

25.01 Smart Track/Universal Hub ISO Message Processing - Input/Output: User's Guide

FINAL (version 2.13)

Date updated: March 26, 2007

Copyright © 2005 by The Depository Trust & Clearing Corporation (“DTCC”). All rights reserved. This work is proprietary and is intended for the exclusive use of DTCC’s Participants and other users of DTCC’s services. No part of this work may be reproduced or distributed (including by transmission) in any form or by any means, or stored in any information storage and retrieval system, without DTCC’s prior written permission.

All requests for additional copies of this work or inquiries about this work should be directed to DTCC Participant Interface Planning, The Depository Trust & Clearing Corporation, 55 Water Street, New York, NY 10041, USA.

Revision History

Date	By	Version	Description
8/21/2002	Gwen Aarons	1.0	Initial document for GCAH
11/28/2002	Gwen Aarons	2.0	Add changes for Universal Hub
03/21/2003	Gwen Aarons	2.1	Changed message structures for Acknowledgements and receipts and reformatted document NOTE: the ISOACK and ISORCP messages do not have Message type numbers assigned at this time
4/21/2003	Gwen Aarons	2.2	In the ISORCP record changed tag for ADTC to REAS Assigned ISO message numbers to ISOACK and ISORCP Changed ISOSUM header to reflect that here is no receiving participant for the summary record Added xxx6 function type to allow FTP and NDM users to receive ISOACK error messages that may occurred after DTCC accepted the message Changed output password record formats for NDM/FTP
4/22/2003	Gwen Aarons	2.3	Added section 1.5 and section 9 to identify the DTF file to be used
5/6/2003	Gwen Aarons	2.4	Added section 8.2.1 for back end DFTP/NDM messages
6/20/2003	Gwen Aarons	2.5	Added <ol style="list-style-type: none"> 1. IMS functions to the table definitions 2. Information that allows the Recipient of a message sent to DTCC to be identified by 'INTDTC' for a one sided transaction 3. Information that allows the field submitter to be identified as 'INTDTC' for a one sided transaction that is being sent from DTCC to a Participant. 4. One sided transactions will not be able to have the RECEIPT option
8/26/2003	Gwen Aarons	2.6	1. Changed value in position 47 of the ISOOUT record. Converted value from submitter's DTC signon number to the actual Participant number or BIC number 2. Changed maximum length of an FTP record from 28004 to 27994 based on DTCC limit for FTP

Date	By	Version	Description
11/17/2003	Gwen Aarons	2.7	<ol style="list-style-type: none"> 1. Corrected rl size from 3 to 4 positions. 2. added note about Users using the record size of the record not the max size
9/21/2004	Gwen Aarons	2.8	<ol style="list-style-type: none"> 1. Corrected description of receipt processing for NDM and FTP in section 1.3 2. corrected name of receipt file from ICMRCP to ISORCP in section 1.3 3. corrected rules of receipts in section 4.0 4. Changed STLR to STL in column product description of input and output table to eliminate confusion and added label of 'and function' to protocol column
1/03/2005	Gwen Aarons	2.9	<ol style="list-style-type: none"> 1. Changed FTP password record layout from version 1 to version 2 in section 3.2.1.1 (increased size of Signon and password fields 2. Changed NDM password record n section 3.2.1.2 record to add test/prod indicator in a filler field 3. Changed FTP output password record from version 01 to 02 in section 4.1.1 and increased size of Signon and password fields- also add filler and therefore the fields after the password offsets are now different
11/03/2005	Gwen Aarons	2.10	<p>Changed required value of 'Delivery Monitoring' field in Blk2 for the ISOINP and ISOACK messages from space(' ') to '2'..(see page # 35 and 38)</p> <p>NOTE: ; Reason for change - SWIFT does not allow space but '2' is valid . IF a user is NOT using SWIFT a value of ' ' will still work</p>

Date	By	Version	Description
11/17/2005	Gwen Aarons	2.11	<p>1. Changed output ISOACK message from a MT532 message type to a MT598 (see 8.2 page # 38)</p> <p>Due to constraints of aM598 record length we had to remove the field the header portion of the original message that was originally in the MT532(Position 87)</p> <p>2. For MQ users – the MQMD there is a field MSGTYPE – If the Input MSGTYPE is set to “1” the reply message will have a “2” in the MSGTYPE field</p> <p>NOTE: If a user is using Merva the MT533 receipt message will not work. Please contact DTCC to discuss the use of an MT598 for receipts</p>
10/10/2006	Gwen Aarons	2.12	1. added SET5 and SET7 to function lists in document
3/26/2007	Gwen Aarons	2.13	<p>Inserted section 8.2(page38) that defines the new format of the MT598 message.</p> <p>This message replaces the old MT532 message that the ARMS project was using.</p> <p>NOTE: SWIFT does not support the MT532 message so we converted to MT598 for other functions</p>

Table of Contents

1. OVERVIEW 1

1.1. TRANSMITTING ISO MESSAGES TO DTCC 1

1.2. RETRIEVING ISO FORMATTED MESSAGES FROM DTCC 1

1.3. RECEIPT PROCESSING 2

1.4. BACK END ERROR PROCESSING FOR FTP/NDM USERS 2

1.5. DTF HUB PARTICIPANT/USER ELIGIBILITY FILE 3

2. ISO PROCESSING INTRODUCTION 4

2.1. INITIATING AN ISO MESSAGE TRANSMISSION 4

2.2. ISO FORMAT INPUT TRANSMISSIONS 4

2.3. ISO FORMAT OUTPUT TRANSMISSIONS 5

3. FTP/NDM ISO FORMATTED INPUT TRANSMISSION FILE REQUIREMENTS 6

3.1. OVERVIEW 6

3.2. FTP/NDM INPUT TRANSMISSION RECORDS 6

 3.2.1. *Security Record* 6

 3.2.2. *Input Transaction Data Message Format for Transmission* 8

3.3. INPUT TRANSMISSION ACKNOWLEDGMENT AND ERROR PROCESSING 9

 3.3.1. *Security Error* 9

 3.3.2. *Severe Error* 9

 3.3.3. *Edit Error Processing on a Single Message (Transaction)* 9

3.4. ISO INPUT MESSAGE ACKNOWLEDGMENT FILE (RESPONSE FILE FROM DTCC) 10

 3.4.1. *Transmission Error* 10

 3.4.2. *Valid Transmission* 10

3.5. ACKNOWLEDGMENT RECORD FORMATS 11

 3.5.1. *NDM Acknowledgment Security Error Record* 11

 3.5.2. *FTP Acknowledgment Security Error Record (CF2ERR)* 12

 3.5.3. *Acknowledgment and Summary Total Record (ISOSUM)* 14

 3.5.4. *Acknowledgment Rejected Detail Message (ISOACK)* 17

4. INITIATING AN ISO OUTPUT/RECEIPT TRANSMISSION USING FTP OR NDM 18

4.1. ISO OUTPUT/RECEIPT REQUEST - TRANSMISSION SECURITY RECORD 18

 4.1.1. *FTP Security (PASSWD) Record for Output/Receipt Processing* 19

 4.1.2. *NDM Security (PASSWD) Record for Output/Receipt Processing* 21

4.2. ISO OUTPUT MESSAGES 23

4.3. ISO RECEIPT MESSAGE 23

5. FTP PROCESSING REQUIREMENTS 24

5.1. FILE REQUESTS TO DTCC 24

 5.1.1. *Input Transmission File Functions (see function descriptions in section 2.2)* 24

 5.1.2. *Request to Retrieve Data Functions (see function descriptions in section 2.3)* 25

5.2. FILE SENT TO USER BY DTCC- ACKNOWLEDGMENT FILE, OUTPUT RESPONSE FILE 25

 5.2.1. *Transmitter's input Acknowledgement and Error messages File* 25

 5.2.2. *User's Message Output Response or Receipt File* 25

5.3. NAMING CONVENTIONS 25

 5.3.1. *Local File Names* 25

 5.3.2. *Remote File Names* 25

5.4. FTP COMMANDS 26

5.4.1.	<i>Put Command</i>	26
5.4.2.	<i>Get Command</i>	26
6.	NDM PROCESSING REQUIREMENTS	28
6.1.	FILE REQUESTS TO DTCC - (INPUT TRANSMISSION FILE, OUTPUT REQUEST FILE).....	28
6.1.1.	<i>Input Transmission File Functions</i>	28
6.1.2.	<i>Requests to Retrieve Data Functions</i>	28
6.2.	FILE SENT TO USER BY DTCC - (ACKNOWLEDGMENT FILE, OUTPUT RESPONSE FILE).....	29
6.2.1.	<i>Acknowledgement and Error Record File</i>	29
6.2.2.	<i>User's (Receiver) Transmission Output Response File</i>	29
6.2.3.	<i>NDM Parameters to be set</i>	29
7.	MQSERIES PROCESSING REQUIREMENTS	30
7.1.	INTRODUCTION.....	30
7.2.	MQSERIES PROCESSING OVERVIEW	30
7.2.1.	<i>ISO Processing MQSeries Cutoffs</i>	30
7.2.2.	<i>Signon and Function Eligibility</i>	30
7.2.3.	<i>MQSeries Message Size, Format and Character Sets</i>	30
7.3.	MQSERIES INPUT PROCESSING	31
7.3.1.	<i>MQSeries Input Queue Name</i>	31
7.3.2.	<i>MQSeries Message ID and Correlation Sets</i>	31
7.3.3.	<i>Message Reply Queues - (return acknowledgments or errors on Input to a User)</i>	31
7.3.4.	<i>Outgoing Response messages (Input Acknowledgments and Errors)</i>	32
7.4.	OUTGOING OUTPUT MESSAGE STRUCTURE.....	33
7.4.1.	<i>Outgoing/Receipt Message Function Table</i>	33
7.4.2.	<i>Outgoing Output Messages</i>	33
7.4.3.	<i>MQSeries Receipt Processing</i>	33
8.	FTP/NDM/MQ RECORD FORMATS	35
8.1.	MQ/FTP/NDM INPUT ISO MESSAGE FORMAT (ISOINP)	35
8.2.	MQ/FTP/NDM ACKNOWLEDGMENT ISO MESSAGE FORMAT (ISOACK-VERSION 2) FOR ALL FUNCTIONS EXCEPT 'GCV',CDL & CDB'	38
8.3.	MQ/FTP/NDM ACKNOWLEDGMENT ISO MESSAGE FORMAT (ISOACK) FOR T GCV, CDL AND CDB USERS ONLY.....	46
8.3.1.	<i>Special ISOACK format for FTP/NDM Back End Error Acknowledgments</i>	54
8.4.	MQ/FTP/NDM OUTPUT ISO MESSAGE FORMAT (ISOOUT) CREATED AT DTCC FOR THE RETRIEVAL OF MESSAGES.....	61
8.5.	MQ/FTP/NDM RECEIPT ISO MESSAGE FORMAT (ISORCP)	65
9.	DTF HUB PARTICIPANT/USER ELIGIBILITY FILE (HUBPT) FORMATS	70

1. Overview

This document describes the communications and processing requirements for Participants who wish to use the DTCC Universal Hub to transmit ISO messages to DTCC and retrieve ISO messages from DTCC.

Transmissions to and from DTCC use one of the following Communication Protocols:

- FTP - TCP/IP File Transfer Protocol software
- NDM – CONNECT:Direct (formerly Network DataMover) software
- MQSeries (IBM software)

The Participant Eligibility file will be available using DTCC's DTF facility. The DTF facility uses only the FTP or NDM protocols.

1.1. Transmitting ISO Messages to DTCC

To transmit ISO Messages to DTCC via one of DTCC's supported communication protocols, it is necessary to prepare an input transaction file or message using the ISO 15022 format and for FTP and NDM users create a password record defined by DTCC (see section 3.2.1). Users will either submit an FTP request; execute an NDM JCL, or route messages via MQSeries in order to communicate with DTCC's computer system.

After the transmission has been received at DTCC, some of the information passed will be verified to confirm that the DTCC Signon for the user submitting the message is allowed to process the message information (BIC Number or Participant ID) being submitted.

- If a message (or a group of messages) is transmitted via FTP or NDM:
 1. A summary message (ISOSUM- see section 3.4) will always be returned for the entire transmission. The summary message indicates the status of the transmission and any severe (critical) errors that may have been detected in the input transmission.
 2. If there are message level edit errors, an acknowledgment message (ISOACK- see section 8.2) will be returned for each message in error.
 3. There may be some errors that cannot be determined until after the summary (ISOSUM) and any errors (ISOACK) are returned. These are called back-end errors and must be retrieved by the user by requesting the function type xxx6 (where xxx= the function – ex: STL6). The type of error that may appear is a duplicate reference id. The format of the message will be the ISOACK- see section 8.2.
- If a message is transmitted via MQSeries, an acknowledgment MQSeries message (ISOACK- see section 8.2) will be returned in the User's MQSeries 'ReplytoQueue'. If there are errors the original message will also be returned with an error reason code.
-
- Valid transactions are processed as long as no severe errors are detected.

NOTE: The Universal Hub will edit only the ISO header blocks (1,2,and 3) information; message type editing will NOT be the responsibility of the Universal Hub.

1.2. Retrieving ISO Formatted Messages from DTCC

FTP or NDM users are required to prepare a request transaction file in their DTCC specified protocol format to retrieve ISO formatted messages from DTCC. MQSeries users will retrieve their messages by reading a queue.

To communicate with DTCC's computer system users will either:

- Submit an FTP request

- Execute NDM JCL
 - For MQSeries retrieval - Identify a queue destination name that will be set-up at DTCC to retrieve messages. Please contact a member of DTCC's Participant Interface Planning group to identify the Queue manager and name. If a 'QMGR' does not exist for a specific user, messages will not be created.
- Note:** This is completed when access to the function is created and is not required again unless the name is changed. Separate queues may be set up for each function.

When an FTP or NDM retrieval request is received at DTCC, the requestor's Signon is verified to confirm the eligibility to retrieve the requested data. Then, an output transmission file is created for the User. In case of any errors, a file is created detailing the errors that have been detected with the transmission.

1.3. Receipt processing

The transmitter (original sender) will have a Receipt created by DTCC when the opposite party (receiver) picks up the message that was transmitted for MQ users. NDM and FTP users will have a receipt generated when the opposite party(receiver)'s file is available for pickup

Note: Receipts will only be generated for functions that are identified to DTCC as requiring receipts.

The message will be formatted as an ISO formatted receipt (ISORCP- see section 8.5). FTP or NDM users are required to prepare a request transaction file in their DTCC-specified format using the Receipt function name for retrieval (e.g. STL8, BUY8) to retrieve ISO formatted receipt messages from DTCC. MQSeries users will retrieve their messages by reading a queue defined as their receipt queue. This queue may be the same the retrieval queue.

To communicate with DTCC's computer system users will either:

- Submit an FTP request
- Execute NDM JCL
- For MQSeries retrieval - Identify a queue destination name that will be set-up at DTCC to retrieve receipt messages. Please contact a member of DTCC's Participant Interface Planning group to identify the Queue manager and name. If a 'QMGR' does not exist for a specific user, messages will not be created.

Note: This is completed when access to the function is created and is not required again unless the name is changed. Separate queues may be set up for each function.

The Receipt message will contain the ISO header reference number that will be equal to the values passed by the Original sender in Header Block 3.

1.4. Back end error processing for FTP/NDM users

FTP or NDM users are required to prepare a request transaction file in their DTCC specified protocol format to retrieve ISO formatted back end errors that may occur from DTCC. The function to be used is 'xxx6' where xxx is the function's system id. (ex: STL6) MQSeries users will NOT have back-end errors.

To communicate with DTCC's computer system users will either:

- Submit an FTP request
- Execute NDM JCL

When an FTP or NDM retrieval request is received at DTCC, the requestor's Signon is verified to confirm the eligibility to retrieve the requested data. Then, an output transmission file will be created for the User that contains all the ISOACK type formatted errors messages that may have occurred.

1.5. DTF Hub Participant/User Eligibility file

A Hub Participant eligibility file will be created every evening around 9:00 pm for retrieval using DTCC's DTF facility. This file will contain a separate record for every hub function that the Participant has requested eligibility from DTCC.

The format of the file is described in section 9 DTF Hub Participant/User Eligibility File (HUBPT) Formats page 70.

2. ISO Processing Introduction

2.1. Initiating an ISO Message Transmission

In order to initiate an ISO formatted message transmission, it is necessary to complete one of the following:

- Transmit to DTCC the appropriate NDM-based JCL. For further information on JCL requirements, please consult the *Network Data Mover via CCF-II User's Guide* and e section 6 titled NDM Processing requirements in this document.
- Issue appropriate FTP commands (FTP-based). For further information on FTP commands, please consult the *File Transfer Protocol (FTP) Function User's Guide* and section 5 titled FTP Processing Requirements in this document.
- Route messages to/from DTCC via IBM MQSeries. For further information on MQSeries requirements at DTCC refer to the *Standard for MQSeries User's Guide* and section 7 titled MQSeries Processing Requirements in this document.

These guides are available from your DTCC Participant Interface Planning representative or on-line at www.dtc.org. (The web site requires a user to register with DTCC and obtain a valid DTCC user id and password).

2.2. ISO Format Input Transmissions

To transmit ISO formatted messages to DTCC via one of DTCC's supported communication protocols, it is necessary to prepare an input transaction file/message in the DTCC-specified format.

The ISO Format Input functions are available subject to the cutoff time(s) indicated in the table below.

<i>Product Description</i>	<i>DTCC Communications Protocols and Functions</i>			<i>Description</i>	<i>Cutoff Time</i>
	<i>FTP</i>	<i>NDM</i>	<i>MQ</i>		
GCV	GCV5	GCV5	GCV7	Global Corp Action Validations	00:00-24:00 Eastern Time
IMS	IMS5	IMS5	IMS7	Inventory Management System	00:00-24:00 Eastern Time
SET	SET5	SET5	SET7	Settlement Input	00:00-20:00 Eastern Time
STL	STL5	STL5	STL7	Stock Loan Recalls	00:00-24:00 Eastern Time

NOTE: Transmissions that are received by DTCC after the cutoff time will be rejected without further processing.

2.3. ISO Format Output Transmissions

ISO formatted output transmissions provide users with the ability to retrieve ISO formatted messages from DTCC.

<i>Product Description</i>	<i>DTCC Communications Protocols and Functions</i>			<i>Description</i>	<i>Cutoff Time</i>
	<i>FTP</i>	<i>NDM</i>	<i>MQ</i>		
IMS	IMS9	IMS9	IMS9	Inventory Management System Notifications	00:00-24:00 Eastern Time
IMS	IMS6	IMS6	N/A	Inventory Management Errors for the Sender	00:00-24:00 Eastern Time
STL	STL9	STL9	STL9	Stock Loan Recall Retrieval	00:00-24:00 Eastern Time
STL	STL8	STL8	STL8	Stock Loan Recall Receipts to Sender	00:00-24:00 Eastern Time
STL	STL6	STL6	N/A	Stock Loan Recall Errors for the Sender	00:00-24:00 Eastern Time

NOTE: Transmissions that are received by DTCC after the cutoff time will be rejected without further processing.

3. FTP/NDM ISO Formatted Input Transmission File Requirements

NOTE: A discussion of ISO Format Input For MQSeries messages can be found in section 7 titled MQSeries Processing Requirements in this document. The section below deals exclusively with FTP/NDM requirements

3.1. Overview

The ISO formatted transmission file consists of a Security record (PSW) used for control purposes, and one or more data records that contain ISO formatted messages.

When an ISO formatted input transmission takes place via FTP or NDM, the actual ISO records have a variable data length up to 27800 characters and a 4-byte variable length field so that the actual record size is 27804.

3.2. FTP/NDM Input Transmission Records

There are two record types that must appear in an ISO Format Input Transmission file. The following is an example of an ISO transmission:

```
SECURITY RECORD                (type "PSW")
ISO FORMAT HEADER RECORD and ISO FORMAT RECORD TYPES (MTxxx, etc)
ISO FORMAT HEADER RECORD and ISO FORMAT RECORD TYPES (MTxxx, etc)
•
•
•
```

3.2.1. Security Record

The Security Record has two different formats depending upon the protocol being used:

1. FTP protocol Security ('PASSWD') record
2. NDM protocol Security ('PSW') record

3.2.1.1 FTP Security ('PASSWD') record

The FTP Security record specifies the input transmitter's (the User's) Signon ID and a legitimate password, together with the unique identifying number of the transmission. The Security record ('PASSWD') must be the first record in the file. Since this record is for security purposes **only**, it is not returned in DTCC's acknowledgment file.

FTP Transmission Security ('PASSWD') Record Format				
Position	Length	Format	Field Name	Description
1	1	Character	Feedback indicator	Must contain value of space
2	1	Character	Test/Production Indicator	Must contain value of 'P'
3	6	Character	Record Type	Must contain value of "PASSWD".
9	2	Numeric	Record Suffix	Must contain value of '01'
11	2	Numeric	Version Number	Must contain value of '02'
13	6	Character	User Reference	Reference id created by user (Optional)
19	8	Character	Addressee	For DTCC use only, do not use
27	9	Character	Signon ID	Format: The entity portion(the first 5 characters) A hyphen (-) and the Individual portion(the last 3 characters) Nnnnn-xxx –valid DTCC Participant number or Gnnnn-xxx- group User ID For example 02199001 would be typed as 02199-001
36	8	Character	Password Field	Valid password, to be obtained from DTCC
44	6	Character	Function Name	Function Name (e.g., STL5). Note: Left aligned with trailing spaces
50	4	Numeric	Transmission ID	Identifies transmission is unique for submitter Non-zero value; zero filled
54	Length of input data	Character	Filler	For DTCC use only, do not use

3.2.1.2 NDM Security ('PSW') record

The NDM Security record specifies the input transmitter's (the User's) Signon ID and a legitimate password, together with the unique identifying number of the transmission. The Security record (PSW) must be the first record in the file. Since this record is for security purposes **only**, it is not returned in DTCC's acknowledgment file.

NDM Transmission Security Record Format				
Position	Length	Format	Field Name	Description
1	3	Character	Record Type	Must contain value of "PSW"
4	6	Character	Signon ID	nnnnn -valid DTCC Participant number or Gnnnn- group User ID Note: Left aligned with trailing spaces
10	6	Character	Password Field	Valid password, to be obtained from DTCC
16	6	Character	Activity Type	Function Name (e.g., STL5)
22	3	Numeric	Transmission ID	Unique transmission identifier. Non-zero value; zero filled
25	4	Character	Filler	For DTCC use only, do not use
29	9	Character	Filler	For DTCC use only, do not use
38	1	Character	Test/Prod Indicator	Must be set to a value of 'P'
39	1	Character	Filler	For DTCC use only, do not use
40	2	Character	Version Number	Version Number (current value must be set to 01)
42	Length of input data	Character	Filler	For DTCC use only, do not use

3.2.2. Input Transaction Data Message Format for Transmission

Each ISO formatted message starts with 3 ISO Header Blocks (1,2, and 3) followed by the detail information for the message type specified in block 2 of the header.

A Unique reference number key must be submitted for every message in Header Block 3 immediately after the tag '108'. This reference number key will be used to identify the original message when a receipt is created.

NOTE: Each reference number submitted to DTCC will be checked for uniqueness for that given day starting at midnight EST. If not unique, the message will be rejected and a reject message will be created for the submitter

See section 8.1 for the detailed format layout of the input message.

3.3. *Input Transmission Acknowledgment and Error Processing*

There are three basic levels of ISO Format Input Transmission File error Processing.

Please Note: The Universal Hub is only editing the header blocks 1,2, and 3 of each message.

3.3.1. Security Error

A security error causes an entire ISO Input Transmission file to be rejected. This happens when either:

- The security record (PSW) fails to properly identify the transmitter to DTCC's system, or
- The User's Signon is not eligible for the selected function at DTCC.

When a security error is detected, the Acknowledgment File contains a single error record (see section 3.5 titled Acknowledgment Record Formats for the FTP or NDM record format).

3.3.2. Severe Error

A Severe Error causes the entire ISO Input transmission File to be rejected (canceled). This happens when one of the following conditions occur:

- The transmission is quiesced (usually due to system problems)
- The Input Transmission encounters an application error at DTCC.

When a severe error is detected, the Acknowledgment File contains a single record (ISOSUM), (see section 3.5.3 titled Acknowledgment and Summary Total Record (ISOSUM) for the record format rules).

3.3.3. Edit Error Processing on a Single Message (Transaction)

A message (transaction) edit error causes ONLY the transaction in error to be rejected. This happens when one of the following occurs:

- The Sender's Participant ID or BIC ID is not valid for the Signon passed in the PSW record.
- A field contains invalid data (e.g., invalid BIC number format).
- A field is not properly formatted, e.g., non-numeric data in a numeric field.
- An unused field in the Block 1, 2 or 3 HDR portion of the ISO formatted record is not properly initialized (*see ISO Input Message Acknowledgment File 3.4 section below for record layout*).

When an Edit Error is detected, the ISO message is rejected and returned in the Acknowledgment file with the appropriate error flags.

SPECIAL NOTE: DTCC accepts and processes all ISO messages that are deemed valid but does NOT return them in the acknowledgment file.

3.4. ISO Input Message Acknowledgment File (Response File from DTCC)

The ISO Acknowledgment File details whether an error occurred or a valid transmission was completed.

Please Note: The Universal Hub is only editing the header blocks 1,2, and 3 of each message.

The ISO Acknowledgment File may contain the following three record types:

3.4.1. Transmission Error

The Acknowledgment Error Record used for Security or Severe Errors:

- ERR - for a Security Violation
- ISOSUM - for a Severe Error

3.4.2. Valid Transmission

The ISO acknowledgment summary (control) record (ISOSUM):

- If no Security Violations are detected, a Summary record is the first record of the Acknowledgment File. There is only one ISOSUM record for a given file transmission.
- If the ISO formatted transmission is accepted as fully valid, the Acknowledgment File containing the summary information in the Summary total record (ISOSUM).
- If any error occurred in the ISO format record as result of Edit errors, the record will be rejected and written to the Acknowledgment File as an ISO Acknowledgment Detail message record (see section 8.2 MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK-VERSION 2)).
- Any Detail message records (ISOACK) will follow the ISOSUM record.

NOTE:

1. For the NDM protocol the entire Acknowledgment File is transmitted to the User
2. For the FTP protocol users must pick up their Acknowledgment File using an FTP request.

3.5. Acknowledgment Record Formats

3.5.1. NDM Acknowledgment Security Error Record

An NDM ISO Acknowledgment Security error is created whenever a security violation occurs; that is, if the Security record (PSW) contains incorrect information. If a security error occurs the 'ERR' record is the only acknowledgment record returned.

NOTE: This record will not have an ISO header and will work the same as any other NDM security error process record

NDM Acknowledgment Security Error Record (ERR) Format				
Position	Length	Format	Field Name	Description
1	3	Character	Record Type	Value is 'ERR' The only record returned because of detection of a security violation
4	8	Character	Signon ID	Signon ID value received in the PSW record transmission, regardless of validity
12	8	Character	Filler	For DTCC use only, do not use
20	6	Numeric	Processing date	MMDDYY, date when the ISO file was received and processed
26	6	Numeric	Activity Type	Activity type submitted in the PSW record
32	3	Numeric	Transmission ID	Transmission ID specified in the PSW record Number
35	2	Character	Filler	For DTCC use only, do not use
37	3	Character	Error Status	Indicator of type of security violation detected by DTCC: 222 - Invalid password detected; use correct password, or contact DTCC to obtain correct password. 333 - Ineligible Signon detected; Signon submitted is ineligible for ISO function: contact DTCC. 901 - Internal Error contact DTCC See *Note below for access to complete list
40	5	Numeric	Filler	For DTCC use only, do not use
45	6	Character	Arrival Time	HHMMSS, time when this ISO transmission arrived at DTCC
51	6	Numeric	Edit Completion Time	HHMMSS, time when DTCC completed editing this ISO transmission
57	24	Numeric	Error Description	Description of error.

*Note: For a complete list of all NDM error status codes, please request a copy of the DTCC *Network DataMover (NDM) (Alias CONNECT:Direct) Function User's Guide* from your DTCC PIP representative.

3.5.2. FTP Acknowledgment Security Error Record (CF2ERR)

An FTP ISO Acknowledgment Security error is created whenever a security violation occurs; that is, if the Security record (PSW) contains incorrect information. If a security error occurs the 'CF2ERR' record is the only acknowledgment record returned.

NOTE: This record will not have an ISO header and will work the same as any other FTP security error process record

FTP Acknowledgment Security Error Record (CF2ERR) Format				
Position	Length	Format	Field Name	Description
1	1	Character	Feedback Indicator	Always contains a '?'
2	1	Character	Test/Prod Indicator	Always contains a 'P'
3	6	Character	Record Type	Value of 'CF2ERR'
9	2	Character	Record Suffix	Value of '01'
11	2	Character	Version Number	Indicates the version number (e.g. current or previous format)
13	6	Character	User Reference Number	For DTCC use only, do not use
19	8	Character	Filler	For DTCC use only, do not use
27	8	Character	Signon ID	Signon of User submitting transmission. Value is from the Security Record
35	6	Numeric	Processing Date	Current date in MMDDYY format
41	6	Character	Function code	Function Name (e.g., STL5)
47	4	Numeric	Transmission ID	Unique transmission identifier. Value is from the Security Record

FTP Acknowledgment Security Error Record (CF2ERR) Format				
Position	Length	Format	Field Name	Description
51	3	Character	Error status	201 - File Id is not in DTCC's table - Please contact DTCC (FTP) 210 - File Id not passed-field must be prefixed by \$ (FTP) 222 - Invalid password detected; use correct password, or contact DTCC to obtain correct password. 333 - Ineligible Signon detected; Signon submitted is ineligible for ISO function: contact DTCC. 901 - Internal Error contact DTCC See *Note below for complete list
54	6	Numeric	Time Processing Started	Time processing started Format=hhmmss
60	6	Numeric	Time Error Message Step Completed	Time error message step completed Format=hhmmss
66	45	Character	Error Description	Description of error status
111	189	Character	Filler	For DTCC use only, do not use

*Note: For a complete list of all FTP error status codes, please request a copy of the DTCC *File Transfer Protocol (FTP) Function User's Guide* from your DTCC PIP representative

3.5.3. Acknowledgment and Summary Total Record (ISOSUM)

An ISO Summary record (ISOSUM), shown below, is created when the transmission is processed. It will indicate whether the transmission had a *Severe Error* or was *Processed Successfully*.

NOTE: If the transmission was processed successfully but had detail data record errors, each detail message record with errors will be returned with the field and error codes attached in the ISO record type MT532 (see section 8.2 MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK-VERSION 2)).

The format of the FTP/NDM ISO Summary record is shown below:

FTP/NDM Acknowledgment Summary Record (ISOSUM) Format				
Position	Length	Format	Field Name	Description
1	1	Character	Starting Delimiter	Must contain value of ‘{’
2	2	Character	Block Identifier	Must contain value of ‘1:’
4	1	Character	Message Identifier	Must contain value of ‘F’
5	2	Character	Protocol Identifier	Must contain value of ‘01’
7	12	Character	Sender’s Signon ID	Sender’s Signon ID
19	4	Numeric	Session Number	Session number Will contain ‘0000’ or the data that occurred on the first message of the original transmission
23	6	Numeric	Sequence Number	Sequence number Will contain ‘000000’ or the data that occurred on the first message of the original transmission
29	1	Character	Ending Block Delimiter	Must contain value of ‘}’
30	1	Character	Starting Delimiter	Must contain value of ‘{’
31	2	Character	Block Identifier	Must contain value of ‘2:’
33	1	Character	Input/Output Identifier	Must contain value of ‘I’
34	3	Character	ISO 15022 Message Type	Value of ‘598’
37	12	Character	Receiver’s ID	Value of Spaces (Since the summary record may be for many different receivers).

FTP/NDM Acknowledgment Summary Record (ISOSUM) Format				
Position	Length	Format	Field Name	Description
49	1	Character	Message Priority	Must contain value of 'N'
50	1	Character	Delivery Monitoring	Value of Space
51	1	Character	Ending Block Delimiter	Must contain value of '}'
52	1	Character	Starting Block Delimiter	Must contain value of '{'
53	2	Character	Block Identifier	Must contain value of '3:'
55	5	Character	Tag for Version number	Must contain value of '{113:'
60	4	Character	Version number	Must contain value of '0301'
64	1	Character	Ending Tag Delimiter	Must contain value of '}'
65	1	Character	Ending Block Delimiter	Must contain value of '}'
66	1	Character	Starting Block Delimiter	Must contain value of '{'
67	2	Character	Block Identifier	Must contain value of '4:'
69	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC
71	5	Character	Transaction reference id Tag	Value of ':20::'
76	7	Character	Type of Record	Value = 'ISOSUM-'
83	5	Character	Function submitted	Value = 'xxxx-' where xxxx= function name
88	4	Character	Transmission number	The user assigned number identifying the FTP/NDM transmission
92	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC
94	5	Character	Sub Message Type Tag	Value of ':12::'
99	3	Character	Sub Message Type value	Value of 'ICM'
102	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC
104	6	Character	Proprietary message tag	Value of ':77E::'

FTP/NDM Acknowledgment Summary Record (ISOSUM) Format				
Position	Length	Format	Field Name	Description
110	1	Character	Transmission Response Code	Blank or the reason the entire transmission was rejected. The following codes may appear: P = past function cutoff time, A = Accepted with no detail record errors X = Accepted with rejected detail records E = Transmission error- data not processed
111	3	Character	Transmission Error status	When Transmission response code is 'E': 100 - Error during FTP Process - contact DTCC 901 - Internal Error - contact DTCC When Transmission response code is 'A': 000 = NO Transmission errors When Transmission response code is 'X': 000 = NO Transmission errors
114	9	Numeric	Total Valid Records	Total number of Valid records received by DTCC for an accepted transmission
123	9	Numeric	Total Invalid Records	Total number of invalid (erroneous) records received by DTCC for an accepted transmission
132	8	Character	Transmission Date	Date data transmitted (MMDDYYYY)
140	8	Character	Start Transmission Time	Time transmission started (HH:MM:SS)
148	8	Character	End Transmission time	Time transmission ended (HH:MM:SS)
156	27	Character	Message Text	Summary message text.
183	3	Character	End of text	'CrLf-' – value of x'0D2560' in EBCDIC
186	1	Character	Ending Delimiter	Value of '{'

3.5.4. Acknowledgment Rejected Detail Message (ISOACK)

Zero or more rejected records will be created in a message error is detected.
Please refer to section 8.2 MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK-
VERSION 2) for the message structure of the acknowledgment message.

4. Initiating an ISO Output/Receipt Transmission Using FTP or NDM

ISO formatted transmissions provide users with the ability to retrieve ISO formatted messages from DTCC.

NOTE: Please refer to section 7 titled MQSeries Processing Requirements in this document for a discussion about receiving ISO formatted messages from DTCC via MQSeries. The section below deals exclusively with FTP/NDM requirements.

When initiating an ISO output request, a user transmits an output request Security record via FTP or NDM requesting either:

- A specific ISO function that contains ISO formatted messages that have been stored in DTCC's tables by a business application and is awaiting pick-up by the appropriate Receiver.
- A specific ISO function that will allow Senders of data to pick up receipt acknowledgments when the receiver has *requested pick up of the original* messages. Receipts will contain an identifier to the original message (will be supplied by user).

The submitted Output request contains one record, the DTCC transmission file Security record (PSW). This record is used to control DTCC User access to ISO Format functions.

4.1. ISO Output/Receipt Request - Transmission Security Record

The ISO output/receipt request utilizes the Transmission Security Record to identify the requestor to the DTCC system. In addition to the Signon and password, Users will also provide Control information (BIC number) to verify the request.

The Security Record has two different formats depending upon the protocol being used:

- FTP protocol Security ('PASSWD') record
- NDM protocol Security ('PSW') record

4.1.1. FTP Security (PASSWD) Record for Output/Receipt Processing

FTP Output Request Record (PASSWD) Format				
Position	Length	Format	Field Name	Description
1	1	Character	Feedback indicator	Must contain value of space
2	1	Character	Test/Production Indicator	Must contain value of 'P'
3	6	Character	Record Type	Must contain value of "PASSWD".
9	2	Numeric	Record Suffix	Must contain value of '01'
11	2	Numeric	Version Number	Must contain value of '02'
13	6	Character	User Reference	Reference id created by user (Optional)
19	8	Character	Addressee	DTCC Internal use
27	9	Character	Signon ID	Format: The entity portion(the first 5 characters) A hyphen (-) and the Individual portion(the last 3 characters) Nnnnn-nnn –valid DTCC Participant number or Gnnnn-nnn- group User ID For example 02199001 would be typed as 02199-001
36	8	Character	Password Field	Valid password, to be obtained from DTCC.
44	6	Character	Function Name	Function Name (e.g., STL9). Note: Left aligned with trailing spaces
50	4	Numeric	Transmission ID	Identifies transmission is unique for submitter Non-zero value; zero filled.
54	10	Character	Filler	For DTCC use only, do not use
64	3	Character	Retrieval Processing type	Allows user to select type of data to be retrieved. This field is used in conjunction with 'Date-Time retrieval parm fields' beginning in position 66: <ul style="list-style-type: none"> • UNP – all data that has not been delivered yet (a date and time may optionally be entered) • PRC – all data that has been processed for a certain date and time • ALL – all Unprocessed and

FTP Output Request Record (PASSWD) Format				
Position	Length	Format	Field Name	Description
				<p>Processed data a certain date and time</p> <ul style="list-style-type: none"> • Default = spaces – treated same as UNP but optional date/time fields are NOT allowed
67	12	Character	BIC ID or Participant ID	<p>BIC ID number or Participant ID number associated with firm requesting output data</p> <p>Participant ID should be 8 bytes will trailing blanks</p>
79	6	Character	Retrieval start date selection (optional)	<p>Starting Retrieval Date</p> <p>Format = YYMMDD</p> <p>Not required for 'UNP' processing type (if not entered will retrieve all unprocessed data for the last 5 days)</p>
85	4	Character	Retrieval start time selection (optional)	<p>Starting Retrieval time</p> <p>Format = HHMM</p> <p>Not required</p> <p>Default = 0000</p> <p>(if not entered will retrieve for entire day or all unprocessed data)</p>
89	2	Character	Retrieval end hour	<p>Ending Retrieval time</p> <p>Format = HH</p> <p>Not required – if entered the Retrieval start time must also be entered</p> <p>Default = 23</p> <p>NOTE: Will retrieve everything up to the last second of the hour specified (ex: Retrieval start time = 0910 retrieval end time = 15 The file will contain everything from 091000-155959)</p>
91	1	Character	Number of retrieval days (optional)	<p>How many days from the 'Retrieval Start Date and time' forward –</p> <p>Default = 0 (retrieves data for the 'Retrieval start date' date only</p> <p>Max value = 5 days</p>
92	8	Character	Filler	For DTCC use only, do not use

4.1.2. NDM Security (PASSWD) Record for Output/Receipt Processing

NDM Output Request Record (PSW) Format				
Position	Length	Format	Field Name	Description
1	3	Character	Record Type	Must contain value of "PSW".
4	6	Character	Signon ID	nnnnn -valid DTCC Participant number or Gnnnn - group User ID Note: Left aligned with trailing spaces.
10	6	Character	Password Field	Valid password, to be obtained from DTCC
16	6	Character	Function Name	Function name (e.g., STL9)
22	3	Numeric	Transmission ID	Unique Transmission ID. Non-zero value; zero filled
25	3	Character	Retrieval Processing type	Allows user to select type of data to be retrieved. This field is used in conjunction with Date-Time retrieval fields beginning in position 40: <ul style="list-style-type: none"> • UNP – all data that has not been delivered yet (a date and time may optionally be entered) • PRC – all data that has been processed for a certain date and time • ALL – all Unprocessed and Processed data a certain date and time • Default = spaces – treated same as UNP but optional date/time fields are NOT allowed
28	12	Character	BIC ID or Participant ID	BIC ID number or Participant ID number associated with firm requesting output data Participant ID should be 8 bytes will trailing blanks
40	6	Character	Retrieval start date selection (optional)	Starting Retrieval Date Format = YYMMDD Not required for 'UNP' processing type (if not entered will retrieve all

NDM Output Request Record (PSW) Format				
Position	Length	Format	Field Name	Description
				unprocessed data for the last 5 days)
46	4	Character	Retrieval start time selection (optional)	Starting Retrieval time Format = HHMM Not required Default = 0000 (if not entered will retrieve for entire day or all unprocessed data)
50	2	Character	Retrieval end hour	Ending Retrieval time Format = HH Not required – if entered the Retrieval start time must also be entered Default = 24 NOTE: Will retrieve everything up to the last second of the hour specified (ex: Retrieval start time = 0910 retrieval end time = 15 The file will contain everything from 091000-155959
52	1	Character	Number of retrieval days (optional)	How many days from the 'Retrieval Start Date' forward Default = 0 (retrieves data for the 'Retrieval start date' date only)
53	2	Character	Version Number	Version number (current value must be set to 01)
55	26	Character	Filler	For DTCC use only, do not use

4.2. ISO Output Messages

Based on the ISO function requested, data will be extracted into a sequential file. The ISO output message format is described in section 8.4 MQ/FTP/NDM Output ISO Message Format (ISOOUT) created at DTCC for the retrieval of messages.

4.3. ISO Receipt Message

Based on the ISO Receipt function requested, data will be extracted into a sequential file for each receipt processed. The ISO Receipt record is described in section 8.5 MQ/FTP/NDM Receipt ISO Message Format (ISORCP)

5. FTP Processing Requirements

A Participant (User) who wants to initiate an ISO transmission via FTP must complete the setup requirements specified in the *File Transfer Protocol (FTP) Function User's Guide* before initiating an FTP request.

FTP Users can issue appropriate FTP commands to submit or retrieve ISO message files.

5.1. File Requests to DTCC

NOTE: **The maximum size of a message (excluding headers) is currently limited to 27,000 characters.**

*Users should use the **ACTUAL RECORD SIZE** for the ISO message type submitted, not the maximum size available whenever possible since the messages are variable length.*

**For Ex: If the ISO function msg MTxxx is 200 bytes long
the LRCL should be 200+87+5+4 =296**

ISO Header Blocks (87 bytes)	Block 4 Start Tags (5 bytes)	ISO Message Data (Maximum length of 27,000)	End Tags (4 bytes)
------------------------------------	------------------------------------	--	--------------------------

5.1.1. Input Transmission File Functions (see function descriptions in section 2.2)

*Note: For variable length records the file LRCL length is always 4 bytes more than the record size. Therefore based on the example above the LRCL would be 300.
The BLKSIZE is also 4 bytes bigger than the LRCL therefore in the example the BLKSIZE would be 304.*

ISO FUNCTION	FTP Request	RECFM	Max LRCL	Max BLKSIZE
GCV5	put (Input)	VB	27,100	27,104
IMS5	put (Input)	VB	27,100	27,104
SET5	put (Input)	VB	27,100	27,104
STL5	put (Input)	VB	27,100	27,104

5.1.2. Request to Retrieve Data Functions (see function descriptions in section 2.3)

ISO FUNCTION	FTP Request	RECFM	LRCL	BLKSIZE
IMS9	put (Request)	FB	80	N/A
IMS6	put (Request)	FB	80	N/A
STL9	put (Request)	FB	80	N/A
STL8	put (Request)	FB	80	N/A
STL6	put (Request)	FB	80	N/A

5.2. File Sent to User by DTCC- Acknowledgment file, output response file

5.2.1. Transmitter’s input Acknowledgement and Error messages File

This file is created from an Input Request.

ISO FUNCTION	FTP Request	RECFM	LRCL	BLKSIZE
XXX5 Where XXX5 = any of the functions listed in table 5.1.1 above	get (Ack)	VB	28,004	28,004

5.2.2. User’s Message Output Response or Receipt File

This file is created when an Output request is received at DTCC.

ISO FUNCTION	FTP Request	RECFM	LRCL	BLKSIZE
YYYY Where YYYY = any of the functions listed in table 5.1.2 above	get (Output)	VB	28,004	28,004

5.3. Naming Conventions

FTP files are named according to specific rules. These names also change according to the order in which they are used. There are two types of file names - local and remote.

5.3.1. Local File Names

Files created by the User to be transmitted to DTCC are local files. The user can choose their names freely, however, they cannot contain blanks. Otherwise, naming of local files is only restricted by the user’s computer operating system.

5.3.2. Remote File Names

Files that are created on the DTCC mainframe (i.e., resulting from the *put* of a local file) are considered remote files and must be named according to the following format:

‘FTXaaaaa.bccc.\$FTXDRVR.\$dddddd.\$eeffff.g’

Any file sent to DTCC via an FTP ‘put’ is identified by its file identification name, which corresponds to a function name. For ISO functions the file does requires a password record and detail records. Their formats are described in previous sections of this document.

Entry	Description
'FTX	'FTX (literal). Please note the single quotation mark.
aaaaa.	The first five digits of your RACF Signon followed by a period.
b	P (for production). All input and output testing can be done in production.
ccc.	The last three digits of your RACF Signon followed by a period.
\$FTXDRVR	\$FTXDRVR - corresponds with the Process/Driver name at DTCC.
\$dddddd.	dddddd - The four to six byte function/File-id you are requesting followed by a period. Right-justified blanks are not required for function names less than six bytes.
\$eee	eee - A three byte location code assigned by the user. The first three characters must be alphabetic.
ffff.	A four-digit sequence number assigned by the user followed by a <u>period</u> .
IMPORTANT: A remote file name cannot be reused for 7 calendar days. The three byte location code (eee) and four digit sequence number (ffff) form a parameter that can be varied to generate remote file names that are unique over a 7 day period. For example, functions that are requested once a day, could alter the four digit sequence numbers to reflect the date, e.g., 0401 (April 1), 0402 (April 2), etc.	
g'	I (for input- files being sent to DTCC) or O (for Output- files being retrieved from DTCC), Followed by a single quotation mark.

5.4. FTP Commands

The following commands are used for the actual FTP process after the local files have been created and named. Users should pay attention to the interaction between the *put* and *get* commands within this process.

5.4.1. Put Command

The *put* command submits a function request file to DTCC. The syntax is:

```
put local_file name 'remote_file_name'
```

For example, a User with the RACF Signon of 99999201, submitting a local file named *input.data*, containing function=STL5, could use the following command:

```
put input.data 'FTX99999.P201.$FTXDRVR.$STL5.$abc0001.I'
```

5.4.2. Get Command

The *get* command retrieves the response file from the input request, whether it be an Acknowledgment Summary Record or Error record (e.g., for STL5 input ISO messages) or actual input data (e.g., if the User submitted a request to retrieve Messages STL9).

The syntax is:

```
get 'remote_file_name' local_file_name
```

In the 'remote_file_name', the I (for input) needs to change to an O (for output), and the \$ must be stripped off (removed) from each node. Users may also want to change the name of the 'local_file_name' so it will not wipe out the requesting record contained in the original local file.

Using the example cited above for the *put* command, the User retrieving the response file from their *input.data* STL5 request could use the following command, which changes the 'local_file_name' from *input.data* to *retrieved.messages*:

```
get 'FTX99999.P201.FTXDRVR.STL5.abc0001.O' retrieved.messages
```

Please note that in the output dataset the \$ prefix of each node is stripped out, and the output length is a variable length record of 91 to 27994 bytes for any FTXDRVR response record (see "ACKN" record).

6. NDM Processing requirements

The following describes the NDM procedures and requirements associated with the ISO Functions. Users should also refer to the *Network Data Mover via CCF-II User's Guide* if any setup requirements are needed.

This guide is available from your DTCC Participant Interface Planning representative or on-line at www.dtc.org. (The web site requires a user to register with DTC and obtain a valid DTCC user id and password).

6.1. File Requests to DTCC - (Input Transmission file, Output Request file)

NOTE: **The maximum size of a message data (excluding headers) is currently limited to 27,000 characters**

Users should use the ACTUAL RECORD SIZE for the ISO message type submitted, not the maximum size available whenever possible since the messages are variable length

**For Example: If the ISO function msg MTxxx is 200 bytes long
the record size should be 200+87+5+4 =296**

ISO Header Blocks (87 bytes)	Block 4 Start Tags (5 bytes)	ISO Message Data (Maximum length of 27,000)	End Tags (4 bytes)
--	--	---	----------------------------------

6.1.1. Input Transmission File Functions

Note: For variable length records the file LRCL length is always 4 bytes more than the record size. Therefore based on the example above the LRCL would be 300. The BLKSIZE is also 4 bytes bigger than the LRCL therefore in the example the BLKSIZE would be 304.

<i>ISO FUNCTION</i>	<i>NDM PROCEDURE</i>	<i>RECFM</i>	<i>Max LRCL</i>	<i>Max BLKSIZE</i>
GCV5	NDMFILIN	VB	27,100	27,104
IMS5	NDMFILIN	VB	27,100	27,104
SET5	NDMFILIN	VB	27,100	27,104
STL5	NDMFILIN	VB	27,100	27,104

6.1.2. Requests to Retrieve Data Functions

<i>ISO FUNCTION</i>	<i>NDM PROCEDURE</i>	<i>RECFM</i>	<i>LRCL</i>	<i>BLKSIZE</i>
IMS9	NDMDDMOU	FB	80	N/A
IMS6	NDMDDMOU	FB	80	N/A
STL9	NDMDDMOU	FB	80	N/A

<i>ISO FUNCTION</i>	<i>NDM PROCEDURE</i>	<i>RECFM</i>	<i>LRCL</i>	<i>BLKSIZE</i>
STL8	NDMDDMOU	FB	80	N/A
STL6	NDMDDMOU	FB	80	N/A

6.2. File sent to User by DTCC - (Acknowledgment file, Output Response file)

6.2.1. Acknowledgement and Error Record File

This file is created from an Input Request.

<i>ISO FUNCTION</i>	<i>NDM PROCEDURE</i>	<i>RECFM</i>	<i>Max LRCL</i>	<i>Max BLKSIZE</i>
XXX5 Where XXX5 = any of the functions listed in table 6.1.1 above	NDMFILIN	VB	28,004	28,008

6.2.2. User's (Receiver) Transmission Output Response File

This file is created when an Output request is received at DTCC.

<i>ISO FUNCTION</i>	<i>NDM PROCEDURE</i>	<i>RECFM</i>	<i>LRCL</i>	<i>BLKSIZE</i>
YYYY Where YYYY = any of the functions listed in table 6.1.2 above	NDMDDMOU	VB	28,004	28,008

6.2.3. NDM Parameters to be set

Please modify the following Parameter in the NDM General Process Member to the default value below since this parameter is overwritten by the size of the dataset passed in the variable length record:

RL=0001

Set the SACCT parameter to indicate the return dataset name. The RL field should contain 0001 but the actual record length will be the record length specified for the function.

SACCT = \RD= return dataset RL= 0001 RS=I

For further information on JCL requirements, please consult the *Network Data Mover via CCF-II User's Guide*.

7. MQSeries Processing Requirements

7.1. Introduction

This section gives a brief overview of MQSeries ISO processing requirements. It assumes a working knowledge of how to transmit data to DTCC using the MQSeries protocol. For documentation related to MQSeries at DTCC please contact a DTCC Participant Interface Planning (PIP) representative.

7.2. MQSeries Processing Overview

7.2.1. ISO Processing MQSeries Cutoffs

The ISO processing application allows data to be queued even after the business cutoff has been taken. During this time period Participants are allowed to transmit to DTCC. Their messages will remain in the queue until the business function cutoff has been opened (function is available). Then the data will be processed by the business application.

NOTE: To allow for necessary system maintenance, there may be an interruption of service during the day. If a message is received during that time, it will be rejected as being received past cutoff and will require the user to resubmit the transactions.

7.2.2. Signon and Function Eligibility

The User's Signon will be verified for eligibility to input the function record. Any error will be stored in the User's 'ReplytoQmgr/ReplytoQ' (the reply to queue specified in the MQMD header).

7.2.3. MQSeries Message Size, Format and Character Sets

- The maximum size of a message (excluding headers) is limited to 27,000 characters.
- Participants are required to send messages with the format identifier of MQFMT_STRING. This informs MQSeries that the data is in the format of a character string.
- MQSeries provides for the translation of character string messages across platforms.
- Participants are allowed to send data in the native character set of their environment, either ASCII or EBCDIC.
- DTCC will send responses in DTCC's native character set, which is EBCDIC.

7.3. MQSeries Input Processing

7.3.1. MQSeries Input Queue Name

The rule for naming the input queue to DTCC is as follows:

ICM.hhh_ffff.INP01

Where hhh is the DTCC high level system name (e.g., STL, IMS, etc.) and ffff is the function input being requested (e.g., STL7, IMS7).

The following lists the DTCC queue name required for each ISO input function.

ISO MQSeries Function	Description	DTCC Queue Name
GCV7	Global Corp Action Validations for ISO messaging	ICM.GCS_GCV7.INP01
IMS7	Inventory Management System for ISO Messaging	ICM.IMS_IMS7.INP01
SET7	Settlement Input (DO, PO, etc)	ICM.SET_SET7.INP01
STL7	Stock Loan Recall Hub for ISO messaging	ICM.HUB_STL7.INP01

7.3.2. MQSeries Message ID and Correlation Sets

- MQSeries assures that a unique message ID is generated on a put to a queue, if the proper options are specified.
- To assist in problem resolution and message tracking, the User is required to use this feature.
- Responses delivered from DTCC will contain the participant’s original MQSeries message ID in the MQSeries Correlation ID.

Incoming Message Structure

- The data being transmitted via MQSeries is composed of 3 ISO Header blocks and the actual ISO message.
- The ISO Input Message structure format is described in section 8.1 MQ/FTP/NDM Input ISO Message Format (page 35).
- The MQSeries Message Descriptor precedes the data record.
- A message contains one transaction, to a maximum length of 27,000 bytes (excluding header block information).
- The following is an example of a typical input MQSeries message.

MQMD	ISO Header Blocks (87 bytes)	Block 4 Start Tags (5 bytes)	ISO Message Data (Maximum length of 27,000)	End Tags (4 bytes)
-------------	--	--	---	------------------------------

7.3.3. Message Reply Queues - (return acknowledgments or errors on Input to a User)

When a user transmits ISO formatted messages, the ‘ReplytoQmgr’ and ‘ReplyToQ’ name specified in the MQSeries message header is used to return any errors encountered to the user (see section 7.3.4).

NOTE: If the ‘ReplytoQ’ field contains an empty queue name or the User’s ‘ReplytoQ’ manager is invalid (not defined at DTCC), the error message will NOT be available for retransmission.

7.3.4. Outgoing Response messages (Input Acknowledgments and Errors)

- The processing of a message results in one response message being sent to the ‘ReplyToQ’ name indicated in the MQMD header of the original message.
- The Reference number submitted in header block 3 of the original message will be stored in the header block 3 of the response message, allowing the User a method of reconciling Messaging input to Messaging responses.
- The ISO Input Acknowledgment Message structure format is described in section 8.2 MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK-VERSION 2) (page 38).
- The following is an example of a typical input response MQSeries message

MQMD	ISO Header Blocks (87 bytes)	Block 4 Start Tags (5 bytes)	ISO Acknowledgment Data (Minimum of 195) (Maximum length of 443)	End Tags (4 bytes)
-------------	--	--	---	------------------------------

- There is a special outgoing message used when a back-end error is encountered. The following is an example of the output response message.

MQMD	ISO Header Blocks (134 bytes)	Block 4 Start Tags (5 bytes)	ISO Acknowledgment Data (Minimum of 195) (Maximum length of 443)	End Tags (4 bytes)
-------------	---	--	---	------------------------------

7.4. *Outgoing Output Message Structure*

ISO Format output messages are written to an MQSeries Queue created at the User’s site if they are eligible for that function.

The name of the **queue** must be given to the User’s Participant Interface Planning representative to be added to DTCC’s system. You may view this queue name using the DDMI function

The data being transmitted to the User via MQSeries adheres to the ISO transaction format standard. An MQSeries Message Descriptor will simply precede it. A message contains one transaction, to a maximum length of 27,000 bytes (excluding header information).

7.4.1. Outgoing/Receipt Message Function Table

<i>ISO MQSeries Function</i>	<i>Description</i>	<i>User Queue Manager and Queue Name</i>
IMS9	Inventory Management System	Name defined by User
STL9	Stock Loan Recalls	Name defined by User
STL8	Stock Loan Recalls Hub	Name defined by User

7.4.2. Outgoing Output Messages

DTCC’s Universal Hub will retrieve a message and send it to the User’s Queue Manager based on the user supplied Queue Name and Remote Queue manager.

All messages are composed of three parts:

- The MQSeries Message Descriptor
- The output message headers
- Followed by the original message

The ISO Output Message structure format is described in section 8.4MQ/FTP/NDM Output ISO Message Format (ISOOUT) created at DTCC for the retrieval of messages. (Page 61).

The following is an example of a typical output MQSeries message.

MQMD	ISO Header Blocks (134 bytes)	Block 4 Start Tags (5 bytes)	ISO Message Data (Maximum length of 27,000)	End Tags (4 bytes)
-------------	---	--	---	------------------------------

NOTE: The Output ISO header message length is different than the input header message length.

7.4.3. MQSeries Receipt Processing

NOTE: Some functions may not offer the Receipt Option. Please check Function table in section 7.4.1 Outgoing/Receipt Message Function Table above.

An MQSeries User (Sender) submits a message to DTCC to be transmitted to another party (Receiver).

1. When the Recipient (opposite party) is an FTP or NDM user, an MQSeries Receipt (ISORCP) will be put on the Sender's receipt queue with a Receipt message 'DTCC has sent message to Recipient' after the FTP or NDM retrieval process has been completed.
Note: The data after the tag ':22H:REDE//' will also contain 'DELE'.

2. When the Recipient (opposite party) is an MQSeries User:
 - An MQSeries Receipt message will be put on the Sender's receipt queue with the Receipt message 'DTCC has sent message to Recipient' when DTCC puts in on the Recipient's MQSeries Queue.
Note: The data after the tag ':22H:REDE//' will also contain 'DELE'.
 - Another MQSeries receipt message will be put on the Sender's receipt queue with the Receipt message 'Recipient has picked up message' when DTCC receives notification that the Recipient has read the message from their queue.
Note: The data after the tag ':22H:REDE//' will also contain 'RECE'.

3. DTCC's Universal Hub will retrieve a message and send it to the User's Queue Manager based on the user supplied Queue Name and Remote Queue manager for Receipts (this may be the same as the output queue name and manager or different).
All Receipt messages are composed of three parts:
 - The MQSeries Message Descriptor
 - The output message header
 - Followed by the new message type MT533)

4. The ISO Receipt Message structure format is described in section 8.5 MQ/FTP/NDM Receipt ISO Message Format (ISORCP) (page 65)

The following is an example of a typical Receipt MQSeries message.

MQMD	ISO Header Blocks (116 bytes)	Block 4 Start Tags (5 bytes)	ISO Receipt Data (Length of 214)	End Tags (4 bytes)
------	----------------------------------	------------------------------------	-------------------------------------	--------------------------

8. FTP/NDM/MQ Record Formats

8.1. MQ/FTP/NDM Input ISO Message Format (ISOINP)

Each ISO formatted message starts with 3 ISO Header blocks followed by the detail information for the message type specified in Header block 2.

The format of the input message is shown below:

MQ/FTP/NDM Input ISO Message Format (ISOINP)				
Position	Length	Format	Field Name	Description
Header Block 1- Basic Header				
1	1	Character	Starting Delimiter	Must contain value of ‘{’
2	2	Character	Block Identifier	Must contain value of ‘1.’
4	1	Character	Message Identifier	Must contain value of ‘F’
5	2	Character	Protocol Identifier	Must contain value of ‘01’
Submitter’s BIC Number or DTCC Participant ID				
7	8	Character	Bank/Firm Code	One of the following: 1. Submitter’s Bank Identifier code 2. User’s Participant ID (If Signon is a Group User this number must be connected in DTCC’s Group user eligibility tables)
15	1	Character	Logical terminal	Identifies Terminal Type
16	3	Character	Branch Code	Identifies Branch
19	4	Numeric	Session Number	Default value of ‘0000’ Assigned by submitter based on ISO standards; Not validated by DTCC
23	6	Numeric	Sequence number	Default value of ‘000000’ assigned by submitter based on ISO standards; Not validated by TCC
29	1	Character	Ending Block Delimiter	Must contain value of ‘}’
Header Block 2- Application Header				

MQ/FTP/NDM Input ISO Message Format (ISOINP)				
Position	Length	Format	Field Name	Description
30	1	Character	Starting Delimiter	Must contain value of ‘{’
31	2	Character	Block Identifier	Must contain value of ‘2:’
33	1	Character	Input/Output Identifier	Must contain value of ‘I’
34	3	Character	ISO 15022 Message Type	Must contain valid ISO message type
<i>Recipient’s BIC Number or Participant Number</i>				
37	8	Character	Bank/Firm Code	One of the following based on function definition: 1. Recipient’s Bank Identifier code 2. DTCC Participant ID of Recipient 3. Value of ‘INTDTC’- Internal DTC User Note: Used only when the function used is a one party transaction and the Recipient of the message is an internal DTC application.
45	1	Character	Logical terminal	Identifies Terminal Type when a BIC id is entered in Position 37-44
46	3	Character	Branch Code	Identifies Branch when a BIC id is entered in Position 37-44
49	1	Character	Message Priority	Must contain value of ‘N’
50	1	Character	Delivery Monitoring	Must contain value of ‘2’
51	1	Character	Ending Block Delimiter	Must contain value of ‘}’
<i>Header Block 3 - User Header</i>				
52	1	Character	Starting Delimiter	Must contain value of ‘{’
53	2	Character	Block Identifier	Must contain value of ‘3:’
55	5	Character	Tag for Version number	Must contain value of ‘{113:’
60	4	Character	Version number	Must contain value of ‘0301’
64	1	Character	Ending Tag Delimiter	Must contain value of ‘}’

MQ/FTP/NDM Input ISO Message Format (ISOINP)				
Position	Length	Format	Field Name	Description
65	5	Character	Tag for Submitter's Reference Key	Must contain value of '{108:'
70	16	Character	Submitter's Reference Key	Unique key created by Submitter used to identify transaction when receipt is issued for submitter
86	1	Character	Ending Tag Delimiter	Must contain value of '}'
87	1	Character	Ending Block Delimiter	Must contain value of '}'
<i>Text (Data) Block</i>				
88	1	Character	Starting Delimiter	Must contain value of '{'
89	2	Character	Block Identifier	Must contain value of '4:'
91	2	Character	Carriage Return/Line Feed	'CrLf' – value of x'0D25' in EBCDIC
93	Length of Record 1-27000)	Data fields		
	3	Character	End of TEXT Carriage Return/Line Feed and dash(-)	'CrLf-' – value of x'0D25-' in EBCDIC
	1	Character	End of Text Data Block 4	Must contain Value of '}'

8.2. MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK-VERSION 2) for all functions except ‘GCV’, CDL & CDB’

An ISO Acknowledgment record (ISOACK), shown below, is created:

1. When an MQ message is processed. This message will always be created. It will indicate whether the message had a function or transaction level *Error* or was *Processed Successfully*
2. When an FTP/NDM transmission is processed this message will be created only when there is a transaction edit error.
3. When an FTP/NDM transmission has a back-end error. This means the error occurred after the transmission was accepted by DTCC. Currently the only cause for this error is that the reference number submitted is a duplicate. There is a separate layout for this message. See section 8.3.1 Special ISOACK format for FTP/NDM Back End Error Acknowledgments, Page number 54.

There are three types of acknowledgment messages that can be generated as a result of ISO processing:

- Function Level Error - The function is quiesced or not identified in the Hub profile table. The acknowledgment message for a function level error contains the ICM an 8-byte error code that is broken down into a 4-byte data code, a 4-byte error code describing the reason for the function error.
- Transaction Edit Error - When an individual transactions fails editing, the acknowledgment message contains an 8-byte error code that is broken down into a 4-byte data code, a 4-byte error code describing the reason for the Edit error.
- All TransactionsValid - For messages where a transaction has passed editing, the user will simply receive a reply message containing the reference number that identifies the original message.

The format of the acknowledgment message is shown below:

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK) –VERSION 2					
Position	Length	Format	Field Name	Description	M/O*
Header Block 1- Basic Header					
1	1	Character	Starting Delimiter	Must contain value of ‘{’	M
2	2	Character	Block Identifier	Must contain value of ‘1:’	M
4	1	Character	Message Identifier	Must contain value of ‘F’	M
5	2	Character	Protocol Identifier	Must contain value of ‘01’	M
Recipient Number					
7	12	Character	Recipient’s BIC Number/	One of the following: 1. Recipient’s Bank Identifier	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK) –VERSION 2					
Position	Length	Format	Field Name	Description	M/O*
			Participant ID	code 2. User’s Participant ID (same number passed to DTCC in the ISOINP message)	
19	4	Numeric	Session Number	Session number - contains the number that was entered by the user in the originally submitted input message	M
23	6	Numeric	Sequence Number	Sequence number – contains the number that was entered by the user in the originally submitted input message	M
29	1	Character	Ending Block Delimiter	Must contain value of ‘}’	M
Header Block 2- Application Header					
30	1	Character	Starting Delimiter	Must contain value of ‘{’	M
31	2	Character	Block Identifier	Must contain value of ‘2:’	M
33	1	Character	Input/Output Identifier	Must contain value of ‘O’	M
34	3	Character	ISO 15022 Message Type	Value of ‘598’	M
37	4	Character	Time message was accepted at DTCC	Format =HHMM	M
41	6	Character	Date message was accepted at DTCC	Format=YYMMDD	M
Submitter Number					
47	12	Character	Submitter’s BIC Number/Participant ID	One of the following based on function definition: 1. Submitter’s Bank Identifier code 2. Submitter’s Participant ID (same number passed to DTCC in the ISOINP message) 3. Value of ‘INTDTC’- Internal DTC User	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK) –VERSION 2					
Position	Length	Format	Field Name	Description	M/O*
				Note: Used only when the function used is a one party transaction and the Recipient of the message is an internal DTC application.	
59	4	Numeric	Session Number	1. Value input by submitter based on ISO standards for Hub functions 2. Default value of '0000' if not passed by DTCC application	M
63	6	Numeric	Sequence number	1. Value input by submitter based on ISO standards for Hub functions 2. Default value of '000000' if not passed by DTCC application	M
69	6	Character	Date message was sent from DTCC to a Recipient	Format=YYMMDD	M
75	4	Character	Time message was sent from DTCC to a Recipient	Format =HHMM	M
79	1	Character	Message Priority	Must contain value of 'N'	M
80	1	Character	Ending Block Delimiter	Must contain value of '}'	M
Header Block 3 - User Header					
81	1	Character	Starting Block Delimiter	Must contain value of '{'	M
82	2	Character	Block Identifier	Must contain value of '3:'	M
84	5	Character	Tag for Version number	Must contain value of '{113:'	M
89	4	Character	Version number	Must contain value of '0301'	M
93	1	Character	Ending Tag Delimiter	Must contain value of '}'	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK) –VERSION 2					
Position	Length	Format	Field Name	Description	M/O*
94	5	Character	Tag for Submitter's Reference Key	Must contain value of '{108:'	M
99	16	Character	Submitter's Reference Key	Unique key created by Submitter used to identify transaction when receipt is issued for submitter	M
115	1	Character	Ending Tag Delimiter	Must contain value of '}'	M
116	5	Character	Tag for Submitter's Reference Key	Must contain value of '{115:'	M
121	11	Character	Time with seconds that message was sent from DTCC to a Recipient	Timestamp in format HH.MM.SS.MM	M
132	1	Character	Ending Block Delimiter	Must contain value of '}'	M
<i>Text (Data) Block</i>					
133	1	Character	Starting Block Delimiter	Must contain value of '{'	M
134	2	Character	Block Identifier	Must contain value of '4:'	M
136	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
138	4	Character	Tag	Value of ':20:'	M
142	16	Character	Submitter's Reference Key	Unique key created by Submitter used to identify transaction when receipt is issued for submitter	M
158	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
160	4	Character	Tag	Value of ':12:'	M
164	3	Character	Value	Value of '532'	M
167	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
169	5	Character	Tag	Value of ':77E:'	M
174	9	Character	Start of GENL Block	Value of ' :16R:GENL'	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK) –VERSION 2					
Position	Length	Format	Field Name	Description	M/O*
183	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
185	15	Character	DTCC Function Type Tag	Value of ':22F:TRAN/DTCY/'	M
200	4	Character	DTCC Function Type	DTCC Function Name (e.g., STL7, STL5, IMS7, IMS5)	M
204	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
206	12	Character	ACK Message Status Tag	Value of ':25D:/MSGs/'	M
218	4	Character	ACK -Message Status	Value = 'ACCP' or 'ERRR'	M
222	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
224	11	Character	Date/Time Transaction Received at DTCC Tag	Value of ':98C:SENR/'	M
235	8	Character	Date Transaction Received at DTCC	Format=YYYYMMDD	M
243	6	Character	Time Transaction Received at DTCC	Format=HHMMSS	M
249	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
251	11	Character	Date/Time Acknowledgement created at DTCC Tag	Value of ':98C:PROC/'	M
262	8	Character	Date Acknowledgment Created at DTCC	Format=YYYYMMDD	M
270	6	Character	Time Acknowledgment Created at DTCC	Format=HHMMSS	M
276	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
278	9	Character	End of GENL	Value of ':16S:GENL'	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK) –VERSION 2					
Position	Length	Format	Field Name	Description	M/O*
			Block		
287	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
289	13	Character	Start of ACCTINFO Block	Value of ‘ :16R:ACCTINFO’	M
302	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
304	15	Character	Submitter of Message Tag	Value of ‘:95R:PTYA/DTCY/’	M
319	8	Character	Submitter of Message DTCC Signon Number	Submitter of Message DTCC Signon ID	M
327	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
329	15	Character	Recipient of Message Tag	Value of ‘:95R:PTYB/DTCY/’	M
344	8	Character	Recipient of Message DTCC Signon Number	One of the following based on function definition: 1. Recipient of Message Signon ID 2. Value of ‘INTDTC’- Internal DTC User NOTE: Used only when the function used is a one party transaction and the Recipient of the message is an internal DTC application.)	M
352	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
354	13	Character	End of ACCTINFO Block	Value of ‘ :16S:ACCTINFO’	M
367	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
369	9	Character	Start of STAT Block	Value of ‘ :16R:STAT’	O
	2	Character	Carriage	CrLf – value of x’0D25’ in	O

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK) –VERSION 2					
Position	Length	Format	Field Name	Description	M/O*
			Return/Line Feed	EBCDIC	
	15	Character	DTCC Field in Error Tag	Value of ‘:24B:ERRR/DTCY/’	O
	4	Character	DTCC Field in Error Code	4-byte data field code (a list of codes is available using PTS function ICMF or through retrieval of a DTF file from DTCC))	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	O
	15	Character	DTCC Reason code for error Tag	Value of ‘:24B:REJT/DTCY/’	O
	4	Character	DTCC Reason code for Error	4-byte error code (a list of codes is available using PTS function ICMF or through retrieval of a DTF file from DTCC)	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	O
	11	Character	DTCC Text Description of Error Tag	Value of ‘:70C:REAS//’	O
	35	Character	DTCC Text Description of Error	35 character Text description of Field in Error	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	O
	35	Character	DTCC Text Description of Error	35 character Text describing reason why a field has an error	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	O
	11	Character	Additional information	Value of ‘:70G:ADTX//’	O
	2	Character	Literal used to identify the original MT message type	Value of ‘MT’	O
	3	Character	Original Message type	xxx (ex:543)	O

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK) –VERSION 2					
Position	Length	Format	Field Name	Description	M/O*
	7	Character	Literal used to identify tracking number	Value of ‘TRK ID:’	O
	16	Character	CDTS Tracking number(For future use only)	Tracking number or spaces	O
	12	Character	filler	For future	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	O
	9	Character	End of STAT Block	Value of ‘ :16S:STAT’	O
	3	Character	End of TEXT Carriage Return/Line Feed	‘CrLf-‘ – value of x’0D2560’ in EBCDIC	M
	1	Character	End of Text Data Block 4	Must contain Value of ‘}’	M

*M/O: M=Mandatory, O=Optional.

8.3. MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK) ONLY for GCV, CDL and CDB users

An ISO Acknowledgment record (ISOACK), shown below, is created:

4. When an MQ message is processed. This message will always be created. It will indicate whether the message had a function or transaction level *Error* or was *Processed Successfully*
5. When an FTP/NDM transmission is processed this message will be created only when there is a transaction edit error.
6. When an FTP/NDM transmission has a back-end error. This means the error occurred after the transmission was accepted by DTCC. Currently the only cause for this error is that the reference number submitted is a duplicate. There is a separate layout for this message. See section 8.3.1 Special ISOACK format for FTP/NDM Back End Error Acknowledgments, Page number 54.

There are three types of acknowledgment messages that can be generated as a result of ISO processing:

- Function Level Error - The function is quiesced or not identified in the Hub profile table. The acknowledgment message for a function level error contains the ICM an 8-byte error code that is broken down into a 4-byte data code, a 4-byte error code describing the reason for the function error.
- Transaction Edit Error - When an individual transactions fails editing, the acknowledgment message contains an 8-byte error code that is broken down into a 4-byte data code, a 4-byte error code describing the reason for the Edit error.
- All Transactions Valid - For messages where a transaction has passed editing, the user will simply receive a reply message containing the reference number that identifies the original message.

The format of the acknowledgment message is shown below:

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
<i>Header Block 1 - Basic Header</i>					
1	1	Character	Starting Delimiter	Must contain value of ‘{’	M
2	2	Character	Block Identifier	Must contain value of ‘1:’	M
4	1	Character	Message Identifier	Must contain value of ‘F’	M
5	2	Character	Protocol Identifier	Must contain value of ‘01’	M
<i>DTCC Submitter Number</i>					
7	12	Character	Submitter’s BIC	One of the following:	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
			Number/ Participant ID	3. Submitter's Bank Identifier code 4. User's Participant ID (same number passed to DTCC in the ISOINP message)	
19	4	Numeric	Session Number	Session number - contains the number that was entered by the user in the originally submitted input message	M
23	6	Numeric	Sequence Number	Sequence number – contains the number that was entered by the user in the originally submitted input message	M
29	1	Character	Ending Block Delimiter	Must contain value of ‘}’	M
Header Block 2 -Application Header					
30	1	Character	Starting Delimiter	Must contain value of ‘{’	M
31	2	Character	Block Identifier	Must contain value of ‘2:’	M
33	1	Character	Input/Output Identifier	Must contain value of ‘I’ (NOTE: For STL6 function this field will contain an O)	M
34	3	Character	ISO 15022 Message Type	Value of ‘598’	M
Recipient's BIC Number or Participant ID					
37	12	Character	Recipient's BIC Number/Participant ID	One of the following based on function definition: 4. Recipient's Bank Identifier code 5. Recipient's Participant ID (same number passed to DTCC in the ISOINP message) 6. Value of ‘INTDTC’- Internal DTC User Note: Used only when the function used is a one party transaction and	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
				the Recipient of the message is an internal DTC application.	
49	1	Character	Message Priority	Must contain value of 'N'	M
50	1	Character	Delivery Monitoring	Must contain value of '2'	M
51	1	Character	Ending Block Delimiter	Must contain value of '}'	M
Header Block 3 - User Header					
52	1	Character	Starting Block Delimiter	Must contain value of '{'	M
53	2	Character	Block Identifier	Must contain value of '3:'	M
55	5	Character	Tag for Version number	Must contain value of '{113:'	M
60	4	Character	Version number	Must contain value of '0301'	M
64	1	Character	Ending Tag Delimiter	Must contain value of '}'	M
65	5	Character	Tag for Submitter's Reference Key	Must contain value of '{108:'	M
70	16	Character	Submitter's Reference Key	Unique key created by Submitter used to identify transaction when receipt is issued for submitter	M
86	1	Character	Ending Tag Delimiter	Must contain value of '}'	M
87	1	Character	Ending Block Delimiter	Must contain value of '}'	M
Text (Data) Block					
88	1	Character	Starting Block Delimiter	Must contain value of '{'	M
89	2	Character	Block Identifier	Must contain value of '4:'	M
91	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
93	4	Character	Tag	Value of ':20:'	M
97	16	Character	Submitter's	Unique key created by	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
			Reference Key	Submitter used to identify transaction when receipt is issued for submitter	
113	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
115	4	Character	Tag	Value of ':12:'	M
119	3	Character	Value	Value of '532'	M
122	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
124	5	Character	Tag	Value of ':77E:'	M
129	9	Character	Start of GENL Block	Value of ' :16R:GENL'	M
138	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
140	15	Character	DTCC Function Type Tag	Value of ':22F:TRAN/DTCY/'	M
155	4	Character	DTCC Function Type	DTCC Function Name (e.g., STL7, STL5, IMS7, IMS5)	M
159	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
161	12	Character	ACK Message Status Tag	Value of ':25D:/MSG//'	M
173	4	Character	ACK -Message Status	Value = 'ACCP' or 'ERRR'	M
177	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
179	11	Character	Date/Time Transaction Received at DTCC Tag	Value of ':98C:SENR//'	M
190	8	Character	Date Transaction Received at DTCC	Format=YYYYMMDD	M
198	6	Character	Time Transaction Received at DTCC	Format=HHMMSS	M
204	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
206	11	Character	Date/Time Acknowledgement created at DTCC Tag	Value of ‘:98C:PROC//’	M
217	8	Character	Date Acknowledgment Created at DTCC	Format=YYYYMMDD	M
225	6	Character	Time Acknowledgment Created at DTCC	Format=HHMMSS	M
231	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
233	9	Character	End of GENL Block	Value of ‘:16S:GENL’	M
242	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
244	13	Character	Start of ACCTINFO Block	Value of ‘ :16R:ACCTINFO’	M
257	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
259	15	Character	Submitter of Message Tag	Value of ‘:95R:PTYA/DTCY/’	M
274	8	Character	Submitter of Message DTCC Signon Number	Submitter of Message DTCC Signon ID	M
282	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
284	15	Character	Recipient of Message Tag	Value of ‘:95R:PTYB/DTCY/’	M
299	8	Character	Recipient of Message DTCC Signon Number	One of the following based on function definition: 3. Recipient of Message Signon ID 4. Value of ‘INTDTC’- Internal DTC User NOTE: Used only when the function used is a one party transaction and the Recipient of the message	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
				is an internal DTC application.)	
307	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
309	13	Character	End of ACCTINFO Block	Value of ' :16S:ACCTINFO'	M
322	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
324	9	Character	Start of STAT Block	Value of ' :16R:STAT'	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	O
	15	Character	DTCC Field in Error Tag	Value of ' :24B:ERRR/DTCY/'	O
	4	Character	DTCC Field in Error Code	4-byte data field code (a list of codes is available using PTS function ICMF or through retrieval of a DTF file from DTCC))	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	O
	15	Character	DTCC Reason code for error Tag	Value of ' :24B:REJT/DTCY/'	O
	4	Character	DTCC Reason code for Error	4-byte error code (a list of codes is available using PTS function ICMF or through retrieval of a DTF file from DTCC)	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	O
	11	Character	DTCC Text Description of Error Tag	Value of ' :70C:REAS//'	O
	35	Character	DTCC Text Description of Error	35 character Text description of Field in Error	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	O

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
	35	Character	DTCC Text Description of Error	35 character Text describing reason why a field has an error	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	O
	11	Character	Additional information	Value of ':70G:ADTX//'	O
	2	Character	Literal used to identify the original MT message type	Value of 'MT'	O
	3	Character	Original Message type	xxx (ex:543)	O
	7	Character	Literal used to identify tracking number	Value of 'TRK ID:'	O
	16	Character	CDTS Tracking number(For future use only)	Tracking number or spaces	O
	12	Character	filler	For future	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	O
	9	Character	End of STAT Block	Value of ':16S:STAT'	O
	3	Character	End of TEXT Carriage Return/Line Feed	'CrLf-' – value of x'0D2560' in EBCDIC	M
	1	Character	End of Text Data Block 4	Must contain Value of '{'	M

*M/O: M=Mandatory, O=Optional.

8.3.1. Special ISOACK format for FTP/NDM Back End Error Acknowledgments

Since the back end errors are retrieved using the output system, the format of the ISO headers follows the output standard.

Therefore the header record for the ISOACK uses the output format when the message is retrieved using the xxx6 function type (where xxx = system; ex STL, IMS, etc).

The format of the xxx6 (Back End) acknowledgment message is shown below:

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
1	1	Character	Starting Delimiter	Must contain value of ‘{’	M
2	2	Character	Block Identifier	Must contain value of ‘1:’	M
4	1	Character	Message Identifier	Must contain value of ‘F’	M
5	2	Character	Protocol Identifier	Must contain value of ‘01’	M
7	12	Character	Submitter’s BIC Number/ Participant ID	One of the following: 1. Submitter’s Bank Identifier code 2. User’s Participant ID (same number passed to DTCC in the ISOINP message)	M
19	4	Numeric	Session Number	Session number - contains the number that was entered by the user in the originally submitted input message	M
23	6	Numeric	Sequence Number	Sequence number – contains the number that was entered by the user in the originally submitted input message	M
29	1	Character	Ending Block Delimiter	Must contain value of ‘}’	M
30	1	Character	Starting Delimiter	Must contain value of ‘{’	M
31	2	Character	Block Identifier	Must contain value of ‘2:’	M
33	1	Character	Input/Output Identifier	Must contain value of ‘I’ (NOTE: For STL6 function	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
				this field will contain an O	
34	3	Character	ISO 15022 Message Type	Value of '598'	M
37	12	Character	Recipient's BIC Number/Participant ID	<p>One of the following based on function definition:</p> <p>5. Recipient's Bank Identifier code</p> <p>6. Recipient's Participant ID (same number passed to DTCC in the ISOINP message)</p> <p>7. Value of 'INTDTC'- Internal DTC User</p> <p>Note: Used only when the function used is a one party transaction and the Recipient of the message is an internal DTC application.</p>	M
49	1	Character	Message Priority	Must contain value of 'N'	M
50	1	Character	Delivery Monitoring	Must contain value of '2'	M
51	1	Character	Ending Block Delimiter	Must contain value of '}'	M
52	1	Character	Starting Block Delimiter	Must contain value of '{'	M
53	2	Character	Block Identifier	Must contain value of '3:'	M
55	5	Character	Tag for Version number	Must contain value of '{113:'	M
60	4	Character	Version number	Must contain value of '0301'	M
64	1	Character	Ending Tag Delimiter	Must contain value of '}'	M
65	5	Character	Tag for Submitter's Reference Key	Must contain value of '{108:'	M
70	16	Character	Submitter's Reference Key	Unique key created by Submitter used to identify transaction when receipt is issued for submitter	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
86	1	Character	Ending Tag Delimiter	Must contain value of ‘}’	M
87	1	Character	Ending Block Delimiter	Must contain value of ‘}’	M
Text (Data) Block					
88	1	Character	Starting Block Delimiter	Must contain value of ‘{’	M
89	2	Character	Block Identifier	Must contain value of ‘4:’	M
91	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
93	4	Character	Tag	Value of ‘:20:’	M
97	16	Character	Submitter’s Reference Key	Unique key created by Submitter used to identify transaction when receipt is issued for submitter	M
113	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
115	4	Character	Tag	Value of ‘:12:’	M
119	3	Character	Value	Value of ‘532’	M
122	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
124	5	Character	Tag	Value of ‘:77E:’	M
129	9	Character	Start of GENL Block	Value of ‘ :16R:GENL’	M
138	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
140	15	Character	DTCC Function Type Tag	Value of ‘:22F:TRAN/DTCY/’	M
155	4	Character	DTCC Function Type	DTCC Function Name (e.g., STL7, STL5, IMS7, IMS5)	M
159	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
161	12	Character	ACK Message Status Tag	Value of ‘:25D:/MSG\$//’	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
173	4	Character	ACK -Message Status	Value = 'ACCP' or 'ERRR'	M
177	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
179	11	Character	Date/Time Transaction Received at DTCC Tag	Value of ':98C:SENR//'	M
190	8	Character	Date Transaction Received at DTCC	Format=YYYYMMDD	M
198	6	Character	Time Transaction Received at DTCC	Format=HHMMSS	M
204	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
206	11	Character	Date/Time Acknowledgement created at DTCC Tag	Value of ':98C:PROC//'	M
217	8	Character	Date Acknowledgment Created at DTCC	Format=YYYYMMDD	M
225	6	Character	Time Acknowledgment Created at DTCC	Format=HHMMSS	M
231	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
233	9	Character	End of GENL Block	Value of ':16S:GENL'	M
242	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
244	13	Character	Start of ACCTINFO Block	Value of ':16R:ACCTINFO'	M
257	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
259	15	Character	Submitter of Message Tag	Value of ':95R:PTYA/DTCY//'	M
274	8	Character	Submitter of Message DTCC	Submitter of Message DTCC Signon ID	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
			Signon Number		
282	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
284	15	Character	Recipient of Message Tag	Value of ':95R:PTYB/DTCY/'	M
299	8	Character	Recipient of Message DTCC Signon Number	One of the following based on function definition: 8. Recipient of Message Signon ID 9. Value of 'INTDTC' - Internal DTC User NOTE: Used only when the function used is a one party transaction and the Recipient of the message is an internal DTC application.)	M
307	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
309	13	Character	End of ACCTINFO Block	Value of ':16S:ACCTINFO'	M
322	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	M
324	9	Character	Start of STAT Block	Value of ':16R:STAT'	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	O
	15	Character	DTCC Field in Error Tag	Value of ':24B:ERRR/DTCY/'	O
	4	Character	DTCC Field in Error Code	4-byte data field code (a list of codes is available using PTS function ICMF or through retrieval of a DTF file from DTCC))	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	O
	15	Character	DTCC Reason code for error Tag	Value of ':24B:REJT/DTCY/'	O

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
	4	Character	DTCC Reason code for Error	4-byte error code (a list of codes is available using PTS function ICMF or through retrieval of a DTF file from DTCC)	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	O
	11	Character	DTCC Text Description of Error Tag	Value of ':70C:REAS//'	O
	35	Character	DTCC Text Description of Error	35 character Text description of Field in Error	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	O
	35	Character	DTCC Text Description of Error	35 character Text describing reason why a field has an error	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	O
	11	Character	Additional information	Value of ':70G:ADTX//'	O
	2	Character	Literal used to identify the original MT message type	Value of 'MT'	O
	3	Character	Original Message type	xxx (ex:543)	O
	7	Character	Literal used to identify tracking number	Value of 'TRK ID:'	O
	16	Character	CDTS Tracking number(For future use only)	Tracking number or spaces	O
	12	Character	filler	For future	O
	2	Character	Carriage Return/Line Feed	CrLf – value of x'0D25' in EBCDIC	O
	9	Character	End of STAT Block	Value of ':16S:STAT'	O
	3	Character	End of TEXT	'CrLf-' – value of x'0D2560'	M

MQ/FTP/NDM Acknowledgment ISO Message Format (ISOACK)					
Position	Length	Format	Field Name	Description	M/O*
			Carriage Return/Line Feed	in EBCDIC	
	1	Character	End of Text Data Block 4	Must contain Value of '{ }'	M

8.4. MQ/FTP/NDM Output ISO Message Format (ISOOUT) created at DTCC for the retrieval of messages.

Each ISO output message starts with 3 header blocks created by the Universal Hub followed by the detail information Message Type information. The detail portion of the record is either:

- Passed by the submitter (Hub functions) to be retrieved by the Recipient
- Created by an Internal DTCC system and the party to retrieve this message are designated by that system.

The format of the Output message is shown below:

MQ/FTP/NDM Output ISO Message Format (ISOOUT) for retrieval from DTCC				
Position	Length	Format	Field Name	Description
Header Block 1- Basic Header				
1	1	Character	Starting Delimiter	Must contain value of '{'
2	2	Character	Block Identifier	Must contain value of '1:'
4	1	Character	Message Identifier	Must contain value of 'F'
5	2	Character	Protocol Identifier	Must contain value of '01'
Recipient's BIC Number or Participant ID				
7	8	Character	Bank/Firm Code	One of the following: 1. Recipient's Bank Identifier code 2. Recipient's Participant ID (If Signon is a Group User this number must be connected in DTCC's Group user eligibility tables)
15	1	Character	Logical terminal	Identifies Terminal Type
16	3	Character	Branch Code	Identifies Branch
19	10	Numeric	Session Number and Sequence number	Assigned by DTCC
29	1	Character	Ending Block Delimiter	Must contain value of '}'
Header Block 2- Application Header				
30	1	Character	Starting Delimiter	Must contain value of '{'
31	2	Character	Block Identifier	Must contain value of '2:'
33	1	Character	Input/Output Identifier	Must contain value of 'O'

MQ/FTP/NDM Output ISO Message Format (ISOOUT) for retrieval from DTCC				
Position	Length	Format	Field Name	Description
34	3	Character	ISO 15022 Message Type	Must contain valid ISO message type
37	4	Character	Time message was accepted at DTCC	Format =HHMM
41	6	Character	Date message was accepted at DTCC	Format=YYMMDD
<i>DTCC Submitter Number</i>				
47	8	Character	DTCC Signon	<p>One of the following based on function definition:</p> <ol style="list-style-type: none"> 1. Submitter's Bank Identifier code 2. Submitter's Participant ID (same number passed to DTCC in the ISOINP message) 3. Value of 'INTDTC'- Internal DTC User <p>Note: Used only when the function used is a one party transaction and the Submitter of the message is an internal DTC application.</p>
55	1	Character	2. Logical terminal	Always spaces
56	3	Character	3. Branch Code	Always spaces
59	4	Numeric	Session Number	<ol style="list-style-type: none"> 1. Value input by submitter based on ISO standards for Hub functions 2. Default value of '0000' if not passed by DTCC application
63	6	Numeric	Sequence number	<ol style="list-style-type: none"> 1. Value input by submitter based on ISO standards for Hub functions 2. Default value of '000000' if not passed by DTCC application
69	6	Character	Date message was sent from DTCC to a	Format=YYMMDD

MQ/FTP/NDM Output ISO Message Format (ISOOUT) for retrieval from DTCC				
Position	Length	Format	Field Name	Description
			Recipient	
75	4	Character	Time message was sent from DTCC to a Recipient	Format =HHMM
79	1	Character	Message Priority	Must contain value of 'N'
80	1	Character	Ending Block Delimiter	Must contain value of '}'
Header Block 3 - User Header				
81	1	Character	Starting Delimiter	Must contain value of '{'
82	2	Character	Block Identifier	Must contain value of '3:'
84	5	Character	Tag for Version number	Must contain value of '{113:'
89	4	Character	Version number	Must contain value of '0301'
93	1	Character	Ending Tag Delimiter	Must contain value of '}'
94	5	Character	Tag for Submitter's Reference Key	Must contain value of '{108:'
99	16	Character	Submitter's Reference Key	Unique key created by Submitter used to identify transaction when receipt is issued for submitter Note: For Functions that do not have an input message this field will contain the DTC Session and sequence number
115	1	Character	Ending Tag Delimiter	Must contain value of '}'
116	5	Character	Tag for Expanded Time	Must contain value of '{115:'
121	11	Character	Expanded time – (Since the block 1 and 2 do not allow for seconds in the time field, this field gives the time down to the second)	Format= 'HH.MM.SS.mm' One of the following: 1. Time when DTCC received message from submitter for HUB functions 2. Time when message was

MQ/FTP/NDM Output ISO Message Format (ISOOUT) for retrieval from DTCC				
Position	Length	Format	Field Name	Description
				created by DTCC application
132	1	Character	Ending Tag Delimiter	Must contain value of ‘}’
133	1	Character	Ending Block Delimiter	Must contain value of ‘}’
<i>Text (Data) Block</i>				
134	1	Character	Starting Delimiter	Must contain value of ‘{’
135	2	Character	Block Identifier	Must contain value of ‘4:’
137	2	Character	Carriage Return/Line Feed	‘CrLf’ – value of x’0D25’ in EBCDIC
137	Length of Record 1-27000)	Data fields		
	3	Character	End of TEXT Carriage Return/Line Feed and dash(-)	‘CrLf-’ – value of x’0D2560’ in EBCDIC
	1	Character	End of Text Data Block 4	Must contain Value of ‘}’

8.5. MQ/FTP/NDM Receipt ISO Message Format (ISORCP)

A Receipt message will be created for a Hub application based on the function profile table. A receipt is generated:

- For MQ users
 1. When DTCC creates the MQ message and puts it on the Recipient's queue.
 2. When the Recipient picks up the message at their location, data will be extracted into a sequential file for each receipt processed.
- For FTP/NDM users – when the job submitted by a user for retrieval completes successfully.

The ISO Receipt record is described below.

MQ/FTP/NDM Receipt (Acknowledgment) ISO Message Format (ICMRCP)					
Position	Length	Format	Field Name	Description	M/O*
1	1	Character	Starting Delimiter	Must contain value of '{'	M
2	2	Character	Block Identifier	Must contain value of '1:'	M
4	1	Character	Message Identifier	Must contain value of 'F'	M
5	2	Character	Protocol Identifier	Must contain value of '01'	M
7	12	Character	Submitter's BIC Number/Participant ID	One of the following: 1. Submitter's Bank Identifier code 2. Submitter's DTCC Participant ID	M
19	10	Numeric	Session Number and Sequence number	Assigned by DTCC	M
29	1	Character	Ending Block Delimiter	Must contain value of '}'	M
30	1	Character	Starting Delimiter	Must contain value of '{'	M
31	2	Character	Block Identifier	Must contain value of '2:'	M
33	1	Character	Input/Output Identifier	Must contain value of 'O'	M

MQ/FTP/NDM Receipt (Acknowledgment) ISO Message Format (ICMRCP)					
Position	Length	Format	Field Name	Description	M/O*
34	3	Character	ISO 15022 Message Type	Value of '533'	M
37	4	Character	Time original input message was accepted at DTCC	Format =HHMM	M
41	6	Character	Date original input message was accepted at DTCC	Format=YYMMDD	M
47	8	Character	Bank/Firm Code/DTCC Participant Number	One of the following: 1. Recipient's Bank Identifier code Recipient's Participant ID	M
55	1	Character	Logical terminal	Identifies Terminal Type	M
56	3	Character	3. Branch Code	Identifies Branch	M
59	4	Numeric	Session Number	Value input by submitter based on ISO standards for Hub functions	M
63	6	Numeric	Sequence number	Value input by submitter based on ISO standards for Hub functions	M
69	6	Character	Date receipt message was sent from DTCC to a Submitter	Format=YYMMDD	M
75	4	Character	Time receipt message was sent from DTCC to a Submitter	Format =HHMM	M
79	1	Character	Message Priority	Must contain value of 'N'	M
80	1	Character	Ending Block Delimiter	Must contain value of '}'	M
81	1	Character	Starting Delimiter	Must contain value of '{'	M
82	2	Character	Block Identifier	Must contain value of '3:'	M
84	5	Character	Tag for Version number	Must contain value of '{113:'	M
89	4	Character	Version number	Must contain value of '0301'	M

MQ/FTP/NDM Receipt (Acknowledgment) ISO Message Format (ICMRCP)					
Position	Length	Format	Field Name	Description	M/O*
93	1	Character	Ending Tag Delimiter	Must contain value of ‘}’	M
94	5	Character	Tag for Submitter’s Reference Key	Must contain value of ‘{108:’	M
99	16	Character	Submitter’s Reference Key	Unique key created by Submitter used to identify transaction when receipt is issued for submitter	M
115	1	Character	Ending Tag Delimiter	Must contain value of ‘}’	M
116	1	Character	Ending Block Delimiter	Must contain value of ‘}’	M
Text (Data) Block					
117	1	Character	Starting Block Delimiter	Must contain value of ‘{’	M
118	2	Character	Block Identifier	Must contain value of ‘4:’	M
120	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
122	9	Character	Start of GENL Block	Value of ‘:16R:GENL’	M
131	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
133	15	Character	DTCC Function Type Tag	Value of ‘:22F:TRAN/DTCY/’	M
148	4	Character	DTCC Function Type	DTCC Function Name (e.g., STL8)	M
152	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
154	11	Character	Message Sent or Received (Picked-up) by Recipient Tag	Value of ‘:22H:REDE//’	M
165	4	Character	Message Sent or Received (Picked-up) by Recipient	Value = ‘DELI’ or ‘RECE’	M
169	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M

MQ/FTP/NDM Receipt (Acknowledgment) ISO Message Format (ICMRCP)					
Position	Length	Format	Field Name	Description	M/O*
171	11	Character	Message Text describing reason for receipt Tag	Value of ‘:70C:REAS//’	M
182	35	Character	Message Text describing reason for receipt	Value is one of following 1. DTCC has sent message to Recipient 2. Recipient has picked up message	M
217	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
M1/M2 belows – means either the M1 tag or M2 tag is used(never both) – but one of them is Mandatory					
219	11	Character	Date/Time Transaction Sent from DTCC to Recipient Tag	Value of ‘:98C:SENS//’	M1*
230	8	Character	Date Transaction Sent from DTCC to Recipient	Format=YYYYMMDD	M1*
238	6	Character	Time Transaction Sent from DTCC to Recipient	Format=HHMMSS	M1*
219	11	Character	Date/Time Transaction was picked up by Recipient Tag	Value of ‘:98C:SENR//’	M2*
230	8	Character	Date Transaction was picked up by Recipient	Format=YYYYMMDD	M2*
238	6	Character	Time Transaction was picked up by Recipient	Format=HHMMSS	M2*
244	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
246	9	Character	End of GENL Block	Value of ‘:16S:GENL’	M
255	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M

MQ/FTP/NDM Receipt (Acknowledgment) ISO Message Format (ICMRCP)					
Position	Length	Format	Field Name	Description	M/O*
257	13	Character	Start of ACCTINFO Block	Value of ‘ :16R:ACCTINFO’	M
270	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
272	15	Character	Submitter of Message Tag	Value of ‘:95R:PTYA/DTCY/’	M
287	8	Character	Submitter of Message DTCC Signon Number	Submitter of Message’s DTCC Signon	M
295	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
297	15	Character	Recipient of Message Tag	Value of ‘:95R:PTYB/DTCY/’	M
312	8	Character	Recipient of Message DTCC Signon Number	Recipient of Message’s DTCC Signon	M
320	2	Character	Carriage Return/Line Feed	CrLf – value of x’0D25’ in EBCDIC	M
322	13	Character	End of ACCTINFO Block	Value of ‘ :16S:ACCTINFO’	M
335	3	Character	End of TEXT Carriage Return/Line Feed	‘CrLf-’ – value of x’0D2560’ in EBCDIC	M
338	1	Character	End of Text Data Block 4	Must contain Value of ‘}’	M

*M/O: M=Mandatory, O=Optional .

9. DTF Hub Participant/User Eligibility File (HUBPT) Formats

A record will be created for every Hub function that the Participant has requested DTCC to process. Therefore there may be many rows for a single Participant.

The process for retrieving the HUBPT file uses the DTF facility and either the FTP or NDM protocols.

Note: For information about the DTF facility, please contact your DTCC PIP representative.

The format of the file is shown below:

Hub Participant/User Eligibility file (HUBPT)				
Position	Length	Format	Field Name	Description
1	8	Character	DTCC Participant number or User	Contains the User ID at DTCC who has elected to participate in a selected function
9	10	Character	Participant/user Description	Name of Participant/User
19	3	Character	Hub Function	3-byte field that defines the Hub function. (ex: 'STL', 'IMS')
22	30	Character	Hub function description	Description of Hub function
52	8	Character	Date file created	Creation Date format = YYYYMMDD
60	21	Character	Filler	For DTCC use only