09/19/2005

Date	Revision
03/28/2005	000001
09/19/2005	000002

Purpose

- To provide a standard tool to help organizations develop file and segment headers in support of passing transactions.

Background

- When exchanging data between sites, there are requirements at each site to provide data accountability, data tracing, data retransmissions, etc. In order to satisfy these requirements, each data group, whether a JANAP message or DDN file, must contain unique identification criteria. It will be impossible for DAASC to provide service to 150,000 customers and satisfy the requirements mentioned above unless standard file identification criteria and a standard file format are implemented.
- The DDN file format is structured to support transactions in the following formats.
    - Defense Logistics Standard System (DLSS) format, a fixed length formats which have been in use since 1965.
    - Variable length transactions.
- The enclosed information describes the DAASC DDN file format necessary to support the (DLSS) data requirements over DDN as exists today as well as variable length transaction requirements.

ESTABLISHING A DAASC DDN ACCOUNT

A prospective DAASC DDN customer is required to submit a 'DAASC DDN Questionnaire' System access Request (SAR), located on DAASC's web pages. Upon receipt of the SAR, DAASC will initiate an internal customer approval process.

Once the approval process and security information is completed, etc. is completed, the DDN customer will be provided a logon identification and password. DAASC will require a logon identification, password and ftp permissions on the DDN customers system. A loopback account will be established at DAASC and the DDN customer will be required to have a successful test with both of the DAASC sites located at Dayton and Tracy. During this test period, the DDN customer will be assisted by the DAASC HELP Desk. When all testing is completed, an implementation date will be coordinated between the DDN customer and DAASC.

After implementation of the DDN customer's account, the DAASC HELP Desk will be the point of contact for any questions or problems concerning the DDN connection.

DDN FILE COMPONENTS

1. A DDN file will consist of a FILE HEADER and one or more DATA SEGMENTS. The format of the data segments depends on whether they contain DLSS, or variable length transactions.

2. A DATA SEGMENT will consist of transactions (not to exceed 9997) preceded by a SEGMENT HEADER (see File and Data Segment Constraints section).

COMPONENT DEFINITIONS

1. The FILE HEADER (FH) is the first record of the file. It's purpose is to uniquely identify the file and provide accountability for the complete file.

2. The SEGMENT HEADER (SH) is the first record of a DLSS or variable length data segment. It's purpose is to uniquely identify a data segment and provides the ability to include multiple data segments within a single file.

3. The DATA SEGMENT is a unique group of transactions preceded by a segment header.

MULTIPLE DATA SEGMENTS

Multiple data segments may be included within a single file as described in the examples below:

1. DLSS transactions and TEXT transactions may be transferred in the same file. The DLSS transaction segment will be preceded by one segment header and the TEXT transaction segment will be preceded by a separate segment header.

2. DLSS MILSTRIP transactions and DLSS MILSBILLS transactions may be transferred in the same file. The DLSS transaction segment will be preceded by a segment header with a Content Identifier of IAZZ, and the billing transaction segment will be preceded by a separate segment header with a Content Identifier of IFBB.

3. DLSS and variable length transactions may be transferred in the same file.

DATA ELEMENT DELIMITERS AND LINE TERMINATORS

1. The delimiter used to separate all data elements within the File Header and Segment Header will be determined by the third (3rd) character in the (FH) record, this allows flexibility in setting the delimiter character.

NOTE ** The character used as the record separator CANNOT be used as the delimiter character, nor can it be used as data within the transmission.

NOTE ** The delimiter character and the record separator character(s) will be shown (for print purposes in this documentation), as an asterisk * and a new line character <nl> respectively.

2. The <nl> characters are required ONLY on the (FH) and (SH) records since they are variable length. DAASC discards the <nl> character(s) upon receipt and deblocks the transactions using the transaction length or transaction separator dependent on whether the transactions within the data segment are fixed length or variable length respectively. DAASC does not append the <nl> character to each transaction when sending a data segment consisting of fixed length transactions. The transaction length and the transaction separator are both specified in the (SH) record. All <nl> characters must be included in the file byte count in the (FH) record.

FILE AND DATA SEGMENT CONSTRAINTS

1. During the discussions to establish DDN connectivity between two sites, the maximum file size, segment size, and file delivery schedule will be negotiated. (see Maximum Input File Size Restrictions)

2. When exchanging files with DAASC, the file naming conventions described later in this document will be followed. Two sites exchanging data not involving DAASC may negotiate other file naming conventions.

3. If a DATA SEGMENT within a file is addressed to DAASC, the number of transactions in the DATA SEGMENT is limited to 9997. Any DATA SEGMENT within a file not addressed to DAASC may be limited to 498 transactions. In either case the maximum segment size is limited to 1 MB.

4. DAASC provides a link between the DDN network and all other networks serviced by DAASC. For example, a DAASC DDN customer can deliver data to DAASC via DDN and have DAASC "switch" the data (based on the Segment Receivers Address (SRA) in the SEGMENT HEADER), to another network for delivery to the ultimate destination. When a DATA SEGMENT is received, DAASC looks at the SRA in each SEGMENT HEADER to determine how to process the DATA SEGMENT. If the DATA SEGMENT is addressed to DAASC, each transaction will be validated and processed through the DAASC routing process according to the service rules for that transaction type. If the DATA SEGMENT is not addressed to DAASC, the entire DATA SEGMENT will be properly formatted for delivery over the network on which the destination resides. Therefore, DATA SEGMENTS delivered to DAASC with a SRA of other than DAASC may be limited to 498 transactions (in case data is required to be in JANAP 128 format for delivery).

FILE AND SEGMENT HEADER CONSTRUCTION

1. The FILE HEADER and SEGMENT HEADER fields may be negotiated as optional when data is being exchanged between 2 sites not involving DAASC. When exchanging data with DAASC, all FILE HEADER and SEGMENT HEADER fields are required except as noted in the sections titled File Header Field Descriptions and Segment Header Field Descriptions.

File Header (FH)

Bytes	Acronym	Meaning	Description	Validation Rules
2	FH	File Header Identifier, FH	Two byte file header identifier, (FH)	must be FH
1	*	File Header Delimiter	The delimiter used within the file header	unique
variable	FBC	File Byte Count	Total byte count of the entire file excluding the file header count. The maximum file size will be negotiated between the originator and receiver	numeric
1	*	delimiter		
variable	FOA	File Originators Address	The IP address, Communications Routing Identifier (COMM R/I), or other negotiated identifier of	

			the site transferring this file. The FOA along with the FSN, FTD, and FTT will uniquely identify this file	
1	*	delimiter		
1-4	FSN	File Serial Number	The serial number of the file (typically will start at 1 at the beginning of each RADAY and be incremented by 1 for each successive file transferred during that day).	numeric
1	*	delimiter		
6	FTD	File Transfer Date(YYMMDD)	The date (in the format YYMMDD), the file was transferred.	MM - (01-12), DD - (01-31), YY - (00-99)
1	*	delimiter		
4	FTT	File Transfer Time	The time (in the format HHMM), the file was transferred. The time will be in Greenwich Mean Time (ZULU).	HH - (00-23), MM - (00-59)
1-2		<nl> or <cr nl>		

Example:

FH*1060*132.159.2.2*10*900128*1500nl

NOTES:

- When exchanging files with DAASC, all fields requiring Alpha characters MUST be in UPPER case.
- The delimiter character follows the File Header (FH) identifier and separates each data element within the File Header except the last element which is followed by the new line <nl> code. When there is no data being transmitted for a defined element, the delimiter is transmitted to preserve the data element count unless the blank elements are the last in the File Header (FH). In that case, transmission of the <nl> code indicates that all non-transmitted elements are blank.
- A file header will vary in length depending on users required data. The file header ends with the line feed and all characters after the line feed are to be included in the byte count.

Segment Header (SH)

Bytes	Acronym	Meaning	Description	Validation Rules
2	SH	Segment Header Identifier, SH	Two byte segment header identifier, (SH).	must be SH
1	*	delimiter	The delimiter used within the segment header	unique
1	TF	Transaction Format	The transaction format is used to indicate whether the transactions are fixed or variable length. An (F) indicates the transactions are fixed length, and a (V) indicates the transactions are variable length.	F or V
1	*	delimiter		
1	TS	Transaction Separator	The transaction separator is used for variable length transactions only. If the transactions are fixed length, no entry is required but the delimiter character (*) must be entered to denote end of field.	unique
1	*	delimiter		
variable	TL	Transaction Length	Actual transaction length in bytes if the transactions are fixed (can be any length ,not necessarily 80), or the maximum transaction length in bytes if the transactions are variable length	numeric
1	*	delimiter		
variable	TC	Transaction Count	The number of transactions in the data segment	numeric
1	*	delimiter		
1	PREC	Precedence	The precedence of the data segment. P - Priority R - Routine O - Immediate	alpha
1	*	delimiter		
4	CI	Content Indicator	Identifies the file content so an automated process can determine if a special processing of the data segment is required. IAZZ - Normal MILSTRIP transactions IFBB - Billing transactions ZYUW -	alpha

			Narrative file ZYVW - Service file etc.	
1	*	delimiter		
variable	SOA	Segment Originators Address	The IP address (Communications Routing Identifier (COMM R/I)), or other negotiated identifier of the site creating the data segment. The COMM R/I is required in this field when exchanging data with DAASC. The SOA along with the SSN, SCD, and SCT will uniquely identify this data segment.	
1	*	delimiter		
variable	SOP	Segment Originators PLA/Office Symbol	The Plain Language Address (PLA) and optional Office Symbol (OFC) of the site creating the data segment. This field would be supplied on narrative type segments only	
1	*	delimiter		
1-4	SSN	Segment Serial Number	The serial number of the data segment (typically will start at 1 the beginning of each RADAY and be incremented by 1 for each successive data segment created during that day).	numeric
1	*	delimiter		
6	SCD	Segment Creation Date	The date (in the format YYMMDD), the data segment was created	MM - (01-12), DD - (01-31), YY - (00-99)
1	*	delimiter		
4	SCT	Segment Creation Time	The time (in the format HHMM), the data segment was created. The time will be in Greenwich Mean Time (ZULU).	HH - (00-23), MM - (00-59)
1	*	delimiter		
variable	SRA	Segment Receivers Address	The IP address (Communications Routing Identifier (COMM R/I)), or other negotiated identifier of the site to receive the data segment. The COMM R/I is required in this field when exchanging data with DAASC	
1	*	delimiter		

variable	SRP	Segment Recievers PLA/Office Symbol	The Plain Language Address (PLA) and optional Office Symbol (OFC) of the site to receive the data segment. This field would be supplied on narrative type segments only	
1-2		<nl> or <cr lf>		

Example:

```
SH*F**80*4*R*IAZZ*RUEOHNJ**14*900128*1430*RUSAZZA**nl
```

NOTES:

- The delimiter character follows the File Header (FH) identifier and separates each data element within the Segment Header (SH) except the last element which is followed by the new line <nl> code. When there is no data being transmitted for a defined element (the element), the delimiter is transmitted to preserve the data element count unless the blank elements are the last in the Segment Header (SH). In that case, transmission of the <nl> code indicates that all non-transmitted elements are blank.
- A segment header will vary in length depending on users required data. The segment header ends with the line feed and all segment headers will be included in the byte count.

SEGMENT EXAMPLES

Single Segment - 100 Fixed length 77 byte DLSS transactions with <nl> characters.

- ```FH*8151*26.16.0.47*1*900128*1535nl```
- ```SH*F**77*100*P*IAZZ*RUEOHNJ**1*900128*1400*RUSAZZA**nl```
- ```DLSS MILSTRIP Transactions ....nl```
- ```.....98 more occurances with...............................nl```
- ```DLSS MILSTRIP Transactions.................................nl```

Single Segment - 100 Fixed length 80 byte DLSS transactions without <nl> characters.

- ```FH*8051*26.16.0.47*1*900128*1535nl```
- ```SH*F**80*100*P*IAZZ*RUSAZZA**1*900128*1400*RUEOHNJ**nl```
- ```DLSS MILSTRIP Transactions...........80 bytes.............>```
- ```.....98 more occurances..............80 bytes.............>```
- ```DLSS MILSTRIP Transactions...........80 bytes.............>```

Single Segment - 498 variable length DATA transactions separated by @, (max. transaction length, 256 bytes)

- ```FH*7010*RUSAZZA*2*900128*1635nl```
- ```SH*V*@*256*498*R*IAZZ*RUSAZZA**1*900128*1600*RUEOHNJ**nl```

- Variable Length Data Transactions@
- transaction....2@
- transaction3@
- transaction 4is a longer transaction@
- .... more occurances@
- Variable Length Data Transactions@

Single Segment - 25 variable length TEXT transactions separated by '\' (backwards-slash) ....(max. transaction length, 80 bytes)

- FH*857*RUSAZZA*1*900128*2312nl
- SH*V**80*25*P*ZYUW*RUSAZZA*ORIGPLAD//OFC//*1*900128*2310*RUEDKFA *DESTPLAD//OFC//nl
- Variable Length Text Transactions\
- Variable Length Text Transactions\

Multiple Segments

- Segment 1, contains 3 fixed length 80 byte DLSS transactions
- Segment 2, contains 50 variable length data transactions separated by @ , max. transaction length, 45 bytes
- Segment 3, contains 25 variable length text transactions separated by \, max. transaction length, 160 bytes

- FH*10754*132.159.2.2*1*900128*0215nl
- **SH*F**80**3*R*IFBB*RUEOHNJ**1*900128*0128*RUSAZZA**nl
- ......
- DLSS Billing Transactions
- ......
- **SH*V*@*45**50*P*IAZZ*RUEOHNJ**3*900128*0140*RUSAZZA**nl
- ......
- Variable Length Data Transactions@
- ......
- **SH*V**160**25*P*ZYUW*RUEOHNJ*ORIGPLAD//OFC//*4*900128*0152*RUSAZZ A*DESTPLAD//OFC//nl
- ......
- Variable Length Text Transactions\

File Structure

- FH*1060*132.159.2.2*10*900128*1500nl
- SH*F**77*4*R*IAZZ*RUEOHNJ**14*900128*1430*RUSAZZA**nl
- MILSTRIP transaction 1
- MILSTRIP transaction 2
- MILSTRIP transaction 3

- `MILSTRIP transaction 4`
- `SH*F**78*8*R*IFBB*RUEOHNJ**15*900128*1452*RUSAZZA**nl`
- `MILSBILLS transaction 1`
- `MILSBILLS transaction 2`
- `MILSBILLS transaction 3`
- `MILSBILLS transaction 4`
- `MILSBILLS transaction 5`
- `MILSBILLS transaction 6`
- `MILSBILLS transaction 7`
- `MILSBILLS transaction 8`

The message header / segment header information is recorded at DAASC and checked for prior receipt. The message / segment will be rejected as duplicate if addressed to DAASC for internal routing. Messages passing through DAASC (pass through) addressed to another DAASC customer are not subjected to the DUPLICATE message check rules.

Segment header information is used to uniquely identify a message received and transmitted by DAASC. It is important that the originator of these segments insure the information is unique (not replicated in subsequent files) to provide all systems the ability to track messages and perform automated DUPLICATE message processing when required.

FILE NAMING CONVENTIONS

In order to provide a method for identifying input files (which have been transferred into the DAASC system for processing), and output files awaiting release from the DAASC system, the file naming technique described below will be implemented at DAASC.

Files Input to DAASC

Whenever a file transfer is initiated into DAASC, the first character of the destination filename should begin with a "Z" (ZEBRA) indicating the file is in the process of being transferred. The name will also contain the DAASC assigned SITEID. The SITEID along with the serial number and RADAY extension will insure the filename is unique. Whenever the file transfer has completed normally, the user must rename the file into the input filename format described below.

Files named with the first letter "I" (India) will be processed immediately. Files beginning with the letter "Z"' (Zebra) will be deleted after 24 hours. Once the file has been processed by DAASC, the prefix will be changed to the letter "P" (Papa) indicating the file was processed. Processed files will remain in the customer directory for a short period until they are removed by a cyclic process.

Maximum Input File Size Restrictions

The single file size sent to DAASC should not exceed 5MB (5,000,000) bytes. The modernized systems at DAASC process the files directly into memory and must impose this restriction in order to preserve system resources. Files exceeding the maximum size may be rejected. Rejected file processing is a manual process and will result in delayed customer notifications..

Input Filename Format

Position Meaning

- pos 1 Input File Identifier (alpha I) as in India
- pos 2 - 5 Users ID
- pos 6 - 8 Serial number - This would begin at 1 each day and be incremented by 1 for each file for that day.
- pos 9 Constant period (.)
- pos 10 - 12 Julian Day

EXAMPLE:

- **Z**QAZZ001.365   file is busy in the process of being delivered to DAASC
- **I**QAZZ001.365   file is ready, and will be processed immediately
- **P**QAZZ001.365   file has been processed by DAASC

Files Output from DAASC

Whenever a file is ready to be released from the DAASC system, the file will be PUSHED to the user by DAASC.

DAASC will support either of two file naming formats when pushing a file into a DDN users host system. The Primary Pushed Output Filename Format will be used unless the DDN users host system does not support it, in that case the Alternate Pushed Output Filename Format will be used. The filename format to be used will be determined during the initial testing when adding a new DDN user. When DAASC begins a file transfer to PUSH a file to a user, the file in the users system will be named with a unique interim filename starting with "C" or "Z". When the file transfer has completed normally, the file will be renamed in the users system to the proper pushed output filename format as described below. The DAAS site identifier (position 1) will identify whether the file was transferred from DAAS Dayton (D) or from DAAS Tracy (T). The filename will be in upper case.

The DAASC file name may be prefixed by a directory, pathname or data set name if required by the user.

Primary Pushed Output Filename Format

- pos 1 DAAS Site Identifier, (D) = Dayton or (T) = Tracy
- pos 2 - 5 Users ID
- pos 6 - 8 Serial number - This would begin at 1 each day and be incremented by 1 for each file.
- pos 9 Constant period (.)
- pos 10 - 12 Julian Day

Example:

- **Z**QAZZ001.365   (Dayton) file is busy in the process of being delivered by DAASC
- **D**QAZZ001.365   (Dayton) file is ready for customer processing
- **C**QAZZ001.365   (Tracy) file is busy in the process of being delivered by DAASC
- **T**QAZZ001.365   (Tracy) file is ready for customer processing

Exception to the three digit sequence number in the filename

If the increment of pos 6 - 8 Serial number causes the number to exceed 999 in the same JULIAN day, the number will be expanded to 4 digits to prevent overwriting an unprocessed file on the users system.

- **D**QAZZ997.364
- **D**QAZZ998.364
- **D**QAZZ999.364
- **D**QAZZ0001.364   ====> the exception
- **D**QAZZ0002.364
- **D**QAZZ001.365   ====> the normal rollover of serial number

Alternate Pushed Output Filename Format

- pos 1 DAAS Site Identifier, (D) = Dayton or (T) = Tracy
- pos 2 - 5 Users ID
- pos 6 - 8 Serial number - This would begin at 1 and be incremented by 1 for each file transferred, through 999, and then reset back to 1.
- pos 9 – 11 Julian Day

Example:

- **Z**QAZZ001365   (Dayton) file is busy in the process of being delivered by DAASC
- **D**QAZZ001365   (Dayton) file is ready for customer processing
- **C**QAZZ001365   (Tracy) file is busy in the process of being delivered by DAASC
- **T**QAZZ001365   (Tracy) file is ready for customer processing

 If a file transfer terminates abnormally, leaving a partial file in the users system, DAASC will attempt to remove that partial file by resending the file with the same file name which will overlay the file on most systems.

 The above applies only to sites supplying one host address and one login id to all DAAS sites. The customer will be responsible for partial file cleanup (file names beginning with "C" or "Z") in any of the following cases:

- The customer has supplied multiple host addresses or
- The customer's system supports multiple versions of identical file names.

** NOTE: A customer should allow ample time for DAAS to rename "C" or "Z" file before attempting to delete it. Please contact DAASC if you need further assistance.

** NOTE: A customer will never process a "C" or "Z" file. These are temporary files which have failed to be renamed or deleted. The should be removed by the system administrator after 24 hours.

09/19/2005

DAASC OPERATIONAL DESCRIPTION

DAASC has two operational sites; one located in Dayton, Ohio and the other located in Tracy, California. Each DDN customer is assigned identical accounts on the DDN host system at both DAASC sites. The DDN host at both sites can be accessed using the same logon id and password.

Files being created and pushed to DAASC by the DDN customer should be alternated between the two DAASC sites in order to balance the processing workload. This will insure that the data will be processed in the most efficient and timely manner once received by DAASC. This also provides an alternate path to DAASC should connectivity to one of the sites be interrupted.

Files to be transferred to DAASC must be created in ASCII stream mode and pushed to DAASC by the DDN customer via FTP. The FTP transfer mode is recommended to be in BINARY mode.

Sample FTP session – where
- XXXX is your assigned Site ID
- nnn is an incremental sequence number
- RRR is the Raday

```
FTP
open DNS or Host IP address
login: YourLogin
password: YourPassword
binary
put yourfile ZXXXXnnn.RRR
quit
```

```
Check the status of the FTP transmission at this point.  If successful,
then
```

```
FTP
open DNS or Host IP address
login: YourLogin
password: YourPassword
rename yourfile IXXXXnnn.RRR
quit
```

The file (IXXXXnnn.RRR) will be immediately processed.

VALIDATION and REJECTS

When DAASC processes a file which has been input by a customer, numerous validations are performed on both the file and the individual segments. Either the entire file or an individual segment can be rejected due to a validation failure. If an entire file is rejected because of a failure in one of the file validations, a service message will be created and sent back to the customer identifying the file and the problem. The file can be corrected and resent to DAASC by the customer with the same filename and without changing any of the identifying criteria in the file header (FH) record or any segment headers contained within the file.

If the file consists of multiple segments, only the segments failing the segment validations will be rejected and all segments passing the validations will be processed on thru DAASC. Therefore, only those rejected segments should be corrected and resubmitted to DAASC, not the entire file.

** NOTE: Service messages sent to the customer from DAASC can be identified by 'ZYVW' or 'ZYUW' in the content indicator (CI) element of the segment header.

DAASC performs duplicate checking on each segment addressed to DAAS. A customer cannot resend a previously rejected segment without changing either the segment serial number (SSN), the segment creation date (SCD), or the segment creation time (SCT) in the segment header (SH) record.

COMMON DDN MESSAGE VALIDATION ERRORS

FILE HEADER DOES NOT BEGIN WITH FH - file header is required to begin with FH. Reference 'FILE AND SEGMENT HEADER CONSTRUCTION' documentation.

SEGMENT HEADER DOES NOT BEGIN WITH SH - segment header is required to begin with SH. Reference 'FILE AND SEGMENT HEADER CONSTRUCTION' documentation.

SEGMENT TRANSACTION COUNT HAS INCORRECT LENGTH - transaction count is invalid. NOTE: This error is often caused by the incorrect positioning of control characters (nl and crlf).

SEGMENT TRANSACTION LENGTH IS INCORRECT - transaction length is invalid.

DDN FILE BYTE COUNT ERROR - byte count in the file header is incorrect. File byte count must be corrected and resubmitted. NOTE: The byte count includes the entire file excluding the number of bytes in the file header and the first line feed.

DDN DOCUMENTS DON'T ADD UP TO HEADER COUNT - DDN document count stated in the segment header does not equal the document count. Correct document count and resend. NOTE: review documents for incorrect format or shifted data.

UNKNOWN DDN SITE ID - denotes a user's site id is incorrect or has not been established in the DAASC tables. Site id must be corrected and the file resubmitted.