58

Prohibited 6
Public loop device 4
Public NL_Port 4, 9, 15

Q

Query 4

R

RCS 32, 64
Receive Data Field
    Buffer to Buffer 18, 20, 37, 42, 56
    Class 3 21, 22, 43, 44, 57, 58
Relative Offset
    by Information Category 20, 42, 56
    Continously Increasing 20, 42, 56
    Random 20, 42, 56
Remote Device 4
Remote Loop 4
Required 7
RES 32, 64
RLS 32, 64
RNC 32, 64
RRQ 32, 64
RSI 32, 64
RSS 32, 64
RTV 32, 64

S

Scope 1
Sequences
    Class 3 Concurrent 21, 22, 43, 44, 57, 58
    Concurrent 20, 42, 56
    Open per Exchange 21, 22, 43, 44, 57, 58

T

TEST 33, 65
Timers
    Error Detect (E_D_TOV) 20, 42, 56, 68
    Loop Initialization Sequence Hold (LIS_HOLD_TIME) 68
    Resource Allocation (T_R_TOV) 68
    Resource Recovery (RR_TOV) 68
TPRLO 33, 65
Transfer 26, 50
TRANSFER state 13

U

Unfairness 26, 50

X

X.500 Service 4
X_ID Interlock 23, 45, 59
X_ID Invalidation 23, 45, 59
X_ID Reassignment 23, 45, 59