

Protocol for NCMS

Future & Option Market Clearing System

Version 1.0

February 2020



NSE Clearing Ltd
Exchange Plaza, Plot No. C/1, G Block,
Bandra-Kurla Complex, Bandra (E),
Mumbai - 400 051.

Notice

© Copyright NSE Clearing Ltd (NCL). All rights reserved. Unpublished rights reserved under applicable copyright and trades secret laws.

The contents, ideas and concepts presented herein are proprietary and confidential. Duplication and disclosure to others in whole, or in part is prohibited

CONTENTS

CHAPTER 1	INTRODUCTION.....	4
CHAPTER 2	GENERAL GUIDELINES.....	5
	INTRODUCTION.....	5
	MESSAGE STRUCTURE DETAILS	5
	GUIDELINES FOR PROGRAMMERS	5
	DATA TYPES USED.....	6
	MESSAGE HEADER	6
	ERROR MESSAGE.....	8
	HEART BEAT EXCHANGE.....	8
CHAPTER 3	NCMS COMMUNICATION.....	10
	INTRODUCTION.....	10
	PACKET FORMAT.....	10
	PACKET VALIDATION	11
	PROCESSING BY HOST	11
	PROCESSING BY MEMBER SYSTEM	11
CHAPTER 4	LOGON PROCESS.....	12
	INTRODUCTION.....	12
	ORDER OF EVENTS TO BE FOLLOWED DURING LOGON FOR NCMS FEED.....	12
	LOGON REQUEST & RESPONSE.....	12
	LOGON ERROR.....	14
CHAPTER 5	NCMS MESSAGE DOWNLOAD.....	15
	INTRODUCTION.....	15
	NCMS MESSAGE DOWNLOAD REQUEST.....	15
	TRADE/ ACTION STREAM	17
APPENDIX.....		26
	LIST OF ERROR CODES	26
	LIST OF TRANSACTION CODES.....	27
FAQS.....		28

Chapter 1 Introduction

The NCL NCMS is the trade data feed of NCL. It disseminates information about members / users trades on a real time basis. The data is sent to clients on a TCP/IP communication protocol.

Members need to connect to separate gateways called 'NCMS Gateways' using TCP/IP communication protocol. Login access to NCMS gateway will be via NCMS system user credentials only.

Any changes to the login credentials during the day on the NCMS system will be effective on NCMS gateway on the same day.

Chapter 2 General Guidelines

Introduction

This chapter provides general guidelines for the designers and programmers who develop NCMS API. It also provides information on data types and data size, which can help in understanding various structures.

Message Structure Details

The message structure consists of two parts namely message header and message data. The message header consists of the fields of the header which is prefaced with all the structures. Transaction code, an important field of the message header, is a unique numeric identifier which is sent to or received from the NCMS system. This is used to identify the transaction between the user and the host end.

Guidelines for Programmers

1. All time fields are number of seconds from midnight January 1 1980.

All alphabetical data must be converted to upper case except password before sending to the host. A combination of alphabet, numbers and special characters are allowed in the password. More details on password are explained in later chapters in this document. No NULL terminated strings should be sent to the host end. Instead, fill it with blanks before sending. The strings received from the host end are padded with blanks and are not NULL terminated.

2. All the structures should be defined in the following manner:
 - Items of type char or unsigned char, or arrays containing items of these types, are byte aligned.
 - All structures are pragma pack 2.

- All other types of structure members are word aligned.
- 3. All numeric data must be set to zero (0) before sending to the host, unless a value is assigned to it.
- 4. All reserved fields mentioned, should be mapped to CHAR buffer and initialized to NULL.

Note:

- The values of all the constants and transaction codes given in the document are listed in Appendix.
- The suffix IN in the transaction codes implies that the request is sent from the Client to the NCMS Host end whereas OUT implies that the message is sent from the NCMS Host end to Client

Data Types Used

Table 2.1 DATA TYPES

Data Type	Size of Bytes	Signed / Unsigned
CHAR	1	Signed
INT	4	Signed
SHORT	2	Signed
LONG	8	Signed
DOUBLE	8	Signed and Floating Point
BIT	1 bit	NA

BLANK – Filled with spaces (" ").

Message Header

Each structure is prefaced with a MESSAGE HEADER which is an interactive header. Some data in the header are fixed whereas some data are variable and set differently for each transaction code. The structure of the Message Header is as follows:

6

NCL Confidential

Table 2.2 MESSAGE HEADER

Structure Name	MESSAGE HEADER		
Packet Length	40 bytes		
Field Name	Data Type	Size in Byte	Offset
TransactionCode	SHORT	2	0
RESERVED	LONG	4	2
AlphaChar [2]	CHAR	2	6
User Id	LONG	4	8
ErrorCode	SHORT	2	12
TimeStamp [8]	DOUBLE	8	14
RESERVED [8]	CHAR	8	22
RESERVED [8]	CHAR	8	30
MessageLength	SHORT	2	38

Field Name	Brief Description
TransactionCode	Transaction message number. This describes the type of message received or sent.
AlphaChar [2]	It should be set to BLANK.
ErrorCode	This field should be set to zero while sending messages to the host. In the messages coming from the host, this field describes the type of error. Refer to List of Error Codes in Appendix.
User Id	Member systems must populate relevant User ID
TimeStamp	This field should be set to numeric zero while sending to the host. This is used in host end.
MessageLength	This field should be set to the length of the entire message, including the length of message header while sending to host.

Error Message

When the Error Code in the Message Header is having non zero value, ERROR RESPONSE is sent.
The Error Message will describe the error received.

Table 2.7 ERROR_RESPONSE

Structure Name	ERROR RESPONSE		
Packet Length	180 bytes		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER (Refer Table No 2.2)	STRUCT	40	0
Reserved	CHAR	14	40
Error Message	CHAR	128	54

Field Name	Brief Description
ErrorMessage	Stores the error message. Refer to List of Error Codes in Appendix.

Heart Beat Exchange

Member systems must exchange heartbeat signals with NCMS system during periods of inactivity. NCMS Host will consider the member system as inactive after missing two heartbeats in succession, and disconnect the socket connection. Heartbeats will carry following data in *MessageData* segment of the message. Heartbeat is to be sent only if there is inactivity for 30 seconds. The format is MESSAGE_HEADER with following detail.

Table 2.9 HEARTBEAT

Structure Name	HEARTBEAT		
Packet Length	40 bytes		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER (Refer Table No 2.2)	STRUCT	40	0

Field Name	Brief Description
TransactionCode	The transaction code is HEARTBEAT (23506)

Chapter 3 NCMS Communication

Introduction

TCP/IP communication protocol shall be used between Member System and NCMS Host end as per the Network setup.



Packet Format

Packet structure for communication between Member System and Host End

This structure is applicable to all messages that flow between Client and NCMS Host

Length (2 bytes)	Sequence number (4 bytes)	Checksum(MD5) for Message data (16 bytes)	Message Data (Variable length)
---------------------	------------------------------	---	-----------------------------------

- Max length will be the predefined value of 1024 bytes.
 Length = size of length field (2 bytes) +
 Size of sequence number field (4 bytes) +
 Size of the checksum field (16 bytes) +
 Size of Message data (variable number of bytes as per the transcode).
- Sequence number will start from 1 and will be incremented for every packet.
- Message data will be of variable length and comprises of 40 bytes of message header + variable sized data buffer as per transcode being sent.
- The checksum algorithm used will be MD5. Checksum is applied only on the Message data field and not on the entire packet.

NCL Confidential

- For more details on MD5 refer: [RFC 1321 \(rfc1321\) - The MD5 Message-Digest Algorithm](http://www.faqs.org/rfcs/rfc1321.html)
(<http://www.faqs.org/rfcs/rfc1321.html>)

Packet Validation

Validation will be done for all requests flowing between Member System and Host End.

Validation will be done through the combination of Checksum, Sequence Number and length field.

Processing by Host

Before sending the request to Host, Member System will have to generate a sequence number and checksum value. All the requests being sent from Front-End will be sent in the format described above.

If validation of sequence number, checksum value & length fails at Host End then the disconnection of the socket connection between Member System and Host End will happen.

Processing By Member System

On receiving the response from Host, Member software is expected to validate sequence number, checksum value & length field.

Sequence number must be in sequential order. For any fresh connection the number should start from 1. Checksum field and the checksum recalculated on the data field must match. Length field must be less than or equal to 1024.

If any one of these validations fails, the Member System needs to drop the connection and reestablish a fresh connection.

Chapter 4 Logon Process

Introduction

This section describes how a user logs on to the NCMS system. It covers the log-on request and the system responses.

The trader, after issuing a sign-on request, waits for the system response. The response could be a successful logon or an error message.

Order of Events to Be Followed During Logon for NCMS Feed

The following sequence explains the order in which transaction codes are sent and received during log-on process.

Sequence No	Transaction Code	Sent By	Received By
1	SIGN_ON_REQUEST_IN (2300)	Client	Host End
2	SIGN_ON_REQUEST_OUT (2301)	Host End	Client
3	NCMS_DOWNLOAD_REQUEST (8000)	Client	Host End

Logon Request & Response

When the user wants to establish a connection with the host, he sends SIGN_ON_REQUEST_IN (2300) request. In response to this request SIGN_ON_REQUEST_OUT (2301) is sent from host.

Table 4.1 SIGNON_IN

Structure Name	SIGNON IN		
Packet Length	278 bytes		
Transaction Code	SIGN_ON_REQUEST_IN (2300) SIGN_ON_REQUEST_OUT (2301)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER(Refer Table No 2.2)	STRUCT	40	0
UserId	LONG	4	40
Reserved	CHAR	8	44
Password	CHAR	8	52
Reserved	CHAR	8	60
Reserved	CHAR	38	68
BrokerId	CHAR	5	106
Reserved	CHAR	119	111
Reserved	CHAR	16	230
Reserved	CHAR	16	246
Reserved	CHAR	16	262

Field Name	Brief Description
TransactionCode	The transaction code is SIGN_ON_REQUEST_IN (2300) in login request and SIGN_ON_RESPONSE_OUT (2301) in login response
AlphaChar [2]	In SIGN_IN_RESPONSE_OUT this field contains the number of stream types from which the NCMS data feed is sent. User needs to send the DC_DOWNLOAD_REQUEST (8000) for each stream type to download the trade data.
UserId	This field should contain User ID of user.

Field Name	Brief Description
Password	This field should contain the password entered by the user. A combination of alphabet, numbers and special characters are allowed in the password. The user should enter the valid password for a successful Logon.
BrokerId	This field should contain the member ID.

Logon Error

In case of any error, the structure returned is:

ERROR RESPONSE (Refer to [Error Message](#) in Chapter 2)

Field Name	Brief Description
TransactionCode	The transaction code is SIGN_ON_REQUEST_OUT (2301).
ErrorCode	This contains the error number. Refer to List of Error Codes in Appendix.

Chapter 5 NCMS Message Download

Introduction

NCL NCMS data feed sends the user intended trade confirmation packets to the connected users, for this user has to send the Drop Dopy Download Request on NCMS gateways. In response to this request the trade packets are sent to the user.

The separate request from user is need to be sent for each stream. The number of stream is obtained in SIGNON_OUT from host during login sequence.

NCMS Message Download Request

Table 5.1 NCMS MESSAGE DOWNLOAD

Structure Name	NCMS MESSAGE DOWNLOAD		
Packet Length	48 bytes		
Transaction Code	NCMS_DOWNLOAD_REQUEST (8000)		
Field Name	Data Type	Size in Byte	Offset
MESSAGE HEADER(Refer to Table)	STRUCT	40	0
SequenceCounter	DOUBLE	8	40

Field Name	Brief Description
TransactionCode	The transaction code is NCMS_DOWNLOAD_REQUEST (8000).
AlphaChar	This contains the stream type of the host to which it has to send the DC_DOWNLOAD_REQUEST. 2 – Trade stream 1 – Action stream The alpachar is the character array of size 2. The stream type of the host is sent in the first byte of the alphachar.

15

NCL Confidential

Field Name	Brief Description
	The number of streams is obtained in SIGN_ON_REQUEST_OUT from host during login sequence.
SequenceCounter	This contains the sequence number of the last message received by the user. This can be obtained from the Time Stamp1SequenceCounter of the last received trade confirmation packet's MESSAGE HEADERmessage. To retrieve the messages from the beginning of the day, this field should be set to '0'.

Trade/ Action Stream

In response to NCMS_DOWNLOAD_REQUEST the below mentioned trade/ action packets are sent to user. The trades with sequence number greater than the sequence number, sent in Sequence Counter field of NCMS_DOWNLOAD_REQUEST are only downloaded to the user. After recovering the old messaged the trades happening in online market are also sent in the same packet structure.

Table 5.2 TRADE_CONFIRMATION

Structure Name		TRADE_ACTIVITY_ST		
Packet Length		324 bytes		
Transaction Code		TRADE_CONFIRMATION (6001) GIVEUP_BUY_APPROVE (2001) GIVEUP_SELL_APPROVE (2002) TRADE_MODIFY_IN (5445)		
Sr No	Field Name	Data Type	Size in Byte	Offset
1.	MESSAGE HEADER (Refer to Table 2.2)	STRUCT	40	0
2.	TradeStatus	SHORT	2	40
3.	BookingType	SHORT	2	42
4.	MarketTypeCode	SHORT	2	44
5.	TradingPartyProprietaryIdentification	CHAR	35	46

17

NCL Confidential

Structure Name		TRADE_ACTIVITY_ST		
Packet Length		324 bytes		
Transaction Code		TRADE_CONFIRMATION (6001) GIVEUP_BUY_APPROVE (2001) GIVEUP_SELL_APPROVE (2002) TRADE_MODIFY_IN (5445)		
Sr No	Field Name	Data Type	Size in Byte	Offset
6.	Reserved1	CHAR	1	81
7.	TradingPartyBranchIdentification	SHORT	2	82
8.	BuySellIndicator	SHORT	2	84
9.	TradeQuantityUnit	LONG	8	86
10	DealPrice	LONG	8	94
11.	TradingCapacity	CHAR	4	102
12	ClientIdentification	CHAR	10	106
13	ParticipantIdentification	CHAR	12	116
14	OpenCloseFlag	CHAR	1	128
15	CoverUncoverFlag	CHAR	1	129

18

NCL Confidential

Structure Name		TRADE_ACTIVITY_ST		
Packet Length		324 bytes		
Transaction Code		TRADE_CONFIRMATION (6001) GIVEUP_BUY_APPROVE (2001) GIVEUP_SELL_APPROVE (2002) TRADE_MODIFY_IN (5445)		
Sr No	Field Name	Data Type	Size in Byte	Offset
16	TradeDate	LONG	8	130
17	LastUpdatedDate	LONG	8	138
18	OrderIdentification	DOUBLE	8	146
19	OrderBookingDate	LONG	8	154
20	MarketIdentifierCode	SHORT	2	162
21	ProprietaryMarketIdentification	CHAR	35	164
22	BrokerIdentification	CHAR	5	199
23	SequenceCounter	LONG	8	204
24	ClearingMemberIdentification	CHAR	5	212
25	Reserved2	CHAR	1	217

19

NCL Confidential

Structure Name		TRADE_ACTIVITY_ST		
Packet Length		324 bytes		
Transaction Code		TRADE_CONFIRMATION (6001) GIVEUP_BUY_APPROVE (2001) GIVEUP_SELL_APPROVE (2002) TRADE_MODIFY_IN (5445)		
Sr No	Field Name	Data Type	Size in Byte	Offset
26	ContractIdentification	INT	4	218
27	ClearingParticipantTradeFlag	CHAR	1	222
28	Reserved3	CHAR	1	223
29	PlaceOfClearingIdentification	SHORT	2	224
30	TradeIdentification	INT	4	226
31	TradingSessionIdentification	CHAR	20	230
32	LocationIdentification	CHAR	20	250
33	PartyRole	SHORT	2	270
34	MasterReference	CHAR	16	272
35	TaxIdentification	CHAR	35	288

20

NCL Confidential

Structure Name		TRADE_ACTIVITY_ST		
Packet Length		324 bytes		
Transaction Code		TRADE_CONFIRMATION (6001) GIVEUP_BUY_APPROVE (2001) GIVEUP_SELL_APPROVE (2002) TRADE_MODIFY_IN (5445)		
Sr No	Field Name	Data Type	Size in Byte	Offset
36	Reserved ₄	CHAR	1	323

Field Name	Brief Description
TransactionCode	The transaction code is TRADE_CONFIRMATION (6001).
TradeStatus	Valid trade status (activity type) codes: 11 = ORIG- OriginalTrade 12 = AMUI- ModifiedTrade 13 = CANC- CancelledTrade 15 = UM20- TradeModificationReject 16 = REJT- TradeCancelReject 17 = CONF- GiveupConfirmation 18 = RJCT- GiveupRejection
BookingType	Valid codes: 1 = REGU- Regular 2 = CFOD - ContractForDifference 3 = TRSW- TotalReturnSwap For Action it is always 1
MarketTypeCode	1 = AUCTION- Auction 2 = NORM- Normal 3 = BLKO- Block 4 = ODLT- OddLot For Action it is always 2
TradingPartyProprietaryIdentification	Provides the identification of the trading party. User Id - Trading member firm specific dealer id allotted by the Stock Exchange. It represents the person who supplied the order to the order book For Action this field will be BLANK

TradingPartyBranchIdentification	Identifies a specific branch of a financial institution. Some of the Stock Exchanges may have Dealer ID - Branch No - Corporate Manager ID hierarchy for users assigned to the Trading Member firm. In such cases, it would be a branch number associated with the TradingPartyProprietaryIdentification (User Id). For Action this field will be BLANK
BuySellIndicator	Valid codes: 1 = BUYI- Buy 2 = SELL- Sell
TradeQuantityUnit	Quantity expressed as a number, eg, a number of shares
DealPrice	Specifies the price of the traded financial instrument
TradingCapacity	Valid codes (Account Type): 1 = AGEN- Agent 2 = PROP- Proprietary+ For Action this field will be BLANK
ClientIdentification	(Account Number) Unique and unambiguous identification of the client counterparty
ParticipantIdentification	(Participant Code) Identification of participant account
OpenCloseFlag	Valid codes: OPEN- O CLOSE- X This field will be always O
CoverUncoverFlag	Valid codes: UCOV- U COVR- C This field will be always U

TradeDate	(Activity Time) Specifies the date/time on which the trade was executed
LastUpdatedDate	(Last Modified Time) Date when this trade was last modified
OrderIdentification	This field contains the order number
OrderBookingDate	(Order Entry Date and Time) Date and time when an entry is posted to an account on the account servicer's books For Action this field be BLANK
MarketIdentifierCode	(Exchange) Market Identifier Code. Identification of a financial market, as stipulated in the norm ISO 10383 "Codes for exchanges and market identifications" Exchange ID, coded as: N - NSE B - BSE M - MSE
ProprietaryMarketIdentification	(Market Type – 'N') Context, or geographic environment, in which trading parties may meet in order to negotiate and execute trades among themselves
BrokerIdentification	Party that identifies a broker when required (for example, authorised broker or prime broker)
SequenceCounter	Counter to identify a single data set within the whole transfer
ClearingMemberIdentification	Identification of the clearing member Clearing Member code of NCL with whom the Broker ID/ trading member is linked to clear and settle trades
ContractIdentification	(Token) Identification of the contract
ClearingParticipantTradeFlag	(CP Trade flag) If 'Y', trade is executed on behalf of CP

PlaceOfClearingIdentification	Clearing corporation identifier Designated CC of Broker ID: 4995/4995I - NCL 4994/4994I - ICCL 4996/4996I - MCCIL
TradeIdentification	(Combination of Trade Number + Token Number + Exchange ID) Reference assigned to the trade by the investor or the trading party. This reference will be used throughout the trade life cycle to access/update the trade details
TradingSessionIdentification	(Member Session ID) Identification of a specific execution time bracket code through its trading session name or description.
LocationIdentification	Trader Location id
PartyRole	Role of the party. Valid codes: 4 = CLEA- ClearingFirm 26 = BROK- Broker 3 = CLIE- Client 28 = CUST- Custodian 24 = CUMR- Customer 1 = EXEC- ExecutingFirm 46 = FORE- ForeignFirm 5 = INVE- Investor
MasterReference	(Trade ID - Group of Trades at same price) Unique and unambiguous identifier for a group of individual trades, as assigned by the exchange. This identifier links the individual trades together. For Action this field be BLANK
TaxIdentification	(PAN) Number assigned by a tax authority to an entity.

Appendix

List of Error Codes

Error Code ID	Error Code Value	Description of Error Code
ERR_INVALID_USER_TYPE	16001	Invalid User Type
ERR_USER_ALREADY_SIGNED_ON	16004	User already signed on.
ERR_INVALID_SIGNON	16006	Invalid sign-on, Please try again.
ERR_SIGNON_NOT_POSSIBLE	16007	Signing on to the NCMS system is restricted. Please try later on.
ERR_INVALID_BROKER_OR_BRANCH	16041	Member does not exist in the system.
ERR_USER_NOT_FOUND	16042	User does not exist in the system.
ERR_PROGRAM_ERROR	16056	Program error.
ERR_SYSTEM_ERROR	16104	System could not complete your transaction - ADMIN notified.
ERR_CANT_COMPLETE_YOUR_REQUEST	16123	System not able to complete your request. Please try again.
ERR_USER_IS_DISABLED	16134	This user is disabled. Please call NSE clearing.
ERR_INVALID_USER_ID	16148	Invalid User ID entered.
ERR_INVALID_TRADER_ID	16154	Invalid Trader ID entered.
ERR_BROKER_NOT_ACTIVE	16285	The broker is not active.

List of Transaction Codes

Transaction Code	Code	Structure
SIGNON IN	2300	SIGN_ON_REQUEST_IN
SIGN ON OUT	2301	SIGN_ON_REQUEST_OUT
NCMS MESSAGE DOWNLOAD	8000	DROP_COPY_DOWNLOAD_REQU EST
TRADE_TRANSCODE	6001	TRADE_ACTIVITY_ST
GIVEUP_APRV_BUY_CONF	2001	TRADE_ACTIVITY_ST
GIVEUP_APRV_SELL_CONF	2002	TRADE_ACTIVITY_ST
TRDR_TRADE_MODIFY_IN	5445	TRADE_ACTIVITY_ST

FAQs

Q 1 – What do I need to do before I try connecting directly to NCMS system?

You will have to inform NCL – Help Desk, they will help you with the workflow.

Q 2 – Where to connect?

NCL shall provide a list of addresses, IP address and Port numbers, that member systems must connect to. Member systems must initiate a TCP socket connection to this address.

Q 3 – How to connect?

Member systems must initiate a TCP socket connection to the address given by the NCL. After TCP socket connection, member systems has to follow the login process mentioned in this document

Q 4 – How to Logoff?

Member system has to shut down his TCP connection properly.

Q 5 – What User Ids / Passwords to be used for login to NCMS?

NCMS user ID and password should be used.

Q 6 – How to reset the password through NCMS?

Through NCMS user can't reset the password but the password change done on NCMS system will be get reflected in NCMS API. New login on NCMS API, after password reset on NCMS system, should be done with new password.

Q 7 – With the same user id can we take login on Interactive channel for actions and on NCMS channel?

Yes.

Q 8 – What information shall be provided in the NCMS?

Following information will be sent through NCMS system

- Trade executed
- Trade confirmation
- Trade confirmation rejection
- Trade modification (CP Code) confirmation
- Trade modification (CP Code) reject
- Trade cancel confirmation
- Trade cancel reject

Q 9 – Will clearing member also get trade data?

Yes. All the trades related to a clearing member will be available.

Q 10 – How shall we know that we have received all the trades (End of Day)?

No explicit message will be sent to indicate end of messages.

Q 11 – What happens if I login late or miss receiving some trade in the NCMS channel?

During download request user needs to specify the sequence number from where the messages download should start.

Q 12 – Time from which login available to the system?

Details shall be clarified through a circular.

Q 13 – Can I connect to the NCMS channel after close of market?

Details shall be clarified through a circular.

Q 14 – Till when I can connect to the NCMS channel?

Details shall be clarified through a circular.